

# SISTEMA DE GERENCIAMENTO ODONTOLÓGICO

Gerenciador de Clínicas Odontológicas





# Objetivo:

- Cadastro detalhado de clientes
- Registro de procedimentos odontológicos
- Controle de despesas
- Controle de estoque
- Gerenciamento das formas de pagamento
- Emissão de relatórios financeiros
- Acompanhamento de avaliações pós-tratamento

# Cliente:

- Clinico odontológico de Quixada - CE
- Informações reunidas através de reuniões com o cliente.

# Requisitos:

- Funcionais
- Não funcionais

RF001: Permitir o cadastro detalhado de pacientes, incluindo informações pessoais, histórico medico e contatos.

RNF001: O sistema deve ter uma interface intuitiva e de fácil utilização para todos os usuários.

RF002: Permitir o registro de tratamentos realizados, incluindo procedimentos, datas e dentistas responsáveis.

RNF002: O sistema deve garantir a segurança dos dados dos pacientes, utilizando criptografia e medidas de proteção.

RF003: Possibilitar o controle de despesas, permitindo o registro de gastos com materiais, equipamentos e serviços.

RNF003: O sistema deve ter um tempo de resposta rápido, mesmo em períodos de alta demanda.

## HU02

**CARD: Como dentista ou administrativo, quero me autenticar no sistema para ter acesso às minhas funções.**

### Conversation:

- O usuário (dentista ou administrativo) deve:
  - Informar o CPF e a senha cadastrados para realizar a autenticação.
- O sistema deve:
  - Verificar as credenciais informadas e, se válidas, liberar o acesso ao sistema.
  - Exibir uma mensagem de erro caso as credenciais sejam inválidas, solicitando a correção dos dados.

### Confirmation:

- O usuário pode se autenticar no sistema com CPF e senha válidos.
- O sistema direciona o usuário ao módulo de pacientes após a autenticação bem-sucedida.
- Caso as credenciais estejam incorretas, uma mensagem de erro é exibida.

## HU11

**CARD: Como administrativo, quero registrar entradas e saídas de materiais e equipamentos odontológicos para controlar o estoque.**

### Conversation:

- O usuário administrativo deve ser capaz de:
  - Cadastrar materiais e equipamentos odontológicos, informando:
    - Nome
    - Descrição
    - Quantidade em estoque
    - Preço
  - Registrar retiradas de materiais e equipamentos, informando:
    - Quantidade

### Confirmation:

- É possível cadastrar materiais e equipamentos odontológicos com as informações necessárias (nome, descrição, quantidade mínima, preço).
- As retiradas de materiais e equipamentos são registradas corretamente com quantidade.
- O usuário consegue visualizar o estoque atual, incluindo a identificação de itens, nome e descrição.



## HU12

# CARD: Como administrativo, quero cadastrar pacientes no sistema para agilizar o atendimento e manter um histórico completo

