



DOCUMENTAÇÃO DO SISTEMA MANGABA CERRADO

Empresa Júnior CodaMais  
Tecnologia em Sistemas para Internet

2025/1

## 

Sumário

[**1. Equipe e gestão do projeto 3**](#_2gi7v9s6wnwg)

[Gestão do Projeto 3](#_8dawaqlm2iqp)

[**2. Descrição, características e usuários do sistema (Sprint 1 - Revisão) 3**](#_blpwls2t94ks)

[**3. Requisitos (Sprint 1 - Revisão) 4**](#_o55ragmc6jsg)

[3.1. Requisitos funcionais 4](#_otlizpm4jctv)

[3.2. Requisitos NÃO funcionais 5](#_dty8c9ijp93y)

[**4. Diagrama de casos de uso (Sprint 1 - Revisão) 5**](#_husiyegbsug8)

[**5. Banco de dados (Sprint 1 e 2 - Revisão) 6**](#_8mkltlzce58v)

[5.1. Modelo relacional diagramado (Sprint 1) 6](#_7oi3ibjzj94q)

[5.2. Modelo físico do banco de dados (Sprint 2) 6](#_bdpkgpr4chfx)

[**6. Front-end (Sprint 2) 6**](#_1uowarmot3ml)

[6.1. Protótipos (Sprint 1 - Revisão) 6](#_cvliipgvbo78)

[6.2. Implementação (Sprint 2) 12](#_ab81srzak6yh)

[**7. Back-end (Sprint 2 e 3) 12**](#_staadnttjhui)

[7.1. Diagrama de classes (Sprint 2) 12](#_2dv3jsuabmfy)

[7.2. Implementação do back-end (Sprint 3) 13](#_iwckd71dhm76)

[7.2.1 Suporte a estatística (Sprint 3) 13](#_c973a7y5nm5t)

[**8. Integração front-end e back-end (Sprint 4) 13**](#_6lsgx38qq0uw)

[**9. Referências 14**](#_4bv9t8hx5pun)

## 

## 1. Equipe e gestão do projeto

A equipe responsável pelo desenvolvimento do Sistema de Mapeamento do Cerrado é composta pelos seguintes membros e suas respectivas responsabilidades:

* **Ana**: Implementação de HTML e CSS
* **Pedro**: Implementação de HTML e CSS
* **Sandro**: Banco de dados, JavaScript e documentação
* **Lalia**: Banco de dados e documentação
* **Maria Fernanda**: Implementação de HTML e CSS
* **Maria Vitória**: Documentação e JavaScript
* **Thaís**: Banco de dados e JavaScript
* **Pablo**: Design e apoio na construção do documento e do sistema

### Gestão do Projeto

A gestão do projeto é realizada de maneira colaborativa, com as seguintes práticas e processos:

* **Divisão de Tarefas**: As tarefas são distribuídas conforme as habilidades e especialidades de cada membro da equipe, garantindo uma alocação eficiente dos recursos.
* **Comunicação do Andamento**: O progresso de cada tarefa é monitorado e comunicado por meio de reuniões semanais, onde são discutidos os avanços, dificuldades e ajustes necessários.
* **Registro de Tarefas**: Todas as tarefas a serem realizadas são registradas em uma ferramenta de gerenciamento de projetos, permitindo a visualização do progresso e garantindo o acompanhamento adequado.
* **Reuniões da Equipe**: A equipe se reúne semanalmente para revisar o andamento das atividades, alinhando objetivos e ajustando as estratégias conforme necessário.

Essa estrutura organizacional e as práticas de gestão são fundamentais para o bom andamento do desenvolvimento do sistema e para garantir o cumprimento dos prazos estabelecidos.

## 2. Descrição, características e usuários do sistema (Sprint 1 - Revisão)

Responder às perguntas de **quem usará o sistema**, **o que o sistema fará**, **onde e quando ele será usado**. Nesta seção, a equipe de projeto **examina sistemas atuais**, identifica as **oportunidades de aperfeiçoamento** e **desenvolve um conceito para o novo sistema**.

## 3. Requisitos (Sprint 1 - Revisão)

O desenvolvimento de um sistema eficaz começa com a definição de seus requisitos, que são as características essenciais ou funcionalidades que o sistema deve possuir para atingir seus objetivos. Para o Sistema de Mapeamento do Cerrado, trabalhamos com a equipe envolvida no projeto, composta por professores, alunos e representantes da Secretaria de Educação, para obter uma visão clara sobre as necessidades e expectativas em relação ao sistema. Realizamos reuniões presenciais e levantamentos detalhados para compreender as funcionalidades que o sistema deve oferecer e as especificações técnicas necessárias.

Durante as discussões, foi apresentado um protótipo inicial, e os participantes foram questionados sobre as funcionalidades essenciais, a usabilidade da plataforma e os requisitos de segurança. Também foram compartilhados sistemas semelhantes, como plataformas de geolocalização e redes sociais, para ilustrar como algumas funcionalidades poderiam ser integradas ao sistema.

Com base nas informações coletadas e nas sugestões dos envolvidos, os requisitos funcionais e não funcionais foram definidos e organizados nas subseções 3.1 e 3.2 deste documento. Esses requisitos são essenciais para o desenvolvimento do sistema e garantirão que ele atenda tanto às necessidades da comunidade escolar quanto às exigências técnicas para garantir sua operação eficaz e segura.

### 3.1. Requisitos funcionais

**RF01 - Registro e Geolocalização de Espécies:** O sistema deve permitir o registro de espécies do Cerrado com a geolocalização de sua localização exata. Usuários devem poder marcar pontos no mapa, onde as espécies foram avistadas ou registradas.

