

Aurora

AfterLife

Plano de projeto

Felipe Duarte de Santana
Guilherme José Carvalho Cavalcanti
Igor Leonardo Rodrigues Gomes
Jonathas Mendes Alves
Mailson Daniel Lira Menezes
Mateus Gondim Romão Batista
13/09/2010

Histórico das revisões

Data	Versão	Descrição	Autor
11/09/10	1.0	Descrição inicial do plano de projeto	Felipe Duarte Jonathas Alves Igor Gomes Guilherme Cavalcanti Mailson Lira Mateus Gondim

Conteúdo

1. Objetivos.....	5
2. Escopo.....	6
3. Referências	7
4. Fases do plano	8
5. Cronograma	10
6. Riscos	12
6.1. Falta de expertise na definição do cronograma	12
6.2. Mudanças de requisitos	12
6.3. Problemas pessoais	12
6.4. Problemas com a infra-estrutura	13
6.5. Distribuição de tarefas desbalanceada	13
7. Releases	15
8. Plano de recursos	16
8.1. Alocação de recursos humanos	16
8.2. Alocação de recursos de software	17
8.3. Alocação de Recursos de Hardware.....	17
8.4. Alocação de infra-estrutura	17
8.5. Treinamento pessoal.....	17
9. Custo	18

1. Objetivos

Este documento apresenta uma visão geral do plano para o desenvolvimento de um sistema de gerenciamento de cemitérios, auxiliando no planejamento, gerenciamento e execução do projeto que será descrito. O objetivo principal é esquematizar detalhadamente os passos do processo de criação do software.

Neste documento, serão descritas cada uma das fases do desenvolvimento do projeto e seus principais marcos, bem como o cronograma em que estas fases estarão englobadas. Além disso, informações essenciais como a alocação de recursos e o orçamento estimado também são especificados.

Desta forma, o conteúdo deste documento também orienta os interessados / envolvidos no projeto (Stakeholders), provendo uma padronização que facilita o entendimento comum destes e fornece apoio a todo o gerenciamento do projeto.

2. Escopo

O plano de projeto descreverá toda a construção de um sistema para o gerenciamento de cemitérios. O programa conterà os cadastros dos funcionários e defuntos, o registro de todos os sepultamentos, com os responsáveis pelos defuntos, agendamento de velórios, rodízio de covas.

Atualmente, estas atividades são feitas manualmente. Isto acarreta perda de tempo para criação dos registros e principalmente para a busca dos mesmos, uma vez que é preciso buscar em cadernos e livros de registros. Como o tempo, o número deste cresce bastante, dificultando ainda mais a manutenção da base de dados. Além disso, o risco de perda de dados é muito grande, no caso de acidentes como incêndios e enchentes. O nosso sistema objetiva a digitalização de todo este processo, através da criação de um software específico para as necessidades de um cemitério. Os dados agora serão guardados em um banco de dados, o que deixa o tempo de acesso a eles praticamente instantâneo. Sendo digitais, o espaço físico para armazenamento de dados é bastante reduzido e o risco de perdas por acidentes também, com o uso de backups.

Além disso, o sistema também automatiza a administração do rodízio de covas. Dois anos após o sepultamento, a cova está apta a receber outro corpo. Esta manutenção era feita manualmente e o software irá criar esta lista de covas “aptas” automaticamente.

3. Referências

1. Cronograma.xls - especificação das datas relevantes no processo de desenvolvimento e a relação com as outras atividades dos membros do projeto.
Disponível em: <http://www.cin.ufpe.br/~if682>
2. PlanoDeProjeto.docx - este documento traça as diretrizes e serve como guia para a manufatura do Plano de Projetos.
Disponível em: [http://www.cin.ufpe.br/~if682/templates e exemplos/Plano%20de%20Projeto.doc](http://www.cin.ufpe.br/~if682/templates_e_exemplos/Plano%20de%20Projeto.doc)
3. Sommerville, Ian. Software Engineering, Addison-Wesley, 6ª edição.

4. Fases do plano

O processo da disciplina de Engenharia de Software e Sistemas deve seguir as seguintes fases: Concepção, Requisitos [Elicitação, Classificação e Detalhamento dos Casos de Uso], Análise [dos casos de Uso, início da arquitetura], Projeto [definição da arquitetura, modelo dependente de tecnologia, utilização de design patterns e outros], Codificação [parte de construção do código] e Testes.

1. Concepção (define o escopo do projeto)

Nesta fase, serão definidos os rumos do projeto por meio da análise das necessidades dos usuários e conceitos da aplicação. Serão considerados e elaborados o cronograma, os custos do projeto, os recursos necessários, a divisão de tarefas entre os projetistas e os possíveis riscos do projeto.

