



MedStock Sprint 2

Integrantes:

Felipe Rivetti Mizher

Pedro Hosken Fernandes Guimarães

Rafael Rehfeld Martins de Oliveira

Vitor Augusto Viana Azevedo

Quais são as funções que o sistema deve executar e Como o sistema deve executar tais funções?

Requisitos funcionais (RF):		
Código	Requisito	Descrição
RF 01	Cadastrar Remédio	Cada remédio deve ser cadastrado, com o seu id, nome, preço, estoque
RF 02	Cadastrar Compra Médico	Quantidade do pedido, data de validade do lote e atualizar estoque do remédio
RF 03	Cadastrar Compra Paciente	Quantidade do pedido e atualizar estoque do remédio
RF 04	Notificar Médico de falta de Estoque	Gerar uma mensagem inteligente para quando o remédio estiver acabando
RF 05	Notificar Médico de vencimento de Lote	Gerar uma mensagem inteligente para quando o lote estiver vencendo
RF 06	Login Médico	Na aba de Login do médico colocar seu CRM e sua senha
RF 07	Cadastro Médico	Na aba de fazer cadastro, colocar nome, email, data de nascimento, <u>crm</u> e senha
RF 08	Alterar senha	Possibilitar alteração de senha caso tenha esquecido ou apenas queira alternar
RF 09	Alterar <u>email</u>	Possibilidade de alternar <u>email</u>
RF 10	Consultar Estoque	O sistema deve permitir que o médico consulte o estoque de um medicamento a qualquer momento
RF 11	Gerar Relatórios de Consumo	O sistema deve permitir a geração de relatórios mensais com os remédios mais consumidos
RF 12	Histórico de Compras	Histórico de Compras: O sistema deve armazenar um histórico das compras realizadas pelo médico ou paciente

Quais são as funções que o sistema deve executar e Como o sistema deve executar tais funções?

Regras de negócio (RF):

Código	Descrição
RN 01	O médico só pode realizar compras se estiver com CRM válido
RN 02	Não permitir a compra de remédios, por pacientes, que estão com o estoque zerado
RN 03	A primeira notificação de vencimento do lote deve ser gerada, no mínimo, 30 dias antes da data de expiração do lote.
RN 04	O sistema só deve permitir a compra de remédios que ainda estejam dentro do prazo de validade.
RN 05	A compra de pacientes deve utilizar uma estrutura FIFO, retirando remédios do lote mais antigo disponível.

Requisitos não funcionais (RNF):

Código	Categoria	Descrição
RNF 01	Disponibilidade	O sistema deve estar disponível 24/7 para consulta e notificações
RNF 02	Segurança	Os dados do médico e das compras devem ser criptografados e protegidos contra acessos não autorizados
RNF 03	Tempo de Resposta	O sistema deve processar todas as solicitações (cadastro de remédio, compra, notificações) em até 2 segundos.

Diagramas



Diagrama Entidade- Relacionamento (DER)

