**– DOCUMENTAÇÃO –**

**-Carlos Eduardo F. de Paiva**

**-Flavio Barto Machado**

**-Matheus Gabriel Tavares**

**-João Rafael Paton**

Introdução sobre o projeto:

(30/03/2025)

Atualmente, o laboratório da faculdade enfrenta dificuldades na gestão de cadastros de pacientes e na emissão de laudos. O processo atual não é eficiente, o que pode gerar atrasos, dificuldades na organização das informações e problemas na recuperação de registros.

Dessa forma, é necessário um sistema moderno e otimizado que permita:

* O **cadastro eficiente de pacientes**, garantindo a organização e acessibilidade dos dados.
* A **geração e impressão de laudos** de forma ágil e padronizada, reduzindo erros e melhorando o fluxo de trabalho no laboratório.

Esse sistema visa melhorar a eficiência operacional do laboratório, proporcionando maior praticidade e segurança no armazenamento e recuperação das informações dos pacientes e seus respectivos laudos.

Objetivos do projeto:

(30/03/2025)

O objetivo deste projeto é desenvolver um sistema eficiente para a gestão do laboratório da faculdade, garantindo maior organização, acessibilidade e automação no processamento de informações. Para isso, o sistema deve:

1. **Facilitar o cadastro de pacientes** – Criar uma interface intuitiva para registrar e consultar pacientes de forma rápida e organizada.
2. **Automatizar a geração e impressão de laudos** – Permitir a criação de laudos de maneira padronizada, reduzindo erros e agilizando o processo.
3. **Garantir a segurança e integridade dos dados** – Implementar mecanismos para proteger as informações dos pacientes, evitando perdas e acessos não autorizados.
4. **Melhorar a acessibilidade da informação** – Permitir que os profissionais do laboratório encontrem rapidamente os cadastros e laudos necessários.
5. **Otimizar o fluxo de trabalho** – Reduzir a burocracia e tempo gasto com processos manuais, aumentando a eficiência geral do laboratório.

Tecnologias utilizadas no projeto:

