**Plano de Projeto**

***Correio Web***

**Versão 2.0**

**Histórico de Alterações**

| **Data** | **Versão** | **Descrição** | **Autores** |
| --- | --- | --- | --- |
| 14/10/2024 | 1.0 | Definição de requisitos, primeiros protótipos e plano de projeto. | Catarina Freisleben, Felipe Antunes, Geffté Caetano, Lucas Arruda e Maria dos Anjos. |
| 28/10/2024 | 2.0 | Incluir tecnologia utilizada, melhorar a descrição do escopo do produto e inserção do Diagrama de Classes de Entidades. | Catarina Freisleben, Felipe Antunes, Geffté Caetano, Lucas Arruda e Maria dos Anjos. |

## **Introdução**

Este documento registra o Plano de Projeto de Desenvolvimento do Sistema Correio Web.

## **Escopo do Projeto**

O escopo do projeto delimita o trabalho que deve ser feito ao longo do projeto. Desta forma, a presente seção descreve o escopo do produto, que contém a lista de funcionalidades a serem implementadas no projeto, bem como as restrições do projeto e suas premissas, quando existirem.

### Escopo do Produto

O sistema de caixa postal digital será responsável por facilitar a comunicação entre usuários internos, permitindo a gestão de mensagens. As funcionalidades previstas para serem implementadas no projeto incluem:

* **CRUD:**

**Mensagem:**

Envio de mensagens: permite a escrita e envio de mensagens para outros usuários.

Leitura de mensagens: permite a visualização das mensagens recebidas.

Alteração do status de leitura: permite que o usuário altere o status das mensagens como lidas ou não lidas.

Exclusão de mensagens: permite que o usuário exclua as mensagens recebidas.

**Usuário:**

Cadastrar: permite que um usuário se cadastre no site.

Alteração de usuário: permite que o usuário altere uma informação cadastrada.

Exclusão de conta: permite que o usuário exclua sua conta no site.

* **Login de usuário**: permite autenticar o usuário para acessar.

### Restrições e Premissas do Projeto

Nesta seção, na Tabela 1, são descritas as restrições e as premissas do projeto. As restrições documentam qualquer limitação referente a algum compromisso do projeto, que necessariamente deve ser atendida. As premissas, por outro lado, definem condições que devem ser satisfeitas para que o plano de projeto proposto seja realmente exequível.

| **Tipo** | **Descrição** |
| --- | --- |
| Autenticação obrigatória | Apenas usuários autenticados poderão acessar e gerenciar mensagens dentro do sistema​. |
| Acesso restrito | Cada usuário interno só poderá acessar suas próprias mensagens, sem permissão para visualizar ou interagir com mensagens de outros usuários​. |

**Tabela 1 - Premissas e restrições do projeto.**

## **Ferramentas e Tecnologias de Desenvolvimento**

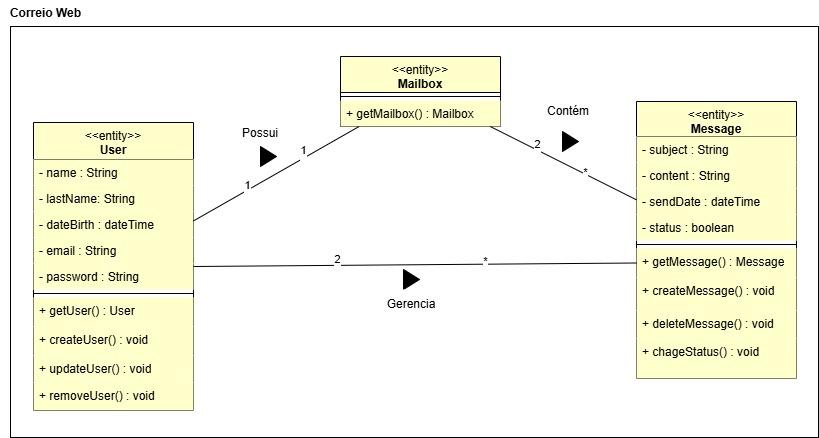
Nesta seção são descritas as ferramentas e tecnologias adotadas para o desenvolvimento do projeto. Para cada ferramenta ou tecnologia utilizada, indicar a categoria (planejamento e acompanhamento de projeto, modelagem, controle de versões, comunicação, linguagem de programação backend, linguagem de programação frontend, confecção de protótipo, ferramenta de teste funcional, ferramenta de teste de unidade, etc), o nome e uma breve justificativa da escolha.

| **Categoria** | **Ferramenta/**  **Tecnologia** | **Justificativa** |
| --- | --- | --- |
| Confecção de Protótipo | Figma | Ferramenta que colabora no design da interface do usuário (UI). |
| Controle de Versões | GitHub | Facilita o acompanhamento do desenvolvimento. |
| Linguagem de Marcação de Hipertexto | HTML | Estrutura o conteúdo da aplicação. |
| Linguagem de Programação Frontend | CSS | Estiliza a interface e define o layout. |
| Linguagem de Programação Frontend | JavaScript | Manipula elementos dinâmicos e controle de interações de usuário, além de realizar comunicação assíncrona com o backend. |
| Linguagem de programação backend | Node.js | Software para conectar com o banco de dados. |
| Sistema de Gerenciamento de Banco de Dados (SGBD) | PostgreSQL   |  | | --- | | Sistema de gerenciamento de banco de dados (SGBD) utilizado para armazenar informações sobre usuários e mensagens. |

**Tabela 2 - Ferramentas e tecnologias utilizadas no projeto.**

## **Planejamento de Banco de Dados**

Nesta seção é definida a estrutura do banco de dados do projeto. Cada entidade principal do sistema foi mapeada no Diagrama de Classes de Entidades, Imagem 1, que visa representar as principais tabelas e relações do banco de dados da aplicação.



**Imagem 1 - Diagrama de Classes de Entidades.**

O diagrama de classes apresentado acima representa a estrutura de entidades principais do sistema Correio Web, definindo as classes **User**, **Mailbox** e **Message**. Cada classe é detalhada com seus atributos e métodos essenciais, além das associações entre elas, que refletem o relacionamento entre os elementos da aplicação.

## **Descrição das Entidades e Relacionamentos**

* **User**: classe que representa os usuários do sistema.
  + **Atributos**: name, lastName, dateBirth, email e password.
  + **Métodos**: getUser() que recupera os dados do usuário, createUser(), updateUser() e removeUser() que permitem gerenciar as informações dos usuários.
  + **Relacionamentos**: cada usuário (User) possui uma única caixa postal (Mailbox) associada, e cada usuário (User) pode gerenciar inúmeras mensagens (Message).
* **Mailbox**: é uma entidade intermediária entre o usuário (User) e as mensagens (Message).
  + **Métodos**: getMailbox() que retorna a caixa postal associada ao usuário (User).
  + **Relacionamentos**: cada caixa postal (Mailbox) pode conter múltiplas mensagens (Message), mas está associada exclusivamente a um único usuário (User).
* **Message:** esta classe representa as mensagens dentro da Mailbox,
  + **Atributos:** subject, content, sendDate e status.
  + **Métodos**: getMessage() que retorna uma mensagem específica, createMessage(), deleteMessage() servem para a criação e remoção de mensagens e changeStatus() altera o status de uma mensagem.
  + **Relacionamentos:** cada mensagem (Message) está associada a duas caixas postais (Mailbox), a do emissor (User) que a escreveu e do receptor (User) que a recebeu, assim como cada mensagem (Message) é escrita por emissor (User) e recebida por um receptor (User).