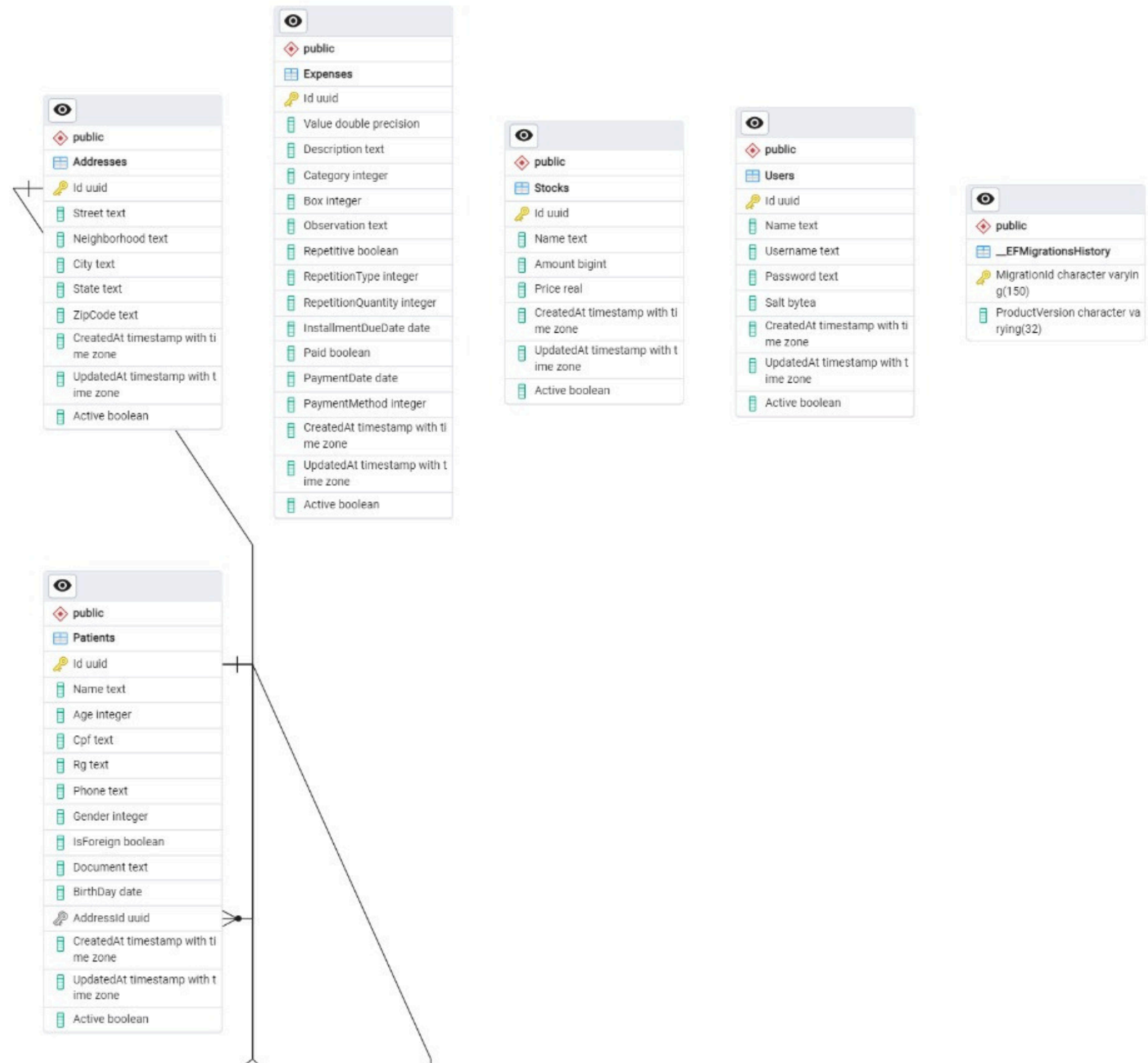
### Conversation:

- O usuário administrativo deve ser capaz de cadastrar pacientes no sistema, informando os seguintes dados:
  - Nome completo
  - Data de nascimento
  - CPF
  - RG
  - Endereço
  - Telefone
  - E-mail
- O objetivo é garantir um atendimento mais rápido e eficiente, além de manter um registro completo de cada paciente.