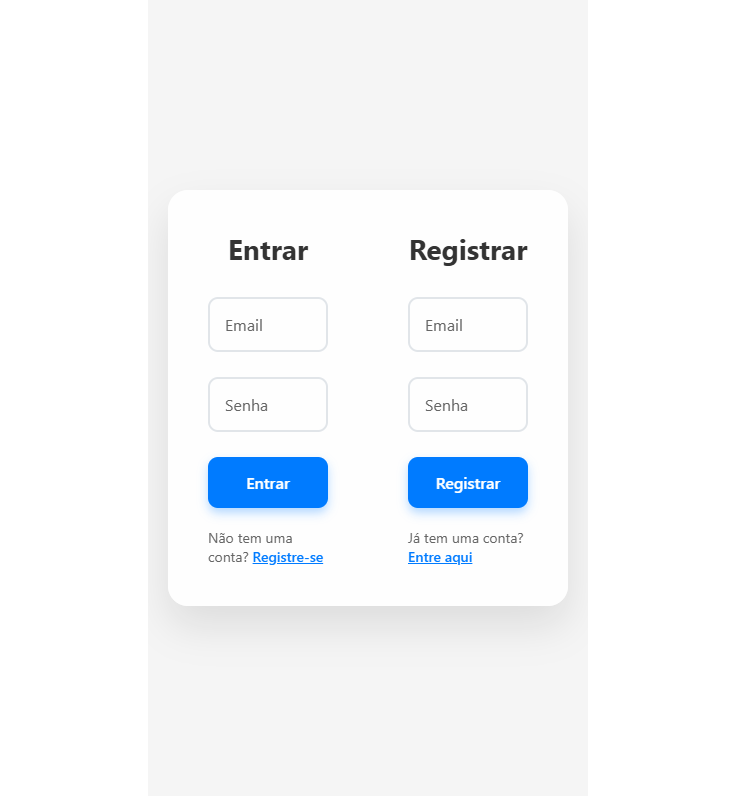
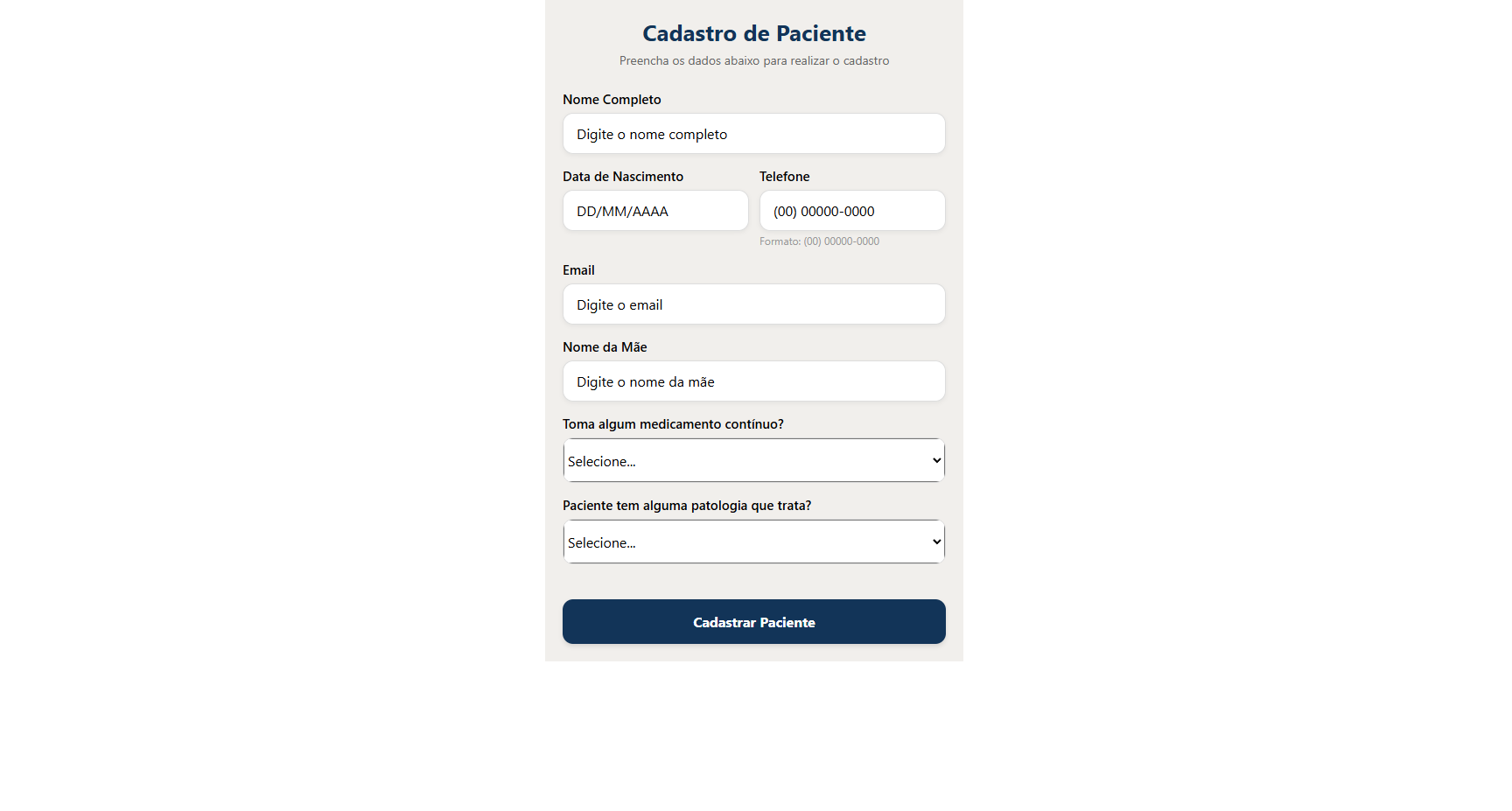
* Node.js
* PHP
* React Native
* MySQL
* Python

Reuniões e definições feitas:

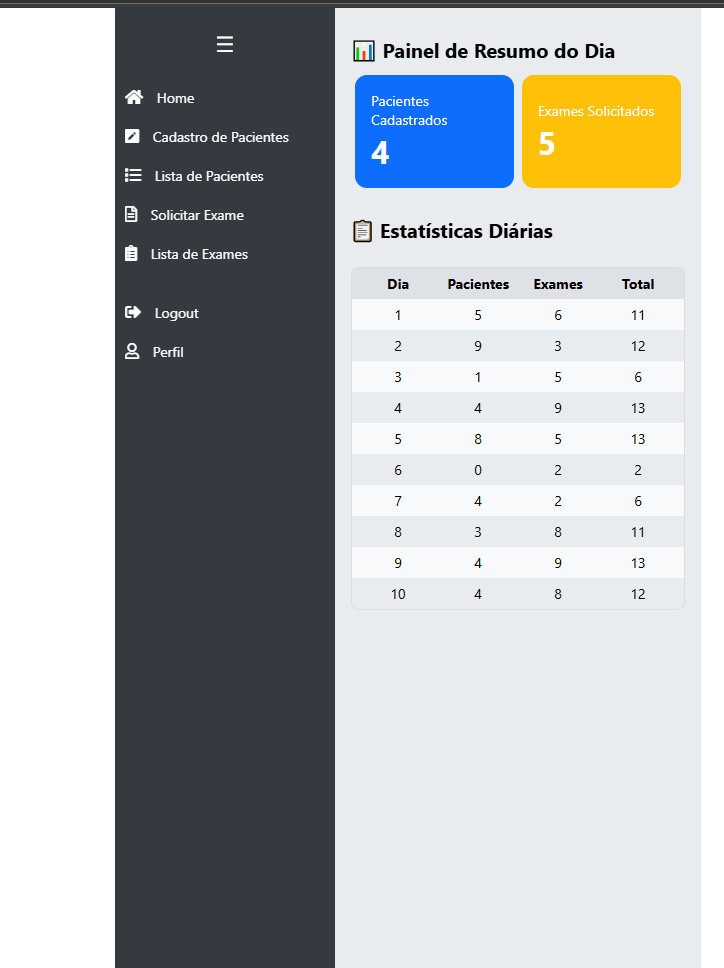
* Reunião Inicial - Definição de Necessidades (25/03/25)
  + Definido o que é necessário e as necessidades mais urgentes.
  + Um sistema no qual permite o funcionamento interino do laboratório tanto de farmácia quanto de Biomedicina.
  + Impressão de laudos
  + Cadastro de clientes com conexões com o Banco de Dados
  + Adicionar informações sobre o paciente de maneira mais eficaz e rápida.
* 2° Encontro(31/03/25)
  + Mostrado os primeiros resultados de telas simples.
  + Mostrado e aprovado a documentação.
  + Recebido o formulário no qual era feito até então
  + Discutidos temas nos quais precisavam ser adicionados e quais eram relevantes
  + Discutidos meios de melhorar o atual formulário, com toques adicionais como: Alunos não podem ter informações de nomes.
  + Modelos bases de laudos.
* 3° encontro (14/04/25)
  + Mostrado pessoalmente o local de trabalho e como vai funcionar o projeto
  + mostrado a nossa parte do projeto, e qual caminho ele está tomando.

|  | | | |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Telas :

Login  
  
Cadastro de clientes

Home



Levantamento de Entidades e Atributos

Entidades são objetos ou conceitos que possuem dados importantes para o sistema. A partir do banco de dados criado, podemos identificar as seguintes entidades e seus atributos:

* Paciente
  + idPaciente (VARCHAR 8): Identificador único do paciente (chave primária).
  + dataCadastro (TIMESTAMP): Data e hora de cadastro do paciente no sistema.
  + nomeCompleto (VARCHAR 100): Nome completo do paciente.
  + dataNascimento (DATE): Data de nascimento do paciente.
  + idade (INT): Idade do paciente.
  + telefone (VARCHAR 20): Número de telefone do paciente.
  + email (VARCHAR 100): Endereço de e-mail do paciente.
  + nomeMae (VARCHAR 100): Nome da mãe do paciente.
  + nomeMedicamento (VARCHAR 100): Medicamento que o paciente utiliza (caso aplicável).
  + nomePatologia (VARCHAR 100): Patologia que o paciente tem (caso aplicável).
* Exame
  + idExame (INT): Identificador único do exame (chave primária).
  + idPaciente (VARCHAR 8): Relacionamento com o paciente que fez o exame (chave estrangeira).
  + laboratorio (VARCHAR 100): Nome do laboratório que realizou o exame.
  + exameTexto (TEXT): Descrição do exame realizado.
  + dataExame (TIMESTAMP): Data e hora em que o exame foi realizado.
  + resultado (TEXT): Resultado do exame realizado.

Requisitos Funcionais

São as funções que o sistema *precisa realizar*. Exemplos: cadastrar paciente, consultar exames, gerar relatórios.

* Cadastro de Pacientes: O sistema deve permitir o cadastro de novos pacientes, com campos para nome completo, data de nascimento, telefone, e-mail, entre outros.
* Atualização de Dados do Paciente: O sistema deve permitir a atualização dos dados dos pacientes cadastrados (ex: nome, telefone, e-mail).
* Consulta de Pacientes: O sistema deve permitir a consulta dos pacientes cadastrados com filtros como nome, telefone e e-mail.
* Cadastro de Exames: O sistema deve permitir o cadastro de exames realizados pelos pacientes, com dados como laboratório, texto do exame e resultado.
* Consulta de Exames: O sistema deve permitir a consulta dos exames realizados pelos pacientes, com possibilidade de filtrar por data, paciente e laboratório.
* Relatórios: O sistema deve permitir a geração de relatórios para visualizar os pacientes e exames, com informações como quantidade de exames realizados por paciente, lista de exames pendentes, etc.

### **Requisitos Não Funcionais**

São as qualidades que o sistema deve ter ou como ele deve funcionar. Exemplos: ser rápido, ser seguro, ter backup, seguir a LGPD(Lei Geral de Proteção de Dados Pessoais (Lei nº 13.709/2018)).

* implementando controle de acesso baseado em permissões de usuário.
* Necessário um servidor para ser utilizado.
* Necessário um meio de acesso à internet.
* Necessário Backup.

### **DER ( DIAGRAMA ENTIDADE-RELACIONAMENTO)**

### 