2. Requisitos (Elicitação, Classificação e Detalhamento dos Casos de Uso)

Na fase de requisitos, há contatos com o cliente e os demais envolvidos a fim de se colher os requisitos do sistema, ou seja, os serviços que o sistema proposto deverá oferecer ao cliente final. Com estes requisitos em mão, é possível uma definição mais detalhada de estimativas de tempo e custo, bem como a alocação dos projetistas às tarefas. Esta fase pode ser modificada no decorrer do desenvolvimento.

3. Análise (dos casos de Uso, início da arquitetura)

Nesta fase, serão discutidos os possíveis erros e as discordâncias entre a visão do cliente e o que foi elaborado. Um bom plano de análise é fundamental para melhorar o desempenho do programa e reduzir ao máximo o prazo de entrega do mesmo.

4. Projeto (definição da arquitetura, modelo dependente de tecnologia, utilização de design patterns e outros)

Nesta etapa, será analisada e definida a arquitetura que potencializará a conclusão do programa, tanto em questão de tempo, como em questão de recursos utilizados durante a sua fase de codificação. Também serão definidas as interfaces entre os módulos e o modelo do banco de dados.

5. Codificação (parte de construção do código)

Os requisitos serão finalmente implementados, levando em consideração o que foi discutido com o cliente, o que lhe foi mostrado e o que se desejou modificar. Nesta etapa, a função dos desenvolvedores é essencial para que não surjam novos erros e que os erros já existentes sejam corretamente analisados.

6. Testes

O principal objetivo da fase de teste é inicialmente verificar a consistência do que foi desenvolvido com o que foi estabelecido a se desenvolver. Consecutivamente iremos localizar e expor os pontos fracos do software,

validar as suposições feitas nas especificações de design e requisito através de demonstrações concretas e validar as funções do software conforme projetadas.

5. Cronograma

Segue abaixo o cronograma de atividades:

		Sistema gerenciador de cemitérios	85 dias	Qui 19/08/10	Qua 15/12/10	
		Concepção	18 dias	Qui 19/08/10	Seg 13/09/10	
		Entrevista com o cliente	1 dia	Qui 19/08/10	Qui 19/08/10	
		Definição do nome	1 dia	Ter 31/08/10	Ter 31/08/10	
		Definição dos cargos	1 dia	Sex 03/09/10	Sex 03/09/10	
		Elaboração do plano de projeto	1 sem	Seg 06/09/10	Dom 12/09/10	4;5
		Entrega do plano de projeto	1 dia	Seg 13/09/10	Seg 13/09/10	6
		Elaboração	32 dias	Ter 14/09/10	Qua 27/10/10	
		Requisitos	15 dias	Ter 14/09/10	Seg 04/10/10	
		Elicitação de requisitos	1 dia	Ter 14/09/10	Ter 14/09/10	
		Análise de requisitos	2 dias	Qua 15/09/10	Qui 16/09/10	10
		Prototipagem	6 dias	Sex 17/09/10	Sex 24/09/10	11
		Validação de requisitos	1 dia	Seg 27/09/10	Seg 27/09/10	12
		Elaboração do documento de requisitos	4 dias	Ter 28/09/10	Sex 01/10/10	13
		Entrega do documento de requisitos	1 dia	Seg 04/10/10	Seg 04/10/10	14
		Análise/Arquitetura	18 dias	Seg 04/10/10	Qua 27/10/10	
		Definição da arquitetura	2 dias	Seg 04/10/10	Ter 05/10/10	
		Análise de projeto	3 dias	Qua 06/10/10	Sex 08/10/10	17
		Elaboração do banco de dados	4 dias	Seg 11/10/10	Qui 14/10/10	18
		Elaboração do plano e projeto de testes	5 dias	Seg 11/10/10	Sex 15/10/10	
		Entrega do plano e projeto de testes	1 dia	Seg 18/10/10	Seg 18/10/10	20
		Elaboração do documento de análise e projeto	6 dias	Ter 19/10/10	Ter 26/10/10	
		Entrega do documento de análise e projeto	1 dia	Qua 27/10/10	Qua 27/10/10	22
		Implementação	15 dias	Sex 15/10/10	Qui 04/11/10	
		Implementar negócios	3 sems	Sex 15/10/10	Qui 04/11/10	
		Implementar GUI	1 sem	Sex 15/10/10	Qui 21/10/10	
		Implementar banco de dados	2 sems	Sex 15/10/10	Qui 28/10/10	19
		Integração de camadas	5 dias	Sex 29/10/10	Qui 04/11/10	26;27
		Testes	5 dias	Sex 05/11/10	Qui 11/11/10	24
		Entrega do projeto	1 dia	Seg 08/11/10	Seg 08/11/10	
		Preparação da apresentação	5 dias	Ter 09/11/10	Seg 15/11/10	30
		Apresentação do projeto	1 dia	Qua 17/11/10	Qua 17/11/10	
		Transição	5 dias?	Seg 22/11/10	Sex 26/11/10	
		Implantação	1 dia	Seg 22/11/10	Seg 22/11/10	
		Treinamento dos usuários	4 dias	Ter 23/11/10	Sex 26/11/10	

