



MINISTÉRIO DA CIÊNCIA E TECNOLOGIA
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

INPE-14412-TDI/1129

**UMA ARQUITETURA PARA APOIO E AUTOMAÇÃO DE
PROCESSOS DE GERÊNCIA DE PROJETOS DE SOFTWARE**

Luiz Fernando Borrego Filho

Dissertação de Mestrado do Curso de Pós-Graduação em Computação Aplicada,
orientada pelo Dr. Nilson Sant'Anna, aprovada em 10 de dezembro de 2003.

INPE
São José dos Campos
2007

681.3.06

Borrego Filho, L. F.

Uma arquitetura para apoio e automação processos de gerência de projetos de software/ L. F. Borrego Filho. – São José dos Campos: Instituto Nacional de Pesquisas Espaciais (INPE), 2003.

304 p.; - (INPE-14412-TDI/1129)

1. Gerenciamento de Projetos. 2. Definição de Processos.
3. Modelagem de Processo. 4. Qualidade de Software. 5.
Computação da Aplicada. I.Título.

Aprovado pela Banca Examinadora em
cumprimento a requisito exigido para a
obtenção do Título de **Mestre em**
Computação Aplicada.

Dr. José Demisio Simões da Silva



Presidente/INPE, SJCampos-SP

Dr. Nilson Sant'Anna



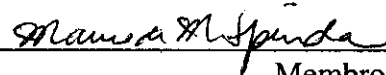
Orientador/INPE, SJCampos-SP

Dr. Germano de Souza Kienbaum



Membro da Banca/INPE, SJCampos-SP

Dr. Mauro de Mesquita Spinola



Membro da Banca
Convidado POLI/USP, São Paulo-SP

Candidato: Luiz Fernando Borrego Filho

São José dos Campos, 10 de dezembro de 2003.

*A meus avós, pelos princípios e lições de vida,
a meus pais, irmã e namorada, pelo apoio.*

AGRADECIMENTOS

Agradeço a Deus, por ter me capacitado e permitido que concluísse mais essa etapa da minha vida.

Agradeço ao meu orientador, pela oportunidade e confiança não somente neste trabalho mas também na minha vida profissional.

Agradeço à SESIS, bem como aos meus companheiros e colegas de trabalho, pelo ambiente, logística e liberação para a realização e implementação desse trabalho.

Agradeço aos meus colegas do curso da CAP e aos colegas do Grupo de Engenharia de Software, pelo convívio e troca de conhecimentos.

RESUMO

Várias pesquisas apontam fracassos em projetos decorrentes de falhas na gerência dos mesmos, evidenciando a área, a qual passou a ser largamente pesquisada e discutida.

Dada a amplitude do tema-gestão de projetos, o foco do trabalho norteia os processos relativos ao gerenciamento do tempo, gerenciamento dos recursos humanos e gerenciamento dos custos, os quais são denominados neste trabalho de processos essenciais de gerenciamento de projetos.

Neste trabalho são analisadas algumas abordagens e modelos de maturidade para processos de gerenciamento de projetos e de desenvolvimento de software, com o intuito de levantar e analisar os requisitos para a sistematização e automação dos processos de gerenciamento de projetos.

Também caberá a esta atividade, usar uma Linguagem de Modelagem de Processo (PML) que tenha maior acessibilidade e facilidade de interpretação pela comunidade de engenharia de software. Uma vez modelados os processos, é proposta uma arquitetura para apoio, sistematização e automação dos processos de gerenciamento de projetos, os quais são analisados e descritos durante o trabalho.

AN ARCHITECTURE FOR SUPPORT AND AUTOMATION OF THE SOFTWARE PROJECT MANAGEMENT PROCESS

ABSTRACT

Several surveys point out that failures in projects are due to failures in the project management, causing this area, to be widely surveyed and discussed.

Since project management is a large research area, this work will focus on the time, human resource and costs management, which are designated herein as the essential project management processes.

Some maturity models and approaches are analyzed for project management processes and software development process, with the intent of setting the requirements for the projects management processes automation and systematization.

This analysis phase makes use of a process modeling language – PML, for easiness of interpretation by the software engineering community. Once these processes are modeled, an architecture for supporting, systematization and automation of the project management process is proposed, being analyzed and described during the dissertation.

SUMÁRIO

LISTA DE FIGURAS

LISTA DE TABELAS

CAPÍTULO 1. INTRODUÇÃO	21
CAPÍTULO 2. GERÊNCIA DE PROJETOS	29
2.1 Conceitos	30
2.1.1 Projeto	30
2.1.2 Gestão de Projetos	32
2.1.3 Processo	33
2.2 Características dos Projetos	34
2.2.1 Fases do Projeto	34
2.2.2 Ciclo de Vida do Projeto.....	35
2.3 Estrutura e Organização dos Processos de Gerenciamento de Projetos	40
2.4 Estimativas no Gerenciamento de Projetos	46
2.4.1 Análise por Pontos de Função - FPA.....	47
2.4.1.1 Estimativa do Tamanho do Software - Metodologia #1	48
2.4.1.2 Estimativa do Tamanho do Software - Metodologia #2	48
2.4.1.3 Estimativa do Prazo do Projeto.....	49
2.4.1.4 Estimativa do Esforço do Projeto.....	50
2.4.1.5 Custo Estimado para o Projeto	50
2.4.1.6 Estimativa do Número de Instruções Fontes	51
2.4.2 Estimativas Utilizando o COCOMO (Constructive Cost Model)	52
2.4.2.1 Modo Orgânico (Convencional).....	53
2.4.2.2 Modo Difuso.....	53
2.4.2.3 Modo Restrito	54
2.4.3 Valor Agregado (<i>Earned Value</i>).....	54
CAPÍTULO 3. LINGUAGENS, ABORDAGENS E NORMAS	61
3.1 Conceitos	61
3.1.1 Processo de Software	61
3.1.2 Melhoria de Processo de Software	63
3.1.3 Modelo de Processo e Linguagem de Modelagem de Processo	64
3.2 SPICE – ISO/IEC 15504	67

3.3 CMMI	89
3.3.1 SW-CMM.....	91
3.3.2 O Modelo <i>Continuos</i>	97
3.3.3 Processos de Gerenciamento de Projetos do CMMI	101
3.4 PMBOK2000	114
3.5 ISO 10006	117
3.6 Unified Process Modelling - UPM	122
3.6.1 Escopo do UPM	124
3.6.2 Elementos Principais.....	124
3.6.3 Terminologia	127
3.6.4 Notação Gráfica	127
3.7 Ambientes de Engenharia de Software - PSEE	129
3.7.1 e-WebProject®.....	140
CAPÍTULO 4 PROCESSOS ESSENCIAIS DE GERENCIAMENTO DE PROJETOS	143
4.1 Gerenciamento do Prazo do Projeto	146
4.1.1 Definição das Atividades.....	148
4.1.1.1 Entradas para a Definição das Atividades	148
4.1.1.2 Ferramentas e Técnicas para a Definição das Atividades	151
4.1.1.3 Saídas da Definição das Atividades	152
4.1.2 Seqüenciamento das Atividades	153
4.1.2.1 Entradas para o Seqüenciamento das Atividades	153
4.1.2.2 Ferramentas e Técnicas para o Seqüenciamento das Atividades.....	155
4.1.2.3 Saídas do Seqüenciamento das Atividades.....	157
4.1.3 Estimativa da Duração das Atividades.....	158
4.1.3.1 Entradas para a Estimativa da Duração das Atividades	159
4.1.3.2 Ferramentas e Técnicas para a Estimativa da Duração das Atividades	162
4.1.3.3 Saídas da Estimativa da Duração das Atividades	163
4.1.4 Desenvolvimento do Cronograma.....	163
4.1.4.1 Entradas para o Desenvolvimento do Cronograma.....	164
4.1.4.2 Ferramentas e Técnicas para o Desenvolvimento do Cronograma.....	168
4.1.4.3 Saídas do Desenvolvimento do Cronograma	171
4.1.5 Controle do Cronograma.....	173
4.1.5.1 Entradas para o Controle do Cronograma.....	173
4.1.5.2 Ferramentas e Técnicas para o Controle do Cronograma.....	174
4.1.5.3 Saídas do Controle do Cronograma	175
4.2 Gerenciamento do Custo do Projeto	176
4.2.1 Planejamento dos Recursos	179

4.2.1.1	Entradas para o Planejamento dos Recursos.....	181
4.2.1.2	Ferramentas e Técnicas para o Planejamento dos Recursos	182
4.2.1.3	Saídas do Planejamento dos Recursos.....	182
4.2.2	Estimativa dos Custos.....	183
4.2.2.1	Entradas para a Estimativa dos Custos.....	183
4.2.2.2	Ferramentas e Técnicas para a Estimativa dos Custos.....	185
4.2.2.3	Saídas da Estimativa dos Custos	187
4.2.3	Orçamentação dos Custos.....	189
4.2.3.1	Entradas para a Orçamentação dos Custos.....	189
4.2.3.2	Ferramentas e Técnicas para a Orçamentação dos Custos.....	190
4.2.3.3	Saídas da Orçamentação dos Custos	190
4.2.4	Controle dos Custos.....	191
4.2.4.1	Entradas para o Controle dos Custos.....	191
4.2.4.2	Ferramentas e Técnicas para o Controle dos Custos.....	193
4.2.4.3	Saídas do Controle dos Custos.....	194
4.3	Gerenciamento dos Recursos Humanos do Projeto.....	196
4.3.1	Planejamento Organizacional	199
4.3.1.1	Entradas para o Planejamento Organizacional.....	200
4.3.1.2	Ferramentas para o Planejamento Organizacional.....	202
4.3.1.3	Saídas do Planejamento Organizacional.....	203
4.3.2	Montagem da Equipe.....	205
4.3.2.1	Entradas Para a Montagem da Equipe.....	205
4.3.2.2	Ferramentas e Técnicas para a Montagem da Equipe.....	207
4.3.2.3	Saídas da Montagem da Equipe.....	208
4.3.3	Desenvolvimento da Equipe	208
4.3.3.1	Entradas Para o Desenvolvimento da Equipe	209
4.3.3.2	Ferramentas e Técnicas Para o Desenvolvimento da Equipe.....	210
4.3.3.3	Saídas do Desenvolvimento da Equipe.....	212
4.4	Organização e Estrutura dos Processos.....	213
4.4.1	Processos de Iniciação	214
4.4.2	Processos de Planejamento.....	215
4.4.3	Processos de Execução.....	220
4.4.4	Processos de Controle.....	223
4.4.5	Processos de Encerramento.....	225
CAPÍTULO 5	ARQUITETURA DE UMA APLICAÇÃO PARA SUPORTE E AUTOMAÇÃO DE PROCESSOS DE GERÊNCIA DE PROJETOS.....	227
5.1	Histórico	227
5.2	Análise dos Requisitos.....	230
5.2.1	Definir Infra-Estrutura Organizacional.....	232
5.2.1.1	Caso de Uso Definir Pessoas.....	232

5.2.1.2	Caso de Uso Definir Equipes.....	232
5.2.1.3	Caso de Uso Definir Recursos de Infra-Estrutura.....	233
5.2.1.4	Caso de Uso Alocar Pessoas às Equipes.....	234
5.2.1.5	Caso de Uso Alocar Pessoas aos Projetos.....	234
5.2.1.6	Caso de Uso Definir Capacitação dos Recursos.....	234
5.2.1.7	Caso de Uso Definir Parâmetros de Contratação e Custos.....	234
5.2.1.8	Caso de Uso Definir Papel.....	234
5.2.1.9	Caso de Uso Definir Permissões de Acesso aos Papéis.....	235
5.2.2	Definir Modelos de Processos.....	235
5.2.3	Planejar Projeto.....	236
5.2.3.1	Caso de Uso Definir Atividades.....	236
5.2.3.2	Caso de Uso Seqüenciar Atividades.....	237
5.2.3.3	Caso de Uso Estimar Tempos das Atividades.....	237
5.2.3.4	Caso de Uso Desenvolver Cronograma.....	237
5.2.3.5	Caso de Uso Definir Artefatos de Entrada e Saída das Atividades	237
5.2.3.6	Caso de Uso Alocar Recursos Humanos.....	238
5.2.3.7	Caso de Uso Alocar Recursos de Infra-Estrutura.....	238
5.2.3.8	Caso de Uso Estabelecer <i>Baselines</i>	238
5.2.3.9	Caso de Uso Importar Modelos de Processos.....	238
5.2.3.10	Caso de Uso Definir Métricas.....	239
5.2.4	Monitorar Projetos.....	239
5.2.4.1	Caso de Uso Monitorar <i>Baselines</i>	239
5.2.4.2	Caso de Uso Analisar Esforços em Tarefas.....	240
5.2.4.3	Caso de Uso Definir Agentes.....	240
5.2.4.4	Caso de Uso Apontar Tempos.....	241
5.3	Proposta de Arquitetura.....	241
5.3.1	Aplicação.....	241
5.3.1.1	Ferramenta de Infra-Estrutura Organizacional.....	244
5.3.1.2	Ferramenta de Modelagem de Processos.....	245
5.3.1.3	Ferramenta para Planejamento de Projetos.....	246
5.3.1.4	Ferramenta para Monitoramento de Projetos.....	248
5.3.1.5	Serviço de Controle de Acesso.....	249
5.3.1.6	Estimador.....	250
5.3.1.7	Serviço de Apoio à Geração do Cronograma de Projeto.....	252
5.3.1.8	Serviço Otimizador de Cronograma.....	252
5.3.1.9	Agentes Pró-Ativos.....	253
5.3.1.10	Sistema de Análise e Extração de Informações.....	254
5.3.2	Repositório.....	254
CAPÍTULO 6	CONCLUSÃO.....	257
6.1	Resultados.....	257
6.2	Considerações Finais.....	270

6.3 Trabalhos Futuros	272
REFERÊNCIAS BIBLIOGRÁFICAS	275
APÊNDICE A APLICAÇÃO PARA PLANEJAMENTO E ACOMPANHAMENTO DE PROJETOS INTEGRADA AO PSEE E-WEBPROJECT®	291

LISTA DE FIGURAS

1.1 - Comparação das Pesquisas de 1999 e de 2001 do MCT	25
2.1 - Relacionamento da Gerência de Projetos com Outras Disciplinas da Administração	32
2.2 - Modelo do ciclo de vida codifica-remenda.....	36
2.3 - Modelo do ciclo de vida cascata.....	37
2.4 - Modelo de ciclo de vida sashimi.....	37
2.5 - Modelo de ciclo de vida espiral	38
2.6 - Modelo de ciclo de vida prototipagem evolutiva	39
2.7 - Modelo do ciclo de vida entrega por estágios	39
2.8 - Modelo do ciclo de vida entrega evolutiva.....	40
2.9 - Relacionamento entre os grupos de processos em cada fase	42
2.10 - Sobreposição dos grupos de processos em cada fase	42
2.11 - Interações entre as fases	43
2.12 - Gráfico completo da projeção do VA.....	59
3.1 – Caracterização do processo de software.....	68
3.2- Modelo de avaliação do processo	74
3.3 - Níveis de modelagem definidos pela OMG	123
3.4 - Pacote de elementos básicos.....	125
3.5 - O PSEE centraliza e integra o apoio para engenharia do processo, gerência de projetos e engenharia de software	132
3.6 - Componentes e camada conceitual do ambiente.....	141
3.7 - Arquitetura física do ambiente.....	142
4.1 - Processos essenciais de gerenciamento de projetos.....	146
4.2 - Processo de gerenciamento do tempo.....	147
4.3 - Modelo do processo definição das atividades	149
4.4 - Modelo do processo de seqüenciamento de atividades	154
4.5 - Modelo do processo estimativa da duração das atividades	159
4.6 - Modelo do processo desenvolvimento do cronograma	164
4.7 - Modelo do processo controle do cronograma.....	173
4.8 - Processos de gerenciamento dos custos do projeto	177
4.9 - Modelo do processo de planejamento dos recursos	180
4.10 - Modelo do processo estimativa dos custos	184
4.11 - Modelo do processo de orçamentação dos custos	189
4.12 - Modelo de processo de controle de custos	192
4.13 - Processos de recursos humanos	196
4.14 - Modelo de processo planejamento organizacional.....	200
4.15 - Modelo de processos montagem da equipe.....	206
4.16 - Modelo do processo de desenvolvimento da equipe.....	209
4.17 - Modelo do grupo de processo iniciação	215
4.18 - Modelo dos processos principais do grupo de planejamento	219
4.19 - Modelo dos processos facilitadores do grupo de planejamento	221
4.20 - Modelo dos processos principais do grupo de execução	221
4.21 - Modelo dos processos facilitadores do grupo de execução	222
4.22 - Modelo dos processos principais do grupo de controle.....	224

4.23 - Modelo dos processos facilitadores do grupo de controle.....	224
4.24 - Modelo dos processos do grupo de encerramento	226
5.1 - Diagrama de contexto	231
5.2 - Diagrama de caso de uso da definição de infra-estrutura organizacional.....	233
5.3 - Caso de uso definir modelos de processos.....	235
5.4 - Diagrama de casos de uso planejar projeto	236
5.5 - Diagrama de caso de uso do monitorar projeto.....	240
5.6 - Arquitetura proposta.....	242
A.1 - Arquitetura proposta	291
A.2 - Manutenção de recursos humanos.....	292
A.3 - Manutenção das equipes organizacionais	293
A.4 - Manutenção dos projetos.....	293
A.5 - Interface para planejamento do projeto	294
A.6 - Manutenção das sub-tarefas.....	295
A.7 - Manutenção das tarefas predecessoras	295
A.8 - Alocação dos recursos de infra-estrutura na tarefa	296
A.9 - Alocação dos recursos humanos na tarefa.....	296
A.10 - Manutenção das <i>constraints</i> da tarefa	297
A.11 - Diagrama de <i>Gantt</i>	297
A.12 - Diagrama de <i>Gantt</i> com respectiva alocação de pessoas	298
A.13 - Comparativo entre o tempo planejado e o realizado.....	299
A.14 - Time-sheet.....	299
A.15 - Interface para o controle dos custos	300
A.16 - Serviço para controle de acesso.....	301
A.17 - <i>Desktop</i> pessoal do usuário.....	301
A.18 - Lista das tarefas instanciadas ao usuário	302
A.19 - Interface de tarefas disponíveis para instanciar.....	303
A.20 - Interface de parte do processo de geração de cronograma.....	303

LISTA DE TABELAS

1.1 - Principais problemas de extrapolação de prazos e custos estimados	22
1.2 - Principais fatores organizacionais que tornam gerentes de projetos ineficazes	23
1.3 - Comparação das pesquisas de 1999 e de 2001 do MCT.....	24
3.1 - Modelo de processos (SPICE, 1995)	68
3.2 - Níveis de capacitação FONTE: (SPICE, 1995)	73
3.3 - Práticas genéricas para o nível 1: Executado Informalmente.....	75
3.4 - Práticas genéricas para o nível 2: Planejado e Monitorado.....	75
3.5 - Práticas genéricas para o nível 3: Bem Definido (Estabelecido).....	77
3.6 - Práticas genéricas para o nível 4: Controlado e Quantificado.....	78
3.7 - Práticas genéricas para o nível 5: Melhoramento Contínuo.....	78
3.8 - Práticas genéricas.....	98
3.9 - Práticas específicas para planejamento.....	100
3.10 - Estereótipos propostos para gráficos. FONTE: (OMG 2000)	128
4.1 - PMBOK2000 x CMMI x SPICE.....	144
4.2 - Sigla das áreas de conhecimento	214
5.1 - Abordagens e processos analisados.....	230
5.2 - Atores X Casos de Uso X Funcionalidades.....	243
6.1 - Relacionamento das funcionalidades com o PMBOK2000	258
6.2 - Relacionamento das funcionalidades com o CMMI.....	261
6.3 - Relacionamento das funcionalidades com o SPICE	267

CAPÍTULO 1

INTRODUÇÃO

“Todos executam projetos em seu dia-a-dia, a maior parte deles de maneira inconsciente” (Frame, 1995).

Apesar do evidente aumento da utilização do gerenciamento de projetos por parte das empresas, ainda se observam altos índices de falhas em projetos (Rad e Raghavan, 2000) (Forman, 2000).

Sant’Anna (2000) complementa apontando que boa parte dos projetos de software não atingem seus objetivos, como: custos, prazos, satisfação do cliente; ou seja, os projetos fracassam por falta de um bom gerenciamento de projetos.

Num estudo mundial realizado pela Standish Group (Bottoni, 2001), foi revelado que um grande número de projetos de software falha em suas estimativas, não cumprindo prazos e custos.

Esta pesquisa mostra que 31% dos projetos de Tecnologia de Informação - TI são cancelados antes de serem concluídos, gerando um prejuízo de aproximadamente 81 bilhões de dólares por ano. Outros 53% dos projetos ultrapassam em quase 90% as estimativas de custo e prazo planejados, representando um prejuízo de aproximadamente 59 bilhões de dólares, enquanto que apenas 16% dos projetos de TI iniciados são completados dentro do prazo e orçamento previstos.

Já o levantamento realizado nas empresas e instituições norte-americanas (Zimmerer; Yasin, 1998) revela que a principal razão para os projetos terem problemas de extrapolação de prazos e custos estimados é a falha em utilizar ferramentas de gerenciamento de projeto de forma sistemática, como mostra a Tabela 1.1.

TABELA 1.1 - Principais Problemas de Extrapolação de Prazos e Custos Estimados.

Razões	Porcentagem
Falha em utilizar ferramentas de gerenciamento de projeto sistematicamente	39,5%
Liderança fraca por parte do gerente de projeto	32,9%
Resposta lenta às solicitações de clientes	9,2%
Falta de decisões e medidas corretivas tomadas em tempo hábil	5,2%
Falta de comunicação interorganizacional	3,9%

FONTE: Zimmerer; Yasin (1998).

Este mesmo levantamento também revela que o principal fator organizacional que contribui para a ineficiência dos gerentes de projetos é a falta de comprometimento e suporte da alta gerência, como pode ser observado através da Tabela 1.2.

Em De Marco (1989), na “Anatomia do Fracasso do Projeto”, o autor relata a sua experiência em auditorias de projetos de software, e menciona que alguns projetos fracassaram por falta de competência do gerente.

Mas, em outros casos em que os gerentes superavam as características associadas a bons gerentes, seus projetos também fracassaram. Segundo o autor, a ocorrência do fracasso não foi devido ao fato da equipe cometer muitos erros ou projetarem mal, simplesmente, não conseguiram atender as expectativas originais, ou seja, a culpa nesses casos é de “expectativas presunçosas e irracionais”. Pressman (1995) complementa observando, com horror, como os gerentes lutam futilmente com projetos assustadores, contorcendo-se sob prazos finais impossíveis.

TABELA 1.2 - Principais fatores organizacionais que tornam gerentes de projetos ineficazes.

Fatores Organizacionais	Porcentagem
Falta de comprometimento e apoio da alta gerência (organizacional)	31,5%
Resistência à mudança	18,4%
Sistema de recompensa inconsistente	13,2%
A organização reage a eventos e não a um planejamento antecipativo	9,2%
Falta de recursos	7,9%

FONTE: Zimmerer; Yasin (1998).

Para DeMarco (1989), “não se pode controlar o que não se pode medir”, portanto, o uso de métricas no processo de desenvolvimento de software pode trazer benefícios expressivos, principalmente nas estimativas de tempo e custo, pois erros nas estimativas de tempo e custo são a principal causa de insucesso no gerenciamento de projetos de software.

Além das métricas, o uso dos dados históricos e das informações coletadas a partir das métricas aplicadas pode auxiliar a tomada de decisões no gerenciamento de projetos (DeMarco, 1989) (Paulk et al., 1993) (Pressman, 1995) (SPICE, 1995) (PMI, 2000) (CMMI, 2001) (Vasques, 2002) (Borrego Filho et al., 2002a) (Borrego Filho et al., 2002b).

Por outro lado, as organizações modernas, na busca por maior competitividade, adotam abordagens de melhoria de processos para melhorar a qualidade e a confiabilidade de seus produtos e reduzir custos de desenvolvimento (Sant’Anna et al., 2002).

O crescente interesse das organizações de TI em conhecer e adotar essas abordagens pode ser analisado através dos Relatórios de Qualidade e Produtividade no Setor de Software Brasileiro de 1999 (SPIA, 2000) e de 2001 (SPI, 2002), que são editados bienalmente pela Secretaria de Política de Informática do Ministério da Ciência e Tecnologia - MCT. Os resultados desses dois relatórios são comparados na Tabela 1.3 e na Figura 1.1.

TABELA 1.3 - Comparação das pesquisas de 1999 e de 2001 do MCT.

<i>Categoria</i>	ISO/IEC 12207		CMM		SPICE		ISO 9000
	<i>1999</i>	<i>2001</i>	<i>1999</i>	<i>2001</i>	<i>1999</i>	<i>2001</i>	<i>2001</i>
Conhece e usa sistematicamente	16	16	1	4	8	16	81
Conhece e começa a usar	53	34	14	13	36	71	62
Conhece, mas não usa	121	226	121	232	165	223	219
Não conhece	255	134	308	160	234	105	56
Total de empresas pesquisadas	445	410	444	409	443	415	418

FONTE: SPIA (2000) e SPI (2002).

Mas, alguns relatos de experiência em grandes projetos de software (Cunha; Nakanishi, 1993) (Sant'Anna et al., 1998), revelam que são necessários outros fatores, além da adoção de abordagens, como o *Capability Maturity Model - CMM* (Paulk et al., 1993) ou o *Software Process Improvement and Capability Determination - SPICE / ISO-15504* (SPICE, 1995). Os projetos necessitam de estruturação em termos organizacionais, com arranjos de equipes, com políticas de responsabilidades e com estruturas otimizadas para troca de informações e trâmite de produtos.

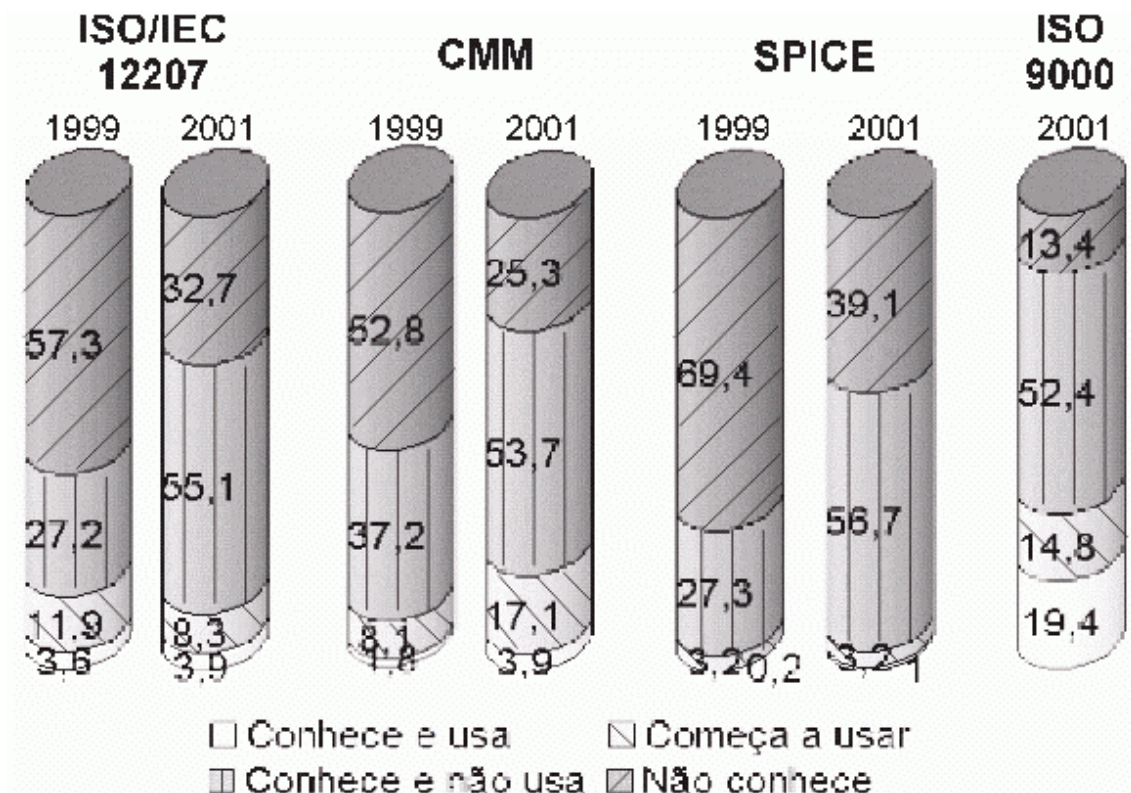


FIGURA 1.1 - Comparação das Pesquisas de 1999 e de 2001 do MCT.

Além disso, o caos no gerenciamento de projetos de software impede que o uso de técnicas de engenharia de software traga os benefícios e resultados esperados (Belloquim, 2002).

Crawford (2000) reitera, justificando que a maioria dos problemas relativos a projetos ocorrem devido à falta de processos adequados e padronizados de gerenciamento, ou seja, a maior causa das falhas nos projetos não são as especificidades do que efetivamente deu errado, mas sim a falta de procedimentos, metodologias e padrões.

Nesse sentido, iniciativas recentes como o *Project Management Office* (PMO) (Goodpasture, 2000) (Rad e Raghavan, 2000), ou Escritório de Gerenciamento de Projetos - EGP, surgem como um elemento organizacional responsável pelo apoio à minimização dos problemas de falta de processos bem definidos e padronizados, isto ocorre através da divulgação das práticas de gerenciamento

de projetos na organização e pelo fornecimento da infra-estrutura necessária para um gerenciamento corporativo de projetos.

Mas alguns estudos, apresentados por Dinsmore (2002), revelam que os Escritórios de Gerenciamento de Projetos podem não ser uma solução viável para grande parte das empresas e sua evidência nos dias de hoje pode ser fruto de um modismo passageiro.

O *Project Management Institute* (PMI), por sua vez, é uma instituição dedicada ao progresso e disseminação das melhores práticas da atividade de Gestão de Projetos, e define uma abordagem para os processos de gerenciamento de projetos através do livro *Project Management Body of Knowledge* (PMBOK) (PMI, 2000).

Porém, a implementação de processos de forma manual ou semi-automatizada não traz benefícios expressivos, se comparados ao potencial da automação de processos por ambientes de engenharia de software (Sant'Anna et al., 2002).

Uma área da engenharia de software em ascensão e fundamental para a automação de processos de software é a de ambientes de desenvolvimento de software, atualmente chamados de Ambientes de Engenharia de Software Centrado em Processos. Proveniente da língua Inglesa, estes ambientes são chamados originalmente de *Process Centered Software Engineering Environments* (PSEE) (Christie, 1993) (Ben-Shaul; Kaiser, 1995) (Ambriola et al., 1997) (Fuggetta, 2000).

Trabalhos estão sendo realizados pela comunidade na área de ambientes de engenharia de software, principalmente no suporte e automação de processos (Ambriola et al., 1997), sejam eles de desenvolvimento ou de gestão (Sant'Anna, 2000).

“Por todos esses motivos de insucesso em projetos, são crescentes os trabalhos e estudos na arte da gerência e gestão de projetos” (Borrego Filho et al., 2002 a) (Borrego Filho et al., 2002 b).

Rad e Raghavan (2000) afirmam que o gerenciamento de projetos é uma das disciplinas que mais cresce em praticamente todas as indústrias de hoje. Seus principais objetivos visam manter um balanço lógico e eficiente entre custo, prazo, qualidade e escopo dos objetivos ou produtos do projeto.

Belzer (2001) acrescenta, mostrando que o aumento da complexidade do mundo dos negócios faz com que as empresas necessitem de uma maior capacidade de coordenar, gerenciar e controlar suas atividades de maneira a responder mais rapidamente aos estímulos externos. Essa coordenação e controle de atividades são o foco do gerenciamento de projetos e estão intimamente ligados ao sucesso da implementação de estratégias de negócios por meio de projetos.

Pelos vários motivos e causas de insucesso apresentados anteriormente, optou-se por dar continuidade ao trabalho de pesquisa “Um Ambiente Integrado para o Apoio ao Desenvolvimento e Gestão de Projetos de Software para Sistemas de Controle de Satélite (AMBGES)” (Sant’Anna, 2000), uma vez que nessa tese, o tema apresentado foi proposto como trabalho futuro de pesquisa.

Dessa forma, o objetivo desse trabalho será: a) analisar os modelos de processo e de maturidade para a gerência de projeto; b) interpretar os modelos de processos e apresentá-los de forma transparente para a comunidade de engenharia de software, através de uma Linguagem de Modelagem de Processos (PML) acessível; c) propor uma arquitetura de uma aplicação capaz de suportar os requisitos e a automação dos processos de interesse a esse trabalho e d) integrar a implementação dessa arquitetura a um PSEE.

O trabalho está organizado da seguinte forma, o Capítulo 2, Gerência de Projetos, mostra uma visão ampla da área gerência de projetos e apresenta os

principais conceitos, características, estrutura dos processos e estimativas para o gerenciamento de projetos. O Capítulo 3, Linguagens Abordagens e Normas, apresenta os principais modelos de qualidade, modelos de maturidade, modelos de processo, linguagens de modelagem de processo e os conceitos que norteiam esses elementos e que são utilizados ao longo do trabalho. O Capítulo 4, Processos Essenciais de Gerenciamento de Projetos, descreve e mostra a modelagem dos processos de gerenciamento de projetos que fazem parte do escopo desse trabalho. O Capítulo 5, Arquitetura de uma Aplicação para Suporte e Automação de Processos de Gerência de Projetos, apresenta a modelagem e a proposta de uma arquitetura que suporte e automatize os processos analisados e modelados nesse trabalho. O Capítulo 6, Conclusão, mostra os resultados, as conclusões obtidas a partir dos resultados do trabalho e os trabalhos futuros.

CAPÍTULO 2

GERÊNCIA DE PROJETOS

Segundo Frame (1995), as pessoas realizam projetos desde os primórdios da civilização. As caçadas organizadas por nossos ancestrais pré-históricos foram projetos, bem como a construção das pirâmides e da Grande Muralha da China, que guardadas as devidas proporções com relação à época em que foram realizados, podem ser comparados a grandes projetos mais atuais como o Projeto Manhattan, que construiu a primeira bomba atômica, ou Projeto Apollo, que permitiu ao homem chegar à Lua.

Ainda de acordo com Frame (1995), todos executam projetos em seu dia-a-dia, a maior parte deles de maneira inconsciente, isto vem acontecendo, rotineiramente, há mais de dez mil anos. Porém, o gerenciamento de projetos como uma disciplina é um desenvolvimento recente e teve seu início como um subproduto dos esforços da Segunda Guerra Mundial com o Projeto Manhattan, o qual envolveu um grande orçamento, prazos curtos e complexidade de recursos. Nos últimos anos, o gerenciamento de projetos vem se tornando cada vez mais estudado e utilizado.

A maior parte do conhecimento necessário para gerenciar projetos é quase específico da disciplina gerência de projetos, no entanto, o conhecimento e práticas de outras disciplinas, como a administração geral e as disciplinas das áreas de aplicação são indispensáveis para o gerenciamento de projetos (PMI, 2000). Uma visão do relacionamento e da sobreposição, não proporcional, das disciplinas é mostrada na Figura 2.1.

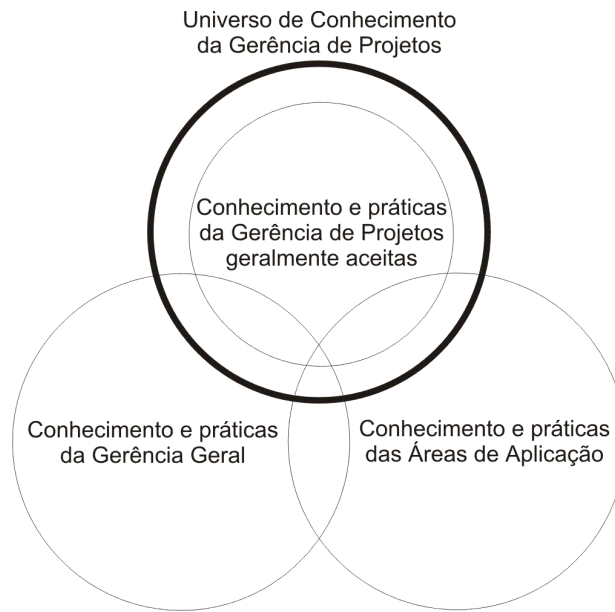


FIGURA 2.1 - Relacionamento da gerência de projetos com outras disciplinas da administração.

FONTE: PMI (2000).

2.1 Conceitos

Alguns autores divergem em suas definições, no entanto, na maioria dos casos as divergências estão apenas nos termos usados nas definições.

2.1.1 Projeto

Lewis (2000) define projeto como um trabalho único que possui início e fim claramente definidos, um escopo de trabalho especificado, um orçamento e um nível de performance a ser atingido. Porém, para que um trabalho possa ser considerado um projeto, este precisa ter mais de uma tarefa associada, ou seja, trabalhos constituídos de uma única tarefa não são considerados projetos.

Goodpasture (2000) segue a mesma linha, definindo projeto como o conjunto de tarefas únicas, interdependentes e não repetitivas, planejadas e executadas de forma a produzir algum resultado.

Para Nicholas (1990), projeto pode ser definido em termos de propósito, estrutura organizacional, complexidade, interesse e ciclo de vida.

Meredith e Mantel (1985) definem projeto com um conjunto de tarefas que podem ser divididas em sub-tarefas que precisam ser executadas para se alcançar os objetivos. Tal projeto é complexo o suficiente para que a execução dessas sub-tarefas requeira uma coordenação cuidadosa e controle em termos de duração, precedência, custo e performance. O próprio projeto precisa ser coordenado perante outros projetos executados concomitantemente.

Nos modelos do *Capability Maturity Model Integration* (CMMI), um projeto é um conjunto gerenciado de recursos inter-relacionados que fornece um ou mais produtos a um cliente ou usuário final. Este conjunto de recursos, tipicamente, opera de acordo com um plano que tem um começo e um fim bem definidos. Um plano é freqüentemente documentado e define: o produto a ser entregue ou implementado, os recursos e custos necessários, o trabalho a ser feito, e um cronograma para realizar o trabalho. Um projeto pode ser composto por outros projetos (CMMI, 2001).

Já o PMI (2000), traz uma definição mais completa e abrangente, que contempla as definições anteriores. Para esse instituto, projeto é um empreendimento temporário com o objetivo de criar um produto ou serviço único. Temporário, porque cada projeto tem um começo e um fim bem definidos. Único significa que o produto ou serviço produzido é de alguma forma diferente de todos os outros produtos ou serviços semelhantes. Para muitas organizações, projetos são o meio de responder a requisitos que não podem ser atendidos através dos limites normais de operação da organização.

Ainda segundo o PMI (2000), os projetos são desenvolvidos em todos os níveis da organização. Eles podem envolver uma única pessoa ou milhares delas. Podem durar poucas semanas ou vários anos. Os projetos podem envolver uma unidade isolada da organização ou atravessar as fronteiras organizacionais. Os projetos são críticos para a realização da estratégia de

negócios da organização porque projetos são os meios pelos quais as estratégias são implementadas.

2.1.2 Gestão de Projetos

Apesar do rotineiro envolvimento das pessoas em projetos desde os primórdios da civilização, a natureza destes projetos mudou (Nicholas, 1990). Os projetos modernos envolvem grande complexidade técnica e requerem uma alta diversidade de habilidades. Para lidar com esta nova e complexa natureza das atividades ligadas aos projetos modernos e com a incerteza inerente a essa complexidade, novas formas de gestão se desenvolveram. A moderna administração, ou gestão de projetos é uma delas.

O gerenciamento de projetos, como uma área distinta de prática gerencial, é relativamente novo e seus métodos não são muito conhecidos por grande parte dos gerentes (Nicholas, 1990). Seu início data da década de 50, com os militares americanos, mas somente no final dos anos 80 começou a se espalhar fora da esfera militar.

A gestão de projetos proporciona às empresas ferramentas poderosas que melhoram a habilidade da organização para planejar, organizar, executar e controlar as atividades de maneira a conseguir atingir os resultados esperados dentro do prazo e custo previstos, mesmo em projetos de grande complexidade (Meredith; Mantel, 1985).

Para Lewis (2000), o gerenciamento de projetos consiste no planejamento, programação e controle das atividades que precisam ser executadas para que os objetivos do projeto sejam atingidos.

Frame (1995) afirma que a gestão de projetos também está baseada em muitos dos princípios da administração geral; por isso, também envolve negociação, solução de problemas, política, comunicação, liderança e estudo de estrutura organizacional.

Nicholas (1990) complementa afirmando que as soluções para problemas impostos por demandas que mudam rapidamente e por tecnologias complexas precisam ser de alguma forma adaptáveis às novas condições. Como resposta a essas demandas, novas abordagens de gerenciamento surgiram adotando a abordagem sistêmica ou por processo. Essas teorias utilizam conceitos gerais de sistemas e análise de sistemas na tarefa de gerenciamento.

A elaboração dos processos de gerenciamento de projetos é crucial para a boa gestão dos projetos. Para Crawford (2000) a grande maioria dos fracassos em gestão de projetos se deve à falta de processos ou processos mal formulados.

Rad e Raghavan (2000) acentuam a importância dos processos no gerenciamento de projetos afirmando que, no passado, o foco da gestão de projetos estava em alocar pessoal competente para assegurar o sucesso do projeto; apesar dessa abordagem ser necessária, o pensamento atual diz que procedimentos, processos, políticas e ferramentas mais formalizadas são vitais para o planejamento e gerenciamento dos projetos.

Para o PMI (2000), gerência de projetos é a aplicação de conhecimentos, habilidades, ferramentas e técnicas para projetar atividades que visem atingir os requisitos do projeto.

2.1.3 Processo

A *Institute of Electrical and Electronics Engineers* (IEEE) define processo como a seqüência de passos realizados para atingir um propósito (IEEE, 1991).

Na ISO 9000 (ABNT, 2000b), processo é definido como um conjunto de atividades inter-relacionadas ou interativas que transformam insumos (entradas) em produtos (saídas).

O PMI (2000) define processo como uma série de ações que geram um resultado. Os projetos são compostos por processos, e esses, são realizados por pessoas, e normalmente se enquadram em uma das duas categorias:

- **Processos da gerência de projetos** se relacionam com a descrição, a organização e a conclusão do trabalho do projeto;
- **Processos orientados ao produto** se relacionam com a especificação e a criação do produto do projeto.

2.2 Características dos Projetos

Como os projetos possuem um caráter único, a eles está associado um certo grau de incerteza. As organizações que desenvolvem projetos usualmente os divide em várias fases, visando um melhor controle gerencial e uma ligação mais adequada de cada projeto aos seus processos. O conjunto de fases de um projeto é conhecido como ciclo de vida do projeto.

2.2.1 Fases do Projeto

Cada fase do projeto é marcada pela conclusão de um ou mais produtos da fase (*deliverables*). Um subproduto é um resultado de trabalho (*work product*) tangível e verificável. Os subprodutos do projeto e também as fases compõem uma seqüência lógica, criada para assegurar uma adequada definição do produto do projeto.

A conclusão de uma fase é geralmente marcada pela revisão dos principais subprodutos e pela avaliação do desempenho do projeto, tendo em vista: determinar se o projeto deve continuar na sua próxima fase e detectar e corrigir erros a um custo aceitável. Estas revisões de fim de fase são comumente denominadas saídas de fase (*phase exits*), passagens de estágio (*stage gates*) ou pontos de término (*kill points*).

Cada fase normalmente inclui um conjunto de subprodutos específicos projetados com o objetivo de estabelecer um controle gerencial desejado. A maioria desses itens está relacionada com o principal subproduto da fase. As fases, tipicamente, adotam nomes provenientes desses itens: levantamento de necessidades, desenho ou especificação (*design*), implementação ou construção, testes, documentação, implantação com inauguração (*start-up*), manutenção (*turnover*).

2.2.2. Ciclo de Vida do Projeto

A Engenharia de Software trata o software como produto, e como todo produto industrial, o software tem seu ciclo de vida, que pode ser caracterizado da seguinte forma (Paula Filho, 2000):

- é concebido a partir da percepção de uma necessidade;
- é desenvolvido, transformando-se em um conjunto de itens entregue a um cliente;
- entra em operação, sendo usado dentro de algum processo de negócio, e sujeito a atividades de manutenção, quando necessário;
- é retirado de operação, ao final de sua vida útil.

Na literatura existem diversos modelos de ciclos de vida, alguns clássicos, outros híbridos, algumas extensões e até alguns comerciais. A seguir, apresentam-se os principais e mais utilizados modelos de ciclo de vida.

O ciclo de vida mais caótico é aquele que pode ser chamado de codificação remenda - Figura 2.2 (Paula Filho, 2000). Partindo apenas de uma descrição, às vezes nem isso, os desenvolvedores começam imediatamente a codificar, remendando à medida que os erros vão sendo descobertos. Nenhum processo definido é seguido. Muitas vezes, esse modelo pode tornar-se atraente porque não exige nenhuma sofisticação técnica ou gerencial. Por outro lado, é um

modelo de alto risco, impossível de gerir e que não permite assumir compromissos confiáveis. Porém, é bastante provável que este seja o ciclo de vida mais utilizado.

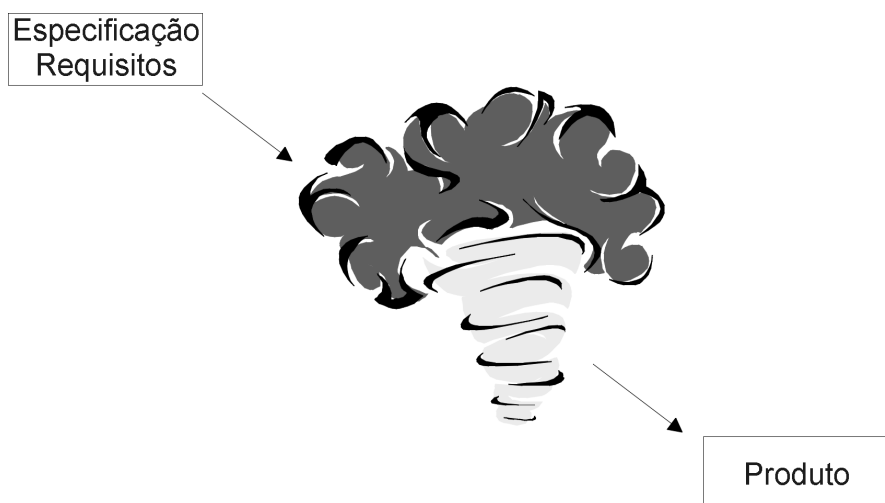


FIGURA 2.2 - Modelo do ciclo de vida codifica-remenda.

FONTE: Paula Filho (2000).

No modelo de ciclo de vida cascata - Figura 2.3 (Paula Filho, 2000), também conhecido como ciclo de vida clássico (Pressman, 1995), os principais sub processos são executados em estrita seqüência, o que permite demarcá-lo com pontos de controle bem definidos. Estes pontos de controle facilitam muito a gestão dos projetos, o que faz com que esse processo seja, em princípio, confiável e utilizável em projetos de qualquer escala. Por outro lado, se interpretado literalmente, é um processo rígido e burocrático, em que as atividades de requisitos, análise e projeto (*design*) têm de ser muito bem dominadas, pois não são permitidos erros. O modelo de cascata puro é de baixa visibilidade para o cliente, que só recebe o resultado final do projeto.

Na prática, é sempre necessário permitir que, em fases posteriores, haja revisão e alteração de resultados das fases anteriores. Uma variante que permite superposição entre as fases e a realimentação de correções é o

modelo sashimi - Figura 2.4 (Paula Filho, 2000). A superposição das fases torna difícil gerenciar projetos baseados nesse modelo de ciclo de vida.

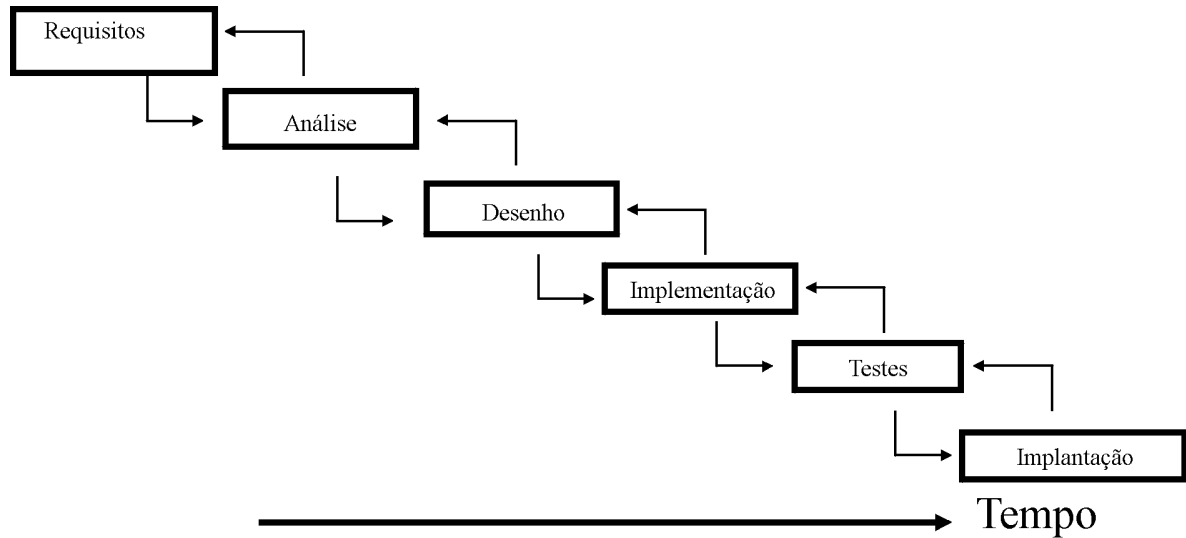


FIGURA 2.3 - Modelo do ciclo de vida cascata.

FONTE: Paula Filho (2000) e Pressman (1995).

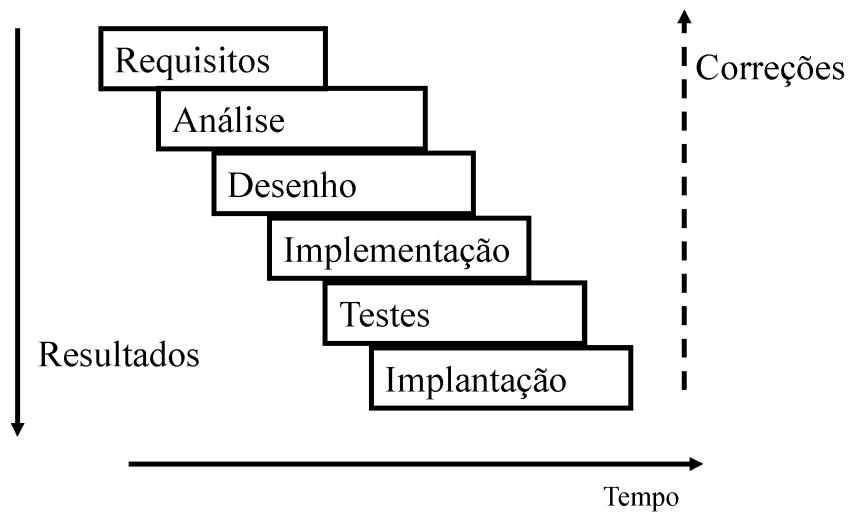


FIGURA 2.4 - Modelo de ciclo de vida sashimi.

FONTE: Paula Filho (2000).

Um modelo de ciclo de vida radicalmente diferente é o modelo em espiral - Figura 2.5 (Pressman, 1995) (PMI, 2000) (Paula Filho, 2000). O produto é desenvolvido em uma série de iterações. Cada nova iteração corresponde a uma volta na espiral. Isso permite construir produtos em prazos curtos, com novas características e recursos que são agregados na medida em que a experiência descobre sua necessidade. As atividades de manutenção são usadas para identificar problemas. O principal problema do ciclo de vida em espiral é que ele requer gestão muito sofisticada para ser previsível e confiável.

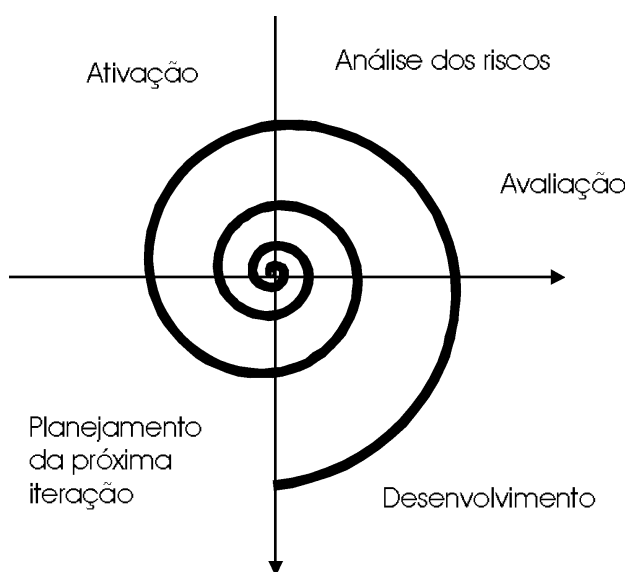


FIGURA 2.5 - Modelo de ciclo de vida espiral.

FONTE: Pressman (1995), PMI (2000) e Paula Filho (2000).

Uma variante do modelo em espiral é o modelo de prototipagem evolutiva - Figura 2.6 (Paula Filho, 2000). Nesse modelo, a espiral é usada não para desenvolver o produto completo, mas para construir uma série de versões provisórias que são chamadas de protótipos. Os protótipos cobrem cada vez mais requisitos, até que se atinja o produto desejado. A prototipagem evolutiva permite que os requisitos sejam definidos progressivamente, e apresenta alta flexibilidade e visibilidade para os clientes. Entretanto, também requer gestão

sofisticada, e o projeto (*design*) deve ser de excelente qualidade, para que a estrutura do produto não se degenere ao longo dos protótipos.

O modelo de entrega por estágios - Figura 2.7 (Paula Filho, 2000) difere do modelo de cascata pela entrega ao cliente de liberações parciais do produto. Isso aumenta a visibilidade do projeto, o que geralmente é um fator muito importante no relacionamento com o cliente. No entanto, apresenta os demais defeitos do modelo em cascata.

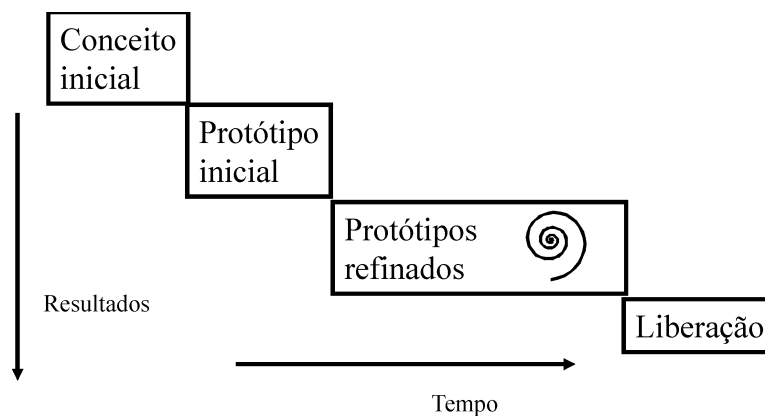


FIGURA 2.6 - Modelo de ciclo de vida prototipagem evolutiva.

FONTE: Paula Filho (2000).

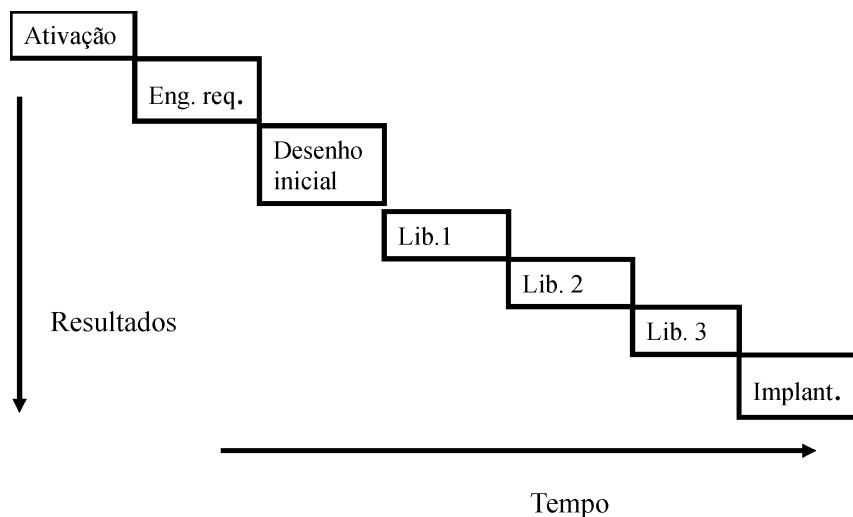


FIGURA 2.7 - Modelo do ciclo de vida entrega por estágios.

FONTE: Paula Filho (2000).

Uma combinação dos modelos de cascata e prototipagem evolutiva forma o modelo de entrega evolutiva - Figura 2.8 (Paula Filho, 2000). Este modelo permite que, em pontos bem definidos, os usuários possam avaliar partes do produto e fornecer realimentação quanto às decisões tomadas. Facilita também o acompanhamento do progresso de cada projeto, tanto por parte de seus gerentes como dos clientes. A principal dificuldade continua ser a realização do projeto (*design*) inicial, ele deve produzir uma arquitetura de produto robusta, que se mantenha íntegra ao longo dos ciclos de liberações parciais.

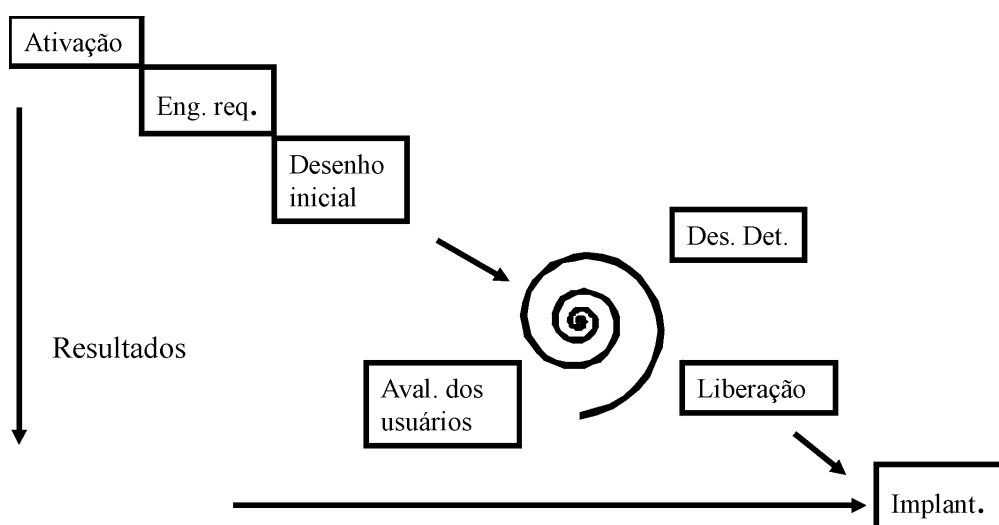


FIGURA 2.8 - Modelo do ciclo de vida entrega evolutiva.

FONTE: Paula Filho (2000).

2.3 Estrutura e Organização dos Processos de Gerenciamento de Projetos

A gerência de projetos é um esforço interativo, ou seja, uma ação, ou a falta de ação numa área, usualmente afeta também outras áreas. As interações podem ser diretas e claras, ou podem ser incertas e sutis (PMI, 2000).

Para auxiliar no entendimento da natureza da integração na gerência de projetos, e para enfatizar a importância da própria integração, o PMBOK (PMI,

2000) descreve a gerência de projetos em termos de seus processos e de suas interações.

Os processos de gerência de projetos podem ser organizados em cinco grupos, cada um deles contendo um ou mais processos:

- Processo de iniciação - autorização do projeto ou fase.
- Processo de planejamento - definição e refinamento dos objetivos e seleção da melhor das alternativas de ação para alcançar os objetivos aos quais o projeto estiver comprometido em atender.
- Processos de execução - coordenar pessoas e recursos para realizar o plano.
- Processos de controle - assegurar que os objetivos do projeto estão sendo atingidos, através da monitoração regular de seu processo para identificar variações do plano para que, ações corretivas possam ser tomadas quando necessárias.
- Processos de encerramento - formalizar a aceitação do projeto ou fase e encerrá-lo de uma forma organizada.

Os grupos de processos se relacionam pelos resultados que produzem, isto é, o resultado ou saída de um grupo torna-se entrada para outro, como mostrado na Figura 2.9.

Além disso, os grupos de processos de gerência de projetos não são separados ou descontínuos, nem acontece uma única vez durante todo o projeto, e eles são formados por atividades que se sobrepõem, ocorrendo em intensidades variáveis ao longo de cada fase do projeto, como poderá ser observado através da Figura 2.10.

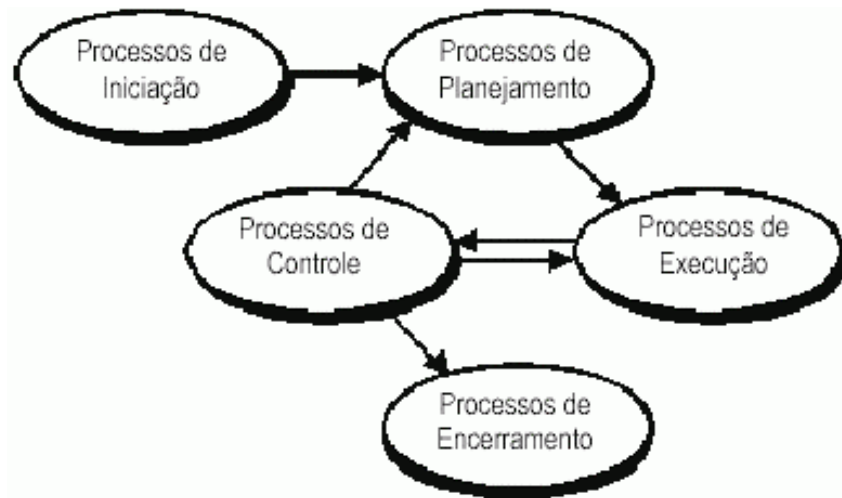


FIGURA 2.9 - Relacionamento entre os grupos de processos em cada fase.

FONTE: PMI (2000).