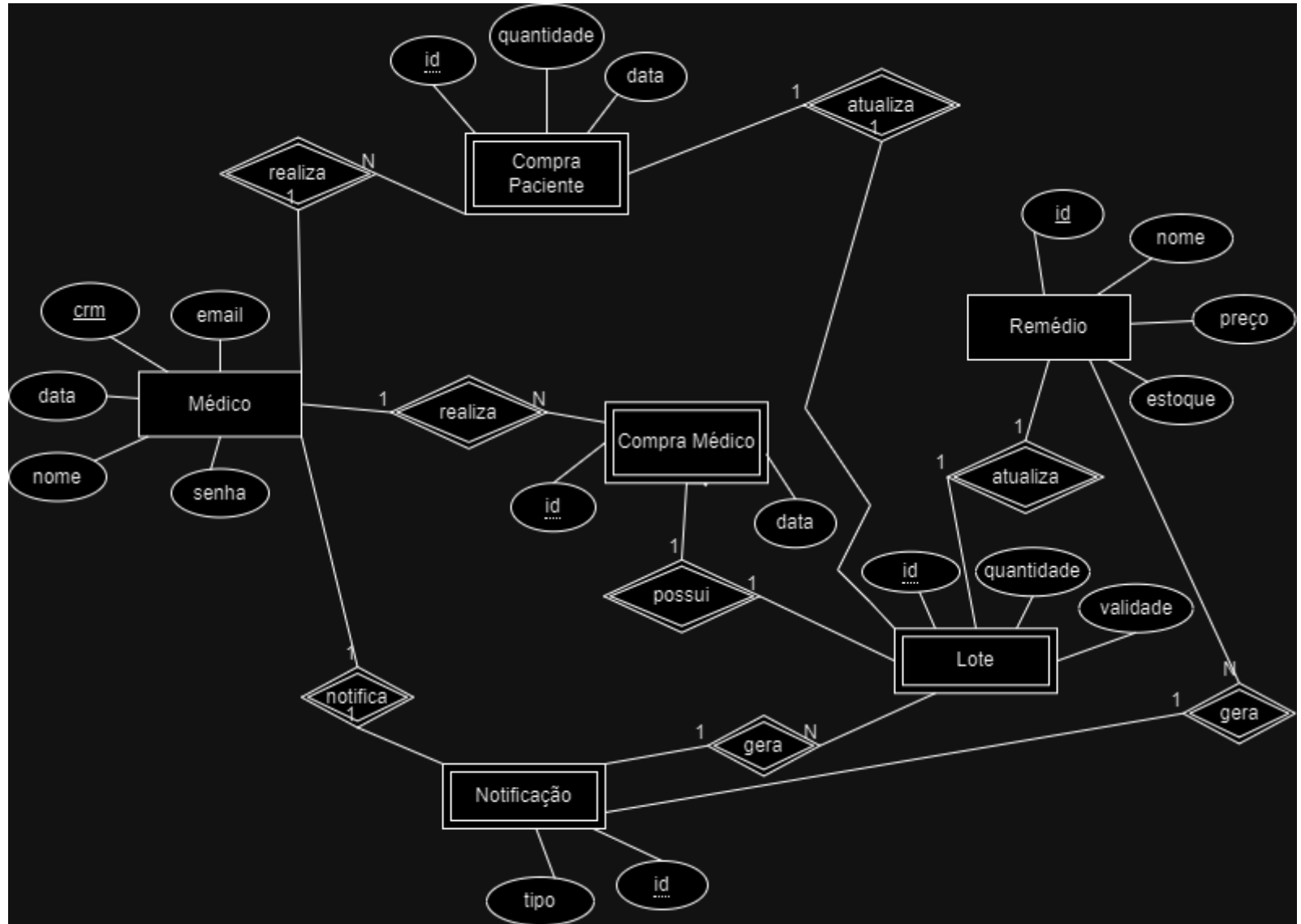


Diagrama de Esquema (DSC)

