

**UNIVERSIDADE FEDERAL DO ACRE**

**CURSO DE BACHARELADO EM SISTEMAS DE INFORMAÇÃO**

**Documento de Requisitos do Sistema**

**Aplicação Web de Gerenciamento de Atividades**

**HISTÓRICO DE REVISÕES**

|  |  |  |  |
| --- | --- | --- | --- |
| Data | Versão | Descrição | Autor |
| 25/01/2023 | 0.0 | Elaboração do documento de requisitos | Gabriel Martins |
| 25/01/2023 | 0.0.1 | Revisão e adições no documento de requisitos | Clécio Elias |

**CONTEÚDO**

[1 INTRODUÇÃO 4](#_Toc125500413)

[1.1 IDENTIFICAÇÃO DOS REQUISITOS 4](#_Toc125500414)

[1.2 DEFINIÇÃO DE PRIORIDADES 4](#_Toc125500415)

[1.2.1 Essencial 4](#_Toc125500416)

[1.2.2 Importante 4](#_Toc125500417)

[1.1.3 Desejável 4](#_Toc125500418)

[2 REQUISITOS FUNCIONAIS 4](#_Toc125500419)

[2.1 [RF001] - Adicionar atividades 5](#_Toc125500420)

[2.2 [RF002] - Adicionar imagens, textos e editá-los nas atividades 5](#_Toc125500421)

[2.3 [RF003] - Adicionar subatividades dentro de uma atividade 5](#_Toc125500422)

[2.4 [RF004] - Possuir uma agenda com calendário 5](#_Toc125500424)

[2.5 [RF005] - Opção para adicionar notas 5](#_Toc125500423)

[2.6 [RF006] - Possuir uma hierarquia de atividades 5](#_Toc125500423)

[2.7 [RF007] - Opção para adicionar arquivos prontos](#_Toc125500423) 6

[2.8 [RF008] - Gerar relatórios sobre as atividades](#_Toc125500423) 6

[3 REQUISITOS NÃO FUNCIONAIS](#_Toc125500425) 6

[3.1 [NF001] - O sistema deve ser desenvolvido com boas práticas de programação.](#_Toc125500426) 7

[3.2 [NF002] - O sistema deve ter uma boa performance de velocidade.](#_Toc125500427) 7

[3.3 [NF003] - O sistema deve ser desenvolvido de forma a ser fácil de manter e atualizar.](#_Toc125500428) 7

[3.4 [NF004] – O sistema deve ser desenvolvido com usabilidade e acessibilidade em mente.](#_Toc125500429) 7

[3.5 [NF005] – O sistema deve ser seguro e prover tratativa de erros. 7](#_Toc125500430)

4 [FLUXOGRAMA DE REQUISITOS](#_Toc125500425) 7

[REFERÊNCIAS BIBLIOGRÁFICAS 7](#_Toc125500431)

# 1 INTRODUÇÃO

Este documento especifica as funcionalidades e restrições de uma Aplicação Web Para Gerenciamento de Atividades, dando uma visão das características gerais do sistema e seus requisitos funcionais e não funcionais.

## 1.1 IDENTIFICAÇÃO DOS REQUISITOS

Por convenção, a referência a requisitos é feita através do nome da subseção onde eles estão descritos seguidos do identificador do requisito, de acordo com a especificação a seguir: [nome da subseção. identificador do requisito]. Por exemplo [Adicionar atividades.RF001] se refere ao requisito funcional da subseção Adicionar Atividades com identificador de requisito [RF001].

## 1.2 DEFINIÇÃO DE PRIORIDADES

### 1.2.1 Essencial

Requisitos indispensáveis e que afetam diretamente no bom funcionamento do sistema, tornando-os obrigatórios e máxima prioridade.

### 1.2.2 Importante

Requisitos sem os quais o sistema pode funcionar corretamente, porém não de forma totalmente satisfatória, portanto, sendo de prioridade moderada.

### 1.1.3 Desejável

Funcionalidades extras que agregam ao sistema, mas não possuem impacto no seu funcionamento básico, resultando em uma menor prioridade.