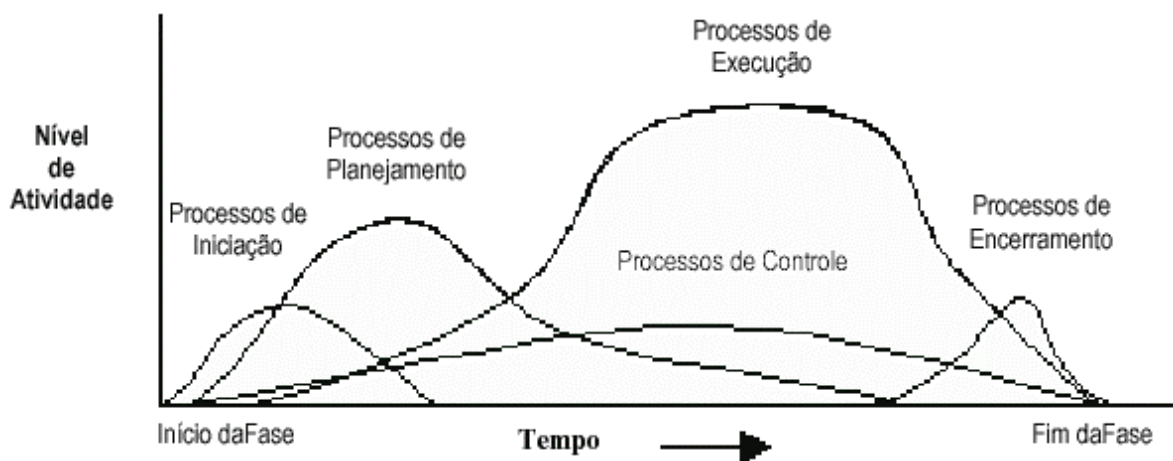


FIGURA 2.10 - Sobreposição dos grupos de processos em cada fase.

FONTE: PMI (2000).

Por fim, as interações dos grupos também atravessam as fases, de tal forma que o encerramento de uma fase fornece a entrada para o início da próxima. Esta interação está ilustrada na Figura 2.11.

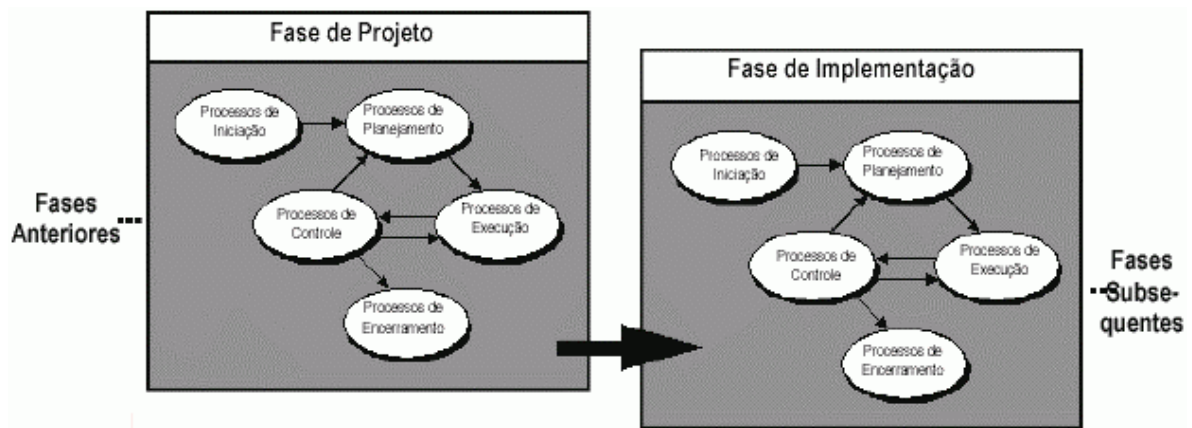


FIGURA 2.11 - Interações entre as fases.

FONTE: PMI (2000).

Num grupo de processos, os processos individuais são ligados por suas entradas e saídas. Considerando-se estas ligações, pode-se descrever cada processo em termos de:

- Entradas – documentos ou itens documentáveis que influenciarão o processo.
- Ferramentas e técnicas – mecanismos aplicados às entradas para criar as saídas.
- Saídas – documentos ou itens documentáveis resultantes do processo.

A estrutura e os processos de gerenciamento de projetos podem ser organizados de formas diferentes da apresentada anteriormente, que é a proposta do PMI. A seguir, serão apresentadas outras duas formas de organizar a estrutura e os processos de gerenciamento de projetos.

Para Paula Filho (2000), a gestão de projetos de desenvolvimento de software é organizada da seguinte forma:

- Procedimentos para gestão de requisitos: regulam o cadastramento, a manutenção e a alteração dos requisitos de um projeto de software.
- Procedimentos de planejamento de projetos: tratam da previsão de tamanho, esforços, recursos, prazos e riscos de um projeto de software.
- Procedimentos de controle de projetos: tratam do acompanhamento do progresso e dos riscos de um projeto, assim como do replanejamento e do balanço final.
- Garantia da qualidade e gestão das configurações.
- Procedimentos de contratação de projetos de desenvolvimento de software.

Já Sommerville (2001) trata o gerenciamento de projetos de software da seguinte forma:

- Planejamento do Projeto
 - Plano de qualidade - descreve os procedimentos para teste de qualidade que serão utilizados em um projeto.
 - Plano de validação - descreve a abordagem, os recursos e os métodos utilizados para a validação do sistema.
 - Plano de gerenciamento da configuração - descreve os procedimentos de gerenciamento e as estruturas a serem utilizadas.
 - Plano de manutenção - prevê os requisitos de manutenção do sistema, os custos de manutenção e o esforço necessário.

- Plano de desenvolvimento da equipe - descreve como as habilidades e as experiências dos membros da equipe de projeto serão desenvolvidas.
- Programação do Projeto
 - Identificar atividades.
 - Identificar dependências de atividades.
 - Estimar recursos para as atividades.
 - Alocar pessoas para as atividades.
 - Criar diagramas de projeto.
- Gerenciamento de Riscos
 - Identificação de riscos - são identificados os possíveis riscos de projeto, produto e negócios.
 - Análise de riscos - são avaliadas as possibilidades e as consequências da ocorrência desses riscos.
 - Planejamento de riscos - são traçados planos para enfrentar os riscos, seja evitando-os, seja minimizando seus efeitos sobre o projeto.
 - monitoramento de riscos - o risco é constantemente avaliado e os planos para diminuição de riscos são revisados, à medida que mais informações sobre eles se tornam disponíveis.

Nesta sessão, foi dada maior atenção às definições do PMBOK (PMI, 2000) por que são mais abrangentes; porém, as outras serão de fundamental importância quando, ocasionalmente, for tratado dos processos específicos e das possíveis

adaptações, dos processos propostos pelo PMBOK2000 para a gerencia de projetos de software.

2.4 Estimativas no Gerenciamento de Projetos

Medidas, na disciplina de Engenharia de Software, são denominadas de métricas, as quais podem ser definidas como métodos de determinar, quantitativamente, a extensão na qual o projeto, processo e produto de software têm certos atributos. Isto inclui a fórmula para determinar o valor da métrica como também a sua forma de apresentação, as diretrizes de utilização e interpretação dos resultados obtidos no contexto do ambiente de desenvolvimento de software (Daskalantonakis, 1992).

A realidade do dia-a-dia (DeMarco, 1989) (Pressman, 1995) e as pesquisas (Zimmerer; Yasin, 1998) (Bottoni, 2001) mostram que, apesar da massiva disseminação da gerência de projetos (Rad, Ragavan, 2000) (Forman, 2000), um grande número de projetos fracassa, não atingindo seus objetivos quanto a prazos, custos e satisfação do cliente (Sant'Anna, 2000).

Alguns autores (DeMarco, 1989) (Presman, 1995) associam o fracasso a problemas nas estimativas. De acordo com esses autores, falhas nas estimativas de prazos e custos são a principal causa de fracasso em projetos.

Normalmente são usados cinco tipos de estimativas: tempo, custos, tamanho, esforço e produtividade (Cantor, 1998). Para a projeção de cada uma dessas estimativas, existem vários métodos e técnicas, como por exemplo:

- LOC - *Lines Of Code* (DeMarco, 1989), método antigo e ineficaz para vários autores (Pressman, 1995).
- Analogia com projetos similares ou baseado na experiência do Gerente de Projetos ou do Estimador (Rouiller, 2001) (PMI, 2000).

- Pontos por Função ou *Function Point* (IFPUG, 1994), método bastante difundido cientificamente e utilizado comercialmente, inclusive por Órgãos do Governo do Brasil.
- *Constructive Cost Model* - COCOMO (Boehm, 1981), este método também é amplamente conhecido e utilizado pela comunidade de engenharia de software.
- Pontos por Caso de Uso ou *Use Case Point*, método mais recente que se originou da evolução do Pontos por Função, foi desenvolvido principalmente para projetos que utilizam metodologias orientadas a objetos pois permite um maior grau de precisão (Belloquim, 2002).
- Valor Agregado - VA (Vargas, 2002) ou *Earned Value* - EV, é focado no relacionamento entre o custo atual e o trabalho físico realizado no projeto.

Nas sessões seguintes serão apresentados alguns dos principais métodos e técnicas usadas atualmente.

2.4.1 Análise por Pontos de Função - FPA

O Ponto de Função (Fernandes, 1995) (IFPUG, 1994) mede o tamanho do software pela quantificação de sua funcionalidade externa, baseada no projeto lógico ou a partir do modelo de dados. A contagem dos Pontos de Função tem como objetivos:

- Medir o que o cliente/usuário requisitou e recebeu.
- Medir independentemente da tecnologia utilizada para implementação.
- Propiciar uma métrica de tamanho para apoiar análises de qualidade e produtividade.

- Propiciar um veículo para estimativa de software.
- Propiciar um fator de normalização para comparação entre softwares.

A técnica de FPA foi desenvolvida por Albrecht em 1974 quando trabalhava na IBM e aprimorada por Capers Jones e o International Function Point User's Group - IFPUG.

O IFPUG considera a abordagem de Albrecht como a Metodologia#1. A Metodologia#2 em desenvolvimento pelo próprio IFPUG (IFPUG, 1994) pressupõe a medição do software a partir do Modelo de Dados preliminar, o que permite realizar a primeira medição antes mesmo de iniciar o Projeto Lógico.

2.4.1.1 Estimativa do Tamanho do Software - Metodologia #1

Os procedimentos para a contagem dos Pontos de Função (ou tamanho de um software) de acordo com o *Counting Practice Manual - Release 4.0* do IFPUG (IFPUG, 1994), são:

- a) Determinar o Tipo de Contagem.
- b) Identificar as Fronteiras da Contagem.
- c) Determinar os Pontos de Função Não Ajustados.
- d) Determinar o Valor do Fator de Ajuste.
- e) Calcular o Ponto de Função Ajustado.

2.4.1.2 Estimativa do Tamanho do Software - Metodologia #2

Os procedimentos para a utilização da metodologia #2 são descritos a seguir.

- a) Determinar o Tipo de Contagem.

b) Identificar as Fronteiras da Contagem.

c) Determinar os Pontos de Função.

2.4.1.3 Estimativa do Prazo do Projeto

O emprego do Function Points para a estimativa do prazo de um projeto tem suas limitações, a não ser que a empresa se disponha a utilizar ferramentas comerciais disponíveis no mercado que objetivam a geração de estimativas dessa natureza (Fernandes, 1995).

Uma dessas ferramentas é o *Checkpoint*, desenvolvida pela SPR - *Software Productivity Research* (Jones, 1991), a qual consiste de um sistema especialista que, a partir da informação sobre o tamanho do software em Pontos de Função e sobre atributos do ambiente de desenvolvimento, gera uma série de estimativas, inclusive o prazo esperado para o projeto.

Na falta de um instrumento dessa natureza, o procedimento é realizar inferências sobre uma base de dados que contenha uma série histórica com informações acerca de projetos já completados em termos de: plataforma de hardware/software, processo de desenvolvimento (metodologia empregada), prazo, esforço, custo e tamanho.

O procedimento para realizar inferência, a partir do tamanho do software em Ponto de Função é:

- Determinar o Ambiente de Hardware/Software de Desenvolvimento.
- Determinar o Processo Que Será Utilizado Para o Desenvolvimento.
- Determinar o Prazo do Projeto.

2.4.1.4 Estimativa do Esforço do Projeto

O esforço estimado para um projeto é medido pela alocação dos recursos humanos em termos de homem/mês ou homem/hora.

A inferência a ser realizada é similar à anterior. Entretanto, a transformação pode ser direta caso o *data set* contiver o dado sobre a produtividade média do desenvolvimento conforme a plataforma de hardware/software e respectiva metodologia ou se houver valor de Ponto de Função similar ao estimado.

A interpolação também pode ser empregada nos casos de valores de PF's estimados, intermediários aos existentes no *data set*.

2.4.1.5 Custo Estimado para o Projeto

O custo estimado para o projeto é o produto do custo do homem/mês ou homem/hora, multiplicado pelo esforço previsto sendo que, de forma simplificada, o $\text{Custo} = f(\text{Esforço})$.

Aqui, deve-se observar duas abordagens para tratar o custo do homem/mês:

- Custeio Direto
- Custeio Por Absorção Total

O custeio direto considera apenas o salário + encargos ou em termos de valores médios ou aquele que representa o recurso específico a ser empregado no projeto, de acordo com a respectiva faixa salarial.

O custeio por absorção total já é mais complexo, pois deve considerar um rateio entre diversos tipos de despesas incidentes sobre o desenvolvimento do software, tais como: recursos computacionais, equipamentos auxiliares, serviços contratados, custos administrativos, custos com materiais de

consumo, custos de pessoal referente às áreas de assessoria, chefia do desenvolvimento e assim sucessivamente.

O conceito que está por trás desta abordagem é que a área de desenvolvimento “vende” serviços de análise e programação ou funções, que são medidos pelo consumo de Homens/Mês ao projeto.

Portanto o custo do Homem/Mês no modo de absorção total é:

(salários + encargos) + parcela correspondente de rateio

Para simplificar, pode-se considerar a relação desenvolvida por Bandeira (1984), cujo custo de pessoal no desenvolvimento de software representa cerca de 60 a 70% do custo total, excluindo itens de investimento que sejam necessários.

Outra forma de determinar o custo estimado do projeto é através do custo unitário para produzir um Ponto de Função.

2.4.1.6 Estimativa do Número de Instruções Fontes

De acordo com Jones (1991), a predição do número de instruções fontes para um software é baseada na observação empírica do nível da linguagem e sobre o número de instruções requerido para implementar um ponto de função.

O nível da linguagem, a partir das pesquisas realizadas na IBM por Capers Jones e Albrecht, foi normalizado a partir de “instruções equivalentes em Assembler” na linguagem alvo, a fim de produzir um ponto de função.

Por exemplo, uma instrução fonte em Cobol equivale a três em Assembler, já o PL/1 equivale a quatro.

Para a estimativa do número de instruções fontes do software deve-se determinar a linguagem (ou linguagens) de desenvolvimento e realizar a transformação.

2.4.2 Estimativas Utilizando o COCOMO (Constructive Cost Model)

O método COCOMO foi desenvolvido por Boehm (1981), para estimar esforço, prazo, custo e tamanho da equipe para um projeto de software.

O método foi derivado a partir de um *data set* compreendendo 63 projetos, cobrindo áreas como: negócios, controle, científica, suporte e sistema operacional.

Existem três modelos neste método:

- COCOMO Básico.
- COCOMO Intermediário.
- COCOMO Detalhado.

O COCOMO Básico é uma versão aplicável à grande maioria dos projetos de software: pequenos a médios softwares, desenvolvidos *in-house*.

Entretanto, as estimativas fornecidas pelo COCOMO Básico são limitadas, pois desconsideram alguns fatores, tais como restrições de hardware, qualificação e experiência do pessoal, uso de modernas técnicas e ferramentas e outros atributos do projeto.

O COCOMO Intermediário já adiciona estes fatores, proporcionando estimativas mais acuradas.

O COCOMO Detalhado, por sua vez, apresenta técnicas para se estimar tanto a nível de módulo, subsistema e sistema, individualizando, a cada fase do projeto, os atributos de custo.

Um método de estimativa, para projetos de software, tem certa exatidão se o mesmo pode estimar os custos de desenvolvimento com 20% de variação em relação ao previsto e 30% em relação ao prazo.

De acordo com Boehm (1981), o COCOMO Intermediário enquadra-se dentro desta definição.

O COCOMO categoriza os projetos de software em três tipos fundamentais: modo orgânico, modo difuso e modo restrito.

2.4.2.1 Modo Orgânico (Convencional)

No modo orgânico, equipes relativamente pequenas desenvolvem sistemas num ambiente altamente “familiar”, *in-house*. A maior parte das pessoas engajadas no projeto tem experiência prévia com sistemas similares na organização, bem como tem um completo entendimento de como o sistema em desenvolvimento contribuirá para os objetivos da empresa.

Isto significa que a maioria das pessoas engajadas no projeto pode contribuir para o mesmo em seus estágios iniciais, sem gerar uma grande sobrecarga de comunicação sobre o que o sistema fará e qual será a divisão de trabalho para a equipe.

Um projeto orgânico é relativamente flexível sobre a forma como o software atende às especificações de requisitos e interfaces.

Outras características do modo orgânico são: a) ambiente estável de desenvolvimento; b) algoritmos simples; c) prêmio relativamente baixo para término antes do prazo; d) tamanho relativamente pequeno, projetos na faixa de 50.000 instruções fontes.

2.4.2.2 Modo Difuso

O modo difuso representa um estágio intermediário entre os modos orgânico e restrito. Significa uma mistura das características dos modos orgânico e detalhado.

Suas características básicas são:

- a equipe mescla grande e pouca experiência com a aplicação;
- a equipe mescla grande e pouca experiência com a tecnologia;
- o tamanho do software varia até 300.000 instruções fontes.

2.4.2.3 Modo Restrito

O principal fator que distingue um projeto de software de Modo Restrito é a necessidade para operar conforme grandes restrições. O produto deve operar dentro de um contexto complexo de hardware, software, regras e procedimentos operacionais, tal como um sistema de transferência eletrônica de fundos ou um sistema de controle de tráfego aéreo.

Como resultado, projetos de modo restrito não têm a opção de modificar facilmente suas especificações. São projetos que requerem altos custos de verificação e validação.

Geralmente, o prêmio para término antes do prazo é extremamente alto.

2.4.3 Valor Agregado (*Earned Value*)

O conceito do Valor Agregado - VA (Vargas, 2002), ou *Earned Value* - EV, é focado no relacionamento entre o custo atual que esta sendo gasto e o trabalho físico realizado no projeto. O VA é a métrica do andamento do projeto ou o valor do trabalho concluído até a data. O VA é uma técnica de gerenciamento de projetos que integra o planejamento de recursos e datas com a evolução física.

O EVM (*Earned Value Management*), Gerência do Valor Agregado, é a metodologia que utiliza o Valor Agregado como ferramenta para integrar o gerenciamento de custo, programação de datas, performance técnica e

gerenciamento de risco de um projeto, gerando indicadores de performance consistentes.

Todas as medições incluem um componente de custo. Os custos são exibidos em uma curva S porque os gastos são mínimos no início do projeto, aumentando perto do meio do projeto e são reduzidos no final. Isso significa que as medições do valor agregado também adotarão um formato de curva S (Heldman, 2003).

A seguir serão apresentados os elementos, os índices e as fórmulas para cálculo do desempenho do projeto.

BCWS (*Budgeted Cost of Work Scheduled*)

Valor planejado (orçado) do projeto, acumulado até o período de análise. É o valor que indica a parcela do orçamento que deveria ser gasta, considerando a linha base da atividade, atribuição ou recurso. O BCWS do último período é igual ao valor total orçado para o projeto (BAC).

BCWP (*Budgeted Cost of Work Performed*)

Valor agregado do projeto, acumulado até o período de análise. Corresponde ao somatório dos valores agregados das atividades concluídas. O valor agregado de cada atividade é igual ao valor planejado (orçado) da atividade.

ACWP (*Actual Cost of Work Performed*)

Valor real gasto no projeto, acumulado até o período de análise.

SV (*Schedule Variance*)

Varição de Prazo: indica se o prazo está adiantado ou atrasado em relação ao planejado para esse período de tempo. É a diferença, em termos de custo, entre o valor agregado (BCWP) e o cronograma da linha base (BCWS).

Se SV for positiva, o projeto estará antecipado em termos de custo, se for negativa, o projeto estará atrasado em termos de custo.

$$SV = BCWP - BCWS$$

CV (*Cost Variance*)

Varição de Custo: indica se os custos estão mais altos ou inferiores aos orçados. É a diferença entre o custo previsto para atingir o nível atual de conclusão (BCWP) e o custo real (ACWP), até o período de análise.

Se CV for positiva, o custo estará aquém do valor previsto (linha base), se for negativa, a atividade terá ultrapassado o orçamento.

$$CV = BCWP - ACWP$$

TV (*Time Variance*)

É a diferença determinada em tempo entre o previsto pelo projeto e o realizado. É encontrado graficamente pela projeção da curva de BCWS e BCWP, encontrando a data em que o BCWS agrega o mesmo valor de BCWP. A diferença entre a data atual e essa data representa o atraso ou adiantamento do projeto.

SPI (*Schedule Performance Index*)

É a divisão entre o valor agregado (BCWP) e o valor planejado na linha base (BCWS). O SPI mostra a taxa de conversão do valor previsto em valor agregado.

Se o SPI for menor que 1, indica que o projeto está sendo realizado a uma taxa de conversão menor que a prevista.

$$SPI = BCWP/BCWS$$

CPI (*Cost Performance Index*)

É a divisão entre o valor agregado (BCWP) e o custo real (ACWP). O CPI mostra qual a conversão entre os valores reais consumidos pelo projeto e os valores agregados no mesmo período.

Se o CPI for menor que 1, indica que o projeto está gastando mais do que o previsto.

CPI - BCWP/ACWP

EAC (*Estimated at Completion*)

Valor financeiro que representa o custo final do projeto quando concluído. Inclui os custos reais incorridos (ACWP) e os valores restantes estimados (ETC).

EAC=ACWP+ETC

ETC Menos sensível

EAC = BAC - BCWP

ETC Tradicional

ETC = (BAC-BCWP)/CPI

ETC Mais Sensível

ETC = (BAC-BCWP)/CPIxSPI

PAC (*Plan at completion*)

É a data de término prevista para o projeto.

TAC (*Time at Completion*)

É a data de término projetada para o projeto. É calculada como a razão entre a data prevista PAC e o SPI.

$$TAC = PAC/SPI$$

DAC (*Delay at Completion*)

É a diferença em termos de unidade de tempo entre o prazo previsto e o prazo projetado para o projeto. É determinado pela diferença entre o prazo previsto inicialmente PAC e a data de término projetada TAC.

$$DAC = PAC - TAC$$

BAC (*Budget at Completion*)

Valor total planejado (orçado) para o projeto.

VAC (*Variance at Completion*)

Projeção de Variação de Custo na conclusão do projeto: valor total planejado menos valor estimado na conclusão.

$$VAC = BAC - EAC$$

Na Figura 2.12 é apresentado um gráfico onde foram plotadas as curvas e índices do cálculo do VA.

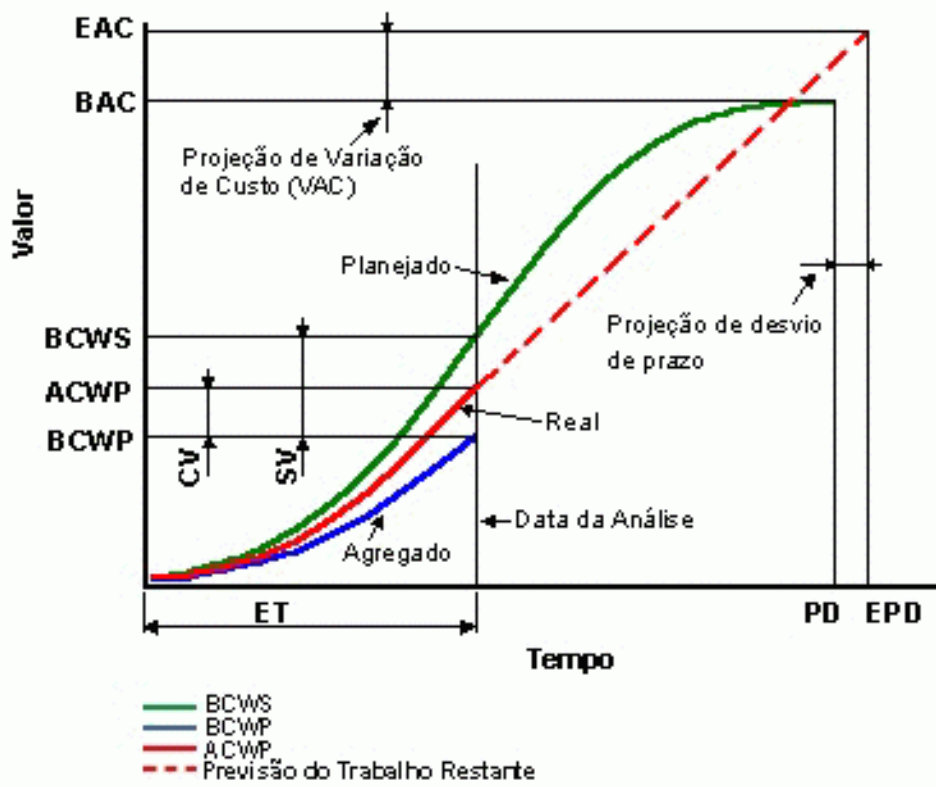


FIGURA 2.12 - Gráfico completo da projeção do VA.

CAPÍTULO 3

LINGUAGENS, ABORDAGENS E NORMAS

Nas sessões seguintes serão apresentadas as normas, os modelos de maturidade, as abordagens de processos e a linguagem de modelagem de processos de software que serão utilizados neste trabalho.

3.1 Conceitos

3.1.1 Processo de Software

Em (Sant'Anna, 2000) pode-se verificar que nos primeiros anos da Engenharia de Software o termo “processo” era utilizado para denotar o “processo de desenvolvimento” de software como um todo, ou seja, um processo macro e abstrato que levava à construção do software. Mais recentemente, outros termos relacionados a “processo”, como: processo de gerenciamento, processo de garantia da qualidade, etc., foram apresentados também como responsáveis e importantes na construção do produto de software final.

Um processo de software inclui atividades técnicas e administrativas que são utilizadas na produção e manutenção de software, por exemplo: determinação e especificação de sistema e requisitos de software, análise e administração de riscos, prototipação de software, *design*, implementação, verificação e validação, controle de qualidade de software e garantia, integração de componentes, documentação, gerenciamento de versões e configuração de software, gerenciamento de dados, evolução do software, administração de projeto, etc. (Madhavji, 1991).

O processo ou um conjunto de processos, que são utilizados por uma organização ou projeto para: planejar, gerenciar, executar, monitorar, controlar e melhorar as atividades relacionadas à construção de software (Rout, 1995).

A ISO/IEC 12207 (ISO, 1995), semelhantemente a ISO 9000 (ABNT, 2000b), define processo de software como um conjunto de atividades inter-relacionadas, que transformam entradas em saídas.

No CMM-SW (Paulk et al., 1993), processo de software é definido como o conjunto de atividades, métodos, práticas e transformações que a pessoa usa para desenvolver e manter o software e os produtos associados.

Já a nova abordagem CMMI (CMMI, 2001) apresenta três conceitos: processo, processo gerenciado e processo definido.

Processo consiste de atividades que podem ser reconhecidas como implementações de práticas do modelo CMMI. Essas atividades podem ser mapeadas para uma ou mais práticas das áreas de processo CMMI para permitir que o modelo seja utilizado para a melhoria e avaliação do processo.

Um processo gerenciado é um processo que é planejado e executado de acordo com uma política; emprega pessoas capacitadas com recursos adequados para produzir saídas controladas; envolve os *stakeholders* relevantes; é monitorado, controlado e avaliado para sua padronização.

Processo definido é um processo gerenciado que é adaptado a partir de diretivas organizacionais extraídas de um conjunto de processos padrão de uma organização; tem uma descrição do processo; e contribui com produtos de trabalho, medição, e outras informações para melhoria do conjunto de processos organizacionais. Um processo definido fornece, para um projeto, uma base para planejamento, execução, e melhoria das tarefas e atividades do projeto. Um projeto pode ter mais de um processo definido.

Sant'Anna (2000) define o processo de engenharia de software como um processo de software composto de um conjunto de processos, capazes de conduzir a organização envolvida com a construção de produtos de software

com qualidade e custos previsíveis, de forma eficiente, gerenciada e com a possibilidade de melhoria constante.

O termo “processo de software” também é denominado, quando o mesmo não caracteriza confusão terminológica no meio utilizado, apenas como "processo".

Um processo, por sua vez, é composto por outros elementos. Não existe um consenso sobre a taxinomia dos elementos que compõem um processo de software. Os trabalhos sobre o assunto divergem entre si, diversos esforços (Conradi et al., 1992) (Feiler et al., 1991) (Dowson et al., 1991) (Madhavji, 1991) (Lonchamp, 1993) foram e estão sendo realizados para tentar encontrar um consenso entre as terminologias adotadas. O que pode ser observado é que qualquer processo de software, de forma geral, possui como principais componentes: subprocessos, recursos humanos e computacionais, metas e estruturas organizacionais (Lonchamp, 1993).

3.1.2 Melhoria de Processo de Software

O programa de melhoria das organizações de maior maturidade tem forte ênfase na automatização dos processos de software. O esforço e a participação na definição e melhoria de processo são reais e fazem parte do trabalho cotidiano (Paulk, 1999) porque, nessas organizações, existe uma cultura de qualidade de processos (Miller, 1998).

Melhoria de processo de software significa entender os processos existentes e mudar estes processos para melhorar a qualidade dos produtos e/ou reduzir custos e tempo de desenvolvimento (Sommerville, 2001).

O SPICE define melhoria de processo de software como toda ação tomada para mudar um processo da organização, de modo a atender às necessidades dos negócios desta organização e alcançar estes objetivos mais eficientemente (SPICE, 1995).

Para Fuggetta (2000), os processos precisam sofrer continuamente mudanças e refinamentos para melhorar a sua habilidade de alcançar os requisitos indicados e as expectativas dos envolvidos; portanto, os processos precisam ser continuamente avaliados e melhorados.

3.1.3 Modelo de Processo e Linguagem de Modelagem de Processo

O processo é o elemento principal de um ambiente de desenvolvimento de software, já que a qualidade de um produto de software está diretamente ligada à qualidade do processo utilizado no seu desenvolvimento (Kellner; Hansen, 1988) (Humphrey; Kellner, 1989).

Diante disso, a modelagem de processos apresenta-se como um recurso indispensável para a formalização dos processos organizacionais, de desenvolvimento de software e de gerência de projetos.

Diferentes modelos de processos podem definir diferentes pontos de vista. Por exemplo, um modelo pode definir os agentes envolvidos em cada atividade, enquanto outro pode estar centrado no relacionamento entre as atividades. Há modelos que são relacionados à cultura organizacional e são focados nas capacidades comportamentais ou nos encargos dos papéis envolvidos no processo de software (Acuña et al., 2000).

Cada modelo observa, foca ou dá prioridade a pontos particulares de cada mundo complexo da construção de software (Feiler, 1993).

Um modelo é sempre uma abstração da realidade e, como tal, representa a descrição de uma realidade parcial e simplificada, isto é, nem todas as partes ou aspectos do processo são levados em conta no modelo. Geralmente, um modelo de processo pode ser dividido em muitos sub modelos, expressando diferentes pontos de vista ou perspectivas (Acuña; Ferré, 2001).

Existem várias linguagens de modelagem de processos, ou *Process Modeling Languages* (PMLs), que podem dar apoio à modelagem de processos. Dentre as mais abordadas, destacam-se a ProNet (Christie, 1993), a E3 - *Environment for Experimenting and Envolving software processes* (Jaccheri et al., 1998) e a PROMENADE - *Process-Oriented Modelling and Enactment of Software Developments* (Franch; Ribó, 1999).

As PMLs podem ser usadas para: entender os processos; projetar processos; treinar e educar; otimizar processos, simular processos e apoiar processos (Fuggetta, 2000).

Vale salientar que diversas técnicas não são recentes e são largamente utilizadas em diversas metodologias. Processos de software possuem suas próprias peculiaridades; devido as suas características, as formas abordadas para modelagem até então provocavam confusão.

Kellner; Hansen (1988) e Humphrey; Kellner (1989), abordam a modelagem de processos de software por meio da utilização de técnicas de análise de sistemas, onde são utilizados diagramas de fluxo de dados (DFD), que fornecem apenas uma visão de como as tarefas são executadas no processo.

Este tipo de modelo oferece uma visão limitada, mostrando apenas uma visão funcional do mesmo, descrevendo “o quê” está sendo feito e qual dado está fluindo, não oferecendo informações sobre precedência e consequência sobre cada tarefa executada no processo; sendo necessário, então, modelar o processo sob outros pontos de vista.

A linguagem STATEMATE (Kellner ; Hansen, 1988) (Humphrey; Kellner, 1989) fornecia outros tipos de recursos, fornecendo subsídios para se modelar o processo sob outros dois pontos de vista: o comportamental, representado o comportamento (“quando” e “como”), através de diagramas de transição de estados e o estrutural, mostrando quem executa cada tarefa, representada por diagramas de módulo.

Os processos podem ser modelados usando metodologias orientadas a objetos (BenShaul; Kaiser., 1994) (Georgakopoulos; Hornick, 1995) (Franch; Ribó, 1999) (Mühlen; Becher, 1999) (Jaccheri et al., 1999) (Rational, 2000). Inclusive, alguns trabalhos mais recentes (Schleicher et al., 1998) (Schleicher, 2000) (Schleicher; Westfechtel, 2001) propõem a extensão do meta-modelo *Unified Modeling Language* - UML (Booch et al., 2000) para apoio à modelagem de processos.

Foram pesquisadas algumas extensões do meta-modelo UML, para que este fosse usado para a modelagem de processos de engenharia de software (Cereja Junior et al., 2002a) (Cereja Junior et al., 2002b) (Cereja Junior et al., 2003) (Genvigir et al., 2002) (Genvigir et al., 2003) (Lahoz Neto et al., 2002) (Abdala et al., 2002) e de gerência de projetos (Borrego Filho et al., 2002a) (Borrego Filho et al., 2002b) (Borrego Filho et al., 2003).

Essas pesquisas se faziam necessárias dada a carência por uma PML Orientada a Objetos - OO com notação UML. Essas características tornam-se convenientes, uma vez que o desenvolvimento de software OO apoiado pela notação UML é uma tendência antiga e que vem se confirmando a cada dia pela comunidade de desenvolvimento de software.

A carência, por PMLs desse tipo, foi contemplada pela descoberta da *Unified Process Model* (UPM) (OMG, 2000). A UPM é uma PML OO baseada na notação UML, e que foi desenvolvida e padronizada pela *Object Management Group* (OMG), o que garante uma maior credibilidade perante a comunidade de engenharia de software.

A UPM será melhor descrita nas sessões seguintes, já que esta PML será usada no decorrer deste trabalho e os processos aqui descritos serão modelados com essa linguagem.

3.2 SPICE – ISO/IEC 15504

O modelo SPICE é uma base para avaliação de processos de software que harmoniza os diversos modelos nos quais ele se baseia, como: SW-CMM, Trillium, *Software Technology Diagnostic* (STD) e Bootstrap. O objetivo é que cada um destes modelos, e outros que venham a ser criados, possam ser definidos como modelos compatíveis com esta infra-estrutura, possibilitando que os resultados de avaliações, segundo cada um destes modelos, possam ser comparados.

O resultado do projeto está previsto para ser transformado em norma, com o nome de norma ISO/IEC 15504, e foi desenvolvido por um grupo chamado JTC/SC7 em elaboração conjunta pela ISO e pelo International Electrotechnical Commission (IEC) como um framework para avaliação de processos de engenharia de software e da organização do projeto e do negócio.

O SPICE pode ser utilizado por organizações envolvidas em planejar, gerenciar, monitorar, controlar e melhorar a aquisição, fornecimento, desenvolvimento, operação, evolução e suporte de software. Dentro da visão do SPICE, a avaliação de processos de software tem como propósito:

- Entender o estado dos processos de software de uma organização para a sua melhoria;
- Determinar a adequação dos processos utilizados em uma organização um requisito particular ou uma classe de requisitos;
- Determinar a adequação dos processos utilizados por uma outra organização, para um determinado contrato ou para uma classe de contratos.

Dentro do contexto de melhoria de processos de software, a avaliação significa a caracterização dos processos de software de uma organização, unidade organizacional ou projeto em termos da capacidade de processos

organizacional ou projeto em termos da capacidade de processos selecionados. A análise dos resultados é feita em relação às necessidades de negócio da organização, identificando os aspectos positivos e negativos, e os riscos associados aos processos. Isto leva a determinar se os processos estão atingindo efetivamente seus objetivos e a identificar causas da baixa qualidade. A Figura 3.1 apresenta a caracterização do processo de software, conforme apresentada nas definições da abordagem SPICE.

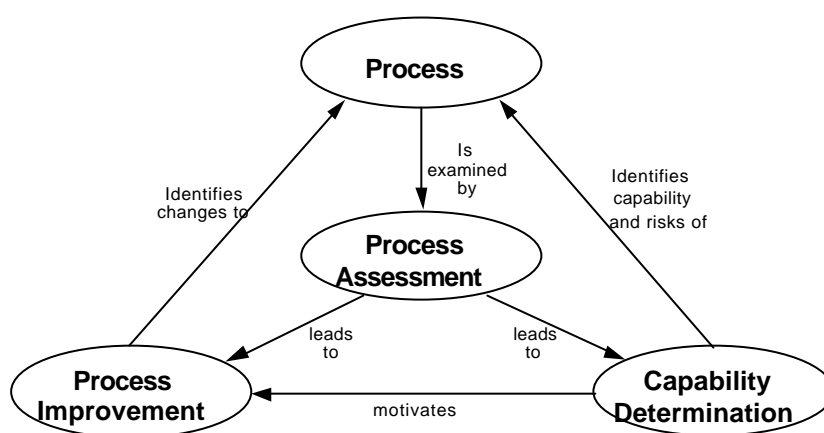


FIGURA 3.1 – Caracterização do processo de software

FONTE: SPICE (1995)

O SPICE estabelece um Modelo de Referência de processos e define um conjunto universal de processos que são fundamentais para uma boa engenharia de software, cobrindo as melhores práticas. O conjunto de processos desse modelo é apresentada no Tabela 3-1.

TABELA 3-1 – Modelo de processos (SPICE, 1995).

CATEGORIAS PROCESSO	DE	SIGLA	PROCESSOS
Cliente-Fornecedor (Customer-Supplier)		CUS.1	Adquirir software ou serviço
		CUS.2	Estabelecer Contrato

(Continua)

TABELA 3.1 – Continuação.

	CUS.3	Identificar as necessidades do cliente
	CUS.4	Realizar auditorias conjuntas
	CUS.5	Empacotar, entregar e instalar produto
	CUS.6	Apoiar a operação do software
	CUS.7	Propiciar serviço de atendimento ao cliente
	CUS.8	Obter a satisfação do Cliente
Engenharia	ENG.1	Desenvolver requisitos e projeto de sistema
	ENG.2	Desenvolver requisitos de software
	ENG.3	Desenvolver projeto do software
	ENG.4	Implementar o projeto de software
	ENG.5	Integrar e testar o software
	ENG.6	Integrar e testar o sistema
	ENG.7	Manter o sistema e o software
Apoio (Support)	SUP.1	Desenvolver documentação
	SUP.2	Executar a gerência de configuração
	SUP.3	Executar a garantia da qualidade

(Continua)

TABELA 3.1 – Conclusão.

	SUP.4	Efetuar a solução de problemas
	SUP.5	Executar a validação dos produtos de trabalho
	SUP.6	Executar revisões conjuntas
Gerência de Projeto	PRO.1	Planejar o ciclo de vida
	PRO.2	Estabelecer o plano do projeto
	PRO.3	Estabelecer Equipes de projeto
	PRO.4	Gerenciar Requisitos
	PRO.5	Gerenciar a Qualidade
	PRO.6	Gerenciar Risco
	PRO.7	Gerenciar Recursos e Cronograma
	PRO.8	Gerenciar Subcontratos
Organização	ORG.1	Construir o negócio
	ORG.2	Definir o processo
	ORG.3	Melhorar o processo
	ORG.4	Realizar Treinamento
	ORG.5	Propiciar reutilização
	ORG.6	Preparar o ambiente de engenharia de software
	ORG.7	Propiciar facilidades de trabalho

A categoria de processo Cliente-Fornecedor (Customer-Supplier) (CUS) consiste em processos que diretamente impactam o cliente, suportando o desenvolvimento e transição do software para o cliente, e provendo o uso e a sua correta operação

A categoria de processo de Engenharia (ENG) consiste em processos que diretamente especificam, implementam, ou mantêm um sistema e produto de software e a documentação do usuário.

Em algumas circunstâncias, não há um "sistema", assim a extensão do processo de engenharia é reduzido ao software e à documentação do usuário, e os processos ENG.1 e ENG.6 não são aplicados.

Enquanto os processos listados a seguir aparecem como "um modelo em cascata", a intenção não é impedir a execução simultânea e iterativa. (A seqüência é determinada e documentada através das práticas base, "Determinar a estratégia de *release*" ENG.1.4 e através de processo, "Plano do ciclo de vida do projeto" PRO.1.)

As entradas para a categoria de processo "Engenharia" possibilita a inclusão de contrato ou acordos que descrevam como o trabalho será feito, e um ou mais planos de como será realizado.

A categoria de processo Projeto (PRO) consiste dos processos que estabelecem o projeto, e coordenam e gerenciam os recursos para produzir um produto ou providenciar um serviço que satisfaça o cliente.

A entrada é um contrato ou documento em que o trabalho é descrito. O foco nesta categoria de processo é o uso efetivo dos recursos (tempo, esforço, pessoas, dinheiro) para atingir os propósitos e objetivos do projeto.

A categoria de processo Apoio (Support) (SUP) consiste dos processos que podem ser empregados por vários outros processos (incluindo outros processos de suporte). Os processos de suporte podem ser empregados em

vários pontos dos ciclos de vida e podem ser executados por organizações que os estejam usando, por clientes ou por organizações independentes.

Para empregar um processo de suporte com outro processo, são requeridos alguns arranjos para o processo de apoio, como exemplo, a formalidade e o rigor do gerenciamento de configuração depende do produto do trabalho e severidade e da necessidade de estabilizar e controlar o produto do trabalho.

Ao avaliar um processo de apoio em particular, considerações especiais devem ser dadas se esse processo foi implementado tão amplamente quanto deveria ser implementado. Também, requererá o julgamento do assessor sobre se a implementação é satisfatoriamente formal e rigorosa para as necessidades particulares da situação.

A Categoria de Processo Organização (ORG) consiste em processos que estabelecem as metas empresariais da organização e desenvolvem processos, produtos e ativos de recursos que, quando usados pelos projetos na organização, ajudarão a organização a alcançar suas metas organizacionais.

Processos organizacionais:

- Construir a infra-estrutura organizacional.
- Alavancar o que de melhor está disponível em qualquer parte da organização (processos efetivos, habilidades avançadas, código de qualidade, boas ferramentas de apoio); e disponibilizá-los a todos.

Ao aplicar as características comuns para os níveis da organização, é observado que o processo definido descrito em uma prática genérica pode ser implementado tanto na organização como também em um projeto.

No Modelo de Referência do SPICE, são definidos ainda seis níveis de capacitação, conforme a Tabela 3.2. Na avaliação de uma organização, são selecionados os processos relevantes e para cada um deles é atribuído um

perfil composto pela porcentagem de adequação a cada um dos níveis de capacitação.

TABELA 3.2 – Níveis de capacitação.

Nível	Nome	Descrição
Nível 0	Processo incompleto	O processo não está implementado. O processo falha na tentativa de atingir seus objetivos.
Nível 1	Processo executado	O processo implementado atinge o seu objetivo definido
Nível 2	Processo gerenciado	O processo executado entrega produtos de trabalho de definida qualidade dentro de cronogramas e recursos definidos
Nível 3	Processo estabelecido	O processo gerenciado é executado usando um processo definido baseado em bons princípios de engenharia de software
Nível 4	Processo previsível	O processo estabelecido é executado consistentemente dentro de limites definidos de controle para atingir seus objetivos
Nível 5	Processo otimizado	O processo previsível otimiza o seu desempenho para atender às necessidades de negócio atuais e futuras, e atinge repetibilidade em atender seus objetivos definidos de negócios.

FONTE: SPICE (1995).

Uma avaliação é conduzida avaliando o processo selecionado contra o modelo de processo definido (Figura 3.2). Este modelo bidimensional consiste em um grupo de práticas base, do processo específico, e um grupo de práticas genéricas.

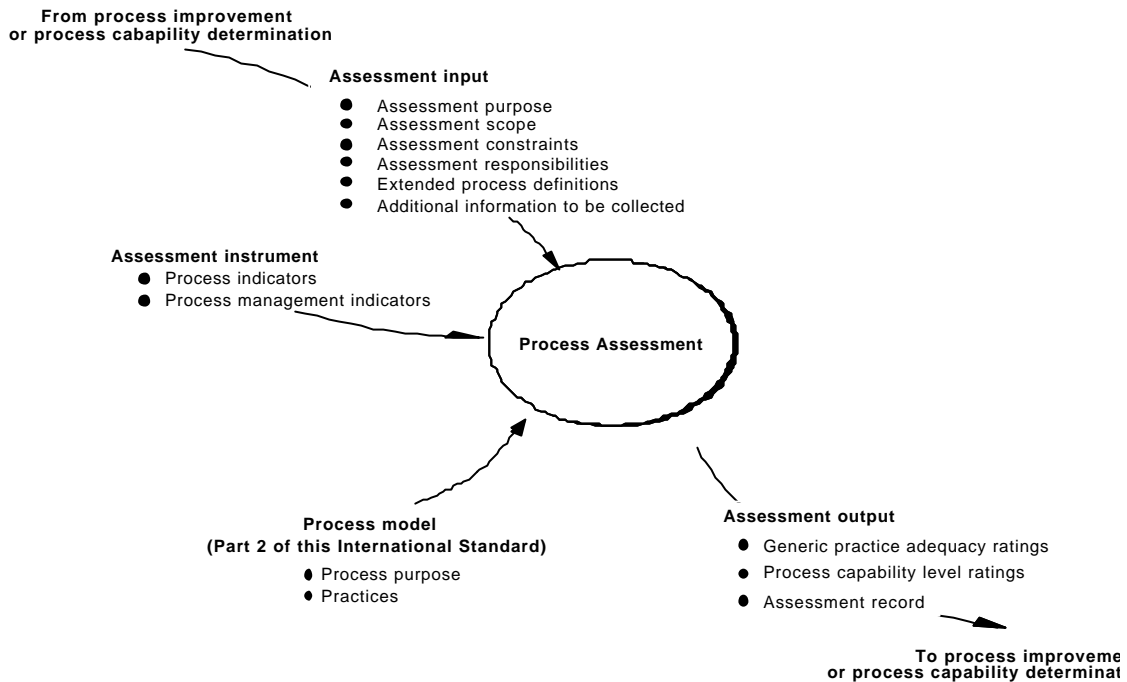


FIGURA 3.2- Modelo de avaliação do processo.

FONTE: SPICE 15504 (SPICE, 1995).

As práticas genéricas, Tabela 3.3,

Tabela 4.4, Tabela 3.5, Tabela 3.6, Tabela 3.7, são aplicadas em todos os processos e se agrupam em características comuns e níveis de capacidade que podem ser usados para determinar como o processo é administrado. A avaliação inclui um grupo de capacidades de processo para cada instância do processo avaliado.

A nomenclatura para as práticas é definida para identificá-las de forma não ambígua e relacioná-las à arquitetura do modelo. A nomenclatura para as

práticas básicas facilita a identificação da categoria do processo, o processo pertencente a cada categoria de processo, e as práticas básicas que pertencem ao processo. Para práticas genéricas, a nomenclatura facilita a identificação para os níveis de capacidade, as características comuns que pertencem a cada nível de capacidade, e as práticas genéricas que pertencem a cada característica comum.

TABELA 3.3 - Práticas genéricas para o nível 1: Executado Informalmente.

	Característica Comum 1.1: Executando as Práticas Básicas
1.1.1	Executar o Processo. Executar a prática básica de um processo para propiciar produtos operacionais e/ou serviços ao cliente. Nota: O Cliente do processo pode ser interno ou externo à organização

TABELA 4.4 - Práticas genéricas para o nível 2: Planejado e Monitorado.

	Característica Comum 2.1: Planejando a eficiência
2.1.1	Alocar Recursos. Alocar recursos adequados (incluindo pessoas) para executar o processo
2.1.2	Atribuir Responsabilidades. Atribuir responsabilidades para o desenvolvimento de produtos e/ou serviços do processo
2.1.3	Documentar o Processo. Documentar a abordagem de execução do processo em padrões ou procedimentos
2.1.4	Propiciar Ferramentas. Propiciar ferramentas apropriadas para suportar a execução do processo

(Continua)

TABELA 3.4 – Conclusão.

2.1.5	Assegurar Treinamento. Garantir que os indivíduos que realizam o processo estarão propriamente treinados para realizá-lo
2.1.6	Planejar o Processo. Planejar a eficiência do processo
	Característica Comum 2.2. Eficiência Disciplinada
2.2.1	Usar planos, Padrões e Procedimentos. Usar planos documentados, padrões e/ou procedimentos quando implementar o processo
2.2.2	Executar o Gerenciamento da Configuração. Colocar os produtos resultantes do processo sobre controle de versão ou gerência da configuração conforme o apropriado.
	Característica Comum 2.3: Verificar a Eficiência
2.3.1	Verificar a Conformidade do Processo. Verificar a conformidade do processo com relação a padrões e procedimentos
2.3.2	Auditar Produtos. Verificar a conformidade dos produtos com relação a padrões e/ou requisitos
	Característica Comum 2.4: Monitorar a Eficiência
2.4.1	Monitorar com Medições. Monitorar o status do processo de acordo com o plano, utilizando medições
2.4.2	Tomar Medidas Corretivas. Tomar medidas corretivas quando o progresso variar de forma significativa do planejado

TABELA 3.5 - Práticas genéricas para o nível 3: Bem Definido (Estabelecido).

	Característica Comum 3.1: Definir um Processo Padrão (Nível Organizacional)
3.1.1	Padronizar o Processo. Documentar o processo padrão ou família de processos para a organização, no qual se descreve como implementar uma prática básica para o processo.
3.1.2	Adequar o Processo Padrão. Adequar a família de processos organizacionais padrão para criar um processo definido o qual é direcionado a atender a uma necessidade específica da organização
	Característica Comum 3.2: Executar o Processo Definido
3.2.1	Utilizar um Processo Bem Definido. Utilizar um processo bem definido para implementar o processo. Um processo bem definido é aquele com entradas, critérios de entrada, tarefas, validação, saídas e critérios de saída bem documentados, consistente e completo.
3.2.2	Executar Revisões Cuidadas. Executar revisões cuidadosas em produtos resultantes da execução do processo.
3.2.3	Utilizar Dados Bem Definidos. Utilizar dados durante a execução do processo definido para permitir o gerenciamento do processo.

TABELA 3.6 - Práticas genéricas para o nível 4: Controlado e Quantificado.

	Característica Comum 4.1: Estabelecendo metas de qualidade mensuráveis
4.1.1	Estabelecer Metas de Qualidade. Estabelecer metas de medição da qualidade para os produtos resultantes dos processos, ou seja para todas as famílias de processos padrões da organização.
	Característica Comum 4.2: Gerenciar Eficiência de Forma Objetiva
4.2.1	Determinar a Capacitação do Processo. Determinar a capacitação do processo definido de forma quantitativa.
4.2.2	Utilizar a Capacitação do Processo. Utilizar ações corretivas, conforme apropriado, quando o processo não estiver sendo executado explorando a capacitação do processo.

TABELA 3.7 - Práticas genéricas para o nível 5: Melhoria Contínua.

	Característica Comum 5.1: Melhorando a Capacitação Organizacional
5.1.1	Estabelecer Metas de Eficiência do Processo. Estabelecer metas quantitativas para melhorar a eficiência do processo da família de processos padrões, com base nas metas de negócio da organização e no atual nível de capacitação.
5.1.2	Melhorar de Forma Contínua o Processo Padrão. Melhorar continuamente o processo, modificando a família de processos padrão organizacionais para incrementar a eficiência

(Continua)

TABELA 3.7 - Conclusão

	Característica Comum 5.2: Melhorando a Eficiência do Processo
5.2.1	Realizar Análise Causal. Realizar análise causal de defeitos
5.2.2	Eliminar Causas do Defeito. Eliminar as causas de defeito no processo definido de forma seletiva
5.2.3	Melhorar Continuamente o Processo Definido. Melhorar continuamente a eficiência do processo, alterando o processo definido para melhorar a eficiência

As práticas básicas são organizadas em uma hierarquia, primeiro, através de categoria de processo, então através do processo. Todas as categorias de processos, e práticas básicas possuem um nome e descrição.

As práticas básicas seguem a descrição de processo. Cada prática básica é numerada para permitir a referência e a fácil identificação. O número e nome da prática básica aparecem em negrito, em seguida, apresenta-se uma descrição da prática. Às vezes, apresenta-se uma nota que provê um exemplo, uma explicação mais detalhada, ou uma nota de referência.

Os processos de gerenciamento de projetos, segundo as definições SPICE, são descritos a seguir com maior detalhe, pois este conjunto de processos será analisado oportunamente durante este trabalho.

Categoria de Processo Projeto (PRO)

A categoria de processo **Projeto** consiste de processos que estabelecem o projeto, coordenam e gerenciam seus recursos para produzir um produto ou fornecer serviços que satisfaçam o cliente.

A entrada é um contrato ou acordo para fazer o trabalho. O foco nesta categoria de processo está na utilização efetiva de recursos (tempo, esforço, pessoas, custos) para acompanhar o propósito e objetivos do projeto.

Os processos pertencentes à categoria do processo Projeto são:

PRO.1 - Planejar o ciclo de vida do projeto

O propósito de Planejar o processo do ciclo de vida do projeto é estabelecer um modelo de ciclo de vida do software apropriado para o projeto .

A entrada para este processo é um acordo ou contrato para desenvolver um produto de software, que identifica os objetivos do projeto com relação a qualidade, custo, e/ou programação do produto.

A saída deste processo é um modelo de ciclo de vida do software com descrições das atividades e tarefas do software a serem executadas pelo projeto e a identificação dos controles do projeto.

PRO.1.1 - Avaliar opções para o desenvolvimento do produto. Avaliar as opções para desenvolvimento do produto, identificar os riscos associados a cada um.

PRO.1.2 - Selecionar o modelo de ciclo de vida do software. Selecionar um modelo de ciclo de vida para o projeto que seja adequado para o escopo, magnitude, e complexidade do projeto.

PRO.1.3 - Descrever as tarefas e atividades. Descrever as atividades do software do projeto e suas tarefas associadas, e o propósito de cada uma.

PRO.1.4 - Estabelecer a seqüência da tarefa. Estabelecer a seqüência da tarefa do software ou ordenação parcial dentro do ciclo de vida do software identificado.

PRO.1.5 - Documentar atividades. Identificar e documentar as atividades e tarefas de desenvolvimento do software, incluindo entradas, saídas, e interfaces.

PRO.2 - Estabelecer plano de projeto

O propósito do processo Estabelecer o Plano de Projeto é estabelecer planos razoáveis para executar a engenharia de software e formar uma base para gerenciar o projeto do software.

Plano do projeto tipicamente documenta os:

- objetivos e propósito do projeto.
- produtos de trabalho a ser desenvolvido.
- modelo de ciclo de vida do software.
- estimativas de software.
- riscos do projeto e planos de mitigação (atenuação).
- programação.
- recursos alocados às atividades do projeto.

As entradas para este processo podem incluir o modelo de ciclo de vida do projeto produzido pelo processo, "Planejar o ciclo de vida do projeto" PRO.1.

PRO.2.1 - Desenvolver *Work Breakdown Structure* - WBS. Desenvolver uma WBS relacionando as tarefas e seqüência do projeto com os recursos exigidos para acompanhá-los.

PRO.2.2 - Identificar os padrões do projeto. Identificar os padrões de software que irão guiar as atividades de software do projeto, consistente com as necessidades do projeto.

PRO.2.3 - Identificar as instalações especializadas. Identificar quaisquer ferramentas, equipamentos, ou salas entre aquelas normalmente disponíveis que serão necessárias para atender qualquer exigência técnica do projeto.

PRO.2.4 - Determinar a estratégia de reutilização. Identificar as oportunidades para reutilização, analisar seus impactos, e determinar a estratégia para reutilização.

PRO.2.5 - Desenvolver estimativas do projeto. Desenvolver estimativas das quais seja necessário satisfazer os requisitos do software para o ciclo de vida completo do software. Os parâmetros para estimativa incluem: tamanho, esforço, custo, programação, recursos.

PRO.2.6 - Identificar os riscos iniciais do projeto. Identificar, avaliar e documentar um conjunto inicial (ou linha base) de riscos do projeto e seus planos de mitigação.

PRO.2.7 - Identificar as medidas do projeto. Identificar o conjunto básico do *status* e outras medidas que serão utilizadas para rastrear o progresso do projeto e ajudar a determinar se o projeto está atendendo seus objetivos.

PRO.2.8 - Estabelecer a programação do projeto. Estabelecer a programação do projeto, baseando-se no modelo de ciclo de vida do software, WBS, estimativas, e planos de mitigação de risco.

PRO.2.9 - Estabelecer compromissos do projeto. Estabelecer compromissos para as estimativas e planos com todos os grupos afetados.

PRO.2.10 - Documentar planos de projeto. Documentar os resultados das atividades neste processo dentro dos planos do projeto.

PRO.3 - Constituição das equipes de projeto

O propósito do processo de Constituição de Equipes de Projeto é estabelecer equipes de projeto com membros qualificados que possam cumprir suas

responsabilidades em sua equipe e trabalhar juntamente como um grupo coerente.

O termo “equipe” pretende cobrir um número de diferentes tipos e durações de equipes, incluindo:

- equipes mistas consistindo de representantes do fornecedor e do cliente;
- equipes de revisão, equipes de desenvolvimento, equipes de análise de problema, equipes de aperfeiçoamento do processo, etc;
- equipes que existam por uma curta duração para resolver um problema específico, ou por longo prazo como uma parte estabelecida do ambiente.

PRO.3.1 - Definir as equipes do projeto. Definir as equipes que serão necessárias para executar o projeto, definindo a estrutura e operando regras para a equipe, habilidades e conhecimentos exigidos.

PRO.3.2 - Capacitar as equipes do projeto. Capacitar as equipes para executar seu trabalho, assegurando que elas tenham:

- um entendimento de seu trabalho;
- uma visão ou senso compartilhado de interesse comum;
- mecanismos ou instalações adequadas para comunicação e trabalho;
- suporte a partir do gerenciamento adequado para que elas sejam treinadas para executar o trabalho.

PRO.3.3 - Manter as interações da equipe de projeto. Obter e manter o contrato na implementação das interações entre as equipes.

PRO.3.4 - Gerenciar os problemas entre equipes. Identificar, rastrear e resolver os problemas que afetam o progresso de mais de uma equipe ou que ameacem a unidade do projeto.

PRO.4 - Gerenciar requisitos

O propósito do processo de Gerenciar Requisitos é estabelecer uma linha base dos requisitos de software, que serve como base para o trabalho do software do projeto, produtos, e atividades; e gerenciar alterações para aquela linha base.

PRO.4.1 - Concordar com os requisitos. Obter acordo com as equipes sobre as exigências do cliente, obtendo os aceites pelos representantes de todas as equipes e de outras partes ligadas contratualmente para trabalhar para esses requisitos.

PRO.4.2 - Estabelecer a linha base das exigências do cliente. Documentar as exigências do cliente e estabelecer uma linha base para a utilização do projeto.

PRO.4.3 - Gerenciar as alterações das exigências do cliente. Gerenciar todas as alterações feitas nas alterações do cliente para assegurar que essas que serão afetadas pelas alterações sejam capazes de avaliar os riscos e impacto, e iniciar o controle da alteração apropriada e ações de mitigação.

PRO.4.4 - Utilizar as exigências do cliente. Utilizar as exigências do cliente como base para planos do projeto de software, especificações das exigências e atividades e produtos de trabalho.

PRO.4.5 - Manter a rastreabilidade. Estabelecer e manter completamente a rastreabilidade das exigências para os produtos de trabalho do projeto do ciclo de vida do software.

PRO.5 - Gerenciar qualidade

O propósito do processo Gerenciar Qualidade é gerenciar a qualidade dos produtos e serviços do projeto para assegurar que o resultado dos produtos e serviços satisfaçam o cliente.

Gerenciar qualidade envolve identificar as características de qualidade exigidas dos produtos do projeto, trabalhando para alcançar esta qualidade, e demonstrando que esta qualidade foi alcançada.

As entradas são as exigências do cliente e os elementos selecionados dos planos do projeto do software (processo PRO.2). As saídas deverão ser integradas nos planos do projeto do software.

PRO.5.1 - Estabelecer as metas de qualidade. Baseado nas exigências do cliente para qualidade, estabelecer metas de qualidade para vários pontos de verificação dentro do ciclo de vida do software do projeto.

PRO.5.2 - Definir as métricas de qualidade. Definir as métricas que medem os resultados das atividades do projeto para ajudar a avaliar se as metas relevantes de qualidade foram alcançadas.

PRO.5.3 - Identificar as atividades de qualidade. Para cada meta da qualidade, identificar as atividades que ajudarão a conquistar essa meta da qualidade e integrar essas atividades no modelo de ciclo de vida do software.

PRO.5.4 - Executar as atividades de qualidade. Executar as atividades de qualidade identificadas.

PRO.5.5 - Avaliar a qualidade. Nas verificações dos pontos identificados dentro do ciclo de vida do software, aplicar as métricas definidas de qualidade para avaliar se as metas de qualidade relevante foram alcançadas.

PRO.5.6 - Tomar ação corretiva. Ao aplicar as metas de qualidade que não foram alcançadas, tomar a ação corretiva.

PRO.6 - Gerenciar riscos

O propósito do processo Gerenciar Riscos é identificar continuamente e mitigar completamente os riscos em um ciclo de vida de um projeto.

Gerenciar riscos envolve identificar continuamente os novos riscos, trabalhando efetivamente para mitigar esses riscos, e avaliar o sucesso dos esforços de mitigação de risco.

Uma base de experiências nos outros processos são adequadas para serem mais efetivas se este processo for amplamente executado.

PRO.6.1 - Estabelecer o escopo do gerenciamento de risco. Determinar o escopo do gerenciamento de risco a ser executado para este projeto, incluindo a severidade, a probabilidade, e o tipo de riscos para identificar e gerência.

PRO.6.2 - Identificar os riscos. Identificar os riscos para o projeto e como eles se desenvolvem.

PRO.6.3 - Analisar e priorizar os riscos. Avaliar a probabilidade de ocorrência, impacto, *time-frame*, causas e inter-relacionamentos de riscos para determinar a prioridade em aplicar os recursos para atenuar esses riscos.

PRO.6.4 - Desenvolver as estratégias de mitigação. Definir as estratégias apropriadas para atenuar cada risco ou grupo de riscos relacionados.

PRO.6.5 - Definir as métricas de risco. Para cada risco (ou grupo de riscos relacionados) definir as métricas que medem a alteração no estado do risco (probabilidade, impacto, *time-frame*) e o progresso das atividades de mitigação (atenuação).

PRO.6.6 - Implementar as estratégias de mitigação. Realizar as estratégias definidas de mitigação.