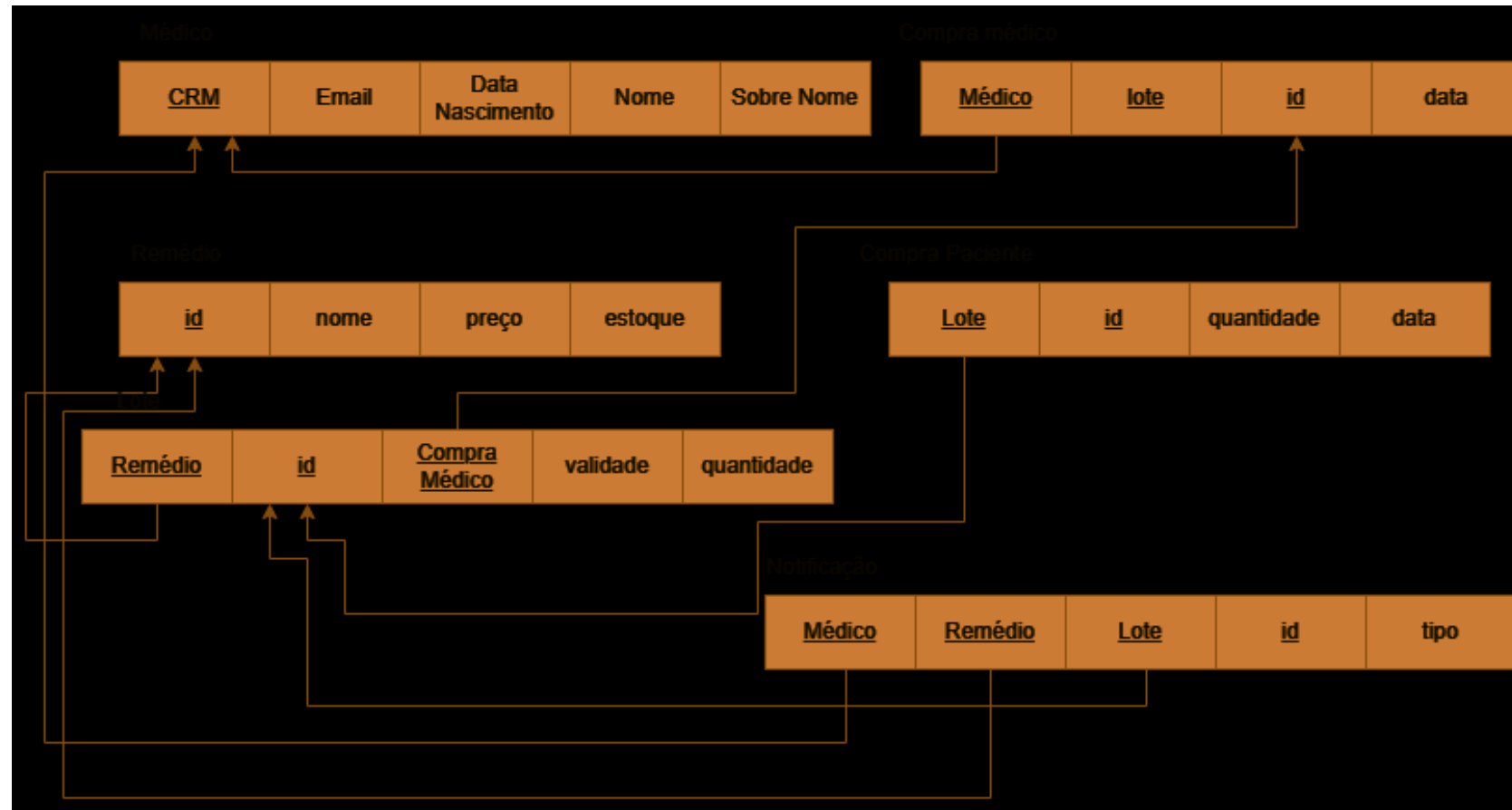
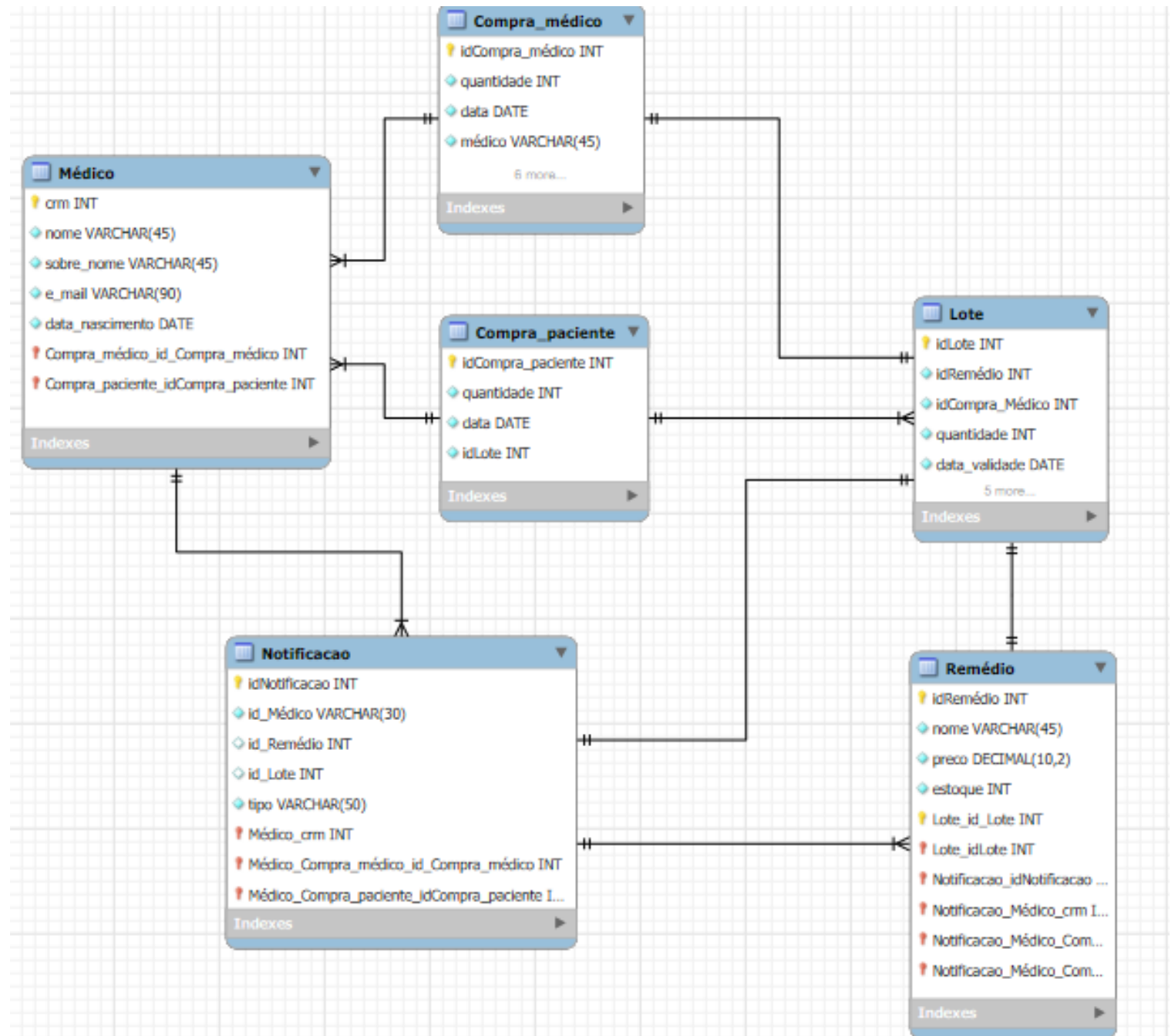


