

UNIVERSIDADE FEDERAL DE SANTA CATARINA

DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA

RAFAEL REITER DE ALBUQUERQUE

EVOLUÇÃO DA FERRAMENTA DE GERENCIAMENTO DE
PROJETOS DOTPROJECT PARA PLANEJAMENTO DE CUSTO

FLORIANÓPOLIS

2012

EVOLUÇÃO DA FERRAMENTA DE GERENCIAMENTO DE
PROJETOS DOTPROJECT PARA PLANEJAMENTO DE CUSTO

Trabalho de conclusão de curso apresentado ao departamento de Informática e Estatística do curso de graduação da Universidade Federal de Santa Catarina como requisito parcial para a obtenção do grau de Graduado em Ciência da Computação.

Orientador: Prof.Dr.rer.nat.
Christiane Gresse vonWangenheim,
PMP

FLORIANÓPOLIS

2012

Lista de Figuras

- Figura 1 - Relação projeto e partes interessadas
- Figura 2 - Grupos de Processos de gerenciamento de projetos
- Figura 3 – Estrutura do processo de estimar custos fonte
- Figura 4 – Curva-S
- Figura 5 – Estrutura do processo de determinar orçamento
- Figura 6 - Evolução mercado de *Software* 2004 -2010
- Figura 7 – Adicionar atividade dotProject
- Figura 8 – Criação/edição projeto
- Figura 9 – Janela de definição do projeto
- Figura 10 – Janela de definição da tarefa
- Figura 11 – Tela configuração projeto
- Figura 12 – Tela criação de atividade
- Figura 13 – Relatório detalhado do projeto
- Figura 14 – Criação de uma nova atividade
- Figura 15 - Cálculos custos do projeto
- Figura 16 – Processo genérico para planejamento de custos
- Figura 17 – Modelagem da solução
- Figura 18 – Arquitetura da ferramenta dotProject com módulo Custos
- Figura 19 – Dependência entre arquivos do projeto
- Figura 20 – Modelagem entidade relacional banco de dados
- Figura 21 – Criar estimativas de custo
- Figura 22 – Adicionar estimativas de RH
- Figura 23 – Estimar recursos não humanos
- Figura 24 – Modificar recursos humanos e não humanos
- Figura 25 – Criar *baseline*
- Figura 26 – Estimar custo dos riscos
- Figura 27 – Visualizar *baseline*/orçamento
- Figura 28 – Estimar reserva gerencial
- Figura 29 – Análise dados objetivo 1
- Figura 30 – Análise dados objetivo 2

Lista de Tabelas

Tabela 1 - Mapeamento de grupos de processos de gerenciamento de projetos e áreas de conhecimento

Tabela 2 – Exemplo Custos Diretos

Tabela 3 – Exemplo Custos Indiretos

Tabela 4 – Exemplo estimativas de custos

Tabela 5 – Exemplo *baseline* de custos

Tabela 6 – Avaliação dos critérios.

Tabela 7 – Resultado avaliação ferramentas

Tabela 8 – Detalhamento tarefa estimar os custos humanos

Tabela 9 – Detalhamento tarefa estimar os custos não humanos

Tabela 10 – Detalhamento tarefa desenvolver orçamento

Tabela 11 – Plano de testes do sistema

Tabela 12 – Questões para avaliação

LISTA DE ABREVIATURAS E SIGLAS

MPEs – Micro e Pequenas Empresas

GP – Gerenciamento de Projetos

CMMI - *Capability Maturity Model Integration*

CMMI-DEV - *Capability Maturity Model Integration for Development*

PMBOK - *Project Management Body of Knowledge*

SEBRAE – Serviço Brasileiro De Apoio As Micro E Pequenas Empresas

RESUMO

Uma das necessidades contemporâneas das empresas de *software* no mercado hoje, certamente é o gerenciamento de *software*. Tal gerenciamento permite alcançar metas e resultados de forma a produzir maior qualidade para os projetos e produtos desenvolvidos. O Project Management Institute (PMI) nos sugere boas práticas de gerência de projetos para tentar melhorar o desempenho de projetos de *software*. O guia PMBOK recomenda 5 grupos de processos e 9 áreas de conhecimento para descrever de forma organizada o trabalho a ser feito dentro do projeto. Dificilmente é encontrada uma ferramenta que forneça funcionalidades baseadas nas melhores práticas de gerenciamento de projetos, e que permitam gerar os artefatos necessários para realizar um projeto. Neste contexto, este trabalho visa evoluir o planejamento de custos dentro da ferramenta de gerenciamento de projetos dotProject de forma a se adequar as boas práticas de gerenciamento de projetos de *software* recomendadas. O estudo das boas práticas de gerenciamento sugeridas pelo PMBOK e CMMI serão utilizadas para a implementação e documentação das funcionalidades inseridas na ferramenta. Este trabalho é realizado por meio de pesquisas na literatura existente sobre o assunto, para captar as práticas mais adequadas para micro e pequenas empresas. Engloba também um estudo acerca do estado da arte e do suporte referente aos processos de planejamento de custos por meio de ferramentas de gerenciamento de projetos. Espera-se conseguir modelar e construir da melhor forma as funcionalidades para a ferramenta tendo em vista que a ferramenta é um suporte a gerenciamento de projetos para micro e pequenas empresas.

SUMÁRIO

RESUMO	6
1. INTRODUÇÃO	9
1.1 Problema	11
1.2 Objetivos	12
1.2.1 <i>Objetivo geral</i>	12
1.2.2 <i>Objetivos específicos</i>	12
1.3 Limites	13
1.4 Método de Pesquisa	13
1.5 Estruturas do trabalho	15
2. FUNDAMENTAÇÃO TEÓRICA	15
2.1 Gerenciamento de Projetos	16
2.2 Planejamento de Custos	21
2.2.1 Estimar os Custos	24
2.2.1.2 Determinar Orçamento	33
2.3 Micro e Pequenas Empresas	40
2.4 dotProject	42
3. ESTADO DA ARTE	44
3.1 Definição do estudo	44
3.2 Avaliação das Ferramentas	46
3.2.1 <i>dotProject</i>	46
3.2.2 <i>Project.net</i>	48
3.2.3 <i>PhpCollab</i>	49
3.2.4 <i>Track+</i>	52
3.2.5 <i>Streber</i>	55
4. MODELAGEM DE UM PROCESSO GENÉRICO PARA PLANEJAMENTO DE CUSTOS	58
5. EVOLUÇÃO DO DOTPROJECT	63
5.1 Análise de Requisitos	63
5.1.1 <i>Requisitos Funcionais</i>	63
5.1.2 <i>Requisitos Não Funcionais</i>	64
5.1.3 <i>Casos de Uso</i>	64
5.2 Modelagem da solução	68
5.2.1 <i>Modelagem da implementação</i>	69
5.3 Implementação	71
5.4 Testes	76
5.5 Publicação	78
6. AVALIAÇÃO	79
6.1 Avaliação por painel de especialistas	79
6.1.1 <i>Execução da avaliação</i>	80
6.1.2 <i>Análise dos dados</i>	81
6.2 Avaliação em relação ao alinhamento ao PMBOK	82
6.3 Discussão	84
7. CONCLUSÃO	84
REFERÊNCIAS	86

APÊNDICE A – FORMULÁRIO DE AVALIAÇÃO DO MÓDULO DE CUSTOS POR PAINEL DE ESPECIALISTAS	89
APÊNDICE B – CÓDIGO FONTE MÓDULO DE CUSTOS.....	92
APÊNDICE C – ARTIGO ACADÊMICO	189

1. Introdução

Atualmente mudanças no ambiente empresarial e tecnológico têm feito cada vez mais empresas adotarem o gerenciamento de projetos no seu cotidiano. A necessidade de rapidez, competitividade, clientes exigentes e lucros cada vez mais difíceis de obter, caracterizaram um cenário ideal para que o gerenciamento e execução de projetos viessem a se encaixar dentro do orçamento e tempo das empresas. No Brasil no ano de 2010 o setor de TI mostrou um crescimento de 21,3% sendo que os setores de *software* e serviços cresceram 24%, constando que cerca de 6,74 bilhões de dólares foram movimentados na área de *software*, o que representou perto de 2,2% do mercado mundial (ABES, 2011).

O bom gerenciamento de projetos seja ele para o lançamento de um novo produto, desenvolvimento de uma nova tecnologia ou até mesmo a readequação de um processo interno produtivo, tornou-se um diferencial competitivo no mercado, mas também uma questão de sobrevivência (RHPortal, 2011). Muitos dos projetos executados nas empresas do mercado hoje em dia apresentam problemas de não cumprimento de prazos, ocasionando prejuízos financeiros e problemas mais sérios relacionado a perda de credibilidade para com o cliente, desmotivação da equipe, entre outros (PMI-RJ, 2010). Perante esta situação muitas destas empresas mostram-se dispostas a investir no desenvolvimento e implementação de metodologias de gerenciamento de projetos (PMI-RJ, 2010). “O gerenciamento de projetos é a aplicação de conhecimentos, habilidades, ferramentas e técnicas às atividades do projeto a fim de atender aos seus requisitos” (PMI, 2008).

Cada projeto contém características e exigências diferentes que podem ou não mudar de acordo com exigência de seus responsáveis.

*“Não temos o luxo de determinar nossos cronogramas.
Eles são nos dito. Então, dado o prazo, nós tentamos
dizer ao cliente o que podemos construir. No final,
geralmente trabalhamos muitas horas extra, porque
eles querem tudo.”*

Gerente de desenvolvimento empresa na Wall
Street.

O cliente tem o hábito de exigir o máximo da equipe responsável pelo projeto atrás de resultados, buscando um projeto que seja rápido, eficiente (dentro do escopo) e de baixo custo (PMI, 2010). Porém esta meta não é de fácil alcance tendo em vista que um projeto muda constantemente e depende muito de seus requisitos (PMI, 2010). Essa meta chamada de tripla restrição (tempo, custo, escopo), se caracteriza por apenas ser a base para a definição do sucesso de um projeto.

Para concluir um projeto com sucesso deve-se dar ênfase a tripla restrição – escopo, tempo e custo, visando a entrega do(s) resultado(s) de um projeto de acordo com o escopo, prazo e orçamento definidos, na qualidade prevista (PMI, 2008). Dentro da tripla restrição, um dos principais determinantes para o sucesso de um projeto é o gerenciamento de custos. Para as pequenas empresas é fundamental o controle de custo já que geralmente seus orçamentos são bastante limitados.

Hoje em dia a busca pelo sucesso em um projeto não depende apenas desses três fatores e sim do complemento destes. Para que um projeto inicie e termine de forma consistente devemos considerar questões importantes para as empresas, principalmente falando de MPEs. A construção inicial de um projeto se caracteriza por definir questões gerenciais (Kerzner, 2004), ou seja, esclarecer alguns tópicos que se caracterizam como restrições do projeto, sendo eles: tempo alocado para o projeto, orçamento disponibilizado, a aceitação do cliente/usuário, a quantidade de mudanças no escopo (buscando sempre o mínimo), a distribuição ou definição do fluxo de trabalho da empresa no projeto (papéis, equipes, etc), entre outros (Kerzner, 2004).

O escopo deste trabalho estará envolvido com os custos estimados em um projeto. O gerenciamento dos custos do projeto inclui o planejamento, monitoramento e controle de custos. Já o planejamento de custos envolve os processos de estimar os custos e determinar o orçamento (PMI, 2008).

Estimar os custos é um processo de desenvolvimento de uma estimativa dos recursos monetários necessários para executar as atividades do projeto (PMI, 2008). A estimativa de custo provê algumas fontes de entrada de informações que por sua vez são derivadas das saídas de outros processos do projeto. Para a estimativa de custo os artefatos mais importantes

de entrada são: *baseline* do escopo, cronograma do projeto, plano de recursos humanos e registro de riscos (PMI, 2008). Porém agora que temos todas as informações de entrada necessárias, como fazer para estimar. Para isso existem várias técnicas e ferramentas aonde podemos citar algumas técnicas, como: opinião especializada, estimativa análoga, estimativa paramétrica e estimativa *bottom-up* (PMI, 2008). O principal resultado desse processo são as estimativas de custos (por item/atividade do projeto) e agregação da estimativa total de custo do projeto.

O outro processo envolvido no planejamento de custo é determinar o orçamento. Este processo é resultado da agregação dos custos das atividades individuais ou pacotes de trabalho de forma a construir uma *baseline* (linha de base) dos custos autorizados, incluindo a reserva de contingência (PMI, 2008). Determinar o orçamento requer a entrada de informações de outros processos do projeto, assim como na estimativa de custos. Algumas das entradas podem ser citadas a seguir como sendo: estimativas de custos da atividade, bases de estimativas e *baseline* do escopo (PMI, 2008). Todo esse processo gera a *baseline* do desempenho de custos, que é definida pela distribuição dos custos do projeto ao longo de seu calendário.. No final acrescentando-se uma reserva gerencial ao valor total da *baseline* determina-se o orçamento do projeto (PMI, 2008).

1.1 Problema

Hoje na prática para gerenciar projetos necessita-se de uma ferramenta de suporte (PMI, 2010), por isso existem várias ferramentas de gerenciamento de *software* no mercado que auxiliam na gerência de projetos. Algumas ferramentas comerciais como MS Project (www.microsoft.com/project), o Oracle Primavera Systems (www.oracle.com/primavera), e de *software* livre como dotProject (www.dotproject.net), OpenProj (openproj.org/openproj) (PMI, 2010), são exemplos. Tipicamente essas ferramentas suportam o planejamento de tempo e monitoria de projetos.

Porém, analisando o grau de suporte oferecido principalmente em relação a modelos de melhores práticas como p.ex. PMBOK ou partes do CMMI-DEV voltado a gerência de

projetos, pode-se observar que essas ferramentas em geral não fornecem um suporte completo (GRESSE et al, 2009).

1.2Objetivos

1.2.1 Objetivo geral

Este trabalho tem como objetivo geral a evolução de um suporte na ferramenta de gerenciamento de projetos dotProject, voltado ao planejamento de custos de projetos alinhado ao PMBOK 4ed..

1.2.2 Objetivos específicos

Os objetivos específicos deste trabalho são:

- O1. Analisar a teoria referente ao planejamento de custos de projetos, no contexto de MPEs, e analisar a ferramenta dotProject;
- O2. Analisar o estado da arte de suporte disponível para o planejamento de custos de projetos pelas principais ferramentas free e open-source;
- O3. Modelar um processo genérico para o planejamento de custo no contexto de micro e pequenas empresas alinhadas ao PMBOK.
- O4. Desenvolver um suporte para o planejamento de custos de projetos na ferramenta dotProject com base no processo modelado;
- O5. Avaliar o processo e o suporte implementado no dotProject por meio de painéis de especialistas.

1.3 Limites

Este trabalho possui as seguintes limitações:

- L1. Este trabalho é voltado somente para o planejamento de custos, tendo que outros grupos de processo e áreas de conhecimento estão fora do escopo do presente trabalho;
- L2. Será desenvolvido um suporte somente na ferramenta de gerenciamento de projetos dotProject, não sendo consideradas outras ferramentas;
- L3. Este trabalho deve estar alinhado ao PMBOK, não considerando outros modelos e/ou normas na área de gerenciamento de projetos;
- L4. Este trabalho será focado em micro e pequenas empresas, sem considerar qualquer outro tipo de empresa.

1.4 Método de Pesquisa

A metodologia utilizada para desenvolver este trabalho envolve estudo da literatura no contexto geral e específico, desenvolvimento e avaliação da implementação realizada na ferramenta de gerenciamento de projeto dotProject.

Na primeira fase do trabalho (referente ao objetivo O1), a compreensão e entendimento do processo de gerência de projetos é analisado. A partir daí serão definidos todos os conceitos e terminologia relacionada ao planejamento de custos.

A ferramenta dotProject é analisada. Também são analisadas as necessidades e características de micro e pequenas empresas principalmente voltado ao planejamento de custos.

A1.1 Análise da GP e principalmente planejamento de custos

A1.2 Análise do dotProject

A1.3 Análise das características de MPEs

Na segunda etapa é realizada uma revisão sistemática do suporte fornecido pelas principais ferramentas de gerenciamento de projetos free e open-source. Essa então envolve a seleção sistemática das cinco principais ferramentas, a definição de indicadores a serem avaliados com base no PMBOK, a execução da análise das ferramentas e a análise e discussão dos resultados.

A2.1 Revisão da seleção das ferramentas e definição dos indicadores unificados

A2.2 Execução da análise

A2.3 Documentação dos resultados e discussão

Na terceira etapa é modelado um processo genérico de planejamento de custo voltado ao contexto de MPEs. A modelagem é baseada em técnicas de modelagem de processo (ACUÑA,2000) definindo as etapas, sequência e para cada etapa os objetivos envolvidos, atividades e os produtos de entrada e saída.

A3.1 Modelagem do processo

Na etapa de desenvolvimento do suporte a ferramenta dotProject é usado o framework de implementação do dotProject, um suporte com base no processo modelado na etapa anterior. Todo o processo seguirá as fases da engenharia de *software*, contendo análise de requisitos, projeto/modelagem, implementação e testes.

A4.1 Análise de requisitos

A4.2 Modelagem

A4.3 Implementação

A4.4 Testes

Por fim a fase de avaliação do processo e suporte da ferramenta implementada é realizado uma avaliação principalmente da utilidade e usabilidade por um painel de especialistas. É definido o objetivo e as medidas da avaliação, projetando o design da avaliação (incluindo a seleção de especialistas). Será realizada a avaliação coletando os dados conforme previsto. Todos os dados coletados serão analisados e discutidos.

É realizada também uma análise teórica do grau de atendimento do suporte fornecido em alinhamento com o PMBOK.

A5.1 Realização do *expert panel*

A5.1.1 Definição da avaliação

A5.1.2 Execução da avaliação

A5.1.3 Análise dos dados

A5.2 Comparação do suporte em alinhamento ao PMBOK

1.5 Estruturas do trabalho

O trabalho está dividido em 6 capítulos: Introdução, Fundamentação Teórica, Estado da Arte & Prática, Modelo de Processo Genérico, Evolução do Project e Avaliação. No segundo capítulo, Fundamentação Teórica, está agrupada os conhecimentos teóricos básicos necessários para o desenvolvimento do trabalho. Também, estarão detalhados os estudos realizados sobre o contexto empresarial brasileiro e os conceitos de gerenciamento de projetos, mais especificamente do grupo de encerramento.

O próximo capítulo, Estado da Arte & Prática, apresenta um estudo comparativo das 5 principais ferramentas de gerenciamento de projetos, a fim de identificar qualidade e deficiências de cada uma, quando utilizadas no cenário de micro e pequenas empresas. O capítulo de construção de modelo genérico apresenta o modelo a ser usado como parâmetro para evoluir uma ferramenta de gerenciamento de projetos.

Por fim os capítulos de evolução e avaliação da ferramenta, apresentam o detalhamento da implementação, assim como os testes de validação e as avaliações do módulo, em termos de utilização feita por gerentes de projeto.

2. Fundamentação Teórica

Neste capítulo primeiramente são apresentados os conceitos básicos do gerenciamento de projetos. Em seguida serão apresentados os conceitos e características mais detalhado do grupo de processo de planejamento com enfoque na área de conhecimento de custos, e por fim características da MPEs e detalhes da ferramenta de gerenciamento de projetos dotProject.

2.1 Gerenciamento de Projetos

Gerenciamento de projetos segundo o PMBOK (2008) é aplicação de conhecimento, habilidades, ferramentas e técnicas as atividades do projeto a fim de atender aos seus requisitos. Para se conduzir um bom gerenciamento de projeto é necessário saber realmente como se caracteriza um projeto, para fim de satisfazer seus requisitos.

Um projeto pode ser considerado uma série de atividades e tarefas que possuem objetivos específicos, data de início e fim definidas, consumo de recursos humanos e não humanos(i.e dinheiro, equipamento, pessoas) e limite de recursos para execução (ou fundos) (Kerzner,2004). Cada projeto finalizado irá resultar em um produto, serviço ou resultado que poderá ser impactante na sociedade, economia e até mesmo no meio ambiente, levando em conta que a maioria dos projetos é realizado para criar um resultado duradouro (PMI,2008). O nível de benefícios que as organizações veem recebendo com a adequação de seus projetos, ou seja, da utilização do gerenciamento de projetos alinhados com as boas práticas de execução do mercado, está relacionado a sua maturidade em relação a metodologia e conceitos perdurados.

Conforme estudo de benchmarking realizado em gerenciamento de projetos no Brasil pelo *Project Management Institute* (PMI) em 2009, existem vários benefícios que o gerenciamento de projetos fornece para as empresas. Os principais benefícios identificados são aumento do comprometimento com objetivos e resultados, disponibilidade de informação para a tomada de decisão, melhoria de qualidade nos resultados dos projetos, aumento da integração entre áreas, aumento da satisfação do cliente (interno/externo), entre muitos outros (PMI, 2008).

O gerenciamento de projetos compõe-se de várias componentes importantes que juntas propõe o projeto. Uma parte inicial importante dentro do projeto são as partes interessadas, que segundo PMBOK (2008) são pessoas e organizações ativamente envolvidas no projeto ou cujos interesses podem ser afetados positivamente ou negativamente pela execução ou término do projeto. A figura 2 ilustra a relação do projeto com suas partes interessadas.

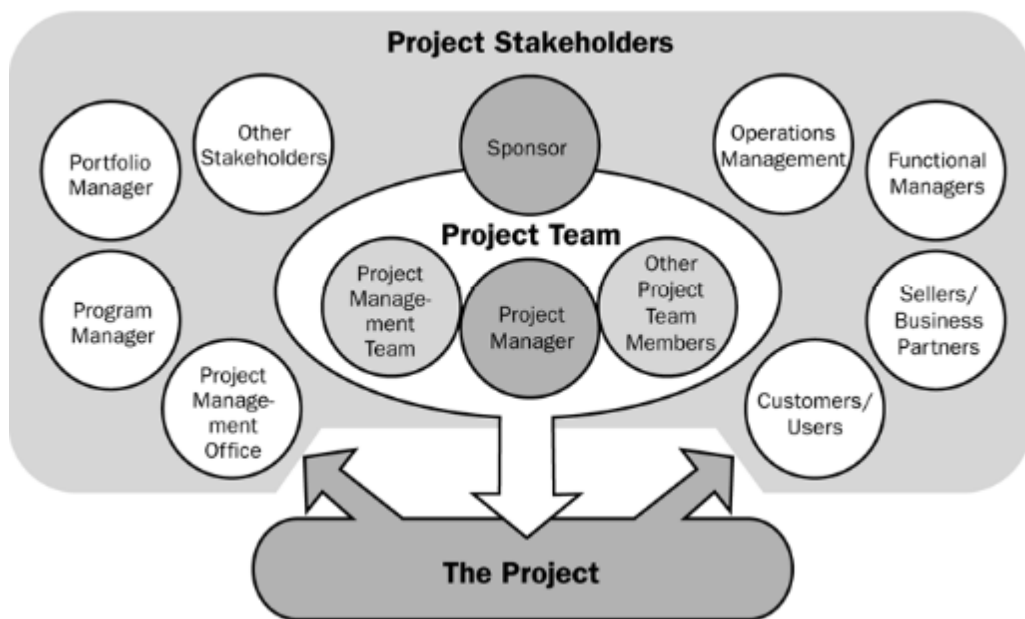


Figura 1 - Relação projeto e partes interessadas (PMI, 2008).

O sucesso de um gerenciamento de projeto inclui gerenciar de forma ativa os requisitos, exigências e mudanças de seus patrocinadores, clientes ou de outras partes interessadas. O gerenciamento de projetos é realizado através de 5 grupos de processos, onde cada processo é caracterizado por suas entradas, ferramentas, técnicas e saídas (PMI, 2008).

Cada processo ou grupo de processo necessita de entradas para que no final de seu processamento seja produzida uma saída consistente. Segundo XAVIER (2009), as entradas de um processo são as informações ou documentos que sofrerão ou gerarão algum tipo de ação ao longo do processo. As ferramentas e técnicas são os mecanismos aplicados às entradas para a geração de saídas (XAVIER, 2009). E as saídas são os documentos ou itens documentáveis que são o resultado do processo (XAVIER, 2009).

Os 5 grupos de processos se caracterizam como a seguir (PMI, 2008):

- **Inicialização** – São os processos realizados para definir um novo projeto ou uma nova fase de um projeto através da obtenção de autorização para iniciar;
- **Planejamento** – Os processos realizados para definir o escopo do projeto, melhorar o detalhamento dos objetivos e definir o caminho e ações a serem seguidas para chegar aos objetivos do projeto;
- **Execução** – São os processos realizados para executar todo o trabalho definido anteriormente no plano de gerenciamento a fim de satisfazer todas as especificações do projeto;
- **Monitoramento e Controle** – Processos para acompanhar, revisar e regular o progresso e o desempenho do projeto, identificar todas as áreas nas quais serão necessárias mudanças no plano e iniciar as mudanças correspondentes.
- **Encerramento** - Os processos realizados para finalizar todas as atividades de todos os grupos de processos, encerrando o projeto ou fase formalmente.

Em geral um processo torna-se entrada para o outro processo, ou então é uma entrega do projeto (PMI, 2008). Apesar de serem distintos, os processos interagem e se sobrepõem ao longo do projeto, e, além disso, são iterativos e podem ser repetidos (PMI, 2008). A figura abaixo demonstra a estrutura de integração dos processos, sendo eles iniciados pelo grupo de processo de iniciação e terminando com o grupo de processo de encerramento.

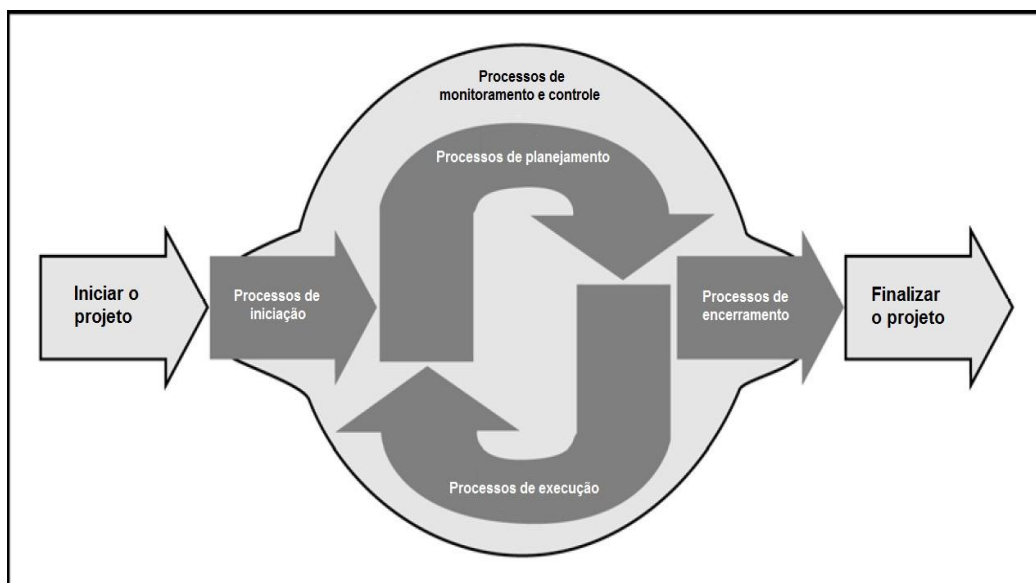


Figura 2 - Grupos de Processos de gerenciamento de projetos Fonte: PMI, 2008.

Considerando cada processo como uma entidade responsável pela execução de áreas específicas de conhecimento, existem 9 áreas de conhecimento que estão relacionadas com cada grupo de processo. Cada área de conhecimento é definida por seus requisitos de conhecimento e descrita em termos de processos que compõem suas práticas, entradas, saídas, ferramentas e técnicas (PMI, 2008).

- **Integração**– Inclui os processos e as atividades necessárias para identificar, definir, combinar, unificar e coordenar os vários processos e atividades de gerenciamento de projeto dentro dos Grupos de processos de Gerenciamento do projeto.
- **Escopo**– Inclui os processos necessários para assegurar que o projeto inclua todo o trabalho necessário, e apenas o necessário, para que o projeto termine com êxito.
- **Tempo**– Inclui os processos necessários para gerenciar o término pontual do projeto.
- **Custos**– Inclui os processos envolvidos em estimativas, orçamentos e controle dos custos, de modo que o projeto possa ser finalizado dentro do orçamento aprovado.
- **Qualidade**– Inclui os processos e as atividades da organização executora que determinam políticas de qualidade, os objetivos e as responsabilidades, de modo que o projeto satisfaça as necessidades para as quais foi empreendido.
- **Recursos humanos** do projeto – Inclui os processos que organizam e gerenciam a equipe do projeto.
- **Comunicações**– Inclui os processos necessários para assegurar que as informações do projeto sejam geradas, coletadas, distribuídas, armazenadas, recuperadas e organizadas de maneira oportuna e apropriadas.
- **Riscos**–Inclui os processos relacionados com o planejamento, identificação, análise, elaboração das respostas, monitoramento e controle dos riscos em um projeto.
- **Aquisições**– Inclui os processos de compra ou aquisição de produtos, serviços ou resultados externos à equipe do projeto necessários para realizar o trabalho.

A tabela 1 demonstra a integração dos processos com as áreas de conhecimentos, onde para cada área de conhecimento certos processos estão envolvidos.

Áreas de conhecimento	Grupos de processo de gerenciamento de projeto				
	Grupo de processo de iniciação	Grupo de processo de planejamento	Grupo de processo de execução	Grupo de processo de monitoramento e controle	Grupo de processo de encerramento
Integração	Desenvolver o termo de abertura do projeto	Desenvolver o plano de gerenciamento do projeto	Orientar e gerenciar a execução do projeto	Monitorar e controlar o trabalho do projeto Realizar o controle integrado de mudanças	Encerrar o projeto ou fase
Escopo		Coletar os requisitos Definir o escopo Criar EAP		Verificar o escopo Controlar o escopo	
Tempo		Definir as atividades Sequenciar as atividades Estimar os recursos das atividades Estimar a duração das atividades Desenvolver o cronograma		Controlar o cronograma	
Custos		Estimar os custos Determinar o orçamento		Controlar os custos	
Qualidade		Planejar a qualidade	Realizar a garantia de qualidade	Realizar o controle da qualidade	
Recursos Humanos		Desenvolver o plano de recursos humanos	Mobilizar a equipe do projeto Desenvolver a equipe do projeto Gerenciar a equipe do projeto		
Comunicações	Identificar as partes	Planejar as comunicações	Distribuir as informações Gerenciar as expectativas das partes interessadas	Reportar o desempenho	

Riscos		Planejar o gerenciamento dos riscos Identificar os riscos Realizar a análise qualitativa dos riscos Realizar a análise quantitativa dos riscos Planejar as respostas aos riscos		Monitorar e controlar os riscos	
Aquisições		Planejar as aquisições	Conduzir as aquisições	Conduzir as aquisições	Encerrar as aquisições

Tabela 1 - Mapeamento de grupos de processos de gerenciamento de projetos e áreas de conhecimento

O foco deste trabalho é no processo de planejamento de custos, portanto o mesmo será detalhado a seguir. Enquanto que os outros processos não se encaixam no escopo deste trabalho portando detalhes e mais informações podem ser encontradas no Guia PMBOK (2008).

2.2 Planejamento de Custos

Os processos de planejamento indicados no PMI (2008), desenvolvem um plano de gerenciamento e geram documentos que serão usados para executar o projeto (PMI, 2008). A própria natureza do gerenciamento de projetos cria iterações (*loops*) de *feedbacks* periódicos para análises mais detalhadas (PMI, 2008). Mudanças significativas ocorridas ao longo do ciclo de vida do projeto acionam uma necessidade de visitar um ou mais processos de planejamento (PMI, 2008). O planejamento de custos funciona de modo iterativo e incremental. Toda vez que uma entrada do processo de planejamento de custos é adicionada, é necessário atualizar seus processos internos de forma a fornecer maior precisão em relação aos custos do projeto.

O processo de planejamento de custos está diretamente relacionado com as atividades, riscos, recursos humanos e materiais identificados para o projeto. O planejamento de custos define como o projeto deverá se comportar financeiramente, detalhando todos os possíveis custos para a execução do projeto. Este processo que se caracteriza como uma análise de custos (NETO, 2008) de um determinado projeto, é um processo que resulta em saídas importantes para o processo decisório de uma empresa, sendo indispensável para executar diversas tarefas gerenciais, tais como formação de preço, otimização da produção, etc. (NETO, 2008).

Para que se possa melhor entender e executar os processos do planejamento de custos é necessário conhecer algumas definições. Essas definições serão a base conceitual dos próximos itens do trabalho, já que tudo que será criado estará referenciando os dois conceitos a seguir.

Custo é o gasto relativo ao bem ou serviço utilizado na produção de bens e serviços. Como exemplo de custo podemos citar os materiais usados no projeto, mão-de-obra utilizada, a energia elétrica consumida, aluguel de equipamentos e todos os outros custos associados a área de produção do projeto (NETO, 2008).

Orçamento é uma descrição de um plano financeiro. É uma lista de estimativas de receitas e despesas para um período de tempo determinado (ECONOMICS, 2012).

Conhecendo os dois conceitos envolvidos no planejamento de custos, é possível identificar e detalhar os dois processos envolvidos. Seguindo o PMBOK planejar os custos do projeto inclui 2 processos que são (PMBOK, 2008):

- **Estimar os Custos** - Processo de desenvolvimento de uma estimativa dos recursos monetários necessários para executar as atividades do projeto.
- **Determinar o Orçamento** – Processo de agregação dos custos estimados de atividades individuais ou pacotes de trabalho para estabelecer uma *baseline* dos custos autorizada.

Estes processos segundo o PMBOK (2008) interagem entre si, e os esforços necessários para sua execução dependem da equipe de gerenciamento do projeto.

Outra parte importante no planejamento de custos são os requisitos levantados pelas partes interessadas do projeto. O processo de planejamento leva em consideração esses requisitos a fim de seguir as restrições impostas ao projeto. Cada parte interessada no projeto pode declarar requisitos de custo diferentes de outro membro da equipe, portanto existe a necessidade de gerenciar essas diferenças (PMI, 2008).

Como todo processo dentro do gerenciamento de projetos, cada processo necessita de entradas provenientes de outros processos para poder construir as saídas desejadas. Primeiramente no planejamento de custos será realizada a estimativa de custos, onde serão estimados todos os custos que estiverem envolvidos com o projeto, desde custos de pessoal a custos materiais. Logo depois será feito o orçamento total do projeto com base nas entradas recebidas.

Para ilustrar o processo de planejamento de custos será usado um exemplo, que será referenciado nos próximos tópicos a fim de demonstrar como funciona o processo de planejamento de custos.

Exemplo. Um cliente, que é dono de uma pizzaria, possui uma proposta de projeto. Atualmente ele já oferece a entrega em domicílio via ligações telefônicas. Para ampliar o seu negócio, ele quer possibilitar que os seus clientes, por meio da Internet, possam encomendar pizzas no site da sua pizzaria. Estas informações então serão processadas por seus dois atendentes, que precisarão ser treinados, visto que atualmente tem pouco conhecimento de TI. Além disso, ele também quer um módulo para dispositivos móveis deste sistema a ser acessado pelos iPhones dos entregadores, por qual eles possam verificar detalhes da entrega (endereço, valor total, etc.). O dono da pizzaria também pretende lançar o sistema daqui a seis meses. Ele conseguiu reservar R\$ 20.000,00 para o projeto.

2.2.1. Estimar os Custos

Estimar os custos de um projeto inclui a identificação e a consideração de alternativas de custos para iniciar e finalizar o projeto (PMI, 2008). As estimativas de custos são geralmente expressas em unidades monetárias (dólar, real, euro etc.) com a finalidade de facilitar comparações dentro ou fora do projeto. Outras unidades tais como pessoa-horas ou pessoa-dias podem ser utilizadas, desde que o seu uso não adultere os custos do projeto. Em alguns casos, as estimativas terão de ser fornecidas usando várias unidades de medida com a finalidade de facilitar o apropriado controle da gerência. As estimativas de custo podem ser beneficiadas por refinamentos ocorridos durante o curso do projeto como reflexo dos detalhes adicionais adquiridos como entradas (PMI, 2008).

A percentagem de precisão das estimativas do projeto está relacionada diretamente com cada grupo de processo. Temos em uma ordem de magnitude aproximada que na fase de iniciação do projeto até a fase de planejamento, as estimativas de custo ficariam entre uma variância de -50 até +100 %, e já para a fase de execução do projeto as estimativas assumiriam um valor definitivo ou ficariam na média na casa dos -10 até +15 % (PMI, 2008).

