

Levantamento de Requisitos do Software

Referências:

IEEE Std. 830 – 1993. IEEE Recommended for Software Requirements Specifications

Filho, W.P.P. Engenharia de Software: Fundamentos, Métodos e Padrões. LTC: Rio de Janeiro, 2001.

INTERGRANTES DO GRUPO

Arthur Souza Lima (RA: 12723112788)
Aurea Reis (RA: 1272313156)
Alfredo Ruas Neto (RA: 1272326487)
Daniel Santos Lopes (RA: 12723130024)
Enrique Silva dos Reis (RA: 12724149018)
Elvis Oliveira dos Reis (RA: 1272318921)
Edioelson Júnior A. B. Teixeira (RA: 1272318423)
Gabriel Silva Magalhães (RA: 1272313274)
Gabriel Moreira de Oliveira (RA: 1272311981)
Hanspeter Dietiker (RA: 1272313332)
Natan Oliveira da Silva (RA: 12723211400)

1 Introdução

1.1 Objetivo

O objetivo deste documento é fornecer uma descrição detalhada dos requisitos do sistema de gerenciamento de Ferramentas de Software, incluindo suas funcionalidades, dentre elas listagem de Ferramentas de Software, na qual haverá filtragem a partir dos Tipos de Ferramentas. Este documento destina-se ao grupo de desenvolvedores do trabalho da A3, das disciplinas Gestão e qualidade de software e Modelos, métodos e técnicas da engenharia de software.

1.2 Escopo do Produto

O sistema de gerenciamento de Ferramentas de Software é uma aplicação web que permite aos usuários, realizar operações de Listagem a partir do tipo de Ferramentas e Nome de Ferramentas de Software.

1.3 Material de Referência

Para o desenvolvimento desse sistema utilizamos a metodologia ágil Kanban: <https://www.atlassian.com/br/agile/kanban>;

Para a implementação de commits usamos o conventional commits: <https://www.conventionalcommits.org/en/v1.0.0/> ;

Para boas práticas seguimos uma linha de raciocínio parecida com o livro Código Limpo de Robert Cecil Martin: [https://www.amazon.com.br/Código-Limpo-Robert CEMartinebook/dp/B085Q2K632](https://www.amazon.com.br/Código-Limpo-Robert-CEMartinebook/dp/B085Q2K632) ;

Para inspiração, utilizamos o site: <https://theresanaiforthat.com/>

1.4 Visão Geral do Sistema

Este documento está estruturado para fornecer uma visão detalhada dos requisitos do sistema de listagem de Ferramentas de Software. Ele inclui seções que descrevem os objetivos, escopo, materiais de referência, bem como uma visão geral das funcionalidades do sistema.

2 Descrição Geral do Sistema

2.1 Interfaces de Usuário

- Funcionará no Windows 7, 10, 11;
- Preparação do ambiente com JDK Java 17, IntelliJ IDEA, Git, PostgreSQL, HTML5, CSS3, VSCode.

3 Requisitos de Adaptação ao Ambiente

- Preparação do ambiente com JDK Java 17, IntelliJ IDEA, Git e PostgreSQL;
- Configuração do Back End com Spring Boot, Lombok, e Dev Tools;
- Utilização do Postman para testar e validar os endpoints da aplicação;
- Configuração do Front End com HTML5, CSS e VS Code;
- Automação do CI/CD com GitActions;
- Gerenciamento de projeto e colaboração com Jira Software, utilizando a Metodologia Ágil Kanban;
- Manutenção atualizada das dependências durante o desenvolvimento do projeto.

3.1 Funções do Produto

- Permite aos usuários buscar Ferramentas de Software específicos pelo filtro de Nome e Tipo de Ferramentas, retornando detalhes completos incluindo site (documentação da Ferramenta), Nome e tipo de Ferramenta.

3.2 Características dos Usuários

- **Usuários Padrão:** podem consultar tipos de Ferramentas de Software, dentre o filtro disponível a partir da aplicação (Nome, Tipo de Ferramenta).

4 Requisitos Específicos

4.1 Visão Geral

O sistema foi desenvolvido para facilitar a gestão de Ferramentas de Software, oferecendo funcionalidade importante. Os usuários podem buscar tipos de Ferramentas que necessitam no momento, através do nome ou pelo tipo da ferramenta. Sendo assim, facilitador para a busca de ferramentas de software em websites.

4.2 Requisitos Funcionais

- **Read (busca) de Ferramentas de Software**
 - Buscar Ferramentas de Software
 - Filtrar através do Nome ou Tipo de Ferramenta de Software

4.3 Requisitos não-funcionais

- **Tecnologias Utilizadas:**
 - Java 17;
 - Spring Boot;
 - IntelliJ IDEA;
 - PostgreSQL;
 - Git;
 - Postman;
 - HTML5;
 - CSS3;
 - Visual Studio Code;
 - Git Actions;
 - Jira Software;
 - Lombok;
 - Dev Tools.

5 Estudo de Viabilidade

5.1 Viabilidade Técnica

- **Tecnologias e Ferramentas Utilizadas:** O projeto utiliza uma variedade de tecnologias bem estabelecidas e amplamente adotadas no desenvolvimento de software.

5.2 Viabilidade Econômica:

- **Benefícios Potenciais:** O sistema de gerenciamento de Ferramentas de Software pode trazer benefícios significativos, como melhorar a eficiência na busca e gestão de ferramentas de software, o que pode resultar em uma economia de tempo e recurso para os usuários.

5.3 Viabilidade Operacional:

- **Manutenção e Suporte:** A manutenção atualizada das dependências durante o desenvolvimento do projeto e a utilização de práticas como automação CI/CD com GitActions podem contribuir para a operacionalidade contínua do sistema.
- **Adoção pelo Usuário:** A integração de metodologias ágeis como Kanban e o uso de tecnologias familiares aos usuários, como IntelliJ IDEA e VSCode, podem facilitar a adoção e a utilização do sistema.

6 Informações de Suporte

- Documentação Spring Boot: <https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/> ;
- Repositório do Projeto: <https://github.com/hanspeterdietiker/A3-Project-Unifacs> .