Diagrama Entidade- Relacionamento Estendido em notação Pé de Galinha




Script da criação do banco de dados

```
1  -- MySQL Script generated by MySQL Workbench
2  -- Thu Sep 26 17:05:44 2024
3  -- Model: New Model    Version: 1.0
4  -- MySQL Workbench Forward Engineering
5
6  • SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
7  • SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
8  • SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,
9      NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION';
10
11  -----
12  -- Schema mydb
13  -----
14
15  -----
16  -- Table `Notificacao`
17  -----
18  • DROP TABLE IF EXISTS `Notificacao` ;
19
20  • CREATE TABLE IF NOT EXISTS `Notificacao` (
21      `idNotificacao` INT NOT NULL AUTO_INCREMENT,
22      `id_Médico` VARCHAR(30) NOT NULL,
23      `id_Remédio` INT NULL,
24      `id_Lote` INT NULL,
25      `tipo` VARCHAR(50) NOT NULL,
26      `Médico_crm` INT NOT NULL,
```

```
27      `Médico_Compra_médico_id_Compra_médico` INT NOT NULL,
28      `Médico_Compra_paciente_idCompra_paciente` INT NOT NULL,
29      PRIMARY KEY (`idNotificacao`, `Médico_crm`, `Médico_Compra_médico_id_Compra_médico`,
30          `Médico_Compra_paciente_idCompra_paciente`),
31      CONSTRAINT `fk_Notificacao_Médico1`
32          FOREIGN KEY (`Médico_crm`, `Médico_Compra_médico_id_Compra_médico`, `Médico_Compra_paciente_idCompra_paciente`)
33          REFERENCES `Médico` (`crm`, `Compra_médico_id_Compra_médico`, `Compra_paciente_idCompra_paciente`)
34          ON DELETE NO ACTION
35          ON UPDATE NO ACTION)
36      ENGINE = InnoDB;
37
38  ❌ CREATE UNIQUE INDEX `idNotificacao_UNIQUE` ON `Notificacao` (`idNotificacao` ASC) VISIBLE;
39
40  • CREATE INDEX `fk_Notificacao_Médico1_idx` ON `Notificacao` (`Médico_crm` ASC, `Médico_Compra_médico_id_Compra_médico` ASC,
41      `Médico_Compra_paciente_idCompra_paciente` ASC) VISIBLE;
42
43  -----
44  -- Table `Compra_paciente`
45  -----
46
47  • DROP TABLE IF EXISTS `Compra_paciente` ;
```


Tela de Login



Login

E-mail:

Senha:

Entrar

Não tem uma conta? [Cadastre-se aqui.](#)

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Login</title>
  <link rel="stylesheet" href="/Cadastro/cadastro.css">
</head>
<body>
  <div class="container">
    
    <h2>Login</h2>
    <form action="#" method="post">
      <div class="form-group">
        <label for="email">E-mail:</label>
        <input type="email" id="email" name="email" required>
      </div>
      <div class="form-group">
        <label for="senha">Senha:</label>
        <input type="password" id="senha" name="senha" required>
      </div>
      <div class="form-group">
        <button type="submit">Entrar</button>
      </div>
      <div class="form-group">
        <p>Não tem uma conta? <a href="../Cadastro/PaginaCadastro.html">Cadastre-se aqui</a>.</p>
      </div>
    </form>
  </div>
</body>
</html>
```

Tela de cadastro

```
<div class="form-group">
  <label for="nome">Nome Completo:</label>
  <input type="text" id="nome" name="nome" required>
</div>

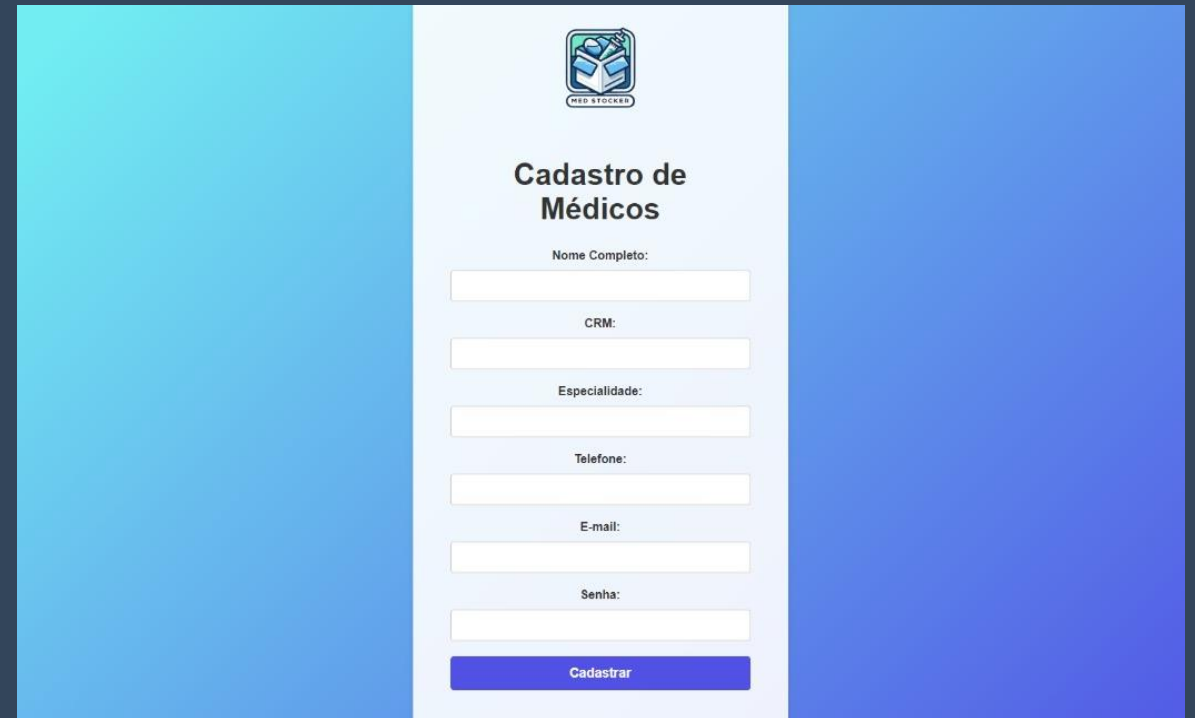
<div class="form-group">
  <label for="crm">CRM:</label>
  <input type="text" id="crm" name="crm" required>
</div>

<div class="form-group">
  <label for="especialidade">Especialidade:</label>
  <input type="text" id="especialidade" name="especialidade" required>
</div>

<div class="form-group">
  <label for="telefone">Telefone:</label>
  <input type="tel" id="telefone" name="telefone" required>
</div>

<div class="form-group">
  <label for="email">E-mail:</label>
  <input type="email" id="email" name="email" required>
</div>