Todas essas variações das estimativas no projeto devem ser minimizadas com ferramentas e técnicas que tentam nos trazer estimativas cada vez mais precisas dos custos do projeto. Para ir mais longe e estimar um custo mais próximo do custo real do projeto, é necessário saber que itens e que custos devem ser adicionados ao projeto. O PMI (2008) descreve resumidamente alguns tipos de custos em que se devem enquadrar os itens a serem estimados. Para obter um melhor entendimento do que deve ser estimado em um projeto, conceitos da contabilidade de custo são apresentados a seguir.

Primeiramente, os **custos diretos** entende-se por aqueles que podem ser imediatamente apropriados a um só tipo de produto ou a um só tipo de serviço (FERREIRA, 2007). Especificamente, são aqueles que podem ser apropriados diretamente a um produto, um serviço, uma atividade, etc. (FERREIRA, 2007). Os custos diretos

constituem todos aqueles elementos de custo individualizáveis com respeito ao produto ou serviço, isto é, se identificam imediatamente com a produção dos mesmos (ZANLUCA).

Exemplos de custos diretos:

- Matéria-prima usada no projeto
- Mão-de-obra direta
- Serviços subcontratados e aplicados diretamente nos produtos ou serviços

Para que se possam obter as estimativas o PMI (2008) cita que a pessoa que determina os valores ou o grupo que prepara as estimativas deve conhecer os valores de custo unitário, como o custo de um funcionário por hora e o custo do material, para cada recurso a fim de estimar os custos das atividades.

Para a contabilidade de custos segundo FAGUNDES, para conhecer o consumo de mão-de-obra, é preciso que a empresa mantenha um sistema de apontamentos, por meio do qual se verifica quais os funcionários que trabalham em cada produto (ou serviço) no período (dia, semana, mês) e por quanto tempo (minutos, horas).

Esses conhecimentos trazem um poder de decisão para estimar os custos dos itens do projeto. Identificados os itens que devem ser estimados para os custos diretos, podemos então realizar efetivamente a estimativa dos custos. O PMBOK cita técnicas e ferramentas para ajudar na estimativa de custos dos itens do projeto, porém nem sempre essas técnicas são usadas na prática (PMI, 2011).

As apropriações dos custos diretos podem ser feitas com relativa facilidade já que podem ser facilmente quantificadas e identificadas para o projeto (FAGUNDES). Podemos determinar o custo de cada elemento em um processo direto. Supondo que os itens diretos do projeto já foram identificados pelo grupo ou pessoa responsável, vamos supor que 1 item necessita de 30 horas de trabalho de um programador júnior para que possa ser finalizada. O programador júnior será remunerado com o valor de R\$12,50 por hora. O custo apropriado para esta tarefa será:

$$30 \text{ pessoa-horas} * R\$12,50/\text{hora} = R\$375,00$$

Em caso do custo de um equipamento, seu valor será inteiramente debitado diretamente no projeto. Agora em caso de empréstimo de equipamento o custo será calculado da mesma forma direta (R\$ por hora ou R\$ por ciclo de trabalho) (FAGUNDES).

Exemplo. Supondo que nosso projeto será realizado por dois alunos do curso de Ciências da Computação com certa experiência em desenvolvimento de aplicações web. Começaremos o planejamento de custos identificando e atribuindo custos aos itens classificados como custos diretos do projeto.

Custos Diretos				
Itens	Quantidade	Esforço(pessoa-hora)/Duração de uso	Custo Unitário	Custo Total
Programador	2	480	R\$8,00	R\$7680,00
Instrutor	1	40	R\$8,00	R\$320,00
Iphone	1	3 meses	R\$2000,00	R\$2000,00

Tabela 2 – Exemplo Custos Diretos

A outra classificação, os **custos indiretos**, são os custos que por não serem perfeitamente identificados nos produtos ou serviços não podem ser apropriados de forma direta para as unidades específicas, ordens de serviço ou produto. Necessitam, portanto, da utilização de algum critério de rateio¹ para sua alocação (por exemplo, aluguel, energia elétrica) (FAGUNDES). É um custo comum a tipos diferentes de bens, sem que se possa separar a parcela referente a cada um, no momento de sua ocorrência (DUTRA, 1995). No contexto mais voltado para o gerenciamento de projetos, PONTES (2010) cita que os custos

¹ Rateio - Divisão ou distribuição proporcional de uma quantidade ou de uma quantia entre vários (Infopédia)

indiretos são os custos que beneficiam todos os projetos, mas que não são identificados com cada projeto específico.

Exemplo de custos indiretos:

- Taxas
- Depreciação
- Benefícios
- Serviços de Zeladoria
- Aluguel
- Energia Elétrica

Portanto o valor dos custos indiretos pode ser obtido conforme a distributividade do seu custo nos projetos da empresa, ou seja, conforme seu uso em diferentes partes da produção (projetos). FAGUNDES também nos mostra que pode ser atribuído alguns critérios para a alocação de custos indiretos.

Exemplos de critérios para custos indiretos

1. Mão-de-obra indireta: pode ser usado o critério de tempo de utilização de mão-de-obra direta, para então definir o custo do item indireto, tomando como base o custo de mão-de-obra direta.

2. Materiais indiretos: pode ser usado o critério de consumo de material direto. Usando parâmetros definidos nos custos diretos e obtendo a parcela de uso para o item referenciado, podemos calcular o custo do item indireto sobre o projeto ou produção.

3. Depreciação Equipamentos: pode ser usado como critério o tempo de utilização da máquina para execução de certa tarefa, assim o valor total da máquina será absorvido em vários projetos.

O conceito de depreciação é um dos mais importantes da contabilidade (DESTRI, 2008). Devido não só a sua presença praticamente em todas as demonstrações de

resultados, como também, devido à sua relevante participação percentual no total de custos da maioria das empresas (DESTRI, 2008). A **depreciação** nada mais é do que o registro contábil do valor de um bem consumido pela empresa num determinado período, em função do seu uso e/ou obsolescência (DESTRI, 2008). Evidentemente, uma máquina adquirida para ser utilizada no processo produtivo de uma empresa, no momento em que entra em funcionamento começa a ser consumida, e parte da sua vida útil ou econômica está sendo absorvida no processo produtivo (DESTRI, 2008). A vida útil econômica de uma máquina, equipamento, veículo, etc, pode ser absorvida não apenas pelo seu uso, mas também pela obsolescência (DESTRI, 2008). O cálculo do valor da depreciação tem por objetivo distribuir o custo do item que a empresa utiliza nas suas atividades pelos projetos futuros, durante a vida útil ou econômica do item (DESTRI, 2008). Isto significa que o cálculo da depreciação depende de 3 fatores; do custo do item, a estimativa da sua vida útil e seu valor de sucata, seguindo a seguinte fórmula:

Por exemplo, se quisermos saber o valor de depreciação de um computador desktop que custou R\$4.000,00, tem uma vida útil de 2 anos e seu valor de sucata é de R\$200,00 , basta utilizarmos a fórmula:

$$\text{Depreciação Anual} = \text{Valor Item} - \text{Valor Sucata} / \text{Vida Útil}$$

$$\text{Depreciação} = 4000,00 - 200,00 / 2$$

$$\text{Depreciação Anual} = \text{R\$1900,00}$$

Exemplo. Depois de visto o conceito de custos indiretos, iremos adicionar ao nosso projeto esses custos de forma a completar os itens faltantes em relação ao primeiro passo executado.

Custos Indiretos				
Item	Quantidade	Tempo	Custo Unitário	Custo total
Aluguel	1	6 meses	R\$300,00	R\$1800,00
Energia Elétrica	1	6 meses	R\$90,00	R\$540,00
Software iPhone	1	1	R\$100,00	R\$100,00
Servidor (depreciado)	1	6 meses	R\$1000,00	R\$1000,00

Notebook (depreciado)	2	6 meses	R\$1000,00	R\$100,00
Treinamento programação aplicativo iPhone para os programadores	1	1	R\$1000,00	R\$1000,00

Tabela 3 – Exemplo Custos Indiretos

Visto os dois principais tipos de custos classificadas na contabilidade de custos, é apresentado uma definição e exemplos de outras duas categorias de custos que o PMI (2008) também cita e é de grande valia para maior entendimento do assunto.

Custos Fixos: são aqueles permanecem constantes, independentemente do volume de produção, ou ainda, os custos que permanecem período após período sem variações, não ocorrendo em consequência de variação no volume de atividade em períodos iguais (FAGUNDES). Alguns exemplos de custos fixos: salários da administração, salários dos supervisores, seguros, depreciações, alguns impostos, juros de empréstimos, aluguel, entre outros.

Custos Variáveis: Os custos variáveis são aqueles que variam proporcionalmente com o volume de atividade (produção) desenvolvida pela empresa. Quanto maior o volume de produção, no período, maior será o custo variável (REZENDE, 2011). Alguns exemplos de custos variáveis: material direto, mão-de-obra direta.

O PMI (2008) concede várias técnicas e ferramentas para que os custos do projeto sejam obtidos, sempre visando obter um melhor resultado. Todos os custos vistos acima podem ser obtidos através de técnicas citadas pelo PMBOK (2008), como é de interesse deste trabalho seguir as melhores técnicas de desenvolvimento e planejamento de projetos de *software*, focaremos os nossos esforços para utilização dessas técnicas.

Os custos são estimados para todos os recursos que serão cobrados do projeto (PMI, 2008). Uma **estimativa de custo** é uma avaliação quantitativa dos custos prováveis dos recursos necessários para completar a atividade (PMI, 2008).

O processo para obter as estimativas de custos do projeto deriva de uma série de procedimentos. As fontes de entradas de informações são derivadas do resultado ou saída de outros processos. Uma vez uma entrada absorvida por um processo, ali ela estará disponível para todos os processos de gerenciamento de custos. Não diferente dos outros processos, a estimativa de custos será estruturada na forma de que recebam as entradas necessárias, processem-nas com suas ferramentas e técnicas, e no final fornecem uma(s) saída(s). Para melhor entender a estrutura e suas necessidades a figura a seguir ilustra tais etapas.

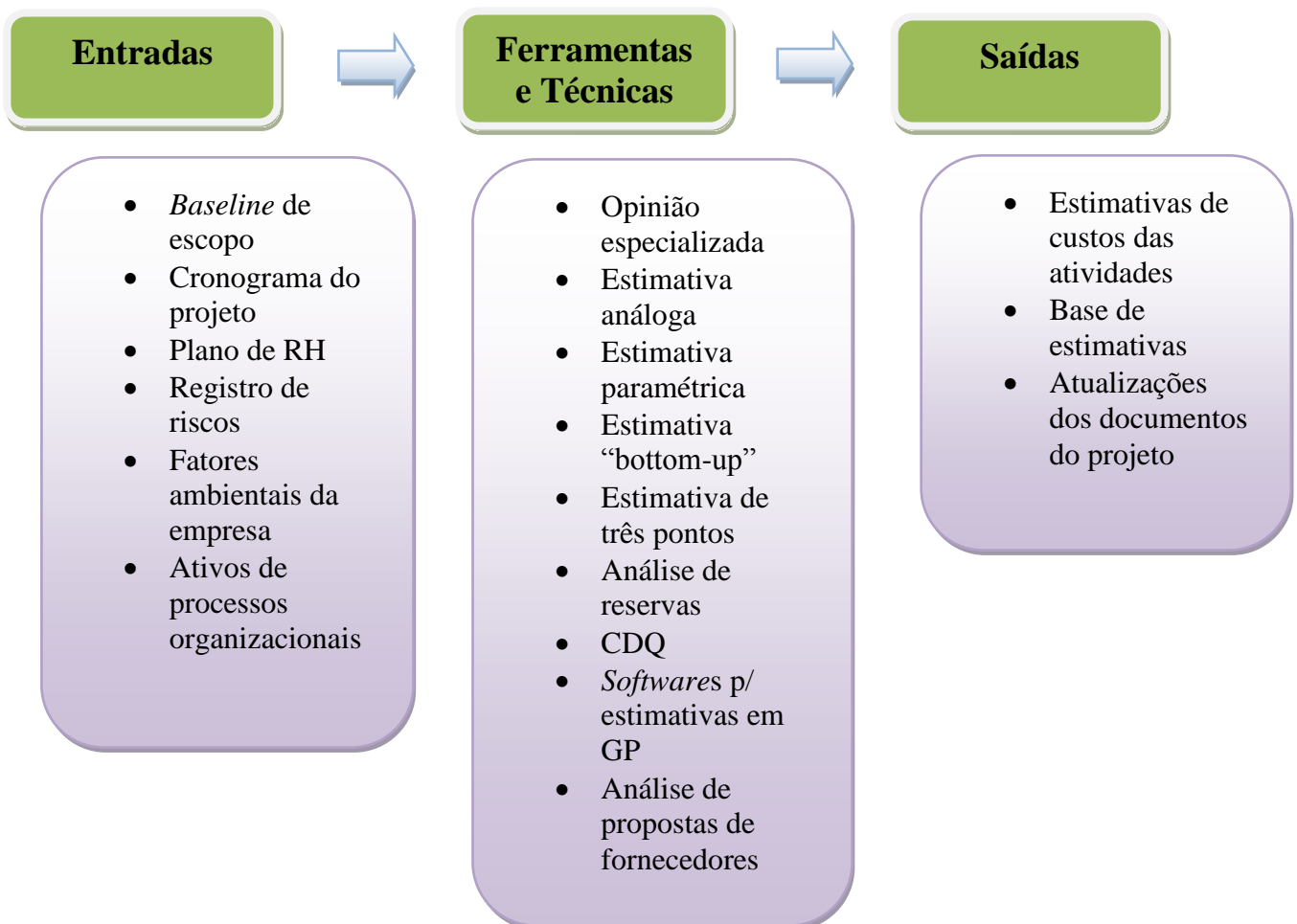


Figura 3 – Estrutura do processo de estimar custos fonte (PMI, 2008)

As **entradas** do processo mostrado na figura 3 derivam de vários processos. Suas entradas são descritas a seguir individualmente (PMI, 2008):

- A **baseline de escopo** nos fornece que premissas de declaração de escopo, relações descritas na EAP ou implicações contratuais (ex. saúde, segurança, proteção, seguro, entre outros) encontradas no dicionário da EAP podem afetar os custos do projeto.
- O **cronograma do projeto** diz que o tipo e a quantidade de recursos alocados interferem diretamente nos custos do projeto. *Obs.:* Os custos do projeto podem sofrer variações ao longo do tempo, por isso, é de suma importância conhecer detalhadamente o cronograma do projeto.
- O **plano de recursos humanos** contém atributos de pessoal, reconhecimentos ou prêmios que podem afetar os custos do projeto.
- O **registro de riscos** nos ajuda na mitigação dos riscos sendo que o plano de custos deve considerar os custos envolvidos.
- Os **fatores ambientais da empresa** são condições de mercado e informações comerciais publicadas influenciam o processo de estimativa dos custos.
- **Ativos de processos organizacionais** são políticas e modelos de estimativas de custos, informações históricas e lições aprendidas que podem influenciar a estimativa de custos do projeto.

Descrito abaixo resumidamente estão às **ferramentas e técnicas** que podem ser usadas para o desenvolvimento do processo (PMI, 2008).

- **Opinião especializada:** especialistas oferecem opiniões técnicas para a composição das estimativas de custos.
- **Estimativa análoga:** usa valores de projetos anteriores como referências.
- **Estimativa paramétrica:** utiliza parâmetros no cálculo das estimativas.
- **Estimativa “bottom-up”:** os custos dos pacotes de trabalho são resumidos em níveis maiores.
- **Estimativa de três pontos:** considera variabilidade e riscos, utilizando uma média ponderada (PERT).

- **Análise das reservas:** aplicação de reservas para contingências.
- **Custo da qualidade (CQD):** aplicação de premissas sobre a qualidade das entregas do projeto.
- **Softwares para estimativa em gerenciamento de projetos:** planilhas, simuladores e ferramentas estatísticas.
- **Análise de propostas de fornecedores:** análise baseada nas respostas de cotações com fornecedores.

Por fim teremos as **saídas** do processo, que poderão se tornar entrada em algum outro processo (PMI, 2008).

- **Estimativa de custos das atividades:** avaliações quantitativas dos custos prováveis para execução do projeto. *Obs.:* Os custos indiretos podem ser incluídos nas atividades ou em níveis mais altos.
- **Base de estimativas:** memória de cálculo das estimativas de custos.
- **Atualizações dos documentos do projeto.**

O custo estimado e aprovado, só pode sofrer alterações sob autorização.

Exemplo. Neste momento produziremos a nossa tabela de estimativas de custos, integrando os custos diretos aos indiretos. A metodologia para tomada de decisões se baseia em experiências passadas com projetos semelhantes. Segue uma tabela com estimativas iniciais para o projeto.

Item	Quantidade	Esforço (pessoa-hora)	Custo Unitário	Custo Total
Pessoal				
Programador	2	480horas	R\$ 8,00/hora	R\$7680,00
Instrutor	1	40horas	R\$8,00/hora	R\$320,00
Equipamento e Material				
Notebook	2		R\$ 1.000,00	R\$ 2.000,00

(depreciado)				
iPhone	1		R\$ 2.000,00	R\$ 2.000,00
Servidor (depreciado)	1		R\$ 3.000,00	R\$ 3.000,00
Treinamento				
Programação aplicativo iPhone para os programadores	1	40 horas	R\$ 1000,00	R\$ 1000,00
Custeio				
Aluguel	1	6 meses	R\$ 300,00	R\$ 1.800,00
Software iPhone	1		R\$ 100,00	R\$ 100,00
Energia Elétrica		6 meses	R\$90,00	R\$540,00

Tabela 4 – Exemplo estimativas de custos

2.2.1.2 Determinar Orçamento

Um **orçamento** determinado fornece ao projeto os recursos financeiros autorizados para a sua execução, tornando a medição dos custos no decorrer do projeto uma comparação com o orçamento autorizado (PMI, 2008). O objetivo do orçamento é realizar a agregação dos custos estimados de atividades para estabelecer uma *baseline* dos custos autorizados, para facilitar e ajudar no gerenciamento de todos os seus custos.

A **baseline do custo** é o orçamento referencial que será utilizado para medir e monitorar o desempenho do custo do projeto. É desenvolvido através da totalização das estimativas de custo por período e, usualmente, é apresentada na forma de Curva-S (PMI, 2008). Uma vez que é bastante comum que projetos tenham gastos menores no início e no fim de seu ciclo de vida, a curva acumulada ganha um formato em “S” (GARCIA, 2008).

Exemplo:

Despesas iniciais de projeto: R\$ 15.000,00 em 2 semanas

Despesas durante a execução: R\$ 200.000,00 em 4 semanas

Despesas de fechamento: R\$ 10.000,00 em 1 semana

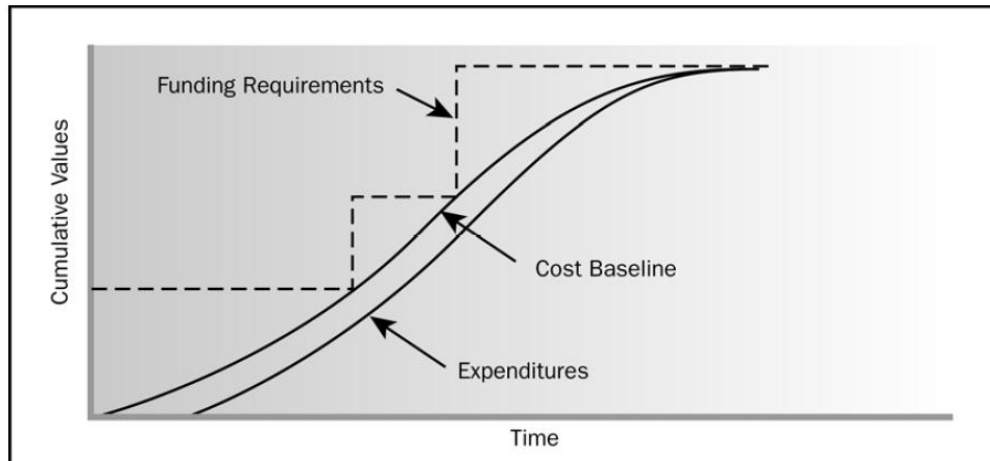


Figura 4 – Curva-S (*Baseline* de custo, gastos e requisitos de recursos financeiros)
(PMBOK, 2008)

Um ponto muito importante na realização do orçamento do projeto é a análise de riscos, que segundo as boas práticas de gerenciamento de projetos (PMI, 2008), deve ser construída no decorrer do gerenciamento do projeto. A ocorrência de riscos no projeto, com certeza, poderá trazer uma série de prejuízos ao projeto em termos de tempo, qualidade e custos. Os riscos podem ter sua origem na área de recursos humanos, como: perda de profissionais-chave no projeto, atrasos no processo de contratação de mão de obra, verba para treinamento, etc, mas também podem ter sua origem em alterações de preços, como: passagens aéreas, hotéis, aluguéis de veículos, variação cambial, no custo etc. (FILHO, 2010).

Existem vários tipos de riscos, os que podem ser previstos e outros que são inesperados ou ainda “inimagináveis”. Os riscos previstos podem ser gerenciados através de abordagens citadas pelo PMBOK (2008), como: evitar um risco, reduzir o impacto ou a probabilidade de ocorrência, contratar um seguro para o risco, elaborar planos de contingência, o qual é disparado caso o risco se torne realidade. Todo risco diante de suas características, deve ter uma abordagem específica em relação ao impacto que pode causar no projeto e da probabilidade de ocorrência. Estes dois fatores (impacto e probabilidade) determinam o grau de severidade de um risco (FILHO, 2010).

Os riscos podem ter um impacto negativo no projeto caso ocorram, e por isso estão diretamente ligados ao plano de reserva de um projeto. A **reserva** é uma provisão orçamentária que tem o objetivo proteger o projeto contra determinados imprevistos que podem acontecer a ponto de conseguir terminar o mesmo dentro do orçamento macro, que segundo VARGAS, é o orçamento somado as reservas calculadas para o projeto. O PMI (2008) cita dois tipos de reservas, reserva de contingência e reserva gerencial, que devem ser definidos para que diminua/anule o impacto dos riscos.

Para a obtenção do valor da reserva de contingência VARGAS orienta o entendimento do conceito de valor esperado. Valor esperado se caracteriza por, a probabilidade de um risco ocorrer multiplicado pelo impacto do risco no projeto. A soma desses valores vai produzir a reserva de contingência.

A **reserva de contingência** é caracterizada como a reserva financeira para os riscos conhecidos do meu projeto, ou seja, riscos identificados no meu plano de riscos (VARGAS).

Exemplo. Em nosso projeto identificamos alguns riscos que podem ocorrer em função de conhecimentos adquiridos em projetos similares anteriores.

Quebra HD

Probabilidade de ocorrência – 10%

Impacto no projeto – R\$300,00

Valor esperado -> $0,1 \times R\$300,00 = R\$30,00$

Roubo Iphone

Probabilidade de ocorrência – 5%

Impacto no projeto – R\$2000,00

Valor esperado -> $0,05 \times R\$2000,00 = R\$100,00$

Logo nossa reserva de contingência será a soma dos valores esperados dos riscos identificados distribuídos uniformemente na *baseline* de custo do projeto.

$(6 \text{ meses} \times R\$30,00) + (3 \text{ meses} \times R\$100,00) = R\$480,00$

A **reserva gerencial** é o custo reservado para mudanças não planejadas no escopo e custos do projeto (PMI, 2009). Ela pode ser conceituada como uma função ou percentual do orçamento do projeto (VARGAS, 2010).

A reserva gerencial de um projeto vai depender das características da organização em questão. Caso esta organização mantenha um gerenciamento mais conservador, a reserva gerencial tende a aumentar, caso contrário, se a organização não for tão conservadora essa reserva tende a ficar menor (VARGAS). Vê-se que empresas adotam como padrão determinados percentuais com base em suas experiências passadas, tais como: 5%, 8% ou 10% (FILHO, 2010).

A reserva gerencial nada mais é que uma poupança aleatória baseada em histórico e/ou grau de risco que eu presumo correr (VARGAS). Não há metodologia alguma que defina estes patamares, apenas a experiência prévia e vivência profissional de gerentes e diretores de projetos (FILHO, 2010).

Resumindo, quanto mais certeza eu tenho e mais precisa minha reserva de contingência for, menos reserva gerencial eu vou precisar, agora quanto mais desconhecido for o meu projeto mais reserva gerencial eu vou precisar (VARGAS). A reserva de contingência é inclusa na *baseline* de desempenho de custos e ambas as reserva inclusas no orçamento.

A determinação do orçamento do projeto como outros processos, se caracteriza pelo recebimento de entradas para produzir saídas. O PMI (2008) instrui como isso deve ser feito. Como objetivo deste trabalho se enquadra nas melhores práticas de gerenciamento de projetos, iremos focar nossos esforços para essas técnicas.

Segundo PMI (2008), segue a estrutura e definição dos processos para determinar o orçamento do projeto.

Entradas

- **Estimativas de custos das atividades:** são agregadas para obter a estimativa de custos dos pacotes.
- **Bases de estimativas:** contém as premissas básicas para inclusão ou exclusão de custos indiretos.
- **Baseline de escopo:** a declaração de escopo fornece as restrições de recursos, a EAP as relações entre as entregas e seus componentes e o dicionário da EAP fornece a descrição do trabalho necessário para completar as atividades.
- **Cronograma do projeto:** incluem as datas, marcos, pacotes de trabalho e contas de controle. Essas informações são úteis para agregar os custos nos períodos do calendário.
- **Calendário de recursos:** utilizado para informar os custos dos recursos durante o projeto.
- **Contratos:** informações de produtos, serviços ou resultados já contratados.
- **Ativos de processos organizacionais:** políticas, procedimentos, ferramentas e métodos adotados pela empresa.

Ferramentas e Técnicas

- **Agregação dos custos:** os custos das atividades são agregados nos pacotes de trabalho, que são agregados em componentes maiores da EAP até o nível do projeto.
- **Análise de reservas:** estabelecimento das reservas gerenciais (que não fazem parte da linha de base de custos) destinadas a possíveis mudanças de escopo.
- **Opinião especializada:** utilização de profissionais especializados (outras áreas, consultores, clientes, etc.) **Relações históricas:** aplicação de estimativas análogas ou paramétricas.
- **Reconciliação do limite de recursos financeiros:** aplicação de reagendamento do trabalho em virtude de do nivelamento dos gastos do projeto.

Saídas

- **Baseline de desempenho de custos:** orçamento no término autorizado. É desenvolvido pelo acúmulo de orçamentos aprovados por período (Curva S).
- **Requisitos de recursos financeiros do projeto:** são os gastos projetados por período (semestre, ano, etc.), incluindo as reservas gerenciais.
- **Atualização dos documentos do projeto:** registro de riscos, estimativa de custos e cronograma do projeto.

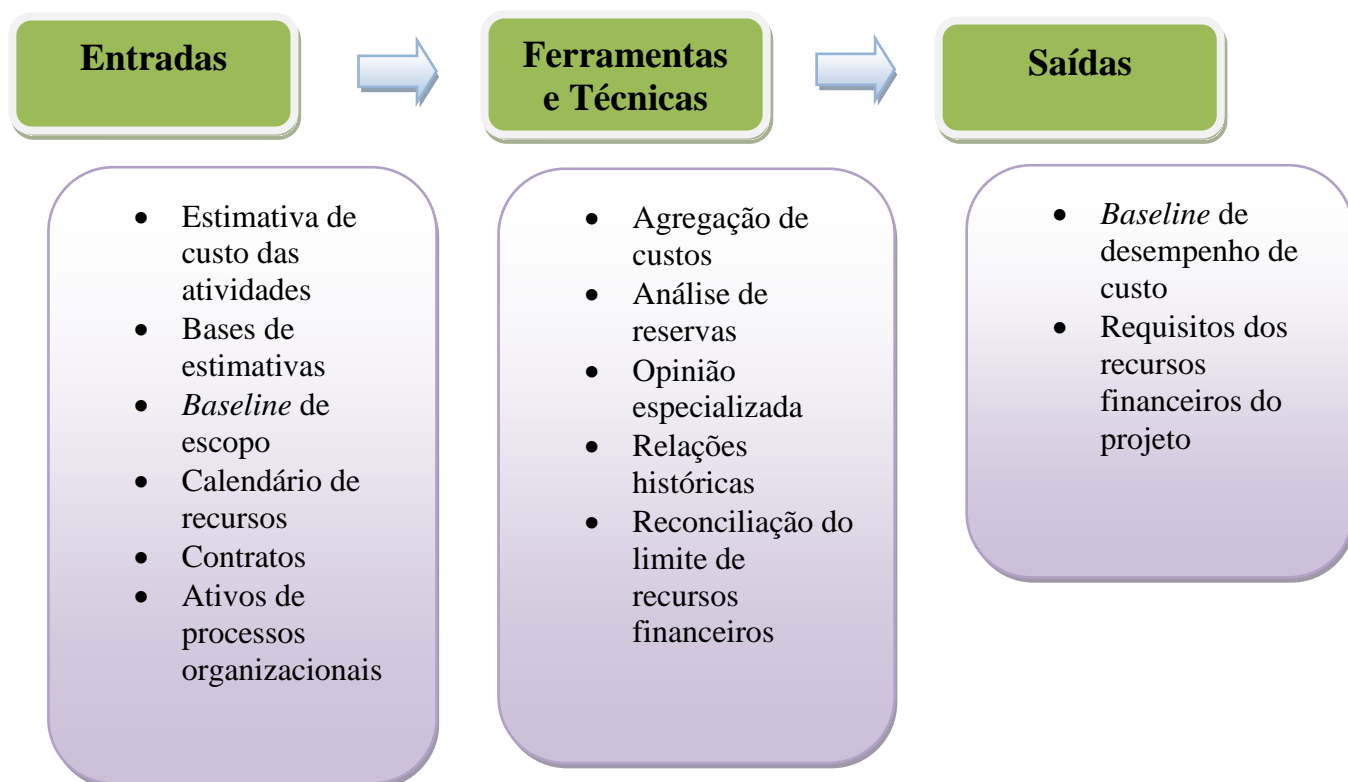


Figura 5 – Estrutura do processo de determinar orçamento

Exemplo. Agora que a equipe desenvolveu as estimativas de custo do projeto, levando em conta as atividades previstas, recursos humanos e não humanos, os dois membros irão determinar o orçamento do projeto. Usando o calendário do projeto criaremos uma *baseline*

de desempenho de custo relacionado com o tempo. Ao final produziremos o orçamento do projeto, identificando suas reservas. Resultado:

Item	Janeiro	Fevereiro	Março	Abril	Mai	Junho	Total
Pessoal							
Programador	R\$ 1280,00	R\$ 1280,00	R\$ 1280,00	R\$ 1280,00	R\$ 1280,00	R\$ 1280,00	R\$7.680,00
Instrutor							
Equipamento e Material							
Notebook	R\$ 333,33	R\$ 333,33	R\$ 333,33	R\$ 333,33	R\$ 333,33	R\$ 333,33	R\$2.000,00
iPhone				R\$ 666,66	R\$ 666,66	R\$ 666,66	R\$2.000,00
Servidor		R\$200,00	R\$200,00	R\$200,00	R\$200,00	R\$200,00	R\$1.000,00
Treinamento							
Programação aplicativo iPhone para os programadores				R\$500,00	R\$500,00		R\$1.000,00
Custeio							
Aluguel	R\$300,00	R\$300,00	R\$300,00	R\$300,00	R\$300,00	R\$300,00	R\$1.800,00
Software iPhone				R\$100,00			R\$100,00
Reservas							
Risco: Quebra de HD	R\$30,00	R\$30,00	R\$30,00	R\$30,00	R\$30,00	R\$30,00	R\$180,00
Risco: Roubo Iphone				R\$100,00	R\$100,00	R\$100,00	R\$300,00
Total	R\$1943,33	R\$2143,33	R\$2143,33	R\$3510,00	R\$3410,00	R\$2910,00	R\$16.060,00

Tabela 5 – Exemplo *baseline* de custos

Orçamento

Orçamento = Custo Total + Reserva Gerencial

Reserva Gerencial = R\$16.060,00 x 10% = R\$1.606,00

Orçamento = R\$ 16.060,00 + R\$1.606,00
= R\$ 17.666,00

2.3 Micro e Pequenas Empresas

Neste capítulo é apresentada uma visão geral das micro e pequenas empresas no cenário do *software*, mostrando suas características e analisando suas necessidades em relação ao gerenciamento de projeto.

A Lei Geral de Micro e Pequenas Empresas, sancionada em 2006, classifica as empresas de acordo com a sua renda, separando-as em 2 grupos: as microempresas são aquelas em que a receita bruta anual é igual ou inferior a R\$ 240.000,00; e empresas de pequeno porte são aquelas com receita bruta anual superior a R\$ 240.000,00 e igual ou inferior a R\$2.400.000,00 (SEBRAE, 2009). O SEBRAE nos diz também que esta classificação pode ser realizada conforme o número de funcionários da organização, essa classificação é caracterizada do tipo, se a empresa relacionada ao comércio e serviços tem até 9 funcionários ela é classificada com micro empresa, já a empresa que tem de 10 a 49 funcionários se enquadra como pequena empresa. Neste trabalho é utilizada a definição de micro e pequena empresa levando em conta o número de funcionários, pois são dados mais fáceis de se obter em uma empresa ao invés de tentar obter a renda bruta anual da empresa.

O ano de 2010 foi um ano de recuperação para o setor de TI no Brasil, que mostrou um crescimento da ordem de 21,3%. Especificamente os setores de *software* e serviços cresceram quase 24 %, um pouco menos que o segmento de hardware. (ABES, 2011). Em 2010, a participação de programas de computador desenvolvidos no país (standard e sob encomenda) atingiu 35% do total do mercado brasileiro de *software*, reforçando a tendência de crescimento que vem sendo apontada desde 2004 (ABES, 2011). Este mercado é explorado por cerca de 8.520 empresas, dedicadas ao desenvolvimento, produção e distribuição de *software* e de prestação de serviços. Das que atuam no desenvolvimento e produção de *software*, 94% são classificadas como micro e pequenas empresas (ABES, 2011).

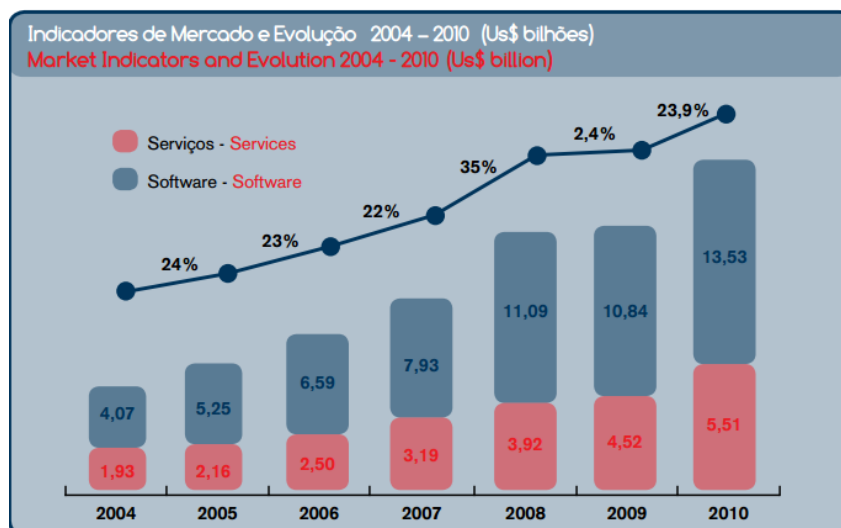


Figura 6 - Evolução mercado de *Software* 2004 -2010(ABES, 2011)

Algumas empresas que se julgam promissoras hoje em dia estão em busca da diversificação dos serviços prestados, fazendo a tentativa de adquirir técnicas mais modernas de gerenciamento de projetos que permitam aperfeiçoar os processos, a fim de cada vez mais garantir a qualidade e preços mais competitivos no mercado. Segundo uma pesquisa realizada pelo SEBRAE mostra que 49,4% das empresas morrem até os dois anos de existência, aumentando para 56,4% até os três primeiros anos e para 59,9% até os quatro anos. Dentre os fatores que contribuem para tal fato, está a forte concorrência, a inexperiência gerencial e de marketing dos proprietários, a alta carga tributária, a falta de capital de giro e a ausência de políticas internas voltadas para a garantia de qualidade dos produtos (WEBER;et.al, 2005).

