**UTFPR-UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ**

*Bacharelado em Engenharia de Software - 6º Período*

**DISCIPLINA:** *Gerência de configuração*

**PROFESSOR:** *Alexandre L’Erario*

Documento de

Projeto de Software

**Nome do Projeto**

**Nomes dos Alunos**

**Cornélio Procópio**

**2020**

* **Introdução**
* **Contexto**

Descrever o cenário atual do negócio a ser impactado pela aplicação. Apresentar o tema do projeto, de forma clara, apresentando ao leitor a área a ser abordada, produtos ou estudos semelhantes.

* **Justificativa**

Descrever a abordagem do projeto, de modo a comunicar seu propósito e importância a todas as pessoas envolvidas. Deve ficar claro por que os clientes e usuários finais precisam da solução. Deve-se utilizar o tempo presente para falar do problema atual e tempo futuro para falar da situação do negócio quando a nova solução for implantada.

Recomenda-se utilizar as seguintes perguntas para este capítulo:

- Qual é o problema?

- Quem é afetado por este problema?

- Qual o impacto deste problema no ambiente estudado?

* **Proposta**

Descrever a solução que será implantada com o desenvolvimento do sistema. Apresentar o impacto do sistema, e como ele soluciona o problema observado.

Apresentar uma descrição em linhas gerais da solução a ser desenvolvida. Independente do que será implementado, este item visa o entendimento global do projeto.

* **Organização do Documento**

Descrever como este documento está organizado.

* **Descrição Geral do Sistema**
* **Objetivos (Gerais e Específicos)**

Apresentar de forma clara o foco do projeto, com uma descrição em linhas gerais da solução a ser desenvolvida. Deve ser descrita a delimitação da solução, que define o ponto central do projeto. Dentro de uma idéia geral do projeto, ressaltar a idéia específica efetivamente a ser desenvolvida, definindo o objetivo geral.

Para cumprir o objetivo geral é preciso delimitar metas mais específicas dentro do trabalho. São elas que, somadas, conduzirão ao desfecho do objetivo geral. Os objetivos específicos são as ações ou passos que colaboram para alcançar o objetivo geral, e também são delimitadores do escopo do trabalho, ou seja, são ações de interesse que levam ao objetivo geral, restringindo o escopo do trabalho a ser desenvolvido. Enfim, os objetivos específicos devem ser cumpridos para se chegar ao objetivo geral.

* **Limites e Restrições**

Limitar o escopo da solução a ser desenvolvida, descrevendo as necessidades que, a princípio, podem ser consideradas da alçada da aplicação mas não serão implementadas. Apresentar restrições tecnológicas ou de projeto, como por exemplo para qual ambiente será desenvolvida a solução ou um orçamento/prazo máximo previsto. Descreva aqui todas as restrições que o software apresenta com relação a desenvolvimento, implantação, uso, ou qualquer outra situação detectada. As restrições podem ser de compatibilidade, de segurança, de ambiente, de manutenibilidade, de operacionalidade, etc.

* **Descrição dos Usuários do Sistema**

Apresentar os atores que serão envolvidos na solução, bem como o papel de cada ator. Deve ser descrito para qual tipo de empresa se destina o sistema e os tipos de usuários que o utilizarão.

* **Desenvolvimento do Projeto**
* **Tecnologias e ferramentas**

Apresentar as tecnologias, ferramentas e técnicas que serão utilizadas para desenvolvimento e implantação do sistema (linguagem de programação, sistema gerenciador de banco de dados, ferramentas, etc.). Organize em tópicos (Banco de Dados, Modelagem, Gerenciamento de Projeto, etc.) e apresente as ferramentas que serão utilizadas. Não é preciso descrever detalhadamente a tecnologia/ferramenta, mas deve ficar claro o que vai ser usado no desenvolvimento do projeto.

* **Metodologia de desenvolvimento**

Apresentar o modelo de ciclo de vida ou processo a ser utilizado e o motivo da escolha. **Descrever como o modelo vai ser aplicado na realização do projeto** (quantidade de protótipos, ou fases, definição de módulos e artefatos, etc.) conforme o modelo escolhido.

* **Cronograma previsto**

Definir o cronograma de desenvolvimento do projeto. Elaborar o cronograma por semana, definindo o responsável por cada tarefa. **O cronograma deve contemplar todas as tarefas previstas no processo de desenvolvimento de software** (descrito no item 3.2 Metodologia de desenvolvimento), conforme definido para o desenvolvimento do sistema.