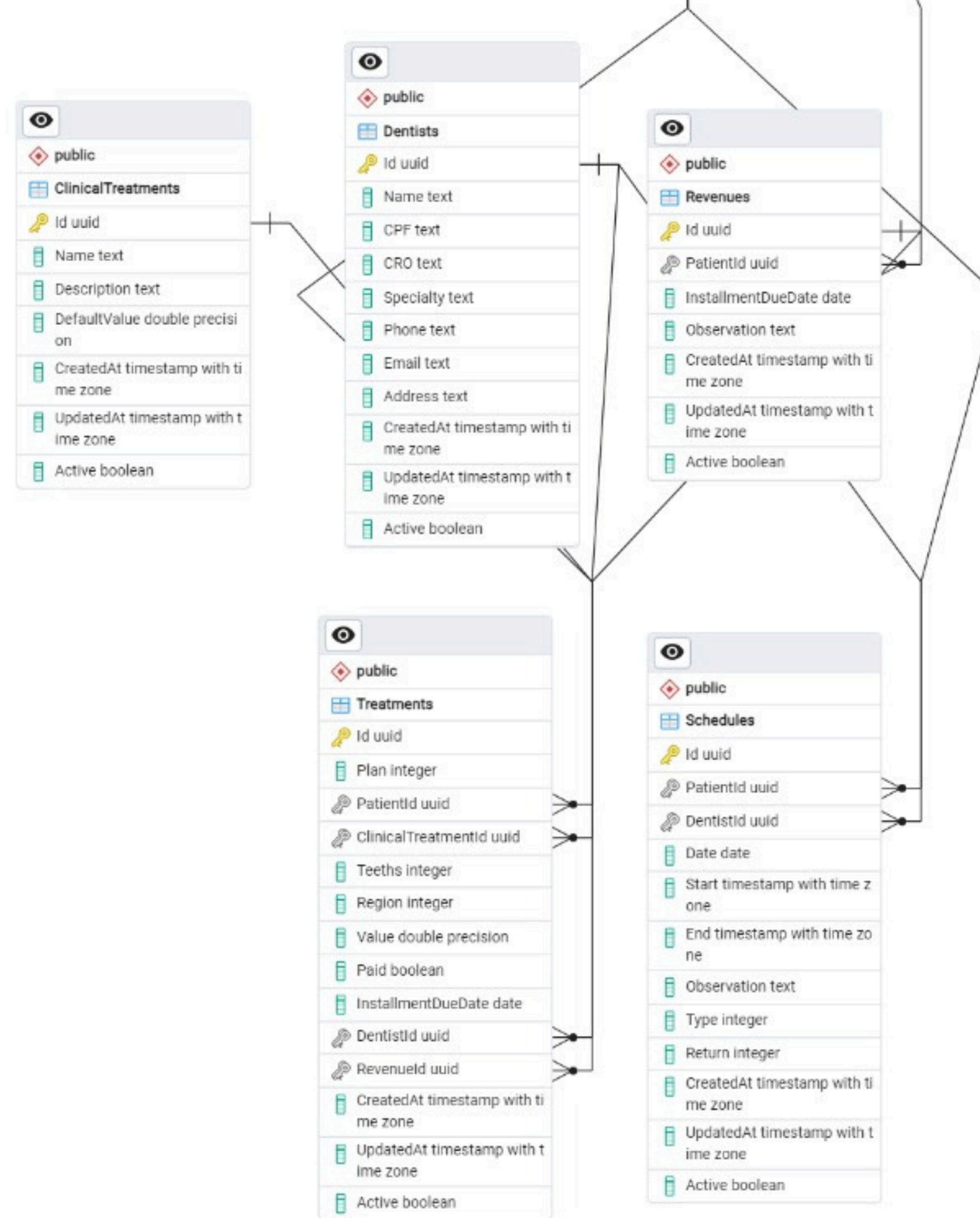
### Confirmation:

- É possível cadastrar pacientes com todos os campos obrigatórios (nome completo, data de nascimento, CPF, RG, endereço, telefone, e-mail).
- O sistema exibe corretamente os dados cadastrados, garantindo que o histórico do paciente está completo.