# 2 REQUISITOS FUNCIONAIS

Descrições das diversas funções que clientes e usuários precisam que o software ofereça, ou também, os casos de uso.

## 2.1 [RF001] - Adicionar atividades

**Prioridade:** ☒ Essencial ☐ Importante ☐ Desejável

Possibilidade de adicionar título, descrição, data de início, fim e prioridade.

Possibilidade de adicionar imagens relacionadas à atividade.

## 2.2 [RF002] - Adicionar imagens, textos e editá-los nas atividades

**Prioridade:** ☒ Essencial ☐ Importante ☐ Desejável

Possibilidade de editar os detalhes de uma atividade já adicionada, incluindo título, descrição, data de início e fim, e prioridade.

## 2.3 [RF003] - Adicionar subatividades dentro de uma atividade

**Prioridade:** ☒ Essencial ☐ Importante ☐ Desejável

Possibilidade de adicionar subatividades dentro de uma atividade principal, formando uma hierarquia de atividades.

## 2.4 [RF004] - Possuir uma agenda com calendário

**Prioridade:** ☐ Essencial ☒ Importante ☐ Desejável

Possibilidade de visualizar as atividades em um calendário, com possibilidade de filtrar por data e prioridade, além de agendar atividades para o futuro.

## 2.5 [RF005] – Opção para adicionar notas

**Prioridade:** ☐ Essencial ☒ Importante ☐ Desejável

Possibilidade de adicionar “stick notes” ou “notas de rodapé” tanto nas atividades/subatividades quanto nos dias do calendario.

## 2.6 [RF006] – Possuir uma hierarquia de atividades

**Prioridade:** ☐ Essencial ☒ Importante ☐ Desejável

Possuir uma hierarquia visual de atividades e subatividades, podendo abri-las para visualização e/ou edição das mesmas.

## 2.7 [RF007] – Opção para adicionar arquivos prontos

**Prioridade:** ☐ Essencial ☒ Importante ☐ Desejável

Possibilidade de adicionar relatórios, atividades, evidências e ou arquivos já prontos na plataforma.

## 2.8 [RF008] - Gerar relatórios sobre as atividades

**Prioridade:** ☐ Essencial ☒ Importante ☐ Desejável

Possibilidade de gerar relatórios sobre as atividades, incluindo informações como título, descrição, data de início e fim, prioridade, além da opção de editar o relatório gerado já que o template pode ser relativo.

# 3 REQUISITOS NÃO FUNCIONAIS

Propriedades do software como manutenibilidade, usabilidade, desempenho, custos e outros.

## 3.1 [NF001] - O sistema deve ser desenvolvido com boas práticas de programação.

**Prioridade:** ☐ Essencial ☒ Importante ☐ Desejável

Algumas boas práticas incluem:

* Utilizar um padrão de arquitetura de software consistente e bem estabelecido, como o MVC (Model-View-Controller).
* Utilizar uma metodologia de desenvolvimento de software, como o Scrum ou o Kanban.
* Utilizar um controle de versão, como o Git, para garantir que as alterações no código sejam rastreadas e revertidas caso necessário, além de ter em registro todo o desenvolvimento do software (github do projeto: <https://github.com/Cleps/plataforma_web-EngSoftw> ).
* Utilizar uma estratégia de gerenciamento de dependências para garantir que as bibliotecas e frameworks utilizados no projeto sejam atualizados de forma controlada e segura.

## 3.2 [NF002] - O sistema deve ter uma boa performance de velocidade.

**Prioridade:** ☒ Essencial ☐ Importante ☐ Desejável

O sistema deve ter um bom desempenho nas transações processadas/segundo, bem como um bom tempo de resposta ao usuário.

## 3.3 [NF003] - O sistema deve ser desenvolvido de forma a ser fácil de manter e atualizar.

**Prioridade:** ☒ Essencial ☐ Importante ☐ Desejável

Diz respeito à manutenibilidade do sistema, que se alia às boas práticas de programação, isso inclui uma boa documentação e escrita limpa do código.