* **Requisitos do Sistema**
* **Requisitos Funcionais**

Apresentar os requisitos funcionais, que especificam ações que o sistema deve ser capaz de executar, ou seja, as funções do sistema. Classifique as funcionalidades quanto a prioridade:

Essencial – deve ser implementado para que o sistema funcione.

Importante – sem este requisito o sistema pode funcionar, mas não da maneira esperada.

Desejável – este tipo de requisito não compromete o funcionamento do sistema.

|  |  |  |
| --- | --- | --- |
| **ID** | **Funcionalidade** | **Prioridade** |
|  |  |  |
|  |  |  |

Criar aqui subitens do capítulo para descrever textualmente, com mais detalhes, as funcionalidades previstas.

* **Requisitos Não-funcionais**

Descrever os requisitos não-funcionais do sistema, que especificam restrições sobre os serviços ou funções providas pelo sistema, categorizando de acordo com a característica envolvida, como: Usabilidade, Padronização, Ambiente, Compatibilidade, Recursos, etc.

|  |  |  |
| --- | --- | --- |
| **ID** | **Requisito** | **Categoria** |
|  |  |  |
|  |  |  |

* **Diagramas de Casos de Uso**

Inclua aqui os diagramas de Casos de Uso desenvolvidos para o sistema, usando os IDs dos itens anteriores como referência quando necessário.

* **Análise do Sistema**

Este item deve apresentar a documentação da análise do sistema conforme o processo ou ciclo de vida descrito no capítulo 3. Organize o capítulo para apresentar os artefatos previstos e o que mais for necessário (protótipos, implementação, versões, telas, etc.), **incluindo no mínimo**:

**- Modelo do Banco de Dados**

Modelo Conceitual: Apresentar o Diagrama Entidade-Relacionamento desenvolvido para o banco de dados do sistema.

Modelo Lógico: Apresentar o esquema relacional (gráfico ou textual) do banco de dados normalizado e apresentando as tabelas com os atributos e restrições (chaves).

Dicionário de dados: Apresentar o dicionário de dados do banco de dados. Documentar cada tabela com seus atributos mostrando nome do atributo, tipo, tamanho, descrição, se é obrigatório ou não, e o que mais for necessário para descrever os dados. Documentar também usuários, stored procedures, funções e qualquer outra implementação ligada ao banco de dados.

**- Diagrama de Classes**

Apresentar o diagrama de classes previsto conforme a fase do projeto.

**- Diagrama de Atividades**

Apresentar o diagrama de atividades, que representa o detalhamento de tarefas e o fluxo de uma atividade para outra de um sistema. Nem todas as tarefas do sistema necessitam de um detalhamento, portanto deve-se considerar no que o diagrama irá auxiliar na implementação do sistema para decidir quais atividades devem ser descritas.

* **Implementação**
* **Protótipos de Telas**

Apresentar o protótipo do sistema, que consiste na interface preliminar contendo um conjunto de funcionalidades e telas. O protótipo é um recurso que deve ser adotado como estratégia para levantamento, detalhamento, validação de requisitos e modelagem de interface com o usuário (usabilidade).

As telas do sistema podem ser criadas na própria linguagem de desenvolvimento ou em qualquer outra ferramenta de desenho. Cada tela deve possuir uma descrição do seu funcionamento, constando pelo menos o objetivo da tela e dinâmica de navegação (de onde é chamada e que outras telas pode chamar). A descrição das telas deve registrar informações que possam ser consultadas para facilitar a implementação e a execução de testes, assim como a que requisitos funcionais se referem.

Se os protótipos de tela foram usados nas fases anteriores, esse item deve ser preenchido conforme os protótipos forem desenvolvidos e concluído com a versão final dos protótipos.

* **Descrição do código**

Descrever o sistema quanto ao código gerado. Explicar a organização dos arquivos, pacotes, classes ou quaisquer estruturas utilizadas no desenvolvimento do projeto, listando os componentes criados e sua estrutura. Use diagramas (Diagrama de Componentes, Diagrama de Pacotes) para ilustrar a implementação.

Descrever também convenções e padronizações para comentários no código, nomenclatura de classes, objetos, funções, etc. Se necessário, use exemplos.

* **Considerações Finais**

Apresentar e discutir os resultados obtidos e sua aplicabilidade. Abordar o que foi atingido e o que não foi, as limitações, possíveis integrações com outros projetos e continuação do sistema em trabalhos futuros.

* **Bibliografia**

Apresentar todas as obras (livros, artigos, Internet, revistas, etc...) utilizadas na elaboração da documentação e na implementação do projeto.