**RF02 - Postagens de Conteúdos com Hashtags:** O sistema deve permitir que os usuários postem conteúdos relacionados ao Cerrado, incluindo textos, imagens e vídeos, com a obrigatoriedade de adicionar hashtags relacionadas ao conteúdo, que serão usadas para categorizar os posts e gerar trending topics.

**RF03 - Interação com Postagens:** Usuários devem poder interagir com postagens de outros, seja por meio de curtidas, comentários ou compartilhamentos, além de poderem buscar conteúdos específicos por localização ou hashtags.

**RF04 - Moderação de Conteúdos (Professores):** Professores terão a capacidade de moderar as postagens, podendo validar ou invalidar o conteúdo postado pelos usuários, bem como banir usuários que violem as regras da plataforma.

**RF05 - Sistema de Avaliação Mútua:** Os professores deverão poder avaliar as postagens de alunos e vice-versa, incentivando a troca de informações e o aprendizado colaborativo sobre o Cerrado.

**RF06 - Cadastro de Usuários com Validação de Identidade:** O sistema permitirá que os usuários se cadastrem com e-mails institucionais e contará com um processo de validação de contas. No futuro, a integração com o Google poderá ser implementada.

### 3.2. Requisitos NÃO funcionais

**RNF01 - Design Responsivo:** O sistema deve ser responsivo, ou seja, deve adaptar-se bem a diferentes dispositivos, incluindo desktops, tablets e smartphones, garantindo a acessibilidade e uma boa experiência de uso.

**RNF02 - Segurança no Cadastro de Menores de Idade:** A plataforma deve garantir que menores de idade possam se cadastrar de forma segura, com a implementação de mecanismos que protejam seus dados pessoais e sua privacidade, conforme as diretrizes legais de proteção de dados.

**RNF03 - Desempenho e Escalabilidade:** O sistema deve ser capaz de suportar um número crescente de usuários e postagens sem comprometer o desempenho. Deve ser escalável para acomodar novas funcionalidades e aumentar a capacidade conforme o crescimento da plataforma.

**RNF04 - Acessibilidade:** O sistema deve seguir as melhores práticas de acessibilidade, garantindo que pessoas com deficiências possam navegar e interagir com a plataforma sem dificuldades. Isso inclui suporte a leitores de tela, contraste adequado de cores e navegação por teclado.

**RNF05 - Segurança da Informação:** A plataforma deve garantir a proteção das informações dos usuários por meio de criptografia e mecanismos de segurança que protejam os dados armazenados e trafegados no sistema, especialmente os dados pessoais dos usuários, como e-mails e informações de geolocalização.

## 4. Diagrama de casos de uso (Sprint 1 - Revisão)

## 5. Banco de dados (Sprint 1 e 2 - Revisão)

### 5.1. Modelo relacional diagramado (Sprint 1)

Fazer a leitura do caderno disponível na referência [3].

### 5.2. Modelo físico do banco de dados (Sprint 2)

CREATE DATABASE MangabaCerrado;

GO

USE MangabaCerrado;

GO

-- Tabela de Usuários

CREATE TABLE Usuarios (

id\_usuario INT IDENTITY(1,1) PRIMARY KEY,

nome NVARCHAR(255) NOT NULL,

email NVARCHAR(255) UNIQUE NOT NULL,

senha\_hash NVARCHAR(255) NULL, -- Para login via email

google\_id NVARCHAR(255) NULL, -- Para login via Google

tipo\_usuario NVARCHAR(50) CHECK (tipo\_usuario IN ('aluno', 'professor', 'admin')) NOT NULL,

data\_cadastro DATETIME DEFAULT GETDATE()

);

-- Tabela de Postagens

CREATE TABLE Postagens (

id\_postagem INT IDENTITY(1,1) PRIMARY KEY,

id\_usuario INT FOREIGN KEY REFERENCES Usuarios(id\_usuario) ON DELETE CASCADE,

titulo NVARCHAR(255) NOT NULL,

descricao TEXT NOT NULL,

localizacao GEOGRAPHY NULL,

data\_postagem DATETIME DEFAULT GETDATE()

);

-- Tabela de Fotos das Postagens

CREATE TABLE Fotos (

id\_foto INT IDENTITY(1,1) PRIMARY KEY,

id\_postagem INT FOREIGN KEY REFERENCES Postagens(id\_postagem) ON DELETE CASCADE,

url\_foto NVARCHAR(500) NOT NULL

);

-- Tabela de Curtidas

CREATE TABLE Curtidas (

id\_like INT IDENTITY(1,1) PRIMARY KEY,

id\_usuario INT FOREIGN KEY REFERENCES Usuarios(id\_usuario) ON DELETE CASCADE,

id\_postagem INT FOREIGN KEY REFERENCES Postagens(id\_postagem) ON DELETE CASCADE,

data\_like DATETIME DEFAULT GETDATE(),

UNIQUE (id\_usuario, id\_postagem) -- Evita duplicação de curtidas

);

-- Tabela de Comentários

CREATE TABLE Comentarios (

id\_comentario INT IDENTITY(1,1) PRIMARY KEY,

id\_usuario INT FOREIGN KEY REFERENCES Usuarios(id\_usuario) ON DELETE CASCADE,

id\_postagem INT FOREIGN KEY REFERENCES Postagens(id\_postagem) ON DELETE CASCADE,

texto TEXT NOT NULL,

data\_comentario DATETIME DEFAULT GETDATE()

);

-- Tabela de Hashtags

CREATE TABLE Hashtags (

id\_hashtag INT IDENTITY(1,1) PRIMARY KEY,

nome NVARCHAR(100) UNIQUE NOT NULL

);

