

# Sistema da Terapeuta

## Objetivo

Aplicativo web fullstack (Node.js + MySQL) para gerenciamento de **pacientes**, **agendamentos**, **atendimentos** e **testes energéticos** realizados por uma terapeuta.

---

## Estrutura do Projeto

```
sistema-Terapeuta/
|
├── backend/           # Backend Node.js com Express e MySQL
|   ├── app.js         # App principal com middlewares e rotas
|   ├── server.js      # Inicializa o servidor na porta definida
|   ├── config/
|   │   └── db.js       # Conexão com o banco MySQL usando mysql2
|   ├── controllers/
|   │   └── pacienteController.js # Funções de controle de pacientes
|   ├── models/
|   │   └── pacienteModel.js    # Lógica de acesso ao banco
|   ├── routes/
|   │   └── pacienteRoutes.js   # Rotas da API para /api
|   ├── docker/
|   │   └── mysql-init/init.sql  # Script SQL para criação do banco
|   ├── .env              # Variáveis de ambiente
|   └── package.json      # Dependências e scripts do projeto
|
├── frontend/           # Frontend HTML/CSS/JS (estático)
|   └── pages/           # Páginas da aplicação (login, dashboard, etc.)
|
├── docker-compose.yml   # Orquestração de app e MySQL
├── Dockerfile           # Imagem do backend
└── README.md           # Documentação principal
```

## Como rodar o projeto com Docker

### Requisitos

- Docker
- Docker Compose

## Comando:

```
docker-compose up --build
```

- Backend na porta **3000**
- MySQL na porta **3306**

---

## API - Endpoints

**Base:** `/api`

**GET** `/api/`

Lista todos os pacientes cadastrados.

**POST** `/api/`

Cria um novo paciente.

```
{
  "nome": "João da Silva",
  "email": "joao@email.com",
  "telefone": "11999999999",
  "data_nascimento": "1990-05-12",
  "historico": "Paciente com histórico de dores crônicas."
}
```

---

## Backend Detalhado

`server.js`

Inicia o servidor com porta definida no `.env`.

`app.js`

Registra middlewares e rotas com Express e CORS.

`pacienteModel.js`

Acesso ao banco de dados MySQL.

- `getTodosPacientes()`
- `adicionarPaciente(paciente)`

pacienteController.js

Controladores de requisições HTTP:

- listarPacientes
- criarPaciente



## Banco de Dados (init.sql)

Tabelas:

- pacientes
- agendamentos
- atendimentos
- testes\_energeticos

Exemplo:

```
CREATE TABLE pacientes (  
  id INT AUTO_INCREMENT PRIMARY KEY,  
  nome VARCHAR(100),  
  email VARCHAR(100),  
  telefone VARCHAR(20),  
  data_nascimento DATE,  
  historico TEXT,  
  criado_em TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```



## Dependências

Produção:

- express
- cors
- dotenv
- mysql2

Desenvolvimento:

- nodemon
-

## 😊 Variáveis de Ambiente ( `.env` )

```
PORT=3000
DB_HOST=db
DB_USER=root
DB_PASSWORD=root
DB_NAME=terapia_system
```

---

## 💕 docker-compose.yml

Define os serviços:

- **app**: Container Node.js
  - **db**: Container MySQL com volume e script de inicialização
- 

## ✍️ Autor

- **Leandro Marques**
- Contato: [Instagram: @marquestechh](#)