Empresas de pequeno porte têm como característica a improvisação, momentos em que o projeto necessita de algum suporte no gerenciamento e a empresa não o têm, porém no final da execução do projeto podem ocorrer conflitos em relação às melhores práticas de gerenciamento de projetos, criando um possível ponto negativo no desenvolvimento (ex. Devido à falta de planejamento e imprevistos durante a execução, o custo esperado do projeto foi excedido) (AZANHA, 2003). Com esse crescimento da produção de *softwares* no Brasil fica evidente que sua qualidade deve crescer junto.

Segundo (PMSURVEY.ORG, 2011) a maioria das empresas possui uma metodologia de gerencia de projetos única, a qual é utilizada para todas as áreas. Porém o restante ou não

possui método algum ou contém metodologias específicas para cada área de gerenciamento (PMSURVEY.ORG, 2011). Em relação aos custos do projeto, a pesquisa mostra que empresas com renda abaixo de 1 milhão de reais, 76% delas citaram o aspecto de cálculo de custos nas metodologias usadas (PMSURVEY.ORG). Agora para a prática da execução do orçamento apenas 69% das mesmas empresas citaram em suas metodologias (PMSURVEY.ORG).

2.4 dotProject

O dotProject é um sistema de gerência de projetos em *software* livre de fácil utilização, com um conjunto de funcionalidades e características que o tornam indicado para implementação em ambientes corporativos, pois atende a diversas necessidades de gerentes e escritórios de projetos (PMOs) (DOTPROJECT, a). O sistema provê mecanismos de planejamento e controle para atividades, cronograma, comunicação, e aquisições.

O dotProject é uma aplicação web e seu acesso é feito através de um navegador, assim sua utilização independe de sistema operacional e instalação na máquina do usuário, pois é executado em um servidor. Em termos técnicos, o dotProject é um sistema escrito em PHP, que utiliza banco de dados MySQL (DOTPROJECT).

O módulo *core* do dotProject nos fornece as funcionalidades básicas para o gerenciamento de projetos listadas abaixo (DOTPROJECT, b):

- **Backup:** É um módulo recente de backup do banco de dados da ferramenta. Não se recomenda usar este módulo como único recurso de segurança por instabilidade.
- **Calendário:** Este módulo permite a visualização de datas de tarefas e compromissos.
- **Organizações:** As organizações, são as entidades que agrupam projetos, tarefas e usuários. É necessária a existência de pelo menos uma organização para se criar um projeto na ferramenta.

- **Contatos:** Permite o cadastro de contatos com endereço, nome e outras informações.

Estar na lista de contatos não permite a pessoa associada efetuar login no sistema.

- **Departamentos:** São subconjuntos contidos nas organizações. Permitem um agrupamento mais refinado de usuários e tarefas.
- **Arquivos:** Funções básicas de armazenamento de arquivos e controle de versão simples.
- **Fóruns:** Permite a criação de fóruns do projeto para discussões, notas e informações.
- **Relatórios:** Este módulo de relatórios está sendo inserido no dotProject e não está disponível em versões anteriores a versão 3.x.
- **Diagrama de Gantt:** Fornece a criação de diagramas de Gantt baseados nas tarefas e atividades definidas.
- **História:** É uma funcionalidade que provê um log histórico de mudanças e revisões

- **Links:** Fornece suporte a gravação e disponibilização de links (para sites ou locais internos/externos).
- **Projetos:** Um projeto, no contexto do dotProject, é o elemento que contém as tarefas.
- **Recursos:** É o módulo que incorpora recursos não humanos ao projeto.
- **Procura inteligente:** Esquema inteligente de procura pelos módulos do sistema.
- **Administração de sistema:** Inclui as opções de configuração da ferramenta.
- **Tarefas:** Tarefas são os componentes de trabalho. Nela são gravadas horas gastas, usuários responsáveis, logs, entre outros.
- **Administração de usuários:** Esquema de gerenciamento de usuários.

Para MPEs um gerenciamento de projeto numa ferramenta como o dotProject é de grande valia, pois mesmo alguns destas funcionalidades sendo feitas manualmente, o controle mais detalhado do projeto facilita o seu gerenciamento. No Brasil existe a comunidade do dotProject que reúne discussões e possibilidades de melhorias provendo um grande suporte a dúvidas (DOTPROJECT, a).

3. Estado da arte

Neste capítulo é elaborada uma análise avaliando o grau de suporte de ferramentas de gerenciamento de projetos utilizadas atualmente, identificando quais ferramentas proveem algum suporte ao processo de planejamento de custos. Também neste capítulo é apresentada uma pequena descrição de cada ferramenta avaliada.

O objetivo desta avaliação é encontrar uma ferramenta que permita evoluir e incluir funcionalidades, no sentido de implementar o processo de planejamento de custos.

3.1 Definição do estudo

Para avaliar é necessário que se estabeleça uma metodologia e critérios de avaliação bem definidos para que os resultados possam ser mais bem identificados. A definição dos critérios utilizados para avaliar as ferramentas no que se refere ao planejamento de custos será usando os principais modelos e normas na área de gerenciamento de projetos, PMBOK.

Escolha das Ferramentas

Para a seleção das ferramentas, foi realizada uma pesquisa no repositório web SourceForge por ser considerado o maior site de aplicações de código aberto . (PEREIRA; et al; 2011). Nesse repositório, foram encontrados 206 *softwares* disponíveis para *download* quando pesquisado pela frase “*Project management*”. (PEREIRA; et al; 2011). A partir daí, foram aplicados os seguintes critérios de inclusão. (PEREIRA; GONÇALVES; WANGENHEIM, 2011):

- Atualização: utilizar o ano inicial de 2008 com o objetivo de excluir ferramentas que não tiveram mais manutenção;
- Popularidade: taxa de *download* de no mínimo 50 *downloads*/semana, considerando as mais utilizadas;
- Equipe: grupo de desenvolvimento de 4 pessoas para possivelmente aumentar a garantia de continuidade do projeto;

- Foco: a ferramenta deve prover de suporte para as características tradicionais de gerenciamento de projetos.

Para a exclusão foram aplicados os seguintes critérios (PEREIRA; GONÇALVES; WANGENHEIM, 2011):

- Tecnologia: sistemas desktop que não oferecem nenhum suporte para coletar e distribuir informação pela web;
- Suporte: suporte para um grande número de processos como o gerenciamento de configuração, o rastreamento de bugs e o gerenciamento de mudanças sem oferecer um suporte devido ao gerenciamento de projetos.
- Especificidade: suporte para uma característica específica de gerenciamento de projeto, por exemplo, simulação de Monte Carlo ou funções para cálculo de esforço ou também para um contexto específico.
- Metodologia: suporte para métodos ágeis como SCRUM ou Agile.

Após a aplicação desses critérios, restaram somente cinco ferramentas, apresentadas em ordem classificatória abaixo. (PEREIRA; GONÇALVES; WANGENHEIM, 2011):

- 1.DotProject
- 2.Project.net
- 3.PhpCollab
- 4.Track+
- 5.Streber

Critérios de Avaliação

Com base nos modelos e normas referenciados neste trabalho, foi pesquisado nas ferramentas um suporte para as principais atividades do planejamento de custos. São admitidos os seguintes suportes para a avaliação:

- P1. *Estimate Cost (Estimar os Custos)*: Estimar o esforço do projeto e custo para produtos de trabalho e tarefas com base no raciocínio de estimação.

- P2. *Determine Budget (Determinar o Orçamento)*: Estabelecer e manter o orçamento do projeto e seu cronograma.

Através dos critérios de suportes levantados acima, cada item será avaliado conforme o grau de suporte oferecido pela ferramenta. Como descritos na tabela 6.

Nota	Descrição
-	Não oferece nenhum suporte
*	Oferece suporte básico, mas as funcionalidades não foram projetadas para este fim
**	Oferece suporte básico, mas as funcionalidades foram projetadas para este fim
***	Oferece suporte completo

Tabela 6 – Avaliação dos critérios. (PEREIRA; GONÇALVES; WANGENHEIM, 2011)

3.2 Avaliação das Ferramentas

3.2.1 dotProject

O dotProject como descrito em 2.4 é um *software* livre distribuído pelo licença GNU-GPL, para gerenciamento de projetos com um conjunto de características e funcionalidades para funcionar em ambientes corporativos atendendo as necessidades dos gerentes de projeto e PMOs.

A ferramenta não possui funcionalidades avançadas em finanças. As funcionalidades oferecidas suprem de forma mínima as necessidades do gerente de projeto na questão de estimativas de custos e orçamento. Para realizar a estimativa de custos é possível informar em cada atividade cadastrada no projeto um custo estimado para aquela atividade. A ação de

estimar um custo, como definido pelas melhores práticas de desenvolvimento (PMBOK), pode ser derivado de várias fontes ou técnicas, mas a ferramenta não fornece nenhum tipo de suporte para análises paramétricas, análises análogas, ou qualquer cálculo de custos.

Bem-vindo Grupo de Qualidade de Software GQS

Adicionar Atividade

lista de atividades : [visualizar este projeto](#)

Projeto: Pizzaria

Nome da Atividade *

* Indica campos requeridos

por aba : por lista

Detalhes Datas Dependências Recursos Humanos Outros Recursos Estimativas

Responsável pela Atividade GQS, Grupo de Qualidade de Software	Tipo da Atividade Unknown	Descrição: <input type="text"/>
Acesso Público	<input type="button" value="Selecione os contatos..."/>	
Endereço da Web <input type="text"/>		
Atividade Pai: Activity A	Orçamento Previsto: \$ <input type="text"/>	

* Indica campos requeridos

Figura 7 – Adicionar atividade dotProject

O dotProject não fornece nenhum tipo de soma das estimativas ou cálculo que produza um valor total estimado para o projeto, tendo o gerente de projetos que realizar esse procedimento manualmente. A ferramenta permite apenas que na criação de um projeto ou edição do mesmo seja feita a estimativa do valor total do projeto (figura 8). O sistema deixa muito a desejar na questão orçamentária de um projeto, pois para que o orçamento do projeto seja calculado deve ser feito uma soma manual dos custos das atividades, sendo necessária também a adição das reservas gerenciais e de contingência ao mesmo. É possível gerar relatórios dentro da ferramenta se o usuário instalar módulos adicionais como o “*Reports*”, porém mesmo assim será necessário todo o procedimento anterior, já que o relatório apenas lista as atividades com seus respectivos custos.

Bem-vindo Grupo de Qualidade de Software GQS Ajuda | Meus dados | A Fazer | Hoje | Sair do sistema

Novo Projeto

lista de projetos

Nome do Projeto *

Responsável pelo Projeto *

Empresa *

Internal Division *

Data Inicial

Data de Encerramento Prevista

Orçamento Previsto \$

Data de Encerramento Real Calculado Dinamicamente

Orçamento Real \$

URL

Local/URL

Prioridade *

Código *

Identificador de Cor *

Tipo de Projeto *

Situação * Progresso

Importar Atividades De:

Scale Imported Tasks:

Descrição

* Indica campos requeridos

Figura 8 – Criação/edição projeto

O sistema peca no item P1 pois contém apenas um suporte básico para esta finalidade, e já no item P2 a ferramenta deixa a desejar, pois seu orçamento tem de ser calculado totalmente externo a ferramenta.

3.2.2 Project.net

O project.Net (PROJECT.NET, 2012) é uma ferramenta *web-based* de gerenciamento de projetos. É *open source*, desenvolvida em Java com banco de dados Oracle. Esta ferramenta possui funcionalidades diferentes para um usuário cliente e um gerente de projetos. As principais funcionalidades dessa ferramenta para o cliente são (PROJECT.NET, 2012):

- Compartilhamento de documentos;
- Gerencia de formulários;
- Grupos de discussão;
- Calendário compartilhado;
- Tarefas;
- Marcos;
- *Workflow* do projeto;

- Repetição de processos;
- Entregáveis;
- Informações do projeto.

As principais funcionalidades para o gerente de projeto são (PROJECT.NET, 2011):

- Planejamento de projeto;
- Gerenciamento de portfólio de projeto;
- Rastreamento de entregáveis;
- Notificação por e-mail de qualquer mudança que ocorra em qualquer projeto;
- Relatório do status do projeto.

Segundo PROJECT.NET existe um módulo de propriedades do projeto aonde um módulo interno chamado “financial status” fornece suporte para alguns itens relacionados a estimativa de custos e obtenção do orçamento, seguintes itens:

- *Budgeted total costs*
- *Current estimated total costs*
- *Actual costs to date*
- *Estimated ROI- return of investment*

As tentativas de instalação da ferramenta apresentaram problemas na formação do banco de dados, portanto, não pudemos avaliar a ferramenta como usuários.

3.2.3 PhpCollab

O php-Collab é uma ferramenta *open source* desenvolvida em php e distribuída pela licença GPL, com foco no gerenciamento de projetos gerenciado pelo ponto de vista do time do projeto e não do gerente de projeto (PHPCOLLAB). Ela fornece informações específicas do projeto a todos os membros do time de projeto e também ao cliente, onde o mesmo pode acompanhar o andamento das atividades (PHPCOLLAB).

O processo de estimar os custos do projeto é muito limitado dentro desta ferramenta. A figura 9 mostra como um projeto é criado e a figura 10 como uma tarefa é adicionada ao projeto. Nos dois casos a ferramenta apenas permite a adição de estimativas de horas de trabalho para o projeto e para as tarefas individualizadas. Não existe nenhum tipo de entrada de dados em que se possam adicionar valores monetários aos projetos e tarefas, e sim apenas horas de trabalho.

▼ **Projecto : Pizzaria**

Detalhes

Nome : Pizzaria
 ID Projecto : 2
 Prioridade : ■ Média
 Phase Enabled : False

Descrição : Um cliente, que é dono de uma pizzaria, possui uma proposta de projeto. Atualmente ele já oferece a entrega de pizzas aos seus clientes, por meio da Internet, possam encomendar pizzas no site da sua pizzaria. Estas informações são enviadas para o cliente, visto que atualmente tem pouco conhecimento de TI. Além disso, ele também quer um módulo para dispositivos móveis, para que eles possam verificar detalhes da entrega (endereço, valor total, etc.). O dono da pizzaria também pretende desenvolver um novo projeto.

Development site url :
 Final site url :
 Dono : [Grupo de qualidade de Software \(gqs\)](#)
 Criado : 2011-10-31 17:53
 Modificado : 2011-10-31 17:57

Organização do Cliente : [UFSC](#)
 Estado : Abrir

Max file size [\[Help\]](#) : 50 KB
 Project folder size [\[Help\]](#) : -
 Tempo Estimado : 20 horas
 Tempo Actual : 0 horas
 Scope creep [\[Help\]](#) : Days
 Site em Projecto : <[Criar...](#)>
 Invoicing : False
 Hourly rate : 0.00

Support

New requests : 0 - [Manage new requests](#)
 Open requests : 0 - [Manage open requests](#)
 Complete requests : 0 - [Manage complete requests](#)

▼ **Tarefas**

<input type="checkbox"/> Nome	Prioridade	Estado	Completion ▲	Data de Finalização
<input type="checkbox"/> Teste Tarefa	■ Alta	Abrir	20 %	2011-11-07

Figura 9 – Janela de definição do projeto

▼ **Tarefa : Teste Tarefa**







Info
Projecto : Pizzaria
Organização do Cliente : UFSC
Criado : 2011-10-31 17:54
Delegado : 2011-10-31 17:54
Modificado : 2011-10-31 17:58

Detalhes
Nome : Teste Tarefa
Descrição : Testando
Delegado a : Grupo de qualidade de Software (qqs)
Estado : Abrir
Completion : 20 %
Prioridade : ■ Alta
Start date : 2011-10-31
Data de Finalização : 2011-11-07
Tempo Estimado : 20 horas
Tempo Actual : 0 horas
Publicado : Não
Invoicing : False
Worked hours : 0.00
Comentários :

Task update history
1. 2011-10-31 17:58 Estado Abrir Show details

Figura 10 – Janela de definição da tarefa

Analisando o requisito P2, a ferramenta não tem nenhum suporte em relação a orçamento do projeto. Apenas nos oferece um relatório com todas as tarefas listadas do projeto, porém sem a adição das estimativas de horas de trabalho, sem alinhamento dos custos em uma linha de tempo e também sem a possibilidade de adição de reservas tanto gerenciais como de contingência.

Portanto a ferramenta nos fornece um suporte básico para o item P1 porém a funcionalidade não foi projetada para conseguir estimar os custos do projeto e nenhum suporte ao item P2.

3.2.4 Track+

Track+ é uma ferramenta colaborativa (*software* livre sob a licença GNU *General Public License* (GPL)) *web-based* para gerenciamento de projetos e de tarefas um (TRACKPLUS, 2012). Seu foco está no acompanhamento de problemas, tarefas e ações (TRACKPLUS, 2012). Esse *software* foi desenvolvido através da linguagem de programação Java, podendo ser utilizado o banco de dados MySQL, Postgres, Microsoft SQL Server, Oracle, IBM DB2, Firebird/InterBase. Essa ferramenta possui como principais recursos (TRACKPLUS, 2012):

- *Workflow* configurável para corresponder ao seu processo;
- Gerencia qualquer tipo de lista como defeitos, casos de teste, marcos, solicitações de melhoria, requisitos, tarefas;
- Anexa arquivos que podem ser indexados;
- Notificação por e-mail automática;
- Integrado com sistemas de controle de versão;
- Trilha de auditoria completa;
- Planejamento de atividades e tarefas;
- Componentes customizáveis como campos, formulários, listas, filtros, etc;
- Emissão de relatório e consultas;
- Suporta múltiplos projetos e tipos de problemas;
- Segurança baseada em grupos e papéis
- Acompanhamento de estimativa de tempo, custo e gastos;
- Monitora e reforça regras de processo do negócio;
- Suporta os principais sistemas de banco de dados e sistemas operacionais.

Analisando a ferramenta podemos notar que diferente das outras já de início ele prevê um possível suporte a custos dentro do projeto. Na inicialização de um projeto existe a opção para levar em consideração tempo e custos, habilitando esta opção é possível adicionar o valor da hora de trabalho e a moeda usada para os possíveis cálculos, como mostrado na figura 11.

A criação de uma atividade na ferramenta nos permite introduzir estimativas de custos de esforço que serão realizados para cada atividade, fornecendo para o sistema a quantidade de horas e seu custo (figura 11). A atividade pode ser relacionada com um conjunto de tarefas que agregadas devem somar o valor total estimado ou planejado para a atividade. Cada tarefa adicionada pode ser expressa em horas ou dias de trabalho e seu custo pode ser definido no mesmo processo. Essas tarefas por sua vez contém uma descrição para explicar o que é a tarefa, porém ela pode ser deixada em branco dificultando a compreensão de futuras análises. O sistema não prevê custos de matérias e recursos não humanos, contudo ela pode ser feita de forma informal, deixando o campo de horas de trabalho em branco e adicionando ao campo descrição o nome dos recursos materiais utilizados, como a figura 12 demonstra.

Principal				Subsystem & Class		Release		Version control		Assign Roles	
Nome:	Pizzaria										
Tipo Projecto:	Generic Project			Estado Projecto:	a correr						
Gerente por defeito:	GQL, Grupo de Qualidade de Sof				Responsável por defeito:	GQL, Grupo de Qualidade de Sof					
Estado assunto inicial:	aberto			Default issue type:	Task						
Default priority:	imediate			Default severity:	não-sério						
Pre-fill by:	<input checked="" type="radio"/> Last created issue <input type="radio"/> Project defaults										
Tempo & Custo:	<input checked="" type="checkbox"/>										
				Horas por dia de trabalho:	8,00						
				Nome moeda:	Real						
				Simbolo moeda:	R\$						
Linking:	<input type="checkbox"/>										
Descrição:											

Figura 11 – Tela configuração projeto

History		Comment (0)		Attachments (0)		Worklog/Costs		Watchers	
+ Add - Apagar ✎ Edit									
Date	Changed By	Subject	Work (h)	Cost	Account	Description			
14/04/2012	GQL, Grupo de Qualidade de Sof	teste custo 2	10,00	50,00	1 - teste				
14/04/2012	GQL, Grupo de Qualidade de Sof	teste custo	24,00	100,00	1 - teste				
16/04/2012	GQL, Grupo de Qualidade de Sof			100,00	1 - teste	Material			
Total actual work: 34,00 Hours Total actual cost: 250,00 R\$									
	Work	Cost	Plan by						
Planned value (PV)	200,00 Hours	2000,00 R\$	GQL, Grupo de Qualidade de Sof! ✎						
Estimated remaining effort	100,00 Hours	1000,00 R\$	GQL, Grupo de Qualidade de Sof! ✎						
	Completion	Actual expense - Estimated remaining effort							
Work	25%	<div style="width: 25%; background-color: green; height: 10px;"></div>							
Cost	20%	<div style="width: 20%; background-color: green; height: 10px;"></div>							

Figura 12 – Tela criação de atividade

O seu comprometimento com as boas práticas de desenvolvimento referenciadas neste trabalho fica prejudicado por esses e outros fatores. O seus recursos para o item P2 ficam ainda mais precários. A ferramenta em si fornece vários tipos de relatórios, mas nenhum deles nos trás dados referentes ao orçamento do projeto. O track+ gera relatórios para o acompanhamento do projeto, como custos atuais versus custos planejados. Um relatório gerado pela ferramenta mostrado na figura 13 nos fornece o valor estimado para cada atividade, porém não fornece nenhum tipo de total ou informações de reservas. Não é possível adicionar reservas gerenciais e nem de contingência diretamente no orçamento.

Detailed Overview



Total No. of Issues: 4

Created by: GQL, Grupo de

Overdue Issues: 1

Responsibles Issues

14/04/2012 22:29

Issue No. 10 (aberto): "Desenvolvimento de aplicação para iPhone"

Project:	Pizzaria - Teste	Originator:	GQL, Grupo de	Task
	-	Priority:	logo	08/11/2011 00:56
Planned	100 Hours + 900 R\$			

Desenvolver aplicação para iPhone dos entregadores.

Issue No. 11 (aberto): "Backup"

Project:	Pizzaria - Teste	Originator:	GQL, Grupo de	Task
	teste	Priority:	imediate	08/11/2011 00:55
Planned	200 Hours + 2,000 R\$			

Fazer backup

Figura 13 – Relatório detalhado do projeto

Como definido no começo, esta ferramenta tem o foco de trabalho no controle e execução do projeto, não tendo bons suportes a planejamento, principalmente planejamento de custos. Como conclusão pode obter que a ferramenta tem um suporte básico para item P1 e nenhum suporte direto para o item P2.

3.2.5 Streber

A ferramenta Streber (STREBER) é uma ferramenta livre baseada em wiki de gerenciamento de projetos escrita em PHP. Esta ferramenta como a maioria das outras apenas provê a adição de custo através de horas de trabalho. Muito limitada, o sistema não permite fazer estimativas de nenhum tipo de recurso não humano, limitando sempre a custos referentes a horas ou dias de trabalho. Como na figura 14 a criação de uma nova atividade nos

limita a colocar o tempo estimado de execução de tal, não podendo estimar um custo monetário para ela. Acontece o mesmo para a adição de tarefas dentro da atividade.

The screenshot shows the 'New task / #0' form in the streberPM application. The form is titled 'Task (new)' and has a navigation bar with 'Home', 'Projects', 'People', 'Companies', and a search field. Below the navigation bar are tabs for 'Pizzeria', 'Tasks', 'Topics', 'Milestones', 'Versions', 'Files', 'Efforts', and 'Changes'. The form itself has a 'Name' field (highlighted in yellow), a 'Display as' dropdown menu set to 'Task', and a tabbed interface with 'Task', 'Timing', 'Description', and 'Display' tabs. Under the 'Timing' tab, there are fields for 'Estimated time' (undefined), 'Estimated worst case' (undefined), 'Planned Start' (with a calendar icon), and 'Planned End' (with a calendar icon). At the bottom, there is a checkbox labeled 'Create another task after submit' and two buttons: 'Cancel' and 'Submit'.

Figura 14 – Criação de uma nova atividade

Por esses motivos a ferramenta não suporta totalmente o item P1, sendo que para uma estimativa de custo mais precisa seria necessária a tradução das horas de trabalho de cada atividade, com base no valor da hora dos executores e gastos de recursos matérias, numa unidade em que a empresa conseguisse medir precisamente seus custos.

Para o item P2 a ferramenta contém uma espécie de soma de horas de trabalho, divididos em horas totais do projeto e horas de cada atividade. Este suporte apenas nos informa a soma de horas que existem no projeto, não podendo ser utilizado com uma linha de tempo ou adicionado ao custo total do projeto reservas gerencias e de contingência (figura 15).

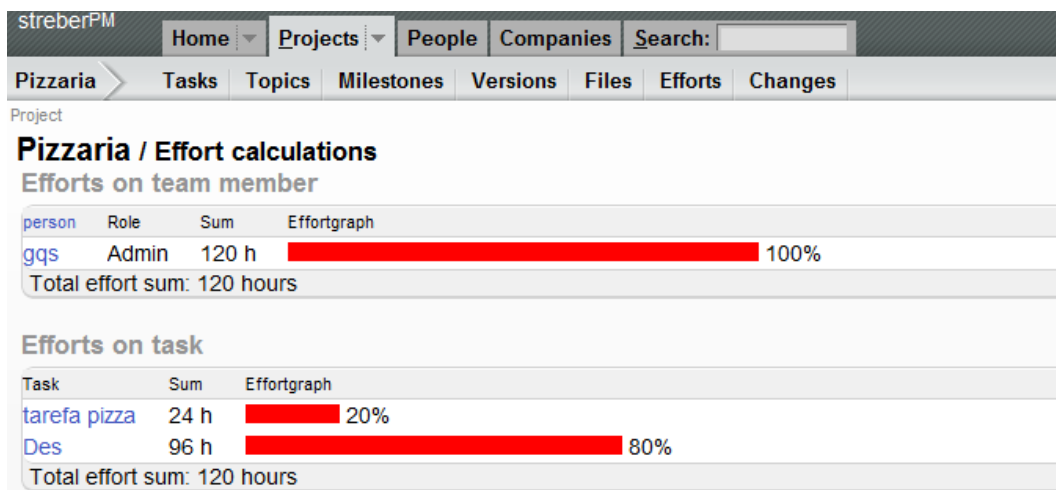


Figura 15 - Cálculos custos do projeto

Discussão

Os resultados da avaliação das cinco ferramentas são representados de acordo com as definições da Tabela 6. A tabela 7 foi criada para mostrar esses resultados. Pode-se notar que a maioria das ferramentas apresenta pouco ou nenhum suporte ao item P2 (determinar o orçamento).

	P1	P2
dotProject	*	*
projectNet	*	*
phpCollab	*	-
Track+	**	-
Streber	*	*

Tabela 7 – Resultado avaliação ferramentas

Agora em relação ao item P1 pode-se notar que todas as ferramentas suportam ao menos o básico, que seria a estimativa das atividades do projeto, mesmo elas sendo de formas limitadas. A ferramenta que melhor se saiu nesse quesito foi a ferramenta track+ que prove um suporte a estimativas de custos mais completa, fornecendo a possibilidade de acrescentar

estimativas de custo para a atividade e detalha-la em tarefas internas suportando cálculos básicos de custos.

Com essa análise foi possível ver que as ferramentas *open source* mais usados no mercado hoje em dia tem um grande déficit de suporte para planejamento de custos. Isso reforça o intuito deste trabalho de tentar melhorar o processo de planejamento de custos dentro de uma ferramenta de gerenciamento de projetos.

4. Modelagem de um processo genérico para planejamento de custos

Para evoluir a ferramenta de gerenciamento de projetos no escopo de planejamento de custos, é definido um processo genérico, alinhado ao PMBOK, com independência de ferramenta. Este modelo é criado para que MPEs consigam executar o processo de planejamento de custos, envolvendo os processos de estimar e orçar o custo do projeto. O objetivo do modelo é generalizar para facilitar o planejamento a fim de atingir as metas do projeto.

Com base nas características da MPEs e conceitos apresentados no capítulo 2, definimos as entradas e saídas para este processo. Definimos como entrada: Plano de Recursos, Registro de riscos, Cronograma do projeto, Plano de Recursos Humanos e Dados Organizacionais de Pessoal. Como saídas temos as estimativas de custos dos recursos humanos e não humanos, a *baseline* de custos e o orçamento do projeto.

Destas entradas atribuídas para o processo, identificamos que o mesmo necessita de algumas informações específicas dos recursos. Essas informações serão usadas no processo de modo a obter dados como data de aquisição, forma de pagamento entre outras. Artefatos especiais de entrada:

Plano de Recursos

Item	Especificação Técnica	Quantidade	Data de Aquisição	Forma de pagamento
Notebook	2ª Geração do Processador Intel® Core™ i5-2450M (2.5GHz até 3.1GHz com Intel® Turbo Boost 2.0, 4 Threads, 3Mb Cache)	2	15/05/2012	A vista
Iphone	iPhone 4S Apple 16GB TIM com Câmera 8MP, Touch Screen, 3G, GPS, MP3, Bluetooth e Wi-Fi - Preto	1	15/07/2012	A prazo – 3 parcelas
Servidor	Processador Intel® Xeon® Quad-Core E3-1220 (3.10GHz, 8M Cache, Turbo/4T (80W) / Memória de 2GB, 1333MHz (1X2GB UDIMM) / Disco Rígido de 500GB SATA	1	10/05/2012	A prazo – 2 parcelas

Dados Organizacionais de Pessoal

Pessoa	Horas	Custo Anual
Programador 1	40horas/semana	R\$36.000,00
Programador 2	40horas/semana	R\$25.500,00
Projetista 1	20horas/semana	R\$30.000,00

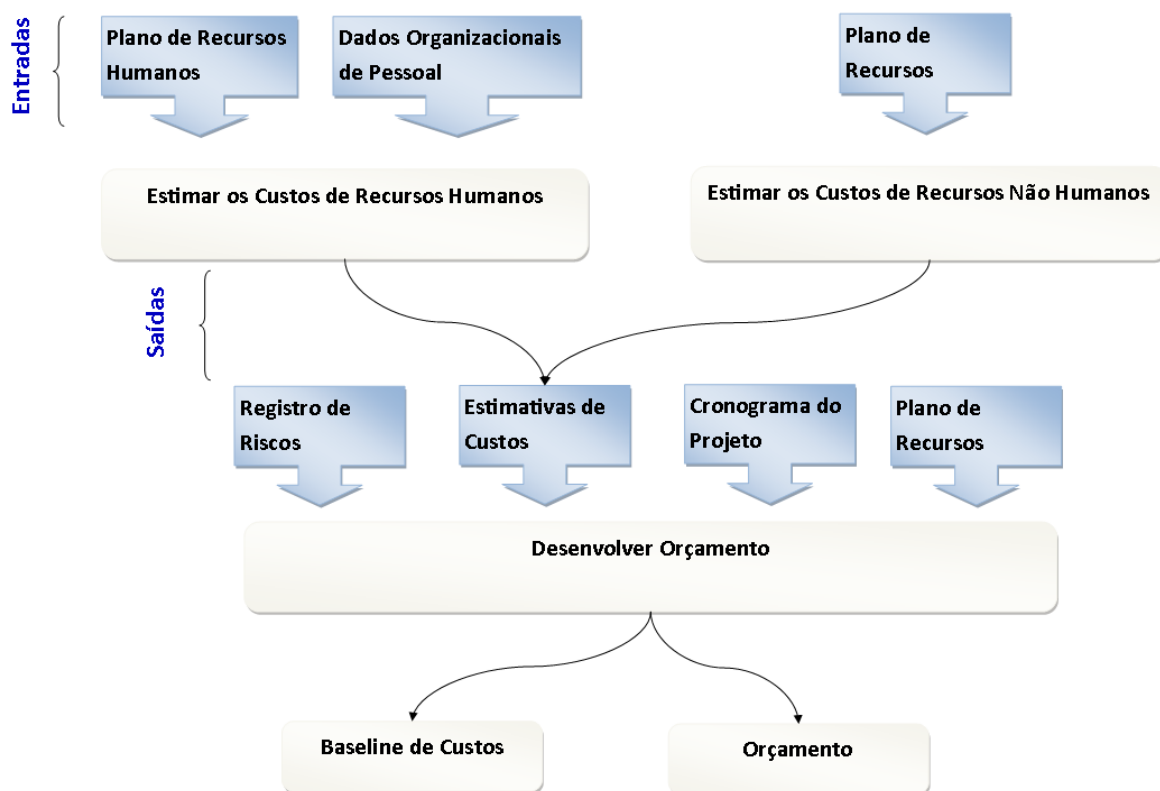


Figura 16 – Processo genérico para planejamento de custos

O modelo é composto das seguintes processos descritos abaixo.

Estimar os Custos dos Recursos Humanos
Objetivo O objetivo desta processo é estimar todos os custos que estão envolvidos com o projeto com caráter humano, ou seja, o custo de pessoal, equipe, pessoas envolvidas no desenvolvimento do projeto. Estas estimativas produzirão valores monetários.
Entradas Plano de Recursos Humanos (descrito secção 2.2.1, estimativas de RH inclusas), Dados Organizacionais de Pessoal.
Saídas Estimativas de custos dos recursos humanos.
Passos 1. Definir quais recursos humanos serão usados, adicionando os recursos ao projeto

<p>(módulo de RH), baseado na secção 2.2.</p> <ol style="list-style-type: none"> 2. Definir o custo/hora ou custo/mês do recurso, com base nos dados organizacionais de pessoal e plano de recursos humanos. 3. Definir a quantidade de esforço (pessoa/horas) que o recurso será utilizado no projeto. 4. Calcular o subtotal, multiplicando a sua quantidade de pessoa/horas (esforço) pelo seu custo (na unidade de tempo escolhida), demonstrado na secção 2.2.1. 5. Repetir os passos 2,3 e 4 para todos os recursos humanos
<p>Responsável Gerente de Projetos</p>

Tabela 8 – Detalhamento tarefa estimar os custos humanos

<p>Estimar os Custos de Recursos Não Humanos</p>
<p>Objetivo O objetivo desta tarefa é estimar todos os custos que estão envolvidos com o projeto sendo eles não humanos, produzindo como resultado estimativas monetárias.</p>
<p>Entradas Plano de Recursos.</p>
<p>Saídas Estimativas de custos, Base de estimativas.</p>
<p>Passos</p> <ol style="list-style-type: none"> 1. Definir todos os recursos não humanos a serem estimados, com base no plano de recursos. 2. Definir o valor unitário do recurso por unidade utilizando técnicas mostradas na secção 2.2 e baseado no plano de recursos. 3. Calcular o subtotal dos custos, multiplicando o valor unitário pela quantidade de unidades a serem adquiridas do recurso corrente. 4. Repetir os passos 2 e 3 para todos os recursos 5. Atualizar a base de dados de recursos (plano de recursos) acrescentando a estimativa de valor unitário do item.
<p>Responsável Gerente de Projetos</p>

Tabela 9 – Detalhamento tarefa estimar os custos não humanos

Desenvolver orçamento
<p>Objetivo</p> <p>O objetivo desta tarefa é desenvolver uma <i>baseline</i> de custos, organizando os custos numa linha de tempo do cronograma do projeto e obter o orçamento completo, incluindo suas reservas.</p>
<p>Entradas</p> <p>Estimativas de Custos, Cronograma do projeto, Registro de Riscos, Plano de Recursos.</p>
<p>Saídas</p> <p><i>Baseline</i> de custos, Orçamento.</p>
<p>Passos</p> <ol style="list-style-type: none"> 1. Alocar os custos de recursos humanos e não humanos no calendário do projeto com base nos prazos de aquisição (ou de início no projeto p/ RH) e forma de pagamento (plano de recursos). 2. Incluir a reserva de contingência <ol style="list-style-type: none"> 2.1 Para cada risco identificado com fator de exposição alto é definido se haverá uma inclusão na reserva de contingência: <ol style="list-style-type: none"> 2.2.1 Identificar o valor total da reserva (impacto) para esse item. 2.2.2 A distribuição da chance de ocorrer nos meses do projeto 3. O valor estimado é distribuído conforme a probabilidade de ocorrência do risco no calendário do projeto. 4. Somando todos os custos estimados e a reserva de contingência, é identificado o valor total da <i>baseline</i> de custos. 5. Definir um valor de reserva gerencial (% do valor da <i>baseline</i> de custos ou valor fixo). 6. Somando o valor da reserva gerencial ao valor total da <i>baseline</i> de custos é identificado o orçamento do projeto.
<p>Responsável</p> <p>Gerente de Projetos</p>

Tabela 10 – Detalhamento tarefa desenvolver orçamento

5. Evolução do DotProject

Neste capítulo é apresentado a evolução da ferramenta dotProject para o suporte ao planejamento de custos. São apresentados os requisitos e casos de uso. Com isso podemos modelar, implementar e testar a solução proposta.