-- Relacionamento entre Postagens e Hashtags

CREATE TABLE PostagemHashtags (

id\_postagem INT FOREIGN KEY REFERENCES Postagens(id\_postagem) ON DELETE CASCADE,

id\_hashtag INT FOREIGN KEY REFERENCES Hashtags(id\_hashtag) ON DELETE CASCADE,

PRIMARY KEY (id\_postagem, id\_hashtag)

);

-- Tabela de Moderação (para Professores/Admins validarem postagens)

CREATE TABLE Moderacao (

id\_moderacao INT IDENTITY(1,1) PRIMARY KEY,

id\_postagem INT FOREIGN KEY REFERENCES Postagens(id\_postagem) ON DELETE CASCADE,

id\_moderador INT FOREIGN KEY REFERENCES Usuarios(id\_usuario) ON DELETE CASCADE,

status NVARCHAR(50) CHECK (status IN ('aprovado', 'rejeitado')) NOT NULL,

comentario\_moderacao TEXT NULL,

data\_moderacao DATETIME DEFAULT GETDATE()

);

-- Tabela de Pontuação para Professores

CREATE TABLE Pontuacao (

id\_usuario INT FOREIGN KEY REFERENCES Usuarios(id\_usuario) ON DELETE CASCADE,

pontos INT DEFAULT 0,

PRIMARY KEY (id\_usuario)

);

## 6. Front-end (Sprint 2)

É esperado nessa entrega uma página desenvolvida. Deve ser incluído no documento imagens das telas desenvolvidas. Caso as telas implementadas estejam diferentes do protótipo no aspecto visual e funcional, descrever o que teve que ser alterado e por quê.

Além disso, sugere-se que a página seja hospedada e disponibilizado o link para testes, contudo a hospedagem da página não é um requisito obrigatório. Pode-se utilizar o GitHub pages, por exemplo, conforme o tutorial disponível [neste link](https://www.alura.com.br/artigos/como-colocar-projeto-no-ar-com-github-pages).

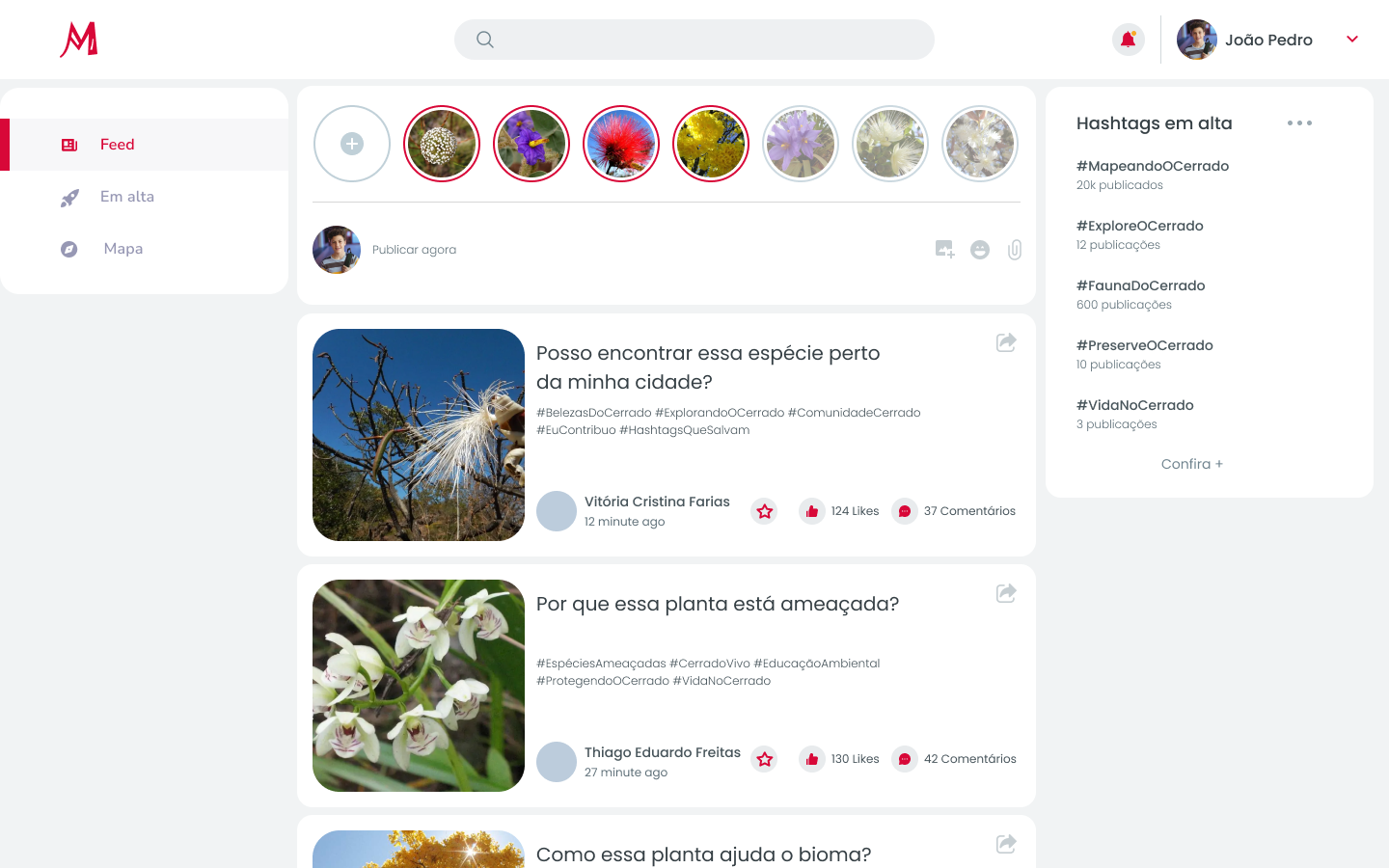
### 6.1. Protótipos (Sprint 1 - Revisão)

Uma interface com o usuário refere-se ao conjunto de elementos com os quais o usuário interage diretamente em um sistema. Essa interface abrange as telas exibidas, os controles utilizados para navegação, os formulários para captura de dados, bem como os relatórios gerados pelo sistema, seja em formato físico, digital ou por meio de outras mídias. A eficácia de uma interface depende de sua capacidade de proporcionar uma experiência fluida e intuitiva, facilitando a interação do usuário com o sistema de maneira eficiente e sem ambiguidades.