# Classes



# Classes



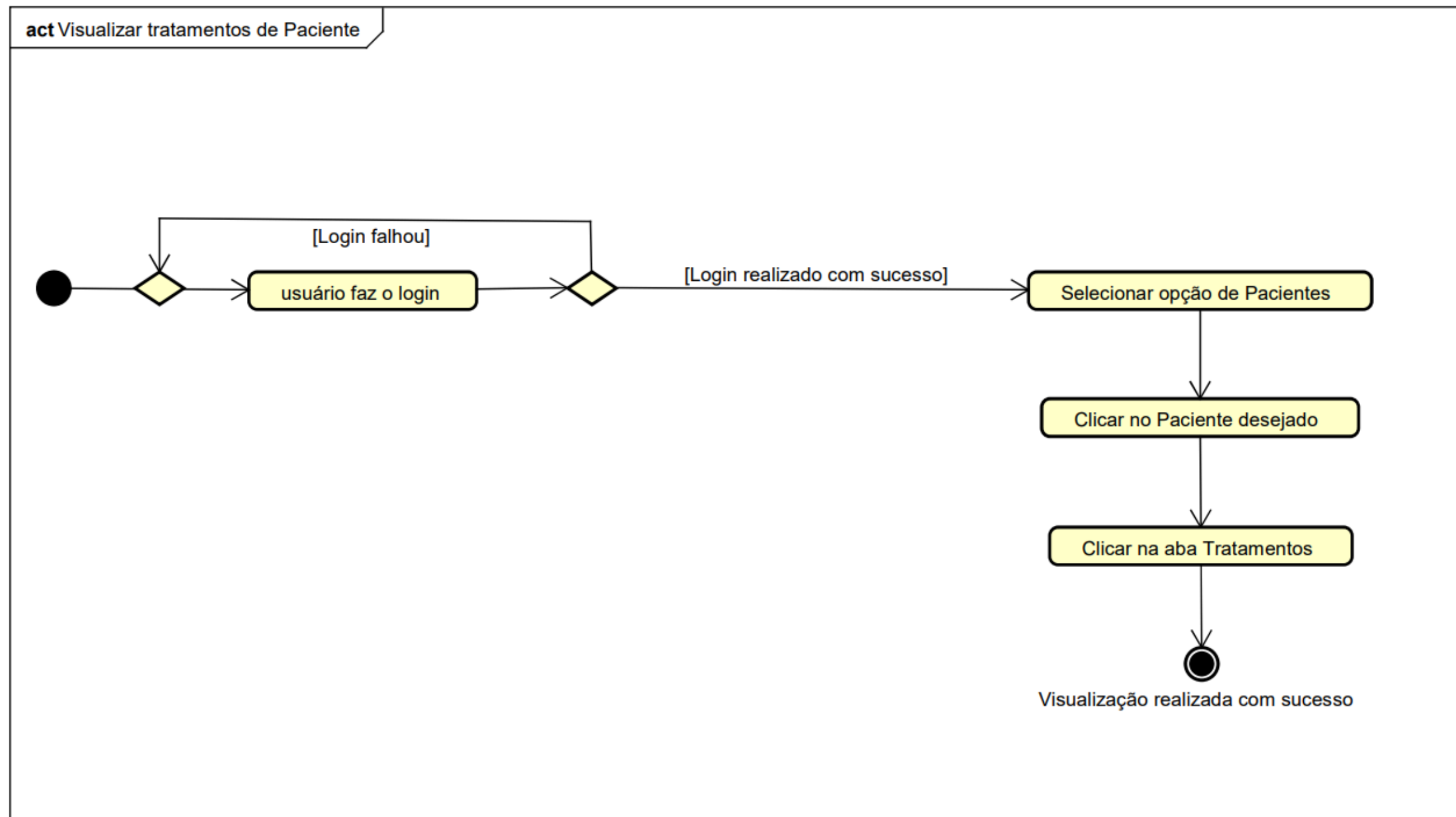


# Diagramas de Atividades

## - 01

Visualizar