5.1 Análise de Requisitos

Para o desenvolvimento do módulo da solução são identificados os requisitos funcionais e não funcionais que satisfazem o processo genérico descrito anteriormente, para que estão sejam desenvolvidas as funcionalidades do trabalho. Todo o desenvolvimento será baseado no processo descrito na secção 4, que por sua vez é baseado na literatura e principalmente no PMBOK.

5.1.1 Requisitos Funcionais

RF1. Com base nos dados cadastrados no sistema, o módulo de custos deve automaticamente calcular as estimativas de custo de RH (por recurso, por período de tempo e o total) e permitir a revisão/modificação dessas estimativas automaticamente calculadas.

RF2. O módulo deve permitir o registro de estimativas de custo unitário dos recursos não humanos.

RF3. O módulo deve calcular automaticamente os subtotais e o total dos recursos não humanos. Calcula os totais por categoria (RH e Não RH).

RF4. O módulo deve criar automaticamente a *baseline* de custos (referente aos recursos humanos e não humanos).

RF5. O módulo deve permitir registrar as estimativas de custos referentes aos riscos com prioridade alta. Permitir que o gerente informe para cada um desses riscos a estimativa de impacto financeiro e a distribuição de probabilidade.

RF6. O módulo calcula automaticamente o total da *baseline* de custos e total por mês.

RF7. O módulo deve permitir a definição de um valor de percentagem para o valor de reserva gerencial.

RF8. O módulo de orçamento calcula automaticamente o orçamento do projeto.

5.1.2 Requisitos Não Funcionais

RNF1. O módulo deverá ser implementado usando o framework de desenvolvimento do dotProject.

RNF2. Deverá ser usada a linguagem PHP e o banco de dados MySQL.

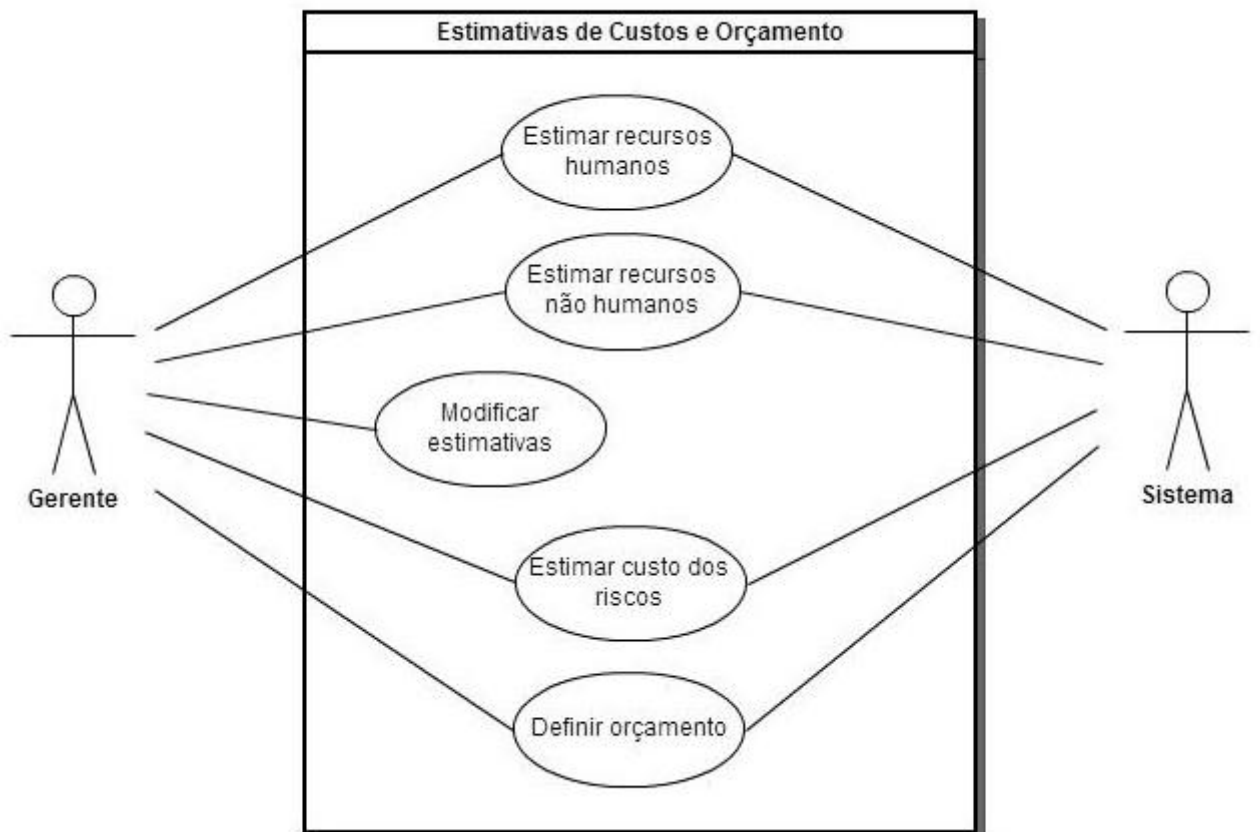
RNF3. O módulo deverá seguir os padrões de interface e usabilidade do sistema dotProject.

5.1.3 Casos de Uso

A partir dos requisitos descritos acima criamos os casos de uso apresentados a seguir.

1. Estimar recursos humanos
2. Estimar recursos não humanos
3. Modificar estimativas

4. Estimar custos dos riscos
5. Definir o orçamento



1. Estimar recursos humanos

Atores

Sistema

Pré-condições

Os dados de recursos humanos devem estar preenchidos corretamente.

Fluxo de eventos

1. O gerente de projetos acessa o módulo de estimativas.
2. O sistema mostra uma tabela com os recursos alocados para o projeto dividida em duas categorias, recursos humanos e recursos não humanos.
3. O gerente clica sobre o botão editar.
4. O gerente insere o intervalo de tempo em que o recurso ficará alocado no projeto.
5. O sistema mostra as estimativas calculadas automaticamente dos recursos humanos por item e o subtotal da categoria.

2. Estimar recursos não humanos

Atores

Gerente de Projetos

Pré-condições

Os dados de recursos não humanos devem estar preenchidos.

Fluxo de eventos

1. O gerente de projetos acessa o módulo de estimativas.
2. O sistema mostra uma tabela com os recursos alocados para o projeto dividida em duas categorias, recursos humanos e recursos não humanos.
3. O gerente preenche o campo disponível para cada recurso não humano com valor de estimativa unitário e o período de alocação do recurso.
4. O sistema mostra as estimativas calculadas automaticamente dos recursos não humanos por recurso e o subtotal da categoria.
5. O gerente clica no botão salvar.

3. Modificar estimativas

Atores

Gerente de Projeto

Pré-condições

As estimativas devem estar calculadas.

Fluxo de eventos

1. O gerente de projetos acessa o módulo de estimativas.
2. O gerente clica no botão editar localizado ao lado de cada recurso.
3. O sistema habilita a edição do período das estimativas de recursos humanos ou não humanos.
4. O gerente altera o(s) campo(s) desejado(s).
5. O gerente clica no botão salvar.

4. Estimar custo dos riscos

Atores

Gerente de Projetos

Pré-condições

Os dados referentes aos riscos devem estar preenchidos.

Fluxo de eventos

1. O gerente de projetos entra no módulo de estimativas e acessa a aba “Orçamento”.
2. O sistema mostra a *baseline* de custos em forma de uma tabela dividida em três categorias, recursos humanos, recursos não humanos e reserva de contingência.
3. O sistema lista dentro da categoria de reserva de contingência todos os riscos associados ao projeto com probabilidade alta.
4. O gerente preenche o campo disponível com o valor de impacto financeiro e seleciona os meses em que o risco tem chance de ocorrer.
5. O sistema calcula automaticamente o valor de cada item e total da categoria reserva de contingência.
6. O gerente clica no botão salvar.

5. Definir o orçamento

Atores

Gerente de Projetos

Pré-condições

A *baseline* de custos deve estar devidamente preenchida.

Fluxo de eventos

1. O gerente de projetos entra no módulo de estimativas e acessa a aba “Orçamento”.
2. O sistema mostra a *baseline* de custos em forma de uma tabela dividida em três categorias, recursos humanos, recursos não humanos e reserva de contingência.
3. O gerente clica em editar.
4. O gerente preenche o campo da reserva gerencial com uma porcentagem.
5. O gerente clica em salvar.
6. O sistema calcula o valor total do orçamento e mostra na tela.

5.2 Modelagem da solução

A modelagem da solução é construída pensando em dois componentes ou duas partes para o módulo, aonde uma é responsável pelos custos de recursos e o outro componente pela criação da *baseline* de custos demonstrando o orçamento. Esses componentes por sua vez dependem de outros módulos, aonde alguns já são do core do dotProject e outros que devem ser integrados ao sistema. A figura 17 mostra a ligação e dependência do módulo desenvolvido para com os outros.

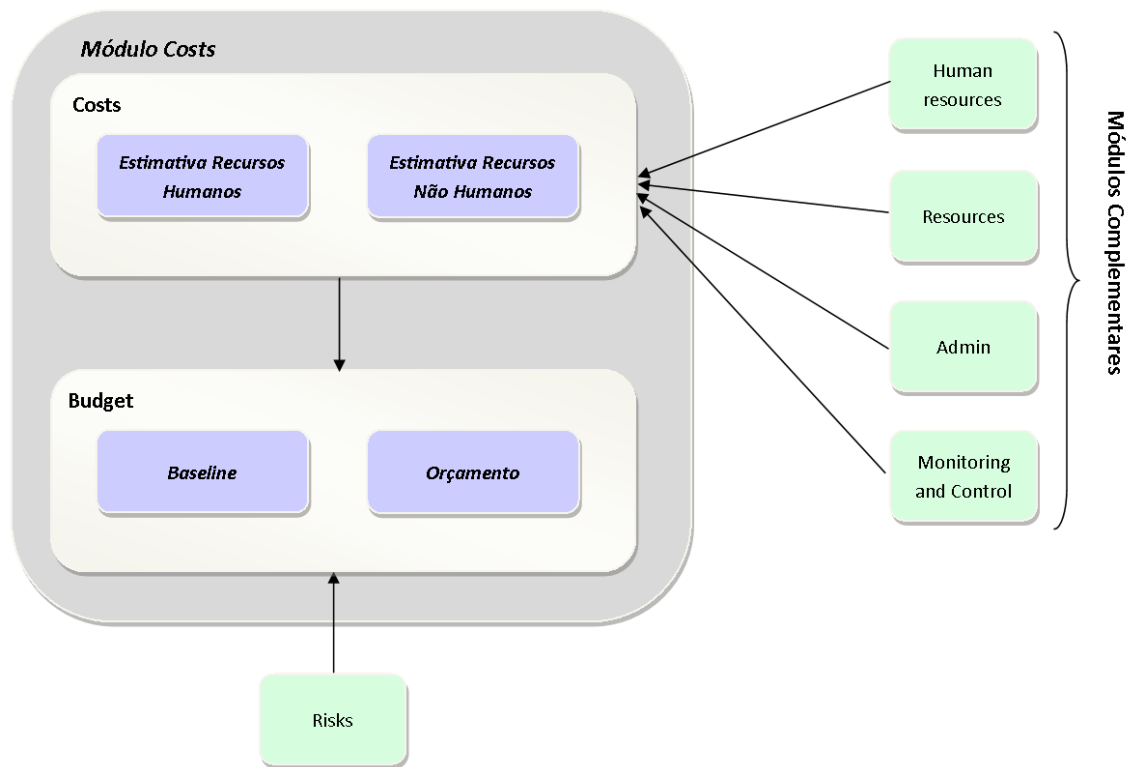


Figura 17 – Modelagem da solução

5.2.1 Modelagem da implementação

A modelagem da implementação foi desenvolvida baseada na arquitetura do dotProject. Esta arquitetura de módulos possibilita uma fácil integração de módulos desenvolvidos via o framework de desenvolvimento da ferramenta. Como mostrado na figura 18 o módulo é adicionado na arquitetura junto aos outros módulos já existentes. O sistema trabalha com o padrão *Model View Controller* (MVC), que visa separar o domínio lógico do sistema da interface do usuário.

Nesta arquitetura é modelado o diagrama de classes da figura 19 para os dois componentes do módulo, seguindo os padrões de desenvolvimentos do dotProject para que o framework faça o trabalho de integrar o módulo automaticamente.

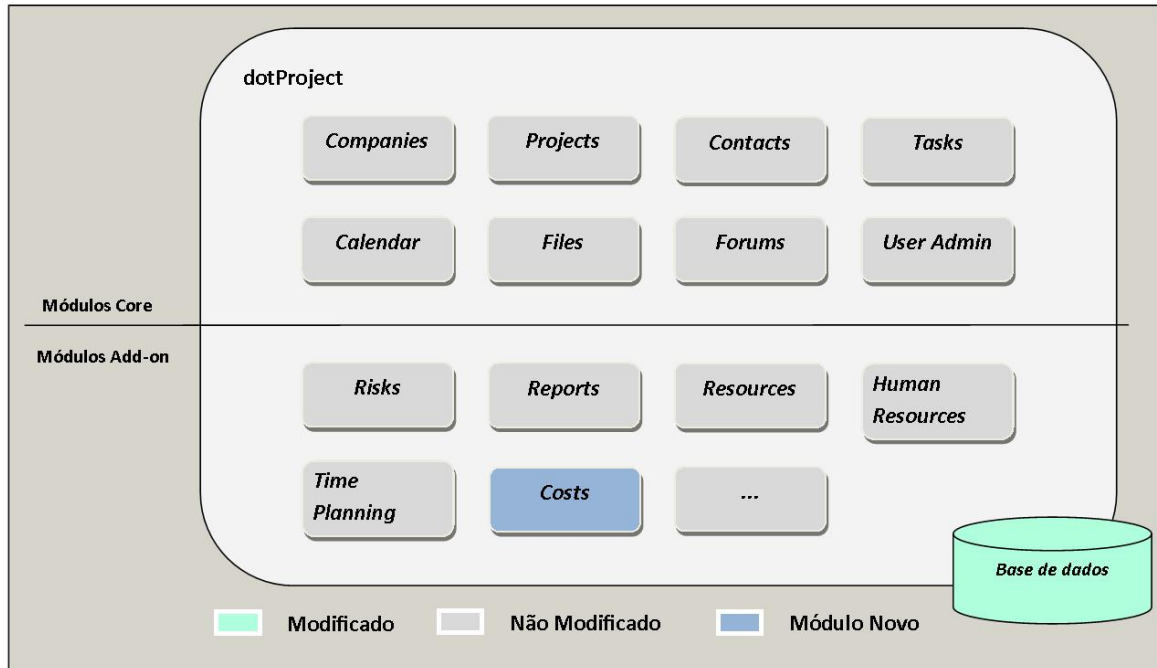


Figura 18 – Arquitetura da ferramenta dotProject com módulo Custos

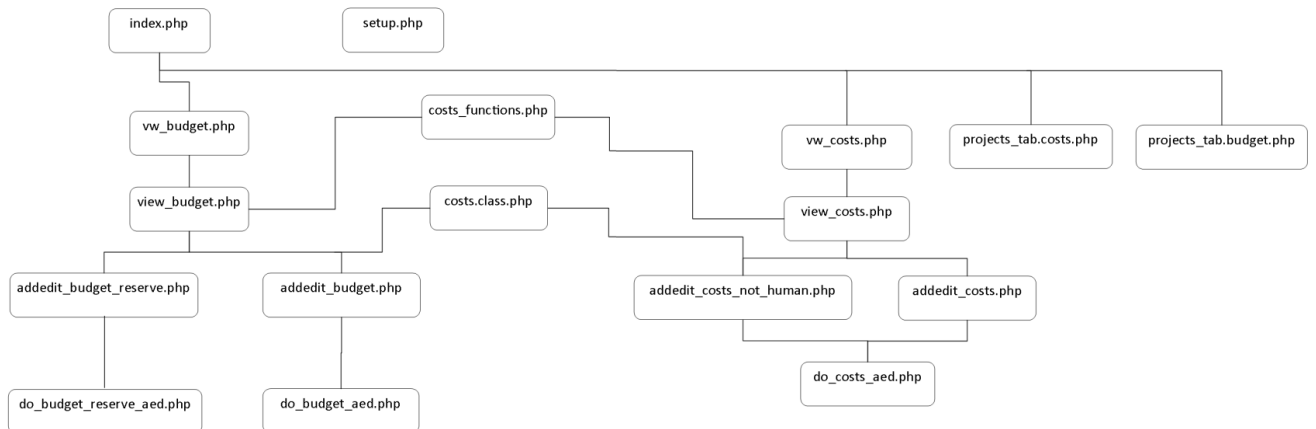


Figura 19 – Dependência entre arquivos do projeto

Para fins de análise de desenvolvimento e evolução foi introduzido uma modelagem lógica simples das tabelas criadas para o módulo relacionadas com a estrutura do banco existente.

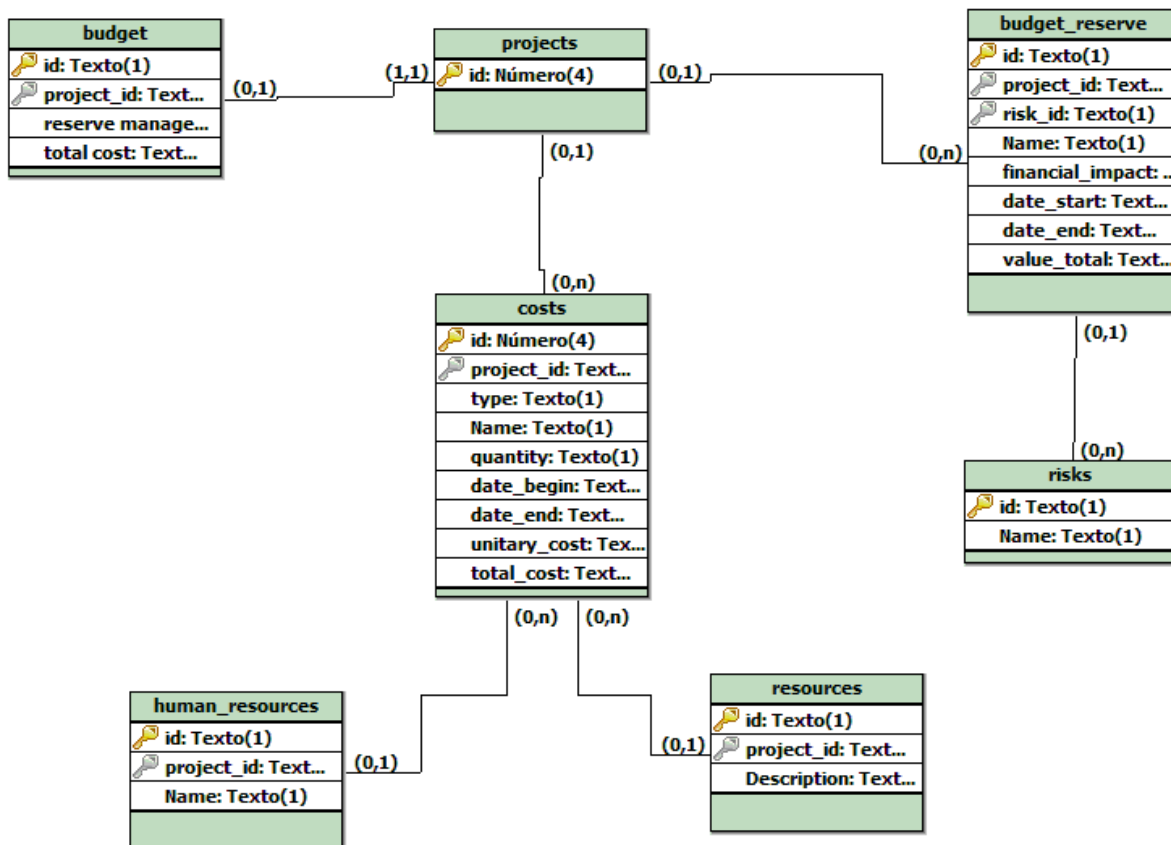


Figura 20 – Modelagem entidade relacional banco de dados

5.3 Implementação

A implementação do módulo de custos foi desenvolvido utilizando a linguagem de programação PHP e a sua interface utilizando a linguagem de marcação HTML. Para o armazenamento das informações foi utilizado o banco de dados MYSQL e o servidor web Apache. Todos esses parâmetros utilizados nos outros módulos e recomendado pelo framework do sistema. O código do módulo pode ser encontrado no apêndice B.

Abaixo são apresentadas as telas do módulo desenvolvido relacionados com os casos de uso acima especificados.

Caso de uso: Estimar recursos humanos

Para a estimativa dos custos deve-se primeiramente acessar o módulo de projetos, navegando até a aba “Costs” aonde será possível criar uma estimativa de custos. Ao clicar no botão “new cost estimative” o usuário é direcionado para a página de edição dos recursos do projeto.

The screenshot shows the 'Costs Project' interface. At the top, there is a navigation bar with tabs: Tasks, Tasks (Inactive), Forums, Gantt Chart, Task Logs, Events, Post mortem analysis, Budget, Costs, Files, Allocations, Risks. Below the navigation bar is a header with a dollar sign icon and the text 'Costs Project'. A 'new cost estimative' button is visible in the top right corner. The main content area contains two tables:

Human Resource Estimative						
Name	Date Begin	Date End	Hours/Month	Hour Cost	Total Cost	
						Subtotal Human Estimatives 0,00

Non-Human Resource Estimative						
Description	Date Begin	Date End	Quantity	Unitary Cost	Total Cost	
						Subtotal Not Human Estimatives 0,00

Figura 21 – Criar estimativas de custo

Assim que aberto será possível estimar os custos, clicando no ícone ao lado dos recursos humanos listados e assim a tela de edição estará disponível. Assim que inserido os dados e salvo, o sistema retorna para a página de estimativas de recursos.

The screenshot shows the 'Estimative Costs' interface. At the top, there is a navigation bar with tabs: Companies, Projects, Tasks, Calendar, Files, Contacts, Forums, Tickets, User Admin, System Admin, Human Resources, Risks, Resources, Closure. Below the navigation bar is a header with a dollar sign icon and the text 'Estimative Costs'. A 'back' button is visible in the bottom left corner. The main content area contains two tables:

Human Resource Estimative						
Name	Date Begin	Date End	Hours/Month	Hour Cost	Total Cost	
Pedro da Silva - Programador	01/09/2012	10/11/2012	80	25,00	4.000,00	
Roberto Carungo - Analista	01/12/2012	10/02/2013	100	10,00	2.000,00	
						Subtotal Human Estimatives 6.000,00

Non-Human Resource Estimative						
Description	Quantity	Date Begin	Date End	Unitary Cost	Total Cost	
Notebook	2	15/08/2012	15/11/2012	1.000,00	2.000,00	
						Subtotal Not Human Estimatives 2.000,00

Figura 22 – Adicionar estimativas de RH

Dados de entrada

Os dados referentes aos recursos humanos provêm do módulo adicional “Human Resources”, aonde todos os recursos humanos alocados para o projeto estarão listados. Cada

recurso humano terá um papel associado, por exemplo, programador, analista, etc. A definição de recurso humano no sistema se caracteriza pelas definições de um usuário, portanto todo recurso humano é também um usuário do sistema. As informações de recursos humanos alocados são trazidas do módulo de recursos humanos e os valores de custo hora alocados para cada usuário (recurso humano) provém do módulo adicional controle e monitoramento.

Caso de uso: Estimar recursos não humanos

A estimativa dos recursos não humanos se decorre do mesmo processo utilizado para o caso de uso de estimar recursos humanos.

The screenshot shows the 'Edit' form for non-human resources in the dotProject 2.1.5 application. The form is titled 'return resource estimatives' and includes the following fields and values:

- Name: Notebook
- Quantity: 2
- Date Begin*: 15/08/2012
- Date End*: 15/11/2012
- Unitary Value*: 1000.00
- Total Value: 2000.00

There are 'Cancel' and 'Submit' buttons at the bottom of the form. A note at the bottom center reads '* Required Fields'. The application header includes 'dotProject 2.1.5' and 'dotProject.net FREE SOFTWARE'. The navigation menu includes 'Companies', 'Projects', 'Tasks', 'Calendar', 'Files', 'Contacts', 'Forums', 'Tickets', 'User Admin', 'System Admin', 'Human Resources', 'Risks', 'Resources', and 'Closure'. The user is logged in as 'Admin Person'.

Figura 23 – Estimar recursos não humanos

Dados de entrada

Aqui a única informação que é possível ser obtida do sistema é o número de recursos alocados para o projeto, utilizando registros do módulo “*Resources*”. Este módulo nos fornece o cadastro e alocação dos recursos não humanos para cada projeto, podendo esses recursos ser alocados para cada tarefa do projeto. A contagem do número de recursos se dá pela quantidade de recursos alocados com disponibilidade de 100% para a tarefa, considerando que se pode alocar o mesmo equipamento para várias tarefas.

Caso de uso: Modificar estimativas

Os recursos humanos como não humanos podem ser editados da mesma forma. Para realizar as alterações nos recursos é necessário acessar a página de projetos na aba *Costs*

clicando no botão “*edit costs estimatives*” ou caso esteja configurado com visível o módulo “*Costs*” é possível acessar diretamente os custos através da aba *Costs* no menu superior.

Depois de acessado é necessário que seja clicado no ícone ao lado de cada recurso para seja habilitado a edição do recurso em questão.

Figura 24 – Modificar recursos humanos e não humanos

Caso de uso: Estimar custo dos riscos

Ao entrar na página de projetos navegue até a aba “*Budget*”, nesta aba será mostrado a *baseline* e orçamento. Clique no botão “*new budget*” onde será redirecionado para a página de edição do orçamento.

Year	2012						2013		
Item	Month 07	Month 8	Month 9	Month 10	Month 11	Month 12	Month 1	Month 2	Total Cost
HUMAN RESOURCE ESTIMATIVE									
Subtotal Human Estimatives	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
NON-HUMAN RESOURCE ESTIMATIVE									
Subtotal Non-Human Estimatives	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
CONTINGENCY RESERVE									
Subtotal Contingency	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
TOTAL	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Budget									
Management Reserve(%)							Subtotal Budget	Total Value	
							0,00	0,00	
Total Budget: 0,00									

Figura 25 – Criar *baseline*

Na página de orçamento na categoria reserva de contingência se encontram os riscos que serão estimados. Para edita-los deve-se clicar no ícone ao lado do nome do risco, e então a tela de edição será aberta.

dotProject 2.1.5

Companies | Projects | Tasks | Calendar | Files | Contacts | Forums | Tickets | User Admin | System Admin | Human Resources | Risks | Resources | Closure

Welcome Admin Person

Help | My Info | **Todo** | Today | Logout

Edit

return to costs baseline

Name: Quebra HD

Financial Impact*: 250

Date Begin*: 01/08/2012

Date End*: 10/12/2012

Total Value: 1250.00

* Required Fields

Cancel Submit

Figura 26 – Estimar custo dos riscos

Na edição após preencher os campos requeridos e salvar os dados serão atualizados na *baseline* de custo sendo eles distribuídos conforme período definido.

dotProject 2.1.5

Companies | Projects | Tasks | Calendar | Files | Contacts | Forums | Tickets | User Admin | System Admin | Human Resources | Risks | Resources | Closure

Welcome Admin Person

Help | My Info | **Todo** | Today | Logout

Budgets

projects budgets

Year	2012						2013		Total Cost(\$)
	Month 07	Month 8	Month 9	Month 10	Month 11	Month 12	Month 1	Month 2	
HUMAN RESOURCE ESTIMATIVE									
Pedro da Silva - Programador	0,00	0,00	1.333,33	1.333,33	1.333,33	0,00	0,00	0,00	4.000,00
Roberto Carungo - Analista	0,00	0,00	0,00	0,00	0,00	666,67	666,67	666,67	2.000,00
Subtotal Human Estimatives	0,00	0,00	1.333,33	1.333,33	1.333,33	666,67	666,67	666,67	6.000,00
NON-HUMAN RESOURCE ESTIMATIVE									
Notebook	0,00	500,00	500,00	500,00	500,00	0,00	0,00	0,00	2.000,00
Subtotal Non-Human Estimatives	0,00	500,00	500,00	500,00	500,00	0,00	0,00	0,00	2.000,00
CONTINGENCY RESERVE									
Quebra HD	0,00	250,00	250,00	250,00	250,00	250,00	0,00	0,00	1.250,00
Roubo celular	0,00	0,00	0,00	0,00	1.500,00	1.500,00	1.500,00	0,00	4.500,00
Subtotal Contingency	0,00	250,00	250,00	250,00	1.750,00	1.750,00	1.500,00	0,00	5.750,00
TOTAL	0,00	750,00	2.083,33	2.083,33	3.583,33	2.416,67	2.166,67	666,67	13.750,00
Budget									
	Management Reserve(%)					Subtotal Budget		Total Value	
	5.00					13.750,00		14.437,50	
Total Budget: 14.437,50									

back

Figura 27 – Visualizar *baseline*/orçamento

Dados de entrada

Todos os dados que são gerados para o grupo “reserva de contingência” são obtidos no módulo “*Risks*”, onde é possível filtrar apenas os riscos associados ao projeto com prioridade alta e muito alta. O módulo de custos obtém os riscos que foram alocados para o projeto em questão e os filtra para mostrar apenas os riscos com probabilidade alta de ocorrer que conseqüentemente podem afetar o projeto, gerando um impacto financeiro no orçamento.

Caso de uso: Definir Orçamento

Ao acessar a aba orçamento módulo de custos, na página de projetos, conseguimos então editar o orçamento para que seja adicionado o valor de reserva gerencial desejado. Para adicionar a percentagem do valor da *baseline* que será atribuída à reserva gerencial, deve-se editar a categoria “Budget” localizado no final da *baseline*.

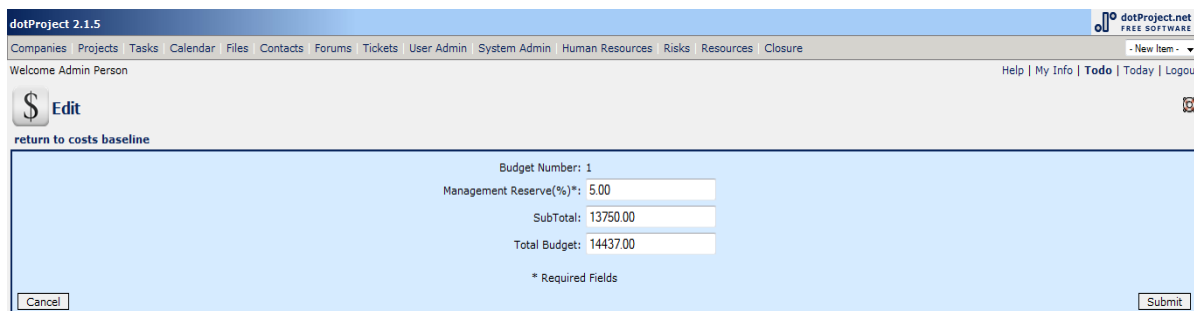


Figura 28 – Estimar reserva gerencial

Após adicionar a percentagem e salvar, o orçamento será mostrado com o novo valor calculado.

5.4 Testes

Os testes, no nível do sistema, são definidos de acordo com os casos de uso. O planejamento e execução dos testes estão esquematizados na tabela 11, que define o caso de uso a ser testado, pré e pós-condições, o procedimento usado e o status da execução.

Nº	Caso de Uso	Dados de Teste	Pré-Requisitos	Passos	Resultado Esperado	Status
1	Estimar recursos humanos	Entrada de dados fictícios (01-09-12,10-11-12,80)	Recursos humanos cadastrados	-Entrar na página de projetos e clicar na aba “Costs”; - Clicar em “new cost estimative”; - Clicar no ícone de lápis ao lado do recurso;	Recursos humanos listados e estimados automaticamente	OK - Realizado em 18/10/2012

				- Preencher os campos definidos e clicar em "submit".		
2	Estimar recursos não humanos	Entrada de dados fictícios (01-09-12,10-11-12,2500)	Uma estimativa de custos já foi iniciada e recursos cadastrados.	Navegar até a página de projetos e acessar a aba "Costs"; - Clicar no botão "edit cost estimative"; - Clicar no ícone de lápis ao lado do recurso; - Preencher os campos definidos e clicar em "submit".	As estimativas dos recursos calculadas corretamente, conseguir editar e salvar os valores para os recursos sem problemas.	OK - Realizado em 18/10/2012
3	Modificar estimativas	Entrada de dados fictícios (100(hours/month))	Uma estimativa de custos já foi iniciada.	- Navegar até a página de projetos e acessar aba "Costs"; - Clicar no botão "edit cost estimative". - Clicar no ícone de lápis ao lado do recurso; - Modificar os campos desejados e clicar em "submit".	Alterar e gravar os parâmetros permitidos sem problemas	OK - Realizado em 18/10/2012
4	Estimar custo dos riscos (reserva de contingência)	Entrada de dados fictícios (800 - impact)	Um orçamento já foi iniciado	Navegar até a página de projetos e acessar aaba "Budget"; - Clicar no botão "edit budget"; - Clicar no ícone de lápis ao lado do recurso; - Preencher o campo de impacto financeiro e clicar em "submit"	O custo unitário e total de cada risco identificado pelo sistema	OK - Realizado em 19/10/2012
5	Definir o	Entrada de dados	Um	Navegar até a	O valor total do	OK -

orçamento	fictícios (15%)	orçamento já foi iniciado	página de projetos e acessar a aba “Budget”;	de orçamento já calculado utilizando os valores de reserva	já Realizado em 19/10/2012
			- Clicar no botão “edit budget”;		
			- Clicar no ícone de lápis ao lado do recurso;		
			- Preencher o campo de reserva gerencial e clicar em “submit”		

Tabela 11 – Plano de testes do sistema

Os testes foram realizados pelo autor deste trabalho, executando todos os testes definidos na tabela 11 que priorizam as funcionalidades do módulo de custos desenvolvido. Na parte do módulo que se definem os custos dos recursos humanos e não humanos foi identificado que os recursos são buscados automaticamente do cadastro de recursos humanos e não humanos e que para apagá-los é necessário desassociar o recurso do projeto, deste modo o recurso pode então ser removido das estimativas. O mesmo ocorre no módulo de riscos onde os riscos associados ao projeto sempre apareceram na reserva de contingência, podendo ser removido somente quando for desassociado do projeto (dentro do cadastro de riscos). Nenhum erro aparente foi encontrado na execução dos testes.

5.5 Publicação

Este módulo desenvolvido foi disponibilizado na internet há 12 dias e obteve 87 *downloads*. A descrição do módulo e o link para *download* pode ser encontrado no fórum internacional dotProject (<http://forums.dotproject.net/showthread.php?p=46423#post46423>).

6. Avaliação

Esta seção apresenta uma avaliação inicial do módulo desenvolvido visando a avaliação da usabilidade/utilidade do:

1. Módulo desenvolvido
2. Grau do suporte fornecido alinhado ao processo do planejamento de custo do alinhamento ao PMBOK.

6.1 Avaliação por painel de especialistas

Para a identificação das medidas a serem levantadas, foi adotado o método GQM – *Goal/Question/Metric* (BASILI; et al.,1994) - um método de medição de *software* para identificar sistematicamente os objetivos da pesquisa e derivar suas medidas. Segundo o método GQM, a partir dos objetivos da avaliação, são definidas questões e medidas para definirem os dados a serem coletados e para guiarem a análise. Os objetivos da avaliação levantados são:

Objetivo 1 Analisar a utilidade do módulo para realizar a estimativa de recursos humanos e não humanos.

Objetivo 2 Analisar a utilidade do módulo na criação de um orçamento para o projeto.

Objetivo 3 Identificar os pontos fortes e fracos da solução proposta.

A partir dos objetivos definidos foram então identificadas as questões para medição, mostradas na tabela 12.

Objetivos	Questões
1. Analisar a utilidade do módulo para realizar a estimativa de recursos humanos e não humanos.	1.1 As informações geradas na criação da estimativa dos recursos são suficientes para análise? 1.2 O grau de detalhamento /especificação dos custos (por recurso separadamente) é útil? 1.3 O módulo é útil para a identificação dos custos de

	recursos humanos e não humanos do projeto? 1.4 O módulo é suficientemente automatizado para fornecer o suporte ao registro de estimativas?
2. Analisar a utilidade do módulo na criação de um orçamento para o projeto.	2.1 As informações geradas na criação do orçamento são suficientes para análise? 2.2 O módulo representa de forma útil o <i>baseline</i> de custos do projeto? 2.3 O módulo é suficientemente automatizado para fornecer o suporte a criação da <i>baseline</i> /orçamento? 2.4 O grau de detalhamento /especificação dos custos (por ex. por categoria) é útil? 2.5 Os parâmetros de entrada para a definição do orçamento são suficientes?
3. Identificar os pontos fortes e fracos da solução proposta.	3.1 Quais foram os principais pontos fortes que você observou? 3.2 Quais são as principais sugestões de melhoria?

