

Documentação de um Produto de Software

Nome do Projeto

Nome dos Alunos:

ÍNDICE DETALHADO

1. INTRODUÇÃO	3
1.1. TEMA	3
1.2. OBJETIVOS A SEREM ALCANÇADOS	3
1.3. ESCOPO PRINCIPAL	3
2. DEFINIÇÃO DO MODELO DE PROCESSO	4
3. REQUISITOS DO SISTEMA DE SOFTWARE	4
3.1. REQUISITOS FUNCIONAIS	4
3.2. REQUISITOS NÃO-FUNCIONAIS	4
4. PROJETO	5
4.1. ARQUITETURA LÓGICA	5
4.2. ARQUITETURA FÍSICA	5
5. PROTÓTIPO DE INTERFACE	6
6. CRITÉRIOS DE QUALIDADE DE SOFTWARE	7
7. TESTES	8
7.1. PLANO DE TESTES	8
7.2. ROTEIRO DE TESTES	8
ANEXO I	9

1. Introdução

1.1. Tema

Neste item deve-se descrever o tema do projeto.

Referência: UC Modelos, métodos e técnicas da engenharia de software

Referência: UC Gestão e Qualidade de software

1.2. Objetivos a serem alcançados

Neste item deve-se descrever os objetivos a serem alcançados, quem é o público alvo, plataforma de desenvolvimento, melhorias do processo atual, automatização, inovação, ...

Referência: UC Modelos, métodos e técnicas da engenharia de software

Referência: UC Gestão e Qualidade de software

1.3. Escopo principal

Neste item deve-se descrever proposta principal do projeto. Principais ações e implementações.

Referência: UC Modelos, métodos e técnicas da engenharia de software

Referência: UC Gestão e Qualidade de software

2. Definição do Modelo de Processo

Neste item deve-se descrever o modelo de processo a ser utilizado para realização do projeto, como por exemplo: cascata, incremental, SCRUM, entre outros.

Referência: UC Modelos, métodos e técnicas da engenharia de software

3. Requisitos do Sistema de Software

3.1. Requisitos Funcionais

Neste item devem ser descritos os requisitos a serem atendidos funcionalmente pelo sistema de uma forma simples, possibilitando a compreensão do comportamento do sistema pela perspectiva do usuário. Os requisitos funcionais podem ser representados por meio de modelo de caso de uso ou user stories (cartão de histórias).

Referência: UC Modelos, métodos e técnicas da engenharia de software

3.2. Requisitos Não-Funcionais

Neste item devem ser apresentados os requisitos não funcionais, que especificam restrições sobre os serviços ou funções providas pelo sistema.

Referência: UC Modelos, métodos e técnicas da engenharia de software

4. Projeto

4.1. Arquitetura Lógica

Neste item deve ser apresentada a arquitetura lógica de implementação, descrever arquitetura em camadas, padrão de projeto, linguagem de programação, banco de dados, componentes externos,

Referência: UC Modelos, métodos e técnicas da engenharia de software

4.2. Arquitetura Física

Neste item deve ser apresentada a arquitetura de infraestrutura do sistema, demonstrando o tipo de arquitetura física, a configuração de hardware, de rede... Para a representação da arquitetura de infraestrutura pode-se utilizar o diagrama de implantação.

Referência: UC Modelos, métodos e técnicas da engenharia de software

5. Protótipo de Interface

Neste item deve ser apresentado o protótipo do projeto. 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).

Referência: UC Modelos, métodos e técnicas da engenharia de software

6. Critérios de Qualidade de Software

Neste item devem ser listados e descritos os critérios de garantia da qualidade do processo que serão considerados no Projeto.

Referência: UC Gestão e Qualidade de software

7. Testes

7.1. Plano de Testes

Neste item deve ser criado o plano de testes do sistema, permitindo a validação do sistema por parte do desenvolvedor, através da verificação dos requisitos do sistema desenvolvido.

Referência: UC Gestão e Qualidade de software

7.2. Roteiro de Testes

Neste item devem ser registrados os testes realizados no sistema tendo como base o Plano de Testes do Sistema. O roteiro de testes deve ser elaborado com base nos casos de uso ou user stories (cartão de histórias) de forma manual ou automatizada.

Referência: UC Gestão e Qualidade de software

Anexo I

Neste item deve ser anexado o roteiro de entrevista ou questionário respondido, caso tenha sido aplicado.

Referência: UC Modelos, métodos e técnicas da engenharia de software