PRO.6.7 - Avaliar os resultados de estratégias de atenuação. Nos pontos identificados de verificação, aplicar as métricas definidas para avaliar o progresso esperado e o nível de sucesso das estratégias de mitigação.

PRO.6.8 - Tomar ação corretiva. Quando o progresso esperado não for alcançado, tomar a ação corretiva.

PRO.7 - Gerenciar os recursos e a programação

O propósito do processo de Gerenciar os Recursos e a Programação é coordenar e gerenciar os recursos do projeto e programar durante a vida do projeto para a finalização do alcance dos objetivos do projeto e os requisitos do software.

Os planos do projeto desenvolvidos no processo PRO.2 são entradas para este processo.

PRO.7.1 - Adquirir recursos. Alocar e distribuir os recursos suficientes e apropriados para as atividades do projeto de software, incluindo os recursos técnicos e de gerenciamento.

PRO.7.2 - Rastrear progresso. Comparar regularmente e informar o *status* do projeto em comparação com os planos do projeto.

PRO.7.3 - Conduzir as revisões de gerenciamento. Regularmente e nos *milestones* principais, conduzir as revisões de gerenciamento para discutir e avaliar:

- *status* em comparação com os planos do projeto e a programação;
- status dos riscos do software;
- estar de acordo com os padrões apropriados;
- disposição para os próximos passos de desenvolvimento.

PRO.7.4 - Conduzir as revisões técnicas. Regularmente e nos principais *milestones*, conduzir as revisões técnicas para discutir e avaliar: o status

técnico em comparação com os planos, problemas técnicos ainda não resolvidos e disposição técnica para os próximos passos de desenvolvimento.

PRO.7.5 - Gerenciar compromissos. Gerenciar os compromissos para as estimativas e planos com todos os grupos afetados, tomando ação quando apropriado.

PRO.8 -Gerenciar os sub contratados

O propósito do processo Gerenciar Sub Contratados é selecionar os sub-contratados qualificados e gerenciar seu desempenho.

As experiências básicas deste processo ajudam a assegurar que o fornecedor e os sub-contratados do fornecedor têm o mesmo entendimento dos objetivos do projeto e exigências do cliente e como eles irão trabalhar para conjuntamente atender esses objetivos com sucesso.

PRO.8.1 - Estabelecer a instrução de trabalho. Estabelecer uma instrução do trabalho a ser executado no sub-contrato.

PRO.8.2 - Qualificar os sub contratados potenciais. Qualificar os sub contratados potenciais através de uma avaliação de sua capacidade para executar a função de software exigida.

PRO.8.3 - Selecionar o sub contratado. Selecionar os sub contratados qualificados para executar as partes definidas do contrato.

PRO.8.4 - Estabelecer e gerenciar os compromissos. Estabelecer e gerenciar os compromissos para o sub contratado.

PRO.8.5 - Manter as comunicações. Intercambiar regularmente as informações sobre o progresso técnico com o sub contratado para manter um entendimento comum do trabalho que está sendo executado.

PRO.8.6 - Avaliar a conformidade. Avaliar a conformidade do sub contratado em comparação com os padrões e procedimentos acordados.

PRO.8.7 - Avaliar a qualidade do sub contratado. Avaliar a qualidade dos produtos e serviços entregues do sub contratado para assegurar a perfeição, a precisão e a conformidade com os padrões e especificações.

3.3 CMMI

É um *framework* que acomoda múltiplas disciplinas e é flexível o suficiente para suportar duas representações diferentes, uma para verificar o nível de maturidade dos processos, outra para o nível de maturidade da organização em geral.

O CMMI (2001) é patrocinado pelo *Department of Defense* (DoD) americano e pela *National Defense Industrial Association* (NDIA), com colaborações da indústria, governo e pelo SEI (*Software Engineering Institute*) americano. Pode ser um modelo de maturidade não somente para engenheiros de sistema e de software, mas também para sistemas de desenvolvimento integrado de processo e produto, os *Integrated Product and Process Development* (IPPD) engenharia de sistemas e de software com fornecedores, os SS, *Supplier Sourcing*, e de engenharia que integre IPPD e SS (Rout, 2002).

O CMMI possui como um de seus objetivos básicos a integração dos diversos modelos, como o SW-CMM, SE-CMM, AS-CMM, People-CMM, IPD-CMM entre outros, eliminando inconsistências e reduzindo duplicações. Pretende também reduzir o custo da implementação dos modelos de processo-base de melhoria e aumentar a clareza e o entendimento através de:

- Terminologia comum;
- Estilo consistente;
- Regras de construção uniforme;

- Componentes comuns.

Pretende-se também garantir compatibilidade com a ISO/IEC 15504, através de dois modos de visão:

- *Staged*: abordagem semelhante ao SW-CMM (com novas KPAs e alterações nas existentes);
- *Continuos*: semelhante ao 15504, com áreas de processo independentes do nível de maturidade.

3.3.1 SW-CMM

O SW-CMM, desenvolvido pelo SEI, é o exemplo mais conhecido dos modelos *Staged*. O SW-CMM define cinco níveis seqüenciais de maturidade e as áreas-chave de processos para os níveis 2, 3, 4 e 5. Deste modo, para uma organização estar com o seu processo de desenvolvimento e manutenção de software no nível 3, este processo tem que satisfazer, caso apropriado, as áreas-chave de processo definidas nos níveis 2 e 3.

Os cinco níveis de maturidade definem uma escala ordinal para medir a maturidade de um processo de software da organização e para avaliar a sua capacidade de processo de software. Os níveis também ajudam uma organização a priorizar seus esforços de melhorias.

Nível de maturidade é um estágio evolutivo bem definido em busca de um processo de software maduro. Cada nível de maturidade fornece uma gama de fundamentos para a melhoria contínua do processo. Cada nível compreende um conjunto de objetivos de processos que, quando satisfeitos, estabilizam um componente importante do processo de software. Alcançando cada nível da estrutura de maturidade, estabelece-se diferentes componentes no processo de software, resultando em um crescimento na capacidade de processo da organização.

As caracterizações dos cinco níveis de maturidade, descritos a seguir, destacam as mudanças, realizadas em cada nível, no processo principal.

1) Inicial. O processo de software é caracterizado como *ad hoc* e até mesmo ocasionalmente caótico. Poucos processos são definidos e o sucesso depende de esforço individual.

2) Repetível. Os processos básicos de gestão de projeto são estabelecidos para acompanhar custo, cronograma e funcionalidade. A necessária disciplina do processo existe para repetir sucessos anteriores em projetos com aplicações similares.

3) Definido. O processo de software para as atividades de gestão e engenharia é documentado, padronizado e integrado em um processo de software padrão para a organização. Todos os projetos utilizam uma versão aprovada do processo de software padrão para desenvolver e manter software.

4) Gerenciado. Medidas detalhadas do processo de software e da qualidade do produto são realizadas. O processo e os produtos de software são quantitativamente compreendidos e controlados.

5) Em Otimização. A melhoria contínua do processo é propiciada pelo *feedback* quantitativo do processo e pelas idéias e tecnologias inovadoras.

O CMM é uma estrutura que representa um caminho para melhoria de processos, recomendada para organizações de software que desejam melhorar seus processos de software. Esta estrutura operacional do CMM é projetada para dar suporte às suas várias formas de utilização. Há, pelo menos, quatro modalidades de aplicação previstas para o CMM:

- As equipes de avaliação poderão utilizar o CMM para identificar pontos fortes e oportunidades de melhoria na organização.

- As equipes de avaliação poderão utilizar o CMM para identificar riscos na seleção de diferentes prestadores de serviço, tanto para a concessão de negócios, quanto para o monitoramento de contratos.
- Gerentes e técnicos poderão utilizar o CMM para entender as atividades necessárias ao planejamento e implementação de um programa de melhorias no processo de software de suas organizações.
- Os grupos de melhoria de processo, tais como o Grupo de Processos (SEPG), poderão utilizar o CMM como um guia para ajudá-los a definir e melhorar o processo de software de suas organizações.

Devido a essas diversas modalidades de uso, o CMM deve ser decomposto em detalhes suficientes para que recomendações reais sobre o processo possam ser derivadas a partir da estrutura dos níveis de maturidade. Essa decomposição também indica os processos e sua estrutura, que caracteriza a sua maturidade e capacidade.

Estrutura Interna dos Níveis de Maturidade

Cada nível de maturidade foi decomposto em partes menores. Com exceção do Nível 1, a decomposição de cada nível de maturidade estende-se desde um resumo de cada nível até sua definição operacional nas práticas-chave. Cada nível de maturidade é composto de várias áreas-chave de processo.

Cada área-chave de processo é organizada em cinco seções denominadas características comuns. As características comuns especificam as práticas-chave que, quando tratadas coletivamente, atingem as metas previstas para a área-chave de processo.

Níveis de Maturidade

Um nível de maturidade representa um estágio evolutivo bem definido com relação à obtenção de um processo de software maduro. Cada nível de maturidade indica um nível de capacidade de processo. Por exemplo, no Nível 2 a capacidade de processo de uma organização foi elevada de uma situação *ad-hoc* para uma situação disciplinada, por meio do estabelecimento de sólidos controles de gestão de projeto.

Áreas-chave de Processo

Com exceção do Nível 1, cada nível de maturidade é decomposto em várias áreas-chave de processo, que indicam as áreas nas quais uma organização deveria focar seus esforços para a melhoria de seu processo de software. As áreas-chave de processo identificam os assuntos que devem ser tratados para se obter um determinado nível de maturidade.

Cada área-chave de processo identifica um grupo de atividades relacionadas, as quais, quando executadas coletivamente, atingem um conjunto de metas consideradas importantes para aumentar a capacidade do processo. As áreas-chave de processo foram definidas para residir em um único nível de maturidade. A forma de atingir as metas de uma área-chave de processo pode diferir entre projetos, de acordo com as diferenças de domínios de aplicação ou de ambientes.

Todavia, todas as metas de uma área-chave de processo devem ser atingidas para que uma organização satisfaça essa área-chave. Quando as metas de uma área-chave são cumpridas de maneira continuada em um projeto, a organização pode ser considerada como tendo institucionalizado a capacidade do processo caracterizada por essa área-chave.

O adjetivo “chave” indica que existem áreas de processo (e processos) que não são imprescindíveis para a obtenção de um determinado nível de maturidade. O CMM não descreve em detalhe todas as áreas de processo que estão relacionadas com desenvolvimento e manutenção de software. Algumas áreas

de processo foram identificadas como determinantes da capacidade do processo, sendo essas descritas no CMM.

Embora outros assuntos afetem o desempenho de um processo, as áreas-chave de processo foram identificadas em função de sua eficácia e eficiência na melhoria da capacidade do processo de uma organização. Elas podem ser consideradas como sendo os requisitos para a obtenção de um nível de maturidade. Para se obter um certo nível de maturidade, as áreas-chave de processo daquele nível devem ser satisfeitas.

Para satisfazer uma área-chave de processo, cada uma das metas dessa área-chave deve ser satisfeita. As metas resumem as práticas-chave de uma área-chave de processo e podem ser utilizadas para determinar se uma organização, ou projeto, efetivamente implementou essa área-chave. As metas traduzem o escopo (abrangência), os limites e a intenção de cada área-chave de processo. As práticas específicas a serem executadas em cada área-chave de processo evoluirão de acordo com a obtenção, por parte da organização, de níveis mais elevados de maturidade. Por exemplo, a maioria das capacidades de estimativas de projeto descritas na área-chave de processo Planejamento de Projeto de Software, do Nível 2, deve evoluir para tratar os dados adicionais de projeto disponíveis nos Níveis 3, 4 e 5. A Gestão Integrada de Projeto, no Nível 3, é uma evolução do Planejamento de Projeto de Software e do Acompanhamento e Supervisão de Projeto, do Nível 2, à medida que o projeto é gerido utilizando um processo de software definido.

Características Comuns

Por conveniência, as áreas-chave de processo são organizadas em características comuns. As características comuns são atributos que indicam se a implementação e a institucionalização de uma área-chave de processo são eficazes, repetíveis e duradouras. As cinco características comuns são listadas a seguir:

Compromissos. Os compromissos descrevem as ações que a organização deve tomar para garantir que o processo seja estabelecido e que será duradouro. Tipicamente envolvem o estabelecimento das políticas organizacionais e o patrocínio da alta gerência.

Habilidades. As habilidades descrevem as pré-condições que devem existir no projeto ou na organização para se implementar de maneira competente o processo de software. Tipicamente envolvem recursos, estruturas organizacionais e treinamento.

Atividades. As atividades descrevem os papéis e os procedimentos necessários para a implementação de uma área-chave de processo. Tipicamente envolvem o estabelecimento de planos e procedimentos, a execução do trabalho, seu acompanhamento e a tomada de ações corretivas quando necessário.

Medições e Análises. As medições e análises descrevem a necessidade de se medir o processo e analisar as medições. Tipicamente incluem exemplos de medições que poderiam ser feitas para determinar a situação e a eficácia das atividades.

Verificação da implementação. A verificação da implementação descreve os passos para se garantir que as atividades sejam executadas em conformidade com o processo estabelecido. Tipicamente engloba revisões e auditorias pela gerência e pela garantia da qualidade de software.

Nas características comuns, as práticas Atividades descrevem o que deve ser implementado para se estabelecer a capacidade do processo. As outras práticas, como um todo, formam a base na qual a organização pode institucionalizar as práticas descritas na característica comum Atividades.

Práticas-chave

Cada área-chave de processo é descrita em termos de práticas-chave que contribuem para satisfazer suas metas. As práticas-chave descrevem a infraestrutura e atividades que mais contribuem para a implementação e institucionalização eficazes da área-chave de processo.

Cada prática-chave consiste de uma sentença simples, freqüentemente seguida por uma descrição mais detalhada, que pode incluir exemplos e explicações mais detalhadas. Estas práticas-chave, também referidas como as práticas-chave de alto nível, estabelecem as políticas, procedimentos e atividades fundamentais para a área-chave de processo. Os componentes da descrição detalhada são freqüentemente referidos como sub práticas.

Para garantir uma realização consistente da meta de documentar planos para o planejamento e acompanhamento do projeto, a organização deve estabelecer um procedimento documentado para produzir estimativas de tamanho de software. Se não forem desenvolvidas a partir de um procedimento documentado, essas estimativas podem variar sem controle, já que diferenças nas suposições de tamanho nunca são explicitadas. A descrição detalhada do que seria esperado num procedimento como esse inclui o uso de dados históricos, suposições documentadas e revisão de estimativas. Esses critérios guiam o julgamento quando um procedimento razoável de estimativas de tamanho é seguido.

As práticas-chave descrevem “o que” deve ser feito, mas não deveriam ser interpretadas de forma obrigatória sobre “como” as metas deveriam ser alcançadas.

Práticas alternativas podem completar as metas da área-chave de processo. As práticas-chave deveriam ser interpretadas racionalmente para julgar se as metas da área-chave de processo são efetivamente alcançadas. As práticas-chave estão contidas em *Key Practices of the Capability Maturity Model, Version 1.1* (Paulk et al., 1993), juntamente com uma orientação sobre sua implementação.

3.3.2 O Modelo *Continuos*

O modelo *Continuos* é composto por áreas de processos universais e fundamentais. Outra dimensão é composta por uma métrica para a avaliação da capacidade de cada processo em uma organização. O exemplo mais conhecido de modelo contínuo é o da ISO/IEC 15504.

Na visão *Continuos*, qualquer área de processo pode ter níveis de capacidade entre 0 e 5 (Incompleto, Realizado, Gerenciado, Definido, Quantitativamente Gerenciado e Otimizado). Para isso, utiliza-se de duas dimensões: objetivos e práticas genéricas (*Generic Practices*, GP's), associadas aos níveis e dissociadas das áreas de processo, objetivos e práticas específicas (*Specific Practices*, SP's), associadas às áreas de processo e dissociadas dos níveis.

As áreas de processo, semelhante às categorias de processo da ISO/IEC 12207, são construídas em blocos.

Cada objetivo específico possui um número específico de práticas, que quando realizadas alcançarão um objetivo específico:

- Gerenciamento do Processo;
- Gerenciamento do Projeto;
- Engenharia;
- Suporte.

As práticas genéricas para o Nível 2 são propostas (Côrtes, 2001), na Tabela 3.8.

TABELA 3.8 - Práticas genéricas.

Nível	Prática Genérica
1	GP 1.1 Realizar práticas bases.
2	GP2.1 Estabelecer uma política organizacional para o planejamento e execução do processo.
	GP2.2 Planejar o processo.
	GP2.3 Prover os recursos necessários para a execução do processo.
	GP2.4 Definir e atribuir responsabilidades.
	GP2.5 Providenciar o treinamento necessário para as pessoas executarem o processo.
	GP2.6 Gerenciar configurações e versões de produtos de trabalho selecionados.
	GP2.7 Monitorar e controlar a execução do processo.
	GP2.8 Verificar a conformidade da execução do processo e dos produtos de trabalho a procedimentos e padrões.
	GP2.9 Submeter à análise gerencial os resultados e atividades do processo.
3	GP3.1 Estabelecer um processo definido.
	GP3.2 Coletar informações de melhoria.

(Continua)

TABELA 3.8 – Conclusão.

4	GP4.1 Estabelecer objetivos quantitativos para o processo.
	GP4.2 Realizar sub processos de forma estabilizada.
5	GP5.1 Garantir melhoria contínua do processo.
	GP5.2 Identificar a verdadeira causa de problemas.

As práticas específicas para planejamento, no *Continuos*, agrupadas por objetivo, são apresentadas na Tabela 3.9:

TABELA 3.9 - Práticas específicas para planejamento.

Objetivos	Práticas Específicas
1- Estimativas	SP1: Estabelecer tarefas e responsabilidades.
	SP2: Fazer estimativas.
	SP3: Determinar esforço e custo.
2-Elaborar planos	SP1: Ciclo de vida.
	SP2: Cronograma.
	SP3: Planos subordinados.
	SP4: Riscos.
	SP5: Capacitações.
	SP6: Coleta de dados da execução.

(Continua)

TABELA 3.9 – Conclusão.

	SP7: Estabelecer e manter planos.
3 - Comprometimento	SP1: Acomodar plano aos recursos.
	SP2: Individual e organizacional.
	SP3: Dos envolvidos.

Para avaliação do Processo de Software, utilizou-se o método padrão proposto pelo SEI:

- Seleção da equipe de avaliação;
- Aplicação do questionário de maturidade;
- Análise das respostas;
- Entrevistas e revisões dos documentos;
- Avaliação baseada no modelo e método proposto pelo CMMI;
- Quadro de verificação das áreas-chave de processo.

3.3.3 Processos de Gerenciamento de Projetos do CMMI

Os processos específicos de gerência de projetos, que são os processos de interesse para este trabalho, são descritos mais detalhadamente a seguir:

Escopo dos Processos de Gerenciamento de Projeto

As áreas do processo de gerenciamento de projeto abrangem as atividades de gerenciamento de projeto relacionadas ao planejamento, monitoramento e controle do projeto. As áreas do processo de gerenciamento de projeto do CMMI são as seguintes:

- Planejamento do Projeto;
- Controle e Monitoramento do Projeto;
- Gerenciamento de Contrato do Fornecedor;
- Gerenciamento de Projeto Integrado;
- Gerenciamento de Risco;
- Gerenciamento de Projeto Quantitativo.

Para descrever as interações entre as áreas do processo de gerenciamento de projeto, é mais usual endereçá-los em dois grupos de área de processo:

- As áreas do processo de gerenciamento de projeto “básicas” são Planejamento do Projeto, Controle e Monitoramento do Projeto, e Gerenciamento de Contrato do Fornecedor.
- As áreas do processo de gerenciamento de projeto “avançadas” são Gerenciamento de Projeto Integrado, Gerenciamento de Risco, e Gerenciamento de Projeto Quantitativo.

Áreas básicas do processo de gerenciamento de projeto

As áreas básicas do processo de gerenciamento de projeto mapeiam as atividades básicas relacionadas à estabilização e manutenção do plano de projeto, estabelecem e mantêm os compromissos, monitoram o progresso em comparação com o plano, tomam ação corretiva, e gerenciam contratos de fornecedor.

O planejamento começa com os requisitos que definem o produto e o projeto (O que Construir). O plano de projeto abrange as várias atividades de engenharia e gerenciamento do projeto que serão executadas pelo projeto. O projeto irá revisar os planos subordinados de vários grupos de suporte e

estabelecer compromissos com esses grupos para suas contribuições ao projeto. Esses planos do grupo de suporte abrangem as avaliações do produto e processo, gerenciamento de configuração, e medição e análise.

A área de processo de controle e monitoramento do projeto inclui as atividades de monitoramento e tomada de ação corretiva. O plano de projeto especifica o nível adequado de monitoramento do projeto, a frequência das revisões de progresso, e as medidas utilizadas para monitorar o progresso. O progresso é determinado primeiramente pela comparação do progresso com o plano. Quanto o status atual se desvia significativamente dos valores esperados, as ações corretivas são tomadas conforme apropriado. Essas ações podem incluir replanejamento.

A área de processo de gerenciamento de contrato de fornecedor encaminha a necessidade do projeto para selecionar efetivamente e gerenciar essas partes do trabalho que são produzidas pelos fornecedores. Uma vez que um componente do produto seja definido e o fornecedor que irá produzi-lo seja selecionado, um contrato do fornecedor é estabelecido e mantido o qual será utilizado para gerenciar o fornecedor. O desempenho e o progresso do fornecedor são monitorados.

Os testes e as revisões da aceitação são conduzidos no componente do produto produzido pelo fornecedor.

As áreas avançadas do processo de gerenciamento de projeto

As áreas avançadas do processo de gerenciamento de projeto endereçam atividades tais como estabelecer um processo definido, que é feito sob medida a partir do conjunto de padrões dos processos da organização; coordenar e colaborar com os *stakeholders* relevantes, gerenciamento de risco, e gerenciar quantitativamente o processo definido do projeto.

Cada uma das áreas avançadas do processo de gerenciamento de projeto é fortemente dependente da capacidade para o planejamento, monitoramento e controle do projeto. (As áreas básicas do processo de gerenciamento de projeto fornecem essa capacidade).

A área Processo de Gerenciamento de Projeto Integrado estabelece e mantém o processo definido do projeto que é feito sob medida a partir do conjunto dos processos padrão da organização. O projeto é gerenciado utilizando-se o processo definido do projeto. O projeto utiliza e contribui para o processo da organização e avaliações de suporte.

O projeto assegura que os *stakeholders* relevantes associados ao projeto coordenem seus esforços de maneira adequada.

Embora o monitoramento e a identificação de risco estejam cobertos nas áreas do processo de Planejamento do Projeto e de Controle e Monitoramento do Projeto, a área do processo de gerenciamento de risco toma uma abordagem avançada mais contínua para gerenciar atividades de risco que incluem identificação de taxonomias e parâmetros de risco; avaliações de risco; e controle de risco.

Os objetivos do desempenho do processo e qualidade para o projeto são baseados naqueles estabelecidos pela organização. O processo definido do projeto estabelecido no gerenciamento do projeto integrado é compreendido, em parte, de elementos do processo e sub processos cujo desempenho do processo pode ser prognosticado. No mínimo, a variação do processo experimentada pelos sub processos, que é crítica ao alcançar os objetivos de desempenho do processo e qualidade do projeto, é subentendida.

A ação corretiva é tomada quando causas especiais da variação são identificadas.

A seguir serão apresentados os processos de gerenciamento de projetos e respectivos componentes.

PLANEJAMENTO DO PROJETO

O propósito do Planejamento do Projeto é estabelecer e manter planos que definam as atividades do projeto.

SG 1 Estabelecer Estimativas

Estimativas dos parâmetros de planejamento de projeto são estabelecidas e mantidas.

SP 1.1-1 Estimar o Escopo do Projeto

Estabelecer e manter uma WBS de alto nível para estimativa do escopo do projeto.

SP 1.2-1 Estabelecer Estimativas de Atributos do Projeto

Estabelecer e documentar as estimativas do documento dos atributos das tarefas e produtos do trabalho.

SP 1.3-1 Definir o Ciclo de Vida do Projeto

Definir as fases do ciclo de vida do projeto para fazer o escopo do esforço de planejamento.

SP 1.4-1 Determinar as Estimativas de Esforço e Custo

Estimativa do esforço e custo do projeto para os atributos das tarefas e produtos do trabalho baseados em estimativa e análise racional.

SG 2 Desenvolver um Plano de Projeto

Um plano de projeto é estabelecido e mantido conforme a base para gerenciar o projeto.

SP 2.1-1 Estabelecer Receita (Orçamento) e Programação

Estabelecer e manter a receita e programação do projeto.

SP 2.2-1 Identificar os Riscos do Projeto

Identificar e analisar os riscos do projeto.

SP 2.3-1 Planejar o Gerenciamento de Dados

Planejar o gerenciamento dos dados do projeto.

SP 2.4-1 Planejar os Recursos do Projeto

Planejar os recursos necessários para executar o projeto.

SP 2.5-1 Planejar as Habilidades e Conhecimentos Necessários

Planejar as habilidades e conhecimentos necessários para executar o projeto.

SP 2.6-1 Planejar o Envolvimento de *Stakeholder*

Planejar o envolvimento com os *stakeholder* identificados.

SP 2.7-1 Estabelecer o Plano de Projeto

Estabelecer e manter o conteúdo global do plano de projeto.

SG 3 Obter Compromisso para o Plano

Os compromissos são estabelecidos e mantidos para o plano de projeto.

SP 3.1-1 Rever os Planos Subordinados

Rever os planos subordinados para entender os compromissos do projeto.

SP 3.2-1 Reconciliar os Níveis de Recurso e Trabalho

Reconciliar o plano de projeto para refletir os recursos projetados e disponíveis.

SP 3.3-1 Obter o Compromisso do Plano

Obter o compromisso dos *stakeholders* relevantes responsáveis por executar e suportar a execução do plano.

CONTROLE E MONITORAMENTO DO PROJETO

O propósito do Controle e Monitoramento do Projeto é fornecer entendimento no progresso do projeto para que as ações corretivas adequadas sejam tomadas quando o desempenho do projeto se desviar significativamente do plano.

SG 1 Monitorar o Projeto em Comparação com o Plano

O progresso e o desempenho atual do projeto é monitorado em comparação com o plano de projeto.

SP 1.1-1 Monitorar os Parâmetros do Planejamento do Projeto

Monitorar os valores atuais dos parâmetros do planejamento do projeto em comparação com o plano de projeto.

SP 1.2-1 Monitorar Compromissos

Monitorar os compromissos em comparação com aqueles identificados no plano de projeto.

SP 1.3-1 Monitorar Riscos de Projeto

Monitorar riscos em comparação com os identificados no plano de projeto.

SP 1.4-1 Monitorar Gerenciamento de Dados

Monitorar o gerenciamento de dados do projeto.

SP 1.5-1 Monitorar o Envolvimento de *Stakeholder*

Monitorar o envolvimento de *stakeholder* em comparação com o plano de projeto.

SP 1.6-1 Conduzir Revisões de Progresso

Revisar periodicamente os problemas, desempenho e progresso do projeto.

SP 1.7-1 Conduzir Revisões *Milestone*

Rever os acompanhamentos e resultados do projeto nos *milestones* do projeto selecionado.

SG 2 Gerenciar a Ação Corretiva para Encerrar

As ações corretivas são gerenciadas para encerrar quando o desempenho do projeto ou resultados se desviarem significativamente do plano.

SP 2.1-1 Analisar Problemas

Coletar e analisar os problemas e determinar as ações corretivas necessárias para mapear os problemas.

SP 2.2-1 Tomar Ação de Correção

Tomar ação corretiva nos problemas identificados.

SP 2.3-1 Gerenciar Ação Corretiva

Gerenciar as ações corretivas para encerramento.

GERENCIAMENTO DE CONTRATO DO FORNECEDOR

O propósito do Gerenciamento de Contrato do Fornecedor é gerenciar a aquisição de produtos e serviços dos fornecedores externos ao projeto para que exista um contrato formal.

SG 1 Estabelecer os Contratos de Fornecedor

Os contratos com os fornecedores são estabelecidos e mantidos.

SP 1.1-1 Analisar as Necessidades e Exigências Determinadas pelo Projeto

Analisar as exigências e necessidades do projeto que serão cumpridas (executadas) pelas fontes externas ao projeto para determinar como as necessidades e exigências serão atendidas.

SP 1.2-1 Selecionar os Fornecedores

Selecionar os fornecedores baseados na avaliação de sua capacidade de atender as exigências especificadas e os critérios estabelecidos.

SP 1.3-1 Estabelecer os Contratos dos Fornecedores

Estabelecer e manter os contratos formais com os fornecedores.

SG 2 Satisfazer os Contratos do Fornecedor

Os contratos com os fornecedores são atendidos tanto pelo projeto quanto pelo fornecedor.

SP 2.1-1 Adquirir os Produtos

Adquirir os produtos para satisfazer as exigências especificadas que são cobertas no contrato do fornecedor.

SP 2.2-1 Executar o Contrato do Fornecedor

Executar as atividades com o fornecedor conforme especificado no contrato do fornecedor.

SP 2.3-1 Conduzir o Teste de Aceitação

Assegurar que o contrato do fornecedor seja satisfeito antes de aceitar o produto adquirido.

SP 2.4-1 Transição de Produtos

A transição dos produtos adquiridos do fornecedor para o projeto.

GERENCIAMENTO DO PROJETO INTEGRADO

O propósito do Gerenciamento de Projeto Integrado é estabelecer e gerenciar o projeto e o envolvimento dos *stakeholders* relevantes de acordo com um processo integrado e definido que seja feito sob medida a partir de um conjunto de processos padrão da organização.

SG 1 Utilizar o Processo Definido do Projeto

O projeto é conduzido utilizando um processo definido que é feito sob medida a partir de um conjunto de processos padrão da organização.

SP 1.1-1 Estabelecer o Processo Definido do Projeto

Estabelecer e manter o processo definido do projeto.

SP 1.2-1 Utilizar as Avaliações do Processo Organizacional para Planejar Atividades do Projeto

Utilizar os componentes de medição repositória do processo da organização para estimar e planejar as atividades do projeto.

SP 1.3-1 Integrar Planos

Integrar o plano de projeto e os planos subordinados para descrever o processo definido do projeto.

SP 1.4-1 Gerenciar o Projeto Utilizando Planos Integrados

Gerenciar o projeto utilizando o plano do projeto, planos subordinados, e processo definido do projeto.

SP 1.5-1 Contribuir para as Avaliações do Processo da Organização

Contribuir com produtos de trabalho, medidas e experiências documentadas para as avaliações do processo da organização.

SG 2 Coordenar e colaborar com os *Stakeholders* Relevantes

Coordenar e colaborar com os *stakeholders* relevantes ao projeto.

SP 2.1-1 Gerenciar o Envolvimento do *Stakeholder*

Gerenciar o envolvimento dos *stakeholders* relevantes no projeto.

SP 2.2-1 Gerenciar Dependências

Participar com os *stakeholders* relevantes para identificar, negociar, e rastrear dependências críticas.

SP 2.3-1 Resolver Problemas de Coordenação

Resolver problemas com os *stakeholders* relevantes.

GERENCIAMENTO DE RISCO

O propósito do Gerenciamento de Risco é identificar os problemas potenciais antes que eles ocorram, para que as atividades de controle de risco possam ser planejadas e invocadas conforme necessário através do ciclo de vida para mitigar os impactos adversos no alcance dos objetivos.

SG 1 Preparo para o Gerenciamento de Risco

É conduzido o Preparo para o Gerenciamento de Risco.

SP 1.1-1 Determinar as Categorias e Fontes de Risco

Determinar as categorias e fontes de risco.

SP 1.2-1 Definir os Parâmetros de Risco

Definir os parâmetros utilizados para analisar e classificar riscos, e os parâmetros utilizados para controlar o esforço do gerenciamento de risco.

SP 1.3-1 Estabelecer uma Estratégia de Gerenciamento de Risco.

Estabelecer e manter a estratégia e métodos a serem utilizados para o Gerenciamento de Risco.

SG 2 Identificar e Analisar Riscos

Os riscos são identificados e analisados para determinar sua importância relativa.

SP 2.1-1 Identificar Riscos

Identificar e documentar os riscos.

SP 2.2-1 Avaliar, Classificar, e Priorizar Riscos

Avaliar e classificar cada risco identificado utilizando os parâmetros e categorias de risco identificados e determinar sua prioridade relativa.

SG 3 Mitigar Riscos

Os riscos são controlados e mitigados, onde apropriado, para reduzir os impactos adversos no alcance dos objetivos.

SP 3.1-1 Desenvolver Planos de Mitigação de Risco

Desenvolver um plano de mitigação de risco para os riscos mais importantes para o projeto, conforme definidos pela estratégia de gerenciamento de risco.

SP 3.2-1 Implementar Planos de Mitigação de Risco

Monitorar o *status* de cada risco periodicamente e implementar o plano de mitigação de risco conforme apropriado.

GERENCIAMENTO DE PROJETO QUANTITATIVO

O propósito da área do processo de gerenciamento de projeto quantitativo é gerenciar quantitativamente o processo definido do projeto para alcançar os objetivos de desempenho do processo e qualidade estabelecida do projeto.

SG 1 Gerenciar Quantitativamente o Projeto

O projeto é gerenciado quantitativamente utilizando os objetivos de desempenho do processo e qualidade.

SP 1.1-1 Estabelecer os Objetivos do Projeto

Estabelecer e manter os objetivos de desempenho do processo e qualidade do projeto.

SP 1.2-1 Compor o Processo Definido

Selecionar os processos e processar elementos que compreendam o processo definido do projeto com base na estabilidade da capacidade histórica e dados.

SP 1.3-1 Selecionar os Sub processos a serem Gerenciados

Selecionar sub processos do processo definido do projeto que serão gerenciados estatisticamente.

SP 1.4-1 Gerenciar o Desempenho do Projeto

Monitorar o projeto para determinar se os objetivos do projeto para desempenho do processo e qualidade serão atendidos e tomar ações corretivas conforme apropriado.

SG 2 Gerenciar Estatisticamente o Desempenho do Sub-processo

O desempenho dos sub processos selecionados dentro do processo definido do projeto é gerenciado estatisticamente.

SP 2.1-1 Selecionar as Técnicas Analíticas e Medidas

Selecionar as técnicas analíticas e medidas a serem utilizadas no gerenciamento estatístico dos sub processos selecionados.

SP 2.2-1 Aplicar os Métodos Estatísticos para Entender a Variação

Estabelecer e manter um entendimento da variância dos sub processos selecionados utilizando as técnicas analíticas e medidas.

SP 2.3-1 Monitorar o Desempenho dos Sub processos Selecionados

Monitorar o desempenho dos sub processos selecionados para determinar sua capacidade para satisfazer seus objetivos de desempenho do processo e qualidade, e tomar ação corretiva conforme necessário.

SP 2.4-1 Gravar os Dados do Gerenciamento Estatístico

Gravar os dados do gerenciamento da qualidade e estatístico no repositório de medição da organização.

3.4 PMBOK2000

O Universo do Conhecimento de Gerência de Projetos - PMBOK é uma denominação que representa todo o somatório de conhecimento dentro da profissão de gerência de projetos. Como qualquer outra profissão o conjunto de conhecimentos baseia-se na contribuição daqueles profissionais e estudantes que aplicam esses conhecimentos no dia-a-dia, desenvolvendo-os.

Este conjunto completo de conhecimentos em gerência de projetos inclui os conhecimentos já comprovados através de práticas tradicionais que são amplamente utilizadas, assim como conhecimentos de práticas mais

inovadores e avançadas que têm tido uma aplicação mais limitada, incluindo tanto material publicado ou não.

O PMBOK2000 (PMI, 2000), última versão do PMBOK, aborda e organiza os processos necessários ao gerenciamento de projetos dividindo-os em nove Áreas de Conhecimento.

As Áreas de Conhecimento da Gerência de Projetos, descrevem os conhecimentos e práticas em gerência de projetos em termos dos processos que as compõem. Essas áreas, bem como os processos, são descritos a seguir:

- Gerência da Integração do Projeto: descreve os processos necessários para assegurar que os diversos elementos do projeto sejam adequadamente coordenados. Ele é composto pelo desenvolvimento do plano de projeto, execução do plano do projeto e controle integrado de mudanças.
- Gerência do Escopo do Projeto: descreve os processos necessários para assegurar que o projeto contemple todo trabalho requerido, e nada mais que o trabalho requerido, a fim de completar o projeto com sucesso. Ele é composto pela iniciação, planejamento do escopo, detalhamento do escopo, verificação do escopo e controle de mudanças do escopo.
- Gerência do Tempo do Projeto: descreve os processos necessários para assegurar que o projeto termine dentro do prazo previsto. Ele é composto pela definição das atividades, o seqüenciamento das atividades, estimativa da duração das atividades, desenvolvimento do cronograma e controle do cronograma.
- Gerência do Custo do Projeto: descreve os processos necessários para assegurar que o projeto seja completado dentro do orçamento

previsto. Ele é composto pelo planejamento dos recursos, estimativa dos custos, orçamento dos custos e controle dos custos.

- Gerência da Qualidade do Projeto: descreve os processos necessários para assegurar que as necessidades que originaram o desenvolvimento do projeto serão satisfeitas. Ele é composto pelo planejamento da qualidade, garantia da qualidade e controle da qualidade.
- Gerência dos Recursos Humanos do Projeto: descreve os processos necessários para proporcionar a melhor utilização das pessoas envolvidas no projeto. Ele é composto pelo planejamento organizacional, montagem da equipe e desenvolvimento da equipe.
- Gerência das Comunicações do Projeto: descreve os processos necessários para assegurar que a geração, captura, distribuição, armazenamento e pronta apresentação das informações do projeto sejam feitas de forma adequada e no tempo certo. Ele é composto pelo planejamento das comunicações, distribuição das informações, relato de desempenho e encerramento administrativo.
- Gerência dos Riscos do Projeto: descreve os processos que dizem respeito à identificação, análise e respostas a riscos do projeto. Ele é composto pelo planejamento da gerência de riscos, análise qualitativa de riscos, análise quantitativa de riscos, desenvolvimento das respostas aos riscos e controle e monitoração de riscos.
- Gerência das Aquisições do Projeto: descreve os processos necessários para a aquisição de mercadorias e serviços fora da organização que desenvolve o projeto. Ele é composto pelo planejamento as aquisições, preparação das aquisições, obtenção de propostas, seleção de fornecedores, administração dos contratos e encerramento do contrato.

A implantação da abordagem do PMI (2000) torna-se interessante, pois os processos definidos pelo PMBOK são genéricos, ou seja, podem ser aplicados a qualquer tipo de projeto, de qualquer atividade ou ramo, além de permitir que sejam feitas adaptações e extensões para projetos de áreas específicas.

Sant'Anna (2000), por exemplo, usou a base estabelecida pelo PMI, ajustou e acrescentou alguns processos e áreas de conhecimento, para que este se adequasse melhor às necessidades específicas dos projetos de software. Projetos esses que sofrem influências de outras abordagens, metodologias e ciclos de vida peculiares à área de desenvolvimento de software.

3.5 ISO 10006

As diretrizes da ISO 10006 (ISO, 1997) e NBRISO 10006 (ABNT, 2000a) utilizam os processos de gerenciamento de projetos como estrutura para discussão de suas aplicações.

Esta norma internacional é aplicável a projetos de complexidade variada, pequenos ou grandes, de pequena ou longa duração, em diferentes ambientes e independentemente do tipo de produto do projeto (incluindo hardware, software, material processado, serviços ou suas combinações).

Estas diretrizes podem necessitar de algumas adaptações para que sejam aplicadas em um projeto particular.

Esta norma internacional não é, por ela mesmo, um guia para gerenciamento de projetos.

A seguir serão exibidos os processos e respectivas descrições segundo a estrutura desta norma.

PROCESSO ESTRATÉGICO

Processo estratégico

Define a direção do projeto e gerencia a realização de outros processos do projeto.

PROCESSOS DE GERENCIAMENTO DE INTERDEPENDÊNCIAS

Inicialização do projeto e desenvolvimento do plano de projeto

Avaliação dos requisitos do cliente e outras partes interessadas, preparando um plano de projeto e iniciando outros processos.

Gerenciamento das interações

Gerenciamento das interações durante o projeto.

Gerenciamento das mudanças

Antecipação a mudanças e gerenciamento destas ao longo de todos os processos.

Encerramento

Conclusão dos processos e obtenção de retroalimentação (*feedback*)

PROCESSOS RELACIONADOS AO ESCOPO

Desenvolvimento conceitual

Definição das linhas gerais sobre o que produto do projeto irá fazer.

Desenvolvimento e controle do escopo

Documentação das características do produto do projeto em termos mensuráveis e controle dos mesmos.

Definição das atividades

Identificação e documentação das atividades e etapas necessárias para se alcançar os objetivos do projeto.

Controle das atividades

Controle do trabalho efetivo realizado no projeto.

PROCESSOS RELACIONADOS AO TEMPO

Planejamento de dependência das atividades

Identificação das inter-relações, interações lógicas e dependências entre as atividades do projeto.

Estimativa de duração

Estimativa da duração de cada atividade em conexão com atividades específicas e com os recursos necessários.

Desenvolvimento do cronograma

Inter-relação dos objetivos de prazo do projeto, dependências das atividades e suas durações como estrutura para o desenvolvimento de cronogramas gerais e detalhados.

Controle do cronograma

Controle da realização das atividades do projeto para confirmação do cronograma proposto ou para realizar as ações apropriadas para recuperar atrasos.

PROCESSOS RELACIONADOS AO CUSTO

Estimativa de custos

Desenvolvimento de estimativa de custos para o projeto

Orçamento

Utilização de resultados provenientes da estimativa de custos para elaboração do orçamento do projeto.

Controle de custos

Controle de custos e desvios ao orçamento do projeto.

PROCESSOS RELACIONADOS AOS RECURSOS

Planejamento de recursos

Identificação, estimativa, cronograma e alocação de todos os recursos principais.

Controle dos recursos

Comparação da utilização real e planejada de recursos corrigindo, se necessário.

PROCESSOS RELACIONADOS ÀS PESSOAS

Definição de estrutura organizacional

Definição de uma estrutura organizacional para o projeto, baseada no atendimento das necessidades de projeto, incluindo a identificação das funções e definindo autoridades e responsabilidades.

Alocação da equipe

Seleção e nomeação de pessoal suficiente com a competência apropriada para atender as necessidades do projeto.

Desenvolvimento da equipe

Desenvolvimento de habilidades individuais e coletivos para aperfeiçoar o desempenho do projeto.

PROCESSOS RELACIONADOS À COMUNICAÇÃO

Planejamento da comunicação

Planejamento dos sistemas de informação e comunicação do projeto.

Gerenciamento das informações

Tornar disponíveis as informações necessárias da organização do projeto aos membros e outras partes interessadas.

Controle da comunicação

Controle da comunicação de acordo com o sistema de comunicações planejado.

PROCESSOS RELACIONADOS AO RISCO

Identificação de riscos

Determinação de riscos no projeto.

Avaliação de riscos

Avaliação da probabilidade de ocorrência de eventos de risco e o impacto destes sobre o projeto.

Desenvolvimento de reação ao risco

Desenvolvimento de planos para reação ao risco.

Controle de riscos

Implementação e atualização dos planos de risco.

PROCESSOS RELACIONADOS À AQUISIÇÃO

Planejamento e controle da compra

Identificação e controle do que deve ser adquirido e quando.

Documentação dos requisitos

Compilação das condições comerciais e requisitos técnicos.

Avaliação dos subcontratados

Avaliação e determinação de quais subcontratados devem ser convidados a fornecer produtos.

Subcontratação

Publicação dos convites à proposta, avaliação das propostas, negociação, preparação e assinatura do subcontrato.

Controle do contrato

Garantia de que o desempenho dos subcontratados atende aos requisitos contratuais.

3.6 *Unified Process Modelling* - UPM

O UPM - Modelo de Processo Unificado é uma proposta inicial de submissão da OMG para um modelo usado na descrição de um processo concreto de desenvolvimento de software ou uma família relacionada de processos de desenvolvimento de software. A execução do processo está fora da extensão do UPM.

Diferente da maioria das PMLs, o UPM aparece como uma proposta de unificação entre as diferentes metodologias propostas para modelagem de processos. Da mesma forma que a modelagem orientada a objetos despertou

uma série de propostas, o que acabou por ocasionar a chamada guerra dos modelos, a modelagem de processos está passando pelo mesmo problema. Organizações, pesquisadores, universidades, criam suas metodologias e nomeiam os elementos dos processos com diferentes termos, causando grande dificuldade para o desenvolvimento de produtos e da pesquisa na área de desenvolvimento de software.

O UPM utiliza uma aproximação orientada a objeto para modelar uma família relacionada de processos e usa a UML como notação. A Figura 3.2 mostra as quatro arquiteturas de modelagem definidas pelo OMG.

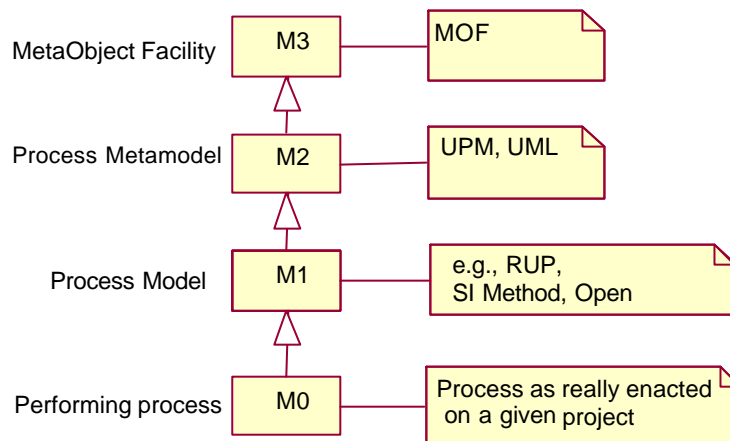


FIGURA 3.2 - Níveis de modelagem definidos pela OMG.

FONTE: OMG (2000).

O UPM encontra-se no nível M2 da arquitetura de quatro camadas e seu metamodelo é definido usando um subconjunto da UML de modo semelhante da UML com o MOF. Este subconjunto da UML corresponde às facilidades apoiadas pelo MOF. O metamodelo da UPM é largamente independente do metamodelo da UML, com a exceção do uso de Diagramas de Atividade.

Um processo executado é aquele que na produção no mundo real ou como o processo é ordenado, sendo nivelado no nível M0. A definição do processo correspondente está nivelado em M1. Por exemplo, o Processo Unificado Rational (Rational, 2000) ou o Método IBM SI estão definidos no nível M1.

Ambos, processos genéricos, como RUP e uma versão específica customizada deste processo usados por um determinado projeto, estão nivelados no M1. É focalizado aqui o meta-modelo que representa o nivelamento M2 e servindo como um *template* para o nível M1.

O UPM pode ser usado para definir todos os tipos de processos, incluindo os que estejam focalizados no uso específico da UML. Instâncias para orientar subclasses para descrever práticas com a UML podem ser criadas como ferramentas para um processo UML específico.

3.6.1 Escopo do UPM

O UPM é um metamodelo para definir processos e seus componentes. Uma ferramenta baseada no UPM seria uma ferramenta para autoria e customização de processo. A execução de um processo, planejar e executar um projeto usando um processo descrito com o UPM, não está no escopo deste modelo.

A proposta inicial da OMG limita-se a definir o conjunto mínimo de processo que modela elementos necessários para descrever qualquer processo de desenvolvimento de software, sem adicionar modelos específicos ou condições para qualquer área específica ou disciplina, como gerenciamento de projeto ou análise.

3.6.2 Elementos Principais

O metamodelo UPM é dividido em seis pacotes, chamados: Nomes, Elementos Básicos, Estrutura de Processo, Componentes de Processo, Orientação e Ciclo de Vida do Processo.

O pacote Elementos Básicos, detalhado na Figura 3.3, define os elementos básicos dos quais o resto do modelo é derivado.

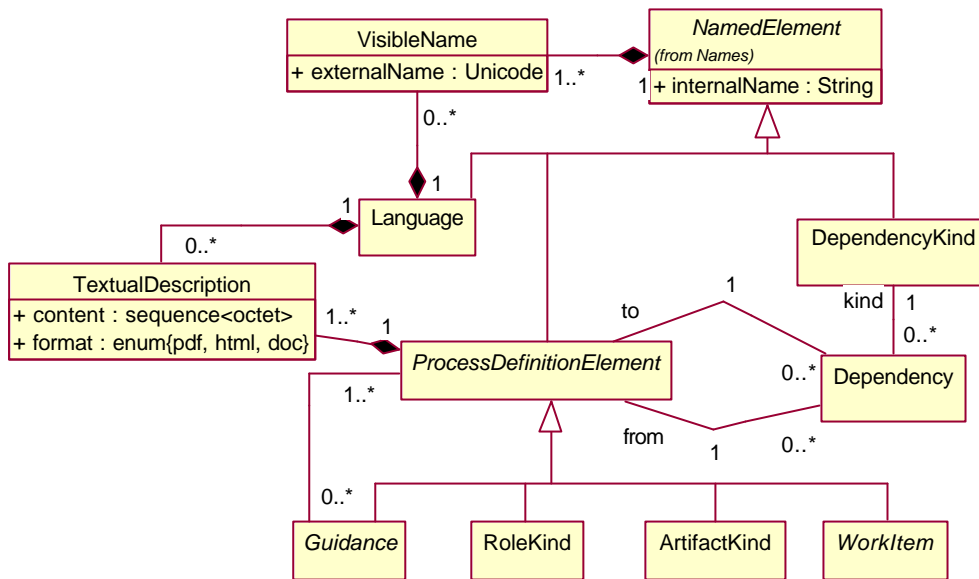


FIGURA 3.3 - Pacote de elementos básicos.

FONTE: OMG (2000).

Os principais elementos do pacote básico são:

- **NamedElement, VisibleName, e Language:** A maioria dos elementos do metamodelo de processo são uma especialização de uma classe abstrata comum, chamada *NamedElement*, com um único atributo: *internalName*. Isto permite que elementos tenham um nome textual pelo qual podem referir-se no modelo de processo. Para permitir a expressão de um processo em qualquer idioma natural, um elemento nomeado pode ser associado a qualquer número de *VisibleName* que contém uma *string* de Unicode do nome do elemento em um idioma natural. Uma associação para o idioma de classe especifica o idioma.
- **ProcessDefinitionElement e TextualDescription:** A maioria dos elementos do metamodelo de processo também são uma especialização de uma classe abstrata comum, chamada *ProcessDefinitionElement*, introduzida para capturar atributos comuns como uma descrição textual. Em particular, todos os elementos de processo têm um ou várias descrições em idiomas naturais. *WorkDefinition*, *ArtifactKind*, *RoleKind*,

e *Guidance* são principais subdivisões de classe de *ProcessDefinitionElement*.

- ***ArtifactKind* e *ArtifactName***: Um Artefato é qualquer coisa produzida, consumida ou modificada por um processo. Pode ser um pedaço de informação, um documento, um modelo, código de fonte, e assim por diante. Um *ArtifactKind* descreve um tipo de artefato.
- ***Guidance***: um ou mais elementos de *Guidance* são associadas com um elemento de processo para prover ajuda.
- ***WorkItem*, *WorkDefinition*, e *WorkDefinitionName***: Um *WorkItem* é um *ProcessDefinitionElement* que descreve o trabalho executado através de papéis. É uma classe abstrata, com duas subdivisões de classe: *WorkDefinition* e *Step*. *WorkItems* podem ser usados em diagramas de atividade. *WorkDefinition* é um *ProcessDefinitionElement*, uma subdivisão da classe *WorkItem* que descreve o trabalho executado por papéis. Suas subdivisões de classe principais são *ActivityKind*, como também *Phase*, *Iteration*, e *Lifecycle* (no Processo empacotam *Lifecycle*). Distintamente *WorkItem* e *WorkDefinition* podem ser recursivamente estruturados, e têm explicitamente entradas e saídas. Um *WorkDefinitionName* é uma classe auxiliar cujas instâncias designam definições de trabalho ao criar definições de trabalho agregado. A introdução destas classes auxiliares permitem a um *WorkDefinition* se referir com múltiplos nomes, semelhantemente ao *ArtifactName* para os *Artefacts*.
- ***ActivityKind* e *Steps***: *ActivityKind* é a principal subclasse concreta de *WorkDefinition*. Descreve uma parte do trabalho executado por um papel: as tarefas, operações, e ações que são executadas por um papel ou com que o papel pode assistir. Um *ActivityKind* pode ser decomposto em elementos atômicos chamado *Steps*. *Steps* são uma subdivisão de

classe de *WorkItem* e, então, não podem ser decompostos. *Steps* são executados pelo mesmo papel que executa a atividade.

- **RoleKind:** Um *RoleKind* define um papel, possibilita a composição de um papel por uma pessoa, ou um grupo das pessoas, pode ser chamado para atuar em um processo. *RoleKind* define responsabilidades sobre os artefatos específicos, e define os papéis que executam e assistem as atividades específicas.

3.6.3 Terminologia

Há um grande número de modelos de processo e padrões. Cada um usa terminologia ligeiramente diferente, às vezes com significado diferente para a mesma palavra inglesa ou frase. Por exemplo, uma *phase* em Fusion (Coleman, 1994) é chamado de *core workflow* no Processo Unificado da Rational (Rational, 2000) e um *Domain* no Método SI da IBM. O UPM designa como *Discipline*, OPEN (Graham et al., 1997) e o Processo Unificado da Rational (Rational, 2000) usam a palavra *Activity* mas com um significado diferente. O UPM usa *translations* (pseudônimos ou sinônimos). Isto também permite nomear os vários elementos de processo pelo termo apropriado em vários idiomas: japonês, francês, e assim por diante. A OMG (2000) apresenta a comparação entre os termos propostos pela OMG para o UPM.






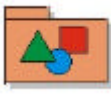
3.6.4 Notação Gráfica

A intenção do UPM é oferecer uma notação gráfica para descrever processos de engenharia de software que são similares aos da UML. Porém, há poucos diagramas da UML que são prontamente possíveis; isso ocorre por causa das dificuldades de se fazer um perfil para UPM com a UML. Assim, a proposta da OMG sobre o UPM não faz um perfil da UML para este. Espera-se que nas próximas submissões seja introduzido esse mapeamento para mostrar a correspondência entre entidades de UPM e UML.

Idealmente, uma notação gráfica para o UPM usaria diagramas de classe para descrever dependências entre elementos de processo, como uma (*work-breakdown*) estrutura de trabalho ou estrutura de produto. Usaria diagrama de atividades para descrever a seqüência de atividades ou diagramas de colaboração para mostrar interação entre vários *roles* (papéis).

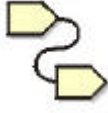


Os estereótipos propostos do UPM para uso em diagramas UML são demonstrados na Tabela 3.10.

TABELA 3.10 - Estereótipos propostos para gráficos.

<i>UPM Element</i>	<i>Icon</i>	<i>Comments</i>
<i>RoleKind</i>		
<i>ActivityKind</i>		
<i>ArtifactKind</i>	   	<i>Simple artifact,</i> <i>Document</i> <i>Model</i> <i>Aggregate of artifacts</i>

(Continua)

TABELA 3.10 – Conclusão.

<i>WorkDefinition</i>		<i>Other than ActivityKind</i>
<i>ProcessComponent</i>		
<i>Process</i>		

3.7 Ambientes de Engenharia de Software - PSEE

Na área de engenharia de software, há cada vez mais interesse em conceber e desenvolver ambientes de engenharia de software que tornem possível e explícita a especificação dos processos de software (Sant'Anna, 2000).

No princípio, ferramentas separadas, como os construtores e compiladores, foram desenvolvidas para automatizar aspectos mecânicos do desenvolvimento de software, como por exemplo, manter o controle dos endereços de memória e associar aos nomes de variáveis. Outras ferramentas como depuradores foram desenvolvidos para apoiar as partes mais intelectuais do processo.

Antes ferramentas apoiavam as atividades simples, como codificação, mas gradualmente foram desenvolvidas ferramentas para também apoiar atividades mais complexas como por exemplo, análise de requisitos e projeto. Sistemas de arquivo permitiram o armazenamento de produtos de trabalho dentro do ambiente computacional. Até agora, foram desenvolvidas ferramentas para atividades individuais do software isoladas das outras atividades.

Os ambientes integrados como: Smalltalk (Ingalls, 1978), Interlisp (Teitelman; Masinter, 1984), e Unix (Ritchie; Thompson, 1974), aumentaram o nível de suporte ao processo permitindo que as várias ferramentas suportassem partes individuais do processo de forma a cooperar e compartilhar os produtos de trabalho. A ferramenta Unix Make (Feldman, 1979) é uma das mais novas ferramentas que permitem ao usuário definir e automatizar partes do processo de desenvolvimento de software. *Shell scripts* servem para o mesmo propósito (Bourne, 1978).

Para integração adequada de ferramentas, alguns ambientes como o PCTE (Boudier et al., 1988) suportam o armazenamento dos objetos, estruturas e tipos, melhor que os arquivos *flat*. Ambientes baseados no Ambiente de Field (Reis, 1990), por exemplo, SoftBench da HP (Cagan, 1990), fornecem mecanismos para integrar ferramentas independentes em um ambiente (distribuído) e para as ferramentas se comunicarem umas com as outras. Esses ambientes mais tarde, foram chamados de **Ambientes de Engenharia de Software**, pois fornecem suporte integrado de múltiplas ferramentas e fornecem suporte para todas as atividades de software.

Ambientes de Engenharia de Software Centrado em Processo, ou *Process centered Software Engineering Environment* (PSEE) (Christie, 1993) (Ben-Shaul; Kaiser, 1995) (Ambriola et al., 1997) (Fuggetta, 2000), foram uma nova classe de ambientes que consideram a engenharia de software como a execução de um processo definido explicitamente. A ênfase em um modelo explícito do processo de software permite estender a visão do desenvolvimento de software para abranger três diferentes tipos de atividades, freqüentemente realizadas por grupos separados dentro da organização de desenvolvimento de software. As três atividades são:

- Atividade de engenharia de processo - define e avalia modelos do processo de software.

- Atividade de gerenciamento de projeto - seleciona um modelo de processo produzido pela engenharia de processo e o adapta para um projeto particular e então monitora a execução deste processo.
- Atividade de engenharia de software - executa um processo específico através do gerenciamento do projeto. Na Figura 3.4, são mostradas essas três atividades e tipos de suporte que um PSEE pode fornecer a cada uma delas. O modelo de processo é o conector dessas três atividades.

Para Kadia (1992), ambiente de engenharia de software pode ser definido como um conjunto de capacidades integradas para suportar os engenheiros de software e os gerentes em seu trabalho.

Na especificação SPICE (1995), esses ambientes são definidos como um conjunto integrado de ferramentas de desenvolvimento de software, para serem utilizadas pelos projetos na organização, suportando o processo de engenharia de software de forma consistente.

Sant'Anna (1993) define como um conjunto de ferramentas integradas a um banco de conhecimento de projetos para apoiar o desenvolvimento, a gerência e a experimentação de software.

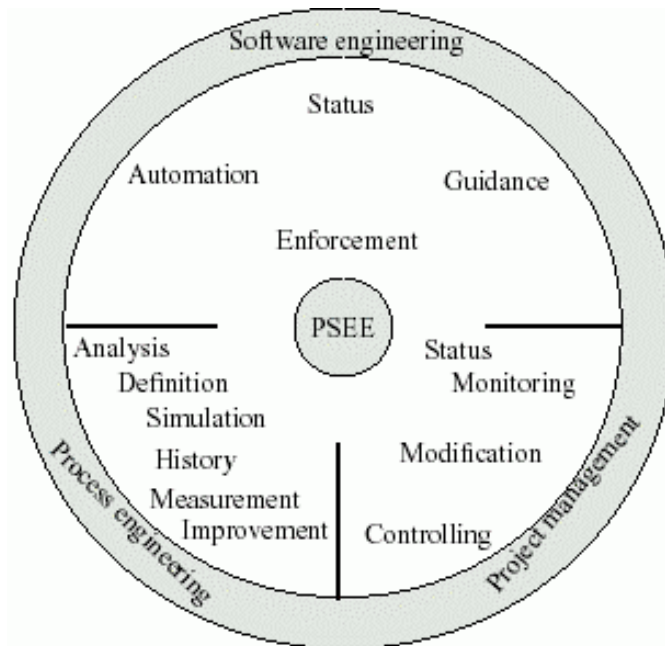


FIGURA 3.4 - O PSEE centraliza e integra o apoio para engenharia do processo, gerência de projetos e engenharia de software.

FONTE: Garg; Jazayeri (1996).

Já em (Sant'Anna, 2000), o autor redefine ambiente de engenharia de software como um conjunto de recursos flexíveis e abrangentes, dando suporte eficiente e de forma integrada ao processo de engenharia de software definido para uma organização.

A tecnologia de processos de software propõe o desenvolvimento e adoção de PSEEs para automatizar a gerência dos processos (Gimenes, 1994). Esta tecnologia traz benefícios, como por exemplo: melhor comunicação entre as pessoas envolvidas e consistência do que está sendo feito; realização de algumas ações automáticas, permitindo que os seus usuários não realizem tarefas repetitivas; fornecimento de informações sobre o andamento do processo quando necessário; possibilidade de reutilização de processo de software e a coleta automática de métricas importantes para o controle e aperfeiçoamento de processos (Reis, 2001).

Ellmer (1995), numa visão futurista, vislumbra que os PSEEs serão a próxima geração de ferramentas *Computer Assisted Software Engineering (CASE)* (Cruz, 2000).

Os PSEEs estão se tornando uma realidade. Um crescente número de sistemas está revelando e aplicando processos para produção de software. Estes produtos e protótipos existentes são baseados em uma variedade de tecnologias e abordagens, como as linguagens e base de dados orientados a objetos, notações orientadas a estados, linguagens baseadas em regras ou mesmo linguagens lógicas (Ambriola et al., 1997).

Ambientes de engenharia centrados em processos se espalham nos ambientes de desenvolvimento de software de dois modos importantes:

- 1) Eles apóiam a definição e uso de um processo explícito de desenvolvimento de software. A forma, capacidade e tipo de apoio, variam entre os sistemas. No mínimo o ambiente deve apoiar a definição e o monitoramento do processo para detectar possíveis desvios e falhas no processo.
- 2) Eles vêem o processo de software mais que uma descrição estática do ciclo de vida do software; eles vêem o processo como uma entidade dinâmica com seu próprio ciclo de vida. A primeira diferença entre um ambiente centrado em processo e os ambientes mais convencionais é o quanto eles apóiam as várias fases do ciclo de vida do processo de software. Realmente, PSEEs diferem de um para o outro, no nível de apoio que cada um provê nas diferentes fases ciclo de vida do processo.

Uma representação explícita de um processo é uma característica de definição de PSEEs. Um modelo de processo pode ser usado pelo PSEE para ajudar no projeto, análise, automação, monitoramento, e compartilhamento de processos de software. Vários PSEEs estão disponíveis agora tanto em comunidades

acadêmicas quanto em comunidades industriais. Contudo, nem todos os PSEEs fornecem o mesmo conjunto de recursos.