Os protótipos apresentados a seguir representam uma diretriz para o desenvolvimento do sistema, servindo como uma visualização das funcionalidades que serão implementadas. Eles são uma ferramenta essencial no processo de design da interface, permitindo uma melhor compreensão das interações e do fluxo do usuário antes da implementação final.

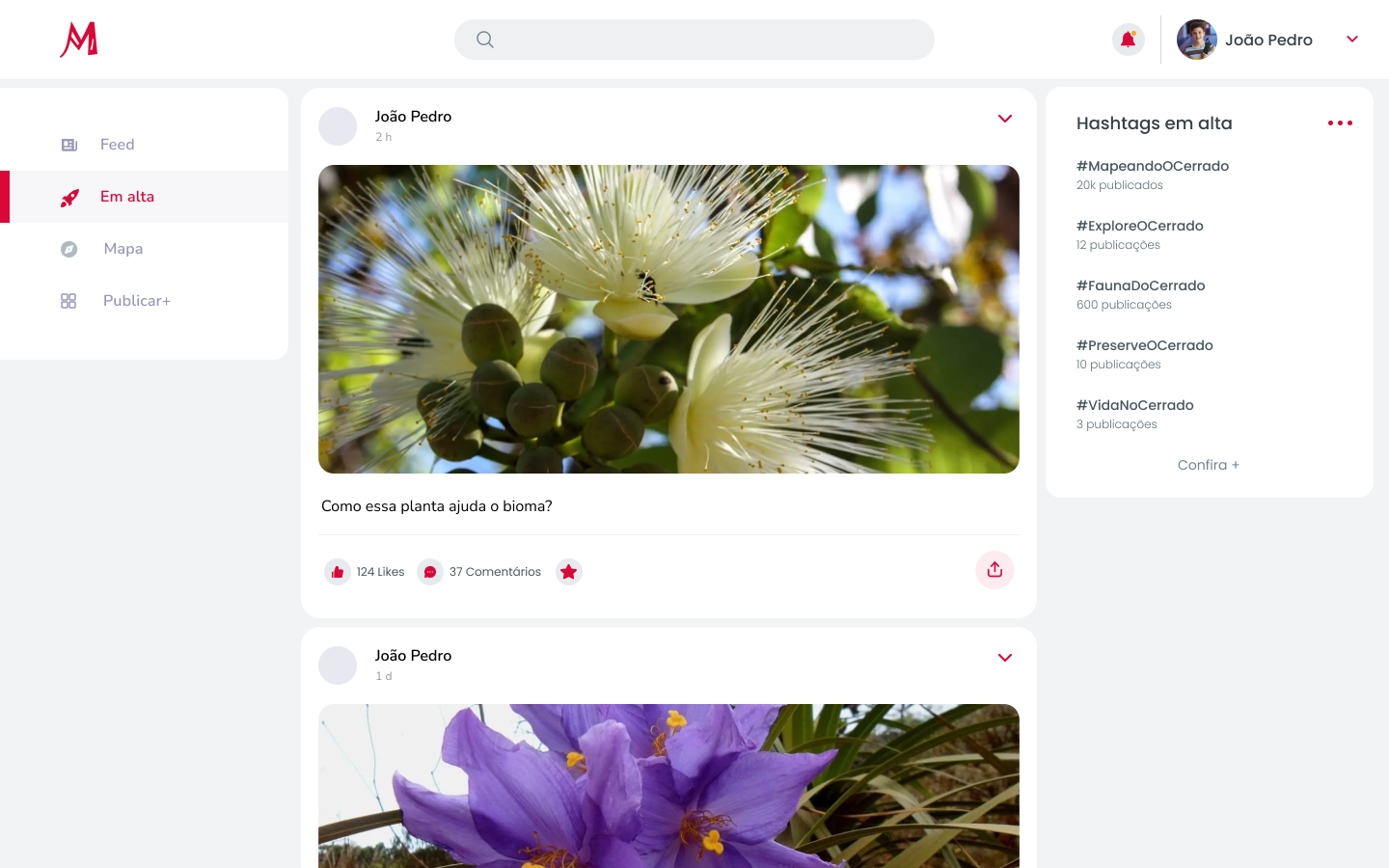
Como destaca o autor Somerville (2011), "os protótipos desempenham um papel crucial no design da interface, fornecendo uma representação tangível do sistema para facilitar a validação precoce das decisões de design e a identificação de melhorias necessárias."