Tabela 12 – Questões para avaliação

A partir das questões identificadas foi criado um questionário usando uma escala Likert de 5 pontos, onde 0 representa “discordo totalmente” e 5 “concordo totalmente” referente as questões dos objetivos 1 e 2 e respostas abertas para as questões referente ao objetivo 3. O formulário criado pode ser encontrado no Apêndice A.

6.1.1 Execução da avaliação

Foram convidados 5 especialistas para participar da avaliação. Eles foram escolhidos devido a proximidade e disponibilidade de participação no curto prazo da pesquisa. A avaliação foi realizada de forma remota durante o mês de outubro de 2012. Durante a execução, foi disponibilizado o sistema dotProject em um servidor e criado usuários na aplicação para cada avaliador. Os participantes receberam instruções sobre o que é a pesquisa e qual o procedimento da avaliação. A avaliação foi feita com base no processo de planejamento de custos definido no presente trabalho, utilizando dados de um projeto fictício. Após a avaliação os profissionais responderam um formulário de avaliação (Apêndice A) sobre o módulo.

6.1.2 Análise dos dados

A análise dos dados é feita por meio da computação das respostas dos participantes da avaliação. Podemos observar mais profundamente que os itens implementados obtiveram uma utilidade boa para as funções propostas.

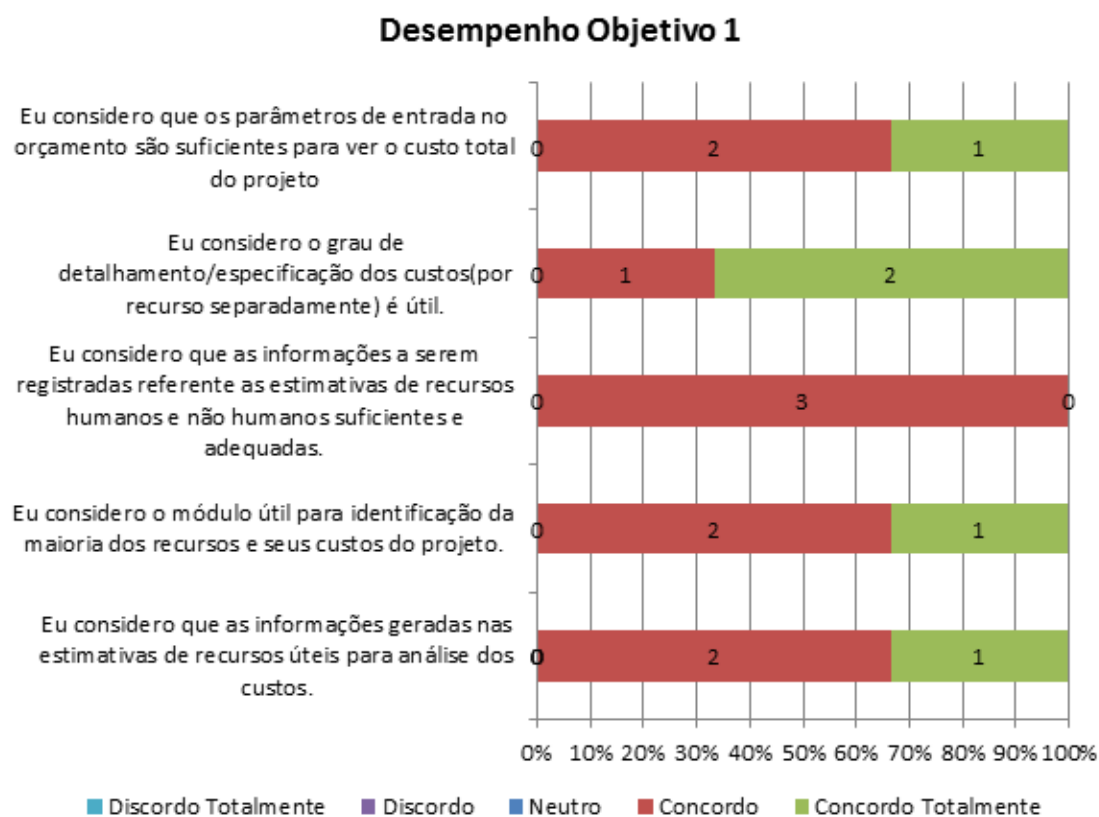


Figura 29 – Análise dados objetivo 1

Como podemos observar na figura 29, a parte do módulo responsável pela montagem dos recursos humanos e não humanos mostrou que para os avaliadores estes itens estão sendo úteis para a análise e identificação dos recursos do projeto. Sendo uma parte importante e crucial para o demonstrativo do orçamento, esta atividade se comportou bem em relação a automatização e grau de detalhamento das informações.

O objetivo 2 já obteve resultados mais modestos, devido a sua maior complexidade. Foi possível observar com a avaliação e comentários dos avaliadores que as informações

apresentadas poderiam ser mais flexíveis a ponto de serem editadas manualmente e não produzidas automaticamente pelo sistema. Isso trás uma possível discussão de várias possibilidades e barreiras, que serão introduzidas no item de discussão deste capítulo. A figura 30 mostra os resultados obtidos desta avaliação.

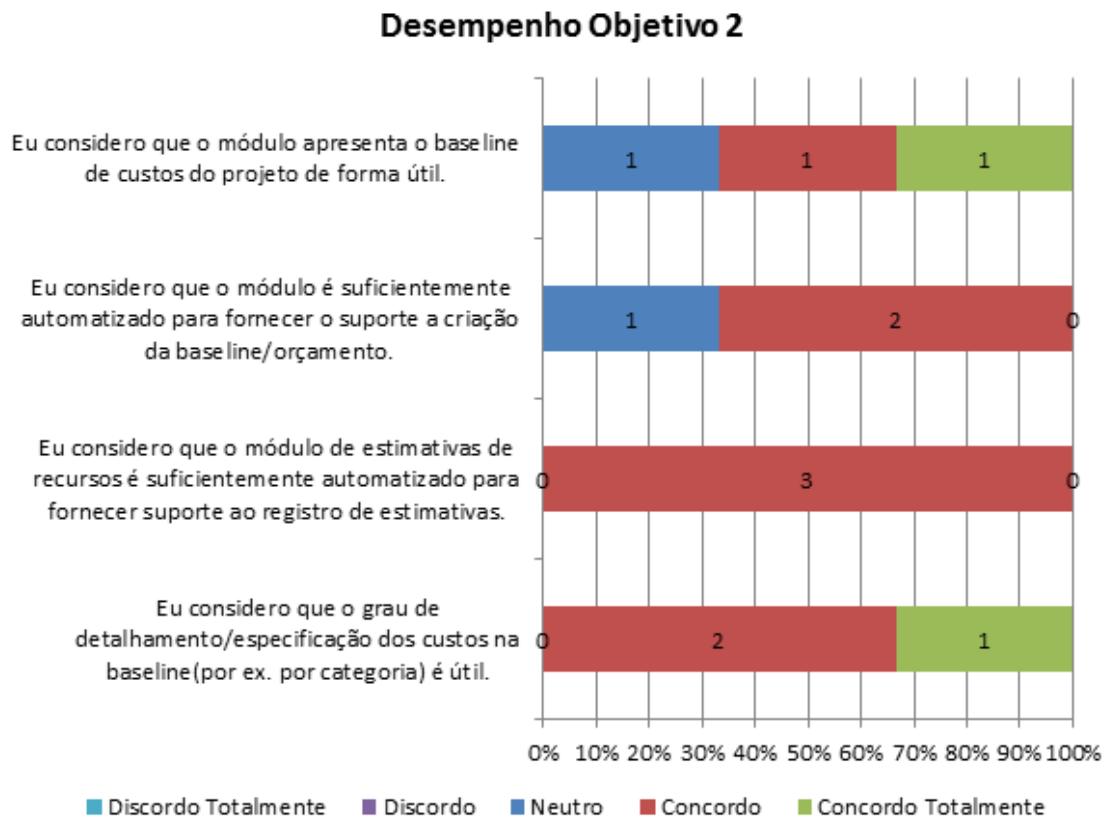


Figura 30 – Análise dados objetivo 2

6.2 Avaliação em relação ao alinhamento ao PMBOK

A fim de avaliar se o módulo desenvolvido está alinhado com as boas práticas recomendadas pelo PMBOK para o encerramento de projetos, é realizada uma avaliação. Esta avaliação segue o mesmo procedimento adotado na revisão do estado da arte, conforme descrito no capítulo 3. Essa avaliação foi realizada pelo autor deste trabalho, utilizando os mesmos critérios apresentados nas avaliações das 5 ferramentas de GP no capítulo 3.

Os resultados obtidos são mostrados na tabela 13 e como podemos observar houve uma evolução do módulo, mas mesmo assim não atende completamente a todos os itens. Para o item P1, observa-se que o módulo supre quase todo o requisitado, pois os custos do projeto são descritos e custeados de forma a poder observar os custos subtotais e totais por item. O que deixa de ser apresentado pelo módulo são os custos separados por atividades, que poderiam ajudar na precisão da estimativa de custo total do projeto, porém ao mesmo tempo na prática muitas vezes não é feito, em projetos razoavelmente pequenos.

	P1	P2
dotProject com o módulo de custos	**	***
dotProject	*	*
projectNet	*	*
phpCollab	*	-
Track+	**	-
Streber	*	*

Tabela 13 – Tabela de avaliação das ferramentas com o módulo de custos

No item P2 o grande avanço foi a criação de uma *baseline* de custos e a visualização do valor de orçamento calculado automaticamente. Antes o sistema não obtinha nenhum cálculo ou assistente de custo do orçamento, apenas um valor de estimativa de orçamento, sem a possibilidade de observar o que estava sendo estimado realmente. Como um item complementa o outro o item descrito acima vale para este também, ou seja, as estimativas de custo das atividades irão impactar na estimativa do orçamento.

Agora um orçamento poderá ser alterado a qualquer momento, sendo possível que no decorrer do projeto os valores e quantidades mudem fazendo com que a *baseline* e o valor do orçamento se adaptem a necessidade exigida pelo projeto.

6.3 Discussão

A partir da avaliação do painel de especialista e avaliação em relação ao alinhamento ao PMBOK, pode-se dizer que após a primeira avaliação o módulo deu indicações de que pode ser útil para o planejamento de custos. Apontando como principais ganhos a possibilidade de criação de uma estimativa de custos para o projeto, que antes não era possível e ainda mais os cálculos de custos produzidos automaticamente pelo sistema requisitando o mínimo de esforço do usuário. Agora também acreditamos que pode ser evoluído em algumas partes para que o usuário tenha mais liberdade de edição e customização dos custos computados no orçamento.

Porém, é importante levar em consideração que esta primeira avaliação pode apresentar ameaças de validade devido às formas de avaliação utilizadas. Primeiramente na avaliação por especialistas foi utilizado um projeto fictício em um escopo pequeno e restrito, isso pode gerar uma quantidade de informações que muitas vezes não condiz com um projeto real, tornando a visualização facilitada para este projeto. Outro ponto é a realização da avaliação por um grupo pequeno de pessoas próximas que por sua vez são envolvidas em grupos de pesquisa, podendo assim apresentar alguma tendência nos resultados. Na avaliação em relação ao alinhamento ao PMBOK pode-se considerar uma avaliação de caráter subjetiva, já que as conclusões foram obtidas pelo próprio autor do trabalho.

Com isso podemos afirmar que a generalização dos resultados fica limitada e que para obter um melhor desempenho na avaliação é necessário realizar uma avaliação mais ampla. Uma avaliação que englobe várias empresas, com um número maior de pessoas envolvidas, utilizando projetos com características mais realistas tornariam os resultados mais concretos para uma possível generalização.

7. Conclusão

Na parte inicial deste trabalho foi realizado um estudo teórico em gerenciamento de projeto baseado no PMBOK (2008), focando os esforços para o planejamento de custos. Foi apresentado o estado da arte a partir da avaliação do nível de suporte de ferramentas *open*

source de gerenciamento de projetos, em relação ao planejamento de custos do projeto. Com base nos estudos e estado da arte foi criado um processo genérico de planejamento de custos. A partir deste processo foi implementado um módulo de suporte ao planejamento de custos dentro da ferramenta dotProject. Uma avaliação inicial da evolução da ferramenta indica a utilidade e o alinhamento ao PMBOK.

Espera-se que este trabalho possa contribuir, principalmente para micro e pequenas empresas, no planejamento de custos dos projetos. E com isso contribuir para a adoção de processos sistemáticos de GP em MPE, aumentando as chances de sucesso de seus projetos e dessa forma aumentar as suas chances de sobrevivência/competividade e crescimento no mercado.

Como trabalhos futuros são possíveis identificar alguns itens baseados nas avaliações realizadas. Para a evolução do módulo é possível modificar a *baseline* de custos, tornando alguns itens listados na mesma editáveis a fim de suprir algumas particularidades de cada projeto. O módulo também pode ser evoluído no quesito integração com outros módulos, para que as informações se tornem mais centralizadas e completas, facilitando o cadastramento e usabilidade do sistema como um todo.

Referências

ABES, ASSOCIAÇÃO BRASILEIRA DAS EMPRESAS DE *SOFTWARE*. Mercado Brasileiro de *Software*: Panorama e Tendências. São Paulo: Associação Brasileira Das Empresas de *Software*, 2010. Disponível em: <<http://www.abes.org.br>>.

ALVES, Renner Librelato Domingos, Gerenciamento Dos Custos, Brasil, 2010.

AZANHA, José. As características de um gerente de projeto. São Paulo, fev. 2003.

BASILI, V. R.; CALDIERA, G.; ROMBACH, H. D. The experience factory. 1994.

DESTRI, Mauro José Teixeira. Porque e para que “Contabilidade para não contadores”>. Vol.1 2008

DOTPROJECT. Brasil – Comunidade dotProject Brasil. Disponível em: <http://www.dotproject.com.br/>>.

DOTPROJECT. dotProject Wiki. Disponível em: <<http://docs.dotproject.net/index.php?title=Main Page>>.

DUTRA, René Gomes. Custos: uma abordagem prática. 4. ed. São Paulo: Atlas, 1995

ECONOMICS, about.com. Need.Know.Accomplish – The New York Times Company. Disponível em: <http://economics.about.com> . [Acesso em: 26 abril 2012].

FAGUNDES, Jair Antônio. Metodologia para a Contabilidade de Custos. Portal ADM. Disponível em <<http://www.portaladm.adm.br>> Acesso em 06 maio.2012.

GARCIA, Claudio Antônio. Projetos de Máquinas. São Paulo: UNIBAN – Universidade Bandeirantes de São Paulo, 2008.

MARTINS, Eliseu. Contabilidade de custos. 8.ed. São Paulo: Atlas, 2001.

NETO , Oscar Guimarães – Análise de Custos. Curitiba : IESDE Brasil S.A,2008.

FERREIRA, José Antônio Stark. Contabilidade de custos. São Paulo: Pearson Prentice Hall, 2007.

FILHO, Armando Terribili. Gestão de Custos e de Riscos em Projetos. Disponível em <http://www.ogerente.com.br>.

ZANLUCA, Júlio César. Contabilidade de Custos. Portal de Contabilidade. Disponível em: <http://www.portaldecontabilidade.com.br>. Acesso em 06 maio. 2012.

GRESSE VON WANGENHEIM, C.; HAUCK, J. C. R.; VON WANGENHEIM, A. Enhancing Open Source *Software* in Alignment with CMMI-DEV. *IEEE Software*, vol. 26, no. 2, Março/Abril 2009.

XAVIER, C. M. da S. Gerenciamento de projetos: Como definir e controlar o escopo do projeto.

2. ed. [S.l.: s.n.], 2009.

PEREIRA, A. M.; GONÇALVES, R. Q.; WANGENHEIM, C. G. von. Comparação de ferramentas open source para gerência de projetos. 2011.

PEREIRA, André Marques. Comparação de ferramentas open source para monitoramento e controle de projetos. Dissertação de Mestrado, Pós-graduação - Universidade Federal de Santa Catarina, Florianópolis, 2012.

PHPCOLLAB. PhpCollab. Disponível em: <http://www.phpcollab.com/blog/>.

PMI, PROJECT MANAGEMENT INSTITUTE. A GUIDE TO THE PROJECT MANAGEMENT BODY OF KNOWLEDGE. 4. ed. Pennsylvania: Project Management Institute, 2008.

PMI, PROJECT MANAGEMENT INSTITUTE - CHAPTERS BRASILEIRO. Estudo de Benchmarking em Gerenciamento de Projetos Brasil. Brasil: Project Management Institute, 2011. Disponível em: <<http://www.pmsurvey.org>>.

PROJECT.NET. Project.net. Disponível em: <<http://www.project.net/>>.

PROJECT.NET. Project Planning Overview. Disponível em: <doc.project/9_0_UG:Planning_Main>

REZENDE, José Antônio. Contabilidade e Análise de Custos: Fundamentos Básicos. 2011

SEBRAE. Lei Geral das Micro e Pequena Empresa, Brasília, 2007.

STREBER. Streber. Disponível em: <<http://www.streber-pm.org/>>.

TRACK+. Track+. Disponível em: <<http://www.trackplus.com/>>.

VARGAS, Ricardo Viana. Reservas de Contingência X Reserva Gerencial. Disponível em <<http://www.ricardo-vargas.com>>

WANGENHEIM, Christiane Gresse Von et al. Best practice fusion of CMMI-DEV v1.2 (PP,PMC, SAM) and PMBOK 2008. *Information And Software Technology*, p. 749-757. 28 mar.2010.

WEBER, Sergio; HAUCK, Jean Carlo Rossa; WANGENHEIM, Christiane Gresse Von. Estabelecendo Processos de *Software* em Micro e Pequenas Empresas. In: SBQS – Simpósio Brasileiro de Qualidade de *Software*, Porto Alegre, Brazil, 2005.

APÊNDICE A – Formulário de avaliação do módulo de Custos por painel de especialistas

Questionário de avaliação - Planejamento de Custos

Este questionário é parte do nosso trabalho de desenvolvimento de um módulo para o encerramento de projetos, para a ferramenta de gerenciamento de projetos dotProject. O trabalho vem sendo realizado pelo graduando Rafael Reiter de Albuquerque, sob a orientação da Prof. rer. nat. Christiane Gresse von Wangenheim, PMP do GQS – Grupo de Qualidade de Software do INCoD - Instituto Nacional para Convergência Digital ([http:// www.incod.ufsc.br](http://www.incod.ufsc.br)).

Nós gostaríamos de saber a sua opinião sobre a utilidade e usabilidade do módulo (acessível em: <http://www.exe.inf.ufsc.br/~reiter/dotproject/>). A implementação atual representa um primeiro protótipo do módulo, que implementa as funções básicas deste processo. Nós gostaríamos de convidá-lo a registrar, editar e visualizar um orçamento. Isto não deve tomar mais do que 15 minutos do seu tempo. Sua participação é totalmente voluntária.

As informações obtidas deste questionário serão tratadas de forma acumulativa, não permitindo a identificação de respostas individuais. Os resultados serão utilizados para melhorar o módulo atual assim como para futuros trabalhos.

O roteiro de execução desta atividade pode ser acessado em http://www.inf.ufsc.br/~reiter/roteiro_atividade_dotProject.html

Se você tem algum questionamento, por favor, entre em contato conosco: Rafael Reiter de Albuquerque (reiter@inf.ufsc.br).

Agradecemos a participação no nosso trabalho,

Rafael Reiter de Albuquerque e Prof. rer. nat. Christiane Gresse von Wangenheim

* Required

Eu considero que as informações geradas nas estimativas de recursos úteis para análise dos custos. *

Discordo totalmente

1 2 3 4 5

Discordo totalmente Concordo totalmente

Eu considero o grau de detalhamento/especificação dos custos(por recurso separadamente) é útil. *

Discordo totalmente

1 2 3 4 5

Discordo totalmente Concordo totalmente

Eu considero que as informações a serem registradas referente as estimativas de recursos humanos e não humano suficientes e adequadas. *

Discordo totalmente

1 2 3 4 5

Discordo totalmente Concordo totalmente

Eu considero que o módulo de estimativas de recursos é suficientemente automatizado para fornecer suporte ao registro de estimativas. *

Discordo totalmente

1 2 3 4 5

Discordo totalmente Concordo totalmente

Eu considero o módulo útil para identificação da maioria dos recursos e seus custos do projeto. *

Discordo totalmente

1 2 3 4 5

Discordo totalmente Concordo totalmente

Eu considero que o módulo apresenta o baseline de custos do projeto de forma útil. *

Discordo totalmente

1 2 3 4 5

Discordo totalmente Concordo totalmente

Eu considero que o grau de detalhamento/especificação dos custos na baseline (por ex. por categoria) é útil. *

Discordo totalmente

1 2 3 4 5

Discordo totalmente Concordo totalmente

Eu considero que o módulo é suficientemente automatizado para fornecer o suporte a criação da baseline/orçamento. *

Discordo totalmente

1 2 3 4 5

Discordo totalmente Concordo totalmente

Eu considero que os parâmetros de entrada no orçamento são suficientes para ver o custo total do projeto *

Discordo totalmente

1 2 3 4 5

Discordo totalmente Concordo totalmente

Quais foram os principais pontos fortes que você observou? *

Quais são as principais sugestões de melhoria? *

Powered by [Google Docs](#)

[Report Abuse](#) - [Terms of Service](#) - [Additional Terms](#)

APÊNDICE B – Código fonte módulo de custos

setup.php

```
<?php

if (!defined('DP_BASE_DIR')) {
    die('You should not access this file directly.');
```

```
}

/**
 * Name: Costs
 * Directory: costs
 * Version 1.0
 * Type: user
 * UI Name: Costs
 * UI Icon: ?
```

```

*/
$config = array();
$config['mod_name'] = 'Costs'; // name the module
$config['mod_version'] = '1.0.1'; // add a version number
$config['mod_directory'] = 'costs'; // tell dotProject where to find this module
$config['mod_setup_class'] = 'SSetupCosts'; // the name of the PHP setup class (used
below)
$config['mod_type'] = 'user'; //'core' for standard dP modules, 'user' for additional
modules from dotmods
$config['mod_ui_name'] = 'Costs'; // the name that is shown in the main menu of the
User Interface
$config['mod_ui_icon'] = 'costs.png'; // name of a related icon //TODO
$config['mod_description'] = 'Costs Plan'; // some description of the module //TODO
$config['mod_config'] = false; // show 'configure' link in viewmods
$config['permissions_item_table'] = 'costs'; // tell dotProject the database table name
$config['permissions_item_field'] = 'cost_id'; // identify table's primary key (for
permissions)
$config['permissions_item_label'] = 'cost_name'; // identify "title" field in table

if (@$a == 'setup') {
    echo dPshowModuleConfig($config);
}

class SSetupCosts {

    function configure() {
        return true;
    }

    function install() {

        $ok = true;
        $q = new DBQuery;

```

```

$sql = "(
        cost_id integer not null auto_increment,
        cost_type_id integer not null,
        cost_project_id integer not null,
        cost_description varchar(50) not null,
        cost_quantity int(11),
        cost_date_begin datetime default null,
        cost_date_end datetime default null,
        cost_value_unitary decimal(9,2),
        cost_value_total decimal (9,2),
        primary key (cost_id),
        foreign key (cost_project_id) references `dotp_projects`(project_id)
    )ENGINE=MYISAM";

```

```
$q->createTable('costs');
```

```
$q->createDefinition($sql);
```

```
$ok = $ok && $q->exec();
```

```
$q->clear();
```

```

$sql = "(
        budget_reserve_id integer not null auto_increment,
        budget_reserve_project_id integer not null,
        budget_reserve_risk_id integer not null,
        budget_reserve_description varchar(50),
        budget_reserve_financial_impact int(11),
        budget_reserve_inicial_month datetime default null,
        budget_reserve_final_month datetime default null,
        budget_reserve_value_total decimal(9,2),
        primary key (budget_reserve_id),
        foreign key (budget_reserve_project_id) references
`dotp_projects`(project_id),
        foreign key (budget_reserve_risk_id) references `dotp_risks` (risk_id)
    )ENGINE=MYISAM";

```

```
$q->createTable('budget_reserve');
$q->createDefinition($sql);
$ok = $ok && $q->exec();
$q->clear();
```

```
$sql = "(
            budget_id integer not null auto_increment,
            budget_project_id integer not null,
            budget_reserve_management decimal(9,2)not null,
            budget_sub_total decimal(9,2) not null,
            budget_total decimal(9,2) not null,
            primary key (budget_id),
            foreign key (budget_project_id) references
`dotp_projects`(project_id)
        )ENGINE=MYISAM";
```

```
$q->createTable('budget');
$q->createDefinition($sql);
$ok = $ok && $q->exec();
$q->clear();
```

```
if (!$ok) {
    return false;
}
return null;
}
```

```
function remove() {
    $q = new DBQuery();
    $q->dropTable('costs');
    $q->exec();
    $q->clear();
}
```

```

        $q->dropTable('budget_reserve');
        $q->exec();
        $q->clear();

        $q->dropTable('budget');
        $q->exec();
        $q->clear();
    }

    function upgrade($old_version) {

        return true;
    }

}

```

index.php

```

<?php
if (!defined('DP_BASE_DIR')) {
    die('You should not access this file directly.');
}

$appUI->savePlace();

// retrieve any state parameters
if (isset($_REQUEST['project_id'])) {
    $appUI->setState('CostsIdxProject', intval($_REQUEST['project_id']));
}

$project_id = $appUI->getState('CostsIdxProject') !== NULL ? $appUI->getState('CostsIdxProject') : 0;

```



```

if (dPgetParam($_GET, 'tab', -1) != -1) {
    $AppUI->setState('CostsIdxTab', intval(dPgetParam($_GET, 'tab')));
}
$stab = $AppUI->getState('CostsIdxTab') !== NULL ? $AppUI->getState('CostsIdxTab') : 0;
$active = intval(!$AppUI->getState('CostsIdxTab'));

require_once($AppUI->getModuleClass('projects'));

$extra = array();
$project = new CProject();
$projects = $project->getAllowedRecords($AppUI->user_id, 'project_id,project_name',
'project_name', null, $extra);
$projects = arrayMerge(array('0' => $AppUI->_('All', UI_OUTPUT_JS)), $projects);

// setup the title block
$titleBlock = new CTitleBlock('Cost Estimatives and Budget', 'costs.png', $m, "$m.$a");
$titleBlock->show();

$stabBox = new CTabBox('?m=costs', DP_BASE_DIR . '/modules/costs/', $stab);
$stabBox->add('vw_costs', 'Estimative Costs');
$stabBox->add('vw_budget', 'Budget');
$stabBox->show();

?>

```

addedit_buget.php

```

<?php
if (!defined('DP_BASE_DIR')) {
    die('You should not access this file directly.');
}

$budget_id = intval(dPgetParam($_GET, 'budget_id', 0));
$projectSelected = intval(dPgetParam($_GET, 'project_id'));

```

```

// check permissions for this record
$canEdit = getPermission($m, 'edit', $budget_id);
if (!(($canEdit && $budget_id) || ($canAuthor && !($budget_id)))) {
    $AppUI->redirect('m=public&a=access_denied');
}

$q = new DBQuery();
$q->addQuery('*');
$q->addTable('budget');
$q->addWhere('budget_id = ' . $budget_id);
//$project_id = $q->loadList();
// check if this record has dependancies to prevent deletion
$msg = "";
$obj = new CBudget();

// load the record data
$obj = null;
if (!db_loadObject($q->prepare(), $obj) && ($budget_id > 0)) {
    $AppUI->setMsg('Budget');
    $AppUI->setMsg("invalidID", UI_MSG_ERROR, true);
    $AppUI->redirect();
}

// setup the title block
$title = $budget_id ? "Edit" : "Add";
$titleBlock = new CTitleBlock($title, 'costs.png', $m, "$m.$a");

if ($budget_id != 0) {
    $titleBlock->addCrumb("?m=costs&a=view_budget&project_id=" .
        $budget_id . '&budget_id=' . $budget_id, 'return to costs baseline');
}

```

```

$titleBlock->show();
?>
<script language="javascript">
    function submitIt() {

        var f = document.uploadFrm;

        var msg = "";
        var foc=false;
        if (f.budget_reserve_management.value < 0) {
            msg += "\nPlease enter a valid value to management reserve";
            if ((foc==false) && (navigator.userAgent.indexOf('MSIE')== -1)) {
                f.budget_reserve_management.focus();
                foc=true;
            }
        }

        if (msg.length < 1) {
            f.submit();
        } else {
            alert(msg);
        }

    }

    function delIt() {
        if (confirm("<?php echo $AppUI->_('Delete this budget?', UI_OUTPUT_JS);
?>")) {
            var f = document.uploadFrm;
            f.del.value='1';
            f.submit();
        }
    }
}

```

```

    }

    function budgetTotal(){
        var management =
document.getElementById('budget_reserve_management').value;
        var subtotal = <?php echo $obj->budget_sub_total ?>;
        var total = (management/100) * subtotal;
        total = total + subtotal;

        document.getElementById('budget_total').value = total;

    }
</script>

```

```

<table width="100%" border="0" cellpadding="3" cellspacing="3" class="std"
name="threads" charset=UTF-8>

```

```

<form name="uploadFrm" action="?m=costs" method="post">
    <input type="hidden" name="dosql" value="do_budget_aed" />
    <input type="hidden" name="del" value="0" />
    <input type="hidden" name="budget_id" value="<?php echo $budget_id; ?>" />

    <tr>
        <td width="100%" valign="top" align="center">
            <table cellspacing="1" cellpadding="2" width="60%">
                <tr>
                    <td align="right" nowrap="nowrap"><?php echo $AppUI->_('Budget
Number '); ?>:</td>
                    <td>
<?php
echo dPformSafe($obj->budget_id);
?>

                </td>
            </table>

```

```

        </tr>
        <tr>
            <td align="right" nowrap="nowrap"><?php echo $AppUI->_('Management Reserve(%)'); ?>*</td>
            <td>
                <input
                    name="budget_reserve_management"
                    id="budget_reserve_management" value="<?php echo dPformSafe($obj->budget_reserve_management); ?>" />
            </td>
        </tr>
        <tr>
            <td align="right" nowrap="nowrap"><?php echo $AppUI->_('SubTotal'); ?></td>
            <td>
                <input name="budget_sub_total" id="budget_sub_total" value="<?php echo dPformSafe($obj->budget_sub_total); ?>" readonly="readonly" />
            </td>
        </tr>
        <tr>
            <td align="right" nowrap="nowrap"><?php echo $AppUI->_('Total Budget'); ?></td>
            <td>
                <input name="budget_total" id="budget_total" value="<?php echo dPformSafe($obj->budget_total); ?>" readonly="readonly" />
            </td>
        </tr>
    </table>
    <br>
    * <?php echo $AppUI->_('Required Fields'); ?>
</tr>
<tr>
<td>

```

```

        <input class="button" type="button" name="cancel" value="<?php echo
$AppUI->_('Cancel'); ?>" onClick="javascript:if (confirm('<?php echo $AppUI-
->_('Are you sure you want to cancel?', UI_OUTPUT_JS); ?>')) {history.back(-1);}"/>
        </td>
        <td align="right">
            <input type="button" class="button" value="<?php echo $AppUI-
->_('Submit'); ?>" onclick="budgetTotal();submitIt();" />
        </td>
    </tr>
</form>
</table>

```

addedit_budget_reserve.php

```

<?php
if (!defined('DP_BASE_DIR')) {
    die('You should not access this file directly.');
```

}

require_once DP_BASE_DIR . "/modules/costs/costs_functions.php";

\$budget_reserve_id = intval(dPgetParam(\$_GET, 'budget_reserve_id', 0));
\$projectSelected = intval(dPgetParam(\$_GET, 'project_id'));

\$q = new DBQuery();
\$q->addQuery('*');
\$q->addTable('budget_reserve');
\$q->addWhere('budget_reserve_id = ' . \$budget_reserve_id);
//\$project_id = \$q->loadList();
// check if this record has dependancies to prevent deletion
\$msg = "";
\$obj = new CBudgetReserve();
\$canDelete = \$obj->canDelete(\$msg, \$budget_reserve_id);

```

// load the record data
$obj = null;
if (!db_loadObject($q->prepare(), $obj) && ($budget_reserve_id > 0)) {
    $AppUI->setMsg('Budget');
    $AppUI->setMsg("invalidID", UI_MSG_ERROR, true);
    $AppUI->redirect();
}

$q->clear();
$q->addQuery('project_start_date,project_end_date');
$q->addTable('projects');
$q->addWhere("project_id = '$projectSelected'");
$datesProject = & $q->exec();
$dateSP = substr($datesProject->fields['project_start_date'], 0, -9);
$dateTemp = substr($datesProject->fields['project_end_date'], 0, -9);
$dateEP = (string)$dateTemp;

// format dates
$date_begin = intval($obj->budget_reserve_inicial_month) ? new CDate($obj->budget_reserve_inicial_month) : null;
$date_end = intval($obj->budget_reserve_final_month) ? new CDate($obj->budget_reserve_final_month) : null;
$df = $AppUI->getPref('SHDATEFORMAT');

// setup the title block
$title = $budget_reserve_id ? "Edit" : "Add";
$titleBlock = new CTitleBlock($title, 'costs.png', $m, "$m.$a");

$scanDelete = getPermission($m, 'delete', $budget_reserve_id);
if ($scanDelete && $budget_reserve_id > 0) {

```

```

$titleBlock->addCrumbDelete('delete contingency', $canDelete, $msg);
}

if ($budget_reserve_id != 0) {
    $titleBlock->addCrumb('?m=costs&a=view_budget&project_id='
    $projectSelected . '&budget_id=' . $projectSelected, 'return to costs baseline');
}
$titleBlock->show();
?>

<!-- import the calendar script -->
<script type="text/javascript" src="<?php echo DP_BASE_URL;
?>/lib/calendar/calendar.js"></script>
<!-- import the language module -->
<script type="text/javascript" src="<?php echo DP_BASE_URL;
?>/lib/calendar/lang/calendar-<?php echo $AppUI->user_locale; ?>.js"></script>

<script language="javascript">
    function submitIt() {

        var f = document.uploadFrm;
        //f.submit();

        var trans = "<?php echo $dateEP; ?>";
        var str1 = String(trans);
        var str2 = document.getElementById("budget_reserve_final_month").value;

        var yr1 = parseInt(str1.substring(0,4),10);
        var mon1 = parseInt(str1.substring(5,7),10);
        var dt1 = parseInt(str1.substring(8,10),10);

        var yr2 = parseInt(str2.substring(0,4),10);

```



```
var mon2 = parseInt(str2.substring(4,6),10);
var dt2 = parseInt(str2.substring(6,8),10);
```

```
var date1 = new Date(yr1, mon1, dt1);
var date2 = new Date(yr2, mon2, dt2);
if(date2 > date1)
{
    alert("'Date end' cannot be greater than date end of the project");
    msg = "Date end cannot be greater than date end project";
    return false;
}
```

```
var msg = "";
var foc=false;
if (f.budget_reserve_financial_impact.value == 0 ||
f.budget_reserve_financial_impact.value < 0) {
    msg += "\nPlease enter a valid value to financial impact(Greater than 0)";
    if ((foc==false) && (navigator.userAgent.indexOf('MSIE')== -1)) {
        f.budget_reserve_financial_impact.focus();
        foc=true;
    }
}
```

```
if (msg.length < 1) {
    f.submit();
} else {
    alert(msg);
}
```

```
}
```

```
function delIt() {  
    if (confirm("<?php echo $AppUI->_('Delete this Contingency cost?',  
UI_OUTPUT_JS); ?>")) {  
        var f = document.uploadFrm;  
        f.del.value='1';  
        f.submit();  
    }  
}
```

```
function monthDiff(d1, d2) {  
    var months;  
    months = (d2.getFullYear() - d1.getFullYear()) * 12;  
    months -= d1.getMonth() + 1;  
    months += d2.getMonth();  
    return months;  
}
```

```
function sumTotalValue()  
{  
    var FI = document.getElementById('budget_reserve_financial_impact').value;  
    var date1 = document.getElementById('budget_reserve_inicial_month').value;  
    var date2 = document.getElementById('budget_reserve_final_month').value;  
    var total = 0;
```

```
    var year1 = date1.substring(0,4);  
    var month1 = date1.substring(4,6);  
    var day1 = date1.substring(6);
```

```
    var year2 = date2.substring(0,4);  
    var month2 = date2.substring(4,6);  
    var day2 = date2.substring(6);
```

```
var diffMonths = monthDiff(new Date(year1,month1,day1),new
Date(year2,month2,day2));
```

```
var diff_date = new Date(year2,month2,day2) - new Date(year1,month1,day1) ;
var num_months = (diff_date % 31536000000)/2628000000;
```

```
if(diffMonths < 0)
```

```
total = FI;
```

```
else
```

```
total = FI * (Math.floor(num_months)+1);
```

```
document.getElementById('budget_reserve_value_total').value = total;
```

```
}
```

```
function popCalendar( field ){
```

```
calendarField = field;
```

```
idate = eval( 'document.uploadFrm.budget_' + field + '.value' );
```

```
window.open(
```

```
'index.php?m=public&a=calendar&dialog=1&callback=setCalendar&date=' + idate,
'calwin', 'width=280, height=250, scrollbars=no' );
```

```
}
```

```
/**
```

```
* @param string Input date in the format YYYYMMDD
```

```
* @param string Formatted date
```

```
*/
```

```
function setCalendar( idate, fdate ) {
```

```
fld_date = eval( 'document.uploadFrm.budget_' + calendarField );
```

```
fld_fdate = eval( 'document.uploadFrm.' + calendarField );
```

```
fld_date.value = idate;
```

```
fld_fdate.value = fdate;
```

```

// set end date automatically with start date if start date is after end date
if (calendarField == 'reserve_inicial_month') {
    if( document.uploadFrm.reserve_final_month.value < idate) {
        document.uploadFrm.budget_reserve_final_month.value = idate;
        document.uploadFrm.reserve_final_month.value = fdate;
    }
}
}
}
</script>