As principais características dos PSEEs, são:

- **Definição de Processo:** Um engenheiro de processo usa um PSEE para definir um processo a ser seguido por um ou mais projetos. Os PSEEs atuais oferecem várias formas e critérios para especificar um processo. Algumas notações gráficas atuais dos PSEEs como ETVX, IDEF, diagrama de flechas, diagrama de estados, diagramas Entidade Relacionamento - ER, redes de Petri, ou diagramas de fluxo de dados. Outros PSEEs permitem ao engenheiro de processo especificar o processo usando as tabelas ou formulários de texto. Outros PSEEs usam linguagem de programação para definir um processo. No paradigma da linguagem de programação, vários critérios foram testados, por exemplo, linguagens de *scripts* (extensões para linguagem de programação *Shell*), linguagens baseadas em regra (extensões para OPS-5 e Prolog), linguagens de base de dados ativadas por *trigger* e linguagens de programação imperativa (extensões para ADA). Finalmente, vários PSEEs fornecem uma combinação de tais características algumas vezes até mesmo permitindo tradução de um para outro.

- **Análise de Processos:** Um modelo de processo dentro de um PSEE pode ser analisado por consistência, integridade e corretude. Por exemplo, um modelo de processo pode ser analisado para determinar o número máximo de atividades paralelas possíveis em um processo (para propósitos de programação), ou para determinar se existem quaisquer atividades redundantes em um processo (as saídas que não estão sendo usadas). O processo pode ser analisado por conformidade com as diretrizes de qualidade que especificam um conjunto de atividades a

serem realizadas em uma ordem particular, ou solicitar a existência de certas atividades de inspeção e revisão.

- **Apresentação do Processo:** Os processos de software descrevem fluxos de atividade que têm uma representação gráfica natural. Os produtos do processo também podem ser visualizados usando os diagramas estruturados. Um PSEE tipicamente suporta o *display* gráfico de informação de processo e produto. Tais *displays* gráficos são úteis tanto no papel quanto nos terminais eletrônicos. Normalmente, o *display* gráfico reflete o formalismo da definição do processo básico, por exemplo, uma definição baseada em redes de Petri poderá ser visualizada como uma Rede de Petri. Recentemente, vários PSEEs estão defendendo a importância de fornecer visualizações múltiplas do processo, por exemplo, um gerenciamento comparado a uma visão de engenharia.

- **Simulação de Processo:** A execução de um processo de software normalmente exige várias pessoas, é realizada em um longo período e exigem um gasto monetário substancial. Para avaliar a conveniência de um processo antes de comprometer todos os recursos para ele, alguns PSEEs suportam simulações de processo.

- **Automação do Processo:** Um processo de software freqüentemente inclui atividades que não exigem intervenção humana em sua execução, pois eles podem ser na maioria das vezes automatizados. Um exemplo típico de uma atividade automatizável é executar testes de regressão em um módulo alterado recentemente. Notificar o pessoal afetado sobre uma alteração em alguma informação do produto é um outro exemplo. Uma vez que um processo foi definido em um PSEE, tais atividades podem ser identificadas (pelo engenheiro do processo) e automatizadas subsequentemente pelo PSEE. O escopo e a extensão de tal automação depende da informação do processo e produto no PSEE, e o formalismo

da definição do processo usado pelo PSEE. Por exemplo, se um PSEE não modela o pessoal envolvido em um processo, ele não poderá suportar efetivamente a notificação de alteração.

- **Monitoramento de Processo:** Um importante aspecto de um PSEE é monitorar a execução de um processo e registrar o histórico das atividades realizadas. Um histórico de processo pode ser usado para desenvolvimento e melhorias futuras do processo. Alguns PSEEs fornecem monitoramento em tempo real de um processo com *displays* gráficos, enquanto outros produzem dados do processo que podem ser analisados e visualizados com ferramentas de análise de dados. A privacidade de informação é um problema crítico aqui, uma vez que os dados monitorados podem ser facilmente mal utilizados. Por outro lado, é geralmente aceito que as métricas de processo devem ser monitoradas para melhoria efetiva das práticas de engenharia de software.

- **Suporte a Alteração de Processo:** Os processos de software se alteram por várias razões: a tecnologia de programação pode alterar, por exemplo, programação orientada a objeto a partir da programação estruturada; novos paradigmas do processo podem ser descobertos, tais como: modelo de ciclo de vida espiral; ou, ferramentas de suporte podem ser desenvolvidas e alteradas, por exemplo, programação orientada a *windows* a partir de terminais baseados em caractere. Uma organização usando um PSEE deve ser capaz de alterar suas definições de processo dentro do PSEE, freqüentemente sem interromper o funcionamento da organização.

- **Interoperabilidade:** Os PSEEs têm que operar em ambientes de computação onde eles possam ou não possam ter controle sobre todas as ferramentas de software que estejam disponíveis. Alguns PSEEs fornecem facilidades de importação e exportação para intercambiar

dados do produto entre as ferramentas existentes e o PSEE. Outros PSEEs estão de acordo com os esquemas de notificação baseados em evento padrão para integrar com as ferramentas existentes.

- **Suporte Multi-Usuário:** Um PSEE tipicamente tem: múltiplos usuários, no mínimo, pessoas com funções de engenheiro de processo, gerente, e engenheiro de software usam o mesmo PSEE. Múltiplos engenheiros de software freqüentemente trabalham no mesmo projeto, conseqüentemente, acessando o mesmo PSEE. Um PSEE fornece suporte para uma equipe de pessoas trabalhando juntas em um processo. Um PSEE assegura que o acesso concorrente para dados de produto e processo seja sincronizado.

- **Guia de Processo:** Os engenheiros de software usam um PSEE para realizar vários passos do processo. Um PSEE freqüentemente orienta um engenheiro de software sobre os possíveis próximos passos, baseado no processo modelado e seu estado atual. Alguns PSEEs fornecem tal orientação apresentando ao usuário uma lista de tarefa dos itens de ação; outros apresentam ao usuário os objetos do produto atual sendo criados e as operações disponíveis neles; e ainda outros apresentam ao usuário um ambiente de programação tradicional com quase nenhuma orientação.

- **Interface de Usuário para Tarefa Específica:** Baseado em um processo modelado, um PSEE pode estender a interface usuário para reduzir o aumento da informação apresentada para o usuário. Ambientes de programação diferentes como Unix, que não têm qualquer entendimento do usuário para ser executado, um PSEE não tem que apresentar todas as ferramentas disponíveis de modo uniforme. Dependendo da tarefa à mão, um PSEE pode limitar as ferramentas que são aplicáveis para aquela tarefa. Semelhantemente, um PSEE poderá limitar os produtos visíveis ao seu usuário.

A explicação anterior apresenta uma união das características comumente encontradas em PSEEs de pesquisa e comercial existentes. Os PSEEs atuais suportam essas características e recursos em vários níveis.

Trabalhos mais recentes em PSEEs, como o *Formalized System Development* (FSD) desenvolvido na *University of Southern California's Information Sciences Institute* (Balzer et al., 1983) foi um ambiente pioneiro com modelos explícitos de atividades de engenharia de software. O FSD fornece suporte à automação e ao monitoramento através do ambiente de engenharia de software.

Uma motivação semelhante guiou uma equipe de pesquisadores no Kestrel Institute para definir um ambiente de programação automática (Smith et al., 1985). Essas abordagens foram posteriormente combinadas em um projeto chamado *Knowledge-based Software Assistant* (KBSA), com a meta de codificar formalmente o processo de software em um ambiente de engenharia de software para fornecer suporte automatizado para as diferentes atividades (Green et al., 1983). No mesmo período, dois outros projetos desenvolveram idéias semelhantes: além de “*intelligence*” no ambiente de programação Gandalf da Universidade Carnegie-Mellon (Kaiser; Feiler, 1987), e o desenvolvimento de um intérprete de programa de processo dentro do projeto Arcadia (Taylor et al., 1988). Baseado neste trabalho do começo dos anos 80, vários projetos foram iniciados no final dos anos 80 e início dos anos 90.

Algumas tendências gerais são evidentes nestes desenvolvimentos. O trabalho mais recente sobre PSEEs foi motivado para tornar os ambientes de engenharia de software mais “inteligentes”. Isto levou ao conceito de desenvolver modelos de processos formais e fazer códigos dentro do ambiente.

Subseqüentemente, os pesquisadores perceberam que o desenvolvimento do modelo do processo é uma atividade não-trivial. Isto levou vários sistemas a fornecerem suporte para desenvolvimento e análise de modelo de processo. Isto é uma tendência para os atuais PSEEs. Finalmente, pode-se indicar a

tendência de que os PSEEs irão direcionar sua ênfase para suportar as contínuas melhorias de processo (CPI) para as organizações. Os atuais PSEEs não tratam este problema adequadamente, mas alguns trabalhos estão surgindo neste sentido.

No país, algumas propostas foram desenvolvidas, e dentre elas se destacam o ambiente ExpSEE (Gimenes, 2000), a estação TABA (Oliveira, 1999) (Travassos, 1994), o APSEE-Prosoft (Reis, 1998) (Moraes, 1997) e o e-WebProject® (Sant'Anna, 2002).

O ExpSEE permite a definição precisa e instanciação de atividades do processo de software em relação a artefatos, atores e ferramentas utilizadas por atividades.

O projeto da Estação TABA, iniciou-se no início dos anos 90 e vem sendo desenvolvida pela Universidade Federal do Rio de Janeiro - UFRJ. A Estação TABA é um meta-ambiente capaz de gerar, através de instanciação, ambientes de desenvolvimento de software adequados às particularidades de processos de desenvolvimento de projetos específicos. Desta forma, a Estação TABA possui facilidades para a definição de processos, métodos e ferramentas CASE a serem utilizadas no processo de desenvolvimento. Esse ambiente foi inicialmente desenvolvido em Eiffel na plataforma UNIX e agora está desenvolvido em C++.

O APSEE-Prosoft está sendo desenvolvido no Instituto de Informática da Universidade Federal do Rio Grande do Sul - UFRGS, num programa de cooperação internacional com o *Institut fur Informatik - Universitat Stuttgart* (Alemanha). O objetivo principal deste ambiente é a gerência do processo de desenvolvimento de software. Sua última versão está sendo desenvolvida em Java.

O ambiente e-WebProject® será melhor detalhado na sessão seguinte, pois este PSEE servirá de base para a definição e implementação da arquitetura que será proposta nas próximas sessões.

Souza et al. (2001), classifica e compara uma série de outros PSEEs, apresentando suas características e principais funcionalidades, em Derniame et al. (1999) são reunidos diversos aspectos mais recentes da tecnologia de processos de software.

3.7.1 e-WebProject®

O PSEE eWebProject® (Sant'Anna et al., 2002) é um produto comercial originado do trabalho de pesquisa "Um Ambiente Integrado para o Apoio ao Desenvolvimento e Gestão de Projetos de Software para Sistemas de Controle de Satélite - AMBGES" (Sant'Anna, 2000) desenvolvido no Instituto Nacional de Pesquisas Espaciais - INPE.

O principal objetivo do ambiente é fornecer um conjunto de ferramentas integradas para trabalho cooperativo, suporte à execução de processos de engenharia de software, gestão do conhecimento organizacional e apoio à infra-estrutura organizacional.

Este PSEE foi desenvolvido em Java e disponibilizado através da WEB/Internet, o que permite uma maior e mais fácil disseminação das informações.

As principais características deste ambiente são: interoperabilidade, flexibilidade, extensibilidade, melhoria dos passos, rapidez, eficiência, facilidade de uso, suporte à grande quantidade de usuários de classes diferentes, apoio ao trabalho cooperativo, apoio à visibilidade dos produtos e dos processos, apoio ao gerenciamento de forma efetiva, suporte a projetos simultâneos, flexibilidade na organização das equipes.

A arquitetura do ambiente, para atender às necessidades antes apresentadas, deve possuir as seguintes classes de componentes:

- Elementos responsáveis pelo armazenamento e recuperação das informações de projeto, promovendo a integração da informação.
- Elementos responsáveis pela prestação de serviços.
- Elementos responsáveis pela interação do usuário com o ambiente.

Estes elementos são organizados de forma a refletir uma organização em camadas hierárquicas, como demonstra a Figura 3.5.

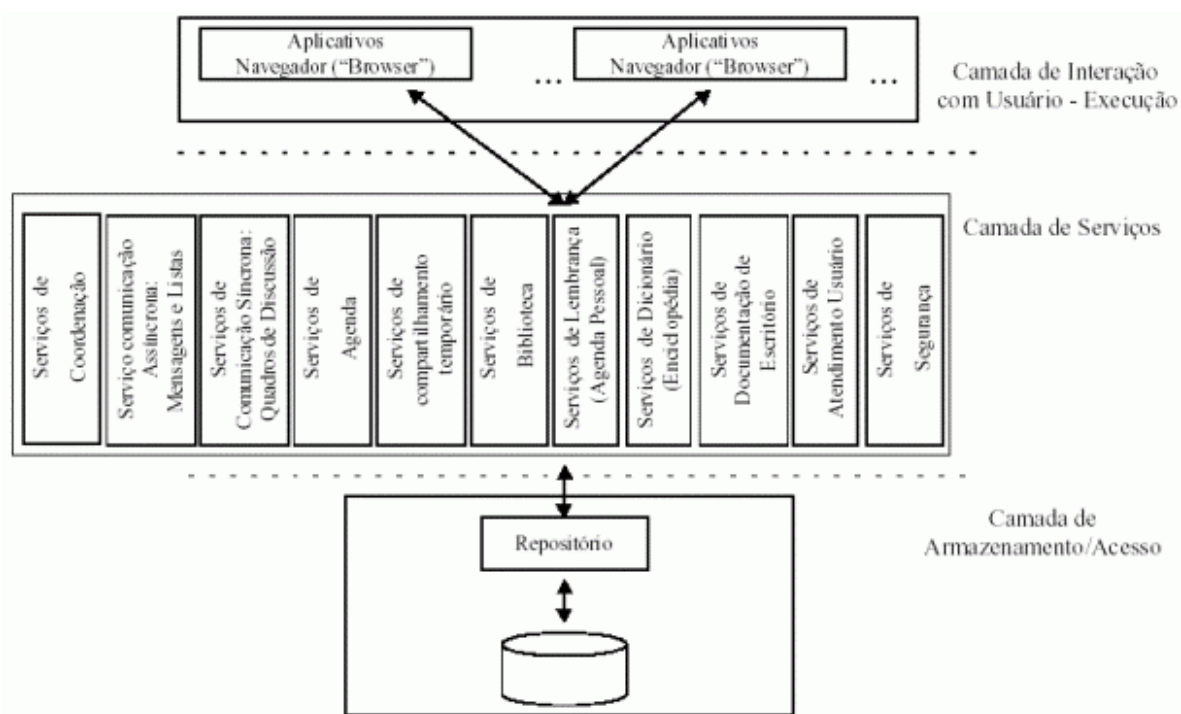


FIGURA 3.5 - Componentes e camada conceitual do ambiente.

FONTE: Sant'Anna (2000).

- **Camada de armazenamento e recuperação:** responsável pelo armazenamento e recuperação das informações de projeto.

- **Camada de serviços:** vários elementos são necessários para que os indivíduos trabalhem de forma cooperativa. Para atender ao desenvolvimento de software, vários serviços podem ser prestados pelo ambiente de modo a atender os requisitos especificados. Estes serviços utilizam os elementos da camada de armazenamento e recuperação para obtenção das informações.
- **Camada de interação com o usuário:** o usuário deverá obter acesso ao ambiente a partir de qualquer ponto de uma rede de computadores. A utilização de navegadores (*browsers*) permite tal facilidade, haja vista a capacidade de operação em modo gráfico e ambientes de janelas, possibilitando interação fácil e eficaz; além disso, existem navegadores para todas as plataformas existentes atualmente.

A arquitetura física é desenvolvida baseado no modelo "n-camadas" para aplicações Web (Conallen, 2000). A Figura 3.6 apresenta os componentes da arquitetura.

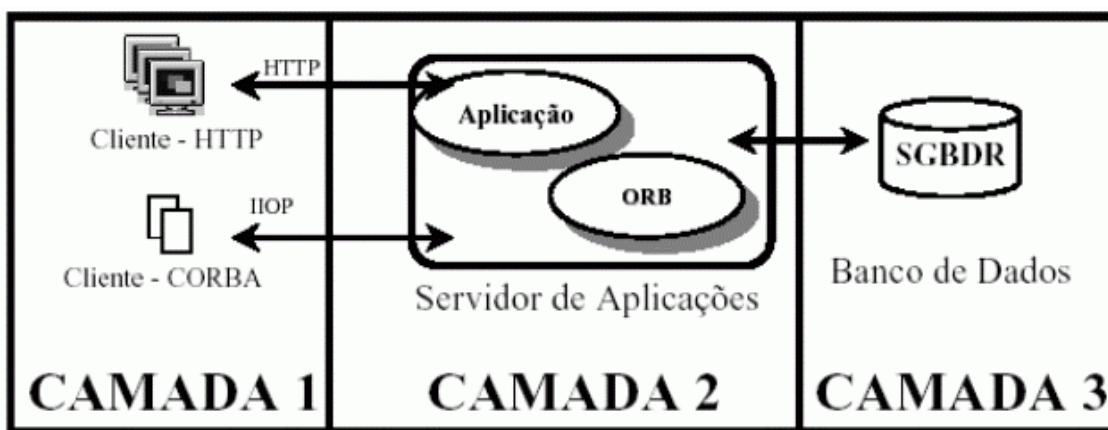


FIGURA 3.6 - Arquitetura física do ambiente.

FONTE: Sant'Anna (2000).

CAPÍTULO 4

PROCESSOS ESSENCIAIS DE GERENCIAMENTO DE PROJETOS

O objetivo final deste trabalho é propor uma infra-estrutura inicial, porém robusta, para apoio aos processos de gerenciamento de projetos. Inicial, porque muitos são os processos que podem ser automatizados, mas é inviável que se automatize, ou mesmo, se crie qualquer tipo de suporte sistematizado para os outros processos, sem antes construir um suporte aos processos que neste trabalho serão chamados de **essenciais**. Robusta, pois posteriormente, quando forem automatizados os outros processos de interesse, os mesmos poderão ser integrados à mesma infra-estrutura, que por sua vez será capaz de suportar extensões para suporte aos outros processos.

Para este trabalho, os processos essenciais são os processos referentes às Gerências de Custo, Tempo e Recursos Humanos.

Será tomado como base o modelo de processos proposto pelo PMI (2000), pois é amplamente aceito e aplicado pela Comunidade de Engenharia de Software e pelas outras comunidades que fazem uso do gerenciamento de projetos. Além disso, a Tabela 4.1 mostra a compatibilidade do PMBOK2000 com as principais abordagens de processos de engenharia de software, como o CMMI e o SPICE.

TABELA 4.1 - PMBOK2000 x CMMI x SPICE.

	PMBOK2000								
	Integração	Escopo	Tempo	Custo	Qualidade	RH	Comunicação	Riscos	Aquisições
CMMI									
PLANEJAMENTO DO PROJETO									
Estabelecer Estimativas		x	x	x		x			
Desenvolver um Plano de Projeto	x	x	x	x		x	x	x	
Obter Compromisso para o Plano		x				x			
CONTROLE E MONITORAMENTO DO PROJETO									
Monitorar o Projeto Em Comparação com o Plano		x	x	x			x	x	
Gerenciar a Ação Corretiva para Encerrar					x			x	x
GERENCIAMENTO DE CONTRATO DO FORNECEDOR									
Estabelecer os Contratos de Fornecedor		x							x
Satisfazer os Contratos do Fornecedor									x
GERENCIAMENTO DO PROJETO INTEGRADO									
Utilizar o Processo Definido do Projeto		x							

(Continua)

Nas sessões seguintes serão apresentados os processos de Gerenciamento do Tempo, Gerenciamento dos Custos e Gerenciamento dos Recursos Humanos modelados em detalhes usando a UPM (OMG, 2000) com as respectivas descrições e especificações, e por fim os grupos de processo para gerência de projetos, proposto pelo PMI, também usando a UPM. Os processos citados anteriormente são mostrados na Figura 4.1.

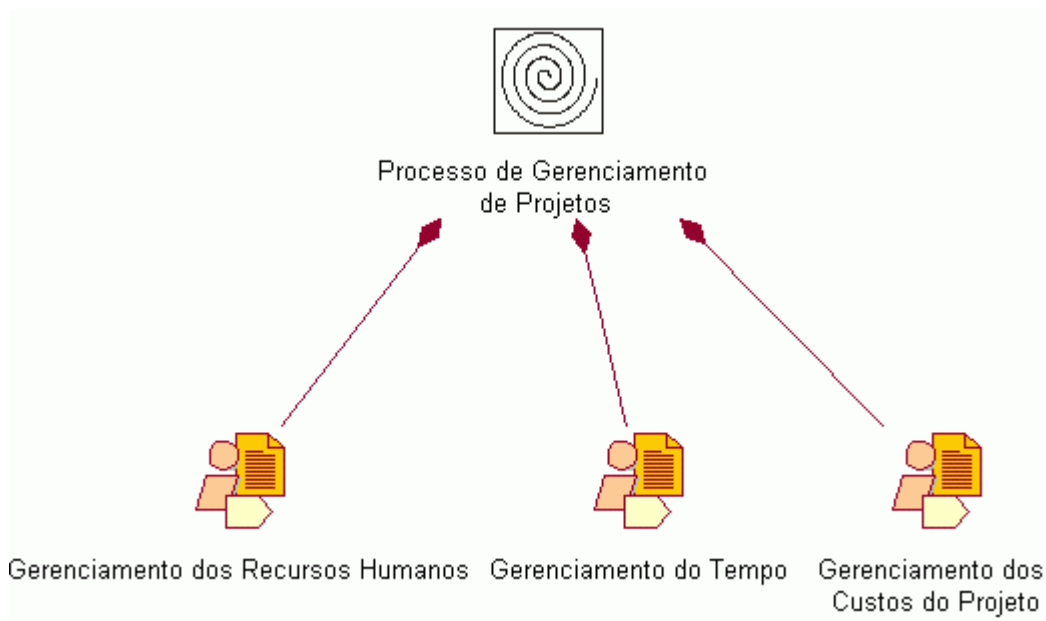


FIGURA 4.1 - Processos essenciais de gerenciamento de projetos.

4.1 Gerenciamento do Prazo do Projeto

O Gerenciamento do Prazo do Projeto inclui os processos necessários para assegurar que o projeto será implementado no prazo previsto. A Figura 4.2 fornece uma visão geral dos processos para criar o cronograma do projeto:

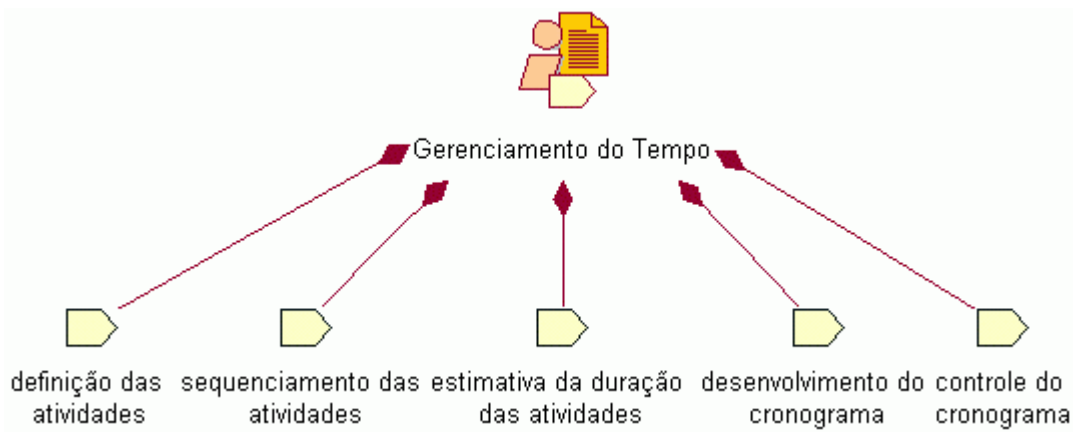


FIGURA 4.2 - Processo de gerenciamento do tempo.

- **Definição das Atividades** – identificar as atividades específicas que devem ser realizadas para produzir os diversos subprodutos do projeto.
- **Seqüenciamento das Atividades** – identificar e documentar as relações de dependência entre as atividades.
- **Estimativa da Duração das Atividades** - estimar a quantidade de períodos de trabalho que serão necessários para a implementação de cada atividade.
- **Desenvolvimento do Cronograma** - analisar a seqüência das atividades, sua duração, e os requisitos de recursos para criar o cronograma do projeto.
- **Controle do Cronograma** - controlar as mudanças no cronograma do projeto.

Estes processos interagem uns com os outros e também com os processos das demais áreas de conhecimento. Cada processo pode envolver esforço de um ou mais indivíduos ou grupos de indivíduos dependendo das necessidades do projeto. Cada processo geralmente ocorre pelo menos uma vez em cada fase do projeto.

Em alguns projetos, especialmente os menores, o seqüenciamento das atividades, as estimativas das durações e o desenvolvimento do cronograma estão tão vinculados que podem ser tratados como um único processo (podem ser realizados por um único indivíduo, durante um curto intervalo de tempo). Esses processos são aqui apresentados como processos distintos porque as ferramentas e técnicas são diferentes para cada um.

O ator que participa na execução dos processos de gerenciamento do tempo é o Gerente do Prazo do Projeto, o qual será representado nos próximos diagramas pelo *RoleKind* correspondente. Neste trabalho foi este o ator definido, porém ele pode variar em outros casos ou cenários particulares, porém, ele deve ter o perfil de conhecer o escopo do trabalho a ser executado para que o mesmo possa definir, sequenciar e estimar os tempos das atividades necessárias para a execução do projeto.

4.1.1 Definição das Atividades

A definição das atividades envolve identificar e documentar as atividades específicas que devem ser realizadas com a finalidade de produzir os diversos níveis de subprodutos identificados na EAP. Está implícita neste processo a necessidade de definir aquelas atividades voltadas para o alcance dos objetivos do projeto. A Figura 4.3 mostra o processo modelado usando UPM com respectivas entradas, saídas e ferramentas.

4.1.1.1 Entradas para a Definição das Atividades

.1 Estrutura analítica do projeto. A EAP é a principal entrada para a definição das atividades. Uma estrutura analítica do projeto (EAP) é um agrupamento de componentes de projeto (orientado para a elaboração de subprodutos *deliverable-oriented*) que organiza e define o escopo total do projeto: o trabalho que não está na EAP está fora do escopo do projeto.

A cada item da EAP é, geralmente, designado um identificador único; estes identificadores podem fornecer uma estrutura para a totalização hierárquica de custos e recursos. Os itens nos níveis mais baixos da EAP são, freqüentemente, referenciados como pacotes de trabalho (*work packages*) especialmente nas organizações que seguem as práticas de gerenciamento pelo *earned value*. Estes pacotes de trabalho podem ainda ser decompostos em uma EAP de subprojeto. Geralmente, este tipo de abordagem é usado quando o gerente do projeto está atribuindo uma parte do trabalho para outra organização, e esta outra organização deve planejar e gerenciar o escopo num nível mais detalhado do que necessita o gerente do projeto na estrutura principal. Estes pacotes de trabalho podem ser mais tarde desdobrados no plano do projeto e cronograma.

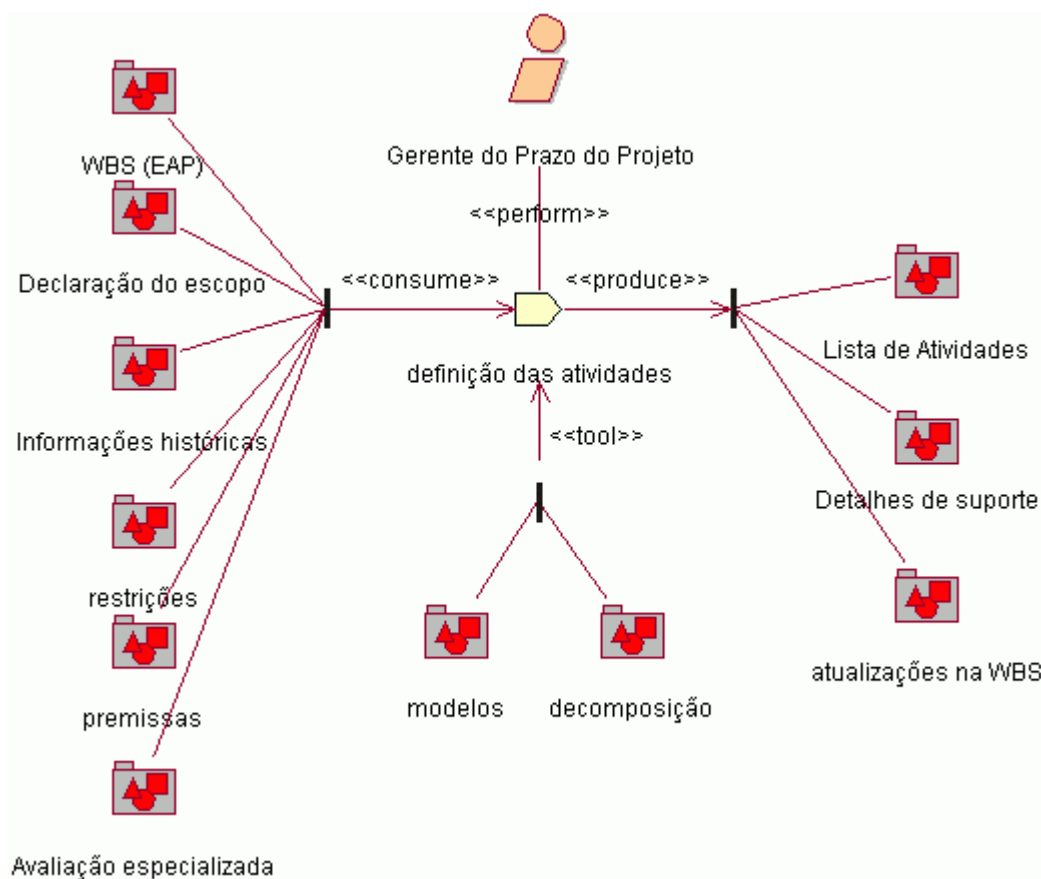


FIGURA 4.3 - Modelo do processo definição das atividades.

A EAP não deve ser confundida com outros tipos de estruturas de decomposição usadas para apresentar informações do projeto. Outras estruturas comumente usadas em algumas áreas de aplicação são:

- Estrutura analítica do projeto - contratual (*Contractual EAP - CEAP*), que é usada para definir o nível de informação que o vendedor fornecerá para o comprador. A CEAP geralmente possui menos detalhes que a EAP usada pelo vendedor para gerenciar o seu próprio trabalho.
- Estrutura de decomposição organizacional (*Organizational Breakdown Structure - OBS*), que é usada para relacionar que elementos de trabalho foram designados para quais unidades da organização.
- Estrutura de decomposição de recurso (*Resource Breakdown Structure – RBS*), que é uma variação da OBS, e é usada, tipicamente, quando os elementos de trabalho são designados para indivíduos.
- Lista de Materiais (*Bill of Materials – BOM*), que apresenta uma visão hierárquica de montagens físicas, submontagens e componentes necessários para fabricar um produto industrializado.
- Estrutura de decomposição do projeto (*Project Breakdown Structure – PBS*), que é, fundamentalmente, o mesmo que a própria EAP. O termo PBS é mais utilizado nas áreas de aplicação onde o termo EAP é, incorretamente, usado para referenciar uma Lista de Materiais (BOM).

.2 Declaração do escopo. A justificativa e os objetivos do projeto contidos na declaração do escopo devem ser considerados, explicitamente, durante a definição das atividades.

.3 Informações históricas. As informações históricas (que atividades foram requeridas em projetos anteriores semelhantes) devem ser consideradas na definição das atividades do projeto.

.4 Restrições. As restrições são fatores que limitarão as opções da equipe de gerência do projeto; um exemplo seria a atribuição de durações máximas para as atividades.

.5 Premissas. Suposições são fatores que, para os propósitos do planejamento, são consideradas verdadeiros, reais, ou certos. Premissas afetam todos os aspectos do planejamento do projeto e são parte da elaboração progressiva do projeto. Equipes de projeto freqüentemente identificam, documentam e validam as premissas como parte de seus processos de planejamento.

Por exemplo, se a data na qual uma pessoa chave estará disponível para o projeto é incerta, a equipe pode assumir uma data de início específica. As premissas geralmente envolvem certo grau de risco.

.6 Avaliação especializada. Uma avaliação especializada é, freqüentemente, requerida para avaliar as entradas deste processo. Tal habilidade pode ser provida por um grupo ou indivíduo com conhecimento ou treinamento especializado, e está disponível em várias fontes, por exemplo:

- Outras unidades dentro da organização.
- Consultores.
- Partes envolvidas, incluindo clientes.
- Associações profissionais e técnicas.
- Grupos industriais.

4.1.1.2 Ferramentas e Técnicas para a Definição das Atividades

.1 Decomposição. Dentro do contexto da definição das atividades, a decomposição significa subdividir os pacotes de trabalho do projeto em

componentes menores e mais manejáveis com a finalidade de fornecer melhor controle do gerenciamento. Em algumas áreas de aplicação, a EAP e a lista de atividades são desenvolvidas paralelamente.

.2 Modelos (Templates). Uma lista de atividades, ou uma parte de uma lista de atividades de projetos anteriores, é freqüentemente utilizada como modelo ou referência para um novo projeto. As atividades nos modelos de referência podem conter também uma lista dos tipos de recursos e suas necessidades de esforço em horas, identificação dos riscos, resultados esperados, e outras informações descritivas.

4.1.1.3 Saídas da Definição das Atividades

.1 Lista de atividades. A lista de atividades deve incluir todas as atividades que serão realizadas no projeto. Deve ser organizada como uma extensão da EAP para assegurar que esta está completa e que não inclui qualquer atividade que não seja requerida como parte do escopo do projeto. Assim como a EAP, a lista de atividades deve incluir descrições de cada atividade para garantir que os membros da equipe do projeto entenderão como o trabalho deverá ser feito.

.2 Detalhes de suporte. Os detalhes de suporte referentes à lista de atividades devem ser documentados e organizados de forma a facilitar seu uso por outros processos da gerência do projeto. Eles devem sempre incluir a documentação de todas as premissas e restrições identificadas. A quantidade de detalhes adicionais varia de acordo com a área de aplicação.

.3 Atualizações na EAP. Ao utilizar a EAP para identificar quais atividades são necessárias, a equipe do projeto pode perceber a falta de algum subproduto ou ainda determinar se a descrição dos subprodutos precisa ser esclarecida ou corrigida.

Qualquer uma destas atualizações deve ser refletida na EAP e na respectiva documentação, como por exemplo a estimativa dos custos. Estas atualizações

são muitas vezes chamadas de *refinamentos* e ocorrem mais freqüentemente quando o projeto envolve uma tecnologia nova ou ainda não testada.

4.1.2 Seqüenciamento das Atividades

O seqüenciamento das atividades envolve identificar e documentar as relações de dependência entre as atividades. As atividades devem ser seqüenciadas corretamente possibilitando mais tarde o desenvolvimento de um cronograma realista e viável. O seqüenciamento pode ser feito com o auxílio de um computador (por exemplo, utilizando softwares de gerência de projeto) ou com técnicas manuais. As técnicas manuais são, geralmente, mais efetivas em projetos menores e em fases iniciais de projetos maiores quando existem poucos detalhes disponíveis. As técnicas manuais e automatizadas podem, também, ser utilizadas em conjunto. A Figura 4.4 mostra o modelo do processo de seqüenciamento de atividades.

4.1.2.1 Entradas para o Seqüenciamento das Atividades

.1 Lista das atividades. A lista de atividades deve incluir todas as atividades que serão realizadas no projeto. Deve ser organizada como uma extensão da EAP para assegurar que esta está completa e que não inclui qualquer atividade que não seja requerida como parte do escopo do projeto. Assim como a EAP, a lista de atividades deve incluir descrições de cada atividade para garantir que os membros da equipe do projeto entenderão como o trabalho deverá ser feito.

.2 Descrição do produto. As características do produto freqüentemente afetam o seqüenciamento das atividades (por exemplo, o layout físico de uma planta a ser construída, as interfaces de subsistemas num projeto de software). Embora esses efeitos sejam freqüentemente visíveis na lista de atividades, a descrição do produto deve ser geralmente revisada para assegurar a precisão.

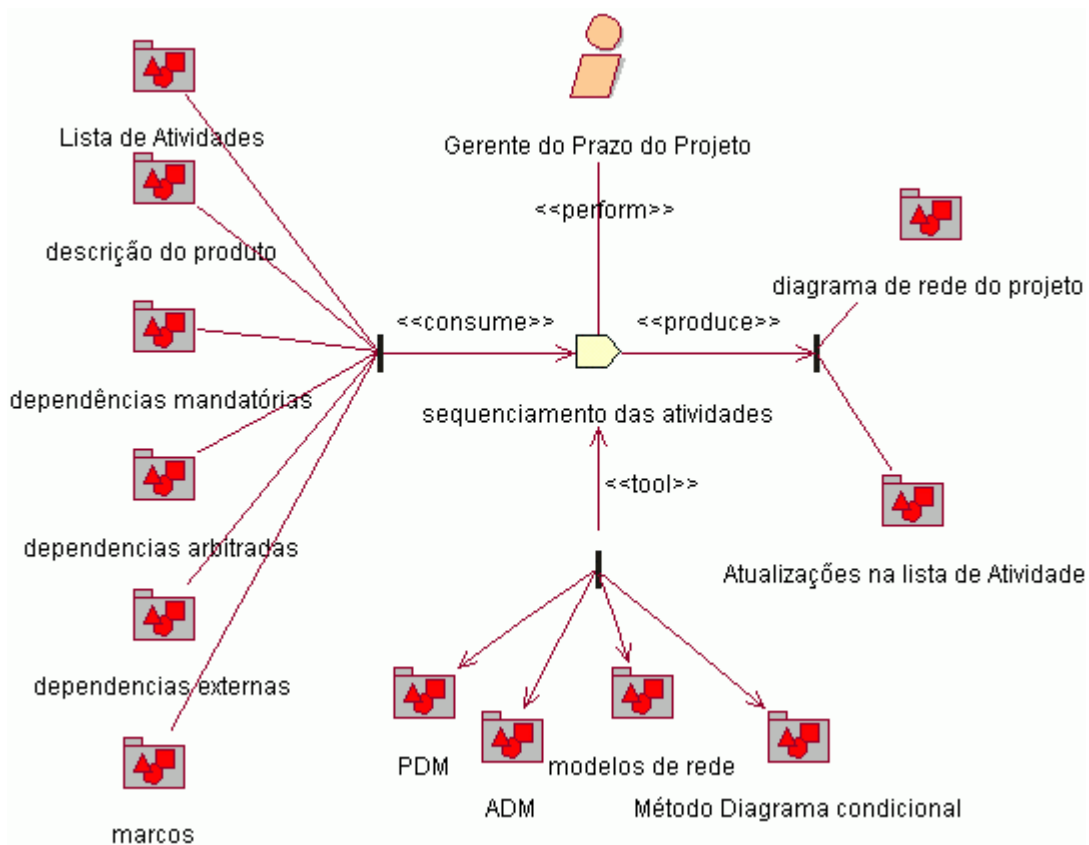


FIGURA 4.4 - Modelo do processo de seqüenciamento de atividades.

.3 Dependências mandatórias. Dependências mandatórias são aquelas inerentes à natureza do trabalho que está sendo feito. Frequentemente, envolvem limitações físicas (por exemplo, em uma construção é impossível levantar a estrutura antes que a fundação tenha sido feita; num projeto eletrônico, o protótipo deve ser construído antes de ser testado). As dependências mandatórias são também chamadas de *lógica rígida (hard logic)*.

.4 Dependências arbitradas. Dependências arbitradas são aquelas definidas pela equipe de gerência do projeto. Devem ser usadas com cuidado (e bem documentadas) já que podem limitar, posteriormente, as opções do cronograma. As dependências arbitradas são usualmente definidas com base no conhecimento de:

- “Melhores Práticas” dentro de uma área de aplicação particular.

- Algum aspecto particular do projeto onde uma seqüência específica é desejada
- Embora existam outras seqüências aceitáveis.

As dependências arbitradas podem, também, ser chamadas de lógica preferida, lógica preferencial ou lógica fina.

.5 Dependências externas. As dependências externas são aquelas que envolvem relacionamento entre atividades do projeto e atividades fora do projeto. Por exemplo, a atividade de teste em um projeto de software pode ser dependente da entrega de um hardware de fornecedor externo; em um projeto de construção, pode ser necessária uma análise de impacto ambiental antes que se possa iniciar a preparação do local.

.6 Marcos. Os eventos “marcos” devem fazer parte do seqüenciamento das atividades para assegurar que sejam atendidos os requisitos necessários para o atingimento dos marcos.

4.1.2.2 Ferramentas e Técnicas para o Seqüenciamento das Atividades

.1 Método do diagrama de precedência (PDM). Este é um método de construção de diagrama de rede que utiliza retângulos para representar as atividades e os conecta por setas que representam as dependências. Esta técnica também é chamada de *atividade no nó* (AON - *Activity-on-node*) e é o método utilizado pela maioria dos pacotes de programas para gerência de projeto. O PDM pode ser feito manualmente ou no computador.

Ele inclui quatro tipos de relacionamento de dependência ou precedência:

- Término/Início (*finish-to-start*) – O início do trabalho da sucessora depende do término da predecessora.

- Término/Término (*finish-to-finish*) – O término do trabalho da sucessora depende do término da predecessora.
- Início/Início (*start-to-start*) – O início do trabalho da sucessora depende do início da predecessora.
- Início/Término (*start-to-finish*) – O término do trabalho da sucessora depende do início da predecessora.

O PDM término/início é o tipo de relacionamento lógico mais comumente usado. Os relacionamentos início/término são raramente usados e assim mesmo apenas por engenheiros profissionais de programação. Usar início/início, término/término ou início/término em softwares de gerência de projetos pode produzir resultados inesperados, uma vez que estes tipos de relacionamentos não foram ainda implementados consistentemente.

.2 Método do diagrama de flecha (ADM - Arrow Diagramming Method). Este é um método de construção de diagrama de rede que utiliza setas para representar as atividades e as conecta por meio de nós que representam as dependências.

Esta técnica é também chamada de *atividade na flecha* (AOA - *Activity-on-arrow*) e, embora menos predominante que o PDM, é ainda a técnica escolhida em algumas áreas de aplicação. O ADM utiliza apenas relações de dependência do tipo fim/início e, às vezes, necessitam da criação de atividades “fantasmas” para definir corretamente o relacionamento lógico. O ADM pode ser feito manualmente ou no computador.

.3 Método do diagrama condicional (CDM - Conditional diagramming method). As técnicas de diagramação tais como GERT (*Graphical Evaluation and Review Technique* - Avaliação Gráfica e Técnicas de Revisão) e modelos de Sistemas Dinâmicos (*System Dynamics*) permitem atividades não seqüenciais como *loops* (por exemplo, um teste deve ser repetido mais de uma

vez) ou desvios condicionados (por exemplo, a atualização de desenho que é necessária apenas se a inspeção detectar erros). Nem o PDM nem o ADM permitem *loops* ou desvios condicionados.

.4 Modelos de rede. Redes padronizadas podem ser utilizadas para auxiliar na preparação do diagrama de rede do projeto. Podem incluir todo o projeto ou apenas uma parte. Partes de uma rede são, freqüentemente, referenciadas como *subnets* ou *fragnets*. *Subnets* são especialmente úteis quando o projeto inclui várias características idênticas ou bastante similares tais como pisos na construção de prédios comerciais, pesquisas clínicas em projetos de pesquisas farmacêuticas ou módulos de programas em projetos de softwares.

4.1.2.3 Saídas do Seqüenciamento das Atividades

.1 Diagrama de rede do projeto. Um diagrama de rede de projeto é um esquema de apresentação das atividades do projeto e dos relacionamentos lógicos (dependências) entre elas. O diagrama de rede de um projeto pode ser elaborado manualmente ou no computador. Pode incluir detalhes de todo o projeto ou ter uma ou mais atividades sumarizadas. O diagrama deve ser acompanhado por uma descrição sumária que descreva a abordagem básica do seqüenciamento. Qualquer seqüência não usual deve ser amplamente descrita. Os diagramas de rede do projeto freqüentemente são chamados de *gráficos de PERT*. Historicamente, o PERT (*Program Evaluation and Review Technique*) foi um tipo específico de diagrama de rede.

.2 Atualizações da lista de atividades. Da mesma maneira que o processo de definição das atividades pode gerar atualizações na EAP, a preparação do diagrama de rede do projeto pode revelar situações em que uma atividade deve ser dividida ou mesmo redefinida com a finalidade de diagramar corretamente o relacionamento lógico.

4.1.3 Estimativa da Duração das Atividades

A estimativa de duração das atividades é o processo de gerar as durações das atividades para entrada do cronograma, a partir das informações do escopo do projeto e dos recursos disponíveis. As entradas para as estimativas de duração se originam tipicamente da pessoa, ou grupo dentro da equipe do projeto, para quem a natureza da atividade específica é mais familiar. A estimativa é, freqüentemente, construída de forma progressiva, e o processo considera a qualidade e a disponibilidade dos dados de entrada. Assim, pode-se assumir que a estimativa vai progressivamente se tornando mais precisa e com uma qualidade conhecida. A pessoa ou grupo da equipe do projeto que estiver mais familiarizada com a natureza de uma atividade específica deve fazer ou, no mínimo, aprovar a estimativa.

Estimar a quantidade ou número de períodos de trabalho exigidos para implementar uma atividade, freqüentemente, requererá também considerações relativas ao tempo de espera (*elapsed time*). Por exemplo, se a cura do concreto requererá 4 dias de *elapsed time*, isso pode requerer dois ou quatro períodos de trabalho baseados em: a) em qual dia da semana será iniciado e b) se os fins de semana serão, ou não, tratados como períodos de trabalho.

A maioria dos softwares que desenvolvem cronograma tratam esse problema automaticamente, através do uso de calendários alternativos de períodos de trabalho.

Embora a duração total do projeto possa também ser estimada, utilizando-se as ferramentas e técnicas apresentadas aqui, ela é mais apropriadamente calculada na saída do desenvolvimento do cronograma. A equipe do projeto pode considerar a duração do projeto uma distribuição probabilística (usando técnicas de probabilidade) ou uma estimativa unívoca (usando técnicas determinísticas).

O modelo do processo de estimativa de duração das atividades pode ser visualizado através da Figura 4.5.

4.1.3.1 Entradas para a Estimativa da Duração das Atividades

.1 Lista de atividades. A lista de atividades deve incluir todas as atividades que serão realizadas no projeto. Deve ser organizada como uma extensão da EAP para assegurar que esta está completa e que não inclui qualquer atividade que não seja requerida como parte do escopo do projeto. Assim como a EAP, a lista de atividades deve incluir descrições de cada atividade para garantir que os membros da equipe do projeto entenderão como o trabalho deverá ser feito

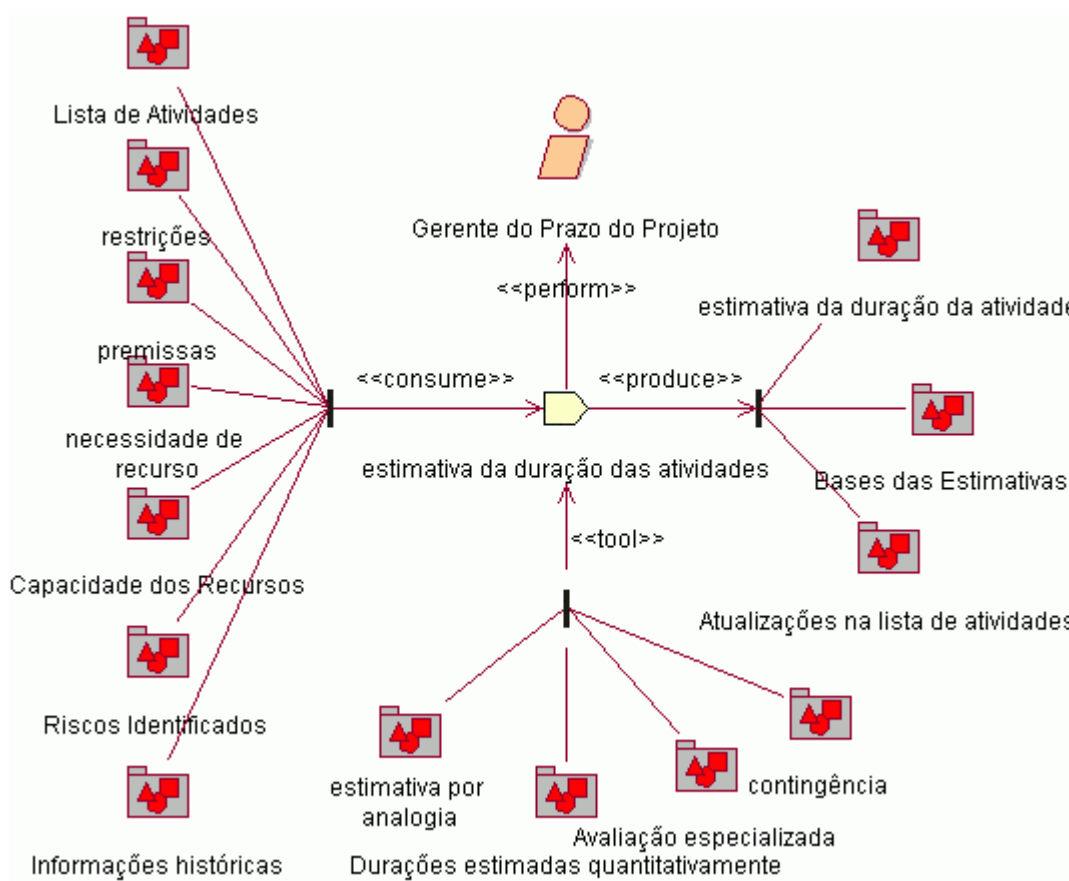


FIGURA 4.5 - Modelo do processo estimativa da duração das atividades.

.2 Restrições. As restrições são fatores que limitarão as opções da equipe de gerência do projeto; um exemplo seria a atribuição de durações máximas para as atividades.

.3 Premissas. Suposições são fatores que, para os propósitos do planejamento, são consideradas verdadeiros, reais, ou certos. Premissas afetam todos os aspectos do planejamento do projeto e são parte da elaboração progressiva do projeto. Equipes de projeto freqüentemente identificam, documentam e validam as premissas como parte de seus processos de planejamento.

Um exemplo seria estabelecer as durações máximas a partir dos períodos de revisões de progresso. Por exemplo, a duração máxima ser dois períodos de revisão.

.4 Necessidade de recursos. A duração da maioria das atividades será significativamente influenciada pelos recursos a elas designadas. Por exemplo, duas pessoas trabalhando juntas podem ser capazes de completar uma atividade de desenho na metade do tempo que levariam para fazê-lo individualmente. Por outro lado, uma pessoa trabalhando meio expediente em uma atividade, geralmente levará , no mínimo, duas vezes o tempo que gastaria trabalhando o expediente completo. Entretanto, na medida em que se incorporam mais recursos, os projetos podem sofrer sobrecarga de comunicação que reduz a produtividade e acarreta um crescimento menor na produção, quando comparada com o aumento de recursos.

.5 Capabilidade dos recursos. A duração da maioria das atividades será significativamente influenciada pela capabilidade dos recursos humanos e materiais a elas designados. Por exemplo, se dois recursos, um sênior e outro junior, forem alocados em tempo integral numa equipe, espera-se que na maioria das vezes o recurso sênior seja capaz de realizar uma determinada atividade em menos tempo que o júnior.

.6 Informações históricas. As informações históricas das durações mais prováveis de muitas categorias das atividades geralmente estão disponíveis em uma ou mais das seguintes fontes:

- Arquivos de projeto - as organizações envolvidas no projeto podem manter registros de projetos anteriores que sejam suficientemente detalhados para auxiliar o desenvolvimento da estimativa de duração das atividades. Em algumas áreas de aplicação, os membros individuais podem manter registros.
- Estimativa de durações em bases de dados comerciais - informações históricas estão freqüentemente disponíveis comercialmente. Estas bases de dados tendem a ser especialmente úteis quando as durações não são dependentes do conteúdo presente do trabalho (por exemplo, quanto tempo leva a cura do concreto, quanto tempo uma agência governamental geralmente leva para responder a certos tipos de requisição).
- Conhecimento da equipe do projeto - os membros individuais da equipe do projeto podem lembrar-se de estimativas ou dados reais anteriores. Embora essas referências possam ser úteis, geralmente são menos confiáveis que os resultados documentados.

.7 Riscos identificados. A equipe do projeto considera também os riscos identificados quando está produzindo as estimativas de duração das atividades, uma vez que os riscos (ameaças ou oportunidades) podem ter uma significativa influência na duração. A equipe do projeto analisa o efeito dos riscos e considera a sua incorporação ou não (e em que extensão) no *baseline* da duração de cada atividade. São incluídos os riscos com alta probabilidade ou impacto.

4.1.3.2 Ferramentas e Técnicas para a Estimativa da Duração das Atividades

.1 Avaliação especializada. As durações, geralmente, são difíceis de estimar, por causa do número de fatores que podem influenciá-las (por exemplo, nível dos recursos, produtividade dos recursos). A avaliação especializada baseada em informações históricas deve ser usada sempre que possível. Se tal conhecimento especializado não está disponível, as estimativas são inerentemente incertas e arriscadas.

.2 Estimativas por analogia. Nas estimativas por analogia, também chamadas de estimativas de cima para baixo (*top-down*), usam-se os valores reais de durações de projetos anteriores ou similares para estimar a duração de uma atividade futura. Ela é freqüentemente utilizada para estimar a duração do projeto quando existe uma quantidade limitada de informações detalhadas sobre ele (por exemplo, nas fases iniciais do projeto). Estimativas por analogia são uma forma de avaliação especializada.

As estimativas por analogia são mais confiáveis quando (a) as atividades anteriores são semelhantes de fato e não apenas na aparência e (b) os indivíduos que preparam as estimativas têm o conhecimento especializado necessário. Cada categoria específica de trabalho (por exemplo, quantidade de desenhos, metros de cabo, toneladas de aço, etc) definida conforme o esforço da engenharia/desenho, quando multiplicada pela taxa unitária da produtividade (isto é, horas/desenho, metros de cabo/hora, etc) pode ser usada para estimar a duração das atividades.

.4 Tempo de reserva (contingência). A equipe do projeto pode, a seu critério, incorporar uma porção adicional de tempo, chamada contingência, “buffer” ou reserva de tempo, que pode ser adicionada à duração da própria atividade, ou em outro ponto do cronograma, como “reserva” de risco do cronograma. Este tempo de reserva pode ser um percentual da duração estimada, ou um número fixo de períodos de trabalho. O tempo de reserva pode, mais tarde, ser

reduzido ou eliminado, na medida em que haja informação mais precisa sobre o projeto. Este tempo de reserva deve ser documentado junto com os demais dados e premissas.

4.1.3.3 Saídas da Estimativa da Duração das Atividades

.1 Estimativas de duração das atividades. As estimativas de duração das atividades são avaliações quantitativas da quantidade mais provável de períodos de trabalho que será requerida para se completar uma atividade. As estimativas de duração das atividades devem sempre incluir alguma indicação da faixa de variação dos possíveis resultados. Por exemplo:

- 2 semanas +/- 2 dias para indicar que a atividade levará no mínimo 8 dias e não mais que 12 dias para ser concluída.
- 15 por cento de probabilidade de exceder 3 semanas para indicar uma elevada probabilidade – 85 por cento que a atividade levará 3 semanas ou menos. O Capítulo 11, Gerência do Risco do Projeto, inclui uma discussão mais detalhada da estimativa da incerteza.

.2 Bases para a estimativa. As premissas feitas na elaboração das estimativas devem ser documentadas.

.3 Atualizações da lista de atividades. Da mesma maneira que o processo de definição das atividades pode gerar atualizações na EAP, a preparação do diagrama de rede do projeto pode revelar situações em que uma atividade deve ser dividida ou mesmo redefinida com a finalidade de diagramar corretamente o relacionamento lógico.

4.1.4 Desenvolvimento do Cronograma

Desenvolver o cronograma significa determinar as datas de início e fim para as atividades do projeto. Se as datas de início e fim não forem realistas, é

improvável que o projeto termine conforme planejado. O processo de desenvolvimento do cronograma deve ser conduzido de forma iterativa (junto com os processos que fornecem entradas, especialmente as estimativas das durações e dos custos) antes da determinação do cronograma do projeto. A Figura 4.6 mostra o modelo do processo.

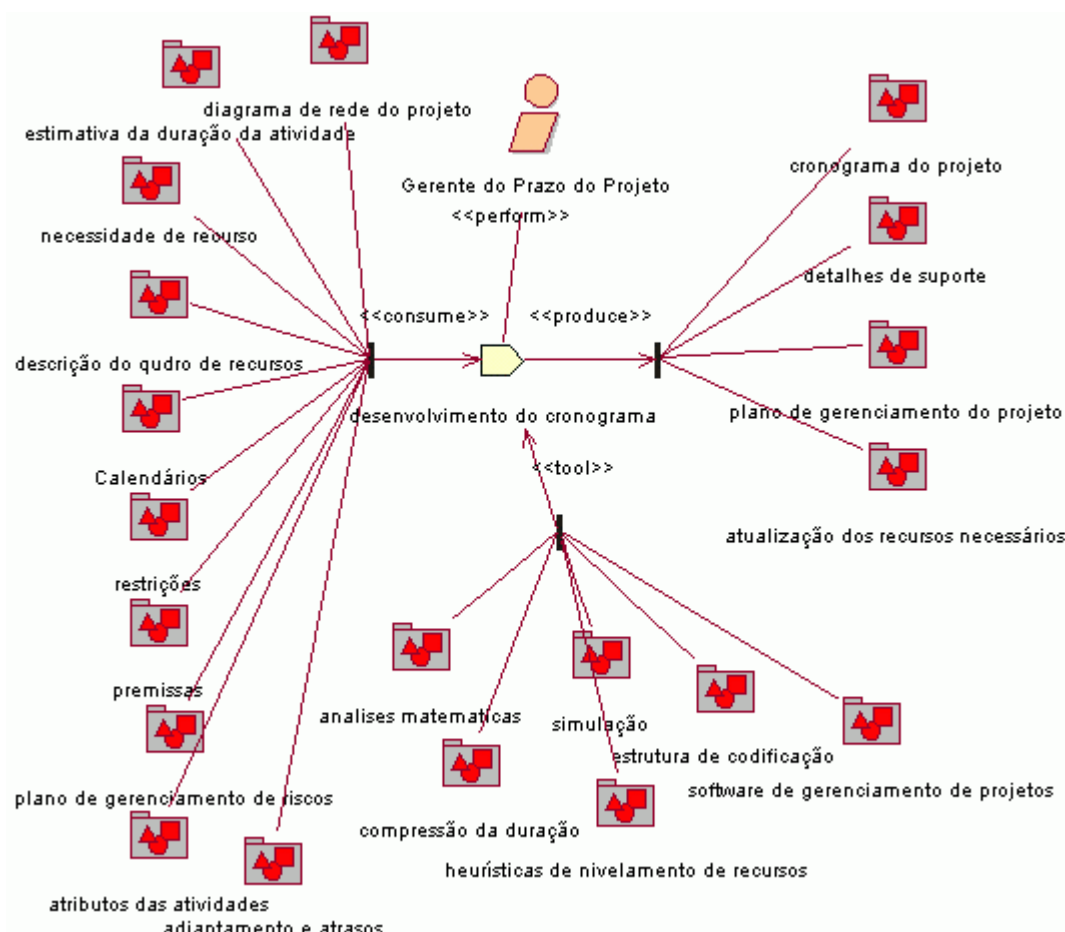


FIGURA 4.6 - Modelo do processo desenvolvimento do cronograma.

4.1.4.1 Entradas para o Desenvolvimento do Cronograma

.1 Diagramas de rede do projeto. Um diagrama de rede de projeto é um esquema de apresentação das atividades do projeto e dos relacionamentos lógicos (dependências) entre elas. O diagrama de rede de um projeto pode ser elaborado manualmente ou no computador. Pode incluir detalhes de todo o

projeto ou ter uma ou mais atividades sumarizadas (*hammocks*). O diagrama deve ser acompanhado por uma descrição sumária que descreva a abordagem básica do seqüenciamento. Qualquer seqüência não usual deve ser amplamente descrita.

Os diagramas de rede do projeto freqüentemente são chamados de gráficos de PERT. Historicamente, o PERT (*Program Evaluation and Review Technique*) foi um tipo específico de diagrama de rede.

.2 Estimativas de duração das atividades. As estimativas de duração das atividades são avaliações quantitativas da quantidade mais provável de períodos de trabalho que será requerida para se completar uma atividade.

As estimativas de duração das atividades devem sempre incluir alguma indicação da faixa de variação dos possíveis resultados.

.3 Necessidade de recursos. A duração da maioria das atividades será significativamente influenciada pelos recursos a elas designadas.

.4 Descrição do quadro de recursos. Para o desenvolvimento do cronograma, é necessário o conhecimento de quais recursos estarão disponíveis, em que tempo e em quais padrões. Por exemplo: os recursos compartilhados ou críticos podem ser especialmente difíceis de alocar visto que sua disponibilidade pode ser altamente variável. A quantidade de detalhes e o nível de especialização na descrição do quadro de recursos variará. Por exemplo, para o desenvolvimento do cronograma preliminar de um projeto de consultoria, pode ser necessário apenas saber que dois consultores deverão estar disponíveis em um intervalo de tempo específico. Entretanto, no cronograma final do mesmo projeto, deverá identificar quais consultores específicos deverão estar disponíveis.

.5 Calendários. Os calendários do projeto e dos recursos identificam os períodos nos quais o trabalho será considerado. Os *calendários do projeto*

afetam todos os recursos (por exemplo, alguns projetos serão executados apenas no horário comercial enquanto outros ocuparão três turnos). Os calendários dos recursos afetam recursos específicos ou uma categoria de recursos (por exemplo, um membro da equipe de projeto pode estar de férias ou em algum programa de treinamento; o contrato de trabalho pode limitar certos trabalhadores a certos dias da semana).

.6 Restrições. As restrições são limitações que restringirão as opções da equipe de gerenciamento do projeto. Há duas categorias principais de restrições que devem ser consideradas durante o desenvolvimento do cronograma:

- **Datas impostas** - Às vezes é necessário fixar o início de uma atividade para que ela não aconteça mais cedo do que determinada data. Da mesma forma, pode-se determinar o fim para que ela não ocorra depois de uma data limite. Embora todas as quatro restrições de data estejam disponíveis nos softwares de gerenciamento de projetos, as restrições “não iniciar mais cedo que” e “não terminar mais tarde que” são as mais utilizadas. Usos típicos das restrições de data incluem situações como uma oportunidade de mercado em um projeto de tecnologia, restrições meteorológicas em atividades externas, obediência da autoridade pública a questões ambientais, fornecimento de material por terceiros não representados no cronograma do projeto, etc.
- **Eventos chave ou marcos principais** - A conclusão de certos resultados em uma determinada data pode ser exigida pelo patrocinador do projeto, pelo cliente ou outras partes envolvidas. Uma vez programadas, essas datas tornam-se fixas e somente podem ser alteradas com grande dificuldade. Os marcos também podem ser usados para indicar interfaces com algum trabalho fora do projeto.