## 3.4 [NF004] – O sistema deve ser desenvolvido com usabilidade e acessibilidade em mente.

**Prioridade:** ☒ Essencial ☐ Importante ☐ Desejável

O sistema deve ser fácil e intuitivo de usar, além de ser acessível a diferentes tipos de dispositivos e resoluções de tela.

## 3.5 [NF005] – O sistema deve ser seguro e prover tratativa de erros.

**Prioridade:** ☒ Essencial ☐ Importante ☐ Desejável

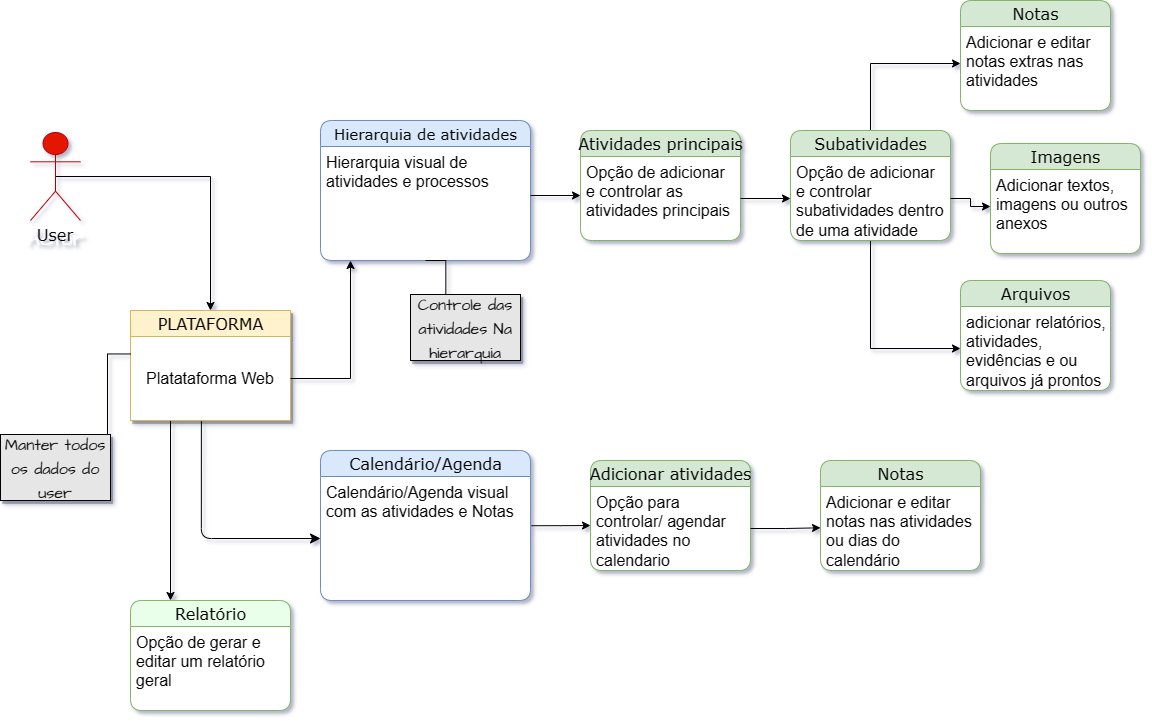
O sistema deve ser confiável e fornecer feedback de possíveis erros ao usuário, além de ter validações de segurança básicas.

## 3.6 [NF006] – O sistema deve ser desenvolvido como uma plataforma web.

**Prioridade:** ☒ Essencial ☐ Importante ☐ Desejável

O sistema deve ser desenvolvido como uma plataforma web que possa ser acessada em qualquer dispositivo por meio de um navegador web.

# 4 FLUXOGRAMA DE REQUISITOS

Figura 1. Fluxograma com os requisitos resumidos.

# REFERÊNCIAS BIBLIOGRÁFICAS

DA SILVA, Felipe Rook. Modulo de Avaliaçao Acadêmica no SIGA Ensino, 2016. Disponível em: https://www2.ufjf.br/diavi//files/2016/07/DocumentosdeRequisitosdoSistema.pdf

CUNHA, Eduardo. Documento de Requisitos, 2011. Disponível em: https://eduardocunha11.github.io/firstblog/aulas/gsi522/P4-Systrack.pdf