```

```

<table width="100%" border="0" cellpadding="3" cellspacing="3" class="std"
name="threads" charset=UTF-8>

```

```

<form name="uploadFrm" action="?m=costs" method="post">
    <input type="hidden" name="dosql" value="do_budget_reserve_aed" />
    <input type="hidden" name="del" value="0" />
    <input type="hidden" name="budget_reserve_id" value="<?php echo
$budget_reserve_id; ?>" />

```

```

<tr>
    <td width="100%" valign="top" align="center">
        <table cellspacing="1" cellpadding="2" width="60%">
            <tr>
                <td align="right" nowrap="nowrap"><?php echo $AppUI->_('Name');
?>:</td>
                <td>
<?php
echo dPformSafe($obj->budget_reserve_description);
?>
                </td>
            </tr>
            <tr>

```

```

        <td align="right" nowrap="nowrap"><?php echo $AppUI-
>_('Financial Impact'); ?>*</td>
        <td>
            <input name="budget_reserve_financial_impact"
id="budget_reserve_financial_impact" value="<?php echo dPformSafe($obj-
>budget_reserve_financial_impact); ?>" />
        </td>
    </tr>
    <tr>
        <td align="right" nowrap="nowrap"><?php echo $AppUI->_('Date
Begin'); ?>*</td>
        <td>
            <input type="hidden" name="budget_reserve_inicial_month"
id="budget_reserve_inicial_month" value="<?php echo (($date_begin) ?
$date_begin->format(FMT_TIMESTAMP_DATE) : ""); ?>"/>
            <!-- format(FMT_TIMESTAMP_DATE) -->
            <input type="text" class="text" name="reserve_inicial_month"
id="date0" value="<?php echo (($date_begin) ? $date_begin->format($df) : ""); ?>"
disabled="disabled" />

            <a href="#" onclick="popCalendar( 'reserve_inicial_month',
'reserve_inicial_month');">
                
            </a>
        </td>
    </tr>
    <tr>
        <td align="right" nowrap="nowrap"><?php echo $AppUI->_('Date
End'); ?>*</td>
        <td>

```

```
<input type="hidden" name="budget_reserve_final_month"
id="budget_reserve_final_month" value="<?php echo (($date_end) ? $date_end-
>format(FMT_TIMESTAMP_DATE) : ""); ?>" />
```

```
<!-- format(FMT_TIMESTAMP_DATE) -->
```

```
<input type="text" class="text" name="reserve_final_month"
id="date1" value="<?php echo (($date_end) ? $date_end->format($df) : ""); ?>"
disabled="disabled" />
```

```
<a href="#" onclick="popCalendar( 'reserve_final_month',
'reserve_final_month');">
```

```

```

```
</a>
```

```
</td>
```

```
</tr>
```

```
<tr>
```

```
<td align="right" nowrap="nowrap"><?php echo $AppUI->_('Total
Value'); ?></td>
```

```
<td>
```

```
<input name="budget_reserve_value_total"
id="budget_reserve_value_total" value="<?php echo dPformSafe($obj-
>budget_reserve_value_total); ?>" readonly="readonly" />
```

```
</td>
```

```
</tr>
```

```
</table>
```

```
<br>
```

```
* <?php echo $AppUI->_('Required Fields'); ?>
```

```
</tr>
```

```
<tr>
```

```
<td>
```

```

        <input class="button" type="button" name="cancel" value="<?php echo
$AppUI->_('Cancel'); ?>" onClick="javascript:if (confirm('<?php echo $AppUI-
->_('Are you sure you want to cancel?', UI_OUTPUT_JS); ?>')) {history.back(-1);}"/>
        </td>
        <td align="right">
            <input type="button" class="button" value="<?php echo $AppUI-
->_('Submit'); ?>" onclick="sumTotalValue();submitIt();" />
        </td>
    </tr>
</form>
</table>

```

?>

addedit_costs.php

```

<?php

if (!defined('DP_BASE_DIR')) {
    die('You should not access this file directly.');
```

```

}
```

```

require_once DP_BASE_DIR . "/modules/costs/costs_functions.php";
```

```

$cost_id = intval(dPgetParam($_GET, 'cost_id', 0));
```

```

$project_id = intval(dPgetParam($_GET, 'project_id', 0));
```

```

// check permissions for this record
```

```

$scanEdit = getPermission($m, 'edit', $cost_id);
```

```

if (!(($scanEdit && $cost_id) || ($scanAuthor && !($cost_id)))) {
```

```

    $AppUI->redirect('m=public&a=access_denied');
```

```

}
```

```

$q = new DBQuery();
$q->addQuery('*');
$q->addTable('costs');
$q->addWhere('cost_id = ' . $cost_id);

// check if this record has dependancies to prevent deletion
$msg = "";
$obj = new CCosts();
$canDelete = $obj->canDelete($msg, $cost_id);

// load the record data
$obj = null;
if (!db_loadObject($q->prepare(), $obj) && ($cost_id > 0)) {
    $AppUI->setMsg('Estimative Costs');
    $AppUI->setMsg("invalidID", UI_MSG_ERROR, true);
    $AppUI->redirect();
}

/* transform date to dd/mm/yyyy */
$date_begin = intval($obj->cost_date_begin) ? new CDate($obj->cost_date_begin) :
null;
$date_end = intval($obj->cost_date_end) ? new CDate($obj->cost_date_end) : null;
$df = $AppUI->getPref('SHDATEFORMAT');

/* Get date end project*/
$q->clear();
$q->addQuery('project_start_date,project_end_date');
$q->addTable('projects');
$q->addWhere("project_id = '$project_id'");
$datesProject = & $q->exec();
$dateTemp = substr($datesProject->fields['project_end_date'], 0, -9);
$dateEP = (string)$dateTemp;

```



```

// setup the title block
$ttl = $cost_id ? "Edit" : "Add";
$titleBlock = new CTitleBlock($ttl, 'costs.png', $m, "$m.$a");

if ($scanDelete && $cost_id > 0) {
    $titleBlock->addCrumbDelete('delete human resource', $scanDelete, $msg);
}

if ($cost_id != 0) {
    $titleBlock->addCrumb(
        '?m=costs&a=view_costs&project_id='.$project_id, 'return resource
estimates' );
}
$titleBlock->show();

?>
<!-- import the calendar script -->
<script type="text/javascript" src="<?php echo DP_BASE_URL;
?>/lib/calendar/calendar.js"></script>
<!-- import the language module -->
<script type="text/javascript" src="<?php echo DP_BASE_URL;
?>/lib/calendar/lang/calendar-<?php echo $AppUI->user_locale; ?>.js"></script>

<script language="javascript">
    function submitIt() {

        var f = document.uploadFrm;
        //f.submit();

        var trans = "<?php echo $dateEP; ?>";
        var str1 = String(trans);
        var str2 = document.getElementById("cost_date_end").value;

```

```
var yr1 = parseInt(str1.substring(0,4),10);
var mon1 = parseInt(str1.substring(5,7),10);
var dt1 = parseInt(str1.substring(8,10),10);
```

```
var yr2 = parseInt(str2.substring(0,4),10);
var mon2 = parseInt(str2.substring(4,6),10);
var dt2 = parseInt(str2.substring(6,8),10);
```

```
var date1 = new Date(yr1, mon1, dt1);
var date2 = new Date(yr2, mon2, dt2);
if(date2 > date1)
{
    alert("'Date end' cannot be greater than date end of the project");
    msg = "Date end cannot be greater than date end project";
    return false;
}
```

```
var msg = "";
var foc=false;
if (f.cost_quantity.value == 0 || f.cost_quantity.value < 0) {
    msg += "\nPlease enter a valid value to hours per month(Greater than 0)";
    if ((foc==false) && (navigator.userAgent.indexOf('MSIE')== -1)) {
        f.budget_reserve_financial_impact.focus();
        foc=true;
    }
}
```

```
if (msg.length < 1) {
    f.submit();
} else {
    alert(msg);
}

}
```

```
function delIt() {
    if (confirm("<?php echo $AppUI->_('Delete this human resource estimative?',
UI_OUTPUT_JS); ?>")) {
        var f = document.uploadFrm;
        f.del.value='1';
        f.submit();
    }
}
```

```
function monthDiff(d1, d2) {
    var months;
    months = (d2.getFullYear() - d1.getFullYear()) * 12;
    months -= d1.getMonth() + 1;
    months += d2.getMonth();
    return months;
}
```

```
function sumTotalValue()
{
    var VU = document.getElementById('cost_value_unitary').value;
    var HM = document.getElementById('cost_quantity').value;
    var date1 = document.getElementById('cost_date_begin').value;
    var date2 = document.getElementById('cost_date_end').value;
    var total = 0;
```

```

var year1 = date1.substring(0,4);
var month1 = date1.substring(4,6);
var day1 = date1.substring(6);

var year2 = date2.substring(0,4);
var month2 = date2.substring(4,6);
var day2 = date2.substring(6);

var diffMonths = monthDiff(new Date(year1,month1,day1),new
Date(year2,month2,day2));

var diff_date = new Date(year2,month2,day2) - new Date(year1,month1,day1) ;
var num_months = (diff_date % 31536000000)/2628000000;

if(diffMonths < 0)
    total = VU * HM;
else
    total = (VU * HM) * (Math.floor(num_months)+1);

document.getElementById('cost_value_total').value = total;
}

function popCalendar( field ){
    calendarField = field;
    idate = eval( 'document.uploadFrm.cost_' + field + '.value' );
    window.open(
'index.php?m=public&a=calendar&dialog=1&callback=setCalendar&date=' + idate,
'calwin', 'width=280, height=250, scrollbars=no' );
}

/**
 * @param string Input date in the format YYYYMMDD
 * @param string Formatted date

```

```

*/
function setCalendar( idate, fdate ) {
    fld_date = eval( 'document.uploadFrm.cost_' + calendarField );
    fld_fdate = eval( 'document.uploadFrm.' + calendarField );
    fld_date.value = idate;
    fld_fdate.value = fdate;

    // set end date automatically with start date if start date is after end date
    if (calendarField == 'cost_date_end') {
        if( document.uploadFrm.date_end.value < idate) {
            document.uploadFrm.cost_date_end.value = idate;
            document.uploadFrm.date_end.value = fdate;
        }
    }
}
</script>

<table width="100%" border="0" cellpadding="3" cellspacing="3" class="std"
name="threads" charset=UTF-8>

<form name="uploadFrm" action="?m=costs" method="post">
    <input type="hidden" name="dosql" value="do_costs_aed" />
    <input type="hidden" name="del" value="0" />
    <input type="hidden" name="cost_id" value="<?php echo $cost_id; ?>" />

    <tr>
        <td width="100%" valign="top" align="center">
            <table cellspacing="1" cellpadding="2" width="60%">
                <tr>
                    <td align="right" nowrap="nowrap"><?php echo $AppUI->_( 'Name' );
?>:</td>

                <td>

                    <?php

```

```

        echo dPformSafe($obj->cost_description);
    ?>
</td>
</tr>
<tr>
    <td align="right" nowrap="nowrap"><?php echo $AppUI->_('Date
Begin'); ?>*</td>
    <td>
        <input type="hidden" name="cost_date_begin"
id="cost_date_begin" value="<?php echo (($date_begin) ? $date_begin-
>format(FMT_TIMESTAMP_DATE) : ""); ?>"/>
        <!-- format(FMT_TIMESTAMP_DATE) -->
        <input type="text" class="text" name="date_begin" id="date0"
value="<?php echo (($date_begin) ? $date_begin->format($df) : ""); ?>"
disabled="disabled" />

        <a href="#" onclick="popCalendar( 'date_begin', 'date_begin');">
            
        </a>
    </td>
</tr>
<tr>
    <td align="right" nowrap="nowrap"><?php echo $AppUI->_('Date
Begin'); ?>*</td>
    <td>
        <input type="hidden" name="cost_date_end" id="cost_date_end"
value="<?php echo (($date_end) ? $date_end->format(FMT_TIMESTAMP_DATE) :
"); ?>"/>
        <!-- format(FMT_TIMESTAMP_DATE) -->
        <input type="text" class="text" name="date_end" id="date1"
value="<?php echo (($date_end) ? $date_end->format($df) : ""); ?>"
disabled="disabled" />

```

```

        <a href="#" onclick="popCalendar( 'date_end', 'date_end');">
            
        </a>
    </td>
</tr>
<tr>
    <td align="right" nowrap="nowrap"><?php echo $AppUI->_('Hours
per Month'); ?>*</td>
    <td>
        <input name="cost_quantity" id="cost_quantity" value="<?php
echo dPformSafe($obj->cost_quantity);?>" />
    </td>
</tr>
<tr>
    <td align="right" nowrap="nowrap"><?php echo $AppUI->_('Unitary
Value'); ?>:</td>
    <td>
        <input name="cost_value_unitary" id="cost_value_unitary"
value="<?php echo dPformSafe($obj->cost_value_unitary); ?>"
readonly="readonly"/>
    </td>
</tr>
<tr>
    <td align="right" nowrap="nowrap"><?php echo $AppUI->_('Total
Value'); ?>:</td>
    <td>
        <input name="cost_value_total" id="cost_value_total"
value="<?php echo dPformSafe($obj->cost_value_total);?>" readonly="readonly" />
    </td>
</tr>

```

```

        </table>
        <br>
        * <?php echo $AppUI->_('Required Fields'); ?>
    </tr>
    <tr>
        <td>
            <input class="button" type="button" name="cancel" value="<?php echo
            $AppUI->_('Cancel'); ?>" onClick="javascript:if (confirm('<?php echo $AppUI-
            >_('Are you sure you want to cancel?', UI_OUTPUT_JS); ?>')) {history.back(-1);}"/>
        </td>
        <td align="right">
            <input type="button" class="button" value="<?php echo $AppUI-
            >_('Submit'); ?>" onClick="sumTotalValue();submitIt()" />
        </td>
    </tr>
</form>
</table>

```

addedit_costs_not_human.php

```

<?php
if (!defined('DP_BASE_DIR')) {
    die('You should not access this file directly.');
```

```

}

require_once DP_BASE_DIR . "/modules/costs/costs_functions.php";

$cost_id = intval(dPgetParam($_GET, 'cost_id', 0));
$project_id = intval(dPgetParam($_GET, 'project_id', 0));

// check permissions for this record
$scanEdit = getPermission($m, 'edit', $cost_id);

```



```

if (!(($ScanEdit && $cost_id) || ($ScanAuthor && !($cost_id)))) {
    $AppUI->redirect('m=public&a=access_denied');
}

$q = new DBQuery();
$q->addQuery('*');
$q->addTable('costs');
$q->addWhere('cost_id = ' . $cost_id);

// check if this record has dependancies to prevent deletion
$msg = "";
$obj = new CCosts();
$canDelete = $obj->canDelete($msg, $cost_id);

// load the record data
$obj = null;
if (!$db_loadObject($q->prepare(), $obj) && ($cost_id > 0)) {
    $AppUI->setMsg('Estimative Costs');
    $AppUI->setMsg("invalidID", UI_MSG_ERROR, true);
    $AppUI->redirect();
}

/* transform date to dd/mm/yyyy */
$date_begin = intval($obj->cost_date_begin) ? new CDate($obj->cost_date_begin) :
null;
$date_end = intval($obj->cost_date_end) ? new CDate($obj->cost_date_end) : null;
$df = $AppUI->getPref('SHDATEFORMAT');

/*Get end date project*/
$q->clear();
$q->addQuery('project_start_date,project_end_date');
$q->addTable('projects');
$q->addWhere("project_id = '$project_id'");

```

```

$datesProject = & $q->exec();
$dateTemp = substr($datesProject->fields['project_end_date'], 0, -9);
$dateEP = (string)$dateTemp;

// setup the title block
$title = $cost_id ? "Edit" : "Add";
$titleBlock = new CTitleBlock($title, 'costs.png', $m, "$m.$a");
if ($scanDelete && $cost_id > 0) {
    $titleBlock->addCrumbDelete('delete non-human resource', $scanDelete, $msg);
}
if ($cost_id != 0) {
    $titleBlock->addCrumb(
'?'m=costs&a=view_costs&project_id='.$project_id, 'return resource
estimates' );
}
$titleBlock->show();
?>
<!-- import the calendar script -->
<script type="text/javascript" src="<?php echo DP_BASE_URL;
?>/lib/calendar/calendar.js"></script>
<!-- import the language module -->
<script type="text/javascript" src="<?php echo DP_BASE_URL;
?>/lib/calendar/lang/calendar-<?php echo $AppUI->user_locale; ?>.js"></script>

<script language="javascript">
function submitIt() {

    var f = document.uploadFrm;
    //f.submit();

    var trans = "<?php echo $dateEP; ?>";
    var str1 = String(trans);

```

```

var str2 = document.getElementById("cost_date_end").value;

var yr1 = parseInt(str1.substring(0,4),10);
var mon1 = parseInt(str1.substring(5,7),10);
var dt1 = parseInt(str1.substring(8,10),10);

var yr2 = parseInt(str2.substring(0,4),10);
var mon2 = parseInt(str2.substring(4,6),10);
var dt2 = parseInt(str2.substring(6,8),10);

var date1 = new Date(yr1, mon1, dt1);
var date2 = new Date(yr2, mon2, dt2);
if(date2 > date1)
{
    alert("'Date end' cannot be greater than date end of the project");
    msg = "Date end cannot be greater than date end project";
    return false;
}

var msg = "";
var foc=false;
if (f.cost_value_unitary.value == 0 || f.cost_value_unitary.value < 0) {
    msg += "\nPlease enter a valid value to unitary value(Greater than 0)";
    if ((foc==false) && (navigator.userAgent.indexOf('MSIE')== -1)) {
        f.budget_reserve_financial_impact.focus();
        foc=true;
    }
}

if (msg.length < 1) {
    f.submit();
}

```

```

    } else {
        alert(msg);
    }

}

function delIt() {
    if (confirm("<?php echo $AppUI->_('Delete Non Human Resource?',
UI_OUTPUT_JS); ?>")) {
        var f = document.uploadFrm;
        f.del.value='1';
        f.submit();
    }
}

function sumTotalValueNH()
{
    var qtd= <?php echo dPformSafe($obj->cost_quantity) ?>;
    var uniValue = document.getElementById('cost_value_unitary').value;

    var total = qtd * uniValue;

    document.getElementById('cost_value_total').value = total;
}

function popCalendar( field ){
    calendarField = field;
    idate = eval( 'document.uploadFrm.cost_' + field + '.value' );
    window.open(
'index.php?m=public&a=calendar&dialog=1&callback=setCalendar&date=' + idate,
'calwin', 'width=280, height=250, scrollbars=no' );
}

```

```

/**
 * @param string Input date in the format YYYYMMDD
 * @param string Formatted date
 */
function setCalendar( idate, fdate ) {
    fld_date = eval( 'document.uploadFrm.cost_' + calendarField );
    fld_fdate = eval( 'document.uploadFrm.' + calendarField );
    fld_date.value = idate;
    fld_fdate.value = fdate;

    // set end date automatically with start date if start date is after end date
    if( calendarField == 'cost_date_end' ) {
        if( document.uploadFrm.date_end.value < idate ) {
            document.uploadFrm.cost_date_end.value = idate;
            document.uploadFrm.date_end.value = fdate;
        }
    }
}
</script>

<table width="100%" border="0" cellpadding="3" cellspacing="3" class="std"
name="threads" charset=UTF-8>

<form name="uploadFrm" action="?m=costs" method="post">
    <input type="hidden" name="dosql" value="do_costs_aed" />
    <input type="hidden" name="del" value="0" />
    <input type="hidden" name="cost_id" value="<?php echo $cost_id; ?>" />

<tr>
    <td width="100%" valign="top" align="center">
        <table cellspacing="1" cellpadding="2" width="60%">
            <tr>

```

```

        <td align="right" nowrap="nowrap"><?php echo $AppUI->_('Name');
?>:</td>

        <td>
            <?php
                echo dPformSafe($obj->cost_description);
            ?>
        </td>
    </tr>
    <tr>
        <td align="right" nowrap="nowrap"><?php echo $AppUI-
>_('Quantity'); ?>:</td>
        <td id="cost_quantity">
            <?php echo dPformSafe($obj->cost_quantity) ?>
        </td>
    </tr>
    <tr>
        <td align="right" nowrap="nowrap"><?php echo $AppUI->_('Date
Begin'); ?>*</td>
        <td>
            <input type="hidden" name="cost_date_begin"
id="cost_date_begin" value="<?php echo (($date_begin) ? $date_begin-
>format(FMT_TIMESTAMP_DATE) : ""); ?>"/>
            <!-- format(FMT_TIMESTAMP_DATE) -->
            <input type="text" class="text" name="date_begin" id="date0"
value="<?php echo (($date_begin) ? $date_begin->format($df) : ""); ?>"
disabled="disabled" />

            <a href="#" onclick="popCalendar( 'date_begin', 'date_begin');">
                
            </a>
        </td>
    </tr>

```

```

        <tr>
            <td align="right" nowrap="nowrap"><?php echo $AppUI->_('Date
Begin'); ?>*</td>
            <td>
                <input type="hidden" name="cost_date_end" id="cost_date_end"
value="<?php echo (($date_end) ? $date_end->format(FMT_TIMESTAMP_DATE) :
"); ?>" />
                <!-- format(FMT_TIMESTAMP_DATE) -->
                <input type="text" class="text" name="date_end" id="date1"
value="<?php echo (($date_end) ? $date_end->format($df) : ""); ?>"
disabled="disabled" />
                <a href="#" onclick="popCalendar( 'date_end', 'date_end');">
                    
                </a>
            </td>
        </tr>
        <tr>
            <td align="right" nowrap="nowrap"><?php echo $AppUI->_('Unitary
Value'); ?>*</td>
            <td>
                <input name="cost_value_unitary" id="cost_value_unitary"
onchange="sumTotalValueNH()" value="<?php echo dPformSafe($obj-
>cost_value_unitary); ?>" />
            </td>
        </tr>
        <tr>
            <td align="right" nowrap="nowrap"><?php echo $AppUI->_('Total
Value'); ?>*</td>
            <td>
                <input name="cost_value_total" id="cost_value_total"
value="<?php echo dPformSafe($obj->cost_value_total); ?>" readonly="readonly" />
            </td>
        </tr>
    </table>

```

```

        </td>
    </tr>

</table>
<br>
* <?php echo $AppUI->_('Required Fields'); ?>
</tr>
<tr>
    <td>
        <input class="button" type="button" name="cancel" value="<?php echo
$AppUI->_('Cancel'); ?>" onClick="javascript:if (confirm('<?php echo $AppUI-
>_('Are you sure you want to cancel?', UI_OUTPUT_JS); ?>')) {history.back(-1);}"/>
    </td>
    <td align="right">
        <input type="button" class="button" value="<?php echo $AppUI-
>_('Submit'); ?>" onclick="submitIt()" />
    </td>
</tr>
</form>
</table>

```

costs.class.php

```

<?php
if (!defined('DP_BASE_DIR')) {
    die('You should not access this file directly. ');
}

//require_once $AppUI->getSystemClass('dp');
require_once($AppUI->getSystemClass('dp'));
require_once($AppUI->getModuleClass('projects'));
/**

```



```

* Costs Class
*/
class CCosts extends CDpObject {
    var $cost_id = NULL;
    var $cost_type_id = NULL;
    var $cost_description = NULL;
    var $cost_quantity = NULL;
    var $cost_date_begin = NULL;
    var $cost_date_end = NULL;
    var $cost_value_unitary = 0;
    var $cost_value_total = 0;

    function CCosts() {
        $this->CDpObject('costs', 'cost_id');
    }

    function check() {
        // ensure the integrity of some variables
        $this->cost_id = intval($this->cost_id);
        return NULL; // object is ok
    }

    function delete() {
        global $dPconfig;
        $this->_message = "deleted";

        // delete the main table reference
        $q = new DBQuery();
        $q->setDelete('costs');
        $q->addWhere('cost_id = ' . $this->cost_id);
        if (!$q->exec()) {
            return db_error();
        }
    }
}

```

```

        return NULL;
    }
}

class CBudgetReserve extends CDpObject {
    var $budget_reserve_id = NULL;
    var $budget_reserve_description = NULL;
    var $budget_reserve_financial_impact = 0;
    var $budget_reserve_inicial_month = NULL;
    var $budget_reserve_final_month = NULL;
    var $budget_reserve_value_total = 0;

    function CBudgetReserve() {
        $this->CDpObject('budget_reserve', 'budget_reserve_id');
    }

    function check() {
        // ensure the integrity of some variables
        $this->budget_reserve_id = intval($this->budget_reserve_id);
        return NULL; // object is ok
    }

    function delete() {
        global $dPconfig;
        $this->_message = "deleted";

        // delete the main table reference
        $q = new DBQuery();
        $q->setDelete('budget_reserve');
        $q->addWhere('budget_reserve_id = ' . $this->budget_reserve_id);
        if (!$q->exec()) {
            return db_error();
        }
    }
}

```

```

        return NULL;
    }
}

class CBudget extends CDpObject {
    var $budget_id = NULL;
    var $budget_reserve_management = NULL;
    var $budget_sub_total = 0;
    var $budget_total = 0;

    function CBudget() {
        $this->CDpObject('budget', 'budget_id');
    }

    function check() {
        // ensure the integrity of some variables
        $this->budget_id = intval($this->budget_id);
        return NULL; // object is ok
    }

    function delete() {
        global $dPconfig;
        $this->_message = "deleted";

        // delete the main table reference
        $q = new DBQuery();
        $q->setDelete('budget');
        $q->addWhere('budget_id = ' . $this->budget_id);
        if (!$q->exec()) {
            return db_error();
        }
        return NULL;
    }
}

```

```
    }  
}
```

?>

costs_function.php

```
<?php
```

```
function getCostValueTotal($id) {  
    $query = new DBQuery;  
    $query->addTable('human_resource');  
    $query->addQuery('*');  
    $sql = $query->prepare();  
    $query->clear();  
    return db_loadList($sql);  
}  
  
function getResources($cond, $project) {  
    $q = new DBQuery;  
    if ($cond == "Human") {  
        $q->clear();  
        $q->addQuery('*');  
        $q->addTable('costs');  
        $q->addWhere("cost_type_id = '0' $project");  
        $q->addOrder('cost_description');  
        $humanCost = $q->loadList();  
        return $humanCost;  
    } else if ($cond == "Non-Human") {  
        $q->clear();  
        $q->addQuery('*');  
        $q->addTable('costs');  
        $q->addWhere("cost_type_id = '1' $project");
```

```
$q->addOrder('cost_description');  
$notHumanCost = $q->loadList();  
return $notHumanCost;  
}  
}
```

```
function diasemana($data) {  
    $ano = substr("$data", 0, 4);  
    $mes = substr("$data", 5, -3);  
    $dia = substr("$data", 8, 9);  
  
    $diasemana = date("w", mktime(0, 0, 0, $mes, $dia, $ano));  
  
    switch ($diasemana) {  
        case "0": $diasemana = "Domingo";  
            break;  
        case "1": $diasemana = "Segunda-Feira";  
            break;  
        case "2": $diasemana = "Terça-Feira";  
            break;  
        case "3": $diasemana = "Quarta-Feira";  
            break;  
        case "4": $diasemana = "Quinta-Feira";  
            break;  
        case "5": $diasemana = "Sexta-Feira";  
            break;  
        case "6": $diasemana = "Sábado";  
            break;  
    }  
  
    echo "$diasemana";  
}
```

```

function diferencaMeses($d1, $d2) {

    return diffDate($d1, $d2, 'M');
}

function insertCostValues($project) {
    // INSERT ON COSTS

    $q = new DBQuery();
    /*
    $q->clear();
    $q->addQuery('dp.user_id,dp.task_id,SUM(((dp.perc_assignment/100)
t.task_duration) as hours_user');
    $q->addTable('tasks', 't');
    $q->innerJoin('user_tasks', 'dp', 'dp.task_id = t.task_id');
    $q->innerJoin('human_resource', 'h', 'dp.user_id = h.human_resource_user_id');
    $q->innerJoin('users', 'usr', 'usr.user_id = h.human_resource_user_id');
    $q->addWhere('t.task_project = ' . $project);
    $q->addGroup('usr.user_id');
    $q->addOrder('usr.user_id ASC');
    $hoursUser = $q->loadList();

    $k = 0;
    $i = 0;
    foreach ($hoursUser as $temp) {
        $array[$i] = $temp['hours_user'];
        $i++;
    }
    *
    /* CRIAR ATUALIZAÇÃO CUSTOS HUMANOS E NÃO HUMANOS */
    /*
    $value = ($row['cost_value'] * $row['cost_quantity']) * diferencaMeses($date1,
$date2);

```

```

$q->clear();
$q->addTable('costs');
$q->addUpdate('cost_description', $row['contact_first_name'] . ' - ' .
$row['contact_last_name']. ' - ' . $row['human_resources_role_name']);
$q->addUpdate('cost_value_unitary', $row['cost_value']);
$q->addUpdate('cost_value_total', $value);
$q->addWhere('cost_id=' . $row['cost_id']. ' and cost_type_id= 0');
$q->exec();
*
*/

$q->clear();
$q->addQuery('DISTINCT usr.user_username,hs.human_resources_role_name,
usr.user_id, mc.cost_value,cts.contact_first_name,cts.contact_last_name');
$q->addTable('tasks', 't');
$q->innerJoin('user_tasks', 'dp', 'dp.task_id = t.task_id');
$q->innerJoin('human_resource', 'h', 'dp.user_id = h.human_resource_user_id');
$q->innerJoin('users', 'usr', 'usr.user_id = h.human_resource_user_id');
$q->innerJoin('contacts', 'cts', 'usr.user_contact = cts.contact_id');
$q->innerJoin('human_resource_roles', 'hr', 'hr.human_resource_id =
h.human_resource_id');
$q->innerJoin('human_resources_role', 'hs', 'hr.human_resources_role_id =
hs.human_resources_role_id');
$q->innerJoin('monitoring_user_cost', 'mc', 'usr.user_id = mc.user_id');
$q->addWhere('t.task_project = ' . $project);
$q->addOrder('usr.user_id ASC');
$res = $q->loadList();

$whereProject = ' and cost_project_id=' . $project;

$humanCost = getResources("Human", $whereProject);

```

```

$date1 = mktime(0, 0, 0, date("m"), date("d"), date("Y"));
$date2 = mktime(0, 0, 0, date("m") + 1, date("d"), date("Y"));
if ($humanCost == null) {
    foreach ($res as $row) {

        $q->addTable('costs');
        $q->addInsert('cost_type_id', 0);
        $q->addInsert('cost_project_id', $project);
        $q->addInsert('cost_description', $row['contact_first_name'] . ' ' .
        $row['contact_last_name'] . ' - ' . $row['human_resources_role_name']);
        $q->addInsert('cost_date_begin', date('Y-m-d H:i:s', $date1));
        $q->addInsert('cost_date_end', date('Y-m-d H:i:s', $date2));
        $q->addInsert('cost_quantity', 1);
        $q->addInsert('cost_value_unitary', $row['cost_value']);
        $q->addInsert('cost_value_total', $row['cost_value']);
        $q->exec();
    }
} else {
    /* ##### UPDATE VALORES DOS CUSTOS HUMANOS
##### */

    $i = 0;
    foreach ($res as $row) {
        $k = 0;
        $array[$i][$k] = $row['cost_value'];
        $k = $k + 1;
        $array[$i][$k] = $row['contact_first_name'] . ' ' . $row['contact_last_name'] . ' -
' . $row['human_resources_role_name'];
        $i++;
    }
    $j = 0;
    foreach ($humanCost as $row) {
        $l = 0;

```



```

        $value = ($array[$j][$l] * $row['cost_quantity']) *
diferencaMeses(substr($row['cost_date_begin'], 0, -9), substr($row['cost_date_end'],
0, -9));
    $q->clear();
    $q->addTable('costs');
    $q->addUpdate('cost_value_unitary', $array[$j][$l]);
    $q->addUpdate('cost_value_total', $value);
    $l = $l + 1;
    $q->addWhere('cost_description="" . $array[$j][$l] . "" and cost_type_id= 0');
    $q->exec();
    $j++;
}
foreach ($res as $row) {
    $name = $row['contact_first_name'] . ' ' . $row['contact_last_name'] . ' - ' .
$row['human_resources_role_name'];
    $bool = true;
    foreach ($humanCost as $column) {
        if ($name == $column['cost_description']) {
            $bool = false;
        }
    }
    if ($bool == true) {
        $q->clear();
        $q->addTable('costs');
        $q->addInsert('cost_type_id', 0);
        $q->addInsert('cost_project_id', $project);
        $q->addInsert('cost_description', $name);
        $q->addInsert('cost_quantity', 1);
        $q->addInsert('cost_date_begin', date('Y-m-d H:i:s'));
        $q->addInsert('cost_date_end', date('Y-m-d H:i:s', $date2));
        $q->addInsert('cost_value_unitary', $row['cost_value']);
        $q->addInsert('cost_value_total', $row['cost_value']);
        $q->exec();
    }
}

```

```
    }  
  }  
}
```

```
$notHumanCost = getResources("Non-Human", $whereProject);
```

```
$q->clear();
```

```
$q->addQuery('r.resource_name,t.task_id,COUNT(r.resource_name) as qntd');
```

```
$q->addTable('tasks', 't');
```

```
$q->innerJoin('resource_tasks', 'rt', 'rt.task_id = t.task_id');
```

```
$q->innerJoin('resources', 'r', 'r.resource_id = rt.resource_id');
```

```
$q->addWhere('t.task_project = ' . $project);
```

```
$q->addWhere('rt.percent_allocated = 100');
```

```
$q->addGroup('r.resource_name');
```

```
$q->addOrder('r.resource_name ASC');
```

```
$resNH = $q->loadList();
```

```
if ($notHumanCost == null) {
```

```
  foreach ($resNH as $row) {
```

```
    $q->addTable('costs');
```

```
    $q->addInsert('cost_type_id', 1);
```

```
    $q->addInsert('cost_project_id', $project);
```

```
    $q->addInsert('cost_description', $row['resource_name']);
```

```
    $q->addInsert('cost_quantity', $row['qntd']);
```

```
    $q->addInsert('cost_date_begin', date('Y-m-d H:i:s'));
```

```
    $q->addInsert('cost_date_end', date('Y-m-d H:i:s', $date2));
```

```
    $q->addInsert('cost_value_unitary', 0);
```

```
    $q->addInsert('cost_value_total', 0);
```

```
    $q->exec();
```

```
  }
```

```
} else {
```

```
/* ##### UPDATE OR INSERTE NON-HUMAN
RESOURCES ##### */
```

```
$i = 0;
foreach ($resNH as $row) {
    $k = 0;
    $array[$i][$k] = $row['qntd'];
    $k = $k + 1;
    $array[$i][$k] = $row['resource_name'];
    $i++;
}
```

```
$j = 0;
```

```
foreach ($notHumanCost as $row) {
    $l = 0;
    $value = $array[$j][$l] * $row['cost_value_unitary'];
    $q->clear();
    $q->addTable('costs');
    $q->addUpdate('cost_quantity', $array[$j][$l]);
    $q->addUpdate('cost_value_total', $value);
    $l = $l + 1;
    $q->addWhere('cost_description="" . $array[$j][$l] . "" and cost_type_id= 1');
    $q->exec();
    $j++;
}
```

```
foreach ($resNH as $row) {
    $bool = true;
    foreach ($notHumanCost as $column) {
        if ($row['resource_name'] == $column['cost_description']) {
            $bool = false;
        }
    }
}
```

```

if ($bool == true) {
    $q->clear();
    $q->addTable('costs');
    $q->addInsert('cost_type_id', 1);
    $q->addInsert('cost_project_id', $project);
    $q->addInsert('cost_description', $row['resource_name']);
    $q->addInsert('cost_quantity', $row['qntd']);
    $q->addInsert('cost_date_begin', date('Y-m-d H:i:s'));
    $q->addInsert('cost_date_end', date('Y-m-d H:i:s', $date2));
    $q->addInsert('cost_value_unitary', 0);
    $q->addInsert('cost_value_total', 0);
    $q->exec();
}
}
}
}