Figura 1 Cronograma de atividades

Para ilustrar de forma mais visível o fluxo de atividades de nossa equipe, recorreremos ao gráfico de Gantt, ideal para a visualização de tarefas. Segue abaixo o gráfico correspondente ao cronograma mostrado acima.

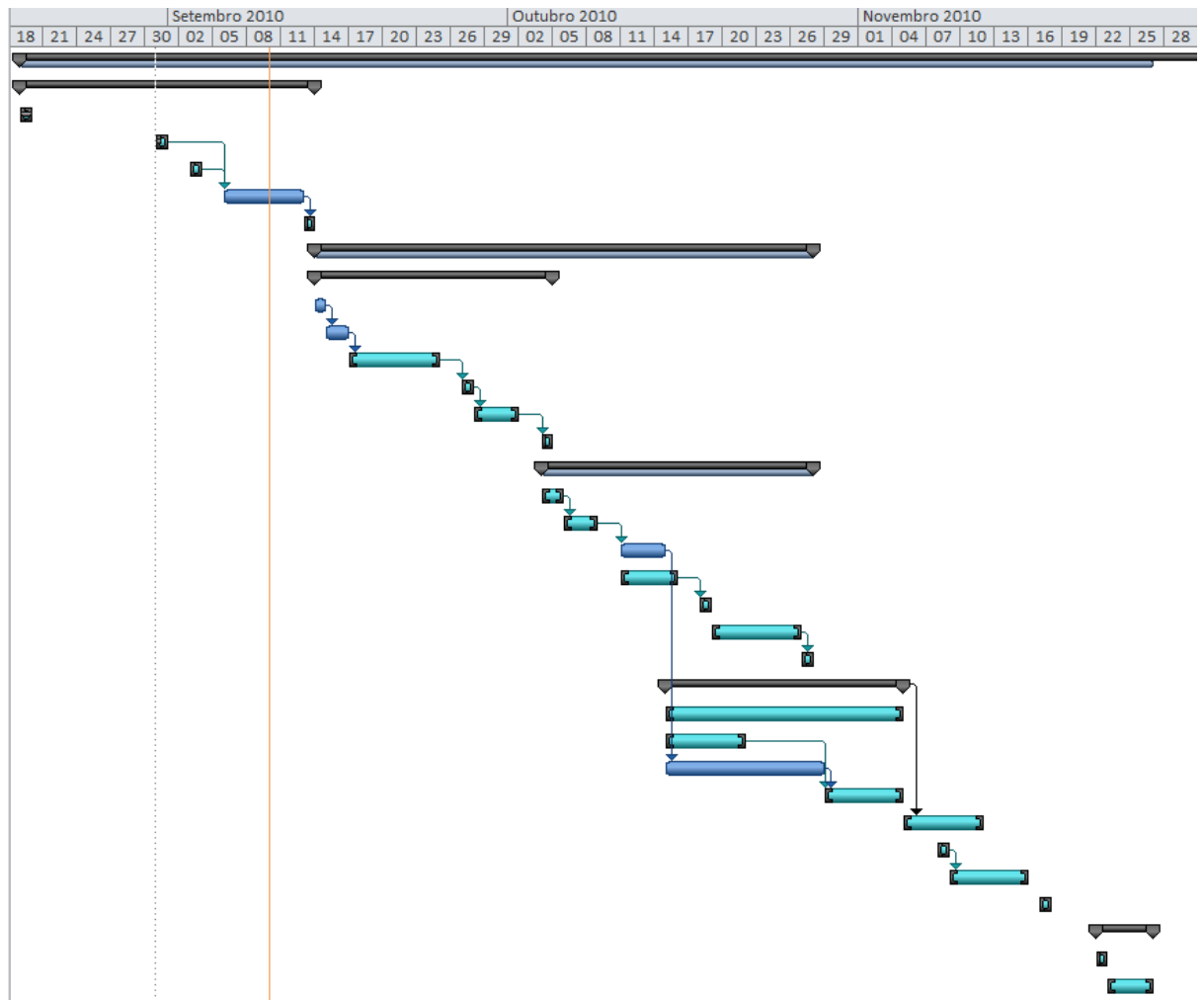


Figura 2 Gráfico de Gantt

6. Riscos

6.1. Falta de expertise na definição do cronograma

Magnitude: Alta

Descrição do risco:

A falta de experiência da equipe pode acarretar cronogramas prematuros, com prazos que não refletem a realidade do tempo necessário para realização das tarefas.