tratamentos de Paciente

2024/09/17

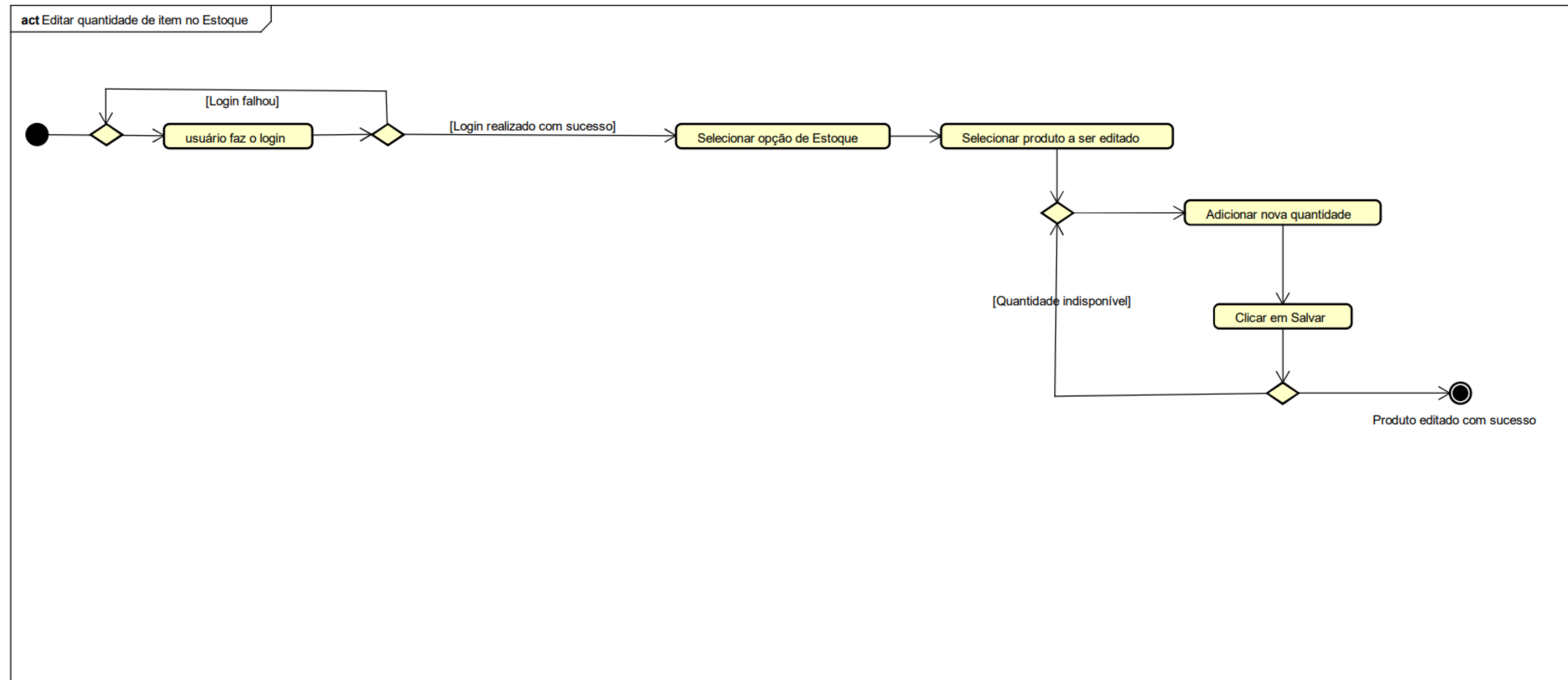


# Diagramas de Atividades

## - 02