```

```

function insertReserveBudget($project) {

```

```

    $q = new DBQuery();

    $q->clear();
    $q->addQuery('r.risk_id,r.risk_name');
    $q->addTable('risks', 'r');
    $q->addWhere("risk_project = " . $project . " and risk_probability = '3' or
risk_probability = '4'");
    $q->addOrder('risk_id');
    $risk = $q->loadList();

    $q->clear();
    $q->addQuery('*');
    $q->addTable('budget_reserve', 'b');

```

```

$q->addWhere("budget_reserve_project_id = " . $project);
$q->addOrder('budget_reserve_risk_id');
$budgets = $q->loadList();

if ($budgets == null) {
    foreach ($risk as $row) {

        $q->addTable('budget_reserve');
        $q->addInsert('budget_reserve_project_id', $project);
        $q->addInsert('budget_reserve_risk_id', $row['risk_id']);
        $q->addInsert('budget_reserve_description', $row['risk_name']);
        $q->addInsert('budget_reserve_financial_impact', 0);
        $q->addInsert('budget_reserve_inicial_month', date('Y-m-d H:i:s'));
        $q->addInsert('budget_reserve_final_month', date('Y-m-d H:i:s'));
        $q->addInsert('budget_reserve_value_total', 0);
        $q->exec();
    }
} else {
    foreach ($risk as $row) {
        $q->addTable('budget_reserve');
        $q->addUpdate('budget_reserve_description', $row['risk_name']);
        $q->addWhere('budget_reserve_project_id=' . $project . ' and
budget_reserve_risk_id=' . $row[risk_id]);
        $q->exec();
    }
    foreach ($risk as $row) {
        $bool = true;
        foreach ($budgets as $column) {
            if ($row['risk_id'] == $column['budget_reserve_risk_id']) {
                $bool = false;
            }
        }
    }
}

```

```

    if ($bool == true) {
        $q->clear();
        $q->addTable('budget_reserve');
        $q->addInsert('budget_reserve_project_id', $project);
        $q->addInsert('budget_reserve_risk_id', $row['risk_id']);
        $q->addInsert('budget_reserve_description', $row['risk_name']);
        $q->addInsert('budget_reserve_financial_impact', 0);
        $q->addInsert('budget_reserve_inicial_month', date('Y-m-d H:i:s'));
        $q->addInsert('budget_reserve_final_month', date('Y-m-d H:i:s'));
        $q->addInsert('budget_reserve_value_total', 0);
        $q->exec();
    }
}
}
}

```

```

function insertBudget($project, $subTotal) {

```

```

    $q = new DBQuery();
    $q->clear();
    $q->addQuery('*');
    $q->addTable('budget');
    $q->addWhere('budget_project_id = ' . $project);
    $q->addOrder('budget_id');
    $res = $q->loadList();
    $resul = $q->exec();

    if ($res == null) {

        $q->addTable('budget');
        $q->addInsert('budget_id', $project);
        $q->addInsert('budget_project_id', $project);
        $q->addInsert('budget_reserve_management', 0);
    }
}

```

```

    $q->addInsert('budget_sub_total', $subTotal);
    $q->addInsert('budget_total', $subTotal);
    $q->exec();
} else {
    $total = (($res[0]['budget_reserve_management'] / 100) * $subTotal);
    $total += $subTotal;

    $q->addTable('budget');
    $q->addUpdate('budget_sub_total', $subTotal);
    $q->addUpdate('budget_total', $total);
    $q->addWhere('budget_id=' . $project);
    $q->exec();
}
}

```

```

function diffDate($d1, $d2, $type = "", $sep = '-') {
    $d1 = explode($sep, $d1);
    $d2 = explode($sep, $d2);
    switch ($type) {
        case 'A':
            $X = 31536000;
            break;
        case 'M':
            $X = 2592000;
            break;
        case 'D':
            $X = 86400;
            break;
        case 'H':
            $X = 3600;
            break;
        case 'MI':
            $X = 60;

```

```
        break;
    default:
        $X = 1;
    }
}
```

```
    return floor((((mktime(0, 0, 0, $d2[1], $d2[2], $d2[0])) - (mktime(0, 0, 0, $d1[1],
$d1[2], $d1[0])))) / $X);
}
```

```
function subTotalBudget($meses, $c, $mtz, $control, $sumColumns) {
    for ($i = 0; $i <= $meses; $i++) {

        echo "<td nowrap='nowrap'> <b>";

        for ($j = 0; $j <= $c; $j++) {
            $sum = $sum + $mtz[$j][$i];
        }
        $sumColumns[$control][$i] = $sum;

        echo number_format($sum, 2, ',', '.');

        echo " </b> </td>";

        $sum = 0;
    }
    return $sumColumns;
}
```

```
function subTotalBudgetRow($meses, $c, $mtz, $control) {
    for ($i = 0; $i <= $meses; $i++) {
        $sum = $sum + $mtz[$control][$i];
    }
}
```



```

    }
    return $sum;
}

function costsBudget($meses, $c, $row, $mStartProject, $mEndProject, $mtz) {

    $monthStart = substr($row['cost_date_begin'], 5, -12);
    $diffMonths = diferencaMeses(substr($row['cost_date_begin'], 0, -9),
substr($row['cost_date_end'], 0, -9));

    if ($diffMonths < 0)
        $diffMonths = 0;

    for ($i = 0; $i <= $meses; $i++) {
        $temp = $mStartProject;
        $mStartProject++;

        if ($monthStart == $temp) {
            $k = $i;
            for ($j = 0; $j <= $diffMonths; $j++) {
                echo "<td nowrap='nowrap'>";
                $mtz[$c][$k] = $row['cost_value_total'] / ($diffMonths + 1);
                echo number_format($mtz[$c][$k], 2, ',', '.');
                echo "</td>";
                $k++;
            }
            $i = $k - 1;
        } else {
            echo "<td nowrap='nowrap'>";
            $mtz[$c][$i] = 0;
            echo number_format(0, 2, ',', '.');

```

```

        echo "</td>";
    }
}

return $mtz;
}

function costsContingency($meses, $c, $row, $mStartProject, $mEndProject, $mtz) {

    $monthStart = substr($row['budget_reserve_inicial_month'], 5, -12);
    $diffMonths = diferenciaMeses(substr($row['budget_reserve_inicial_month'], 0, -9),
substr($row['budget_reserve_final_month'], 0, -9));

    if ($diffMonths < 0)
        $diffMonths = 0;

    for ($i = 0; $i <= $meses; $i++) {
        $temp = $mStartProject;
        $mStartProject++;

        if ($monthStart == $temp) {
            $k = $i;
            for ($j = 0; $j <= $diffMonths; $j++) {
                echo "<td nowrap='nowrap'>";
                $mtz[$c][$k] = $row['budget_reserve_financial_impact'];
                echo number_format($mtz[$c][$k], 2, ',', '.');
                echo "</td>";
                $k++;
            }
            $i = $k - 1;
        } else {
            echo "<td nowrap='nowrap'>";
            $mtz[$c][$i] = 0;
        }
    }
}

```

```

        echo number_format(0, 2, ',', '.');
        echo "</td>";
    }
}

return $mtz;
}

function totalBudget($meses, $sumColumns) {

    for ($i = 0; $i <= $meses; $i++) {

        for ($j = 0; $j <= 2; $j++) {
            $result += $sumColumns[$j][$i];
        }
        echo "<td nowrap='nowrap' width='10%'>";
        echo "<b>";
        echo number_format($result, 2, ',', '.');

        echo "</b>";
        echo "</td>";
        $result = 0;
    }
}

```

?>

do_budget_aed.php

```

<?php

if (!defined('DP_BASE_DIR')) {
    die('You should not access this file directly.');
```

```

}

//add costs sql
$budget_id = intval(dPgetParam($_POST, 'budget_id', 0));
$del = intval(dPgetParam($_POST, 'del', 0));

$not = dPgetParam($_POST, 'notify', '0');
if ($not!='0') {
    $not='1';
}
$obj = new CBudget();
if ($budget_id) {
    $obj->_message = 'updated';
} else {
    $obj->_message = 'added';
}

if (!$obj->bind($_POST)) {
    $AppUI->setMsg($obj->getError(), UI_MSG_ERROR);
    $AppUI->redirect();
}

// prepare (and translate) the module name ready for the suffix
$AppUI->setMsg('Budget');
// delete the item
if ($del) {
    $obj->load($budget_id);
    if (($msg = $obj->delete())) {
        $AppUI->setMsg($msg, UI_MSG_ERROR);
        $AppUI->redirect();
    } else {
        if ($not=='1') {
            $obj->notify();
        }
    }
}

```

```

    }
    $projectSelected = intval(dPgetParam($_GET, 'project_id'));
    if ($projectSelected != null) {
        $AppUI->redirect("m=costs");
    }
    $AppUI->setMsg("deleted", UI_MSG_ALERT, true);
}
}

if (($msg = $obj->store())) {
    $AppUI->setMsg($msg, UI_MSG_ERROR);
} else {
    $obj->load($obj->budget_id);
    if ($not=='1') {
        $obj->notify();
    }
    $AppUI->setMsg($budget_id ? 'UPDATED' : 'ADDED', UI_MSG_OK, true);
}

$AppUI->redirect();

```

?>

do_budget_reserve_aed.php

```

<?php

if (!defined('DP_BASE_DIR')) {
    die('You should not access this file directly.');
}

//add costs sql

```

```

$budget_reserve_id = intval(dPgetParam($_POST, 'budget_reserve_id', 0));
$del = intval(dPgetParam($_POST, 'del', 0));

$not = dPgetParam($_POST, 'notify', '0');
if ($not!='0') {
    $not='1';
}
$obj = new CBudgetReserve();
if ($budget_reserve_id) {
    $obj->_message = 'updated';
} else {
    $obj->_message = 'added';
}

if (!$obj->bind($_POST)) {
    $AppUI->setMsg($obj->getError(), UI_MSG_ERROR);
    $AppUI->redirect();
}

// prepare (and translate) the module name ready for the suffix
$AppUI->setMsg('Cost');
// delete the item
if ($del) {
    $obj->load($budget_reserve_id);
    if (($msg = $obj->delete())) {
        $AppUI->setMsg($msg, UI_MSG_ERROR);
        $AppUI->redirect();
    } else {
        if ($not=='1') {
            $obj->notify();
        }
        $projectSelected = intval(dPgetParam($_GET, 'project_id'));
        if ($projectSelected != null) {

```

```

        $AppUI->redirect("m=costs");
    }
    $AppUI->setMsg("deleted", UI_MSG_ALERT, true);
}
}

if (($msg = $obj->store())) {
    $AppUI->setMsg($msg, UI_MSG_ERROR);
} else {
    $obj->load($obj->budget_reserve_id);
    if ($not=='1') {
        $obj->notify();
    }
    $AppUI->setMsg($budget_reserve_id ? 'Updated' : 'Added', UI_MSG_OK, true);
}

$AppUI->redirect();

```

?>

do_costs_aed.php

```

<?php

if (!defined('DP_BASE_DIR')) {
    die('You should not access this file directly.');
```

```

//add costs sql
```

```

$cost_id = intval(dPgetParam($_POST, 'cost_id', 0));
```

```

$del = intval(dPgetParam($_POST, 'del', 0));
```

```

$not = dPgetParam($_POST, 'notify', '0');
```

```

if ($not!='0') {
    $not='1';
}
$obj = new CCosts();
if ($cost_id) {
    $obj->_message = 'updated';
} else {
    $obj->_message = 'added';
}

if (!$obj->bind($_POST)) {
    $AppUI->setMsg($obj->getError(), UI_MSG_ERROR);
    $AppUI->redirect();
}

// prepare (and translate) the module name ready for the suffix
$AppUI->setMsg('Cost');
// delete the item
if ($del) {
    $obj->load($cost_id);
    if (($msg = $obj->delete())) {
        $AppUI->setMsg($msg, UI_MSG_ERROR);
        $AppUI->redirect();
    } else {
        if ($not=='1') {
            $obj->notify();
        }
        $projectSelected = intval(dPgetParam($_GET, 'project_id'));
        if ($projectSelected != null) {
            $AppUI->redirect("m=costs");
        }
        $AppUI->setMsg("deleted", UI_MSG_ALERT, true);
    }
}

```



```

}

if (($msg = $obj->store())) {
    $AppUI->setMsg($msg, UI_MSG_ERROR);
} else {
    $obj->load($obj->cost_id);
    if ($not=='1') {
        $obj->notify();
    }
    $AppUI->setMsg($cost_id ? 'Updated' : 'Added', UI_MSG_OK, true);
}

$AppUI->redirect();

```

?>

projects_tab.budget.php

```

<?php
if (!defined('DP_BASE_DIR')) {
    die('You should not access this file directly.');
}
require_once DP_BASE_DIR . "/modules/costs/costs_functions.php";

$projectSelected = intval(dPgetParam($_GET, 'project_id'));

// setup the title block
$titleBlock = new CTitleBlock("Budget", './modules/costs/images/costs.png', $m,
"$m.$a");
$titleBlock->show();

```

```
$whereProject = "";
if ($projectSelected != null) {
    $whereProject = ' and cost_project_id=' . $projectSelected;
}
```

```
$q = new DBQuery();
// Get humans estimatives
$q->clear();
$q->addQuery('*');
$q->addTable('costs');
$q->addWhere("cost_type_id = '0' $whereProject");
$q->addOrder('cost_description');
$humanCost = $q->loadList();
```

```
// Get not humans estimatives
$q->clear();
$q->addQuery('*');
$q->addTable('costs');
$q->addWhere("cost_type_id = '1' $whereProject");
$q->addOrder('cost_description');
$notHumanCost = $q->loadList();
```

```
$q->clear();
$q->addQuery('*');
$q->addTable('budget');
$q->addWhere('budget_project_id = ' . $projectSelected);
$q->addOrder('budget_id');
$v = $q->loadList();
```

```
if ($v == null) {
    ?>
    <div align="right">
```

```

        <input class="button" type="button" name="new budget" value="new budget"
onclick="location.href = '?m=costs&a=view_budget&project_id=<?php
echo $projectSelected ?>';" />
    </div>
    <?php
} else {
    ?>
    <div align="right">
        <input class="button" type="button" name="edit budget" value="edit budget"
onclick="location.href = '?m=costs&a=view_budget&project_id=<?php
echo $projectSelected ?>&budget_id=<?php echo $projectSelected ?>';" />
    </div>
    <?php
}
?>

```

```

<!-- ##### ESTIMATIVAS CUSTOS HUMANOS
##### -->

```

```

<table width="100%" border="0" cellpadding="3" cellspacing="3" class="std"
style="border-radius:10px">

```

```

    <?php
    $q->clear();
    $q->addQuery('project_start_date,project_end_date');
    $q->addTable('projects');
    $q->addWhere("project_id = '$projectSelected'");
    $datesProject = & $q->exec();

    $meses = diferencaMeses(substr($datesProject->fields['project_start_date'], 0, -9),
substr($datesProject->fields['project_end_date'], 0, -9));
    $monthStartProject = substr($datesProject->fields['project_start_date'], 5, -12);

```

```
$monthStartProject = substr($datesProject->fields['project_start_date'], 5, -12);
$yearStartProject = substr($datesProject->fields['project_start_date'], 0, -15);
$yearEndProject = substr($datesProject->fields['project_end_date'], 0, -15);
```

```
$years = $yearEndProject - $yearStartProject;
$tempYear = $yearStartProject;
$tempMeses = (12 - $monthStartProject) + 1;
```

```
$sumColumns;
```

```
$c = 0;
```

```
$counter = 1;
```

```
?>
```

```
<tr>
```

```
  <th nowrap='nowrap'><?php echo $AppUI->_('Year'); ?></th>
```

```
  <?php
```

```
    for ($i = 0; $i <= $years; $i++) {
```

```
      echo '<th nowrap="nowrap" colspan="" . $tempMeses . "">';
```

```
      echo $tempYear;
```

```
      echo "</th>";
```

```
      $tempMeses = ($meses - $tempMeses) + 1;
```

```
      $ns = $tempMeses - 12;
```

```
      if ($ns > 0)
```

```
        $tempMeses = 12;
```

```
      $tempYear++;
```

```
    }
```

```
  ?>
```

```
</tr>
```

```
<tr>
```

```
  <th nowrap="nowrap" width="15%"><?php echo $AppUI->_('Item');
```

```
?></th>
```

```
  <?php
```

```
    for ($i = 0; $i <= $meses; $i++) {
```

```
      $mes = $monthStartProject;
```

```

        $monthStartProject++;
        if($mes == 12) $monthStartProject = 1;
    ?>

    <th nowrap='nowrap'>
        <?php echo $AppUI->_('Month ' . $mes); ?>
    </th>
    <?php $counter++;
    }
    ?>
    <th nowrap="nowrap" width="10%"><?php echo $AppUI->_('Total Cost');
?></a></th>
</tr>

<tr>
    <td nowrap='nowrap' align="center" colspan="<?php echo $meses + 2 ?>">
        <b><?php echo $AppUI->_('HUMAN RESOURCE ESTIMATIVE'); ?></b>
    </td>
</tr>

<?php foreach ($humanCost as $row) {
    ?>
    <tr>
        <td nowrap="nowrap"><?php echo $row['cost_description']; ?></td>
        <?php
            $mtz = costsBudget($meses, $c, $row, substr($datesProject->fields['project_start_date'], 5, -12), substr($datesProject->fields['project_end_date'], 5, -12), $mtz);
            $c++;
        ?>

        <td nowrap="nowrap"><?php echo number_format($row['cost_value_total'], 2, ',', '.'); ?></td>

```

```

</tr>
<?php
    $sumH = $sumH + $row['cost_value_total'];
}
?>
<tr>
    <td nowrap="nowrap" align="center" width="15%" cellpadding="3"> <b>Subtotal
Human Estimatives </b> </td>
    <?php
        $sumColumns = subTotalBudget($meses, $c, $mtz, 0, $sumColumns);
    ?>
    <td nowrap="nowrap" cellpadding="3"><b><?php echo number_format($sumH, 2,
',, '); ?></b></td>

</tr>

<br>
<!-- ##### ESTIMATIVAS CUSTOS NAO
HUMANOS ##### -->

<tr>
    <td nowrap='nowrap' align="center" colspan="<?php echo $meses + 2 ?>">
        <b> <?php echo $AppUI->_('NON-HUMAN RESOURCE ESTIMATIVE');
?></b>
    </td>
</tr>

<?php
$c = 0;
foreach ($notHumanCost as $row) {
    ?>
    <tr>

```

```

        <td nowrap="nowrap" width="15%"><?php echo $row['cost_description'];
?></td>
        <?php
            $mtzNH = costsBudget($meses, $c, $row, substr($datesProject-
>fields['project_start_date'], 5, -12), substr($datesProject->fields['project_end_date'],
5, -12), $mtzNH);
            $c++;
        ?>

        <td nowrap="nowrap"><?php echo number_format($row['cost_value_total'], 2,
',, '.'); ?></td>
    </tr>
    <?php
        $sumNH = $sumNH + $row['cost_value_total'];
    }
?>
<tr>
    <td nowrap="nowrap" align="center" width="15%" cellpadding="1"> <b>Subtotal
Non-Human Estimatives </b> </td>
    <?php
        $sumColumns = subTotalBudget($meses, $c, $mtzNH, 1, $sumColumns);
    ?>
    <td nowrap="nowrap" cellpadding="3"><b><?php echo number_format($sumNH,
2, ',', '.'); ?></b></td>

</tr>

<!-- ##### CONTINGENCY RESERVE
##### -->

<?php
$q->clear();

```

```

$q->addQuery('*');
$q->addTable('budget_reserve', 'b');
$q->addWhere("budget_reserve_project_id = " . $projectSelected);
$q->addOrder('budget_reserve_risk_id');
$risks = $q->loadList();
?>

<tr>
  <td nowrap='nowrap' width='100%' align="center" colspan="<?php echo $meses +
2 ?>">
    <b><?php echo $AppUI->_('CONTINGENCY RESERVE'); ?></b>
  </td>
</tr>

<?php
$k = 0;
$c = 0;

foreach ($risks as $row) {
  ?>
  <tr>
    <td>
      <?php echo $row['budget_reserve_description'] ?>
    </td>

    <?php
    $mtzC = costsContingency($meses, $c, $row, $monthSProject,
substr($datesProject->fields['project_end_date'], 5, -12), $mtzC);
    $c++;
  ?>

    <td nowrap="nowrap"><?php
    $sumRowContingency = subTotalBudgetRow($meses, $c, $mtzC, $k);

```



```

        echo number_format($sumRowContingency, 2, ',', '.');
    ?></td>
</tr>
<?php
    $k++;
    $sumC = $sumC + $sumRowContingency;
    ;
}
?>
<tr>
    <td nowrap="nowrap" align="center" width="15%" cellpadding="1"> <b>Subtotal
Contingency </b> </td>
    <?php
        $sumColumns = subTotalBudget($meses, $c, $mtzC, 2, $sumColumns);
    ?>
    <td nowrap="nowrap" cellpadding="3"><b><?php echo number_format($sumC, 2,
',, '.'); ?></b></td>

</tr>

<tr>
    <td nowrap='nowrap' align="center" colspan="<?php echo $meses + 2 ?>"></td>
</tr>

<tr>

    <td nowrap='nowrap' align="center">
        <b><?php echo $AppUI->_('TOTAL'); ?></b>
    </td>
    <?php
        totalBudget($meses, $sumColumns);
    ?>
    <td nowrap="nowrap"><b><?php

```

```

        $subTotal = $sumH + $sumNH + $sumC;
        echo number_format($subTotal, 2, ',', '.');
        ?></b></td>
</tr>

</table>

<!-- #####          CALCULO      DO      BUDGET
##### -->

<?php
$q->clear();
$q->addQuery('*');
$q->addTable('budget');
$q->addWhere('budget_project_id = ' . $projectSelected);
$q->addOrder('budget_id');
$v = $q->exec();
?>

<table width="100%" border="0" cellpadding="3" cellspacing="3" class="std"
style="border-radius:10px">

    <tr>
        <th nowrap='nowrap' width='100%' colspan="6">
<?php echo $AppUI->_('Budget'); ?>
        </th>
    </tr>
    <tr>
        <th nowrap="nowrap"><?php echo $AppUI->_('Managememt Reserve(%));
?></a></th>
        <th nowrap="nowrap"><?php echo $AppUI->_('Subtotal Budget'); ?></a></th>
        <th nowrap="nowrap"><?php echo $AppUI->_('Total Value'); ?></a></th>

```

```

</tr>
<tr>
  <td nowrap="nowrap"><?php echo $v->fields['budget_reserve_management']
?></td>
  <td nowrap="nowrap"><?php echo number_format($subTotal, 2, ',', '.'); ?>
    <input type="hidden" name="budget_sub_total" value="<?php echo
$subTotal; ?>" />
  </td>
  <td nowrap="nowrap"><?php
$budget = ($subTotal + ($subTotal * ($v->fields['budget_reserve_management'] /
100)));
echo number_format($budget, 2, ',', '.');
?>
    <input type="hidden" name="budget_total" value="<?php echo $budget; ?>"
/>
  </td>
</tr>
<tr>
  <td nowrap="nowrap" align="center"><b> Total Budget: <?php echo
number_format($budget, 2, ',', '.'); ?></b></td>

</tr>
</table>

```

projects_tab.costs.php

```

<?php
if (!defined('DP_BASE_DIR')) {
    die('You should not access this file directly.');
```

```

}
require_once DP_BASE_DIR . "/modules/costs/costs_functions.php";

$projectSelected = intval(dPgetParam($_GET, 'project_id'));

```

```

// setup the title block
$titleBlock = new CTitleBlock("Costs Project", '../modules/costs/images/costs.png',
$m, "$m.$a");
$titleBlock->show();

$whereProject = "";
if ($projectSelected != null) {
    $whereProject = ' and cost_project_id=' . $projectSelected;
}

// Get humans estimatives
$humanCost = getResources("Human",$whereProject);

// Get not humans estimatives
$notHumanCost = getResources("Non-Human",$whereProject);

$df = $AppUI->getPref('SHDATEFORMAT');

if($humanCost == null && $notHumanCost == null){
    ?>
    <div align="right">
        <input class="button" type="button" name="new cost estimative" value="new cost
estimative"
                onclick="location.href
                =
                '?m=costs&a=view_costs&project_id=?php echo $projectSelected ?>';" />
    </div>
    <?php
    }else{
        ?>
    <div align="right">

```

```



```

```

<!-- ##### ESTIMATIVAS CUSTOS HUMANOS
##### -->

```

```

<table width="100%" border="0" cellpadding="3" cellspacing="3" class="std"
style="border-radius:10px">

```

```

<tr>
<th nowrap='nowrap' width='100%' colspan="6">
<?php echo $AppUI->_('Human Resource Estimative'); ?>
</th>
</tr>

```

```

<tr>
<th nowrap="nowrap" width="20%"><?php echo $AppUI->_('Name');
?></a></th>
<th nowrap="nowrap"><?php echo $AppUI->_('Date Begin'); ?></a></th>
<th nowrap="nowrap"><?php echo $AppUI->_('Date End'); ?></a></th>
<th nowrap="nowrap" width="10%"><?php echo $AppUI->_('Hours/Month');
?></a></th>
<th nowrap="nowrap" width="15%"><?php echo $AppUI->_('Hour Cost');
?></a></th>
<th nowrap="nowrap"><?php echo $AppUI->_('Total Cost'); ?></a></th>

```

```

</tr>
<?php
foreach ($humanCost as $row) {

```

```

        /* transform date to dd/mm/yyyy */
        $date_begin      =      intval($row['cost_date_begin'])      ?      new
CDate($row['cost_date_begin']) : null;
        $date_end = intval($row['cost_date_end']) ? new CDate($row['cost_date_end']) :
null;
        ?>
        <tr>
            <td nowrap="nowrap"><?php echo $row['cost_description']; ?></td>
            <td nowrap="nowrap"><?php echo $date_begin ? $date_begin->format($df) :
"; ?></td>
            <td nowrap="nowrap"><?php echo $date_end ? $date_end->format($df) : ";
?></td>
            <td nowrap="nowrap"><?php echo $row['cost_quantity']; ?></td>
            <td nowrap="nowrap"><?php echo $row['cost_value_unitary'] ?></td>
            <td nowrap="nowrap"><?php echo number_format($row['cost_value_total'],
2, ',', '.'); ?></td>
        </tr>
        <?php
        $sumH = $sumH + $row['cost_value_total'];
        }
        ?>
        <tr>
            <td nowrap="nowrap" align="right" colspan="6" cellpadding="3"> <b>Subtotal
Human Estimativas </b> </td>
            <td nowrap="nowrap" cellpadding="3"><b><?php echo number_format($sumH,
2, ',', '.'); ?></b></td>

        </tr>
        <br>
        <!-- ##### ESTIMATIVAS CUSTOS NAO
HUMANOS ##### -->
        <tr>
            <th nowrap="nowrap" width="100%" colspan="6">

```

```

        <?php echo $AppUI->_('Non-Human Resource Estimative'); ?>
    </th>
</tr>
<tr>
    <th nowrap="nowrap" width="20%"><?php echo $AppUI->_('Description');
?></a></th>
    <th nowrap="nowrap"><?php echo $AppUI->_('Date Begin'); ?></a></th>
    <th nowrap="nowrap"><?php echo $AppUI->_('Date End'); ?></a></th>
    <th nowrap="nowrap" width="10%"><?php echo $AppUI->_('Quantity');
?></a></th>
    <th nowrap="nowrap" width="15%"><?php echo $AppUI->_('Unitary Cost');
?></a></th>
    <th nowrap="nowrap"><?php echo $AppUI->_('Total Cost'); ?></a></th>
</tr>
<?php
foreach ($notHumanCost as $row) {
    /* transform date to dd/mm/yyyy */
    $date_begin      =      intval($row['cost_date_begin'])      ?      new
CDate($row['cost_date_begin']) : null;
    $date_end = intval($row['cost_date_end']) ? new CDate($row['cost_date_end']) :
null;
    ?>
    <tr>
        <td nowrap="nowrap"><?php echo $row['cost_description']; ?></td>
        <td nowrap="nowrap"><?php echo $date_begin ? $date_begin->format($df) :
"; ?></td>
        <td nowrap="nowrap"><?php echo $date_end ? $date_end->format($df) : ";
?></td>
        <td nowrap="nowrap"><?php echo $row['cost_quantity']; ?></td>
        <td
            nowrap="nowrap"><?php
                echo
number_format($row['cost_value_unitary'], 2, ',', '.'); ?></td>
        <td nowrap="nowrap"><?php echo number_format($row['cost_value_total'],
2, ',', '.'); ?></td>

```

```

        </tr>
        <?php
            $sumNH = $sumNH + $row['cost_value_total'];
        }
    ?>
    <tr>
        <td nowrap="nowrap" align="right" colspan="6" cellpadding="3"> <b>Subtotal
        Not Human Estimatives </b> </td>
        <td
            nowrap="nowrap"
            cellpadding="3"><b><?php
                echo
                number_format($sumNH, 2, ',', '.'); ?></b></td>

    </tr>

</table>

```

view_budget.php

```

<?php
if (!defined('DP_BASE_DIR')) {
    die('You should not access this file directly.');
```

```

}
require_once DP_BASE_DIR . "/modules/costs/costs_functions.php";

$budget_id = intval(dPgetParam($_GET, 'budget_id', 0));
$projectSelected = intval(dPgetParam($_GET, 'project_id'));

$q = new DBQuery;
$q->clear();
$q->addQuery('*');
$q->addTable('budget');
$q->addWhere('budget_id = ' . $budget_id);

```



```

// check if this record has dependancies to prevent deletion
$msg = "";
$bud = new CBudget();

$bud = null;
if ((!db_loadObject($q->prepare(), $bud)) && ($budget_id > 0)) {
    $AppUI->setMsg('Budget');
    $AppUI->setMsg("invalidID", UI_MSG_ERROR, true);
    $AppUI->redirect();
}
//INSERT NA TABELA BUDGET_RESERVE

insertReserveBudget($projectSelected);

// setup the title block
$titleBlock = new CTitleBlock("Budgets", 'costs.png', $m, "$m.$a");
$titleBlock->addCrumb('?m=costs', 'projects budgets');

$titleBlock->show();

$whereProject = "";
if ($projectSelected != null) {
    $whereProject = ' and cost_project_id=' . $projectSelected;
}
// Get humans estimatives
$q->clear();
$q->addQuery('*');
$q->addTable('costs');

```

```

$q->addWhere("cost_type_id = '0' $whereProject");
$q->addOrder('cost_description');
$humanCost = $q->loadList();

// Get not humans estimatives
$q->clear();
$q->addQuery('*');
$q->addTable('costs');
$q->addWhere("cost_type_id = '1' $whereProject");
$q->addOrder('cost_description');
$notHumanCost = $q->loadList();
?>
<script language="javascript">
    function submitIt() {
        var f = document.uploadFrm;
        f.submit();
    }
    function delIt() {
        if (confirm("<?php echo $AppUI->_('Delete this registry?', UI_OUTPUT_JS);
?>")) {
            var f = document.uploadFrm;
            f.del.value='1';
            f.submit();
        }
    }
}
</script>

<!-- ##### ESTIMATIVAS CUSTOS HUMANOS
##### -->

<table width="100%" border="0" cellpadding="3" cellspacing="3" class="tbl">

<?php

```

```
$q->clear();
$q->addQuery('project_start_date,project_end_date');
$q->addTable('projects');
$q->addWhere("project_id = '$projectSelected'");
$datesProject = & $q->exec();
```

```
$meses = diferencaMeses(substr($datesProject->fields['project_start_date'], 0, -9),
substr($datesProject->fields['project_end_date'], 0, -9));
```

```
$monthStartProject = substr($datesProject->fields['project_start_date'], 5, -12);
$monthEndProject = substr($datesProject->fields['project_end_date'], 5, -12);
$monthSProject = substr($datesProject->fields['project_start_date'], 5, -12);
$yearStartProject = substr($datesProject->fields['project_start_date'], 0, -15);
$yearEndProject = substr($datesProject->fields['project_end_date'], 0, -15);
```

```
$years = $yearEndProject - $yearStartProject;
$tempYear = $yearStartProject;
$tempMeses = (12 - $monthStartProject) + 1;
```

```
$sumColumns;
```

```
$c = 0;
```

```
$counter = 1;
```

```
?>
```

```
<tr>
```

```
<th nowrap='nowrap'><?php echo $AppUI->_('Year'); ?></th>
```

```
<?php
```

```
for($i=0; $i <= $years; $i++){
```

```
    echo '<th nowrap="nowrap" colspan="'. $tempMeses ."'>';
```

```
    echo $tempYear;
```

```
    echo"</th>";
```

```
    $tempMeses = ($meses - $tempMeses)+1;
```

```
    $ns = $tempMeses - 12;
```

```

        if($sns > 0)
            $tempMeses = 12;
            $tempYear++;
        }
    ?>
</tr>
<tr>
    <th nowrap="nowrap" width="15%"><?php echo $AppUI->_('Item');
?></a></th>
    <?php
    for ($i = 0; $i <= $meses; $i++) {
        $mes = $monthStartProject;
        $monthStartProject++;
        if($mes == 12) $monthStartProject = 1;
    ?>
    <th nowrap='nowrap'>
        <?php echo $AppUI->_('Month ' . $mes); ?>
    </th>
    <?php
    $counter++;
    }
    ?>
    <th nowrap="nowrap" width="10%"><?php echo $AppUI->_('Total Cost');
?></a></th>
</tr>
<tr>
    <td nowrap='nowrap' align="center" colspan="<?php echo $meses + 2 ?>">
        <b><?php echo $AppUI->_('HUMAN RESOURCE ESTIMATIVE'); ?></b>
    </td>
</tr>

<?php foreach ($humanCost as $row) {
    ?>

```

```

<tr>
    <td nowrap="nowrap"><?php echo $row['cost_description']; ?></td>
    <?php
        $mtz    =    costsBudget($meses,    $c,    $row,    substr($datesProject-
>fields['project_start_date'], 5, -12), substr($datesProject->fields['project_end_date'],
5, -12), $mtz);
        $c++;
    ?>

    <td nowrap="nowrap"><?php echo number_format($row['cost_value_total'],
2, ',', '.'); ?></td>
</tr>
<?php
    $sumH = $sumH + $row['cost_value_total'];
}
?>
<tr>
    <td    nowrap="nowrap"    align="center"    width="15%" cellpadding="3">
<b>Subtotal Human Estimatives </b> </td>
    <?php
        $sumColumns = subTotalBudget($meses, $c, $mtz, 0, $sumColumns);
    ?>
    <td nowrap="nowrap" cellpadding="3"><b><?php echo number_format($sumH,
2, ',', '.'); ?></b></td>

</tr>

<br>
<!-- ##### ESTIMATIVAS CUSTOS NAO
HUMANOS ##### -->

```

```

<?php
$c = 0;
?>

<tr>
  <td nowrap='nowrap' align="center" colspan="<?php echo $meses + 2 ?>">
    <b> <?php echo $AppUI->_('NON-HUMAN RESOURCE ESTIMATIVE');
?></b>
  </td>
</tr>

<?php foreach ($notHumanCost as $row) {
  ?>
  <tr>
    <td nowrap="nowrap" width="15%"><?php echo $row['cost_description'];
?></td>
    <?php
    $mtzNH = costsBudget($meses, $c, $row, substr($datesProject-
>fields['project_start_date'], 5, -12), substr($datesProject->fields['project_end_date'],
5, -12), $mtzNH);
    $c++;
    ?>

    <td nowrap="nowrap"><?php echo number_format($row['cost_value_total'],
2, ',', '.'); ?></td>
  </tr>
  <?php
  $sumNH = $sumNH + $row['cost_value_total'];
}
?>
<tr>
  <td nowrap="nowrap" align="center" width="15%" cellpadding="1">
<b>Subtotal Non-Human Estimatives </b> </td>