Neste caso, os marcos com datas restritivas possibilitam uma adequada interface com o trabalho externo.

.7 Premissas. Suposições são fatores que, para os propósitos do planejamento, são considerados verdadeiros, reais, ou certos. Premissas afetam todos os aspectos do planejamento do projeto e são parte da elaboração progressiva do projeto. Equipes de projeto freqüentemente identificam, documentam e validam as premissas como parte de seus processos de planejamento.

Um exemplo seria estabelecer as durações máximas a partir dos períodos de revisões de progresso. Por exemplo, a duração máxima ser dois períodos de revisão.

.8 Adiantamentos e atrasos. Qualquer uma das dependências podem requerer especificações de adiantamentos e atrasos com a finalidade de definir precisamente o relacionamento (por exemplo, pode haver um atraso de duas semanas entre a fabricação de uma peça do equipamento e a sua instalação ou uso). Um exemplo de avanço seria uma dependência “fim-para-início” com dez dias de avanço: a atividade sucessora começa dez dias antes do término da predecessora.

.9 Plano de gerenciamento de risco. O plano de gerência de risco descreve como a identificação, a análise qualitativamente e quantitativamente, o planejamento de respostas, a monitoração e o controle do risco será estruturado e realizado durante o ciclo de vida do projeto. O plano de gerência de risco não é dirigido para responder a riscos individuais.

.10 Atributos da atividade. Os atributos da atividade – incluindo a responsabilidade (quem realizará o trabalho), área geográfica ou prédio (onde o trabalho deve ser executado), e o tipo da atividade (sumária ou detalhe) – são muito importantes para a seleção e classificação posterior das atividades planejadas de uma forma conveniente para os usuários. A classificação da

EAP também é um atributo importante que permite um adequado ordenamento e classificação das atividades.

4.1.4.2 Ferramentas e Técnicas para o Desenvolvimento do Cronograma

.1 Análise Matemática. Envolve calcular datas teóricas de início e término para todas as atividades do projeto, sem considerar qualquer limitação no quadro de recursos. As datas resultantes não são o cronograma, mas indicam os períodos de tempo dentro dos quais as atividades devem ser programadas dadas as limitações de recursos e outras restrições conhecidas. As técnicas de análise matemática mais amplamente conhecidas são:

- Método de Caminho Crítico (CPM *Critical Path Method*). Calcula de forma determinística uma data única mais cedo e mais tarde, de início e de término para cada atividade, baseado na seqüência lógica especificada da rede e em uma duração estimada única. O enfoque do CPM é o cálculo da flutuação com a finalidade de determinar quais as atividades têm a menor flexibilidade no cronograma. Os principais algoritmos do CPM são freqüentemente utilizados em outros tipos de análises matemáticas.
- GERT – Graphical Evaluation and Review Technique. Permite o tratamento probabilístico tanto para a rede lógica quanto para as estimativas de duração das atividades (por exemplo, algumas atividades podem ser executadas por completo, algumas apenas em parte, e outras mais de uma vez).
- PERT – Program Evaluation and Review Technique. Usa uma rede seqüencial e uma estimativa de duração por média ponderada para calcular as durações das atividades. Embora existam diferenças superficiais, o PERT difere do CPM fundamentalmente porque usa distribuição de médias (valor esperado) em vez da estimativa mais

provável, originalmente usado no CPM. O PERT propriamente dito é atualmente muito pouco utilizado, embora as estimativas similares do PERT (PERT-like) sejam freqüentemente usadas nos cálculos de CPM.

.2 Compressão da duração. A compressão da duração é um caso especial de análise matemática que procura alternativas para reduzir o prazo do projeto sem alterar o seu escopo (por exemplo, satisfazer datas impostas ou outros objetivos de prazo). A compressão de duração inclui técnicas tais como:

- Colisão (*Crashing*) - quais compensações de custo e prazo são analisados para determinar como obter a maior compressão para o mínimo aumento de custo. As colisões nem sempre produzem alternativas viáveis e freqüentemente resultam em aumento de custo.
- Caminho Rápido (*Fast tracking*) - realizar em paralelo atividades que normalmente seriam feitas em seqüência (por exemplo, começar a escrever o código de um projeto de software antes que o desenho esteja completo, ou começar a construir a fundação de uma usina de processamento de petróleo antes de se alcançar 25 por cento da solução de engenharia do processo (*engineering point*). O caminho rápido freqüentemente resulta em retrabalho e usualmente aumenta o risco.

.3 Simulações. A simulação envolve o cálculo de múltiplas durações de projeto com diferentes grupos de premissas nas atividades. A técnica mais comum é a Análise Monte Carlo, na qual uma distribuição de resultados prováveis é definida para cada atividade e utilizada para calcular a distribuição dos resultados prováveis para o projeto. Além disso, análises *what-if* podem ser feitas utilizando a rede de lógica para estimular a geração de diferentes cenários, como atrasar a entrega de um componente principal, estender prazos específicos de engenharia, ou introduzir fatores externos (tais como greve, ou mudanças nos trâmites legais). Os resultados das simulações *what-if* podem

ser usadas para avaliar a viabilidade do cronograma sob condições adversas, e preparar planos de resposta/contingência para enfrentar ou mesmo resolver o impacto de situações inesperadas.

.4 Heurística de nivelamento dos recursos. As análises matemáticas freqüentemente produzem um cronograma preliminar de datas mais cedo que requer, durante certos períodos de tempo, mais recursos do que a disponibilidade real, ou requer alterações inviáveis nos níveis de recursos previstos. As heurísticas tais como “alocar os recursos escassos primeiramente para as atividades do caminho crítico” podem ser aplicadas para desenvolver um cronograma que reflita tal restrição. O nivelamento dos recursos freqüentemente resulta numa duração maior para o projeto do que o cronograma preliminar. Esta técnica é algumas vezes chamada de “Método Baseado em Recursos” (*Resource-based Method*), especialmente quando implementada com otimização computadorizada. A realocação de recursos das atividades mais críticas para as críticas é uma forma comum de retroceder o prazo, ou tanto quanto possível, à sua duração global originalmente prevista. Utilização de horas extras, fins de semana, ou turnos múltiplos devem também ser consideradas para a redução das atividades críticas. Aumento de produtividade baseados no uso de diferentes tecnologia e/ou automação (*automatic welding, electrical pipe cutters, etc*) são outras maneiras de reduzir as durações que haviam estendido o cronograma preliminar. O “caminho rápido”, quando aplicável, é outra possibilidade de redução de duração total do projeto. Alguns projetos podem ter um recurso finito e crítico, exigindo que este recurso seja programado tornando-se como referência a data final do projeto, isto é, conhecido como programação de recurso. *Critical Chain* é uma técnica que modifica o cronograma do projeto tendo em vista os recursos limitados.

.5 Softwares de gerência de projeto. Os softwares de gerência de projeto são amplamente usados no desenvolvimento do cronograma. Outros softwares, interagindo direta ou indiretamente com os softwares de gerenciamento de projetos, podem complementar as necessidades das outras áreas de

conhecimento. Esses produtos automatizam os cálculos das análises matemáticas e do nivelamento dos recursos e, conseqüentemente, permitem uma rápida avaliação sobre diversas alternativas de cronograma. São amplamente utilizados para imprimir ou apresentar as saídas do desenvolvimento do cronograma.

.5 Estrutura de codificação. As atividades devem ter uma estrutura de códigos que permitirá classificações e extrações em diferentes atributos designados às atividades, tais como responsável, área geográfica ou período, fase do projeto, nível de programação, tipo de atividade, e classificação EAP.

4.1.4.3 Saídas do Desenvolvimento do Cronograma

.1 Cronograma do projeto. O cronograma do projeto inclui, para cada atividade, no mínimo as datas de início e de término esperado para cada atividade detalhe (Nota: o cronograma do projeto se mantém preliminar até que os recursos designados tenham sido confirmados. Isto usualmente deverá acontecer no mais tardar até a conclusão do Desenvolvimento do Plano do Projeto) O cronograma do projeto pode ser apresentado de forma sumarizada (*master schedule*) ou em detalhes. Embora possa ser apresentado na forma tabular, ele é mais freqüentemente apresentado na forma gráfica utilizando-se um ou mais dos seguintes formatos:

- Diagrama de rede do projeto acrescido das informações de datas. Estes gráficos usualmente apresentam tanto a lógica do projeto quanto o caminho crítico das atividades.
- Gráficos de barras, também chamados de gráficos de *Gantt*, mostram as datas de início e término das atividades bem como as durações esperadas, e algumas vezes mostram as dependências. São relativamente fáceis de ler sendo, freqüentemente, usados em apresentações gerenciais.

- Gráficos de marcos, semelhantes aos gráficos de barras, porém somente identificando o início ou a conclusão dos principais resultados programados e as interfaces externas principais.

.2 Detalhes de suporte. Os detalhes do suporte do cronograma do projeto incluem, no mínimo, a documentação de todas as premissas e restrições identificadas. A quantidade de detalhamento adicional varia de acordo com a área de aplicação. Por exemplo:

- Em um projeto de construção, incluirá provavelmente itens tais como histograma de recursos, projeções de fluxo de caixa, e cronogramas de pedidos e entregas.
- Em um projeto eletrônico incluirá, provavelmente, apenas histogramas de recursos. As informações freqüentemente fornecidas pelos detalhes de suporte incluem, mas não estão limitadas por:
- Recursos requeridos por período de tempo, freqüentemente na forma de histogramas de recursos.
- Alternativas de cronograma (por exemplo, melhor ou pior caso, recurso nivelado ou não, com ou sem datas impostas).
- Reservas de contingência para cronograma.

.3 Plano de gerência do cronograma. O plano de gerência do cronograma define como as mudanças no cronograma serão gerenciadas. Pode ser formal ou informal, muito detalhado ou bastante amplo, dependendo das necessidades do projeto. É um elemento auxiliar do plano geral do projeto.

.4 Atualização da necessidade de recursos. A atualização do nivelamento dos recursos podem ter um efeito significativo nas estimativas preliminares quanto à necessidade de recursos

4.1.5 Controle do Cronograma

O controle do cronograma consiste em: a) influenciar os fatores que criam mudanças no cronograma, para garantir que as mudanças sejam benéficas, b) determinar que o cronograma foi alterado, e c) gerenciar as mudanças reais, quando e como elas ocorrem. O controle do cronograma deve estar fortemente integrado com os outros processos de Controle Integrado de Mudanças. A Figura 4.7 mostra o modelo do processo.

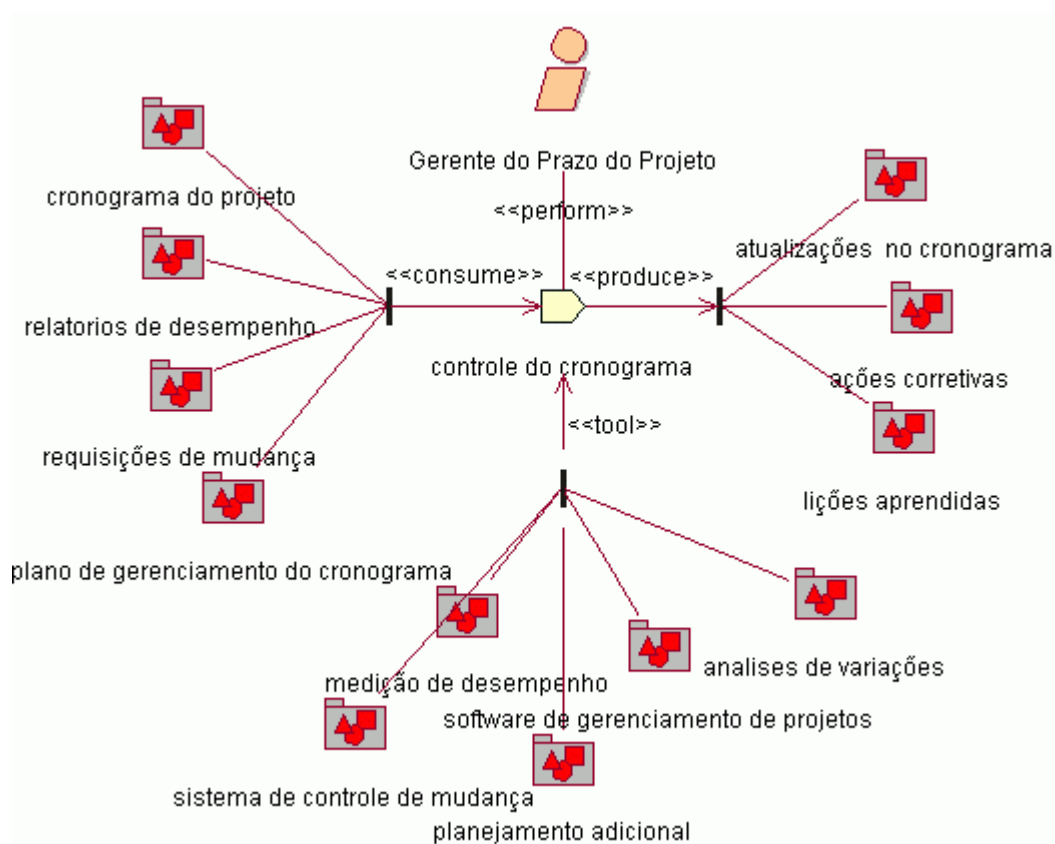


FIGURA 4.7 Modelo do processo controle do cronograma.

4.1.5.1 Entradas para o Controle do Cronograma

.1 Cronograma do projeto. O cronograma do projeto está descrito na Seção 6.4.3.1. O cronograma aprovado do projeto, chamado de cronograma base (que deve ser factível tanto em termos técnicos quanto de recursos) é um

componente do plano geral do projeto. Fornece a base para a medição e o reporte do desempenho do projeto.

.2 Relatórios de desempenho. Os relatórios de desempenho fornecem informações sobre o desempenho do cronograma, tais como que datas planejadas foram alcançadas e quais não foram. Podem também alertar a equipe de projeto para questões que poderão causar problemas no futuro.

.3 Requisições de mudança. As requisições de mudança podem ocorrer de muitas formas – oral ou escrita, direta ou indiretamente, iniciadas internamente ou externamente, e legalmente impostas ou opcionais. As mudanças podem exigir uma dilatação do cronograma ou mesmo permitir que ele seja encurtado.

.4 Plano de gerência do cronograma. O plano de gerência do cronograma define como as mudanças no cronograma serão gerenciadas. Pode ser formal ou informal, muito detalhado ou bastante amplo, dependendo das necessidades do projeto. É um elemento auxiliar do plano geral do projeto.

4.1.5.2 Ferramentas e Técnicas para o Controle do Cronograma

.1 Sistema de controle de mudanças do cronograma. Um sistema de controle de mudanças do cronograma define os procedimentos pelos quais o cronograma do projeto pode ser mudado. Isto inclui papéis de trabalho, sistemas de acompanhamento, e níveis de aprovação necessários para autorizar as mudanças. O controle de mudanças do cronograma deve estar articulado com o sistema integrado de controle de mudanças.

.2 Medição de desempenho. As técnicas de medição de desempenho ajudam a determinar a magnitude de quaisquer variações que ocorram. Uma parte importante do controle de mudanças no cronograma é decidir se a variação do cronograma exige uma ação corretiva. Por exemplo, um grande atraso em uma atividade que não é crítica pode ter um efeito pequeno no projeto global,

enquanto pequenos atrasos em atividades críticas ou “quase” críticas podem requerer ações imediatas.

.3 Planejamento adicional. Poucos projetos se desenvolvem exatamente de acordo com o plano. Mudanças em perspectiva podem requerer estimativas de duração de atividades novas ou revisadas, modificações na seqüência das atividades ou análise de cronogramas alternativos.

.4 Softwares de gerência de projeto. Os softwares de gerência de projeto são amplamente usados na gerência e controle de projetos. A capacidade dos softwares de gerência de projetos em acompanhar datas planejadas versus datas reais e prever os efeitos de mudanças no cronograma, reais ou potenciais, torna-os uma ferramenta útil para o controle do cronograma.

.5 Análise de variações. O desempenho das análises das variações durante o processo de monitoramento do cronograma é um elemento chave para o controle do prazo. A comparação das datas objetivo com as datas de início e fim previstas/reais fornece uma valiosa informação para a detecção de desvios e para a implementação de soluções corretivas no caso de atrasos. A variação de folgas é também um componente de planejamento essencial para se avaliar o desempenho do prazo do projeto. Uma atenção especial deve ser tomada nas atividades críticas e “quase críticas” (analisando-se os dez caminhos “quase críticos” por ordem de suas folgas).

4.1.5.3 Saídas do Controle do Cronograma

1 Atualizações do cronograma. Uma atualização no cronograma é qualquer modificação nas informações de prazos que são utilizadas para gerenciar o projeto. As partes envolvidas afetadas devem ser notificadas. As atualizações do cronograma podem ou não requerer ajustes em outros aspectos do plano geral do projeto. *Revisões* são uma categoria especial de atualização do cronograma. As revisões são mudanças nas datas de início e término no

cronograma aprovado do projeto. Essas datas são geralmente revisadas apenas em resposta a mudanças no escopo ou nas estimativas. Em alguns casos, os atrasos no cronograma podem ser tão severos que é necessário um replanejamento (*rebaselining*) para a manutenção de dados realísticos que venham medir o desempenho. Entretanto, é preciso cuidado antes do replanejamento, uma vez que se perderão dados históricos em relação ao cronograma do projeto. O replanejamento deve ser usado somente como o último recurso para o controle do cronograma; a atribuição de novos objetivos de prazo deve ser a forma normal para se conduzir uma revisão de cronograma.

.2 Ações corretivas. A ação corretiva é tudo aquilo que é feito para compatibilizar o desempenho futuro do cronograma com o plano do projeto. Ações corretivas na área de gerência do tempo freqüentemente envolvem presteza: são ações especiais, tomadas para garantir a conclusão da atividade no prazo ou com o mínimo de atraso possível. As ações corretivas freqüentemente requerem uma análise de causas-raiz para identificar a causa real da variação, possibilitando que a recuperação do atraso possa ser planejada e executada para atividades programadas à frente no projeto e não apenas diretamente naquelas que causaram o desvio.

.3 Lições aprendidas. As causas das variações, as razões “por trás” das ações corretivas tomadas, e outros tipos de lições aprendidas no controle do cronograma devem ser documentadas para que se tornem parte de um banco de dados histórico tanto para o projeto em curso quanto para outros projetos da organização executora.

4.2 Gerenciamento do Custo do Projeto

O Gerenciamento do Custo do Projeto inclui os processos necessários para assegurar que o projeto será concluído dentro do orçamento aprovado. A Figura 4.8 fornece uma visão geral dos principais processos:

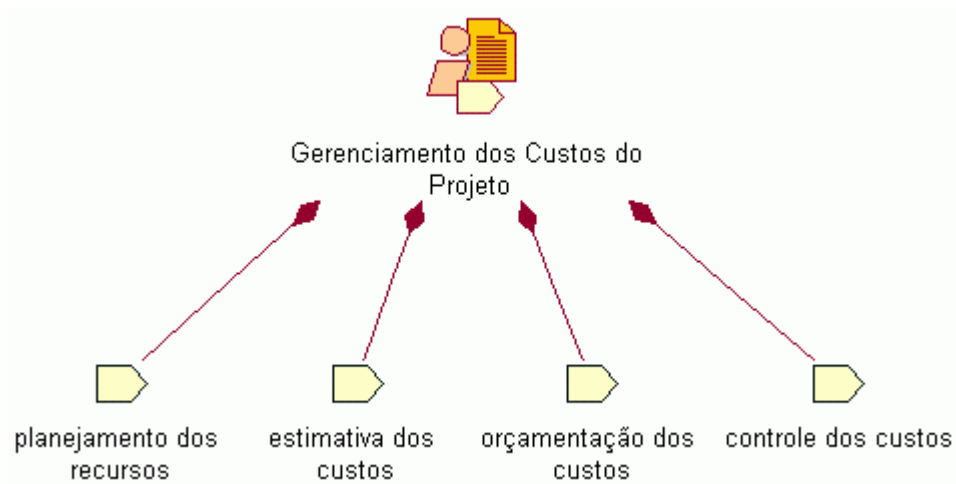


FIGURA 4.8 - Processos de gerenciamento dos custos do projeto.

- **Planejamento dos Recursos** – determinar quais recursos (pessoas, equipamentos, materiais) e em que quantidade devem ser usados para executar as atividades do projeto.
- **Estimativa dos Custos** – desenvolver uma aproximação (estimativa) dos custos dos recursos necessários para executar as atividades do projeto.
- **Orçamentação dos Custos** – alocar as estimativas de custos globais aos itens individuais de trabalho.
- **Controle dos Custos** – controlar as mudanças no orçamento do projeto.

Estes processos interagem-se mutuamente e com os processos das outras áreas de conhecimento. Cada processo pode envolver o esforço de um ou mais indivíduos ou grupos de indivíduos dependendo das necessidades do projeto. Cada processo ocorre, geralmente, pelo menos uma vez em cada fase do projeto.

Embora os processos sejam aqui apresentados como elementos discretos e com interfaces bem definidas, na prática eles podem sobrepor-se e interagir de formas aqui não especificadas.

A gerência do custo do projeto consiste, fundamentalmente, nos custos dos recursos necessários à implementação das atividades do projeto. Entretanto, a gerência do custo do projeto deve, também, considerar os efeitos das decisões do projeto no custo de utilização do produto do projeto. Por exemplo, limitar o número de revisões do projeto pode reduzir os custos do projeto à custa de um aumento no custo de operação do cliente. Esta visão mais ampla da gerência do custo do projeto é, freqüentemente, chamada de custo do ciclo de vida (*life-cycle costing*). As técnicas de custo de ciclo de vida e engenharia de valor são usadas para reduzir custo e prazo, melhorar qualidade e desempenho e otimizar a tomada de decisão.

Em muitas áreas de aplicação, a previsão e análise da perspectiva de desempenho financeiro do produto do projeto são feitas fora do ambiente do projeto. Em outras (por exemplo, projetos de negócios financeiros), a gerência do custo do projeto também inclui esse trabalho. Quando essas previsões e análises estão incluídas, a gerência do custo do projeto inclui processos adicionais e diversas técnicas de administração geral, tais como previsão de retorno do investimento, fluxo de caixa, análises de retorno, entre outras.

A gerência do custo do projeto deve considerar as necessidades de informações das partes envolvidas do projeto – diferentes interessados podem avaliar os custos do projeto de maneiras diferentes e em diferentes momentos. Por exemplo: o custo de aquisição de um item pode ser medido quando de sua contratação, da ordem de compra, da entrega, do armazenamento ou do registro para fins contábeis.

Quando os custos do projeto são usados como componentes de um sistema de premiação e reconhecimento, os custos controláveis e não controláveis devem

ser estimados e orçados separadamente para assegurar que os prêmios reflitam o desempenho real.

Em alguns projetos, especialmente nos menores, o planejamento dos recursos, a estimativa dos custos e o orçamento são tão integrados que podem ser vistos como um único processo. Esses processos são aqui apresentados como processos distintos porque as ferramentas e técnicas são diferentes para cada um. A capacidade de influência nos custos é maior nos estágios iniciais do projeto, e é por essa razão que uma definição mais cedo do escopo é importante, assim como completa identificação de requisitos e execução de um plano consistente.

O ator que participa na execução dos processos de gerenciamento dos custos é o Gerente dos Custos, o qual será representado nos próximos diagramas pelo *RoleKind* correspondente. Vale salientar, que neste trabalho foi este o ator definido, porém ele pode variar em outros casos ou cenários particulares, porém, ele deve ter o perfil de conhecer o escopo do trabalho a ser executado as atividades definidas para a execução e conclusão do trabalho, a fim de que o mesmo seja capaz de planejar, estimar, orçar e controlar os custos do projeto.

4.2.1 Planejamento dos Recursos

O planejamento dos recursos significa determinar quais recursos físicos (pessoas, equipamentos e materiais) são necessários, que quantidades de cada um devem ser usadas, e quando serão necessárias para a realização das atividades do projeto. Deve ser conduzido de forma bem coordenada com a estimativa dos custos. Por exemplo:

- Uma equipe de projeto de construção necessita estar familiarizada com os códigos de construção local. Tal conhecimento é, freqüentemente, prontamente disponibilizado pelos fornecedores

locais. Contudo, se no quadro local de profissionais falta experiência em uma técnica de construção não usual ou especial, a previsão de um custo adicional para um consultor poderá ser a melhor forma para assegurar conhecimento dos códigos de construção local.

- Uma equipe de projetistas do setor automotivo deve estar familiarizada com as mais recentes técnicas de montagem automotiva. O conhecimento requerido pode ser adquirido através da contratação de uma consultoria, do envio de um projetista para um seminário de robótica ou da inclusão de algum profissional de manufatura como membro da equipe. O modelo do processo de planejamento dos recursos pode ser visualizado na Figura 4.9.

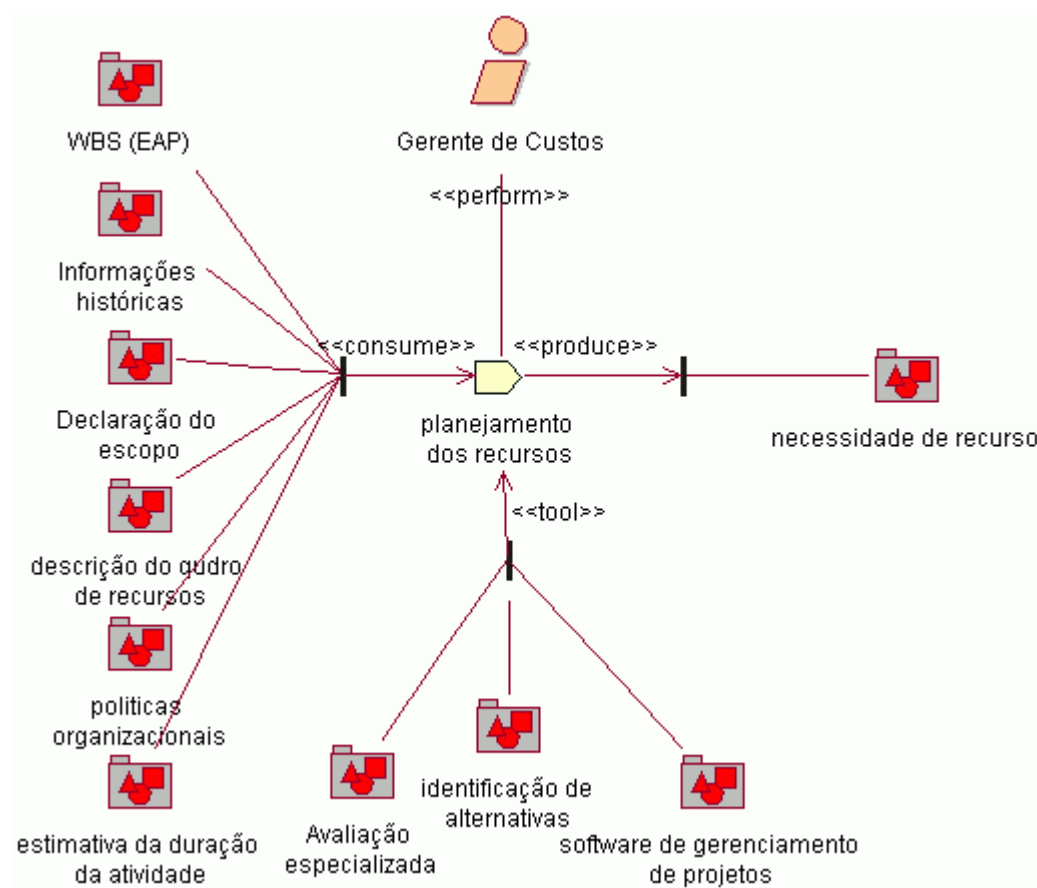


FIGURA 4.9 - Modelo do processo de planejamento dos recursos.

4.2.1.1 Entradas para o Planejamento dos Recursos

.1 Estrutura analítica do projeto - EAP. A EAP identifica os subprodutos e processos do projeto que necessitarão de recursos e, portanto, é a entrada fundamental do planejamento de recursos. Qualquer saída relevante dos outros processos de planejamento devem ser alimentadas através da EAP, para garantir um controle apropriado.

.2 Informações históricas. Devem ser usadas, sempre que disponíveis, as informações históricas relativas aos tipos de recursos que foram requeridos em trabalhos similares de projetos anteriores.

.3 Declaração do escopo. A declaração do escopo contém a justificativa e os objetivos do projeto devendo ambas ser consideradas, explicitamente, durante o planejamento de recursos.

.4 Descrição do quadro de recursos. O conhecimento de quais recursos (pessoas, equipamentos, materiais) estão potencialmente disponíveis é necessário para o planejamento dos recursos. A quantidade de detalhes e o nível de especialização na descrição do quadro de recursos variará. Por exemplo, durante as primeiras fases de um projeto de design de engenharia, o quadro pode incluir “engenheiros júnior e sênior”, apenas em termos gerais. Entretanto, durante as últimas fases do mesmo projeto, o quadro pode ser limitado àqueles indivíduos que têm conhecimento significativo sobre o projeto como resultado de terem trabalhado nas fases iniciais.

.5 Políticas organizacionais. As políticas da organização, relativas tanto ao quadro de pessoal quanto a aluguel ou compra de suprimentos e equipamentos, devem ser consideradas durante o planejamento dos recursos.

.6 Estimativas de duração das atividades. As estimativas de duração das atividades são avaliações quantitativas da quantidade mais provável de períodos de trabalho que será requerida para se completar uma atividade.

4.2.1.2 Ferramentas e Técnicas para o Planejamento dos Recursos

.1 Avaliação especializada. A avaliação especializada freqüentemente será requerida para avaliar as entradas deste processo. Tal conhecimento específico pode ser obtido de qualquer grupo ou indivíduo com conhecimento ou treinamento especializado e está disponível a partir de muitas fontes, incluindo:

- Outras unidades da organização executora.
- Consultores.
- Profissionais e associações técnicas.
- Grupos industriais.

.2 Identificação de alternativas. Este é um termo genérico para qualquer técnica usada para gerar diferentes abordagens do projeto. Existem várias técnicas de gerenciamento freqüentemente usadas, sendo as mais comuns o *brainstorming* e o *lateral thinking* (pensamento lateral).

.3 Software de gerenciamento de projetos. Um software de gerenciamento de projetos tem a capacidade de auxiliar na organização do quadro de recursos. Dependendo da sofisticação do software, pode-se trabalhar com disponibilidades, taxas e calendários de recursos.

4.2.1.3 Saídas do Planejamento dos Recursos

.1 Recursos requeridos. A saída do processo de planejamento dos recursos é a descrição de que tipos de recursos são necessários e em que quantidade para cada elemento no nível mais baixo da EAP. Esses recursos serão obtidos através da montagem da equipe ou contratação.

4.2.2 Estimativa dos Custos

A estimativa dos custos envolve desenvolver uma aproximação (estimativa) dos custos dos recursos necessários para completar as atividades do projeto. Na busca de uma aproximação dos custos, a pessoa responsável pela estimativa considera as causas de variação da estimativa final, a fim de melhor gerenciar o projeto.

Quando o projeto é realizado sob contrato, devem ser tomados cuidados para distinguir custos estimados de preço. A estimativa dos custos envolve elaborar uma avaliação quantitativa dos resultados prováveis - quanto custará para a organização o fornecimento do produto ou serviço envolvido? O preço é uma decisão de negócio – quanto a organização cobrará pelo produto ou serviço – que usa as estimativas de custo apenas como uma entre várias considerações.

A estimativa dos custos inclui identificar e considerar várias alternativas de custo. Por exemplo: na maioria das áreas de aplicação, considera-se amplamente que o trabalho adicional durante a fase de projeto (design) tem o potencial de redução do custo na fase de produção. O processo de estimativa dos custos deve considerar se o custo do trabalho adicional na fase de projeto será contrabalançado pela economia esperada.

A Figura 4.10 mostra o modelo do processo de estimativa dos custos.

4.2.2.1 Entradas para a Estimativa dos Custos

.1 Estrutura Analítica do Projeto. A EAP é usada para organizar a estimativa dos custos e assegurar que todo o trabalho identificado foi estimado.

.2 Necessidades de recursos. A saída do processo de planejamento dos recursos é a descrição de que tipos de recursos são necessários e em que quantidade para cada elemento no nível mais baixo da EAP. Esses recursos serão obtidos através da montagem da equipe ou contratação.

.3 Taxas de recursos. O indivíduo ou grupo que elabora a estimativa deve ter o conhecimento das taxas unitárias (por exemplo, custo horário de pessoal, custo do volume de material por metro cúbico) de cada recurso com a finalidade de calcular os custos do projeto. Se as taxas reais não forem conhecidas, deve se usar estimativas para as mesmas.

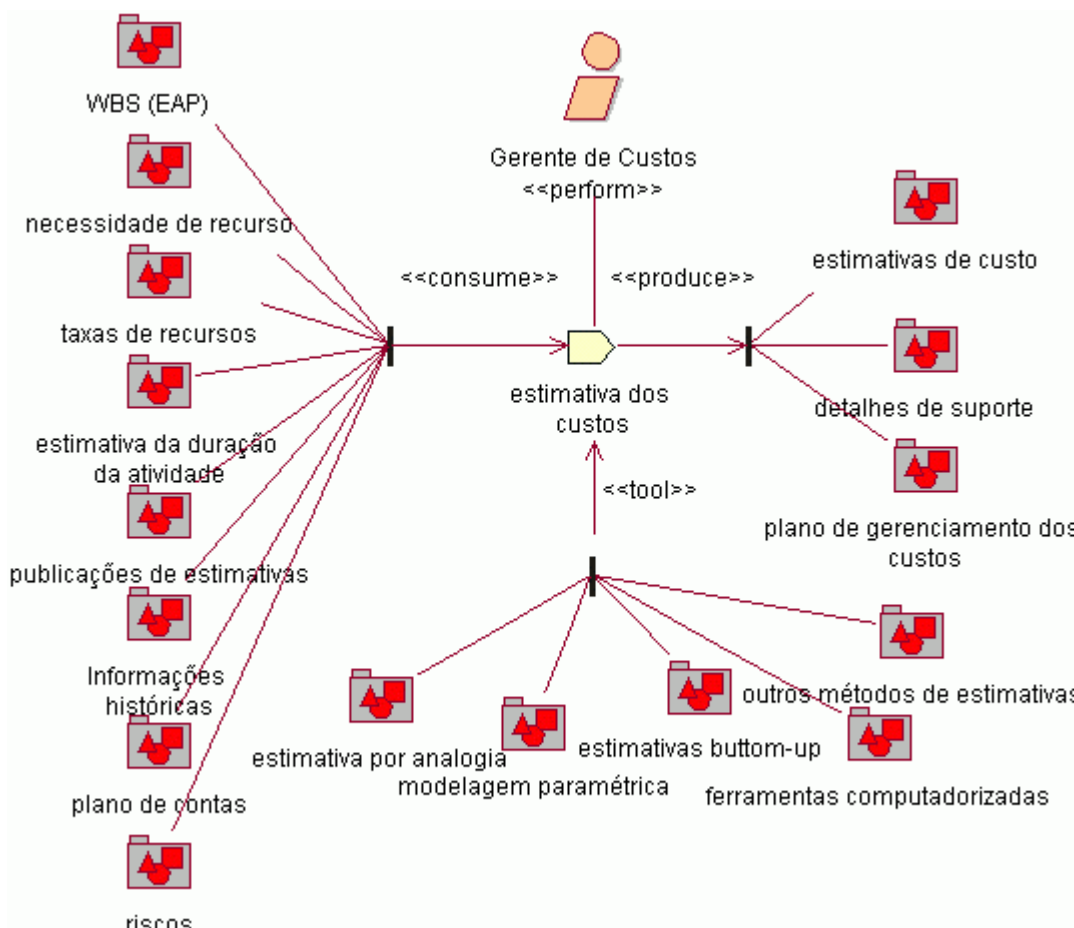


FIGURA 4.10 - Modelo do processo estimativa dos custos.

.4 Estimativas da duração das atividades. As estimativas de duração das atividades afetarão as estimativas dos custos de qualquer projeto no qual o orçamento considera os custos de financiamento (por exemplo, taxas de juros).

.5 Publicações de estimativas. Dados de custos estimados disponíveis comercialmente.

.6 Informações históricas. As informações de custos referentes a diversas categorias de recursos freqüentemente estão disponíveis em uma ou mais das seguintes fontes:

- Arquivos de projetos – as organizações envolvidas no projeto podem conter em seus arquivos resultados de projetos anteriores, os quais, sendo suficientemente detalhados, auxiliarão na elaboração da estimativa dos custos. Em algumas áreas de aplicação, os próprios membros da equipe podem possuir tais registros.
- Base de dados comerciais - informações históricas usualmente estão disponíveis comercialmente.
- Conhecimento da equipe do projeto - os membros individuais da equipe de projeto podem lembrar-se de dados reais ou estimativas anteriores. Embora essas memórias possam ser úteis, geralmente são menos confiáveis do que os resultados documentados.

.7 Plano de contas. O plano de contas descreve a estrutura de codificação utilizada pela organização para reportar as informações financeiras para o seu sistema contábil. As estimativas do custo do projeto devem ser alocadas na categoria contábil correta.

.8 Riscos. A equipe do projeto considera as informações de risco quando está produzindo as estimativas de custo, uma vez que os riscos (sejam ameaças ou oportunidades) podem ter um significativo impacto no custo. A equipe do projeto considera os reflexos dos efeitos do risco quando inclui as estimativas de custo para cada atividade.

4.2.2.2 Ferramentas e Técnicas para a Estimativa dos Custos

.1 Estimativas por analogias. Nas estimativas por analogia, também chamadas de estimativas *top-down*, usam-se os custos reais de projetos

anteriores similares como base para a estimativa do custo do projeto corrente. É freqüentemente usada na estimativa dos custos totais do projeto quando existe uma quantidade limitada de informações detalhadas sobre o projeto (por exemplo, nas fases iniciais). As estimativas por analogia são uma forma de avaliação especializada.

As estimativas por analogia são geralmente menos dispendiosas que outras técnicas, mas, também, freqüentemente menos precisas. São mais confiáveis quando: (a) os projetos anteriores são semelhantes de fato e não apenas na aparência (b) os indivíduos ou grupos que estão preparando as estimativas possuem o *expertise* necessário.

.2 Modelagem paramétrica. No modelo paramétrico utilizam-se características do projeto (parâmetros) em modelos matemáticos para prever os custos do projeto. Os modelos podem ser simples (as construções residenciais custarão um certo valor por unidade de área construída) ou complexos (um modelo de custos de desenvolvimento de software usa 13 fatores de ajuste com 5 a 7 pontos a serem analisados em cada um deles).

Tanto o custo quanto a precisão do modelo paramétrico variam amplamente. Serão provavelmente mais confiáveis quando: (a) as informações históricas usadas no desenvolvimento do modelo forem precisas, (b) os parâmetros usados no modelo forem prontamente quantificáveis, e (c) o modelo for escalonável (por exemplo, quando ele funcionar bem tanto para grandes projetos quanto para projetos menores).

.3 Estimativas de baixo para cima (Bottom-up). Esta técnica envolve estimar o custo das atividades individuais dos pacotes de trabalho, depois sumará-los ou agregá-los para obter a estimativa total do projeto.

O custo e a precisão das estimativas *botton-up* são influenciados pelo tamanho e complexidade das atividades individuais dos pacotes de trabalho: atividades menores aumentam tanto o custo quanto a precisão do processo de estimativa.

A equipe de gerenciamento do projeto deve pesar o aumento da precisão contra o custo adicional.

.4 Ferramentas computadorizadas. As ferramentas computadorizadas, tais como softwares de gerência de projeto e planilhas (*spreadsheets*) são amplamente utilizadas no apoio à estimativa dos custos. Tais produtos podem simplificar o uso das ferramentas descritas anteriormente e, portanto, agilizar as análises entre várias alternativas de custo.

.5 Outros métodos de estimativas de custo. Por exemplo, análise de propostas de vendedores.

4.2.2.3 Saídas da Estimativa dos Custos

.1 Estimativas de custo. As estimativas de custo são avaliações quantitativas dos prováveis custos dos recursos requeridos para a implementação das atividades do projeto. Podem ser apresentadas de forma detalhada ou sumarizadas.

Os custos devem ser estimados para todos os recursos que serão contabilizados no projeto. Isto inclui, mas não está limitado a mão-de-obra, materiais, suprimentos, e categorias especiais, tais como efeitos inflacionários ou reserva de custo.

As estimativas de custos são geralmente expressas em unidades monetárias (dólar, euros, yen, etc.) com a finalidade de facilitar comparações tanto internamente no projeto quanto entre projetos. Em alguns casos, as estimativas poderão ser obtidas usando outras unidades de medida tais como homens-hora ou homens-dia, com os seus custos estimados, para facilitar o apropriado controle gerencial. As estimativas de custo geralmente também levam em conta aspectos do plano de respostas aos riscos, tais como planos de contingência.

As estimativas de custo podem ser melhoradas por refinamentos ocorridos durante o curso do projeto como reflexo dos detalhes adicionais disponíveis. Em algumas áreas de aplicação, existem orientações de quando tais refinamentos devem ser feitos e qual o grau de precisão esperado. Por exemplo, a “*Association for the Advancement of Cost Engineering (ACE) International*” identificou uma progressão de cinco tipos de estimativas para os custos de construção durante o projeto: ordem de grandeza (*order of magnitude*), conceitual, preliminar, definitiva e controle.

.2 Detalhes de suporte. Os detalhes de suporte, no que se refere à estimativa de custos, incluem:

- Uma descrição do escopo do trabalho estimado. É freqüentemente provido através de referência à EAP.
- Documentação das bases da estimativa, ou seja, como foi desenvolvida.
- Documentação de qualquer premissa adotada.
- Uma indicação do intervalo de resultados possíveis, por exemplo R\$ 10.000 +/- R\$ 1.000 para indicar que o item está previsto custar entre R\$ 9.000 e R\$ 11.000. A quantidade e o tipo dos detalhes adicionais variam por área de aplicação. A retenção de esboços e mesmo de rascunhos pode ser valiosa quando fornece uma melhor compreensão de como as estimativas foram desenvolvidas.

.3 Plano de gerenciamento do custo. O plano de gerenciamento do custo descreve como as variações no custo serão gerenciadas (por exemplo, respostas diferentes para problemas maiores ou menores). Um plano de gerenciamento de custo pode ser formal ou informal, muito detalhado ou bastante amplo, baseado nas necessidades das partes envolvidas do projeto. É um elemento componente do plano do projeto.

4.2.3 Orçamento dos Custos

A orçamentação dos custos envolve alocar as estimativas dos custos globais às atividades individuais dos pacotes de trabalho, com a finalidade de estabelecer um *baseline* de custo para medir o desempenho do projeto. A realidade pode impor que as estimativas sejam feitas depois da aprovação do orçamento, mas sempre que possível elas devem ser elaboradas antes de se submeter o orçamento para aprovação. O modelo do processo de orçamentação de custos é apresentado na Figura 4.11.

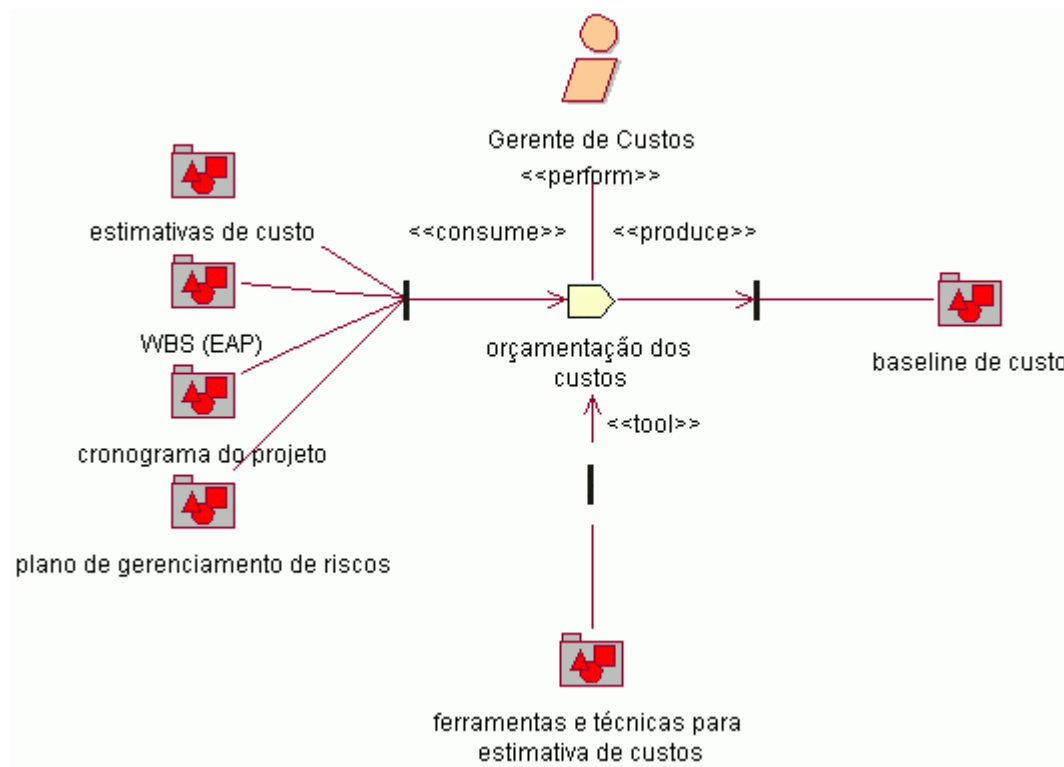


FIGURA 4.11 - Modelo do processo de orçamentação dos custos.

4.2.3.1 Entradas para a Orçamentação dos Custos

.1 Estimativas de custo. As estimativas de custo são avaliações quantitativas dos prováveis custos dos recursos requeridos para a implementação das

atividades do projeto. Podem ser apresentadas de forma detalhada ou sumarizadas.

.2 EAP. A EAP identifica os elementos do projeto para os quais o custo será alocado.

.3 Cronograma do projeto. O cronograma do projeto inclui as datas planejadas de início e as datas esperadas de término para os elementos do projeto cujos custos serão alocados. Esta informação é necessária para que se aloque os custos para o período de tempo em que eles forem realmente ocorrer.

.4 Plano de gerenciamento de riscos. Saliencia-se aqui que o plano de gerenciamento de riscos freqüentemente inclui a contingência de custos, que pode ser determinada a partir da precisão esperada da estimativa.

4.2.3.2 Ferramentas e Técnicas para a Orçamentação dos Custos

.1 Ferramentas e técnicas para a estimativa de custo. As ferramentas e técnicas para o desenvolvimento das estimativas de custo são também usadas no desenvolvimento dos orçamentos das atividades dos pacotes de trabalho.

4.2.3.3 Saídas da Orçamentação dos Custos

.1 Baseline de Custo. O *baseline* de custo é o orçamento referencial (*time-phased*) que será utilizado para medir e monitorar o desempenho dos custos do projeto. Ele é desenvolvido totalizando-se as estimativas de custo por período e, usualmente, é apresentado na forma de Curva-S.

Muitos projetos, especialmente os maiores, podem ter vários *baselines* de custo para medir diferentes aspectos do desempenho de custo. Por exemplo, um plano de gastos ou uma previsão de fluxo de caixa são *baselines* de custo para medição de desembolso.

4.2.4 Controle dos Custos

O controle dos custos está associado a: a) influenciar os fatores que criam as mudanças no *baseline* de custo de forma a garantir que estas mudanças sejam benéficas, b) determinar que o *baseline* de custo foi alterado, e c) gerenciar as mudanças reais, quando e como elas surgirem. O controle dos custos inclui:

- Monitorar o desempenho do custo para detectar e entender as variações do plano.
- Assegurar que todas as mudanças apropriadas estão registradas corretamente no *baseline* de custo.
- Informar as partes envolvidas afetadas sobre as mudanças autorizadas
- Atuar no sentido de manter os custos esperados dentro de limites aceitáveis

O controle de custo inclui pesquisar os “porquês” das variações, tanto positivas quanto negativas. Deve estar fortemente integrado com os outros processos de controle (o controle de mudança de escopo, o controle do cronograma e o controle da qualidade). Por exemplo, uma resposta inadequada para variações do custo pode causar problemas de qualidade ou de prazo, ou produzir, mais adiante no projeto, um nível de risco inaceitável. O modelo do processo de controle de custos é mostrado na Figura 4.12.

4.2.4.1 Entradas para o Controle dos Custos

.1 *Baseline do custo.* O *baseline* de custo é o orçamento referencial (*time-phased*) que será utilizado para medir e monitorar o desempenho dos custos do projeto. Ele é desenvolvido totalizando-se as estimativas de custo por período e, usualmente, é apresentado na forma de Curva-S.

.2 Relatórios de desempenho. Os relatórios de desempenho fornecem informações sobre o escopo do projeto e o desempenho do custo, tais como quais orçamentos estão sendo cumpridos e quais não estão. Os relatórios de desempenho também podem alertar a equipe do projeto para questões que podem causar problemas no futuro.

.3 Requisições de mudança. As requisições de mudança podem ocorrer de muitas formas - oral ou escrita, direta ou indiretamente, iniciada externa ou internamente, e legalmente imposta ou opcional. As mudanças podem exigir um aumento no orçamento ou viabilizar a sua redução.

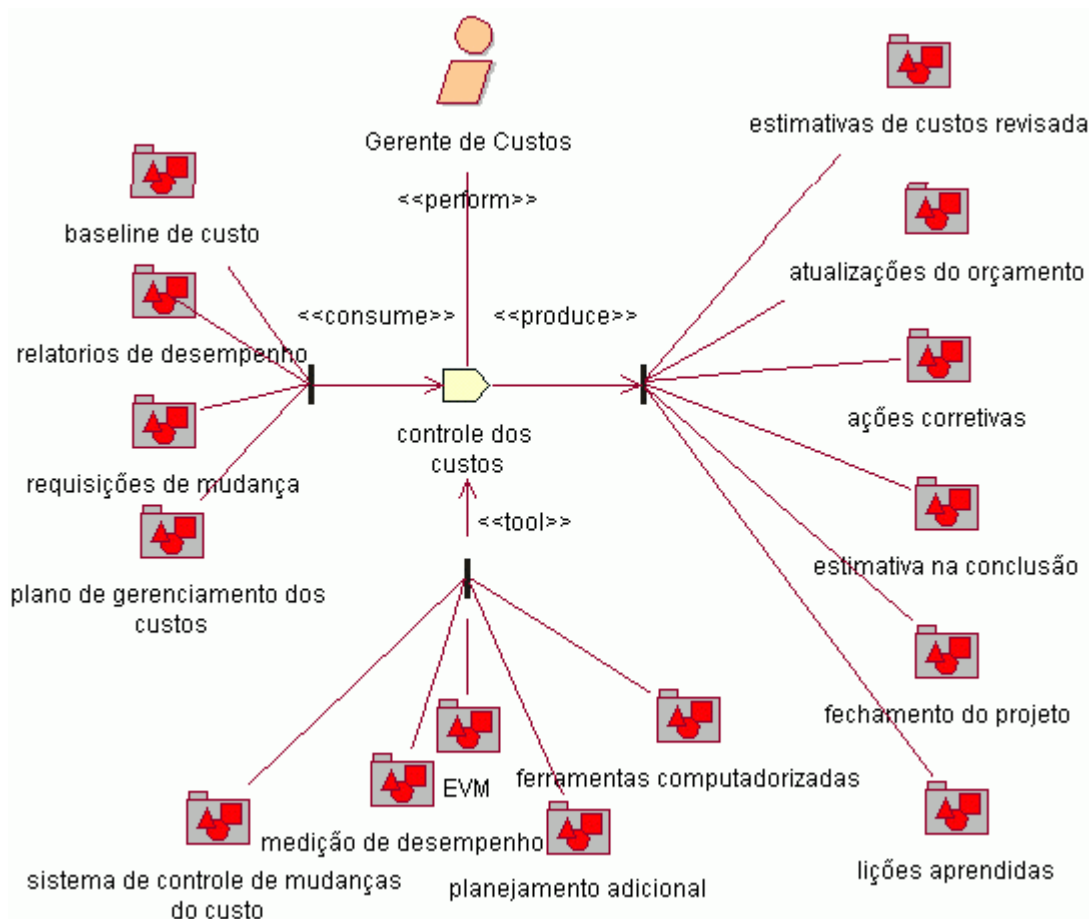


FIGURA 4.12 - Modelo de processo de controle de custos.

.4 Plano de gerenciamento de custo. O plano de gerenciamento do custo descreve como as variações no custo serão gerenciadas (por exemplo, respostas diferentes para problemas maiores ou menores). Um plano de gerenciamento de custo pode ser formal ou informal, muito detalhado ou bastante amplo, baseado nas necessidades das partes envolvidas do projeto. É um elemento componente do plano do projeto.

4.2.4.2 Ferramentas e Técnicas para o Controle dos Custos

.1 Sistema de controle de mudanças de custo. O sistema de controle de mudanças de custo define os procedimentos pelos quais o *baseline* de custo pode ser alterado. Inclui manuais, sistemas de acompanhamento e os níveis de aprovação necessários para autorizar as mudanças. O sistema de controle de mudanças de custo deve estar integrado com o sistema de controle integrado de mudanças.

.2 Medição de desempenho. As técnicas de medição de desempenho auxiliam na avaliação da magnitude de qualquer variação que ocorra. A análise do valor do trabalho realizado (*Earned Value*) é especialmente útil para o controle dos custos. Uma parte importante do controle dos custos é determinar o que está causando a variação e decidir se aquela variação requer uma ação corretiva.

.3 Gerenciamento de Valor Agregado. Todos os Planos de Controle Contábil do gerenciamento do valor agregado devem medir continuamente o desempenho do projeto através do relacionamento de três variáveis independentes: 1) O Valor Planejado [VP], que representa o trabalho físico programado para execução, incluindo o valor estimado deste trabalho (anteriormente denominado Custo Orçado do Trabalho Planejado [COTP]), comparado com 2) Valor Agregado [VA], que é o trabalho físico realmente executado, incluindo o valor estimado para esse trabalho (anteriormente denominado Custo Orçado do Trabalho Executado [COTE]), e com o 3) Custo

Real [CR], gasto para atingir o Valor Agregado. O relacionamento entre 2) Valor Agregado menos 1) Valor Planejado representa a Variação de Prazo [VP]. O relacionamento entre 2) Valor Agregado menos 3) Custo Real representa a Variação de Custo [VC] do projeto. Veja também a Seção 10.3.2.4.

.3 Planejamento adicional. Poucos projetos se desenvolvem exatamente conforme o planejado. Mudanças em perspectiva podem exigir uma estimativa nova ou uma revisão de custos ou, ainda, exigir uma análise quanto a abordagens alternativas.

.4 Ferramentas computadorizadas. As ferramentas computadorizadas, tais como softwares de gerenciamento de projeto e planilhas, são freqüentemente utilizadas para acompanhar o custo planejado *versus* o custo real, e para prever os efeitos das mudanças de custo.

4.2.4.3 Saídas do Controle dos Custos

.1 Estimativas de custo revisadas. As estimativas de custo revisadas são modificações nas informações de custo utilizadas para gerenciar o projeto. As partes envolvidas afetadas devem ser notificadas. As estimativas de custo revisadas podem requerer ou não ajustes em outros aspectos do plano geral do projeto.

.2 Atualizações do orçamento. As atualizações do orçamento são uma categoria especial das estimativas de custo revisadas. As atualizações do orçamento são mudanças no *baseline* aprovado. Esses números são normalmente revisados apenas em resposta a mudanças no escopo. Em alguns casos, as variações de custo podem ser tão severas que um replanejamento (*rebaselining*) seja necessário com a finalidade de possibilitar uma medição realística do desempenho.

.3 Ações corretivas. Uma ação corretiva é qualquer ação tomada no sentido de ajustar o desempenho futuro esperado com o plano do projeto.

.4 Estimativa para a conclusão (*Estimate at Completion*). A estimativa para a conclusão (EAC) é uma previsão do mais provável custo total do projeto baseada no desempenho do projeto e nas quantificações de risco. As técnicas mais comuns de previsão são algumas variações de:

- EAC = custo real até a data mais uma nova estimativa para todo o trabalho restante. Essa abordagem é mais freqüentemente usada quando o desempenho passado revela que as premissas da estimativa original eram bastante imperfeitas, ou que não são mais relevantes, devido a mudanças nas condições atuais. Fórmula: $EAC = CRTR + ETC$
- EAC = custo real até a data, mais o orçamento restante (BAC – COTR). Essa abordagem é mais freqüentemente usada quando as variações correntes são vistas como atípicas e a expectativa da equipe de gerenciamento do projeto é que tais variações não se repetirão no futuro. Fórmula: $EAC = CRTR + BAC - COTR$
- EAC = custo real até a data, mais o orçamento restante do projeto (BAC – COTR), modificado por um fator de desempenho, freqüentemente o índice de desempenho acumulado de custo (IPC). Essa abordagem é mais freqüentemente usada quando as variações correntes são vistas como típicas para variações futuras. Fórmula: $EAC = (CRTR + (BAC - COTR)/IPC)$, sendo este IPC o índice acumulado. Cada uma destas abordagens pode ser a abordagem correta para um dado projeto e alertará a equipe de gerenciamento sempre que as previsões de custo na conclusão do projeto (EAC) se mostrarem além das tolerâncias aceitáveis.

.5 Fechamento do projeto. Devem ser estabelecidos processos e procedimentos para o encerramento ou cancelamento dos projetos. Num exemplo americano, o Statement of Position (SOP 98-1 publicado pelo American Institute of Certified Public Accountants- AICPA) exige que, no caso de um projeto de tecnologia de informação fracassado, todos os custos sejam registrados no trimestre em que o projeto é cancelado.

.6 Lições aprendidas. As causas das variações, as razões por trás das ações corretivas tomadas e outros tipos de lições aprendidas durante o controle de custos, devem ser documentadas de forma a se tornarem parte da base de dados históricos a ser utilizada tanto no projeto corrente como em outros projetos da organização executora.

4.3 Gerenciamento dos Recursos Humanos do Projeto

O Gerenciamento dos Recursos Humanos do Projeto inclui os processos necessários para tornar mais efetivo o uso dos recursos humanos envolvidos no projeto. Isto inclui todas as partes envolvidas no projeto – patrocinadores, clientes, contribuintes individuais e outros. A Figura 4.13 fornece uma visão geral dos seguintes processos principais:

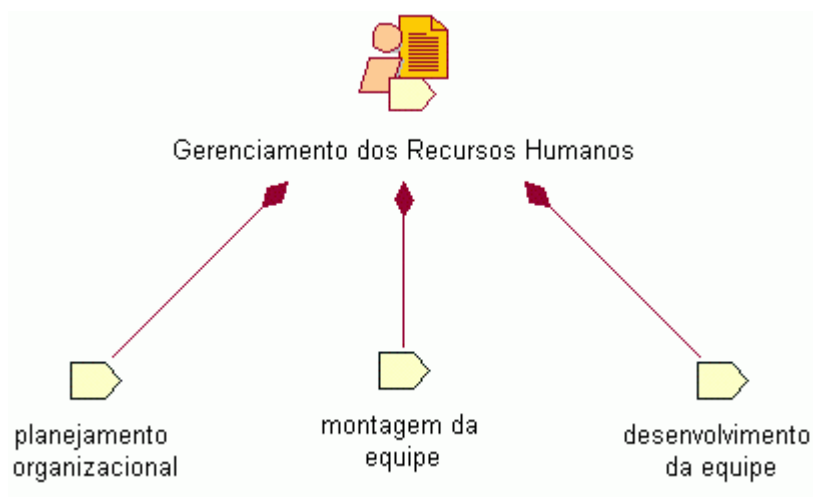


FIGURA 4.13 - Processos de recursos humanos.

- **Planejamento Organizacional** – identificar, documentar e designar os papéis, as responsabilidades e os relacionamentos de reporte dentro do projeto.
- **Montagem da Equipe** – conseguir que os recursos humanos necessários sejam designados e trabalhem no projeto.
- **Desenvolvimento da Equipe** – desenvolver competências individuais e de grupo para elevar o desempenho do projeto.

Estes processos interagem entre si e com os processos das demais áreas de conhecimento. Cada processo pode envolver esforço de um ou mais indivíduos ou grupos, dependendo das necessidades do projeto.

Embora esses processos estejam aqui apresentados como elementos discretos com interfaces bem definidas, na prática eles podem se sobrepor e interagir de outras formas não detalhadas.

O ator que participa na execução dos processos de gerenciamento dos recursos humanos é o Gerente dos Recursos Humanos, o qual será representado nos próximos diagramas pelo *RoleKind* correspondente. Neste trabalho foi este o ator definido, porém ele pode variar em outros casos ou cenários particulares, porém, ele deve ter o perfil conforme descrito a seguir.

Existe corpo de literatura substancial sobre maneiras de lidar com pessoas no contexto operacional contínuo. Alguns dos principais tópicos incluem:

- Liderança, comunicação, e negociação. Principais Habilidades da Administração Geral.
- Delegação, motivação, instrução, mentoraç o e outros assuntos relacionados ao trato com indivíduos.

- Desenvolvimento de equipe, tratamento de conflitos, e outros assuntos relacionados ao trato com grupos.
- Avaliação de desempenho, recrutamento, retenção, relações de trabalho, regulamentações de saúde e segurança e outros assuntos relacionados à administração da função de recursos humanos.

A maior parte desse material aplica-se diretamente à liderança e ao gerenciamento de recursos humanos nos projetos, e o gerente do projeto e a equipe de gerenciamento do projeto devem estar familiarizados com o assunto. Entretanto, eles, também, deveriam estar sensíveis quanto às formas de aplicação desse conhecimento no projeto. Por exemplo:

- A natureza temporária dos projetos faz com que as relações pessoais e organizacionais sejam, geralmente, temporárias e novas. A equipe de gerenciamento do projeto deve tomar cuidado para selecionar técnicas que sejam apropriadas a essas relações transitórias.
- A natureza e a quantidade de partes envolvidas no projeto, normalmente, se alterará à medida que o projeto percorre as fases do seu ciclo de vida. Portanto, as técnicas efetivas numa determinada fase podem não ser em outra. A equipe de gerenciamento do projeto deve estar atenta e utilizar as técnicas que são apropriadas para as necessidades presentes no projeto.
- As atividades administrativas de recursos humanos raramente são uma responsabilidade direta da equipe de gerenciamento. Contudo, a equipe deve estar suficientemente atenta aos requisitos administrativos para garantir conformidade.

Observação: os gerentes de projeto podem, também, ter responsabilidades pela reciclagem e liberação dos recursos humanos, dependendo da indústria e da organização à qual pertencem.

4.3.1 Planejamento Organizacional

Ao planejamento organizacional cabe identificar, documentar e designar os papéis, as responsabilidades e os relacionamentos de reporte do projeto. Os papéis, as responsabilidades e os relacionamentos de reporte podem ser atribuídos a indivíduos ou a grupos. Os indivíduos e os grupos podem fazer parte da organização do projeto ou não. Os grupos internos, normalmente, estão associados a departamentos funcionais específicos, tais como engenharia, marketing ou contabilidade.