Impactos:

Atraso do projeto, aumento dos custos e insatisfação do cliente.

Indicadores:

Adiamentos frequentes nos *milestones* do cronograma.

Estratégia de mitigação e/ou plano de contingência:

- Mitigação: Estruturar o cronograma em uma análise que reflita de forma mais realista possível a realidade, levando em conta possíveis contratempos que comprometam a realização de tarefas no prazo estabelecido.
- Contingência: Usar todo o tempo disponível para recuperar os atrasos.
- Contingência: Replanejar o cronograma.

6.2. Mudanças de requisitos

Magnitude: Moderada

Descrição do risco:

Acontece quando o cliente não aprova os protótipos e/ou exige novos requisitos.

Impactos:

A mudança de requisitos tem como efeito atrasos e aumento de custos.

Indicadores:

Falta de entusiasmo do cliente com os protótipos apresentados.

Estratégia de mitigação e/ou plano de contingência:

- Mitigação: Constantes reuniões com o cliente para análise dos requisitos de forma a evitar mal-entendidos e ambiguidades.
- Contingência: Intensificar e aprofundar as reuniões com o cliente.

6.3. Problemas pessoais

Magnitude: Moderada

Descrição do risco:

Possíveis discussões e desentendimentos entre os integrantes da equipe, além de problemas individuais relacionados a fatores externos ao desenvolvimento do projeto.

Impactos:

Atraso no projeto, além de desmotivação por parte de alguns integrantes, podendo resultar até mesmo em abandono da equipe.

Indicadores:

Discussões ríspidas entre membros da equipe e faltas constantes de integrantes.

Estratégia de mitigação e/ou plano de contingência:

- Mitigação: Os integrantes devem se tratar com respeito, e procurar sempre dialogar sobre pontos de discordância com calma e coerência, a fim de não causar brigas internas.
- Contingência: Com o problema solucionado, realocar a tarefa afetada para horários extras.

6.4. Problemas com a infra-estrutura

Magnitude: Moderada

Descrição do risco:

Laboratórios lotados no Centro de Informática (CIn), onde parte do projeto será desenvolvido, além de possíveis quedas de energias, problemas com a rede local ou danos com os computadores pessoais dos integrantes da equipe.

Impactos:

Quedas de energia e danos nos equipamentos podem levar a perda de dados relevantes. Além disso, os prazos serão comprometidos com o atraso da realização de tarefas.

Indicadores:

Instabilidade na rede de local e de energia, ou no próprio computador no qual se está trabalhando.

Estratégia de mitigação e/ou plano de contingência:

- Mitigação: Preservar as máquinas bem conservadas e atualizadas, garantindo o seu funcionamento eficaz.
- Contingência: Analisar a possibilidade de migrar o ambiente de trabalho da equipe para um local mais estável e/ou com equipamentos mais confiáveis.

6.5. Distribuição de tarefas desbalanceada

Magnitude: Moderada

Descrição do risco:

Tarefas podem ser delegadas a integrantes da equipe sem que haja uma análise de seus respectivos graus de complexidade. Desta forma, integrantes podem estar responsáveis por uma mesma quantidade de tarefas, porém uns com tarefas mais complexas, e outros com tarefas mais simples.

Impactos:

Sobrecarga de alguns integrantes com tarefas pesadas.

Indicadores:

Descompasso entre a realização de tarefas de um ou mais integrantes em relação ao restante da equipe. Além disso, aborrecimentos constantes de certos membros com suas atividades podem ser mais outro indicador desse problema.

Estratégia de mitigação e/ou plano de contingência:

- Mitigação: Em reuniões para definição de cargos e delegação de tarefas, avaliar nível de complexidade de cada tarefa antes de atribuí-las a um membro que já possui tarefas de complexidade moderada/alta.
- Contingência: Redistribuir tarefas, e avaliar a possibilidade de modularizar aquelas que se mostrarem pesadas para apenas um membro.