<div class="form-group">
  <label for="senha">Senha:</label>
```

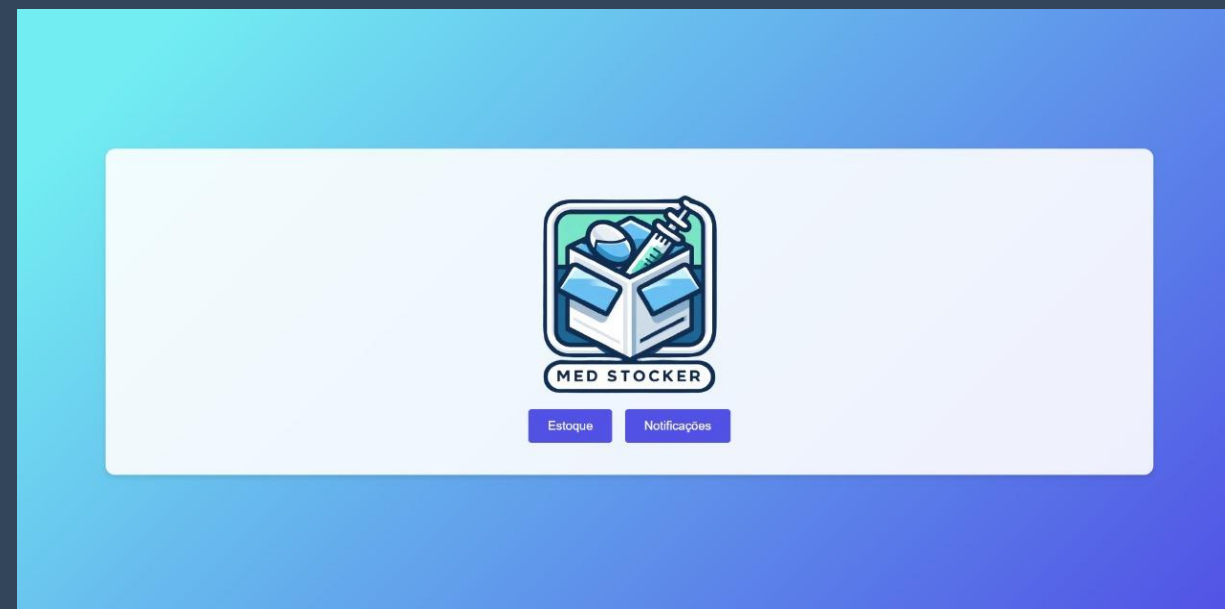


The screenshot shows a web registration form titled "Cadastro de Médicos". At the top, there is a logo for "MED STOCKER" featuring a stylized book and a heart. Below the logo, the title "Cadastro de Médicos" is displayed in a bold, black font. The form consists of several input fields, each preceded by a label: "Nome Completo:", "CRM:", "Especialidade:", "Telefone:", "E-mail:", and "Senha:". Each input field is a simple text box with a light gray border. At the bottom of the form, there is a blue button labeled "Cadastrar". The form is centered on a white background, which is flanked by two vertical blue bars on either side.

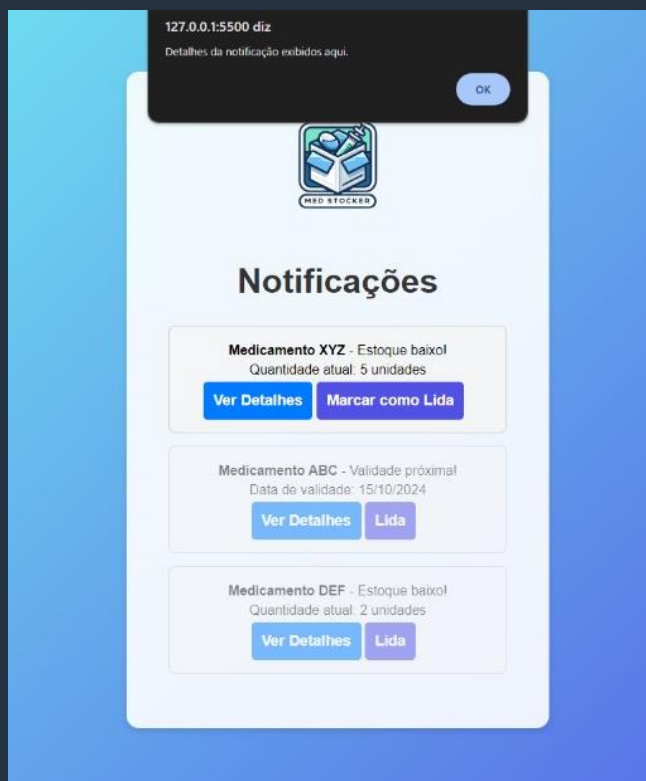
Front-end da aplicação

Tela da Home