Figura 1 - Página Feed do aluno versão web



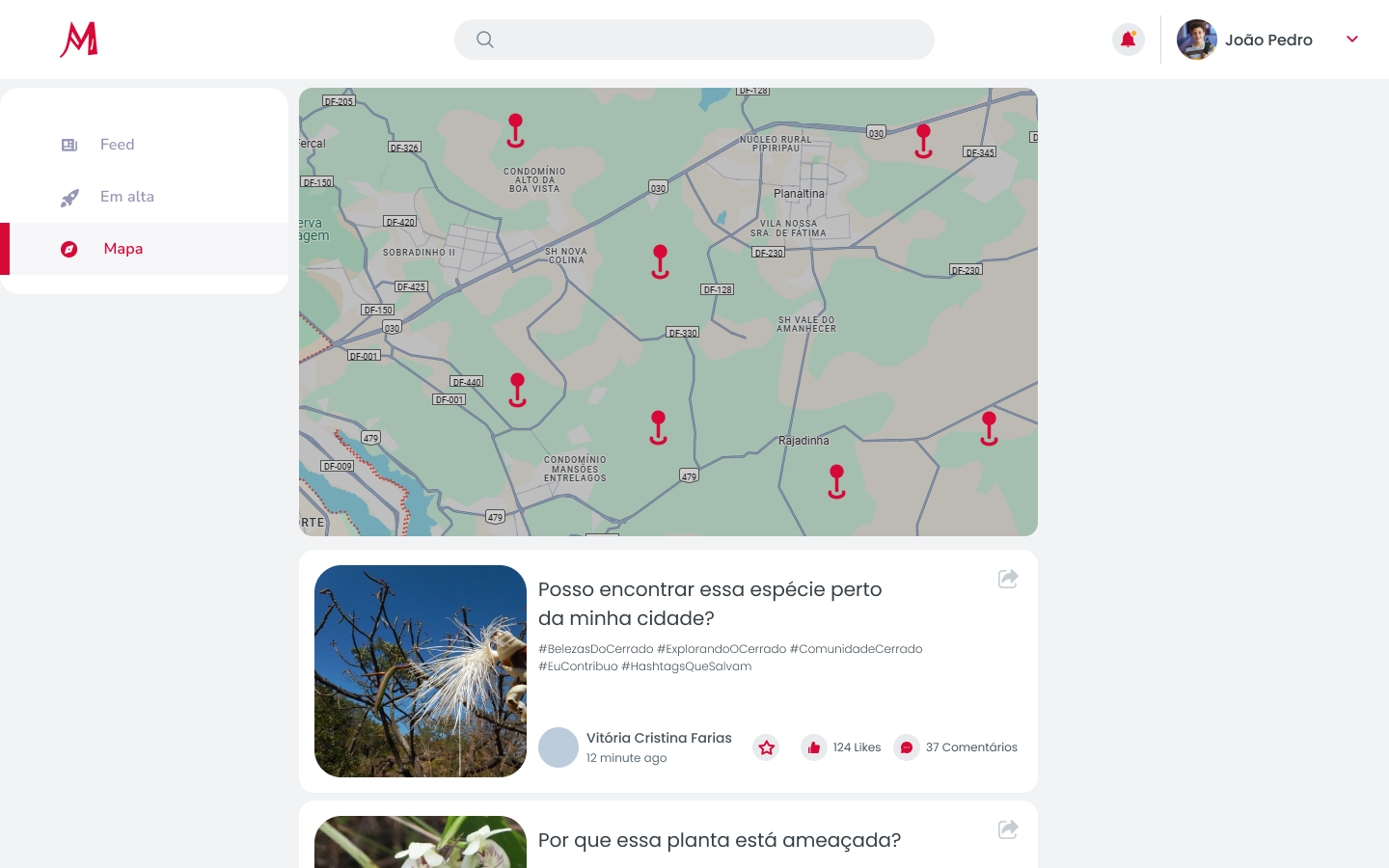
A figura () apresenta a página de Feed do Aluno é onde os estudantes podem visualizar e interagir com as postagens de seus colegas, tanto no feed principal, que exibe publicações permanentes organizadas cronologicamente, quanto nas postagens temporárias (stories), que funcionam de maneira semelhante aos stories do Instagram e desaparecem após um período. No centro da tela, o aluno pode criar e compartilhar publicações em seu próprio perfil, adicionando textos, imagens e outros conteúdos. Já no lado direito da página, há uma seção que exibe as hashtags em alta do dia, permitindo que os alunos acompanhem os temas mais comentados e engajem-se nas discussões mais relevantes dentro da plataforma.