Editar quantidade de item no Estoque

2024/09/17

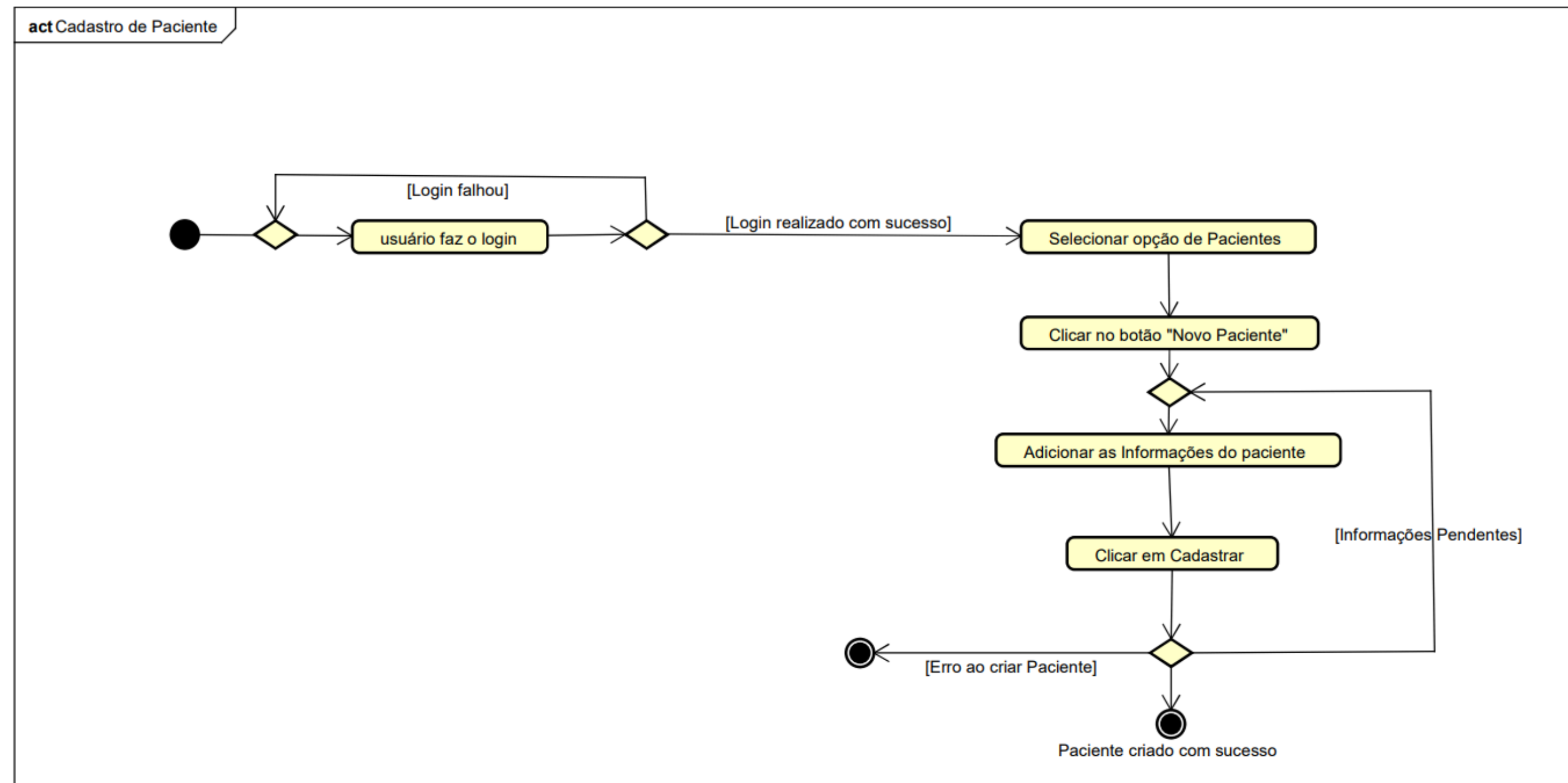


# Diagramas de Atividades