Na maioria dos projetos, a maior parte do planejamento organizacional é feita nas fases iniciais do projeto. Entretanto, os resultados deste processo devem ser revisados, regularmente, ao longo do projeto, para assegurar a continuidade da aplicação. Se a organização inicial não está mais sendo eficiente, ela deve ser imediatamente revista.

O planejamento organizacional está, na maioria das vezes, fortemente ligado ao planejamento das comunicações, uma vez que a estrutura organizacional do projeto terá um efeito direto nos requisitos de comunicação do projeto.

O modelo do processo de planejamento organizacional é mostrado na Figura 4.14.

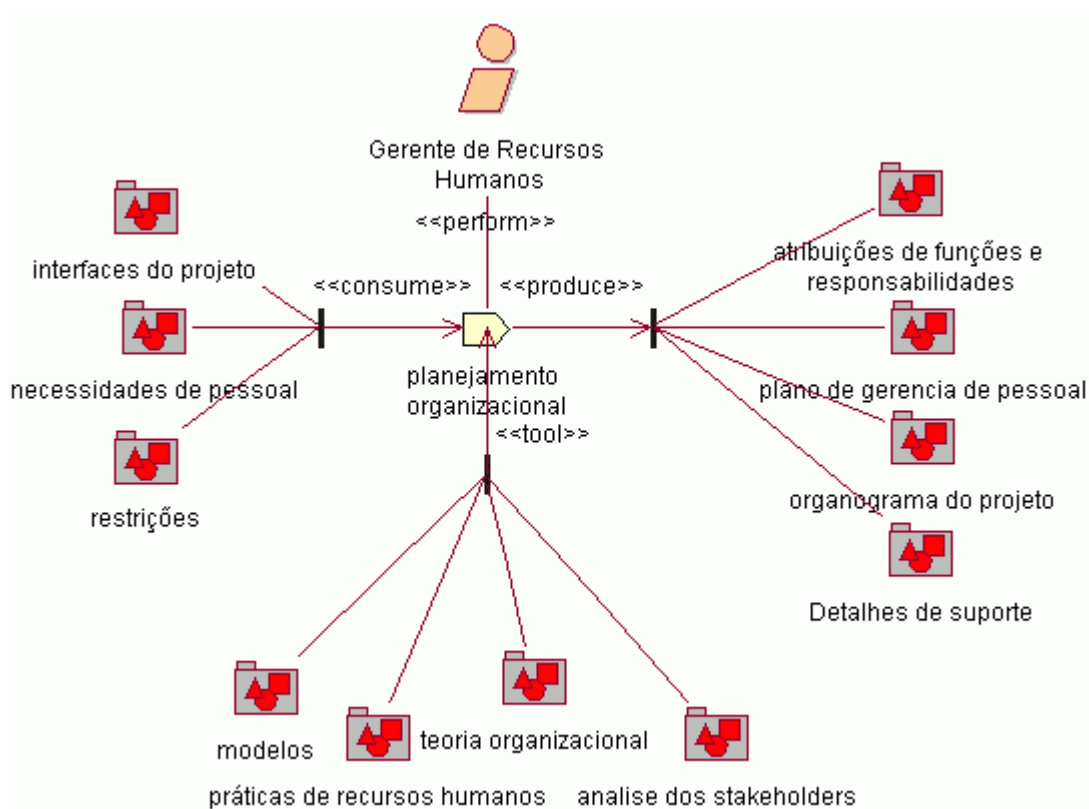


FIGURA 4.14 - Modelo de processo planejamento organizacional.

4.3.1.1 Entradas para o Planejamento Organizacional

.1 Interfaces do projeto. As interfaces do projeto geralmente estão numa das três categorias:

- Interfaces organizacionais - relacionamentos de reporte, formal ou informal, entre as diferentes unidades organizacionais. As interfaces organizacionais podem ser altamente complexas ou muito simples. Por exemplo, o desenvolvimento de um sistema complexo de telecomunicações pode exigir a coordenação de numerosos contratos durante muitos anos. Já a correção de um erro de programação, num sistema instalado num único local, pode requerer pouco mais do que uma simples notificação ao usuário e pessoal de produção, após a conclusão.

- Interfaces técnicas – relacionamentos de reporte, formal ou informal, entre as diferentes disciplinas técnicas. As interfaces técnicas ocorrem tanto dentro das fases do projeto (por exemplo, o desenho do local desenvolvido pelos engenheiros civis deve estar compatível com a superestrutura, desenvolvida pelos engenheiros estruturais) quanto entre as fases do projeto (por exemplo, quando uma equipe de projetistas automotivos passa os resultados do seu trabalho para a equipe de instrumentação que deve criar o esquema de fabricação do veículo).
- Interfaces interpessoais – relacionamentos de reporte, formal ou informal, entre os diferentes indivíduos que trabalham no projeto.

Essas interfaces ocorrem, normalmente, de forma simultânea, como, por exemplo, quando um arquiteto, empregado de uma firma de projetos, explica detalhes importantes para uma equipe de gerenciamento da firma de construção, externa à empresa.

.2 Perfil do pessoal. O perfil do pessoal define que habilidades são necessárias para cada indivíduo ou grupo e quando são necessárias. O perfil de pessoal é um subconjunto dos requisitos gerais de recursos identificados durante o planejamento dos recursos.

.3 Restrições. As restrições são fatores que limitam as opções da equipe do projeto. As opções organizacionais do projeto podem ser restringidas de muitas maneiras. Os fatores que mais podem restringir a forma de organização da equipe incluem, mas não se limitam a:

- Estrutura Organizacional da Empresa – uma organização cuja estrutura básica é uma matriz forte induz papéis relativamente mais fortes para o gerente do projeto do que uma estrutura matriz fraca.

- Acordos Coletivos – acordos com sindicatos ou outros grupos de empregados podem exigir papéis ou relacionamentos de reporte específicos (em essência, o grupo de empregados é uma parte envolvida).
- Preferências da Equipe de Gerenciamento do Projeto – os membros da equipe de gerenciamento do projeto que, no passado, tiveram sucesso com determinadas estruturas, tendem a defender estruturas similares no futuro.
- Pessoal específico alocado – a forma de organização do projeto é, na maioria das vezes, influenciada pelas habilidades e capacidades de indivíduos específicos.

4.3.1.2 Ferramentas para o Planejamento Organizacional

.1 Modelos. Embora cada projeto seja único, a maioria dos projetos será, de alguma forma, semelhante a outro projeto. Usar definições de papéis e responsabilidades, ou relacionamentos de reporte de projetos similares, pode agilizar o processo de planejamento organizacional.

.2 Práticas de recursos humanos. A maioria das organizações possui uma variedade de políticas, manuais e procedimentos que podem auxiliar a equipe de gerenciamento do projeto em vários aspectos do planejamento organizacional. Por exemplo: uma organização que enxerga os gerentes como treinadores tende a possuir documentação sobre o desempenho do papel de "treinador".

.3 Teoria organizacional. Existe um corpo de conhecimento substancial na literatura para descrever como as organizações podem e devem ser estruturadas. Embora apenas um pequeno subconjunto desse corpo na literatura seja especialmente direcionado a organizações de projetos, a equipe

de gerenciamento do projeto deve estar familiarizada com o assunto de forma a ser capaz de responder aos requisitos do projeto.

.4 Análise das partes envolvidas. A identificação das partes envolvidas e suas necessidades devem ser analisadas de forma a garantir que elas serão atendidas.

4.3.1.3 Saídas do Planejamento Organizacional

.1 Atribuição de papéis e responsabilidades. Os papéis do projeto (quem faz o quê) e as responsabilidades (quem decide o quê) devem ser associados às partes envolvidas do projeto. Papéis e responsabilidades podem variar ao longo do tempo. A maioria dos papéis e responsabilidades serão associados a partes envolvidas que estejam ativamente envolvidas no trabalho do projeto, tais como o gerente do projeto, outros membros da equipe de gerenciamento do projeto e contribuintes individuais.

Os papéis e responsabilidades do gerente do projeto são, geralmente, críticas na maioria dos projetos, mas variam significativamente dependendo da área de aplicação.

Os papéis e as responsabilidades do projeto devem estar estreitamente ligadas ao detalhamento do escopo do projeto. A Matriz de Responsabilidades é, normalmente, utilizada para esse propósito. Em grandes projetos, as MR's podem ser desenvolvidas em vários níveis. Por exemplo: uma MR de alto nível pode associar cada grupo ou unidade a um componente da EAP, enquanto uma MR de baixo nível é utilizada dentro do grupo para associar papéis e responsabilidades para as atividades específicas.

.2 Plano de gerenciamento do pessoal. O plano de gerenciamento do pessoal descreve quando e como os recursos humanos serão alocados e retirados da equipe de projeto. O plano de pessoal pode ser formal ou informal,

muito detalhado ou bastante amplo, dependendo das necessidades do projeto. Trata-se de um elemento subsidiário ao plano geral do projeto.

O plano de gerenciamento do pessoal normalmente inclui histogramas de recursos.

Atenção especial deve ser dada à liberação dos membros da equipe do projeto (indivíduos ou grupos), quando não forem mais necessários ao projeto.

Procedimentos adequados de realocação podem:

- Reduzir custos através da redução ou eliminação da tendência de "criar trabalho" para preencher o tempo entre a atual designação e a próxima.
- Levantar a moral da equipe através da redução ou eliminação da incerteza sobre futuras oportunidades de trabalho.

.3 Organograma do projeto. Um organograma é qualquer apresentação gráfica dos relacionamentos de reporte do projeto. Pode ser formal ou informal, muito detalhado ou bastante amplo, dependendo das necessidades do projeto. Por exemplo: um organograma para três ou quatro pessoas em serviços internos do projeto provavelmente não tem o mesmo rigor e detalhamento que um organograma para 3000 de uma equipe de resposta a um desastre.

Uma Estrutura Organizacional do Projeto (EOP) é um tipo específico de organograma que mostra que unidades organizacionais são responsáveis por quais pacotes de trabalho.

.4 Detalhes de suporte . Os detalhes de suporte para o planejamento organizacional variam por área de aplicação e tamanho do projeto. As informações normalmente fornecidas como detalhes de suporte incluem, mas não se limitam a:

- Impacto organizacional - que impactos estão impedidos pela adoção da organização indicada.
- Descrições do trabalho – descrição, por tarefa, das competências, responsabilidades, autoridade, ambiente físico e outras características envolvidas no exercício. Também chamadas de *descrições de cargos*.
- Necessidades de treinamento – se o pessoal a ser alocado ao projeto não possuir as competências necessárias, essas competências deverão ser desenvolvidas durante o projeto.

4.3.2 Montagem da Equipe

A montagem da equipe envolve conseguir que os recursos humanos necessários (indivíduos ou grupos) sejam alocados ao projeto. Na maioria dos ambientes, os “melhores” recursos podem não estar disponíveis, e a equipe de gerenciamento do projeto deve cuidar para garantir que os recursos que estão disponíveis irão atender os requisitos do projeto. O modelo do processo de montagem da equipe pode ser visualizado através da Figura 4.15.

4.3.2.1 Entradas Para a Montagem da Equipe

.1 Plano de gerenciamento de pessoal. O plano de gerenciamento do pessoal descreve quando e como os recursos humanos serão alocados e retirados da equipe de projeto. O plano de pessoal pode ser formal ou informal, muito detalhado ou bastante amplo, dependendo das necessidades do projeto. Trata-se de um elemento subsidiário ao plano geral do projeto.

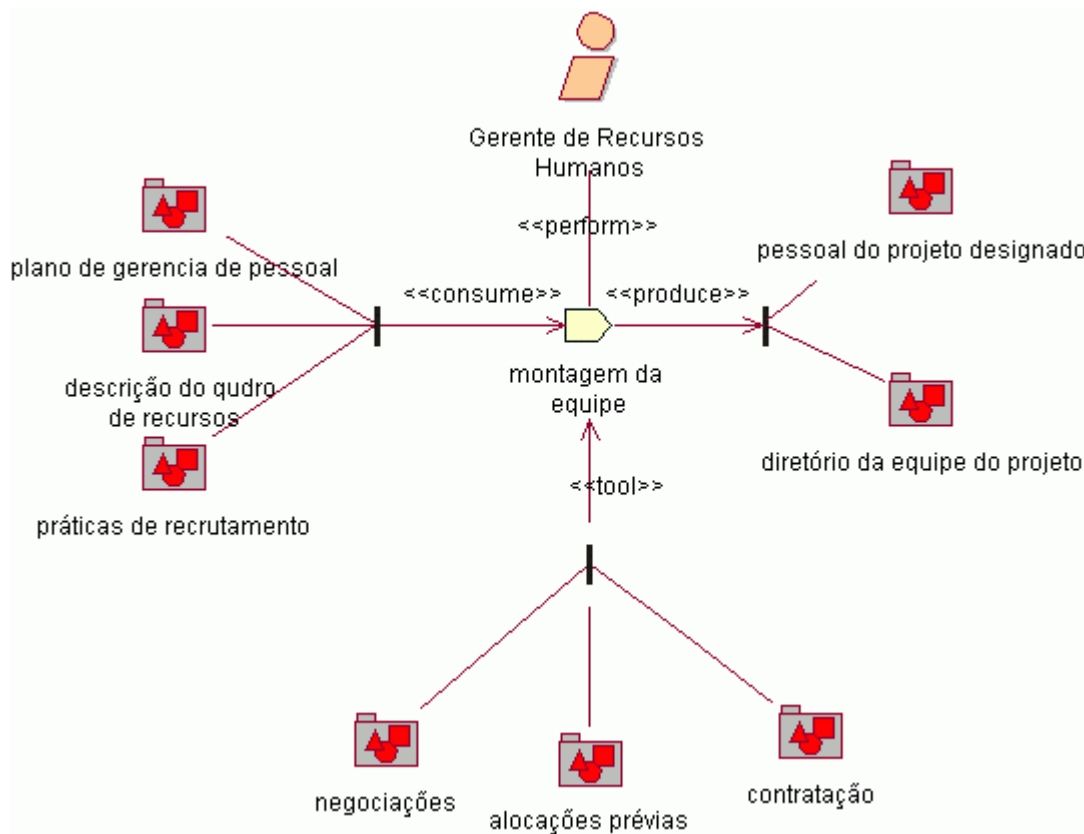


FIGURA 4.15 - Modelo de processos montagem da equipe.

.2 Descrição do quadro de pessoal. Quando a equipe de gerenciamento do projeto é capaz de influenciar ou direcionar a alocação de pessoal, ela deve considerar as características do pessoal potencialmente disponível. As considerações incluem, mas não se limitam a:

- Experiência anterior – os indivíduos ou grupos realizaram trabalhos similares anteriormente? Eles fizeram isso bem?
- Interesses pessoais – os indivíduos ou grupos estão interessados em trabalhar nesse projeto?
- Características pessoais – os indivíduos ou grupos estão aptos a trabalhar juntos como uma equipe?

- Disponibilidade – os indivíduos ou grupos mais desejáveis estarão disponíveis no momento necessário?
- Competências e perícia - que competências são necessárias e em que nível?

.3 Práticas de recrutamento. Uma ou mais de uma das organizações envolvidas no projeto podem ter políticas, manuais ou procedimentos para orientar a alocação de pessoal. Quando elas existem, tais práticas agem como uma restrição ao processo de alocação de pessoal.

4.3.2.2 Ferramentas e Técnicas para a Montagem da Equipe

.1 Negociação. As alocações de pessoal devem ser negociadas na maioria dos projetos. Por exemplo, a equipe de gerenciamento do projeto pode necessitar negociar com:

- Os gerentes funcionais responsáveis, para assegurar que o projeto receba o pessoal habilitado no momento adequado.
- Outras equipes de gerenciamento do projeto, dentro da organização executora, para alocar recursos escassos ou especializados apropriadamente. As habilidades de persuasão da equipe desempenham um papel importante na negociação das alocações de pessoal, assim como as políticas das organizações envolvidas. Por exemplo: um gerente funcional pode ser premiado de acordo com a utilização do pessoal. Isso cria um incentivo para o gerente alocar o pessoal disponível, mesmo que não satisfaça todos os requisitos do projeto.

.2 Alocações prévias. Em alguns casos, o pessoal pode ser previamente designado para o projeto. Esse é normalmente o caso quando: (a) o projeto é o resultado de uma proposta competitiva e o pessoal especificado foi prometido

como parte da proposta ou (b) o projeto se refere a um serviço interno e as alocações de pessoal foram definidas dentro do *Project Charter*.

.3 Contratação. O gerenciamento das aquisições do projeto pode ser utilizado para obter os serviços de indivíduos ou grupos específicos para realizar as atividades do projeto. A contratação é necessária quando a organização não tem o pessoal necessário no seu quadro para concluir o projeto (por exemplo, por uma decisão consciente de não contratar tais indivíduos como empregados de tempo integral, por estar com todo pessoal habilitado comprometido com outros projetos, ou por outras circunstâncias)

4.3.2.3 Saídas da Montagem da Equipe

.1 Alocação do pessoal do projeto. O projeto está com a equipe montada quando as pessoas apropriadas tiverem sido realmente alocadas para trabalhar nele. A equipe pode ser alocada em tempo integral, parcial, ou variável, dependendo das necessidades do projeto.

.2 Relação da equipe do projeto. É uma relação contendo todos os membros da equipe do projeto e outras partes envolvidas. A relação pode ser formal ou informal, muito detalhada ou geral, dependendo das necessidades do projeto.

4.3.3 Desenvolvimento da Equipe

O desenvolvimento da equipe pressupõe o aumento da capacidade das partes envolvidas de contribuir individualmente, bem como o aumento da capacidade da equipe de funcionar como uma equipe. O crescimento individual (gerencial e técnico) é a base necessária para desenvolver a equipe. O desenvolvimento como equipe é crucial para a capacidade do projeto de atingir seus objetivos.

O desenvolvimento da equipe num projeto é particularmente complicado quando os membros se reportam ao gerente funcional e ao gerente de projeto. O gerenciamento efetivo desse duplo relacionamento de reporte é,

normalmente, um fator crítico de sucesso para o projeto e, geralmente, é responsabilidade do gerente do projeto.

O modelo do processo de desenvolvimento da equipe é apresentado na Figura 4.16.

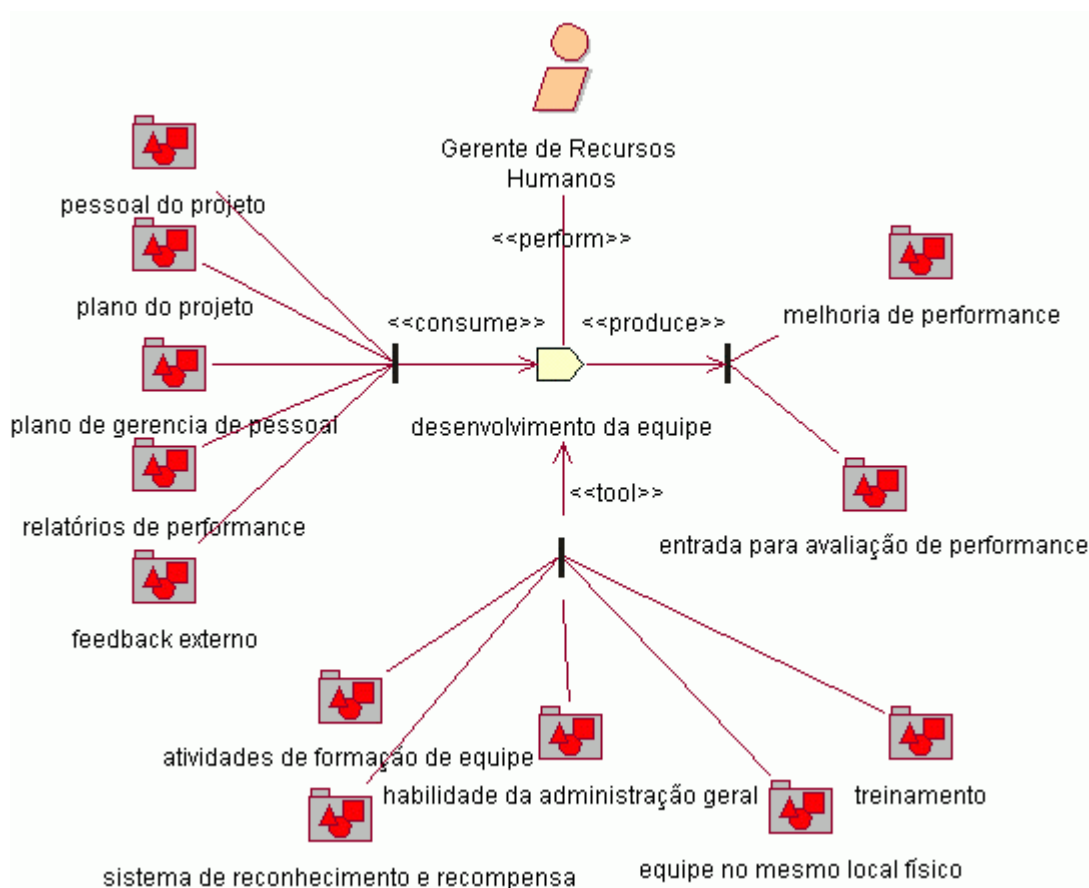


FIGURA 4.16 - Modelo do processo de desenvolvimento da equipe.

4.3.3.1 Entradas Para o Desenvolvimento da Equipe

.1 Pessoal do projeto. As alocações de pessoal definem, implicitamente, as competências individuais e de grupo com as quais se pode contar.

.2 Plano do projeto. O plano do projeto descreve o contexto técnico dentro do qual a equipe opera.

.3 Plano de gerenciamento do pessoal. O plano de gerenciamento do pessoal descreve quando e como os recursos humanos serão alocados e retirados da equipe de projeto. O plano de pessoal pode ser formal ou informal, muito detalhado ou bastante amplo, dependendo das necessidades do projeto. Trata-se de um elemento subsidiário ao plano geral do projeto .

.4 Relatórios de desempenho. Os relatórios de desempenho fornecem um *feedback* para a equipe do projeto acerca do desempenho real comparado com o plano do projeto.

.5 Feedback externo. A equipe do projeto deve periodicamente avaliar o seu próprio desempenho em relação às expectativas daqueles que estão fora do projeto.

4.3.3.2 Ferramentas e Técnicas Para o Desenvolvimento da Equipe

.1 Atividades de desenvolvimento da equipe. As atividades de desenvolvimento da equipe incluem ações gerenciais e individuais tomadas específica e principalmente para melhorar o desempenho da equipe. Muitas ações - tais como envolver membros da equipe, de nível não gerencial, no processo de planejamento ou estabelecer regras para enfrentar e lidar com conflitos - podem aumentar o desempenho da equipe como um efeito secundário. As atividades de desenvolvimento da equipe podem variar, de 5 minutos de agenda em encontros de revisões regulares de posicionamento, até experiências fora do ambiente de trabalho, facilitadas por profissionais experientes, planejadas para melhorar as relações interpessoais entre importantes partes envolvidas.

Existe um corpo de conhecimento substancial na literatura sobre desenvolvimento de equipe. A equipe de gerenciamento do projeto deve estar familiarizada com a variedade de atividades de desenvolvimento de equipes.

.2 Habilidades de administração geral. As habilidades de administração geral são de particular importância para o desenvolvimento da equipe.

.3 Sistemas de reconhecimento e recompensa. Os sistemas de reconhecimento e recompensa são ações formais de gerenciamento que promovem ou reforçam um comportamento desejado. Para serem eficientes, tais sistemas devem fazer ligação entre o desempenho e a premiação de forma clara, explícita e alcançável. Por exemplo: um gerente de projeto que será recompensado por satisfazer os objetivos de custo do projeto deve ter um nível apropriado de controle sobre o pessoal e as decisões de aquisição. Os projetos devem, na maioria das vezes, possuir os seus próprios sistemas de reconhecimento e recompensa, visto que os sistemas da organização executora podem não ser apropriados. Por exemplo: a disposição para trabalhar mais tempo para satisfazer os objetivos agressivos do cronograma deve ser premiada ou reconhecida; a necessidade de trabalhar mais, além do período normal, como resultado de um planejamento precário não deveria ser premiada.

Os sistemas de reconhecimento e recompensa também devem considerar as diferenças culturais. Por exemplo: pode ser muito difícil desenvolver um mecanismo apropriado de premiação da equipe em uma cultura que valoriza o individualismo.

.4 Colocação. A colocação envolve alojar todos, ou quase todos, os membros da equipe do projeto mais ativos no mesmo espaço físico para desenvolver sua capacidade de funcionar como uma equipe. A colocação é amplamente utilizada em grandes projetos, e também pode ser eficiente em pequenos projetos (por exemplo, com uma “sala de guerra” onde a equipe se reúne e afixa cronogramas, atualizações, etc). Em alguns projetos a colocação pode não ser uma opção; onde não for viável, uma alternativa pode ser agendar reuniões presenciais frequentes para estimular a interação.

.5 Treinamento. O treinamento inclui todas as atividades planejadas para aumentar as competências da equipe do projeto. Alguns autores distinguem treinamento, educação e desenvolvimento, mas as distinções não são consistentes nem amplamente aceitas. O treinamento pode ser formal (por exemplo, treinamento em classe, treinamento via computador) ou informal (por exemplo, *feedback* de outros membros da equipe). Existe um corpo de conhecimento substancial na literatura sobre treinamento para adultos.

Se os membros da equipe do projeto não possuem as habilidades técnicas ou gerenciais necessárias, tais habilidades devem ser desenvolvidas como parte do projeto, ou devem ser tomadas medidas para recompor a equipe do projeto apropriadamente. Os custos diretos e indiretos de treinamento são, geralmente, pagos pela organização executora.

4.3.3.3 Saídas do Desenvolvimento da Equipe

.1 Melhorias de desempenho. As melhorias de desempenho da equipe podem resultar de várias fontes e podem afetar muitas áreas de desempenho do projeto, por exemplo:

- Melhorias nas habilidades individuais podem permitir que uma pessoa específica realize as atividades que lhe foram atribuídas de forma mais efetiva.
- Melhorias no comportamento da equipe (por exemplo, enfrentamento e tratamento de conflitos) podem permitir que os membros da equipe do projeto dediquem uma maior parcela de seu esforço às atividades técnicas.
- Melhorias tanto nas competências individuais, quanto de equipe, podem facilitar a identificação e o desenvolvimento melhores formas de condução do trabalho do projeto.

.2 Entradas para avaliações de desempenho. O pessoal do projeto deve, geralmente, fornecer informações para a avaliação do desempenho de quaisquer membros da equipe com os quais interagem de forma significativa.

4.4 Organização e estrutura dos processos

A organização e estrutura dos processos de gerência de projetos propostos pelo PMI (2000), como já apresentado anteriormente, são divididos em cinco grupos: processos de iniciação, processos de planejamento, processos de execução, processos de controle e processos de encerramento. A seguir serão listados e descritos: os processos em seus respectivos grupos, as interações entre os grupos e processos e o modelo de processo de cada grupo usando a PML UPM (OMG, 2000).

Os processos se aplicam à maioria dos projetos durante a maior parte do tempo. Entretanto, nem todos os processos serão necessários, e nem todas as interações se aplicam, em todos os projetos.

A ausência de um processo não significa que ele não deve ser executado. A equipe de gerência do projeto deve identificar e gerenciar todos os processos que são necessários para assegurar o sucesso do projeto.

Os projetos que são dependentes de recursos únicos podem definir papéis e responsabilidades mesmo antes do detalhamento do escopo, uma vez que o que pode ser feito depende dos recursos disponíveis.

Algumas saídas de processos podem ser predefinidas como restrições. Grandes projetos podem necessitar de maiores detalhes.

Em subprojetos e projetos menores, gasta-se um pequeno esforço nos processos cujas saídas já tenham sido definidas ao nível do projeto ou nos processos que tenham apenas uma utilidade marginal.

Para que os processos possam ser identificados quanto a suas respectivas áreas de conhecimento, foram adicionadas duas letras antes do nome de cada processo, dessa forma, essas letras servirão para identificação das áreas de conhecimento de cada processo. As letras e respectivas áreas de conhecimento podem ser identificados a partir da Tabela 4.2.

TABELA 4.2 - Sigla das áreas de conhecimento.

Área de Conhecimento	Sigla
Processos da Gerência do Escopo	ES
Processos da Gerência do Tempo	TM
Processos da Gerência do Custo	CT
Processos da Gerência da Qualidade	QL
Processos da Gerência dos Recursos Humanos	RH
Processos da Gerência da Comunicação	CM
Processos da Gerência de Riscos	RC
Processos da Gerência de Contratos	CT
Processos da Gerência de Integração	IN

4.4.1 Processos de Iniciação

O modelo do processo de iniciação pode ser visualizado através da Figura 4.17.

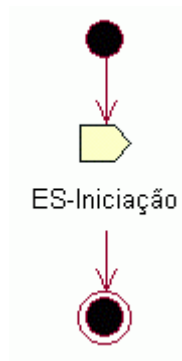


FIGURA 4.17 - Modelo do grupo de processo iniciação.

- Iniciação – obter o comprometimento da organização para o início da próxima fase do projeto.

4.4.2 Processos de Planejamento

O planejamento é de fundamental importância num projeto, porque executar um projeto implica em realizar algo que não tinha sido feito antes. Como consequência, existem relativamente mais processos nessa seção. Entretanto, o número de processos não significa que a gerência de projetos é principalmente planejamento – a quantidade de planejamento deve estar de acordo com o escopo do projeto e com a utilidade da informação desenvolvida. Planejar é um esforço contínuo durante toda a vida do projeto.

Estes processos estão sujeitos a freqüentes interações antes da complementação do plano. Por exemplo, se a data inicialmente prevista para o término for inaceitável, os recursos do projeto, o custo, ou mesmo o escopo podem necessitar de redefinição. Além disto, o planejamento não é uma ciência exata – duas equipes distintas podem gerar planos muito diferentes para o mesmo projeto.

Processos principais - Alguns dos processos de planejamento têm dependências bem definidas, que fazem com que eles sejam executados essencialmente na mesma ordem, na maioria dos projetos. Por exemplo, as

atividades devem ser definidas antes do estabelecimento do seu cronograma e custo. Estes processos principais de planejamento podem interagir várias vezes durante qualquer fase de um projeto. Eles incluem:

- Planejamento do Escopo – desenvolver uma declaração escrita do escopo, como base para futuras decisões no projeto.
- Detalhamento do escopo – subdividir os principais subprodutos do projeto em componentes menores e mais manuseáveis.
- Definição das Atividades – identificar as atividades específicas que devem ser realizadas para produzir os diversos subprodutos do projeto.
- Seqüenciamento das Atividades – identificar e documentar as dependências entre as atividades.
- Estimativa da Duração das Atividades – estimar o número de períodos de trabalho (prazos) que serão necessários para completar as atividades individuais.
- Desenvolvimento do Cronograma – criar o cronograma do projeto a partir da análise da seqüência das atividades, suas durações, e as necessidades de recursos.
- Planejamento da Gerência de Risco – decidir como abordar e planejar a gerência de risco no projeto.
- Planejamento dos Recursos – determinar que recursos (pessoas, equipamentos, materiais, etc.) devem ser utilizados, e em que quantidades, para a realização das atividades do projeto.

- Estimativa dos Custos – desenvolver uma aproximação (estimativa) dos custos dos recursos que são necessários para completar as atividades do projeto.
- Orçamento dos Custos – alocar a estimativa dos custos globais aos pacotes individuais de trabalho.
- Desenvolvimento do Plano do Projeto – agregar os resultados dos outros processos de planejamento construindo um documento coerente e consistente.

O modelo dos processos principais de planejamento podem ser visualizados através da Figura 4.18.

Processos facilitadores. As interações entre os demais processos de planejamento são mais dependentes da natureza do projeto. Por exemplo, em alguns projetos pode haver sido identificado apenas um pequeno risco ou mesmo nenhum, até que a maioria do planejamento tenha sido concluído e a equipe reconheça que as metas de custo e prazo são por demais ousadas, envolvendo assim um risco considerável. Ainda que estes processos facilitadores sejam realizados intermitentemente, e à medida que são necessários, durante o planejamento do projeto, eles não são opcionais. Eles incluem:

- Planejamento da Qualidade – identificar os padrões de qualidade relevantes para o projeto e determinar como satisfazê-los.
- Planejamento Organizacional – identificar, documentar, e atribuir papéis, responsabilidades e relações hierárquicas no projeto.
- Montagem da Equipe – conseguir que os recursos humanos necessários sejam designados e alocados ao projeto.

- Planejamento das Comunicações – determinar as necessidades das partes envolvidas quanto à informação e comunicação: quem necessita de qual informação, quando necessita e como a informação será fornecida.

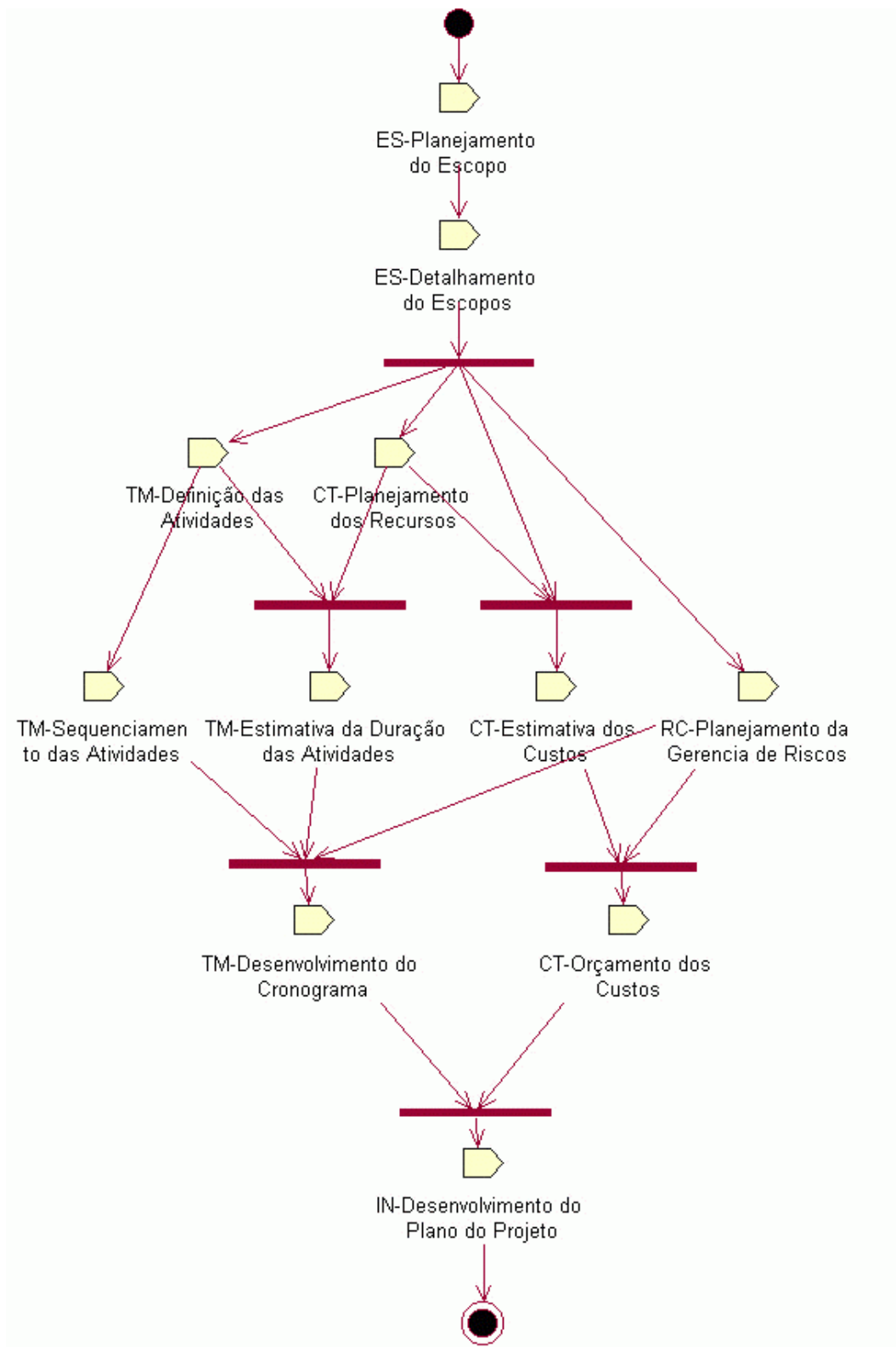


FIGURA 4.18 - Modelo dos processos principais do grupo de planejamento.

- Identificação dos Riscos – determinar os riscos prováveis do projeto e documentar as características de cada um.
- Análise Qualitativa dos Riscos – analisar qualitativamente os riscos e condições para priorizar seus efeitos nos objetivos do projeto.
- Análise Quantitativa dos Riscos – mensurar a probabilidade e impacto dos riscos e estimar suas implicações nos objetivos do projeto.
- Planejamento de Resposta a Riscos – desenvolver procedimentos e técnicas para aumentar as oportunidades e para reduzir ameaças de riscos para os objetivos do projeto.
- Planejamento das Aquisições – determinar “o quê” contratar e “quando”.
- Preparação das Aquisições – documentar os requisitos do produto/serviço a ser adquirido e as fontes possíveis de fornecimento.

O modelo dos processos facilitadores de planejamento podem ser visualizados através da Figura 4.19.

4.4.3 Processos de Execução

Os processos de execução incluem os processos principais e os facilitadores. A Figura 4.20 mostra o modelo dos processos principais e a Figura 4.21 mostra o modelo dos processos facilitadores.

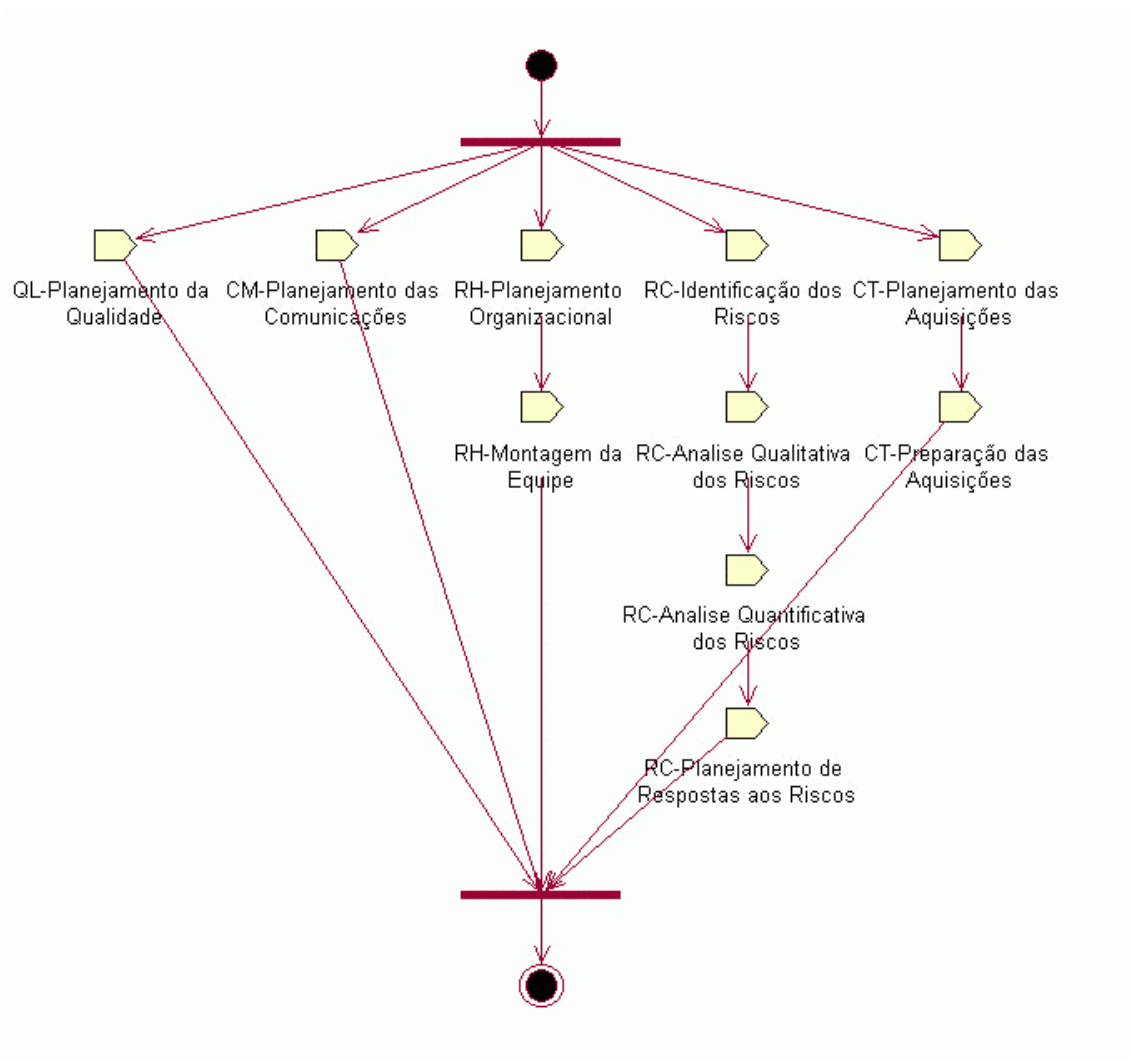


FIGURA 4.19 - Modelo dos processos facilitadores do grupo de planejamento.

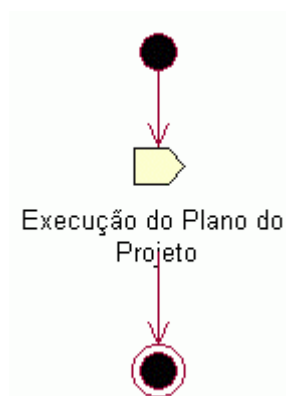


FIGURA 4.20 - Modelo dos processos principais do grupo de execução.

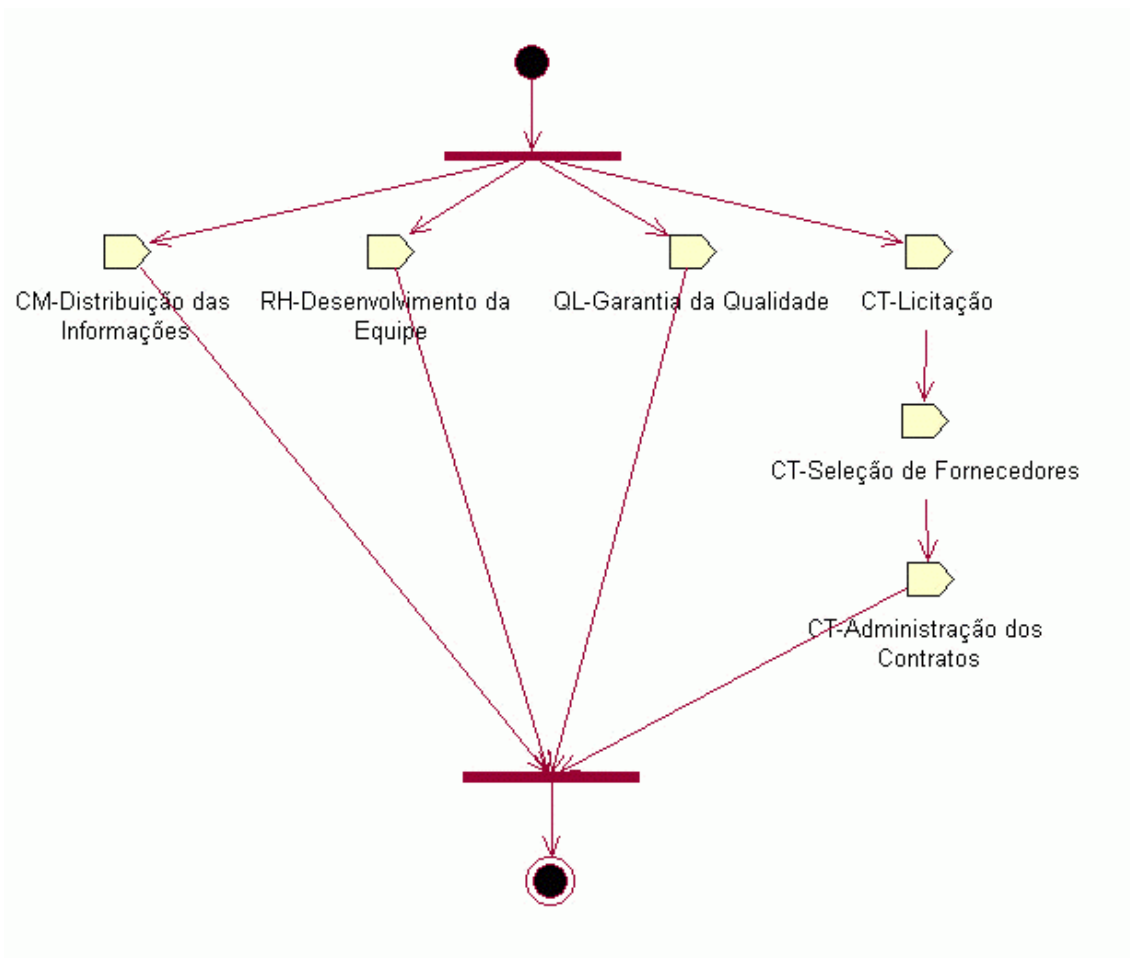


FIGURA 4.21 - Modelo dos processos facilitadores do grupo de execução.

- Execução do Plano do Projeto – levar a cabo o plano do projeto através da realização das atividades nele incluídas.
- Garantia da Qualidade – avaliar regularmente o desempenho geral do projeto, com o objetivo de prover confiança de que o projeto irá satisfazer os padrões estabelecidos de qualidade.
- Desenvolvimento da Equipe – desenvolver habilidades das pessoas e da equipe, enquanto grupo, para melhorar o desempenho do projeto.
- Distribuição das Informações – disponibilizar as informações necessárias, e no momento adequado, às partes envolvidas.

- Pedido de propostas – obter, conforme apropriado em cada caso (cotações de preço, cartas-convite, licitações, concorrências), as propostas de fornecimento dos produtos e/ou serviços pretendidos.
- Seleção de Fornecedores – escolher entre os possíveis fornecedores.
- Administração dos Contratos – gerenciar os relacionamentos com os fornecedores.

4.4.4 Processos de Controle

O desempenho do projeto deve ser monitorado e medido regularmente para identificar as variações do plano. Estes desvios são analisados, dentro dos processos de controle, nas diversas áreas de conhecimento. Na medida em que são identificados desvios significativos (aqueles que colocam em risco os objetivos do projeto), realizam-se ajustes ao plano através da repetição dos processos de planejamento que sejam adequados àquele caso. Por exemplo, ultrapassar a data de término de uma atividade pode requerer ajustes nos recursos humanos, na necessidade ou não de horas-extras, ou no balanceamento entre o orçamento e os objetivos de prazo do projeto. Controlar também inclui tomar ações corretivas, antecipando-se aos problemas.

Os grupos de processos de controle, também, incluem processos principais e facilitadores. A Figura 4.22 mostra o modelo dos processos principais e a Figura 4.23 mostra o modelo dos processos facilitadores.

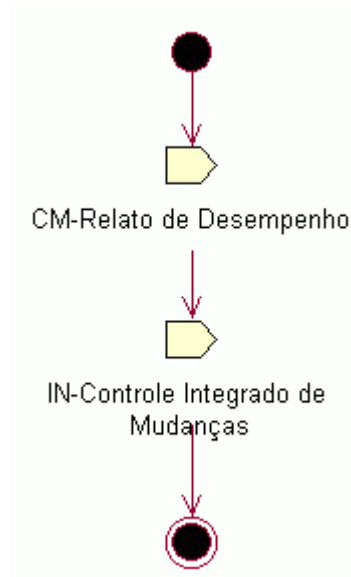


FIGURA 4.22 - Modelo dos processos principais do grupo de controle.

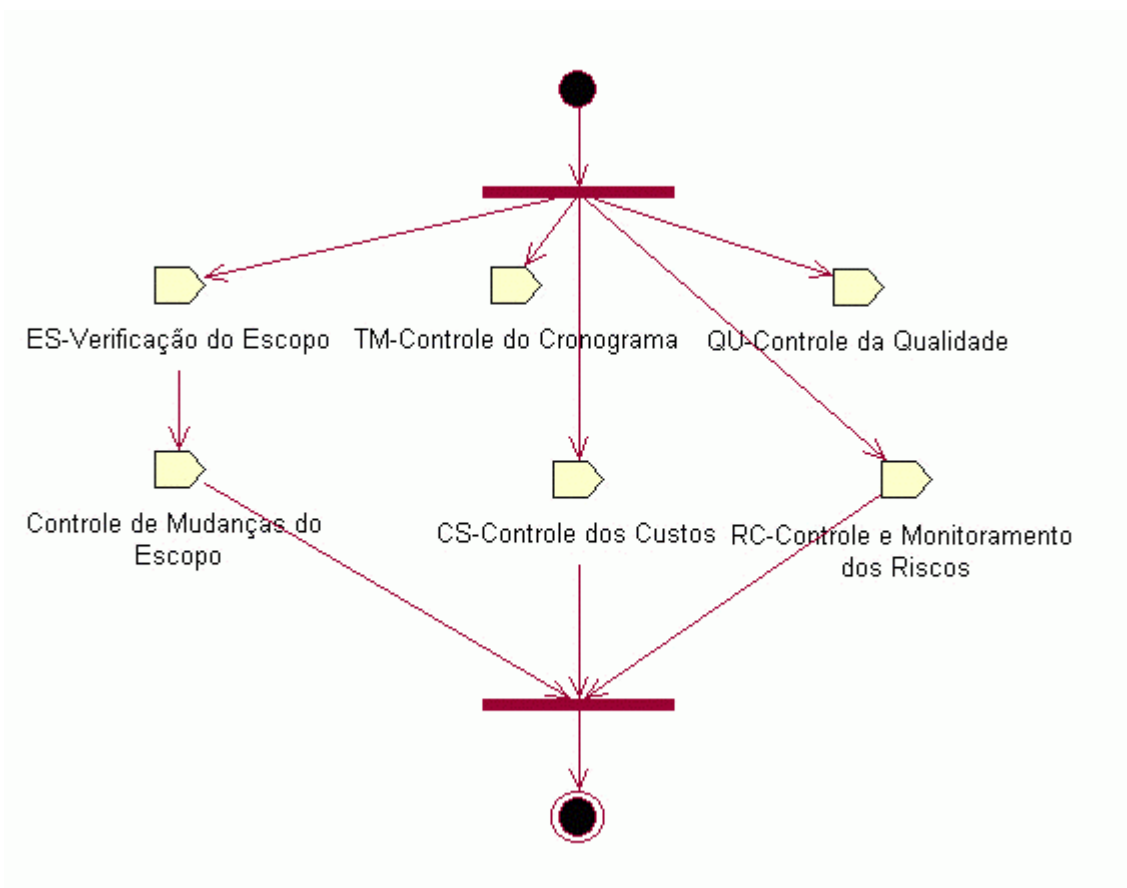


FIGURA 4.23 - Modelo dos processos facilitadores do grupo de controle.

- Controle Integrado de Mudanças – coordenar as mudanças através de todo o projeto.
- Verificação do Escopo – aceitar formalmente os resultados (escopo) do projeto.
- Controle de Mudanças do Escopo – controlar as mudanças no escopo do projeto.
- Controle do Cronograma – controlar as mudanças no cronograma do projeto.
- Controle dos Custos – controlar as mudanças no orçamento do projeto.
- Controle da Qualidade – monitorar resultados específicos do projeto para determinar se eles atingem padrões adequados de qualidade, e identificar ações para eliminar as causas de desempenhos insatisfatórios.
- Relato de Desempenho – coletar e divulgar informações de desempenho. Isto inclui relatórios de status, medidas de progresso, e novas estimativas do projeto.
- Controle e Monitoração de Riscos – acompanhar os riscos identificados, monitorar os riscos residuais e identificar novos riscos, garantindo a execução dos planos de risco e avaliar sua efetividade em reduzir riscos.

4.4.5 Processos de Encerramento

O modelo do processo de encerramento pode ser visualizado através da Figura 4.24.

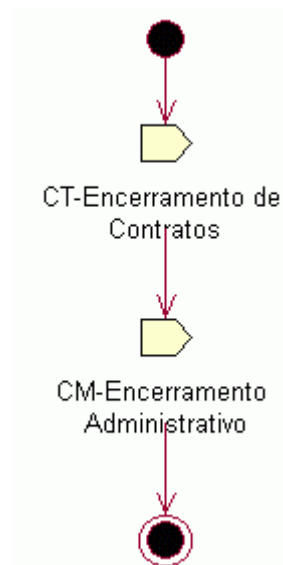


FIGURA 4.24 - Modelo dos processos do grupo de encerramento.

- Encerramento dos Contratos – completar e liquidar o contrato, incluindo a resolução de qualquer item pendente.
- Encerramento Administrativo – gerar, reunir e disseminar informações para formalizar o término da fase ou projeto, incluindo avaliações do projeto e compilação das lições aprendidas para uso em planejamentos de futuros projetos ou fases.

CAPÍTULO 5

ARQUITETURA DE UMA APLICAÇÃO PARA SUPORTE E AUTOMAÇÃO DE PROCESSOS DE GERÊNCIA DE PROJETOS

5.1 Histórico

A proposta inicial desse trabalho era criar um mecanismo que apoiasse a estimativa de tempos de tarefas baseada numa base histórica de conhecimentos de projetos, já que os erros nas estimativas são grandes responsáveis pelo fracasso de projetos, como mostrado em sessões anteriores.

Partindo desse princípio, iniciaram-se os trabalhos de pesquisa para desenvolvimento de um protótipo que implementasse um modelo histórico de informações de projetos e um mecanismo de previsão de tempo.

Durante as pesquisas, pôde-se concluir que seria necessário mais do que uma simples base histórica e um mecanismo de previsão, nessa oportunidade, decidiu-se ampliar o foco da pesquisa e conseqüentemente o trabalho de pesquisa.

Foram analisadas as propostas de alguns processos das principais abordagens aceitas e praticadas pela comunidade de engenharia de software e de gerenciamento de projeto.

Com base nessa análise, tornou-se necessário a modelagem desses processos, nessa oportunidade, foram avaliadas algumas PMLs, e decidiu-se estudar a extensão da UML para a modelagem de processos. Com isso, foi possível a publicação de dois trabalhos com propostas de extensão e modelagem de processos de gerenciamento de projetos. Tais trabalhos foram

publicados em oportunidades diferentes e cada um refletia o estado atual do trabalho de pesquisa.

O primeiro deles foi: “Uma Abordagem para Processos de Gerenciamento do Tempo e dos Recursos Humanos Apoiados por Ambientes Integrados de Engenharia de Software” (Borrego Filho et al., 2002b), publicado nos anais do SIMPROS 2002 - IV Simpósio Internacional de Melhoria de Processos de Software, ocorrido em Recife-PE em Setembro de 2002. O outro foi: “Análise e Modelagem para Automação de Processo de Gerenciamento de Projetos” (Borrego Filho et al., 2002a), publicado nos anais do II WORCAP - Workshop de Cursos de Computação Aplicada do INPE, ocorrido em São José dos Campos, em novembro de 2002.

Hoje, depois da descoberta da UPM, PML OO com notação UML, os modelos de processos estão sendo modelados através dessa PML, já que esta é uma notação desenvolvida pelo OMG.

Desse ponto em diante, as pesquisas se concentravam em encontrar soluções para a construção de uma arquitetura robusta para apoio aos processos de gerenciamento de projetos. Logo no início dessa pesquisa, pôde-se observar que o tema era bastante abrangente, por isso, a pesquisa ficou restrita aos processos que nesse trabalho foram denominados **essenciais**.

Essenciais, porque a sistematização ou automatização dos outros processos se torna inviável, visto que os demais são dependentes ou até mesmo não podem ser executados sem ter informações desses processos essenciais.

Por esse motivo, foram pesquisadas abordagens para processos de gerenciamento de projetos e abordagens e modelos de maturidade para o desenvolvimento de software.

Em vista disso, tomou-se como base o modelo de processos de gerenciamento de projetos proposto pelo PMI (2000), que é amplamente aceito e aplicado

tanto pela Comunidade de Engenharia de Software como pelas outras comunidades que fazem uso do gerenciamento de projetos. Foram também considerados os principais modelos e abordagens de processos de engenharia de software como o CMMI e o SPICE/ISO-15504, uma vez que os mesmos têm características em comum, como mostram as tabelas de compatibilidade apresentadas na sessão anterior.

Nessa oportunidade, também foi possível a publicação de um outro artigo a respeito da definição da arquitetura. O trabalho: “Uma Aplicação para Automatização dos Processos de Gerenciamento de Projetos Suportada por Ambiente de Engenharia de Software” (Borrego Filho et al., 2003a) foi publicado no CITS’2003 Qualidade de Software - XIV Congresso Internacional de Tecnologia de Software, ocorrido em Curitiba-PR em junho de 2003.

Com o avanço das pesquisas e a consolidação dos conceitos propostos, foi possível a publicação do trabalho: “Proposta de Arquitetura para Apoio, Suporte e Automação de Processos de Gerenciamento de Projetos” (Borrego Filho et al., 2003b) publicado nos anais do III Worcap - Workshop dos Cursos de Computação Aplicada do INPE, realizado em São José dos Campos, em novembro de 2003.

Ao longo de todo o período de pesquisa, todos os conceitos e tecnologias analisadas, os quais já foram descritos em sessões anteriores, fizeram parte das pesquisas para o desenvolvimento de um protótipo que fosse capaz de apoiar e automatizar alguns dos processos de gerenciamento de projetos. Essa prática trouxe um grande retorno em termos de análise de viabilidade e um retorno efetivo das necessidades reais de um ambiente de aplicação.

A implementação de protótipos foi de fundamental importância para a análise e modelagem da arquitetura que está sendo proposta nesse trabalho e que será descrita nas sessões seguintes.

5.2 Análise dos Requisitos

Foram analisadas as propostas de alguns processos das principais abordagens aceitas e praticadas pela comunidade de engenharia de software e de gerenciamento de projeto. Na Tabela 5.1, é apresentada uma relação das abordagens que foram analisadas com os respectivos processos de interesse para esse trabalho.

TABELA 5.1 - Abordagens e processos analisados.

Abordagem	Processos
CMM (Paulk et al., 1993)	Áreas chave de processos que tratam do planejamento, rastreamento e acompanhamento do projeto, gerenciamento quantitativo do processo e processo organizacional.
CMMI (CMMI, 2001)	Práticas chave de processos que tratam do planejamento, monitoramento e controle de projetos, medição e análise e processo organizacional.
SPICE - ISO/IEC 15.504 (SPICE, 1995)	Processos de Gerência, Gerência de Projeto, Alinhamento Organizacional, Gerência dos Recursos Humanos e Medição.
ISO/IEC 12207 (ISO/IEC, 1995)	Processos de gerenciamento de projetos.
PMBOK 2000 (PMI, 2000)	Áreas de conhecimento: Gerência do Tempo, Gerência dos Recursos Humanos e Gerência dos Custos.
<i>Personal Software Process</i> - PSP (Hayes; Over, 1997)	Processo de gerenciamento pessoal, apontamento de tempos e estimativas de tempo.

(Continua)

TABELA 5.1 – Conclusão.

<i>Team Software Process</i> - TSP (Humphrey, 2000)	Gerenciamento, acompanhamento e definição de equipes, processo de gerência e estimativas de tempo e custo.
<i>Unified Software Development Process</i> (Jacobson et al., 1999)	Definição e modelagem de processos, <i>workflow</i> de processos e processos de gerenciamento.
<i>Rational Unified Process</i> - RUP (Rational, 2000)	Definição e modelagem de processos, <i>workflow</i> de processos e processos de gerenciamento.
ISO 10006 (ISO, 1997) (ABNT, 2000a)	Processos de gerenciamento de projetos.

Nesse estudo foram levantados e analisados os requisitos necessários à definição, implementação, apoio e automação desses processos.

Para que se possa visualizar de uma forma global os requisitos, foi criado o diagrama de contexto da UML (Booch et al., 2000), conforme Figura 5.1.

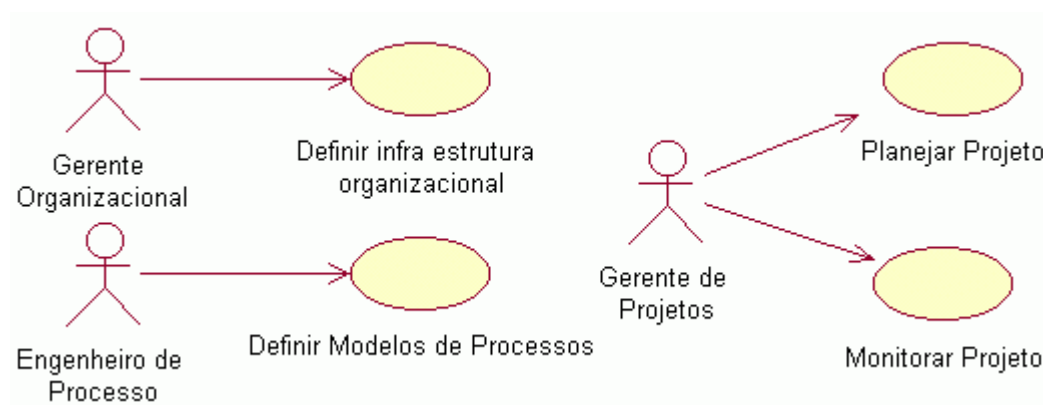


FIGURA 5.1 - Diagrama de contexto.

Os atores definidos no diagrama apresentado anteriormente, bem como nos próximos, podem variar de caso a caso, dependendo do cenário de aplicação,

ou mesmo, dos recursos disponíveis. Esse caso pode ser observado na sessão anterior, onde foi definido um ator para cada processo. Também será o caso do exemplo de aplicação apresentado no apêndice, e neste caso a definição dos atores e respectivas responsabilidades são definidas dinamicamente através do serviço de controle de acesso, o qual será apresentado oportunamente.

A seguir serão descritos cada um dos componentes do diagrama de contexto apresentado anteriormente.

5.2.1 Definir Infra-Estrutura Organizacional

Os principais objetivos dessa funcionalidade é permitir a manutenção das informações organizacionais no que tange à infra-estrutura organizacional e aos recursos. Um melhor detalhamento dessa funcionalidade pode ser observada no Diagrama de Casos de Uso (Booch et al., 2000), mostrado na Figura 5.2.

A seguir será feita uma descrição dos principais objetivos de cada Caso de Uso.

5.2.1.1 Caso de Uso Definir Pessoas

Neste caso de uso poderão ser definidos os recursos humanos envolvidos com a organização, suas alocações em departamentos, áreas, sessões, etc., bem como todas as informações necessárias e cabíveis às pessoas, como: documentos, currículos, dados pessoais, dados trabalhistas, etc.

5.2.1.2 Caso de Uso Definir Equipes

Neste caso de uso poderão ser definidas as equipes ou times. As equipes podem ser organizadas: pelas características ou perfis das pessoas integrantes da equipe; por projeto; por departamento; por especialidade; etc.