Figura 1 - Página Em Alta do aluno versão web



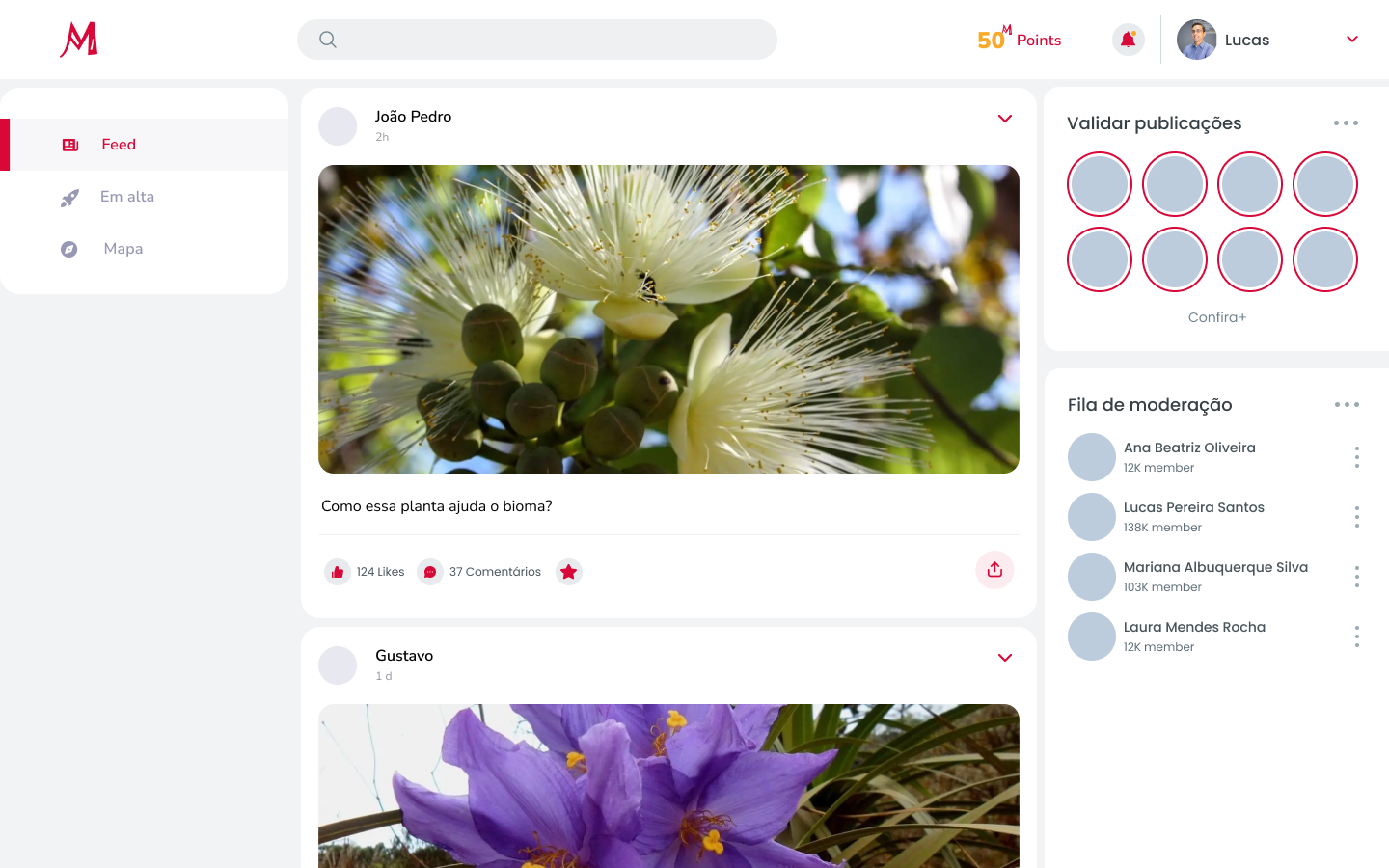
A figura () apresenta a página **Em Alta**, onde os estudantes podem visualizar as postagens mais relevantes do sistema, destacadas com base no engajamento da comunidade. Essa seção funciona de maneira semelhante à área de **hashtags em alta**, mas, em vez de exibir apenas tendências temáticas, reúne as publicações com maior número de interações, como curtidas, comentários e compartilhamentos. Dessa forma, os alunos podem acompanhar os conteúdos mais discutidos e participar ativamente das conversas que estão movimentando a plataforma.

Figura 1 - Página Mapa do aluno versão web



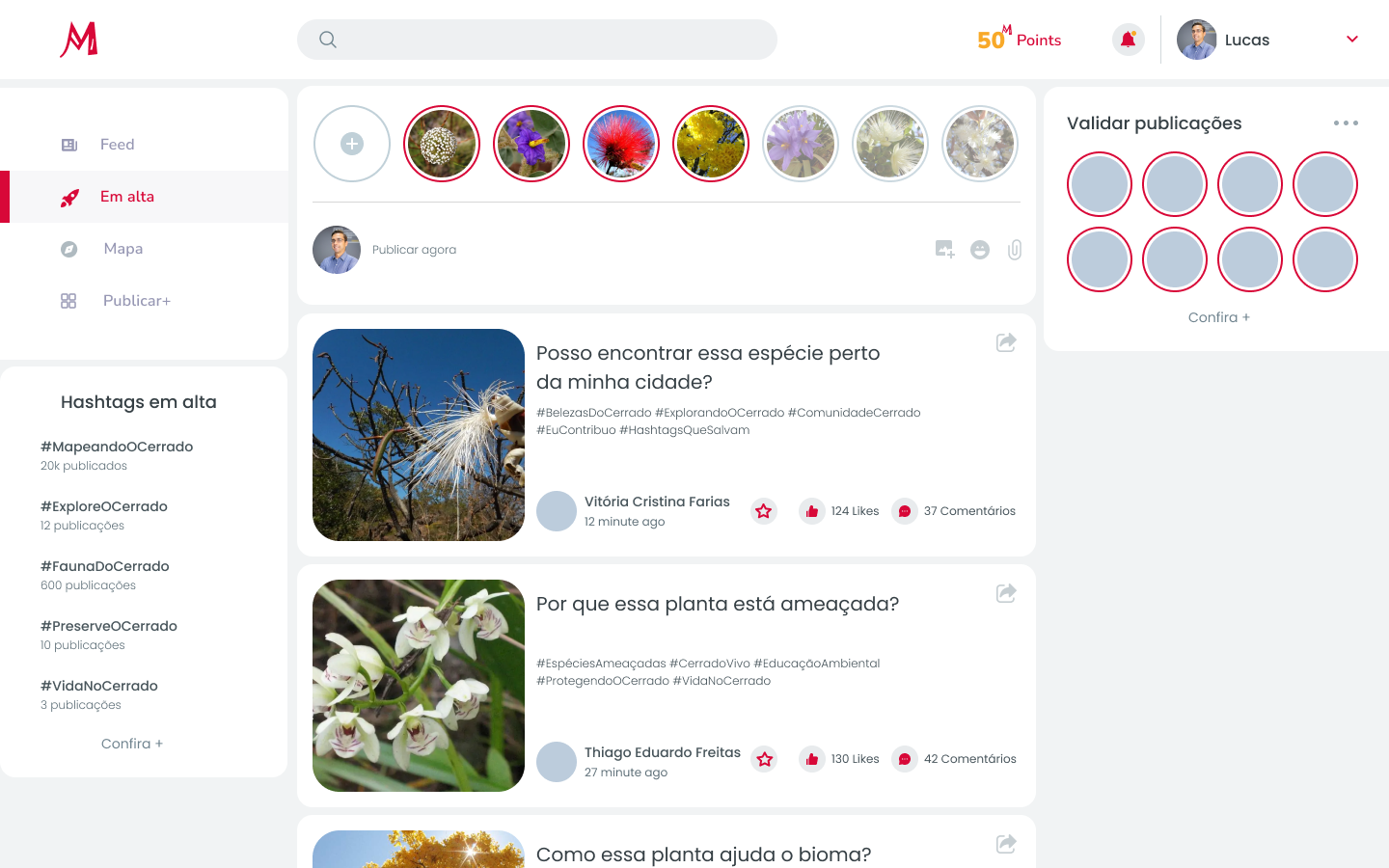
A figura () apresenta a página **Mapas**, onde os alunos podem visualizar as áreas geográficas onde plantas, frutas e animais foram registrados dentro do sistema. A página interativa exibe um mapa detalhado, permitindo que o usuário explore diferentes regiões e descubra os locais específicos onde essas espécies foram fotografadas. Além disso, a página inclui **cards educativos** que fornecem informações sobre cada item registrado, como características, habitat e curiosidades, oferecendo uma experiência de aprendizado enriquecedora. A combinação do mapa com os cards educativos torna a exploração mais interativa e educativa, permitindo que os estudantes aprendam de forma prática e imersiva.