7. Releases

Versão	Descrição	Data
1ª versão	Implementação funcional	1º de outubro de 2010
2ª versão	Implementação funcional	15 de outubro de 2010
3ª versão	Implementação final	8 de novembro de 2010

Tabela 1 Descrição dos releases

8. Plano de recursos

O desenvolvimento do AfterLife se dará no CIn, deste modo não será necessário investimento em infra-estrutura e licenças de software. Eventualmente serão usados os computadores pessoais dos integrantes para uso de recursos não presentes no CIn.

Os recursos financeiros serão então usados para: pagamento e capacitação dos funcionários, alimentação, transporte e diversos.

8.1. Alocação de recursos humanos

A equipe do projeto se reunirá ao menos uma vez por semana para atualização de cronograma, esclarecimento de dúvidas, definição de novas tarefas e possíveis mudanças nos requisitos.

A equipe do projeto é composta de 6 integrantes:

- Arquiteto de Software e Desenvolvedor (Mateus Gondim). Atividades: Prototipação da interface com usuário; Modelagem e definição dos diagramas e arquitetura do sistema; Implementação da integração dos componentes; Realização de testes.
- Gerente de Projetos e Desenvolvedor (Mailson Lira). Atividades: Planejamento, acompanhamento e gerenciamento do projeto; Definição dos requisitos do projeto; Elaboração da análise e projeto; Implementação da integração dos componentes; Documentação do projeto; Acompanhamento dos Riscos e do Plano de Projeto; Deliberar sobre mudanças de estratégias dentro do desenvolvimento.
- Analista de Sistemas e Desenvolvedor (Igor Gomes). Atividades: Definição, Modelagem e Implementação do Banco de Dados; Documentação do projeto; Implementação da integração dos componentes; Realização de testes.
- Analista de Sistemas e Desenvolvedor (Felipe Duarte). Atividades: Modelagem e Implementação do Banco de Dados; Documentação do projeto; Implementação da integração dos componentes; Realização de testes.
- Analista de Sistemas e Desenvolvedor (Guilherme Cavalcanti). Atividades: Prototipação e modelagem da interface com usuário; Implementação da integração dos componentes; Documentação do projeto;
- Analista de Sistemas e Desenvolvedor (Jonathas Alves). Atividades: Filtragem dos requisitos; Análise e Modelagem dos possíveis agentes inteligentes; Documentação do projeto.

8.2. Alocação de recursos de software

Durante o desenvolvimento do projeto *AfterLife* serão usados os seguintes softwares:

Desenvolvimento:

- Eclipse Helios;
- Java JDK 1.6 ou superior;
- Windows 7;
- Arch Linux;
- MySQL
- Oracle 10g;
- Adobe Fireworks.

Gerenciamento:

- Microsoft Office 2010;
- Microsoft Project 2010.

Controle de versão:

- Subversion (SVN).

8.3. Alocação de Recursos de Hardware

Será necessária a alocação de 6 máquinas com a seguinte configuração mínima:

- Processador Dual Core
- 1 GB de RAM
- HD de 40 GB

8.4. Alocação de infra-estrutura

A infra-estrutura usada será a do CIn. Serão usados os laboratórios da graduação e salas de reuniões.

8.5. Treinamento pessoal

O projeto requer o treinamento básico da equipe para o seu correto e satisfatório desenvolvimento. Conhecimento necessário:

- MySQL
- SVN
- Oracle 10g Administrator

9. Custo

A tabela abaixo possui os indicadores para cálculo dos salários mensais dos profissionais, de acordo com o cargo ocupado no processo de desenvolvimento.

Cargo	Carga horária semanal	Custo por hora de trabalho (R\$)	Gasto semanal com alimentação (R\$)	Gasto semanal com transporte (R\$)	Salário mensal (R\$)
Desenvolvedor	6	10,00	10,00	11,20*	324,80
Gerente	8	15,00	10,00	20,00**	600,00

Tabela 2 Salários fixos por cargos

* quatro passes tipo B

** combustível

Produto	Custo individual	Quantidade	Total mensal
Impressões	R\$ 0,10	40	R\$ 4,00
Encadernação	R\$ 2,00	4	R\$ 8,00

Tabela 3 Custos diversos

Cargo	Salário
1 Gerente de projetos	R\$ 600,00
1 Arquiteto de software	R\$ 324,80
3 Analistas de sistema	R\$ 974,40
Custo mensal	R\$ 1899,20

Tabela 4 Listagem dos salários por cargo

O custo total do projeto é estimado em R\$ 15.000,00. Sendo R\$ 7596,80 referentes aos gastos com o quadro de pessoal durante os quatro meses de desenvolvimento, e aproximados 20,50% de lucro para a empresa, num total de R\$ 3080,00; e 20% (R\$ 3000,00) destinado ao órgão de incubação da empresa.