## - 03

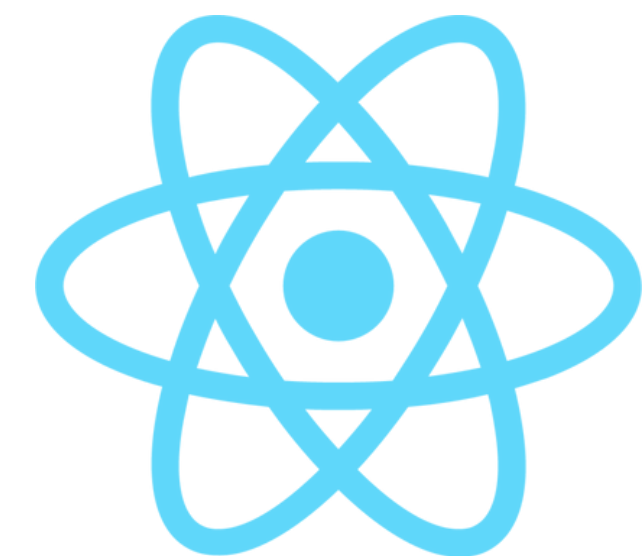
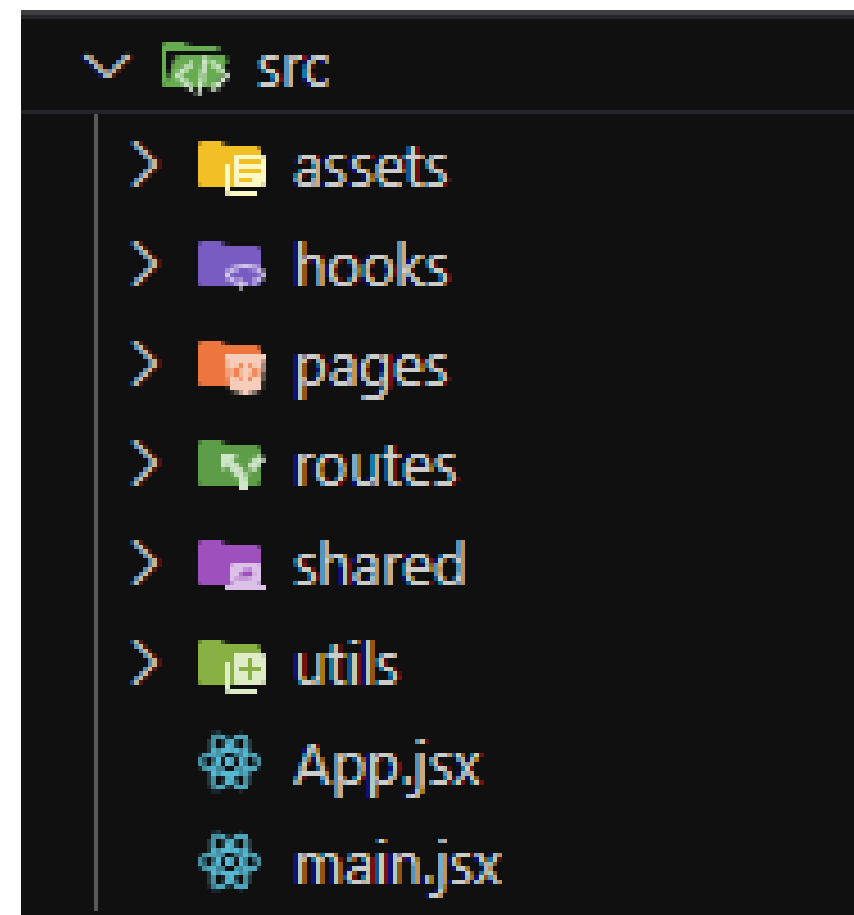
Cadastro de Paciente