```
1 <!DOCTYPE html>
2 <html lang="pt-BR">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Med Stoker</title>
7   <link rel="stylesheet" href="home.css">
8 </head>
9 <body>
10  <div class="container">
11    
12    <div class="buttons">
13      <a href="/Remedios/remedios.html" class="button">Estoque</a>
14      <a href="/Notificacoes/notificacoes.html" class="button">Notificações</a>
15    </div>
16  </div>
17 </body>
18 </html>
```



Tela de Notificações



```
<div class="notificacoes">
  <div class="notificacao">
    <p><strong>Medicamento XYZ</strong> - Estoque baixo!</p>
    <p>Quantidade atual: 5 unidades</p>
    <button class="detalhes">Ver Detalhes</button>
    <button class="marcar-lida">Marcar como Lida</button>
  </div>

  <div class="notificacao">
    <p><strong>Medicamento ABC</strong> - Validade próxima!</p>
    <p>Data de validade: 15/10/2024</p>
    <button class="detalhes">Ver Detalhes</button>
    <button class="marcar-lida">Marcar como Lida</button>
  </div>

  <div class="notificacao">
    <p><strong>Medicamento DEF</strong> - Estoque baixo!</p>
    <p>Quantidade atual: 2 unidades</p>
    <button class="detalhes">Ver Detalhes</button>
    <button class="marcar-lida">Marcar como Lida</button>
  </div>
</div>

<script>
  // Adicionando funcionalidades aos botões
  const marcarLidaButtons = document.querySelectorAll('.marcar-lida');
  marcarLidaButtons.forEach(button => {
    button.addEventListener('click', () => {
      const notificacao = button.parentElement;
      notificacao.style.opacity = '0.5'; // Simulando que a notificação foi lida
      button.textContent = 'Lida'; // Atualiza o texto do botão
      button.disabled = true; // Desabilita o botão
    });
  });

  const detalhesButtons = document.querySelectorAll('.detalhes');
  detalhesButtons.forEach(button => {
    button.addEventListener('click', () => {
      alert('Detalhes da notificação exibidos aqui.');// Simulação de detalhes
    });
  });
});
```

Tela do estoque

