**Projeto Inicial – Gerenciamento de Patrimônio Escolar**

Documentação

Desenvolvido por:

Enzzo Lima

Erika Leite

Gabriel Pereira

Joyce Santos

Sumário

[1. Resumo 3](#_Toc79392597)

[2. Escopo do projeto 3](#_Toc79392598)

[2.1. Sistema Web 3](#_Toc79392599)

[2.1.1. Perfis de Usuário 3](#_Toc79392600)

[2.1.2. Funcionalidades 3](#_Toc79392601)

[3. Modelagem de software 4](#_Toc79392602)

[3.1. Modelo Conceitual 4](#_Toc79392603)

[3.2. Modelo Lógico 4](#_Toc79392604)

[3.3. Modelo Físico 5](#_Toc79392605)

[3.4. Cronograma 5](#_Toc79392606)

[4. Back-End 6](#_Toc79392607)

[4.1. Como executar e testar a API? 6](#_Toc79392608)

[5. UI / UX 7](#_Toc79392609)

[5.1.1. Imagens do layout 7](#_Toc79392610)

[6. Front-end 8](#_Toc79392611)

[6.1.1. Imagens da aplicação na web 8](#_Toc79392612)

# Resumo

Uma nova escola deseja criar um sistema de gerenciamento de seu patrimônio, para facilitar a gestão de suas salas e equipamentos.

# Escopo do projeto

Disponível em < https://miro.com/app/board/o9J\_l\_OcB\_Q=/>.

A aplicação possui os seguintes requisitos:

## Sistema Web

### Perfis de Usuário

1. Administrador: Para o colaborador da área administrativa da escola;
2. Comum: Para colaboradores de almoxarifado realizarem o cadastro de novos equipamentos e salas.

### Funcionalidades

1. O administrador poderá cadastrar qualquer tipo de usuário (administrador ou comum);
2. O administrador poderá cadastrar novos usuários à plataforma e realizar a gestão deles;
3. O administrador poderá excluir salas e equipamentos;
4. O administrador poderá atualizar dados de equipamentos, salas e usuários;
5. O usuário comum poderá inserir novos equipamentos ao sistema;
6. O administrador poderá gerar relatórios de controle;

# Modelagem de software

O banco de dados relacional é uma coleção de objetos, ou seja, são tabelas onde os dados são armazenados. Como exemplo, podemos querer armazenar dados sobre os produtos de um comércio. Para isso, criamos tabelas que guardam diferentes conjuntos de dados relacionados, como dados de fornecedores, dados de compras, dados de saída etc. Cada uma dessas tabelas é uma relação do banco.

De modo geral, a utilização de um banco de dados gera uma melhora na organização, facilita a gestão e o processamento dos dados. Isso acontece porque os dados são organizados de modo a modelar aspectos do mundo real, gerando informações relevantes para os usuários a partir desses dados.

## Modelo Conceitual

O modelo conceitual é um conjunto de suposições baseadas no mundo real que indicarão as regas de negócio de um sistema. Portanto, modelo conceitual é a descrição do sistema proposto na forma de um conjunto de ideias.

Diagrama

Descrição gerada automaticamente

## Modelo Lógico

Este modelo adequa o modelo conceitual para o tipo de banco de dados que será implementado. É desenvolvido na fase de projeto e não é de interesse dos usuários, somente da equipe de desenvolvimento. Nesse modelo que os atributos e as relações entre ficam mais evidentes.

Diagrama, Esquemático

Descrição gerada automaticamente

## Modelo Físico

Este modelo é o projeto físico para implementação do banco de dados. Tem todas as preocupações com desempenho, com a capacidade dos equipamentos etc. Pode-se utilizar o Excel para simular a estrutura real do banco de dados com base no modelo lógico elaborado anteriormente.

Interface gráfica do usuário, Aplicativo, Tabela

Descrição gerada automaticamente

## Cronograma

Todo o cronograma foi desenvolvido utilizando a metodologia Kanban e organizado pela equipe através do Trello.

Link do Trello: < https://trello.com/b/GQBGFsQN/projeto-inicial-3-termo>.

# Back-End

O Back-End da aplicação, ou seja, o código do sistema, foi desenvolvido no Microsoft Visual Studio 2019 na linguagem C#. É uma API (Application Programming Interface) com arquitetura REST (Representational State Transfer), que é um conjunto de normas que possibilita a comunicação entre plataformas através de uma série de padrões e protocolos. As requisições e respostas seguem o protocolo HTTP (HyperText Transfer Protocol) utilizando formato JSON (JavaScript Object Notation) para intercâmbio de dados, possibilitando o acesso do sistema em diversos dispositivos.

## Como executar e testar a API?

São necessários os seguintes programas:

* Microsoft SQL Server;
* Microsoft SQL Management Studio;
* Microsoft Visual Studio;
* Postman.

# UI / UX

O Layout da aplicação, tanto para web quanto para dispositivos móveis, foi desenvolvido no Figma. Para a criação do layout da aplicação foram feitos wireframes, tanto de baixa fidelidade quanto de alta fidelidade, e prototipação de todas as páginas, para saber se a interface e a experiência do usuário tivessem qualidade.

### Imagens do layout

*Web Interface gráfica do usuário, Aplicativo, Site

Descrição gerada automaticamente*

# Front-end

O Front-end da aplicação, ou seja, toda a interface utilizada pelo usuário final, foi desenvolvido no Microsoft Visual Studio Code 2019 utilizando a biblioteca React.

### Imagens da aplicação na web

Página de Login

Interface gráfica do usuário

Descrição gerada automaticamente

Home do sistema

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

Páginas de Cadastros e Edições

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

Tela de celular com publicação numa rede social

Descrição gerada automaticamente

Interface gráfica do usuário, Aplicativo, Teams

Descrição gerada automaticamente