Figura 1 - Página Feed do professor versão web



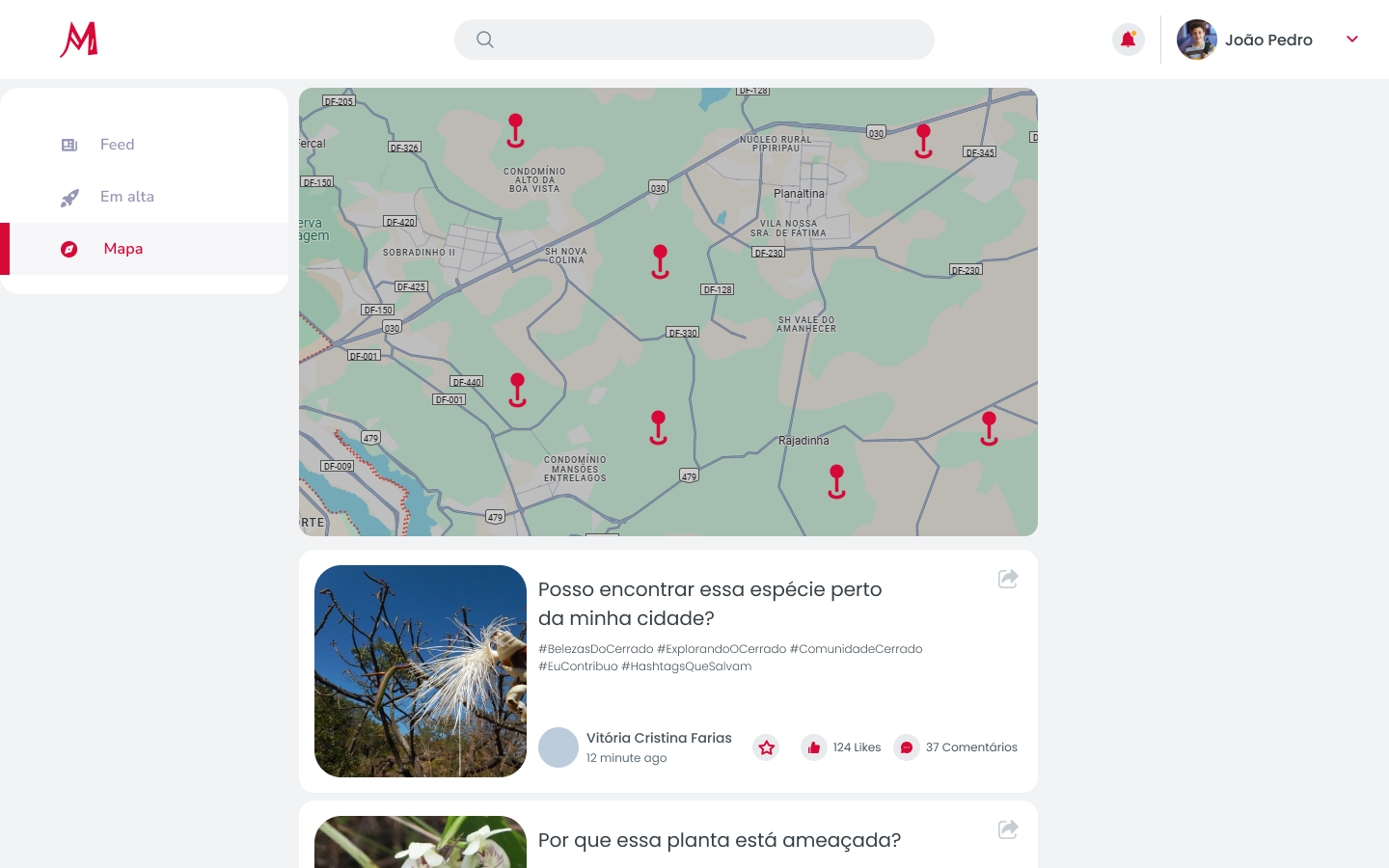
A figura () apresenta a página Feed do Professor, onde os docentes podem visualizar e interagir com as postagens realizadas pelos colegas. O feed principal exibe as publicações permanentes, organizadas cronologicamente, enquanto as postagens temporárias (stories), semelhantes aos stories do Instagram, desaparecem após um período determinado. No centro da tela, o professor pode criar e compartilhar suas próprias postagens, incluindo textos, imagens e outros tipos de conteúdo. À direita da página, uma seção dedicada exibe as hashtags em alta do dia, permitindo que o docente acompanhe as discussões mais relevantes e participe dos temas mais comentados.