```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Estoque de Remédios</title>
7   <link rel="stylesheet" href="/Remedios/remedios.css"> <!-- Ajuste o caminho se necessário -->
8 </head>
9 <body>
10   <div class="container">
11     <h2>Estoque de Remédios</h2>
12     <div class="produtos">
13       <div class="produto">
14          <!-- Substitua pelo caminho da imagem do remédio -->
15         <h3>Remédio XYZ</h3>
16         <p>Preço: R$ 15,00</p>
17         <button class="ver-detalhes" onclick="window.location.href='remedio.html?nome=Remédio%20XYZ'">Ver Detalhes</button>
18       </div>
19
20       <div class="produto">
21          <!-- Substitua pelo caminho da imagem do remédio -->
22         <h3>Remédio ABC</h3>
23         <p>Preço: R$ 20,00</p>
24         <button class="ver-detalhes" onclick="window.location.href='remedio.html?nome=Remédio%20ABC'">Ver Detalhes</button>
25       </div>
26
27       <div class="produto">
28          <!-- Substitua pelo caminho da imagem do remédio -->
29         <h3>Remédio DEF</h3>
30         <p>Preço: R$ 12,00</p>
31         <button class="ver-detalhes" onclick="window.location.href='remedio.html?nome=Remédio%20DEF'">Ver Detalhes</button>
32       </div>
33
34       <!-- Adicione mais produtos conforme necessário -->
35     </div>
36   </div>
37 </body>
38 </html>
```

Estoque de Remédios



Remédio XYZ
Preço: R\$ 15,00
[Ver Detalhes](#)



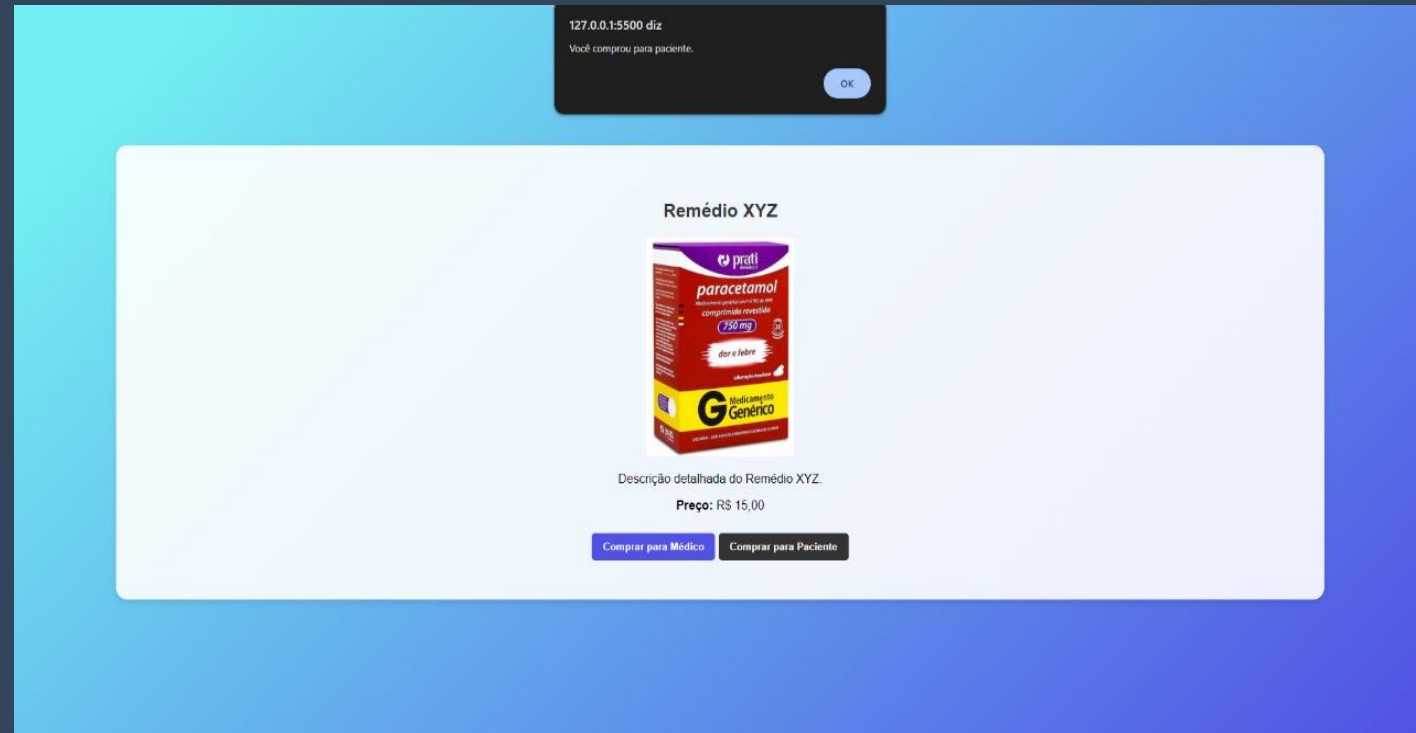
Remédio ABC
Preço: R\$ 20,00
[Ver Detalhes](#)



Remédio DEF
Preço: R\$ 12,00
[Ver Detalhes](#)

Tela do Remédio

```
45 // Função para exibir informações do remédio
46
47 function displayRemedioInfo() {
48     const nome = getQueryParam('nome');
49     const remedio = remedios[nome];
50
51     if (remedio) {
52         document.getElementById('remedio-nome').innerText = nome;
53         document.getElementById('remedio-descricao').innerText = remedio.descricao;
54         document.getElementById('remedio-preco').innerText = remedio.preco;
55         document.getElementById('remedio-imagem').src = remedio.imagem;
56     } else {
57         document.body.innerHTML = '<h2>Remédio não encontrado!</h2>';
58     }
59 }
60
61 // Função para simular a compra
62 function comprar(tipo) {
63     alert(`Você comprou para ${tipo}.`);
64 }
65
```



Obrigado pela
atenção!