5.2.1.3 Caso de Uso Definir Recursos de Infra-Estrutura

Neste caso de uso poderão ser definidos os recursos de infra-estrutura dispostos por uma organização. Esses recursos variam de acordo com a área de atuação de cada organização, mas no caso das organizações que atuam na área de desenvolvimento de software, esses recursos podem ser: salas de reunião, baias de trabalho, computadores, mesas, projetores, softwares etc.

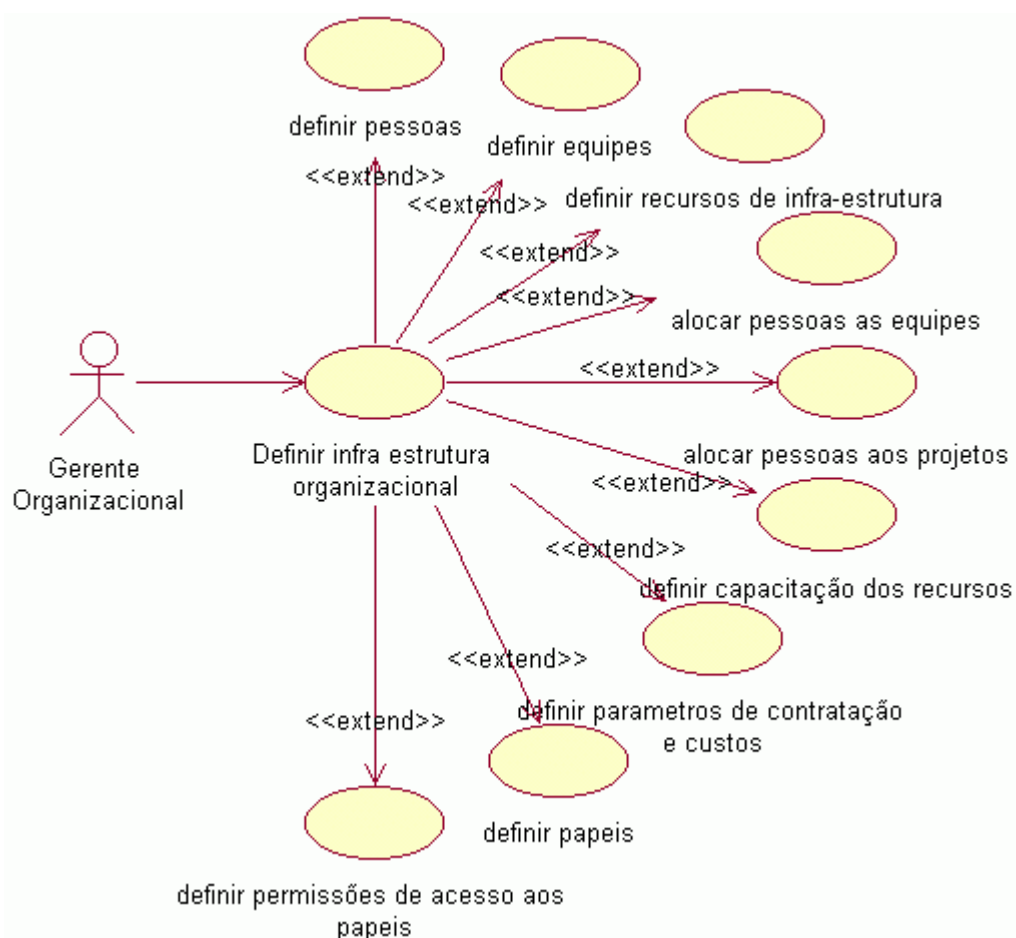


FIGURA 5.2 - Diagrama de caso de uso da definição de infra-estrutura organizacional.

5.2.1.4 Caso de Uso Alocar Pessoas às Equipes

Caso de uso responsável pela alocação das pessoas nas equipes, as quais já foram definidas por outro caso de uso. Esse caso de uso também fica responsável pelo histórico de alocações ao longo do tempo.

5.2.1.5 Caso de Uso Alocar Pessoas aos Projetos

Caso de uso responsável pela alocação das pessoas aos projetos, dessa forma poderão ser designados departamentos, sessões, equipes e respectivas pessoas a participarem de um projeto. Vale salientar que uma mesma pessoa pode participar de diversas equipes e projetos simultaneamente.

5.2.1.6 Caso de Uso Definir Capacitação dos Recursos

Este caso de uso é responsável por definir os perfis de cada pessoa, ou seja, relacionar todas as capacitações ou responsabilidades que uma determinada pessoa pode assumir durante um projeto. Esse relacionamento é feito com a associação de papéis (Booch et al., 2000) às pessoas.

5.2.1.7 Caso de Uso Definir Parâmetros de Contratação e Custos

Neste caso de uso poderão ser definidas as diversas formas de contratação e de custo de recursos humanos. Os tipos de contratação podem ser: por hora, por tarefa, por projeto, por objetivo/meta, prêmios, etc.

5.2.1.8 Caso de Uso Definir Papel

Este caso de uso define quais papéis são necessários para o tipo de atividade de uma organização. No caso das organizações de desenvolvimento de software, as abordagens de processos de desenvolvimento já trazem em suas especificações as necessidades de papéis.

5.2.1.9 Caso de Uso Definir Permissões de Acesso aos Papéis

Este caso de uso define quais os níveis de acesso são permitidos para cada papel, assim quando um recurso humano estiver desempenhando um papel para executar uma tarefa, ou mesmo, quando esse recurso tiver seu perfil definido na alocação do projeto, o mesmo poderá usufruir apenas das permissões definidas para o papel que ele nesse momento exerce.

5.2.2 Definir Modelos de Processos

Essa funcionalidade atende ao requisito imposto pelo diagrama de contexto, conforme apresentado pela Figura 5.3.

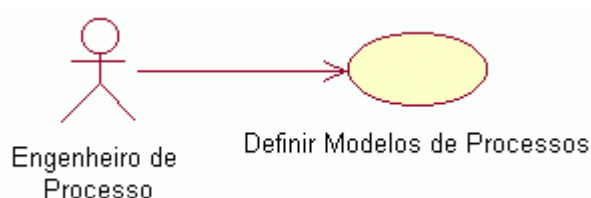


FIGURA 5.3 - Caso de uso definir modelos de processos.

Resumidamente essa funcionalidade deverá proporcionar a modelagem e definição dos processos que são utilizados por uma organização. Essa ferramenta utiliza uma PML específica, a qual é usada para a representação gráfica da composição e dos elementos de um processo.

Essa ferramenta pode prover outros tipos de serviços como simulação do processo, automatização do processo, monitoração do processo, análise do processo, extração de métricas do processo, dentre outros. Essa ferramenta será melhor abordada em outro trabalho relacionado a esse, através de outra dissertação também do grupo de engenharia de software da CAP.

5.2.3 Planejar Projeto

Essa funcionalidade atende ao requisito imposto pelo caso de uso Planejar Projeto, mostrado anteriormente no diagrama de contexto. Com base naquele diagrama foi gerado o Diagrama de Casos de Uso, a fim de demonstrar as funcionalidades, como pode ser visto na Figura 5.4.

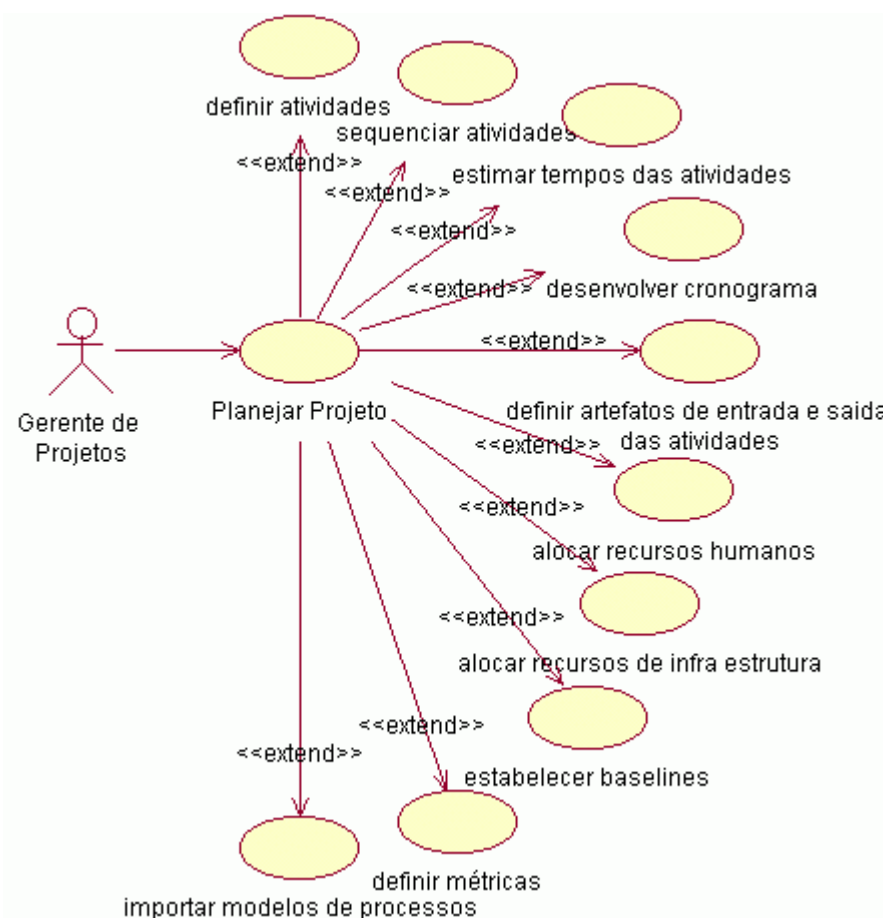


FIGURA 5.4 - Diagrama de casos de uso planejar projeto.

5.2.3.1 Caso de Uso Definir Atividades

Esse caso de uso dará subsídios à criação e definição de atividades, ou tarefas. Através desse caso de uso o gerente de projetos poderá definir a descrição, instrução, datas de início e fim e demais informações pertinentes às tarefas.

5.2.3.2 Caso de Uso Seqüenciar Atividades

Através desse caso de uso será possível estabelecer o seqüenciamento das atividades. Esse seqüenciamento pode ser especificado por *constraints* e por tipo como *finish to start* - FS, *finish to finish* - FF, *start to finish* - SF ou *start to start* - SS (Wysocki et al., 1995).

5.2.3.3 Caso de Uso Estimar Tempos das Atividades

Através desse caso de uso deverá ser possível o uso de métodos ou ferramentas para a previsão de tempo. Existem diversos tipos de métodos e ferramentas disponíveis para esse tipo de previsão, como foi mostrado em sessões anteriores.

5.2.3.4 Caso de Uso Desenvolver Cronograma

Como mostrado em sessões anteriores, existem alguns métodos de criação e desenvolvimento de cronograma, esse caso de uso deverá automatizar a criação de cronograma, baseando-se nesses diversos métodos e técnicas.

5.2.3.5 Caso de Uso Definir Artefatos de Entrada e Saída das Atividades

Esse caso de uso permitirá a definição dos artefatos de entradas e saída para cada atividade, uma vez que as atividades necessitam de artefatos (Booch et al. 2000) de entrada para que os mesmos possam ser usados/processados para obtenção do objetivo de uma tarefa, que é a geração de artefatos de saída. Através desse tipo de controle é possível o monitoramento do fluxo das informações e dos produtos (artefatos) durante o desenvolvimento de um projeto.

Esse tipo de suporte permitirá posteriormente a integração dos processos de garantia da qualidade e dos processos de controle de configuração e versão, por exemplo.

5.2.3.6 Caso de Uso Alocar Recursos Humanos

Nesse caso de uso será possível a alocação dos recursos humanos ao projeto, definindo dessa forma quais pessoas, equipes, departamentos, etc., estarão disponíveis para a execução do projeto.

5.2.3.7 Caso de Uso Alocar Recursos de Infra-Estrutura

Nesse caso de uso será possível a alocação dos recursos de infra-estrutura, que se torna necessário quando a organização dispõe de recursos específicos e que são compartilhados entre os projetos, dessa forma, se faz necessário o gerenciamento do compartilhamento e uso desses recursos.

5.2.3.8 Caso de Uso Estabelecer *Baselines*

Esse caso de uso permite a definição de *baselines* de qualidade, custos e prazos, por exemplo. A principal função da definição de *baselines* é a criação de parâmetros para o acompanhamento do desenvolvimento do projeto.

5.2.3.9 Caso de Uso Importar Modelos de Processos

A importação dos modelos de processos é um grande diferencial dessa proposta, uma vez que esse tipo de funcionalidade não está disponível atualmente nas ferramentas comerciais. O principal objetivo desse caso de uso é permitir o reuso da definição de processos, de maneira que o usuário possa “montar” o projeto de acordo com necessidades específicas e particulares de cada projeto, através da importação de modelos de processos definidos.

Os processos seriam como peças de um jogo de Tangram e as possibilidades de figuras a serem formadas seriam os projetos, ou seja, de um mesmo conjunto de modelos de processos (peças) pode-se desenvolver um grande número de projetos (figuras) bastando apenas importar e seqüenciar (jogar) os

processos definidos de acordo com as necessidades de um dado projeto. Num jogo de Tangram com 7 peças é possível criar cerca de 1600 figuras.

A possibilidade de importar as definições dos processos pode incentivar a definição dos processos, já que o usuário estaria reusando as informações já definidas. Assim, o usuário não é obrigado a definir os processos logo de início, o que poderia gerar uma barreira no uso da aplicação pela burocratização do uso. Ele irá definir os processos quando lhe for conveniente, de forma gradual.

5.2.3.10 Caso de Uso Definir Métricas

Deverá ser possível a aplicação de métricas para futuras análises ou para qualificar e quantificar as atividades, recursos e esforços, tendo em vista, por exemplo, a aferição das estimativas de tempo e custo, uma vez que as estimativas poderão ser apoiadas nos dados históricos.

5.2.4 Monitorar Projetos

Essa funcionalidade deverá disponibilizar as informações do andamento dos projetos, para que os envolvidos possam acompanhar e monitorar o desenvolvimento dos projetos. O Diagrama de Casos de Uso é apresentado na Figura 5.5.

5.2.4.1 Caso de Uso Monitorar *Baselines*

Esse caso de uso deverá prover ferramentas para o monitoramento e acompanhamento das *baselines* definidas. Poderão ser disponibilizados relatórios de acompanhamento e gráficos de controle para, por exemplo, fazer a comparação entre o cronograma inicial (previsto no planejamento do projeto), o cronograma real (cronograma do andamento atual das atividades) e o cronograma de esforços (cronograma do tempo efetivo de trabalho em cada atividade).

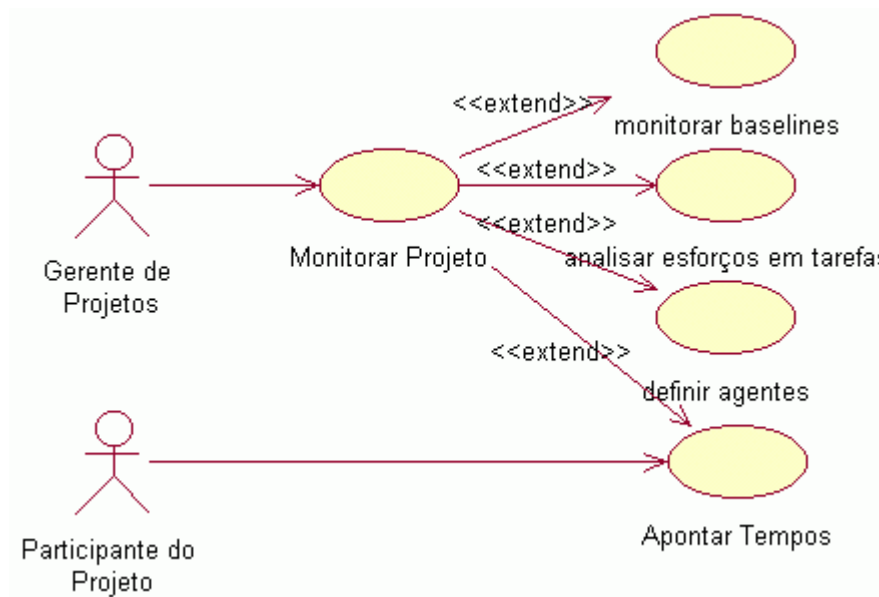


FIGURA 5.5 - Diagrama de caso de uso do monitorar projeto.

5.2.4.2 Caso de Uso Analisar Esforços em Tarefas

Poderá ser dada opção ao gerente de projetos de analisar os tempos de trabalho despendidos por uma pessoa em uma tarefa. Isso é normalmente conhecido como aprovação/rejeição do *time sheet*, essa prática é normalmente usada pelo gerente em equipes sub contratadas e que são remuneradas por hora de desenvolvimento.

5.2.4.3 Caso de Uso Definir Agentes

Esse caso de uso deverá permitir, por exemplo, que sejam construídos sinaleiros ou indicadores, que demonstrem: superalocação ou ociosidade dos recursos, atrasos e desvios no cronograma, etc. Os indicadores poderão ser enviados por e-mail, em forma de alarme aos interessados, por exemplo.

5.2.4.4 Caso de Uso Apontar Tempos

Esse caso de uso deve permitir que os alocados às tarefas possam apontar os tempos de trabalho despendidos na tarefa. Para este objetivo poderá ser usado a ferramenta *time-sheet*.

5.3 Proposta de Arquitetura

Com base nos requisitos levantados e analisados na sessão anterior, torna-se possível a definição de uma proposta de arquitetura para suporte e automação dos processos de gerência de projetos analisados nas sessões anteriores.

O suporte e automação de processos por uma aplicação se torna interessante, já que a mesma pode: fornecer uma infra-estrutura que apóie a manutenção das informações; automatizar, ou executar, partes de processos de gerência de projeto; e implementar serviços que sistematizem métodos, técnicas e modelos necessários.

É importante ressaltar que essa arquitetura foi projetada com a intenção de integrá-la a um PSEE, o que forneceria uma maior eficácia para ambos ambientes, uma vez que essa arquitetura poderia contar com integração junto a todas as outras ferramentas necessárias para o desenvolvimento do projeto, vale porém, salientar que a integração dos ambientes não é necessária para a implementação da arquitetura.

A Figura 5.6 mostra a arquitetura proposta com respectivos componentes.

5.3.1 Aplicação

Como apresentado anteriormente, a aplicação é composta por ferramentas e serviços. São eles:

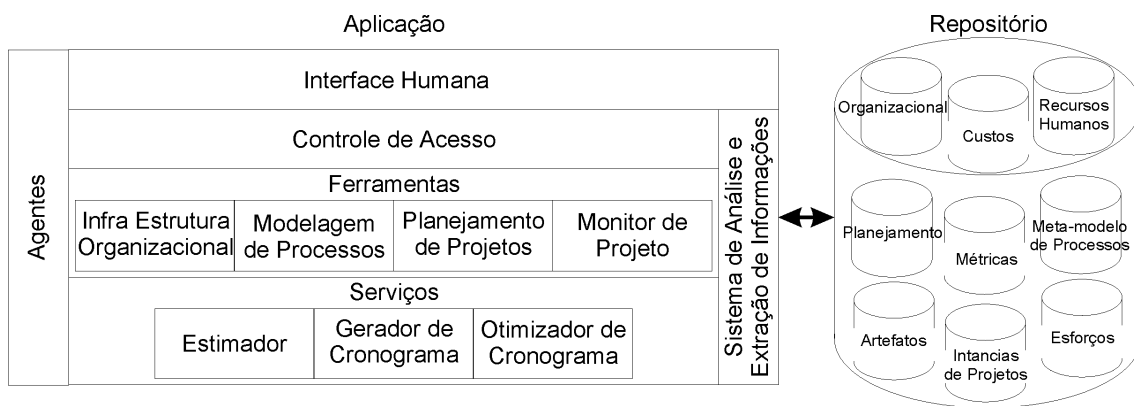


FIGURA 5.6 - Arquitetura proposta.

- 1) Ferramenta de infra-estrutura organizacional.
- 2) Ferramenta de modelagem de processos.
- 3) Ferramenta para planejamento de projetos.
- 4) Ferramenta para monitoramento de projetos.
- 5) Serviço de controle de acesso.
- 6) Serviço estimador de tempo e custo.
- 7) Serviço de apoio à geração do cronograma de projeto.
- 8) Serviço otimizador de cronograma.
- 9) Agentes pró-ativos.
- 10) Sistema de análise e extração de informações.

A Tabela 5.2 mostra a relação entre os casos de uso levantados na análise de requisitos e as funcionalidades oferecidas pela arquitetura, as quais foram apresentados e numerados anteriormente.

TABELA 5.2 - Atores X Casos de Uso X Funcionalidades.

		Funcionalidades									
		1	2	3	4	5	6	7	8	9	10
Organizacional	Definir Infra-Estrutura Organizacional										
	Definir pessoas	X									
	Definir equipes	X									
	Definir recursos de infra-estrutura	X									
	Alocar pessoas às equipes	X									
	Alocar pessoas aos projetos	X									
	Definir capacitação dos recursos	X									
	Definir parâmetros de contratação e de custos	X									
	Definir papéis	X				X					
	Definir permissões de acesso aos papéis	X				X					
Engenheiro de Processos	Definir Modelos de Processo		X								
	Planejar Projeto										
	Definir atividades			X							
	Sequenciar atividades			X							

(Continua)

TABELA 5.2 – Conclusão.

Gerente de Projetos	Estimar tempos das atividades		X		X			X
	Desenvolver cronograma		X			X	X	
	Definir artefatos de entrada e saída das atividades		X					
	Alocar recursos humanos		X					
	Alocar recursos de infra-estrutura		X					
	Estabelecer <i>baselines</i>		X					
	Definir métricas		X		X			
	Importar modelos de processos	X	X					X
	Monitorar Projeto							
	Monitorar <i>baselines</i>			X				
	Analisar esforços em tarefas			X				
	Definir Agentes			X				X
Participante do Projeto	Apontar Tempos			X				

A seguir será apresentada uma descrição das funcionalidades da arquitetura.

5.3.1.1 Ferramenta de Infra-Estrutura Organizacional

Os principais objetivos da ferramenta é permitir a manutenção das informações organizacionais no que tange a infra-estrutura organizacional e os recursos, mantendo informações:

- das pessoas e respectivas alocações em departamentos, áreas, sessões, etc.;
- das equipes e respectivas alocações de pessoas;
- dos perfis das pessoas, indicando quais atores ou responsabilidades cada pessoa poderá exercer durante um projeto;
- dos custos dos recursos humanos, bem como dos tipos de contratação, como: por hora, por tarefa, por projeto, por objetivo/meta, prêmios, etc.

Esse caso de uso irá usar os repositórios de:

- recursos humanos;
- organizacional;
- custos.

5.3.1.2 Ferramenta de Modelagem de Processos

Essa ferramenta deverá proporcionar a modelagem e definição dos processos que são utilizados por uma organização. Essa ferramenta usa uma PML específica, a qual é usada para a representação gráfica da composição e dos elementos de um processo.

Essa ferramenta pode prover outros tipos de serviços, como simulação do processo, automatização do processo, monitoração do processo, análise do processo, extração de métricas do processo, dentre outros. Essa ferramenta será melhor abordada em outro trabalho relacionado a esse, através de outra dissertação também do grupo de engenharia de software da CAP.

Esse caso de uso irá usar o repositório de:

- Meta-Modelo de Processos;
- Artefatos.

5.3.1.3 Ferramenta para Planejamento de Projetos

Como outras ferramentas de planejamento de projetos esta deverá, no mínimo, apoiar a definição e seqüenciamento das tarefas, alocação dos recursos e estabelecimento de cronogramas e *baselines*.

O que pode ser melhorado é a forma de apresentação e preenchimento dos dados na ferramenta. Os dados podem ser solicitados, pela ferramenta, de forma incremental, ou passo a passo, de forma que a ferramenta conduza o usuário a seguir o processo do PMBOK2000, por exemplo, de forma transparente ao usuário.

Por exemplo, quando o usuário da ferramenta, ou ator responsável por essa ação for definir um novo projeto e o mesmo necessite descrever as tarefas, a ferramenta pode, num sistema de interfaces passo a passo, solicitar nesse instante outras informações como o seqüenciamento das tarefas e os tempos estimados; com base nessas informações a ferramenta poderá gerar o cronograma. Para definir os tempos estimados a ferramenta pode fazer uso do serviço Estimador, que será descrito posteriormente.

Com esse tipo de metodologia a ferramenta estará apoiando os processos descritos no PMBOK2000 e estará induzindo o usuário a usar ou mesmo seguir o processo definido. Vale ainda salientar que, nesse exemplo, anteriormente descrito, a ferramenta estaria apoiando alguns dos processos principais de planejamento, descritos pelo PMBOK2000, e até executando um dos processos automaticamente, baseando-se nos dados fornecidos pelos processos anteriores.

Os processos citados no exemplo são: Definição das Atividades, Seqüenciamento das Atividades, Estimativa da Duração das Atividades e Desenvolvimento do Cronograma, os quais foram apresentados na sessão anterior e mostrados através da Figura 4.18.

Além das funcionalidades citadas anteriormente, a ferramenta de planejamento de projetos também deverá dar suporte à:

- alocação dos recursos necessários para o desenvolvimento da tarefa;
- definição dos artefatos de entradas e saída para cada tarefa, o que pode ser integrado aos processos de garantia da qualidade e aos processos de controle de configuração e versão, processos estes que também estão sendo pesquisados em outros trabalhos do Grupo de Engenharia de Software da CAP;
- importação de modelos de processos previamente definidos pela ferramenta de Modelagem de Processos.

A importação dos modelos de processos é um grande diferencial das ferramentas hoje disponíveis comercialmente. O principal objetivo dessa funcionalidade é permitir o reuso da definição de processos, de maneira que o usuário possa “montar” o projeto de acordo com necessidades específicas e particulares de cada projeto, através da importação de modelos de processos definidos.

As métricas também poderão ser aplicadas para qualificar e quantificar as atividades, recursos e esforços, tendo em vista a aferição das estimativas de tempo e custo, uma vez que as estimativas poderão ser apoiadas nos dados históricos.

Esse caso de uso irá usar os repositórios de:

- Recursos Humanos;

- Planejamento;
- Métricas;
- Artefatos;
- Meta-modelo de Processos;
- Organizacional.

5.3.1.4 Ferramenta para Monitoramento de Projetos

Essa ferramenta deverá disponibilizar as informações do andamento dos projetos, para que os envolvidos possam acompanhar e monitorar o desenvolvimento dos projetos.

Essa ferramenta deverá prover diferentes visões para os diferentes envolvidos no projeto, ela terá apoio do serviço de controle de acesso, que identificará qual ator está solicitando o monitoramento naquele instante. Porém, ela deverá proporcionar as diferentes granularidades, ou mesmo, o tratamento das informações de forma diferente de acordo com a necessidade e interesse de cada ator.

Deverá disponibilizar dispositivos e relatórios como comparativos entre as *baselines*, análise das taxas de progresso de cada tarefa e dos agentes que podem ser programados para executar determinados controles ou alarmes.

Outro diferencial é que o apontamento do avanço na execução de cada tarefa é feito pelo recurso alocado à tarefa, e não pelo gerente de projetos, como em algumas ferramentas comerciais. Essa característica foi herdada da abordagem do PSP (Hayes; Over, 1997) onde se tem o conceito de *time sheet*. O *time sheet* já passou a ser adotado e implementado nas últimas versões de algumas ferramentas comerciais.

O principal objetivo do *time sheet* é registrar os tempos despendidos em cada tarefa, permitindo dessa forma o acompanhamento do projeto através das taxas de progresso.

Poderão, também, ser aplicadas métricas às tarefas de modo que seja possível futuras análises ou quantificação ou qualificação das tarefas.

Esse caso de uso irá usar os repositórios de:

- recursos humanos;
- organizacional;
- custos;
- planejamento;
- métricas;
- artefatos;
- instâncias de projetos;
- esforços.

5.3.1.5 Serviço de Controle de Acesso

Por segurança e até mesmo ergonomia, esse serviço deverá proporcionar a adequação da aplicação, e do PSEE, a cada ator (Booch et al., 2000). Isso permitirá que cada usuário tenha seu próprio *desktop*, que será montado dinamicamente a partir das permissões, dos projetos em que estiver alocado e das tarefas atribuídas ao usuário.

Esse serviço permite que o usuário dê permissões aos atores. Dessa forma, quando um recurso estiver desempenhando a função desse ator, e somente

quando estiver, o recurso terá permissão de acesso às funcionalidades permitidas àquele ator.

O recurso assume a função de um ator quando o mesmo é alocado para executar uma tarefa, e esta por sua vez, necessita de um certo ator para ser desempenhada. Dessa forma, no momento que o recurso for executar essa tarefa, o mesmo terá acesso às funcionalidades atribuídas anteriormente.

Esse serviço pode também ser uma iniciativa de solução para o problema de dificuldade de uso, o qual foi citado por Fuggetta (2000), como um fator que inibe o uso e a disseminação dos PSEEs.

Esse serviço irá usar os repositórios de:

- recursos humanos;
- organizacional;
- instâncias de projetos.

5.3.1.6 Estimador

O serviço de estimador pode ser dividido em dois serviços com aplicações diferentes. São eles:

- **Serviço de apoio às estimativas de tempo das atividades.**

Poderá ser realizada uma previsão de tempo baseando-se nas métricas aplicadas às atividades, aos recursos e aos esforços, ou nos metadados das tarefas obtidos na definição das mesmas.

Uma implementação simples desse serviço seria o cálculo da média dos tempos de apontamentos de tarefas em projetos anteriores, os quais estão armazenadas no repositório de instâncias de projetos, com o mesmo tipo, ou que fosse baseada no mesmo modelo de tarefa ou

modelo de processo que poderão ter sido importados pela ferramenta de planejamento de projeto. Ainda poderiam ser usados filtros como: recursos ou equipes, projetos, períodos de tempos pré definidos, etc.

Esse serviço é utilizado principalmente pela ferramenta de planejamento de projeto.

Esse serviço irá usar os repositórios de: recursos humanos, esforços, instancias de projetos, organizacional, métricas

- **Serviço de apoio às estimativas de custo do projeto.**

Poderá ser realizada uma previsão de custo baseando-se nas métricas aplicadas às atividades, aos recursos ou aos esforços, bem como nos meta-dados das tarefas obtidos na definição das mesmas, e nos dados históricos de instâncias de projetos anteriores.

Os custos poderão ser extraídos com base em projetos anteriores ou em tabelas de valores pré-definidas as quais poderiam informar valores de: atividades, homem/hora, homem/mês, homem/projeto, papel/mês, papel/hora e tempo de projeto, etc.

Esse serviço é utilizado principalmente pela ferramenta de planejamento de projeto.

Esse serviço irá usar os repositórios de: recursos humanos, esforços, instancias de projetos, organizacional, métricas, custos

Alguns dos principais métodos e técnicas de estimativas e medição que poderão ser aplicados são COCOMO e Pontos por Função (Fernandes, Kugler, 1989) (Belloquim, 2002); segundo Fernandes (1995), o uso combinado desses métodos podem resultar em boas estimativas de tempo e custo.

Outra técnica de estimativa, mais recente, que também poderá ser avaliada é o Pontos por Caso de Uso, pois promete um maior grau de precisão nas

estimativas (Belloquim, 2002) ou, ainda, o *Earned Value* (Vargas, 2002) que é muito utilizado em gerenciamento de projetos (PMI, 2000)

5.3.1.7 Serviço de Apoio à Geração do Cronograma de Projeto.

Este serviço deverá propor uma lista possível, ou até mesmo coerente, de alocação de pessoas, baseando-se nas informações referentes à capacitação dos recursos humanos e disponibilidade dos recursos de projeto.

Poderão também ser implementados algoritmos ou técnicas para geração de cronograma.

Esse serviço é utilizado principalmente pela ferramenta de planejamento de projeto.

Esse serviço irá usar os repositórios de:

- recursos humanos;
- esforços;
- instancias de projetos;
- organizacional.

5.3.1.8 Serviço Otimizador de Cronograma.

Serviço capaz de seqüenciar as tarefas e alocar os recursos, de forma que os recursos não fiquem ociosos ou superalocados e que o prazo do projeto seja o menor possível. Para a criação do cronograma otimizado, poderão ser empregadas heurísticas sofisticadas, como a análise de desempenho de cada pessoa, aliada a técnicas de alocação de recursos, como o Método de Caminho Crítico (Vasco, 2001).

Algumas das abordagens, já analisadas, relatam a necessidade de manutenção das capacitações dos recursos. Essa informação, aliada aos dados históricos e às métricas aplicadas às atividades, aos recursos e esforços, podem servir de subsídio para que o ambiente proponha um cronograma de projeto otimizado, ou seja, seqüenciar as tarefas e alocar os recursos nas tarefas, de forma que os recursos não fiquem ociosos ou super alocados e o término do projeto seja o menor possível.

Para a criação do cronograma otimizado, poderão ser empregadas heurísticas sofisticadas, como a análise de desempenho de cada pessoa aliada a técnicas de alocação de recursos, como o Método de Caminho Crítico (Vasco, 2001).

Poderão também ser implementados algoritmos para nivelamento de recursos.

Esse serviço irá usar os repositórios de:

- recursos humanos;
- esforços;
- instancias de projetos;
- organizacional.

5.3.1.9 Agentes Pró-Ativos.

Os agentes proporcionam o fluxo dos eventos e das atividades estabelecidas nos processos de gerência (Sant'Anna et al., 2002). Eles podem realizar serviços de auxílio aos envolvidos no projeto e executar atividades para automação de etapas dos processos, além de monitorar o acompanhamento do projeto.

Poderão, por exemplo, ser construídos sinaleiros ou indicadores, que demonstrem: superalocação ou ociosidade dos recursos, atrasos e desvios no

cronograma, etc. Os indicadores poderão ser enviados por e-mail, em forma de alarme aos interessados.

5.3.1.10 Sistema de Análise e Extração de Informações.

O modelo de informações proposto possibilita a extração de vários relatórios, que poderão ser usados no auxílio à tomada de decisões durante o acompanhamento dos projetos. Alguns exemplos de relatórios são:

- comparativo entre o cronograma inicial (previsto no planejamento do projeto), o cronograma real (cronograma do andamento atual das atividades) e o cronograma de esforços (cronograma do tempo efetivo de trabalho em cada atividade);
- visão geral dos projetos da organização;
- visão do andamento do projeto para o cliente;
- estatísticas de alocação/ociosidade dos recursos;
- estatísticas de tempo real ou tempo efetivo de trabalho;
- análise para adequação das pessoas em funções que melhor desempenham;
- visualização do ciclo de vida do projeto e das interações entre as fases e processos de cada projeto.

5.3.2 Repositório

Após a análise dos requisitos das abordagens de processos, foi desenvolvido um modelo de informações (Cougo, 1997) capaz de suportar a implementação, apoio e automação dos processos. Esse modelo de informações deve suportar:

- a definição dos modelos de processo (meta-modelos de processos);
- os requisitos dos modelos de processo;
- as necessidades da execução do processo;
- os repositórios de custos, artefatos, esforços e métricas;
- o planejamento de projetos;
- as instâncias de vários processos;
- o gerenciamento simultâneo de múltiplos projetos.

O modelo de informações também é bastante robusto, a fim de suportar futuras extensões para suporte aos outros processos de gerenciamento de projetos.

Outra característica importante desse modelo é a manutenção das informações dos projetos, pois a formação, análise e avaliação da base histórica de projetos é citada por vários autores como um grande fator de auxílio no sucesso da gerência (DeMarco, 1989) (Fernandes; Kugler, 1989) (Wysocki et al., 1995) (Hayes; Over, 1997) (Smith et al., 2001).

O suporte ao repositório de artefatos e o repositório de métricas se tornam importantes, uma vez que esses são recomendados na especificação do nível 2, e exigidos no nível 3 do CMM (Vasques, 2002). O repositório de métricas também será de grande utilidade na oportunidade em que for tratado sobre as estimativas de tempo, custo e esforço.

Na especificação do nível 4 do CMM, o repositório de métricas e os dados históricos são usados para a tomada de decisões (Vasques, 2002); portanto, se faz necessário o estudo de viabilidade de implantação de algumas métricas, como COCOMO (Boehm, 1981), Pontos por Função (IFPUG, 1994), Pontos por Caso de Uso (Belloquim, 2002), *Earned Value* (Vargas, 2002), ou outras. Fernandes (1989), por exemplo, utiliza os métodos COCOMO e Pontos por

Função, pois segundo o autor, o uso combinado desses métodos resulta em boas estimativas de tempo e custo.

As métricas poderão, também, ser aplicadas para qualificar e quantificar as atividades, recursos e esforços, tendo em vista a aferição das estimativas de tempo e custo, uma vez que as estimativas poderão ser apoiadas nos dados históricos.

A implantação do modelo de informações foi chamado de Repositório, conforme pode ser visualizado na figura que mostra a arquitetura proposta na sessão anterior.

CAPÍTULO 6

CONCLUSÃO

6.1 Resultados

Uma vez concluída a definição dos elementos da arquitetura e da implementação do protótipo, foi criada uma tabela de relacionamento a fim de validar as funcionalidades propostas pela arquitetura, nessa tabela foram relacionados os processos de cada abordagem utilizada (CMMI, SPICE e PMBOK2000), com as respectivas ferramentas e serviços que atendem aos requisitos impostos por cada um dos processos. A seguir, serão relacionados e numerados os elementos propostos pela arquitetura:

- 1) Ferramenta de infra-estrutura organizacional;
- 2) Ferramenta de modelagem de processos;
- 3) Ferramenta para planejamento de projetos;
- 4) Ferramenta para monitoramento de projetos;
- 5) Serviço de controle de acesso;
- 6) Serviço estimador de tempo e custo;
- 7) Serviço de apoio à geração do cronograma de projeto;
- 8) Serviço otimizador de cronograma;
- 9) Agentes pró-ativos;
- 10) Sistema de análise e extração de informações.

O cruzamento das Funcionalidades Oferecidas X Requisitos é mostrado em três tabelas, onde “N” significa o não cumprimento desse item pelas funcionalidades propostas; “X” significa que a funcionalidade daquela coluna atende ao item daquela linha; e “P” significa que a funcionalidade daquela coluna atende parcialmente, ou seja, atende a pelo menos 50% dos requisitos do item daquela linha :

- a Tabela 6.1 mostra o relacionamento das funcionalidades com os processos do PMBOK2000;
- a Tabela 6.2 mostra o relacionamento das funcionalidades com os processos do CMMI
- a Tabela 6.3 mostra o relacionamento das funcionalidades com os processos do SPICE.

TABELA 6.1 - Relacionamento das funcionalidades com o PMBOK2000.

Abordagem/Processos	Funcionalidade									
	1	2	3	4	5	6	7	8	9	10
PMBOK2000										
<i>Gerenciamento da Integração do Projeto</i>										
Desenvolvimento do Plano do Projeto			P							
Execução do Plano do Projeto			P	P						
Controle Integrado de Mudanças	N									
<i>Gerenciamento do Escopo do Projeto</i>										
Iniciação	N									
Planejamento do Escopo	N									

(Continua)

TABELA 6.1 – Continuação.

Detalhamento do Escopo	N									
Verificação do Escopo	N									
Controle de Mudanças do Escopo	N									
Gerenciamento do Tempo do Projeto										
Definição das Atividades		X								
Seqüenciamento das Atividades		X								
Estimativa da duração das atividades					X					
Desenvolvimento do cronograma		X	X			X	X			
Controle do cronograma			X							
Gerenciamento dos Custos do Projeto										
Planejamento dos recursos		X								
Estimativa dos custos					X					
Orçamento dos custos					X					
Controle dos custos			X							
Gerenciamento da Qualidade do Projeto										
Planejamento da Qualidade	N									
Garantia da Qualidade	N									
Controle da Qualidade	N									
Gerenciamento dos Recursos Humanos do Projeto										

(Continua)

TABELA 6.1 – Continuação.

Planejamento organizacional	X																		
Montagem da equipe	X																		
Desenvolvimento da equipe				X															
Gerenciamento das Comunicações do Projeto																			
Planejamento das comunicações						P													
Distribuição das informações						P													
Relato de desempenho				X															
Encerramento administrativo	N																		
Gerenciamento dos Riscos do Projeto																			
Planejamento da Gerência de Riscos	N																		
Identificação dos Riscos	N																		
Análise Qualitativa dos Riscos	N																		
Análise Quantitativa dos Riscos	N																		
Desenvolvimento de Resposta a Riscos	N																		
Controle e Monitoração dos Riscos	N																		
Gerenciamento das Aquisições do Projeto																			
Planejamento da Aquisições	N																		
Preparação das Aquisições	N																		
Obtenção de Propostas	N																		

(Continua)

TABELA 6.1 – Conclusão.

Seleção de Fornecedores	N													
Administração dos Contratos	N													
Encerramento do Contrato	N													

Pode-se observar, pela tabela anterior, que foi possível contemplar outros processos além dos propostos (Recursos Humanos, Custos e Tempo), como parte dos Processos de Comunicação.

TABELA 6.2 - Relacionamento das funcionalidades com o CMMI.

Abordagem/Processos	Funcionalidade									
	1	2	3	4	5	6	7	8	9	10
CMMI										
PLANEJAMENTO DO PROJETO										
<i>SG 1 Estabelecer Estimativas</i>										
SP 1.1-1 Estimar o Escopo do Projeto	N									
SP 1.2-1 Estabelecer Estimativas de Atributos do Projeto	N									
SP 1.3-1 Definir o Ciclo de Vida do Projeto	N									
SP 1.4-1 Determinar as Estimativas de Esforço e Custo						X				
<i>SG 2 Desenvolver um Plano de Projeto</i>										
SP 2.1-1 Estabelecer Orçamento e Programação						X				

(Continua)

TABELA 6.2 – Continuação.

SP 2.2-1 Identificar os Riscos do Projeto	N																		
SP 2.3-1 Planejar o Gerenciamento de Dados					P														
SP 2.4-1 Planejar os Recursos do Projeto			X																
SP 2.5-1 Planejar as Habilidades e Conhecimentos Necessários	X			X															
SP 2.6-1 Planejar o Envolvimento de <i>Stakeholder</i>				P	P														
SP 2.7-1 Estabelecer o Plano de Projeto	N																		
<i>SG 3 Obter Compromisso para o Plano</i>																			
SP 3.1-1 Rever os Planos Subordinados	N																		
SP 3.2-1 Reconciliar os Níveis de Recurso e Trabalho	N																		
SP 3.3-1 Obter o Compromisso do Plano	N																		
CONTROLE E MONITORAMENTO DO PROJETO																			
<i>SG 1 Monitorar o Projeto em Comparação com o Plano</i>																			
SP 1.1-1 Monitorar os Parâmetros do Planejamento do Projeto				X															
SP 1.2-1 Monitorar Compromissos				X															
SP 1.3-1 Monitorar Riscos de Projeto	N																		
SP 1.4-1 Monitorar Gerenciamento de Dados				P	P														

(Continua)

TABELA 6.2 – Continuação.

SP 1.5-1 Monitorar o Envolvimento de <i>Stakeholder</i>			P	P										
SP 1.6-1 Conduzir Revisões de Progresso			X											
SP 1.7-1 Conduzir Revisões <i>Milestone</i>			X											
<i>SG 2 Gerenciar a Ação Corretiva para Encerrar</i>														
SP 2.1-1 Analisar Problemas	N													
SP 2.2-1 Tomar Ação de Correção	N													
SP 2.3-1 Gerenciar Ação Corretiva	N													
GERENCIAMENTO DE CONTRATO DO FORNECEDOR														
<i>SG 1 Estabelecer os Contratos de Fornecedor</i>														
SP 1.1-1 Analisar as Necessidades e Exigências Determinadas pelo Projeto	N													
SP 1.2-1 Selecionar os Fornecedores	N													
SP 1.3-1 Estabelecer os Contratos dos Fornecedores	N													
<i>SG 2 Satisfazer os Contratos do Fornecedor</i>														
SP 2.1-1 Adquirir os Produtos COTS	N													
SP 2.2-1 Executar o Contrato do Fornecedor	N													
SP 2.3-1 Conduzir o Teste de Aceitação	N													
SP 2.4-1 Transição de Produtos	N													

(Continua)

TABELA 6.2 – Continuação.

GERENCIAMENTO INTEGRADO DO PROJETO																			
<i>SG 1 Utilize o Processo Definido do Projeto</i>																			
SP 1.1-1 Estabelecer o Processo Definido do Projeto	X																		
SP 1.2-1 Utilizar as Avaliações do Processo Organizacional para Planejar Atividades do Projeto	P																		
SP 1.3-1 Integrar Planos	P																		
SP 1.4-1 Gerenciar o Projeto Utilizando Planos Integrados	P																		
SP 1.5-1 Contribuir para as Avaliações do Processo da Organização	P																		
<i>SG 2 Coordenar e colaborar com os Stakeholders Relevantes</i>																			
SP 2.1-1 Gerenciar o Envolvimento do Stakeholder							P	P											
SP 2.2-1 Gerenciar Dependências							P												

(Continua)

TABELA 6.2 – Continuação.

PROJETO QUANTITATIVO																		
SG 1 Gerenciar Quantitativamente o Projeto																		
SP 1.1-1 Estabelecer os Objetivos do Projeto	N																	
SP 1.2-1 Compor o Processo Definido			P															
SP 1.3-1 Selecionar os Sub Processos a serem Gerenciados			P															
SP 1.4-1 Gerenciar o Desempenho do Projeto				X														
SG 2 Gerenciar Estatisticamente o Desempenho do Sub Processo																		
SP 2.1-1 Selecionar as Técnicas Analíticas e Medidas	N																	
SP 2.2-1 Aplicar os Métodos Estatísticos para Entender a Variação	N																	
SP 2.3-1 Monitorar o Desempenho dos Sub Processos Selecionados				X														

(Continua)

TABELA 6.2 – Conclusão.

SP 2.4-1 Gravar os Dados do Gerenciamento Estatístico	N																			
---	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

TABELA 6.3 - Relacionamento das funcionalidades com o SPICE.

Abordagem/Processos	Funcionalidade									
	1	2	3	4	5	6	7	8	9	10
SPICE - ISO 15504										
<i>PRO.1 - Planejar o ciclo de vida do projeto</i>										
PRO.1.1 - Avaliar opções para o desenvolvimento do produto.		X	X							
PRO.1.2 - Selecionar o modelo de ciclo de vida do software.		P								
PRO.1.3 Descrever as tarefas e atividades.			X							
PRO.1.4 Estabelecer a seqüência da tarefa.			X							
PRO.1.5 Documentar atividades.			X							
<i>PRO.2 Estabelecer plano de projeto</i>										
PRO.2.1 Desenvolver a estrutura de análise WBS de trabalho.	N									
PRO.2.2 Identificar os padrões do projeto.		P								
PRO.2.3 Identificar as instalações especializadas.	N									
PRO.2.4 Determinar a estratégia de reutilização.	N									

(Continua)

TABELA 6.3 – Continuação.

PRO.2.5 Desenvolver estimativas do projeto.					X				
PRO.2.6 Identificar os riscos iniciais do projeto.	N								
PRO.2.7 Identificar as medidas do projeto.		X	X		X				
PRO.2.8 Estabelecer a programação do projeto.		X				X	X		
PRO.2.9 Estabelecer compromissos do projeto.		X							
PRO.2.10 Documentar planos de projeto.		P							
<i>PRO.3 Constituição das equipes de projeto</i>									
PRO.3.1 Definir as equipes do projeto.	X	X							
PRO.3.2 Capacitar (autorizar) as equipes do projeto.		X							
PRO.3.3 Manter as interações da equipe de projeto.		X							
PRO.3.4 Gerenciar os problemas entre-equipes.	N								
<i>PRO.4 Gerenciar requisitos</i>									
PRO.4.1 Concordar sobre os requisitos.	N								
PRO.4.2 Estabelecer a linha base das exigências do cliente.	N								
PRO.4.3 Gerenciar as alterações das exigências do cliente.	N								
PRO.4.4 Utilizar as exigências do cliente.	N								
PRO.4.5 Manter a rastreabilidade.	N								

(Continua)

TABELA 6.3 – Continuação.

<i>PRO.5 Gerenciar qualidade</i>																			
PRO.5.1 Estabelecer as metas de qualidade.	N																		
PRO.5.2 Definir as métricas de qualidade.	N																		
PRO.5.3 Identificar as atividades de qualidade.	N																		
PRO.5.4 Executar as atividades de qualidade.	N																		
PRO.5.5 Avaliar a qualidade.	N																		
PRO.5.6 Tomar ação corretiva.	N																		
<i>PRO.6 Gerenciar Riscos</i>																			
PRO.6.1 Estabelecer o escopo do gerenciamento de risco.	N																		
PRO.6.2 Identificar os riscos.	N																		
PRO.6.3 Analisar e priorizar os riscos.	N																		
PRO.6.4 Desenvolver as estratégias de mitigação (atenuação).	N																		
PRO.6.5 Definir as métricas de risco.	N																		
PRO.6.6 Implementar as estratégias de mitigação.	N																		
PRO.6.7 Avaliar os resultados de estratégias de atenuação.	N																		
PRO.6.8 Tomar ação corretiva.	N																		

(Continua)

TABELA 6.3 – Conclusão.

<i>PRO.7 Gerenciar os recursos e a programação</i>																			
PRO.7.1 Adquirir recursos.		X																	
PRO.7.2 Rastrear progresso.			X																
PRO.7.3 Conduzir as revisões de gerenciamento.			P																
PRO.7.4 Conduzir as revisões técnicas.	N																		
PRO.7.5 Gerenciar compromissos.		X																	
<i>PRO.8 Gerenciar os sub contratantes.</i>																			
PRO.8.1 Estabelecer a instrução de trabalho.	N																		
PRO.8.2 Qualificar os sub contratantes potenciais.	N																		
PRO.8.3 Selecionar o sub contratante.	N																		
PRO.8.4 Estabelecer e gerenciar os compromissos.	N																		
PRO.8.5 Manter as comunicações.	N																		
PRO.8.6 Avaliar a conformidade.	N																		
PRO.8.7 Avaliar a qualidade do sub contratante.	N																		

6.2 Considerações Finais

Durante o desenvolvimento do trabalho pôde-se observar que o interesse pelo tema Gerência de Projetos está a cada dia maior, dada as pesquisas e estudos de várias áreas de atuação. Este crescente interesse tem sido notado nas constantes citações em congressos, periódicos, revistas especializadas, fóruns

e listas de discussão e até na imprensa convencional através de jornais, revistas e outras publicações.

Vários foram os fatores que motivaram as pesquisas para o desenvolvimento desse trabalho, vários deles já foram citadas nas sessões anteriores, porém vale salientar alguns deles:

- altos índices de falhas em projetos, principalmente em projetos de TI;
- falta de atenção ou mesmo de conhecimento dos processos de gerenciamento de projetos;
- falha no uso de ferramentas de gerenciamento de projetos;
- falta de motivação e resultados efetivos na implantação de processos, tanto pela burocratização, dificuldade e conseqüente aumento na complexidade do trabalho, como no inexpressivo resultado quanto à implantação manual de processos;
- problemas e dificuldades na aceitação, implantação e no uso dos inflexíveis e complexos PSEEs;
- multidisciplinaridade do tema Gerência de Projetos;
- continuidade do trabalho de pesquisa “Um Ambiente Integrado para o Apoio ao Desenvolvimento e Gestão de Projetos de Software para Sistemas de Controle de Satélites” (Sant’Anna, 2000).

Durante o desenvolvimento do trabalho foi imprescindível a implementação e aplicação dos conceitos e tecnologias pesquisadas através do desenvolvimento de um protótipo que contemplasse a definição da arquitetura aqui proposta.

Através dessa prática foi possível obter um retorno em termos de análise de viabilidade e um retorno efetivo das necessidades de um ambiente de aplicação real. A implementação será apresentada na sessão seguinte.

Durante todo o ciclo de desenvolvimento do trabalho teve-se a preocupação quanto à validade do trabalho de pesquisa, quesito este que foi buscado através da publicação de quatro trabalhos: dois em congressos internacionais, reconhecidos e respeitados pela comunidade de engenharia de software, e outros dois em workshops do curso de pós-graduação.

Foram também atingidos os objetivos pretendidos na proposta de dissertação, bem como respeitadas as ressalvas propostas naquela ocasião.

Um dos diferenciais dessa arquitetura, é a possibilidade de integração a um PSEE, e dessa forma torna-se possível usa-la com o intuito de facilitar e até mesmo incentivar o uso de processos, não só os de gerenciamento de projetos, mas também os de desenvolvimento, já que a arquitetura sistematiza e automatiza partes dos processos de gerenciamento de projetos e possibilita o reuso dos processos de desenvolvimento, de forma que a definição e uso dos processos facilita ou até agiliza o desenvolvimento e o gerenciamento do projeto.

Entende-se, também, que o trabalho de interpretação dos modelos de processos e sua respectiva modelagem usando a linguagem de modelagem de processos UPM pode facilitar o acesso da comunidade de engenharia de software a esses processos, devido à facilidade da notação, uma vez que a PML utilizada é baseada na notação UML, que é bastante conhecida e utilizada pela comunidade de engenharia de software.

6.3 Trabalhos Futuros

O tema gestão de projetos é muito amplo e moderno, atualmente existem várias linhas de pesquisa em diferentes tópicos dentro dessa mesma área.

Esse trabalho concretiza-se como um primeiro passo para a sistematização, automação e definição de uma arquitetura capaz de apoiar a implantação de processos de gerenciamento e de desenvolvimento.

A seguir serão apresentados alguns trabalhos de pesquisa que poderão ser desenvolvidos na área de gerência de projetos, bem como para a continuidade desse trabalho:

- Analisar, modelar e automatizar os outros processos de gerenciamento de projetos;
- Pesquisar métodos e técnicas de estimativa para que os mesmos possam quantificar as atividades, recursos, custos a fim de construir formas mais precisas e adequadas a cada cenário de estimativas de tempo, recursos e custos;
- Pesquisar métodos e técnicas para a viabilização da implementação do serviço de otimização de cronograma, possibilitando a análise de cronogramas com recursos e projetos concorrentes;
- Formas mais eficazes para implementação da reutilização de processos de gerenciamento de projetos e para processos de desenvolvimento;
- Implementação do serviço de agentes autônomos e de alarmes inteligentes

REFERÊNCIAS BIBLIOGRÁFICAS

ABDALA, M. A. D.; LAHOZ NETO, C. H.; SANT'ANNA, N. Um estudo para definição de processos das gerencias da qualidade e da configuração em um ambiente integrado para apoio ao desenvolvimento e gestão de produtos de software. In: Workshop dos Cursos de Computação do INPE 2., (WORCAP), 2002, São José dos Campos. **Anais...** São José dos Campos: INPE, 2002.

ACUÑA, S. T.; FERRÉ, X. Software process modelling. In: World Multiconference on Systemics, Cybernetics and Informatics 5., 2001, Orlando, USA. **Proceedings...** Orlando: International Institute of Informatics Systemics, 2001.

ACUÑA, S.; BARCHIN, G.; LASERRE, C.; SILVA, A.; SOSA, M.; QUINCOSES, V. Software engineering and knowledge engineering software process: formalizing the who's who, In: International Conference on Software Engineering and Knowledge Engineering 12., 2000, Chicago, USA. **Proceedings...** Chicago: Knowledge Systems Institute, 2000.

AMBRIOLA, V.; CONARDI, R.; FUNGGETTA, A. Assessing process centered software engineering environments. **Transaction on Software Engineering and Methodology**, v. 6, n. 3, p. 283-328, 1997.

ANSI/IEEE Std 610.12-1990, **IEEE Standard Glossary of Software Engineering Terminology**. Feb. 1991.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR ISO 10006: Sistema de gestão da qualidade: fundamentos e vocabulário. Rio de Janeiro, dez. 2000.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR ISO 9000/2000: Sistema de gestão da qualidade: fundamentos e vocabulário. Rio de Janeiro, dez. 2000.

BALZER, R.; DYER, D.; MORGENSTERN, M.; NEECHES, R. Specifications based computing environments. In: Proceedings National Conference on Artificial Intelligence, (AAAI), 1983, Washington, D.C. **Proceedings...** Washington, D.C.: [S. I.], 1983. p. 12-16.

BANDEIRA, A. Acompanhamento do custo de sistemas. In: Congresso Regional de Informática. 1984, Brasília. **Anais...** Brasília: SUCESU-DF, 1984.

BELLOQUIM, A. Modelagem de software: ontem, hoje e amanhã. **Developers' Magazine**. v. 6, n. 70, p. 10-13, Jan. 2002.

BELZER, K. The Program Office: a business results enable, **Project Management World Today**. Disponível em: <[HTTP://WWW.PMFORUM.ORG/PMWT01/PAPERS01-03.HTM](http://www.pmforum.org/pmwt01/papers01-03.htm)>. Acesso em: 10 mar. 2001.

BEN-SHAUL I. Z.; KAISER G. E. A paradigm for decentralized process modeling and its realization in the oz environment. In: International Conference on Software 16., (IEEE), 1994, Sorrento, Italy. **Proceedings...** Los Alamitos, USA: IEEE Computer Society Press, 1994. p. 179-188.

BEN-SHAUL I. Z.; KAISER G. E. Process support for synchronous groupware activities. **Technical Report CUCS-002-95**. New York: Columbia University, 1995.

BOEHM, B. **Software engineering economics**. New Jersey: Prentice-Hall, 1981.

BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. **UML: guia do usuário**. Rio de Janeiro: Campus, 2000.

BORREGO FILHO, L. F.; SANT'ANNA, N.; CEREJA JR., M. G.; GENVIGIR, E. C. Análise e modelagem para automação de processos de gerenciamento de projetos. In: Workshop dos Cursos de Computação do INPE 2., (WORCAP), 2002, São José dos Campos. **Anais...** São José dos Campos: INPE, 2002.

BORREGO FILHO, L. F.; SANT'ANNA, N.; CEREJA JUNIOR, M. G.; GENVIGIR, E. C.; LUQUE, L. Uma aplicação para automatização dos processos de gerenciamento de projetos suportada por ambientes de engenharia de software. In: Congresso Internacional de Tecnologia de Software, 14., Curitiba. **Anais...** Curitiba: CITS, jun. 2003a.

BORREGO FILHO, L. F.; SANT'ANNA, N.; CEREJA JR., M. G.; GENVIGIR, E. C. Proposta de arquitetura para apoio, suporte e automação de processos de gerenciamento de projetos. In: Workshop dos Cursos de Computação do INPE, 3., (WORCAP), 2003, São José dos Campos. **Anais...** São José dos Campos: INPE, 2003.

BORREGO FILHO, L. F.; SANT'ANNA, N.; CEREJA JUNIOR, M. G.; LUQUE, L.; CASILO, B. H. Uma abordagem para processos de gerenciamento do tempo e dos recursos humanos apoiados por ambientes integrados de engenharia de software. In: Simpósio Internacional de Melhoria de Processos de Software (SIMPROS), 2002, Recife. **Anais...** Recife: Centro de Estudos e Sistemas Avançados, 2002.

BOTTONI, F. Só 16% dos projetos de TI cumprem o prazo e o orçamento. **InfoExame**, São Paulo, maio 2001.

BOUDIER, G.; GALLO, F.; MINOT, R.; THOMAS, I. An Overview of PCTE and PCTE+. In: ACM software engineering symposium on practical software development environments 3, 1988, Massachusetts, USA. **Proceedings...** New York: ACM Press, 1988. p. 248–257.

BOURNE, S. R. The UNIX Shell. **The Bell System Technical Journal**, v.57 n.6 p.1971–1990, July-Aug.1978.

CAGAN, M. R. The HP SoftBench environment: an architecture for a New generation of software tools. **Hewlett-Packard Journal**, v.41, n.3, p. 36–47, June, 1990.

CANTOR, M. R. **Object-oriented project management with UML**. Canada: John Wiley & Sons, 1998.

CEREJA JR., M. G.; SANT´ANNA, N.; BORREGO FILHO, L. F. UML e PML: uma exploração de abordagens para a modelagem de processos. In: Workshop dos Cursos de Computação do INPE (WORCAP),2., 2002, São José dos Campos. **Anais...** São José dos Campos: INPE, 2002.

CEREJA JUNIOR, M. G.; SANT´ANNA, N.; BORREGO FILHO, L. F.; GENVIGIR, E. C.; LUQUE, L.; TAVARES, R. P.; CASILLO, B. H. (PDE) process definition environment: uma ferramenta para apoio a definição de processos de software no ambiente e-webproject. In: Congresso Internacional de Tecnologia de Software (CITS) 14., 2003, Curitiba. **Anais...** Curitiba: CITS, 2003.

CEREJA JUNIOR, M. G.; SANT´ANNA, N.; BORREGO FILHO, L. F.; LUQUE, L.; CASILO, B. H. Modelando um processo de gestão de problemas em projeto - UML e PML uma exploração de abordagens para a modelagem de processos. In: Simpósio Internacional de Melhoria de Processos de Software (SIMPROS), Recife. **Anais...** Recife: SIMPROS, 2002.

CHRISTIE, A.M. **Process-centered development environments: an exploration of issues**. Pittsburgh: Software Engineering Institute, 1993. (Technical Report, CMU/SEI-93-TR-4)

CMMI PRODUCT TEAM. **CMMISM for systems engineering/software engineering, version 1.1 (CMMI-SE/SW, V1.1) continuous representation.** Software Engineering Institute - Carnegie Mellon University, Pittsburgh, CMU/SEI-2002-TR-001, Dec. 2001.

COLEMAN, D. **OOD: the fusion method.** New Jersey: Prentice Hall, 1994.

CONALLEN, J. **Building web applications with UML.** 2. ed. Massachusetts: Addison Wesley, 2000.

CONRADI, R.; FERNSTR, C.; FUGGETTA, A.; SNOWDON, B. Towards a Reference Framework for Process Concepts, In: European Workshop on Software Process Tecnology, 2., 1992, London, UK. **Proceedings...** Trondheim, Springer-Verlag, 1992.

CÔRTEZ M. L. **Curso de extensão em engenharia de software: modelos de qualidade de software.** Campinas: Unicamp, 2001.

COUGO, P. S. **Modelagem conceitual e projeto de banco de dados.** Rio de Janeiro: Campus, 1997.

CRAWFORD, J. K. Improving organizational productivity with a project office, **Contract Management**, v. 40, June, 2000.

CRUZ, T. **Workflow: a tecnologia que vai revolucionar processos.** 2 ed. São Paulo: Atlas, 2000.

CUNHA, J.B.S; NAKANISHI T. Quality and productivity: the control of software system elements. In: The Word Congress on Optimal Design of Structural System: Structural Optimization, 1993, Rio de Janeiro. **Proceedings...** Rio de Janeiro: Herskovits, 1993. v. 2, p. 483-490.

DASKALANTONAKIS, M. K. A Practical view of software measurement ans implementation experiences within motorola. **IEEE Transactions on Software Engineering**, v.18, n. 11, Nov. 1992.

DEMARCO, T. **Controle de projetos de software**: gerenciamento, avaliação, estimativa. Rio de Janeiro: Campus, 1989.