Figura 1 - Página Em Alta do professor versão web



A figura () apresenta a página Em Alta, onde o professor pode visualizar as postagens que estão recebendo maior engajamento dentro da plataforma. Semelhante à seção de hashtags em alta, essa página reúne as postagens mais comentadas, compartilhadas e curtidas, permitindo ao docente acompanhar os conteúdos mais discutidos no momento. Esse espaço facilita o engajamento com os tópicos mais relevantes, permitindo ao professor interagir com as publicações mais populares e contribuir para os debates em curso.

Figura 1 - Página Mapa do professor versão web



A figura () apresenta a página Mapas, uma ferramenta interativa que permite ao professor explorar as áreas geográficas onde plantas, frutas e animais foram registrados no sistema. Através de um mapa detalhado, o docente pode acessar as localizações exatas desses registros, promovendo um aprendizado mais prático e informativo sobre as espécies. A página também inclui cards educativos, que oferecem informações sobre as plantas, frutas e animais registrados, como suas características e habitats, possibilitando um aprofundamento no conteúdo de maneira visual e interativa.

### 6.2. Implementação (Sprint 2)

Neste item, pode-se usar um framework de front-end. Na hipótese de uso de um framework, sugiro algum que o corpo docente domine e/ou que esteja sendo utilizado no semestre atual ou passado: <https://vuejs.org/> etc.

## 7. Back-end (Sprint 2 e 3)

### 7.1. Diagrama de classes (Sprint 2)

Incluir classe Data Access Object (DAO) para servir como interface com o banco de dados para a execução de comandos DQL e DML.

### 7.2. Implementação do back-end (Sprint 3)

Neste item, pode-se usar um framework de back-end com ou sem ORM. Na hipótese de uso de um framework, sugiro algum que o corpo docente domine e/ou que esteja sendo utilizado no semestre atual ou passado: <https://flask.palletsprojects.com/>, <https://www.djangoproject.com/>, <https://spring.io/> etc.

#### 7.2.1 Suporte a estatística (Sprint 3)

Nessa parte, podem ser implementados recursos para apoiar a realização dos testes do produto com possíveis usuários, coletar dados como número de acessos, partes do produto mais acessadas, tempo de acesso, entre outras. Montar um questionário pós teste do aplicativo, com perguntas para melhoria do produto de acordo com o perfil dos usuários. Utilizar o Python para cálculo das medidas estatísticas, principalmente média e desvio padrão. Realizar testes de comparação estatísticos através dos testes estatísticos estudados pela turma. Isso também poderá ser realizado com o Python. Exemplo: [https://lume.ufrgs.br/bitstream/handle/10183/4801/000504797.pdf&sa=D&source=docs&ust=1714068681711844&usg=AOvVaw1M8DB5KAkNftE4rmDG24VA](https://lume.ufrgs.br/bitstream/handle/10183/4801/000504797.pdf)

## 8. Integração front-end e back-end (Sprint 4)

Via *Application Programming Interface* (API), incluindo hospedagem do front-end e back-end.

## 9. Referências

[1] Pfleeger, S. L. **Engenharia de software: teoria e prática**. Editora Pearson, 2003. E-book. ISBN: 9788587918314. Disponível em: <https://plataforma.bvirtual.com.br/Leitor/Publicacao/476/epub/0>. Acesso em: 22 out. 2022.

[2] DENNIS, Alan; WIXOM, Barbara H.; ROTH, Roberta M. **Análise e Projeto de Sistemas**. Grupo GEN, 2014. E-book. ISBN 978-85-216-2634-3. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/978-85-216-2634-3/>. Acesso em: 22 out. 2022.

[3] BAZZI, Cláudio L. **Introdução a banco de dados**. Curitiba: Ed. UTFPR, 2013. e-ISBN: 978-85-7014-114-9. Disponível em: <http://proedu.rnp.br/handle/123456789/1550>. Acesso em: 03 mai. 2023.

[4] KALBACH, James. Design de navegação web. Porto Alegre: Grupo A, 2009. E-book. ISBN 9788577805310. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788577805310/>. Acesso em: 07 mar. 2024.

[5] https://www.portalinsights.com.br/perguntas-frequentes/como-fazer-uma-pesquisa-estatistica-passo-a-passo

[6] Python para estatísticos. Disponível em: https://tmfilho.github.io/pyestbook/intro.html. Acesso em: abril de 2024

[7] ROSA, Paulo Henrique C.. Desenvolvimento de Software tipo aplicativo de dispositivo móvel para auxílio em abordagem estatística na área de saúde. Dissertação de Mestrado. São Paulo: Universidade Brasil, 2020. Disponível em: https://universidadebrasil.edu.br/portal/\_biblioteca/uploads/20210416143305.pdf

SOMERVILLE, Ian. *Engenharia de Software*. 9. ed. São Paulo: Pearson Prentice Hall, 2011.