```

```

<?php
$sumColumns = subTotalBudget($meses, $c, $mtzNH, 1, $sumColumns);
?>
<td nowrap="nowrap" cellpadding="3"><b><?php echo
number_format($sumNH, 2, ',', '.'); ?></b></td>

</tr>

```

```

<!-- ##### CONTINGENCY RESERVE
##### -->

```

```

<?php
$q->clear();
$q->addQuery('*');
$q->addTable('budget_reserve', 'b');
$q->addWhere("budget_reserve_project_id = " . $projectSelected);
$q->addOrder('budget_reserve_risk_id');
$risks = $q->loadList();
?>

```

```

<tr>
<td nowrap='nowrap' width='100%' align="center" colspan="<?php echo $meses
+ 2 ?>">
<b><?php echo $AppUI->_('CONTINGENCY RESERVE'); ?></b>
</td>
</tr>

```

```

<?php
$k = 0;
$c = 0;

foreach ($risks as $row) {

```

```

?>
<tr>
    <td
        nowrap="nowrap"
        <a
href="index.php?m=costs&a=adedit_budget_reserve&budget_reserve_id=<?php
echo $row['budget_reserve_id']; ?>&project_id=<?php echo $projectSelected ?>">
        
        </a>&nbsp;
    <?php echo $row['budget_reserve_description'] ?>
    </td>

    <?php
    $mtzC = costsContingency($meses, $c, $row, $monthSProject,
$monthEndProject, $mtzC);
    $c++;
    ?>

    <td nowrap="nowrap">
        <?php
        $sumRowContingency = subTotalBudgetRow($meses, $c, $mtzC, $k);
        echo number_format($sumRowContingency, 2, ',', '.');
        ?>
    </td>
</tr>
<?php
$k++;
$sumC = $sumC + $sumRowContingency;
;
}
?>
<tr>
    <td nowrap="nowrap" align="center" width="15%" cellpadding="1">
<b>Subtotal Contingency </b> </td>

```



```

        <?php
        $sumColumns = subTotalBudget($meses, $c, $mtzC, 2, $sumColumns);
        ?>
        <td nowrap="nowrap" cellpadding="3"><b><?php echo number_format($sumC,
2, ',', '.'); ?></b></td>

</tr>

<tr>
    <td nowrap='nowrap' align="center" colspan="<?php echo $meses + 2
?>"></td>
</tr>

<tr>

    <td nowrap='nowrap' align="center">
        <b><?php echo $AppUI->_('TOTAL'); ?></b>
    </td>
    <?php
    totalBudget($meses, $sumColumns);
    ?>
    <td nowrap="nowrap"><b>
        <?php
        $subTotal = $sumH + $sumNH + $sumC;
        echo number_format($subTotal, 2, ',', '.');
        ?>
    </b>
    </td>
</tr>

</table>

```

```
<!-- ##### CALCULO DO BUDGET
##### -->
```

```
<?php
insertBudget($projectSelected, $subTotal);
```

```
$q->clear();
$q->addQuery('*');
$q->addTable('budget');
$q->addWhere('budget_project_id = ' . $projectSelected);
$q->addOrder('budget_id');
$v = $q->exec();
?>
```

```
<table width="100%" border="0" cellpadding="3" cellspacing="3" class="tbl">
  <tr>
    <th nowrap='nowrap' width='100%' colspan="6">
      <?php echo $AppUI->_('Budget'); ?>
    </th>
  </tr>
  <tr>
    <th nowrap="nowrap"width="25"></th>
    <th nowrap="nowrap"><?php echo $AppUI->_('Managememt Reserve(%));
?></a></th>
    <th nowrap="nowrap"><?php echo $AppUI->_('Subtotal Budget'); ?></a></th>
    <th nowrap="nowrap"><?php echo $AppUI->_('Total Value'); ?></a></th>
  </tr>
  <tr>
    <td nowrap="nowrap" align="center">
      <a href="index.php?m=costs&a=addedit_budget&budget_id=<?php echo $v-
>fields['budget_id'] ?>">
        
```

```

        </a>
    </td>

    <td nowrap="nowrap"><?php echo $bud->budget_reserve_management ?></td>
    <td nowrap="nowrap"><?php echo number_format($subTotal, 2, ',', '.'); ?>
        <!-- <input type="hidden" name="budget_sub_total" value="<?php echo
    $subTotal; ?>" /> -->
    </td>
    <td nowrap="nowrap"><?php
    $budget = ($subTotal + ($subTotal * ($bud->budget_reserve_management / 100)));
    echo number_format($budget, 2, ',', '.');
    ?>
        <!-- <input type="hidden" name="budget_total" value="<?php echo $budget;
    ?>" /> -->
    </td>
</tr>
<tr>
    <td nowrap="nowrap" align="center"><b> Total Budget: <?php echo
    number_format($budget, 2, ',', '.'); ?></b></td>
</tr>
</table>
<tr>
    <td>
        <input type="button" value="<?php echo $AppUI->_('back'); ?>"
            class="button" onclick="javascript:history.back(-1);" />
    </td>
</tr>

```

view_costs.php

```

<?php
if (!defined('DP_BASE_DIR')) {

```

```

        die('You should not access this file directly.');
```

```
    }
```

```
require_once DP_BASE_DIR . "/modules/costs/costs_functions.php";
```

```
$cost_id = intval(dPgetParam($_GET, 'cost_id', 0));
```

```
$projectSelected = intval(dPgetParam($_GET, 'project_id'));
```

```
$perms = & $AppUI->acl();
```

```
$q = new DBQuery;
```

```
$q->clear();
```

```
$q->addQuery('*');
```

```
$q->addTable('costs');
```

```
$q->addWhere('cost_project_id = ' . $projectSelected);
```

```
// check if this record has dependancies to prevent deletion
```

```
$msg = "";
```

```
$obj = new CCosts();
```

```
// load the record data
```

```
$obj = null;
```

```
if ((!db_loadObject($q->prepare(), $obj)) && ($cost_id > 0)) {
```

```
    $AppUI->setMsg('Estimative Costs');
```

```
    $AppUI->setMsg("invalidID", UI_MSG_ERROR, true);
```

```
    $AppUI->redirect();
```

```
}
```

```
/* Funcao para inserir na tabela de custos */
```

```

insertCostValues($projectSelected);

// setup the title block
$titleBlock = new CTitleBlock("Estimative Costs", 'costs.png', $m, "$m.$a");
$titleBlock->addCrumb('?m=costs', 'projects estimatives');
$titleBlock->show();

$whereProject = "";
if ($projectSelected != null) {
    $whereProject = ' and cost_project_id=' . $projectSelected;
}

/* transform date to dd/mm/yyyy */
$date_begin = intval($obj->cost_date_begin) ? new CDate($obj->cost_date_begin) :
null;
$date_end = intval($obj->cost_date_end) ? new CDate($obj->cost_date_end) : null;
$df = $AppUI->getPref('SHDATEFORMAT');

// Get humans estimatives
$humanCost = getResources("Human", $whereProject);

// Get non humans estimatives
$notHumanCost = getResources("Non-Human", $whereProject);
?>

<!-- ##### ESTIMATIVAS CUSTOS HUMANOS
##### -->

<table width="100%" border="0" cellpadding="3" cellspacing="3" class="tbl">

```

```

<tr>
  <th nowrap='nowrap' width='100%' colspan="7">
    <?php echo $AppUI->_('Human Resource Estimative'); ?>
  </th>
</tr>
<tr>
  <th nowrap="nowrap" width="1%"></th>
  <th nowrap="nowrap" width="20%"><?php echo $AppUI->_('Name');
?></a></th>
  <th nowrap="nowrap"><?php echo $AppUI->_('Date Begin'); ?></a></th>
  <th nowrap="nowrap"><?php echo $AppUI->_('Date End'); ?></a></th>
  <th nowrap="nowrap" width="10%"><?php echo $AppUI->_('Hours/Month');
?></a></th>
  <th nowrap="nowrap" width="15%"><?php echo $AppUI->_('Hour Cost');
?></a></th>
  <th nowrap="nowrap"><?php echo $AppUI->_('Total Cost'); ?></a></th>
</tr>
<?php
foreach ($humanCost as $row) {
  /* transform date to dd/mm/yyyy */
  $date_begin = intval($row['cost_date_begin']) ? new
CDate($row['cost_date_begin']) : null;
  $date_end = intval($row['cost_date_end']) ? new CDate($row['cost_date_end']) :
null;
  ?>
  <tr>
    <td nowrap="nowrap" align="center">
      <a href="index.php?m=costs&a=adddedit_costs&cost_id=<?php echo
($row['cost_id']); ?>&project_id=<?php echo $projectSelected ?>">
        
      </a>
    </td>

```

```

        <td nowrap="nowrap"><?php echo $row['cost_description']; ?></td>
        <td nowrap="nowrap"><?php echo $date_begin ? $date_begin->format($df) :
"; ?></td>
        <td nowrap="nowrap"><?php echo $date_end ? $date_end->format($df) : ";
?></td>
        <td nowrap="nowrap"><?php echo $row['cost_quantity']; ?></td>
        <td
                nowrap="nowrap"><?php
                                echo
number_format($row['cost_value_unitary'], 2, ',', '.'); ?></td>
        <td nowrap="nowrap"><?php echo number_format($row['cost_value_total'],
2, ',', '.'); ?></td>
    </tr>
    <?php
    $sumH = $sumH + $row['cost_value_total'];
    }
    ?>
    <tr>
        <td nowrap="nowrap" align="right" colspan="6" cellpadding="3"> <b>Subtotal
Human Estimatives </b> </td>
        <td nowrap="nowrap" cellpadding="3"><b><?php echo number_format($sumH,
2, ',', '.'); ?></b></td>

    </tr>

</table>
<br>
<!-- ##### ESTIMATIVAS CUSTOS NAO
HUMANOS ##### -->
<table width="100%" border="0" cellpadding="3" cellspacing="3" class="tbl">

    <tr>
        <th nowrap='nowrap' width='100%' colspan="7">
<?php echo $AppUI->_('Non-Human Resource Estimative'); ?>
        </th>

```

```

</tr>
<tr>
  <th nowrap="nowrap" width="1%"></th>
  <th nowrap="nowrap" width="20%"><?php echo $AppUI->_('Description');
?></a></th>
  <th nowrap="nowrap"><?php echo $AppUI->_('Date Begin'); ?></a></th>
  <th nowrap="nowrap"><?php echo $AppUI->_('Date End'); ?></a></th>
  <th nowrap="nowrap" width="10%"><?php echo $AppUI->_('Quantity');
?></a></th>
  <th nowrap="nowrap" width="15%"><?php echo $AppUI->_('Unitary Cost');
?></a></th>
  <th nowrap="nowrap"><?php echo $AppUI->_('Total Cost'); ?></a></th>
</tr>
<?php foreach ($notHumanCost as $row) {
  /* transform date to dd/mm/yyyy */
  $date_begin = intval($row['cost_date_begin']) ? new
CDate($row['cost_date_begin']) : null;
  $date_end = intval($row['cost_date_end']) ? new CDate($row['cost_date_end']) :
null;
  ?>
  <tr>
    <td nowrap="nowrap" align="center">
      <a href="index.php?m=costs&a=adddit_costs_not_human&cost_id=<?php
echo($row['cost_id']); ?>&project_id=<?php echo $projectSelected ?>">
        
      </a>
    </td>
    <td nowrap="nowrap"><?php echo $row['cost_description']; ?></td>
    <td nowrap="nowrap"><?php echo $date_begin ? $date_begin->format($df) :
"; ?></td>
    <td nowrap="nowrap"><?php echo $date_end ? $date_end->format($df) : ";
?></td>

```



```

        <td nowrap="nowrap"><?php echo $row['cost_quantity']; ?></td>
        <td
                nowrap="nowrap"><?php
                                echo
number_format($row['cost_value_unitary'], 2, ',', '.'); ?></td>
        <td nowrap="nowrap"><?php echo number_format($row['cost_value_total'],
2, ',', '.'); ?></td>
    </tr>
    <?php
    $sumNH = $sumNH + $row['cost_value_total'];
    }
    ?>
    <tr>
        <td nowrap="nowrap" align="right" colspan="6" cellpadding="3"> <b>Subtotal
Not Human Estimatives </b> </td>
        <td
                nowrap="nowrap"
                cellpadding="3"><b><?php
                                echo
number_format($sumNH, 2, ',', '.'); ?></b></td>

    </tr>
</table>
<tr>
    <td>
        <input type="button" value="<?php echo $AppUI->_('back'); ?>"
                class="button" onclick="javascript:history.back(-1);" />
    </td>
</tr>
</tr>

```

vw_budget.php

```

<?php
if (!defined('DP_BASE_DIR')) {
    die('You should not access this file directly.');
```

```

$q = new DBQuery();
$q->addTable('budget', 'b');
$q->addQuery('DISTINCT p.project_id, p.project_name');
$q->addJoin('projects', 'p', 'p.project_id = b.budget_project_id');
$q->addOrder('p.project_id');
$res = $q->loadList();

```

```
?>
```

```
<table width='100%' border='0' cellpadding='2' cellspacing='1' class='tbl' >
```

```
<tr>
```

```
<th nowrap='nowrap' width='10%'>
```

```
<?php echo $AppUI->_('Project ID'); ?>
```

```
</th>
```

```
<th nowrap='nowrap' width='85%'>
```

```
<?php echo $AppUI->_('Project Name'); ?>
```

```
</th>
```

```
</tr>
```

```
<?php foreach ($res as $row) {
```

```
?>
```

```
<tr>
```

```
<td nowrap='nowrap'>
```

```
<?php echo $row['project_id'] ?>
```

```
</td>
```

```
<td nowrap='nowrap'>
```

```
<a href="index.php?m=costs&a=view_budget&project_id=<?php
echo $row['project_id']; ?>&budget_id=<?php echo $row['project_id']; ?>">
```

```
<?php echo $row['project_name']; ?>
```

```
</a>
```

```
</td>
```

```
</tr>
```

```
<?php } ?>
```

```
<?php
$q->clear();
?>
</table>
```

vw_costs.php

```
<?php
if (!defined('DP_BASE_DIR')) {
    die('You should not access this file directly.');
}

$q = new DBQuery();

$q->addTable('costs', 'c');
$q->addQuery('DISTINCT p.project_id, p.project_name');
$q->addJoin('projects', 'p', 'p.project_id = c.cost_project_id');
$q->addOrder('p.project_id');
$res = $q->loadList();
?>
<table width='100%' border='0' cellpadding='2' cellspacing='1' class='tbl' >

    <tr>
        <th nowrap='nowrap' width='10%'>
            <?php echo $AppUI->_('Project ID'); ?>
        </th>
        <th nowrap='nowrap' width='85%'>
            <?php echo $AppUI->_('Project Name'); ?>
        </th>
    </tr>

    <?php foreach ($res as $row) {
```

```
?>
<tr>
  <td nowrap='nowrap'>
    <?php echo $row['project_id'] ?>
  </td>
  <td nowrap='nowrap'>
    <a href="index.php?m=costs&a=view_costs&project_id=<?php
echo $row['project_id']; ?>">
      <?php echo $row['project_name']; ?>
    </a>
  </td>
</tr>
<?php } ?>
<?php
$q->clear();
?>
</table>
```

Evolução da Ferramenta de Gerenciamento de Projetos dotProject para Planejamento de Custos

Rafael Reiter de Albuquerque

Departamento de Estatística e Informática – Universidade Federal de Santa Catarina (UFSC)
Campus Universitário Reitor João David Ferreira Lima – 88040-970 – Florianópolis – SC –
Brasil

{reiter}@inf.ufsc.br

***Abstract.** One of the contemporary needs of software companies today is certainly the project management. To perform such management is very useful to be used a tool (software) to assist and improve performance. But it is still difficult to find a free tool that provides support based on best practice project management referenced by PMBOK and CMMI. Therefore, this article aims to show how the development was carried out for the dotProject tool for planning costs in the context of micro and small enterprises. This paper shows how the survey was conducted prior to the development of the tool and the evaluations.*

***Resumo.** Uma das necessidades contemporâneas das empresas de software hoje em dia é certamente o gerenciamento de projetos. Para realizar este gerenciamento é muito útil que seja utilizado uma ferramenta (software) para auxiliar e melhorar o desempenho. Porém ainda hoje é difícil encontrar uma ferramenta livre que forneça suporte baseado nas melhores práticas de gerenciamento de projetos referenciadas pelo PMBOK e CMMI. Por isto este artigo visa mostrar como foi realizada a evolução da ferramenta dotProject para o planejamento de custos no contexto de micro e pequenas empresas. Este artigo apresenta como foi realizado o levantamento do estado da arte, a evolução da ferramenta e as avaliações realizadas.*

1. Introdução

Atualmente mudanças no ambiente empresarial e tecnológico têm feito cada vez mais empresas adotarem o gerenciamento de projetos no seu cotidiano. A necessidade de rapidez, competitividade, clientes exigentes e lucros cada vez mais difíceis de obter, caracterizaram um cenário ideal para que o gerenciamento e execução de projetos viessem a se encaixar dentro do orçamento e tempo das empresas. No Brasil no ano de 2010 o setor de TI mostrou um crescimento de 21,3% sendo que os setores de software e serviços cresceram 24%, constando que cerca de 6,74 bilhões de dólares foram movimentados na área de software, o que representou perto de 2,2% do mercado mundial [ABES, 2011].

Este artigo tem como objetivo geral mostrar como foi a evolução da ferramenta de gerenciamento de projetos dotProject na área de planejamento de custos de projetos alinhado ao PMBOK 4ed..

2. Estado da arte

Para a seleção das ferramentas, foi realizada uma pesquisa no repositório web SourceForge por ser considerado o maior site de aplicações de código aberto [PEREIRA; GONÇALVES; WANGENHEIM, 2011]. A partir daí foram aplicados vários critérios, de inclusão e exclusão, para então filtrar alguns softwares dos diversos encontrados.

Para avaliar o suporte ao planejamento de custos em cada uma destas ferramentas foram definidos critérios de avaliação com base no [PMI 2008] e [SEI 2010]. A definição desses critérios é baseado numa definição feita anteriormente no artigo Wangenheim et al. (2010), revisando e atualizando os critérios em relação das versões atuais dos modelos.

Tabela 1. Critérios de avaliação ferramentas

Item	Critério
P1.Estimate Costs	Estimar o esforço do projeto e custo para produtos de trabalho e tarefas com base no raciocínio de estimação.
P2.Determine Budget	Estabelecer e manter o orçamento do projeto e seu cronograma.

Para indicar o grau de atendimento de cada uma das ferramentas ao critério de avaliação é utilizada uma escala ordinal de 4 pontos conforme definido na tabela 2 [Pereira, Gonçalves e Wangenheim 2011].

Tabela 2. Tabela de escala de pontuação [Pereira, Gonçalves e Wangenheim 2011]

Código	Descrição
-	Não provê nenhum suporte
*	Oferece suporte básico, mas as funcionalidades não foram projetadas para este fim
**	Oferece suporte básico, mas as funcionalidades foram projetadas para este fim
***	Oferece suporte completo

Pode-se notar que a maioria das ferramentas apresenta pouco ou nenhum suporte ao item P2 (determinar o orçamento). Agora em relação ao item P1 pode-se notar que todas as ferramentas suportam ao menos o básico, que seria a estimativa das atividades do projeto, mesmo elas sendo de formas limitadas. O resultado da avaliação é mostrado na tabela 3.

Tabela 3. Tabela de avaliação das ferramentas de gerenciamento de projeto

	dotProject	Project.net	PhpCollab	Track+	Streber
P1. Estimate Costs	*	*	*	**	*
P2. Determine Budget	*	*	-	-	*

3. Evolução da ferramenta dotProject

3.1. dotProject

O dotProject é um sistema de gerência de projetos em software livre de fácil utilização, com um conjunto de funcionalidades e características que o tornam indicado para implementação em ambientes corporativos, pois atende a diversas necessidades de gerentes e escritórios de projetos (PMOs) [DOTPROJECT]. O sistema provê mecanismos de planejamento e controle para atividades, cronograma, comunicação, e aquisições.

O dotProject é uma aplicação web e seu acesso é feito através de um navegador, assim sua utilização independe de sistema operacional e instalação na máquina do usuário, pois é executado em um servidor. Em termos técnicos, o dotProject é um sistema escrito em PHP, que utiliza banco de dados MySQL [DOTPROJECT]. O sistema atualmente desenvolvido nos fornece funcionalidades para que o gerenciamento do projeto seja facilitado e eficiente. O módulo core do dotProject nos fornece as funcionalidades básicas para o gerenciamento de projetos

3.2. Processo genérico para o planejamento de custos em MPEs

Com base nas características da MPEs e literatura apresentada no capítulo 2, definimos as entradas e saídas para este processo. Definimos como entrada: Plano de Recursos, Registro de riscos, Cronograma do projeto, Plano de Recursos Humanos e Dados Organizacionais de Pessoal. Como saídas temos as estimativas de custos dos recursos humanos e não humanos, a *baseline* de custos e o orçamento do projeto.

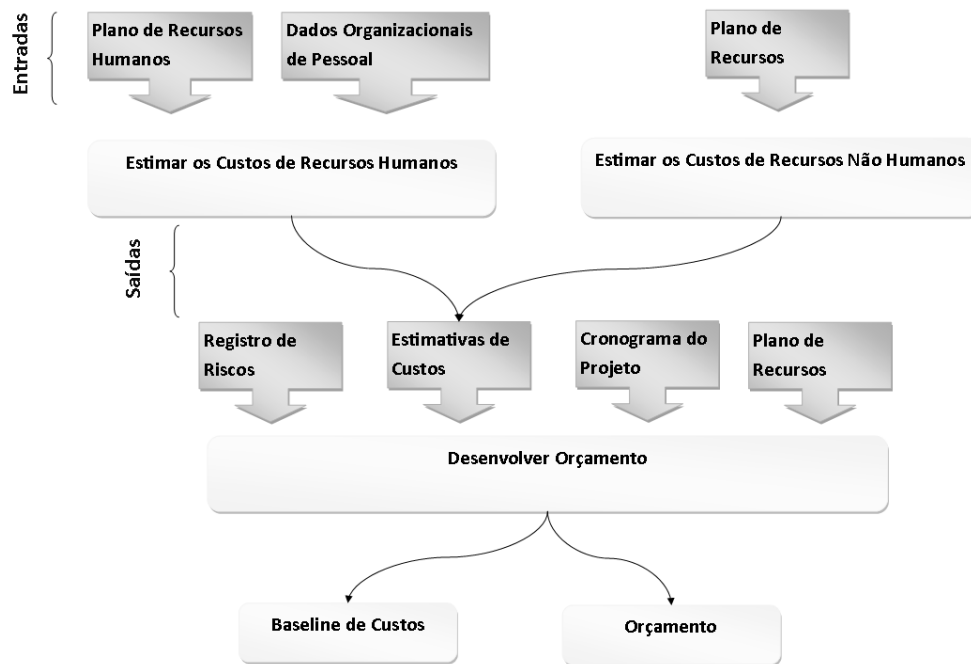


Figura 1. Processo genérico para o planejamento de custos

3.3. Modelagem da arquitetura do sistema

A modelagem da solução foi construída pensando em dois componentes ou duas partes para o módulo, aonde uma é responsável pelos custos de recursos e o outro componente pela criação da *baseline* de custos demonstrando o orçamento. Esses componentes por sua vez dependem de outros módulos, aonde alguns já são do core do dotProject e outros que devem ser integrados ao sistema. A figura 17 mostra a ligação e dependência do módulo desenvolvido para com os outros.

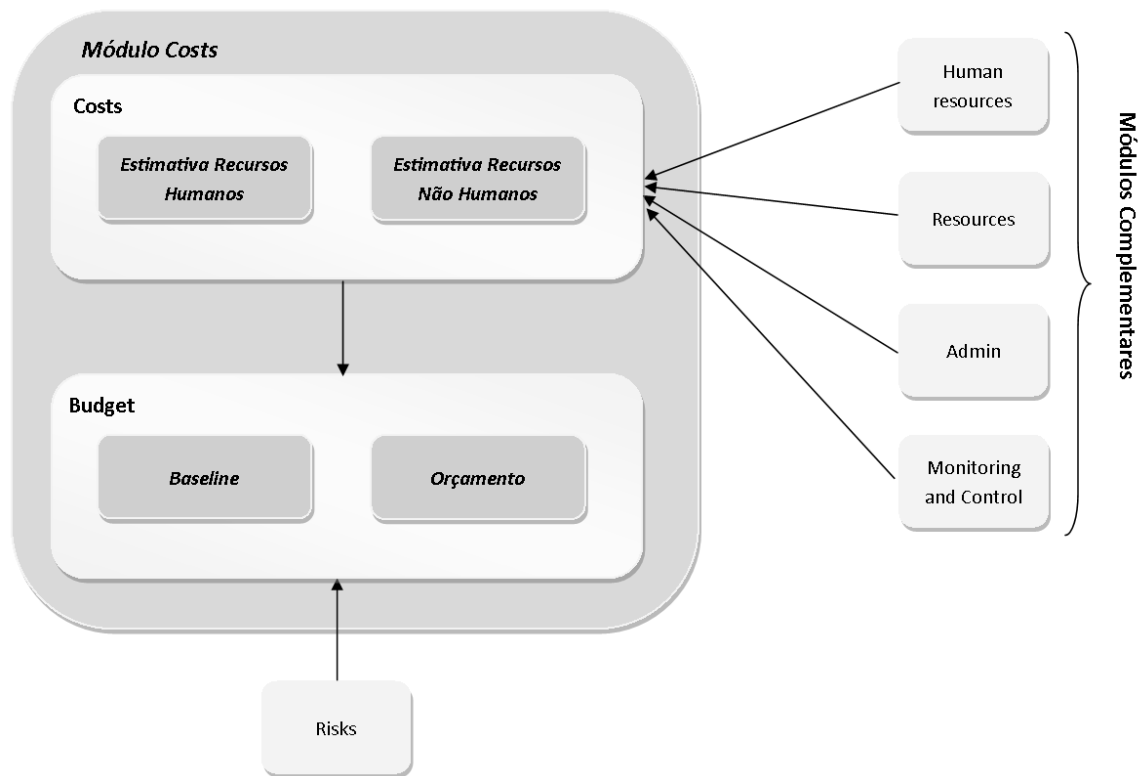


Figura 2. Modelagem da solução

3.4. Implementação do sistema

A implementação do módulo de custos foi desenvolvida utilizando a linguagem de programação PHP e a sua interface utilizando a linguagem de marcação HTML. Para o armazenamento das informações foi utilizado o banco de dados MYSQL e o servidor web Apache. Todos esses parâmetros utilizados nos outros módulos e recomendado pelo framework do sistema.

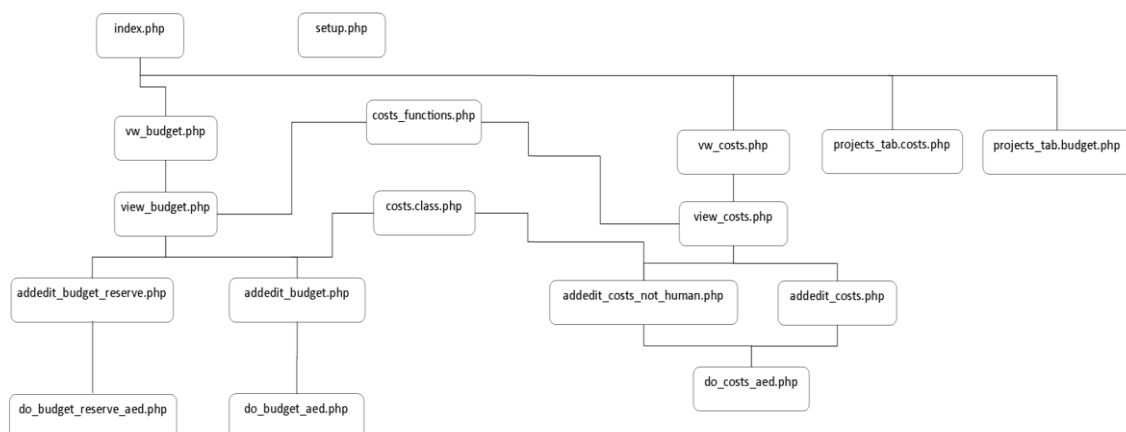


Figure 3. Diagrama de dependências de arquivos do módulo de custos

A figura 4 mostra como as estimativas de custos humanos e não humanos são demonstrados no módulo. Estes valores podem ser editados clicando-se ao ícone ao lado de cada linha, como mostrado na figura 5.

dotProject 2.1.5 dotProject.net
FREE SOFTWARE

Companies | Projects | Tasks | Calendar | Files | Contacts | Forums | Tickets | User Admin | System Admin | Human Resources | Risks | Resources | Closure - New Item -

Welcome Admin Person Help | My Info | **Todo** | Today | Logout

\$ Estimative Costs

projects estimatives

Human Resource Estimative						
Name	Date Begin	Date End	Hours/Month	Hour Cost	Total Cost	
Pedro da Silva - Programador	01/09/2012	10/11/2012	80	25,00	4.000,00	
Roberto Carungo - Analista	01/12/2012	10/02/2013	100	10,00	2.000,00	
Subtotal Human Estimatives					6.000,00	

Non-Human Resource Estimative						
Description	Quantity	Date Begin	Date End	Unitary Cost	Total Cost	
Notebook	2	15/08/2012	15/11/2012	1.000,00	2.000,00	
Subtotal Not Human Estimatives					2.000,00	

Figure 4. Estimativas de custos humanos e não humanos

Os recursos não humanos podem ser editados da mesma forma que os humanos. O módulo calcula valores totais assim que os valores parciais exigidos são informados, facilitando desta forma a visualização dos valores para o usuário.

dotProject 2.1.5 dotProject.net
FREE SOFTWARE

Companies | Projects | Tasks | Calendar | Files | Contacts | Forums | Tickets | User Admin | System Admin | Human Resources | Risks | Resources | Closure - New Item -

Welcome Admin Person Help | My Info | **Todo** | Today | Logout

\$ Edit

return resource estimatives delete human resource

Name: Pedro da Silva - Programador

Date Begin*:

Date End*:

Hours per Month*:

Unitary Value:

Total Value:

* Required Fields

Figure 5. Editando valores de recursos humanos

Como mostrado na figura 6 a *baseline* de custos é montada automaticamente pelo sistema, mostrando subtotais e totais. O orçamento pode ser calculado rapidamente informando um valor de porcentagem sobre o subtotal da *baseline* de custos.

Year		2012						2013		Total Cost(\$)
Item	Month 07	Month 8	Month 9	Month 10	Month 11	Month 12	Month 1	Month 2		
HUMAN RESOURCE ESTIMATIVE										
Pedro da Silva - Programador	0,00	0,00	1.333,33	1.333,33	1.333,33	0,00	0,00	0,00	4.000,00	
Roberto Carungo - Analista	0,00	0,00	0,00	0,00	0,00	666,67	666,67	666,67	2.000,00	
Subtotal Human Estimatives	0,00	0,00	1.333,33	1.333,33	1.333,33	666,67	666,67	666,67	6.000,00	
NON-HUMAN RESOURCE ESTIMATIVE										
Notebook	0,00	500,00	500,00	500,00	500,00	0,00	0,00	0,00	2.000,00	
Subtotal Non-Human Estimatives	0,00	500,00	500,00	500,00	500,00	0,00	0,00	0,00	2.000,00	
CONTINGENCY RESERVE										
Quebra HD	0,00	250,00	250,00	250,00	250,00	250,00	0,00	0,00	1.250,00	
Roubo celular	0,00	0,00	0,00	0,00	1.500,00	1.500,00	1.500,00	0,00	4.500,00	
Subtotal Contingency	0,00	250,00	250,00	250,00	1.750,00	1.750,00	1.500,00	0,00	5.750,00	
TOTAL	0,00	750,00	2.083,33	2.083,33	3.583,33	2.416,67	2.166,67	666,67	13.750,00	
Budget										
		Management Reserve(%)				Subtotal Budget		Total Value		
		5.00				13.750,00		14.437,50		
Total Budget: 14.437,50										

Figure 6. Visão da *baseline* de custos e orçamento

4. Avaliação

Foram realizados dois tipos de avaliação, o primeiro por um painel de especialista e o outro em relação ao alinhamento ao PMBOK.

A primeira avaliação foram levantados 3 objetivos utilizando o método GQM – Goal/Question/Metric [BASILI; CALDIERA; ROMBACH, 1994] - um método de medição de software para identificar sistematicamente os objetivos da pesquisa e derivar suas medidas. Os objetivos foram:

Objetivo 1 Analisar a utilidade do módulo para realizar a estimativa de recursos humanos e não humanos.

Objetivo 2 Analisar a utilidade do módulo na criação de um orçamento para o projeto.

Objetivo 3 Identificar os pontos fortes e fracos da solução proposta.

Depois de aplicada a avaliação os resultados obtidos foram que a parte do módulo responsável pela montagem dos recursos humanos e não humanos mostrou que para os avaliadores estes itens estão sendo úteis para a análise e identificação dos recursos do projeto. Já para o objetivo 2 foi possível observar com a avaliação e comentários dos avaliadores que as informações apresentadas poderiam ser mais flexíveis a ponto de serem editadas manualmente e não produzidas automaticamente pelo sistema.

Na segunda avaliação, ou seja, a avaliação para verificar o alinhamento do módulo ao PMBOK foi utilizada as mesmas técnicas e parâmetros definidos no estado da arte, podendo assim comparar a evolução que a ferramenta obteve.

Os resultados obtidos são mostrados na tabela 4 e como podemos observar o módulo não atende completamente a todos os itens. Para o item P1, observa-se que o módulo supre quase todo o requisitado, pois os custos do projeto são descritos e custeados de forma a poder

observar os custos subtotais e totais por item. O que deixa de ser apresentado pelo módulo são os custos separados por atividades, que poderiam ajudar na precisão da estimativa de custo total do projeto.

Tabela 4. Tabela de avaliação das ferramentas com o módulo de custos

	P1	P2
dotProject com o módulo de custos	**	***
dotProject	*	*
projectNet	*	*
phpCollab	*	-
Track+	**	-
Streber	*	*

No item P2 o grande avanço foi a criação de uma *baseline* de custos e a visualização do valor de orçamento calculado automaticamente. Antes o sistema não obtinha nenhum cálculo ou assistente de custo do orçamento, apenas um valor de estimativa de orçamento, sem a possibilidade de observar o que estava sendo estimado realmente.

6. Conclusão

Espera-se que este trabalho possa contribuir, principalmente para micro e pequenas empresas, na realização dos planos de custos dos projetos. Permitir ao usuário construir um planejamento de custos, que antes não era possível, que lhe estime todos os custos do projeto, tanto pessoal como de material e equipamentos. Conseguir com que o gerente de projetos da empresa identifique claramente os custos planejados do projeto alinhado ao cronograma do projeto. Ainda permitir que no decorrer do projeto seja possível alterar ou observar o andamento o orçamento, a fim de ajudar ao gerente de projetos identificar possíveis gargalos.

Como trabalhos futuros são possíveis identificar alguns itens baseados nas avaliações realizadas. Para a evolução do módulo é possível modificar a *baseline* de custos, tornando alguns itens listados na mesma editáveis a fim de suprir algumas particularidades de cada projeto. O módulo também pode ser evoluído no quesito integração com outros módulos, para que as informações se tornem mais centralizadas e completas, facilitando o cadastramento e usabilidade do sistema como um todo.

Referências

Associação Brasileira das Empresas de Software (ABES). Mercado Brasileiro de Software - Panorama e Tendências. São Paulo, 2011. Disponível em: <http://www.abes.org.br/UserFiles/Image/PDFs/Mercado_BR2011.pdf>.

BASILI, V. R.; CALDIERA, G.; ROMBACH, H. D. The experience factory. 1994.

DOTPROJECT. dotProject. Wiki. Acessado em out.2012. Disponível em: <<http://docs.dotproject.net/index-.php?title=Main Page>>.

Gonçalves, R. Q. Planejamento de tempo em projetos de desenvolvendo de software para micro e pequenas empresas alinhado ao pmbok e cmmi. 2012.

Project Management Institute (PMI), Inc. Um Guia do Conhecimento em Gerenciamento de Projetos. 4. ed. Pennsylvania, 2008.

Software Engineering Institute (SEI). CMMI R for Development, Version 1.3. [S.l.], novembro 2010.

Wangenheim, C. G. V. et al. R. best practice fusion of cmmi-dev v1.2 (pp, pmc, sam) and pmbok 2008. Information and Software Technology, v. 52, n. 7, p. 749–757, julho 2010.