DERNIAME, J.; KABA, B.; WASTELL, D. Software process: principles, methodology and technology. **Lecture Notes in Computer Science**, n.1500. Berlin, Springer 1999.

DINSMORE, P.C. Sixteen reason not to implement a project office. **PM Network**, v. 16, n. 2, Feb. 2002.

DOWSON, M.; NEJMEH, B.; RIDDLE, W. Fundamental software process concepts, In: European Workshop on Software Process Modeling.1991, Milan, **Proceedings...** Milan,1991.

ELLMER, E. Process centered software engineering environments as the next-generation of CASE Tools. Disponível em: <[HTTP://WWW.IFS.UNIVE.AC.AT](http://www.ifs.unive.ac.at)>. Acesso em: 12 fev. 2004.

FEILER, P. H.; HUMPHREY, W. S. **Software process development and enactment**: concepts and definitions. Pittsburgh: Software Engineering, 1991

FEILER, P.H., HUMPHREY, W.S. Software process development and enactment: concepts and definitions, In: International Conference on Software Process, 2., Berlin. **Proceedings...** Berlin,1993.

Feldman, S. M. A program to maintain programs. **Software Practice and Experience**, v.9, n.3 p.255–265, Mar. 1979.

FERNANDES, A. A. **Gerência efetiva de software através de métricas**. São Paulo: Atlas, 1995. 328 p.

FERNANDES, A. A.; KUGLER, J. L. C. **Gerência de projetos de sistemas**: uma abordagem prática. Rio de Janeiro: LTC - Livros Técnicos e Científicos, 1989.

FORMAN, E. H. **Decision by objectives**. Washington DC: George Washington University, 2000.

FRAME, J. D. **Managing projects in organizations: how to make the best use of time, techniques, and people**, New York: Jossey Bass, 1995.

FRANCH, X.; RIBÓ, J. M. Using UML for modelling the static part of a software process. In: Beyond the Standard International Conference, 2., Fort Collins CO, USA. **Proceedings...** Berlin: Springer, 1999. p. 28-30.

FUGGETTA, A. Software process: a roadmap. In: International Conference on Software Engineering, 2000, Limerick, Ireland. **Proceedings...** New York: ACM Press, 2000.

GARG, P. K.; JAZAYERI, M. **Process-centered software engineering environments**. Los Alamitos: IEEE Computer Society Press, 1996.

GENVIGIR, E. C.; SANT'ANNA, N.; BORREGO FILHO, L. F.; CEREJA JR., M. G. Uma abordagem para os processos da engenharia de requisitos. In: Workshop dos Cursos de Computação do INPE (WORCAP), 2., São José dos Campos. **Anais...** São José dos Campos: INPE, 2002.

GENVIGIR, E. C.; SANT'ANNA, N.; BORREGO FILHO, L. F.; CEREJA JUNIOR, M. G.; CASILO, B. H. Modelando processos de software através do UPM - modelo de processo unificado. In: Congresso Internacional de Tecnologia de Software (CITS'2003), 14., 2003, Curitiba. **Anais...** Curitiba: CITS, junho, 2003.

GEORGAKOPOULOS, D.; HORNICK, M. An overview of workflow management: from process modeling to workflow automation infrastructure. **Distributed and Parallel Databases**, v.3, p.119-153, 1995.

GIMENES, I. M. S. **ExpPSEE**: um ambiente experimental de engenharia de software orientado a processos. Relatório (Projeto) – Departamento de Informática, Universidade Estadual de Maringá, Maringá, 2000.

GIMENES, I. M. S. O processo de engenharia de software: ambientes e formalismos. In: Jornada de Atualização em Informática (JAI),13. **Anais...** Caxambú, 1994.

GOODPASTURE, J. C., The project office: finding pearls and avoiding perils. In: Proceedings of the Project Management Institute Annual Seminars & Symposium,2000, Houston. **Proceedings...** Sept. 2000.

GRAHAM, I; HENDERSON-SELLERS, B.; YONESSI, H. **The OPEN process specification**. London: Addison-Wesley, 1997. 314 p.

GREEN, C.; LUCKAM, D.; BALZER, R.; CHEATHAM, T.; RICH, C. **Report on a knowledge based software assistant**. Kestrel Institute, June 1983. (Technical Report KES.U.83.2).

HAYES, W; OVER, J. W. **The personal software process (PSP)**: an empirical study of the impact of PSP on individual engineers. Pittsburgh:CMU/Software Engineering Institute, Dec. 1997. Technical Report (CMU/SEI-97-TR-001 ESC-TR-97-001).

HELDMAN, K. **Gerência de projetos**. Rio de Janeiro: Campus, 2003.

HUMPHREY, W. S. **The Team Software Process (TSP)**. Pittsburgh:CMU/Software Engineering Institute, Nov. 2000. Technical Report CMU/SEI-2000-TR-023) (ESC-TR-2000-023).

HUMPHREY, W. S.; KELLNER, M. **Software process modeling**: principles of entity process models. Pittsburgh/Pensylvania:SEI Software Institute - Carnegie Mellon Institute, 1989. CMU/SEI-89-TR-2.

IFPUG. **Function points counting practice manual** - Release 4.0. Westerville.Ohio,1994.

INGALLS, D. The smalltalk-76 programming system, design and implementation. In: Annual ACM Symposium on Principles of Programming Languages,5. **Proceedings...** Association of Computing Machinery,1978. p. 9-17.

ISO/IEC - INTERNATIONAL STANDARD ORGANIZATION (ISO). **10006 - Quality management** - Guidelines to quality in project management, ISO/IEC, Dec. 1997.

ISO/IEC - INTERNATIONAL STANDARD ORGANIZATION (ISO). **12207 - Software Life Cicle Processes**, ISO/IEC 1995.

JACCHERI MARIA L.; BALDI M.; DIVITINI M. Evaluating the requirements of software process modeling languages and systems, In: Process Support for Distributed Team-based Software Development (PDTSD99), 1999. **Anais...** Orlando, Florida,1999. p. 570-578.

JACCHERI, M. L.; PICCO, G. P.; LAGO, P. Eliciting software process models with the E³ Language. **ACM Transactions on Software Engineering and Methodology**, v. 7, n. 4, p. 368-410, Oct. 1998.

JACOBSON, I.; BOOCH, G.; RUMBAUGH, J. **The unified software development process**. Canada: Addison Wesley Longman, 1999.

JONES, C. **Applied software measurement**. New York: McGraw- Hill,1991.

KADIA, R. Issues encountered in building a flexible software development environment: lessons from the arcadia project, In: ACM SIGSOFT Symposium on Software Development Environments, 1992. **Proceedings...** Washington, 1992.

KAISER, G. E.; FEILER, P. H. An architecture for intelligent assistance in software development. In: International Conference on Software Engineering,9. **Proceedings...**, Monterey, California, 1987. p. 180-188.

KELLNER, M I.; HANSEN, G. A. **Software process modeling. SEI software engineering institute**, Carnegie Mellon University, Pittsburgh, Pennsylvania, 1988. CMU/SEI-88-TR-9.

LAHOZ NETO, C. H., ABDALA, M.; SANT'ANNA, N. O processo de auditar produtos: uma pesquisa referente à sua definição e modelagem para a garantia da qualidade no e-WebProject®. In: Simpósio Internacional de Melhoria de Processo de Software (SIMPROS), Recife, **Anais...** Recife: SIMPROS, setembro 2002.

LEWIS, J. P. **The project manager's desk reference**, 2 ed. Boston: MacGraw-Hill, 2000.

LONCHAMP, J. A Structured conceptual and terminological framework for software process engineering. In: International Conference on the Software Process - Continuous Software Process Improvement,2. **Anais...** 1993.

MADHAVJI, N.H. The process cycle. **IEEE Software Engineering Journal**, v.6, n. 5, p. 234-242, Sept. 1991.

MEREDITH, J. R., MANTEL, S. J. **Project management: a managerial approach**. New York: John Wiley & Sons, 1985.

MILLER, C. Sustaining a continuous improvement culture: from start-up venture to big business in a decentralized culture. In: Software Engineering Process Group (SEPG), 1998, Chicago, USA. **Proceedings...** Chicago, 1998.

MORAES, S. W.; **Um ambiente expert para apoio ao desenvolvimento de software**. 1997. 133p. Dissertação (Mestrado em Informática) – Universidade Federal do Rio Grande do Sul, Porto Alegre,1997.

MÜHLEN Z. M., BECKER J. Workflow process definition language - development and directions of a meta-language for workflow process. In: KnowTech Forum, **Proceedings...** Sept. 1999.

NICHOLAS, J. M. **Managing business and engineering projects**: concepts and implementation. New Jersey: Prentice-Hall, 1990.

OBJECT MANAGEMENT GROUP - OMG. **The unified process model - UPM**. ad/2000-05-05, may 2000.

OLIVEIRA, K.M. **Modelo para construção de ambientes de desenvolvimento orientados a domínio**. 1998. Tese (Doutorado em Engenharia de Sistemas e Computação) – Universidade Federal do Rio de Janeiro, Rio de Janeiro, 1998.

PAULA FILHO, W. P. **Engenharia de software**: fundamentos, métodos e padrões. Rio de Janeiro: LTC, 2000.

PAULK, M. C. Practices of high maturity organizations. In: Software Engineering Process Group (SEPG)11,1999, Atlanta. **Proceedings...** Atlanta, 1999. p. 8-11.

PAULK, M. C.; Curtis, B.; Chrissis, M.B.; Weber, C. V. **Capability maturity modelSM for software, version 1.1**. Pittsburgh: Software Engineering Institute - Carnegie Mellon University, February 1993. CMU/SEI-93-TR-024.

PRESSMAN, R. S. **Engenharia de software**. São Paulo: Makron Books, 1995. 1056 p.

PROJECT MANAGEMENT INSTITUTE - PMI. **A guide to the project management body of knowledge**: PMBOK Guide 2000 edition. Pennsylvania:Project Management Institute, 2000.

RAD, P. F.; RAGHAVAN, A. Establishing an organizational project office. **AACE International Transactions**. Dec. 2000

RATIONAL. **Rational unified process - RUP**. Rational, 2000.

REIS, C. A. L. **APSEE - Uma abordagem baseada em conhecimento para gerência de processos de software evolutivos**. 2001. Proposta de Tese - Universidade Federal do Rio Grande do Sul, Porto Alegre, 2001.

REIS, R. Q. **Uma proposta de suporte ao desenvolvimento cooperativo de software no ambiente PROSOFT**. 1998. 177p. Dissertação (Mestrado em Ciências da Computação) – Universidade Federal do Rio Grande do Sul, Porto Alegre, 1998.

REIS, S. Connecting tools using message passing in the FIELD programming environment. **IEEE Software**, v.7, n. 4, p.276–284, July 1990.

RITCHIE; T. UNIX: a time sharing operating system. **Communications of the ACM**, v.17, n.7, p.365–375, July 1974.

ROUILLER, A. C. **Gerenciamento de projetos de software para empresas de pequeno porte**. 2001. Tese (Doutorado em Ciência da Computação) – Universidade Federal de Pernambuco, Recife, 2001.

ROUT, T. Overview of CMMI: Syst./Sw. Eng. continuos representations, In: Simpósio Internacional de Melhoria de Processo(SIMPROS),4., Recife, **Anais...** Recife: SIMPROS, 2002.

ROUT; T. **Software process assesment - part1**: concepts and introductory guide, version 1, SPICE project, ISO/IEC, 1995.

SANT'ANNA, N.; CEREJA JR, M.G.; BORREGO FILHO, L.F.; LUQUE, L.; CASILLO, B. H. "e-WebProject®" um ambiente integrado para apoio ao desenvolvimento e gestão de projetos de software. In: Conferência internacional de Tecnologia de Software (CITS), Curitiba, **Anais...** Curitiba:CITS, 2002. p. 163-175.

SANT'ANNA N. **Um ambiente experimental para a engenharia de software.** 1993. (INPE-5540-TDI/528). Dissertação (Mestrado em Computação Aplicada) - Instituto nacional de Pesquisas Espaciais, São José dos Campos, 1993.

SANT'ANNA, N. **Um ambiente Integrado para o apoio ao desenvolvimento e gestão de projetos de software para sistemas de controle de satélites.** 2000. (INPE 23567812-3). Tese (Doutorado em Computação Aplicada) - Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2000.

SANT'ANNA, N., NAKANISHI, T., CUNHA, J.B. Aplicação da garantia da qualidade de software ao sistema de controle de satélites. In: Conferência Internacional de Tecnologia de Software: Qualidade de Software,9.,1998, Curitiba, **Anais...**, Curitiba: CITS, 1998, p. 105-116.

SCHLEICHER, A.; Formalizing UML-based process models using graphs transformations, Nagl, Schürr, Münch. In: Applications of Graph Transformations with Industrial Relevance (AGTIVE).1999, Berlin, **Proceedings...** Lecture Notes in Computer Science (LNCS) 1779, Springer-Verlag, 2000.

SCHLEICHER, A.; WESTFECHTEL, B. Beyond stereotyping: metamodeling approaches for the UML. In: Hawaiian International Conference on System Sciences,34., 2001, Los Alamitos, CA. **Proceedings...** IEEE Computer Society Press, 2001.

SCHLEICHER, A.; WESTFECHTEL, B.; JÄGER, D. **Modeling dynamic software process in UML.** Germany, RWTH Aachen, 1998. Technical Report AIB 98-11.

SECRETARIA DE POLÍTICA DE INFORMÁTICA - SPI. **Qualidade e produtividade no setor de software brasileiro.** Brasília: Ministério da Ciência e Tecnologia - MCT, 2002. 258 p.

SECRETARIA DE POLÍTICA DE INFORMÁTICA E AUTOMAÇÃO - SPIA. **Qualidade e produtividade no setor de software brasileiro**. Brasília: Ministério da Ciência e Tecnologia - MCT, 2000. 183 p.

SMITH, D. R.; KOTIK, G. B.; WESTFOLD, S. J. Research on knowledge-based software environments at kestrel institute. **IEEE Transactions on Software Engineering**, v.11, p.1278–1295, Nov. 1985.

SMITH, R. K.; HALE, J. E.; PARRISH, A. S. An empirical study using task assignment patterns to improve the accuracy of software effort estimation. Los Alamitos. **IEEE Transaction on Software Engineering**, v. 27, n. 3, Mar. 2001.

SOMMERVILLE, I. **Software engineering**. 6 ed. [S. I.]: Addison Wesley, 2000.

SOUSA, A. L. R.; REIS, C. A. L.; REIS, R. Q.; PIMENTA, M. S.; NUNES, D. J. Analisando a interação de gerentes e desenvolvedores em ambientes de processo de software: classificação e exemplos. In: Jornadas Chilenas de Computação [I Workshop de Engenharia de Software], 2001, Punta Arenas. **Anais...** 2001.

SPICE SOFTWARE PROCESS IMPROVEMENT AND CAPABILITY DETERMINATION. **Software process assessment part 2: a model for process management version 1.00**, 1995.

TAYLOR, R. N.; BELZ, F. C.; CLARKE, L. A.; OSTERWEIL, L.; SELBY, R.W.; WILEDEN, J. C.; WOLF, A. L.; YOUNG, M. Foundations for the arcadia environment architecture. In: Software Engineering Symposium on Practical Software Development Environments (ACM SIGSOFT/SIGPLAN) **Proceedings...** Boston, Massachusetts: Peter Henderson, 1988. p. 1-13.

TEITELMAN, W.; MASINTER, L. The interlisp programming environment. In: Barstow, D. R.; Shrobe, H. E.; Sandelwall, E. (Org). **Interactive Programming Environments**. [S. I.]: McGraw Hill, 1984. p. 3-18.

TRAVASSOS, G. H., **O Modelo de integração de ferramentas da estação TABA**. 1994. Tese (Doutorado em Engenharia de Sistemas e Computação) – Universidade do Rio de Janeiro, Rio de Janeiro, 1994.

VARGAS, R. V. **Análise de valor agregado em projetos**: revolucionando o gerenciamento de custo e prazo. Rio de Janeiro: Brasport, 2002.

VASCO BRASIL. **Corrente crítica**: gerenciamento de projetos programas ou empreendimentos segundo a teoria das restrições (TOC). São José dos Campos: Phbrasil, 2001.

VASQUES, R. C. Gestão de projetos SW-capability maturity model e capability maturity model integrated. In: CITS - Conferência Internacional de Tecnologia de Software (CITS), 2002, Curitiba, **Anais...** Curitiba: CITS, 2002.

WYSOCKI, R. K.; BECK JR. R.; CRANE, D.B. **Effective project management**: how to plan, manage, and deliver projects on time and within budget. Canada: John Wiley & Sons, 1995.

ZIMMERER, T. W.; YASIN, M. M. A leadership profile of american project managers. **IEEE Engineering Management Review**, v.26, n.4, 1998.

APÊNDICE A

APLICAÇÃO PARA PLANEJAMENTO E ACOMPANHAMENTO DE PROJETOS INTEGRADA AO PSEE E-WEBPROJECT®

Uma vez concluído o estudo e análise das abordagens de processos, a modelagem dos processos e a criação do modelo de informações, foi possível a implementação e integração desses elementos ao PSEE e-WebProject®.

Para apoio, planejamento e monitoramento dos projetos, foi criada uma aplicação para gerenciamento de projetos, pois não foi possível integrar uma ferramenta, já existente, que fosse capaz de usufruir todos os recursos e funcionalidades propostas.

A estrutura da aplicação poderá ser visualizada através da Figura A.1.

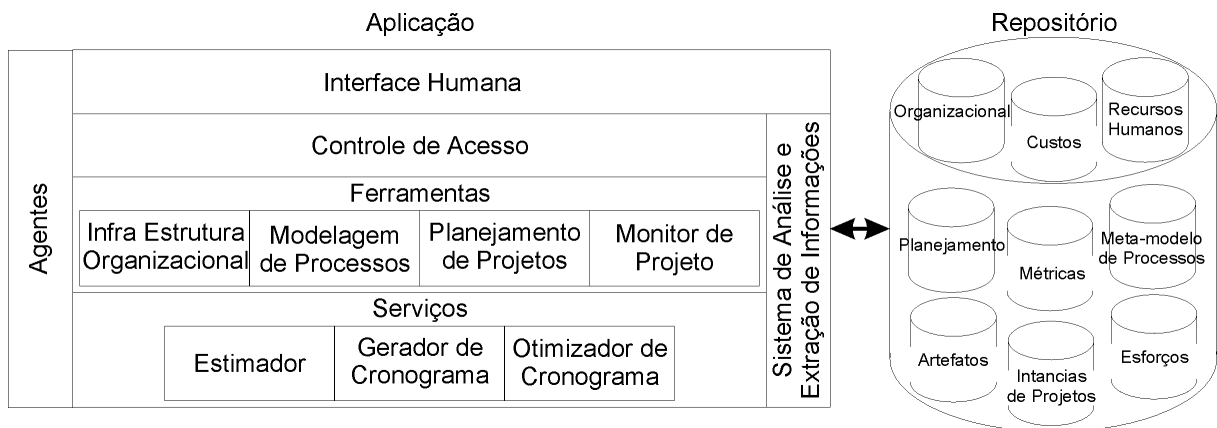


FIGURA A.1 - Arquitetura proposta.

A seguir, serão apresentadas algumas interfaces da aplicação que foram cedidas e desenvolvidas pela empresa SESIS Software.

Ferramenta de Infra-Estrutura Organizacional

Os principais objetivos da ferramenta é permitir a manutenção das informações organizacionais no que tange à infra-estrutura organizacional e recursos.

Na Figura A.2 pode-se observar a interface para manutenção dos recursos humanos do projeto.

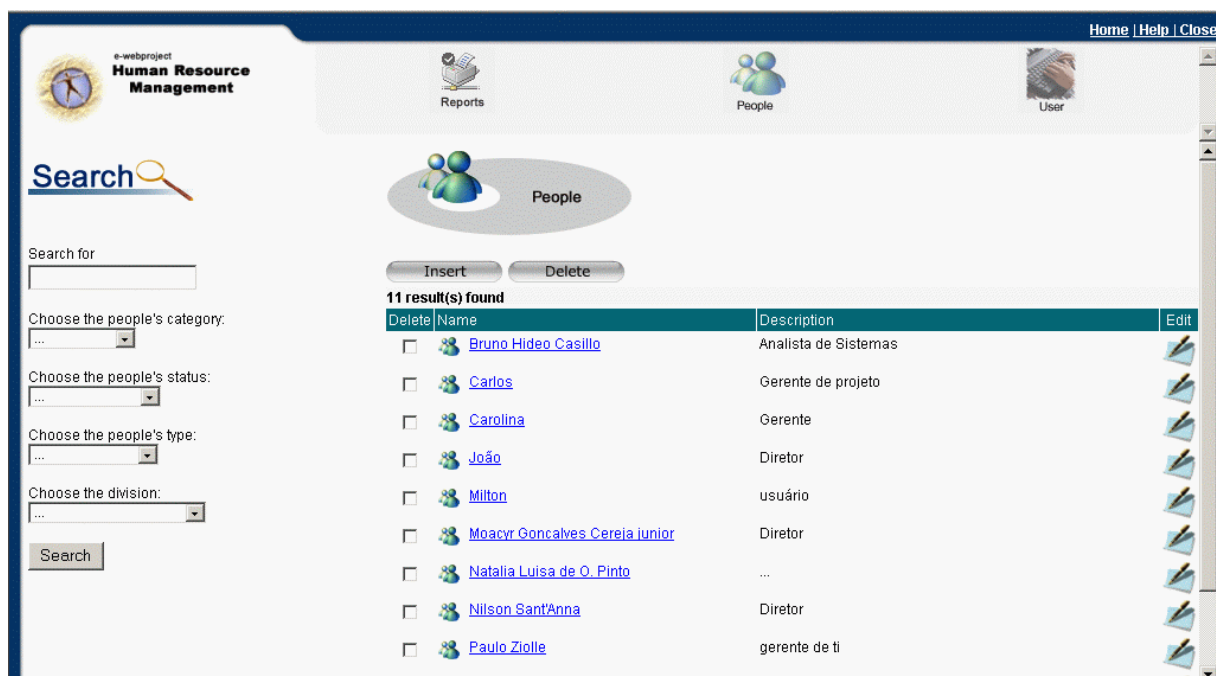


FIGURA A.2 - Manutenção de recursos humanos.

A Figura A.3 mostra a interface para manutenção das equipes da organização e respectiva alocação de pessoas.

Ferramenta para planejamento de projetos

Como outras ferramentas de planejamento de projetos, esta deverá, no mínimo, apoiar a definição e seqüenciamento das tarefas, alocação dos recursos e estabelecimento de cronogramas e *baselines*.

A manutenção dos projetos, que serão gerenciados pela organização pode ser visualizada pela Figura A.4.

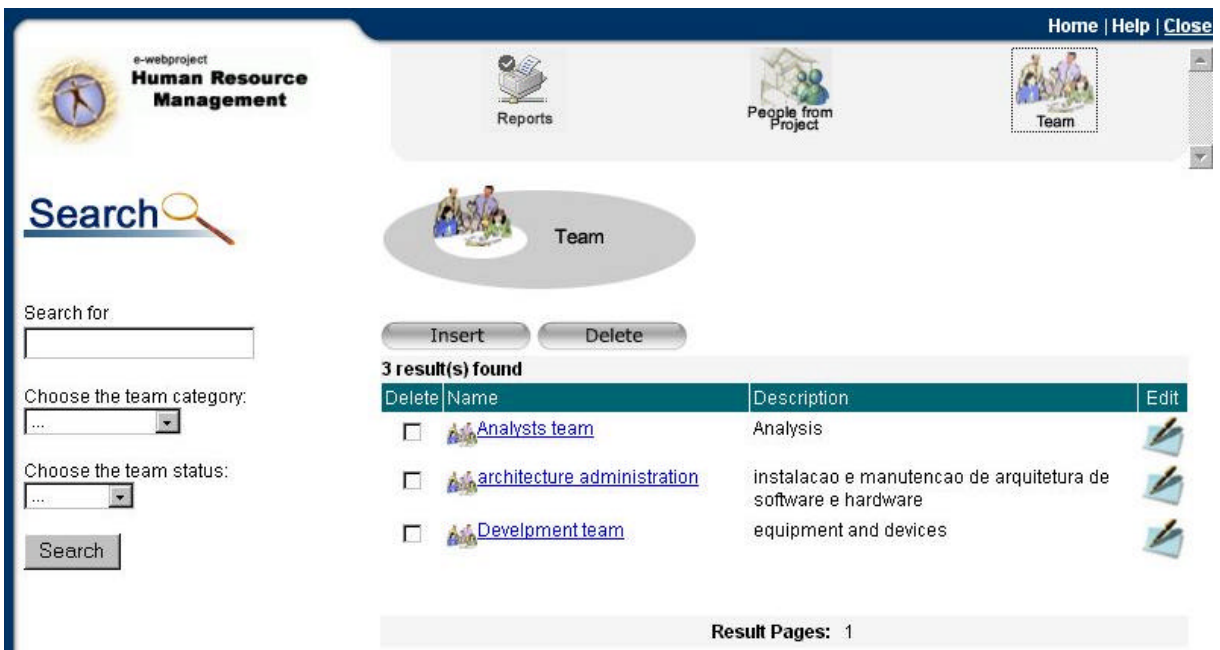


FIGURA A.3 - Manutenção das equipes organizacionais.

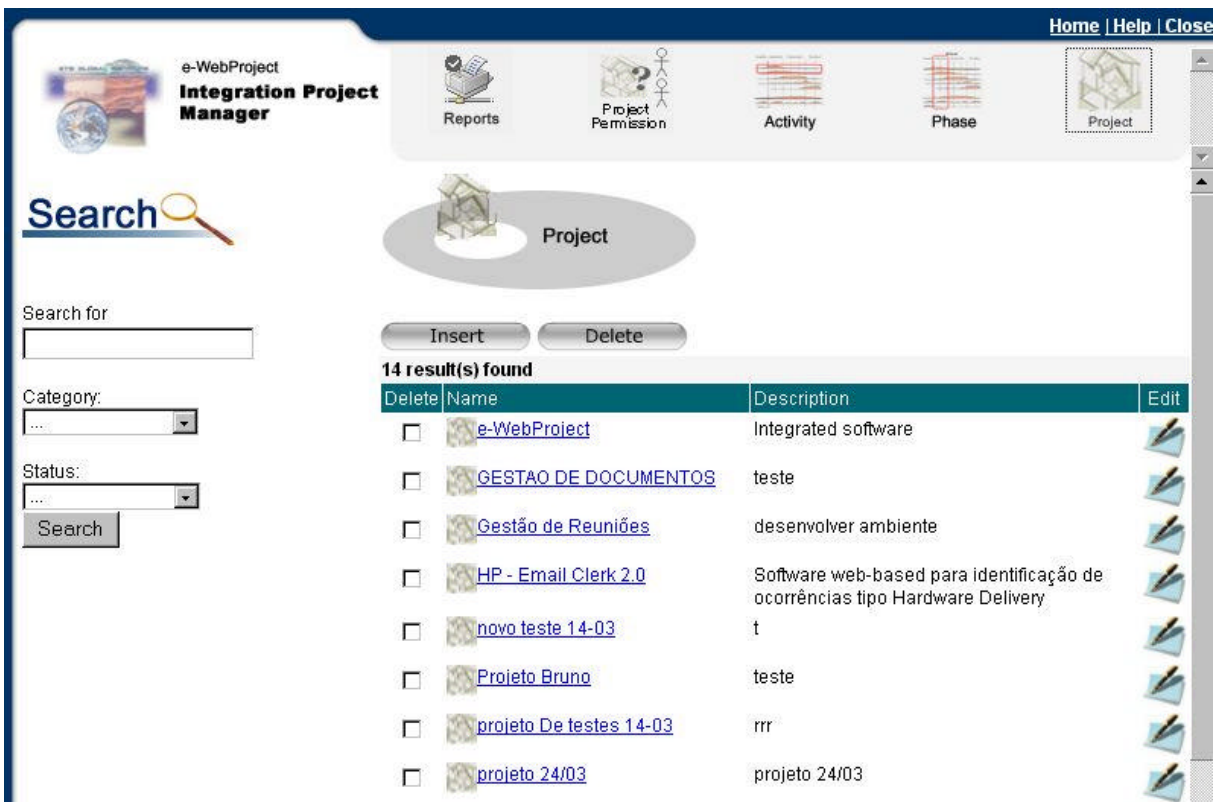


FIGURA A.4 - Manutenção dos projetos.

A Figura A.5 mostra a interface para inserção de uma nova tarefa. Nesta interface poderão ser mantidas as informações de tempo estimado, data de início e fim, instruções para o desenvolvimento, dentre outras.

As outras informações referentes a tarefas poderão ser encontradas nas outras “abas” dessa mesma interface, são elas:

- sub-tarefas mostrada através da Figura A.6;
- relacionamentos e respectivos tipos, *finish to start*, *finish to finish*, *start to start* e *start to finish*, como pode ser observado na Figura A.7;
- os recursos de infra-estrutura necessários para a execução da tarefa, conforme Figura A.8;
- a alocação dos recursos humanos, como mostrado na Figura A.9;
- as *constraints* poderão ser inseridas através da interface mostrada na Figura A.10.

The screenshot displays the 'e-WebProject Project Schedule' web application. At the top, there are navigation links for 'Home', 'Help', and 'Close'. The main interface is divided into several sections:

- Header:** Includes the application logo, 'Project Plan' icon, and 'Reports' icon.
- Form Fields:** 'Name:' (text input), 'Duration: optional' (text input with a 'week' dropdown), and 'Escala Maior/Escala Menor' (dropdowns for 'semana' and 'dia').
- Buttons:** 'Edit', 'Save Plan', 'Insert', and 'Gantt'.
- Project Tree:** A hierarchical list of tasks on the left side, including 'Enterprise Suite', '1 - testar módulos', '2 - testar agenda', '4 - testar project cost', '3 - testar forum', '5 - testar project', '6 - corrigir erros', '14 - comdex2003', '15 - Cadastrar novo projeto', '10 - Fazer modificações na pagina da sesis', '11 - modificar layout', '12 - publicar novas noticias', and '13 - publicar site'.
- Task Plan Form:** A central form for creating a task plan. It includes tabs for 'Plan', 'Sub Tasks', 'Predecessor', 'Resources', 'Allocation', and 'Constraints'. The 'Plan' tab is active, showing fields for:
 - Project:** Enterprise Suite
 - Baseline:** actual
 - Is Based on:**
 - Number:** Type the task plan number. (Input: 1)
 - Name:** Type the task plan name. (Input: testar módulos)
 - Cost: optional:** Type the task plan cost. (Input: 0)
 - Description: optional:** Type the task plan description. (Input:)
 - Creation Date:** (mm/dd/yyyy) Type the task plan creation date. (Input: 08 / 14 / 2003)
 - Start Date:** (mm/dd/yyyy) Type the task plan start date. (Input: 08 / 12 / 2003)
 - Estimated Time: optional:** Type the task plan estimated time. (Input: 7)

FIGURA A.5 - Interface para planejamento do projeto.

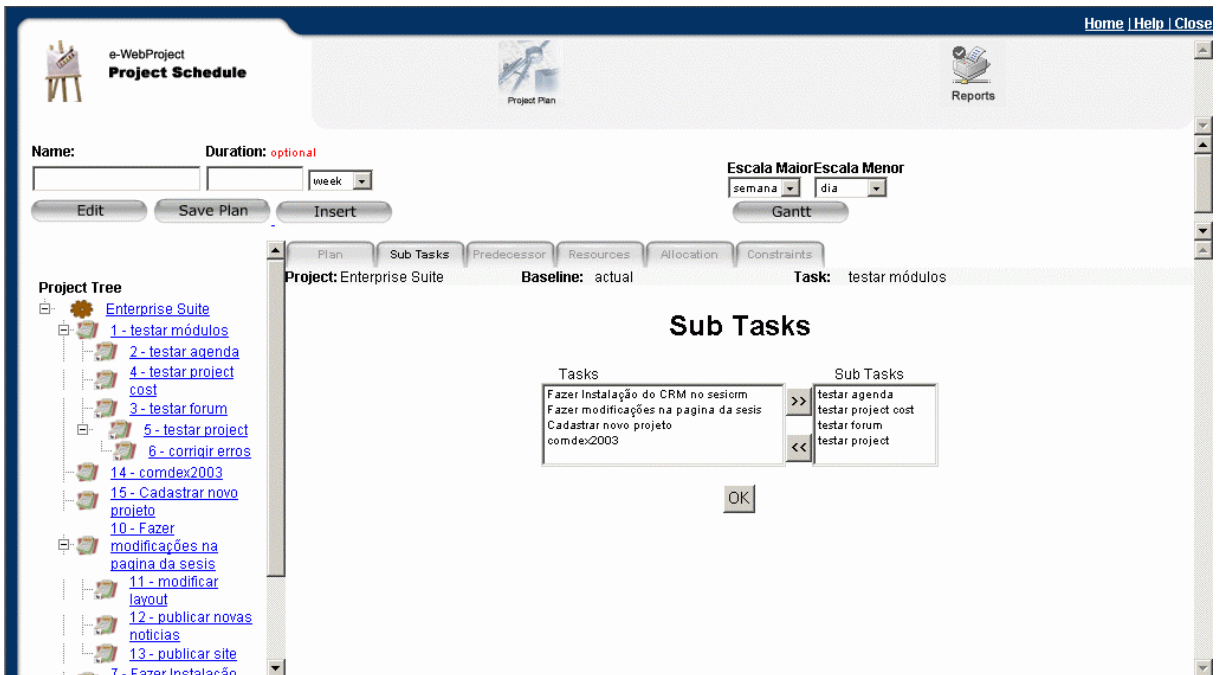


FIGURA A.6 - Manutenção das sub-tarefas.

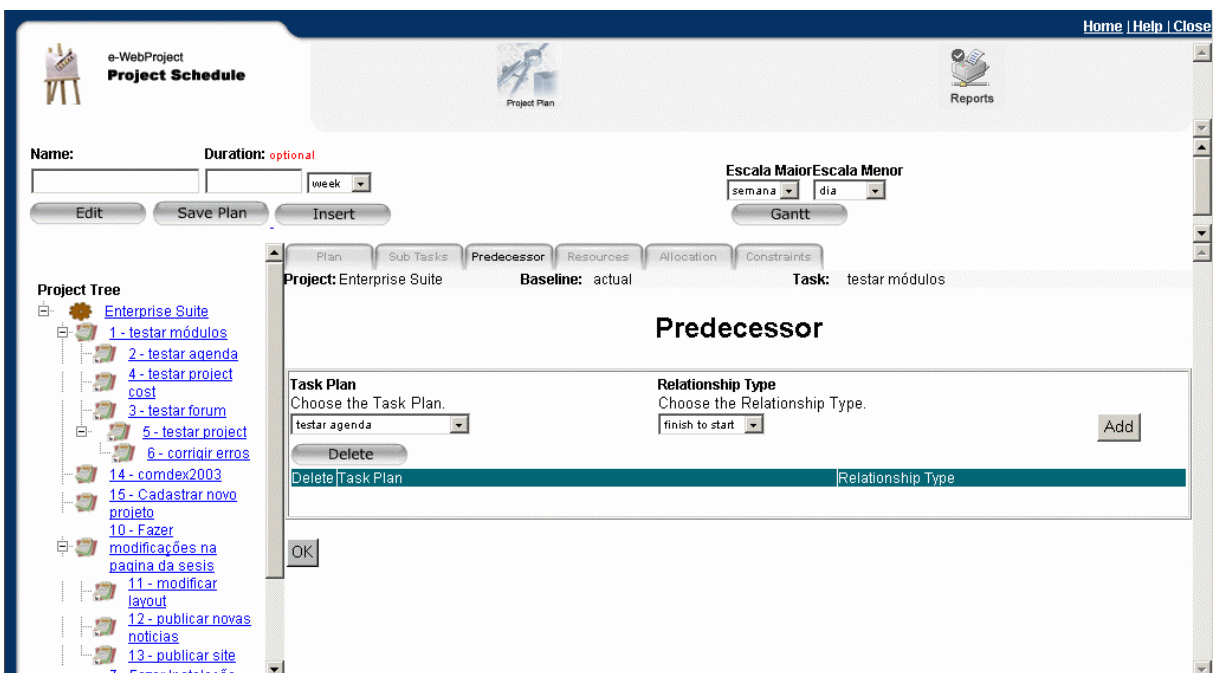


FIGURA A.7 - Manutenção das tarefas predecessoras.

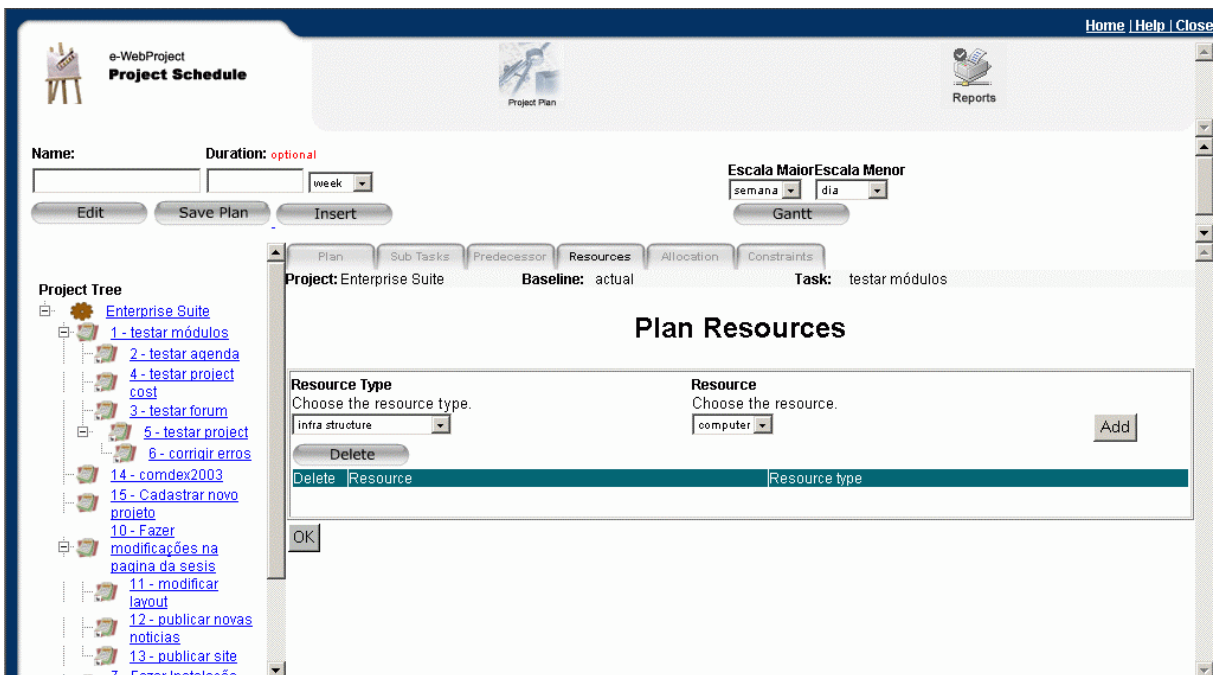


FIGURA A.8 - Alocação dos recursos de infra-estrutura na tarefa.

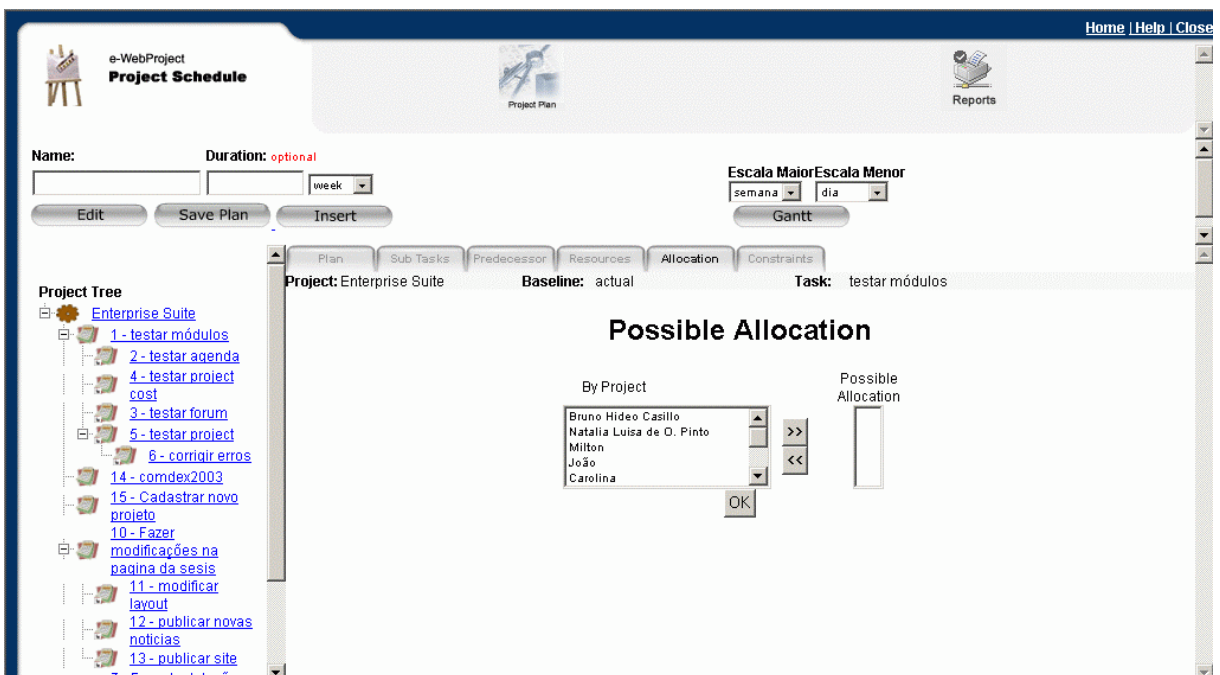


FIGURA A.9 - Alocação dos recursos humanos na tarefa.

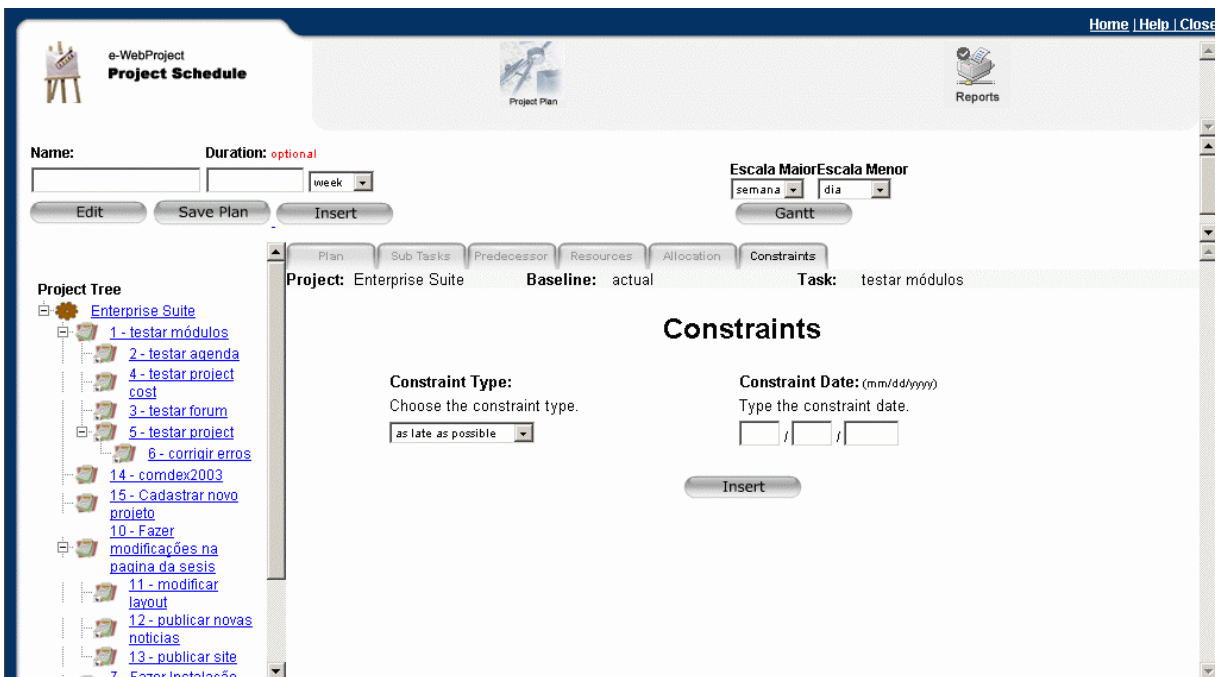


FIGURA A.10 - Manutenção das *constraints* da tarefa.

A visualização do diagrama de *Gantt* pode ser de várias formas, a Figura A.11 e Figura A.12 são exemplos de duas formas possíveis.

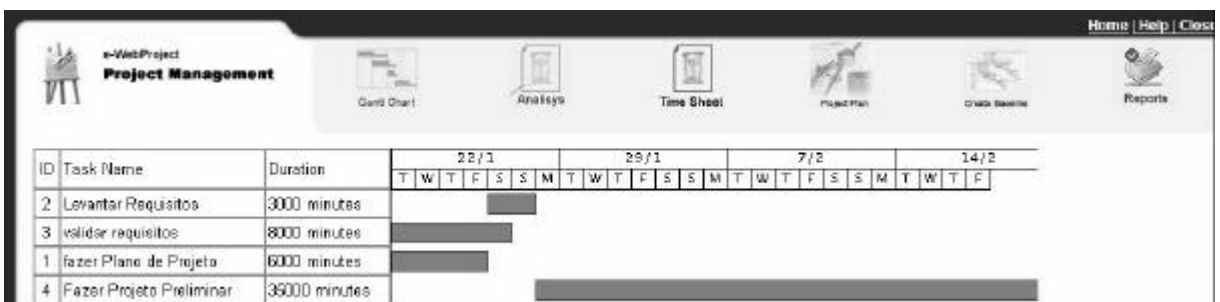


FIGURA A.11 - Diagrama de *Gantt*.

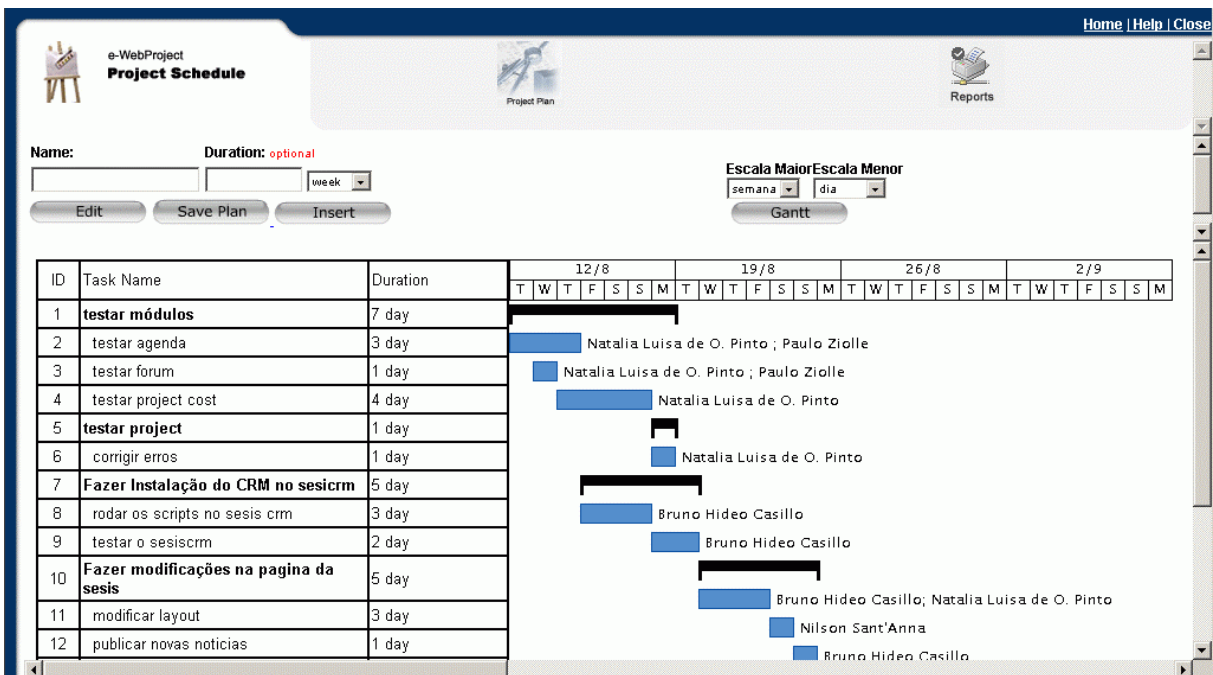


FIGURA A.12 - Diagrama de Gantt com respectiva alocação de pessoas.

Ferramenta para monitoramento de projetos

Essa ferramenta deverá disponibilizar as informações do andamento dos projetos, para que os envolvidos possam acompanhar e monitorar o desenvolvimento dos projetos.

Os tempos das tarefas poderão ser monitorados efetuando a comparação entre o tempo estimado e o tempo realizado de cada tarefa, conforme pode ser visualizado na Figura A.13.

O tempo realizado é extraído da soma dos tempos dos *time-sheets* preenchidos pelas pessoas alocadas às tarefas. A interface individual para preenchimento do *time-sheet* pode ser visualizado na Figura A.14.

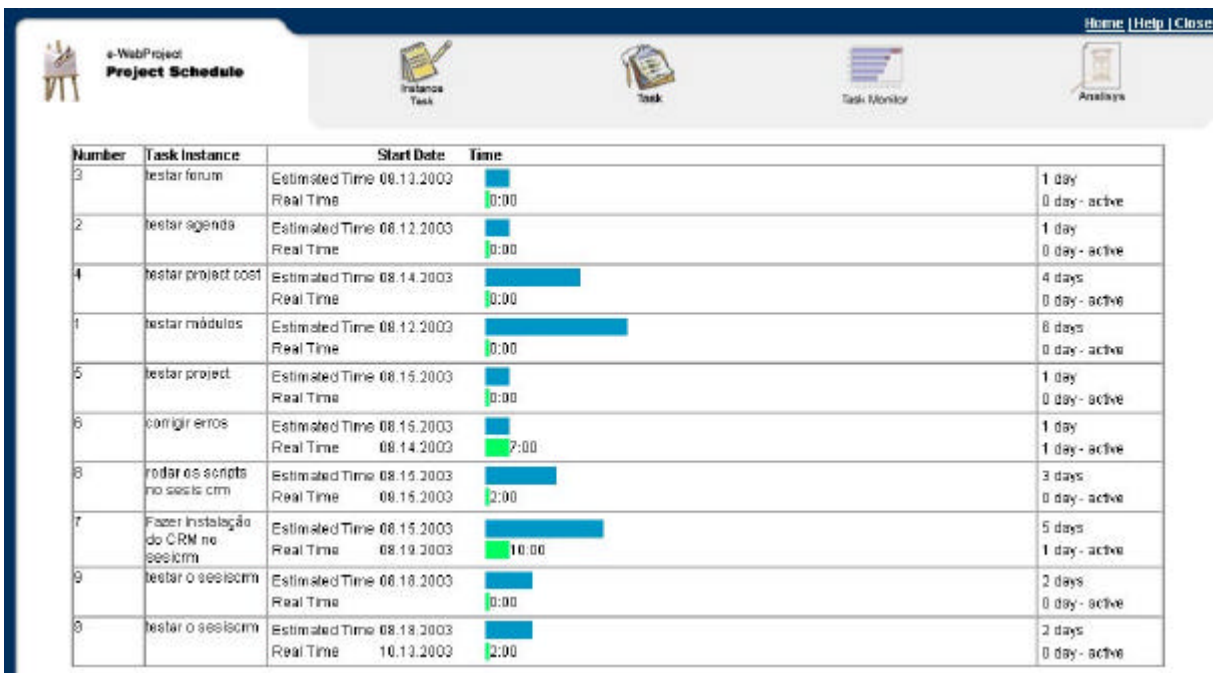


FIGURA A.13 - Comparativo entre o tempo planejado e o realizado.

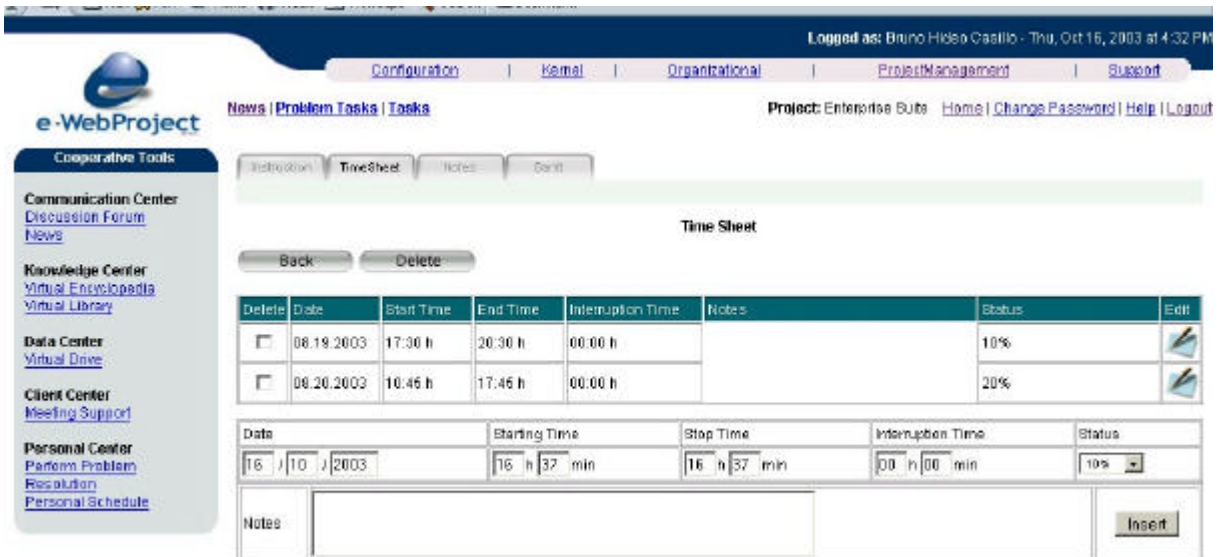


FIGURA A.14 - Time-sheet.

A Figura A.15 mostra o conjunto de opções para a manutenção e controle dos custos do projeto.



FIGURA A.15 - Interface para o controle dos custos.

Serviço de controle de acesso

Por segurança e até mesmo ergonomia, esse serviço deverá proporcionar a adequação da aplicação, e do PSEE, a cada ator. Isso permitirá que cada usuário tenha seu próprio *desktop*, que será montado dinamicamente a partir das permissões, dos projetos em que estiver alocado e das tarefas atribuídas ao usuário.

A interface com todas as opções para a configuração de definição das permissões é mostrada através da Figura A.16.



FIGURA A.16 - Serviço para controle de acesso.

A Figura A.17 mostra o *desktop* pessoal do usuário, o qual foi montado a partir das permissões e tarefas atribuídas a ele.

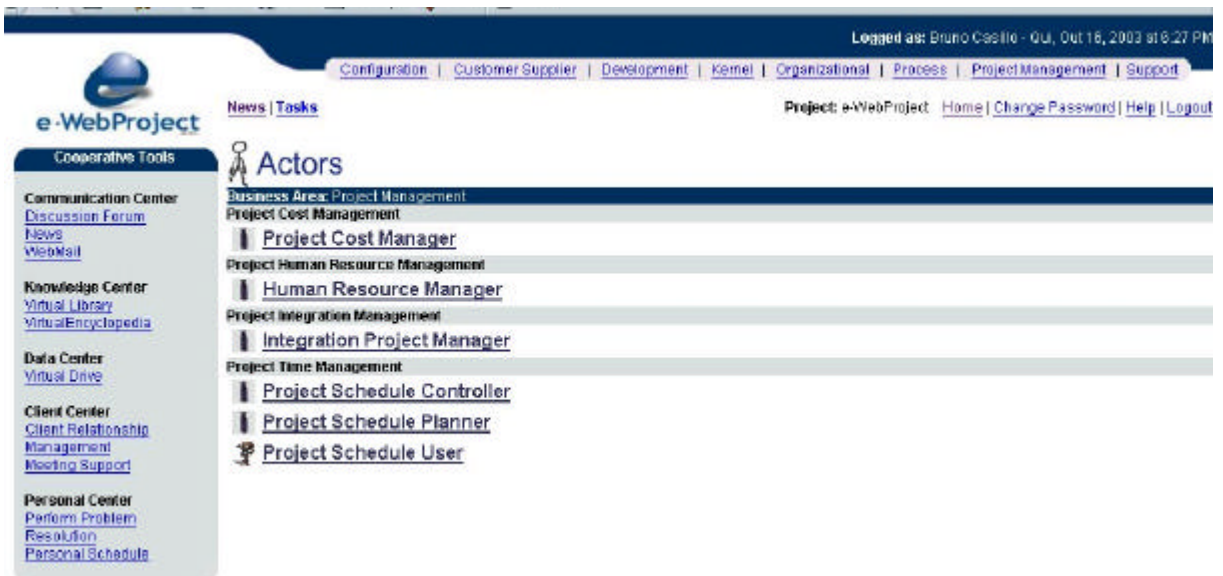


FIGURA A.17 - *Desktop* pessoal do usuário.

A Figura A.18 mostra a lista das tarefas que foram instanciadas ao usuário. A partir dessa interface, o usuário pode ir para a interface de *time-sheet* ou ainda consultar maiores informações da tarefa.

Logged as: Bruno Hideo Castillo - Thu, Oct 16, 2003 at 4:32 PM

Configuration | Kernel | Organizational | ProjectManagement | Support

News | Problem Tasks | Tasks

Project: Enterprise Suite Home | Change Password | Help | Logout

e-WebProject

Cooperative Tools

Communication Center
[Discussion Forum](#)
[News](#)

Knowledge Center
[Virtual Encyclopedia](#)
[Virtual Library](#)

Data Center
[Virtual Drive](#)

Client Center
[Meeting Support](#)

Personal Center
[Perform Problem Resolution](#)
[Personal Schedule](#)

Task List

7 result(s) found

Creation Date	Task	Status
8/15/2003 0:0:0	Fazer instalação do CRM no sesiscrm	active
8/15/2003 0:0:0	corrigir erros	active
8/15/2003 0:0:0	rodar os scripts no sesis crm	active
8/12/2003 0:0:0	testar agenda	active
8/13/2003 0:0:0	testar forum	active
8/18/2003 0:0:0	testar o sesiscrm	active
8/18/2003 0:0:0	testar o sesiscrm	active

Result Pages: 1

FIGURA A.18 - Lista das tarefas instanciadas ao usuário.

Serviço de apoio à geração do cronograma de projeto.

Este serviço deverá propor uma lista possível, ou até mesmo coerente, de alocação de pessoas, baseando-se nas informações referentes à capacitação dos recursos humanos e disponibilidade dos recursos de projeto.

A Figura A.19 mostra a lista de tarefas, que pela ordem lógica do seqüenciamento, já podem ser instanciadas.

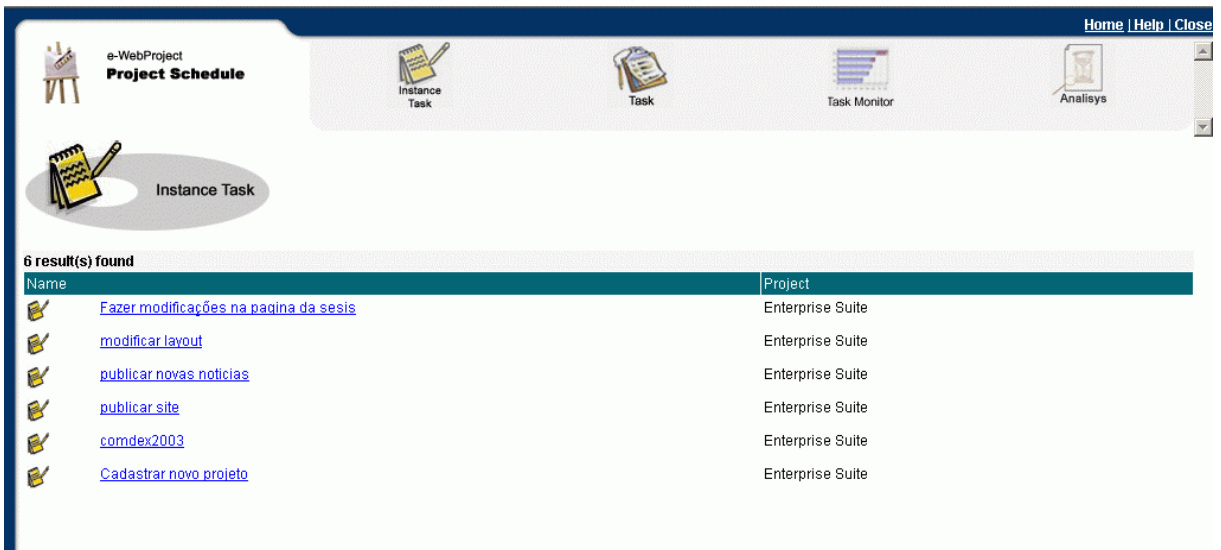


FIGURA A.19 - Interface de tarefas disponíveis para instanciar.

A Figura A.20 mostra a lista de opções para alocação disponibilizada pela aplicação no momento em que o gerente instancia a tarefa.

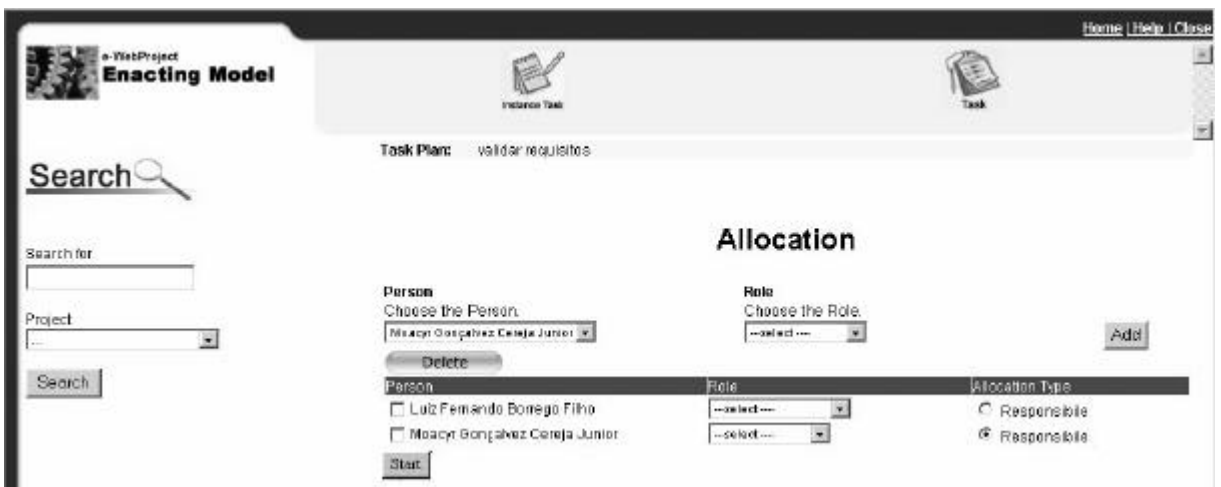


FIGURA A.20 - Interface de parte do processo de geração de cronograma.