2024/09/17

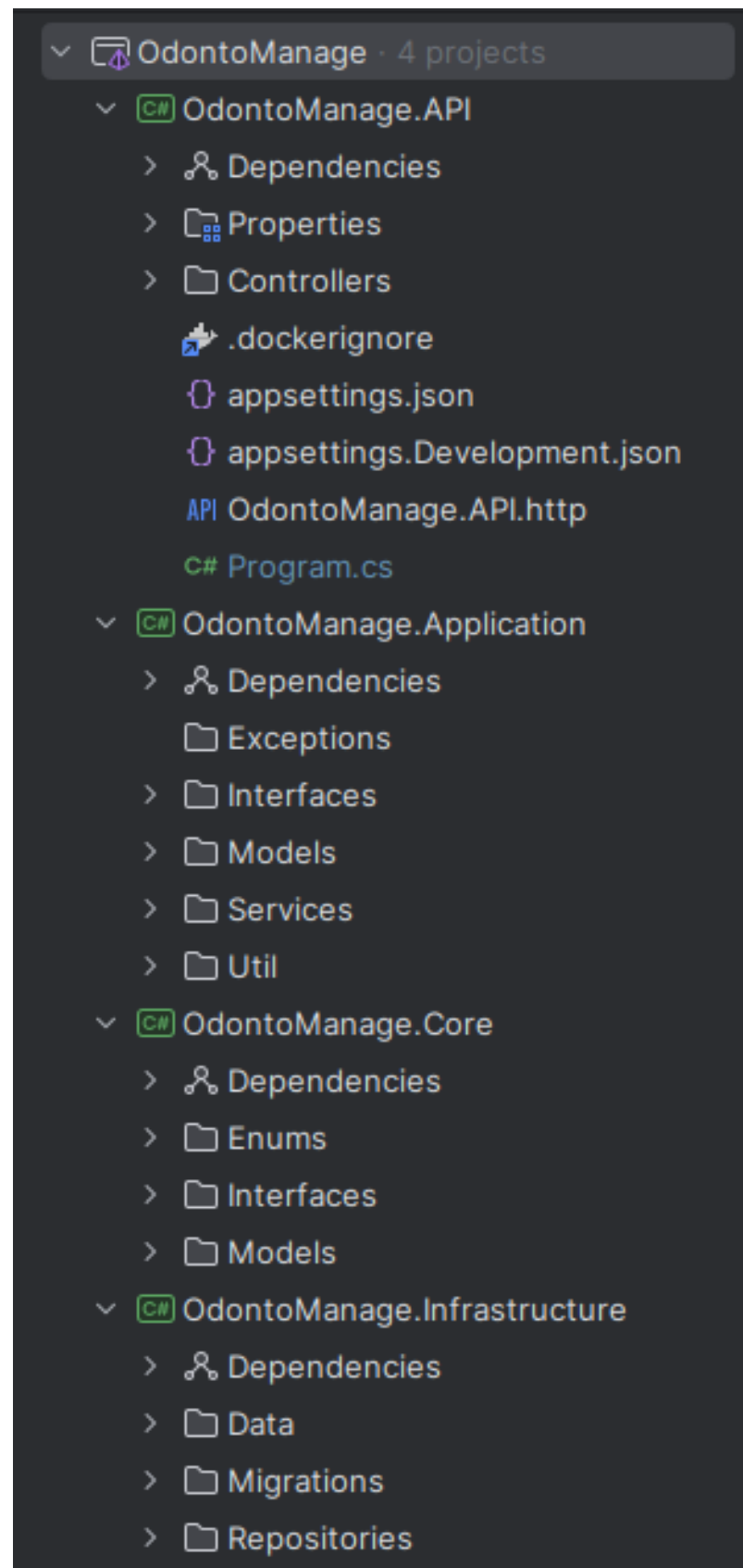


# Front-End

- Context:
  - AuthContext
  - DataContext
- Camadas na integração
  - Serviço
  - API



React



# BACK-END

.NET (C#)


Clean Architecture  
API .

Application .  
Infrastructure .  
Core .

PostgreSQL  
Entity Framework Core  
Docker  
Swagger




# Back (Swagger)

 **Swagger**  
Supported by SMARTBEAR

Select a definition OdontoManage.API v1

## Api Docs v1 OAS3


<https://localhost:7188/swagger/v1/swagger.json>

Authorize 

### Dentist


POST

/dentists

✓ 


GET

/dentists

✓ 


GET

/dentists/all-no-pagination

✓ 


PUT

/dentists/{id}

✓ 

DELETE


/dentists/{id}

✓ 

### Expense


POST

/expenses

✓ 


GET

/expenses

✓ 


PUT

/expenses/{id}


✓ 

DELETE


/expenses/{id}

✓ 


# Deploy

lgmleite/odonto-manage-api 

Last pushed 9 minutes ago











Contém imagens Docker do back-end do projeto de Engenharia de Software. 

API MANAGEMENT

WEB SERVERS 

Tags

This repository contains 7 tag(s).

Tag	OS	Type	Pulled	Pushed
 v0.0.6		Image	8 minutes ago	9 minutes ago
 v0.0.5		Image	7 hours ago	7 hours ago
 v0.0.4		Image	8 hours ago	8 hours ago
 v0.0.3		Image	a day ago	8 days ago
 v0.0.2		Image	9 hours ago	11 days ago

[See all](#)

Docker commands


To push a new tag to this repository:

```
docker push lgmleite/odonto-manage-api:tagname
```

Public View

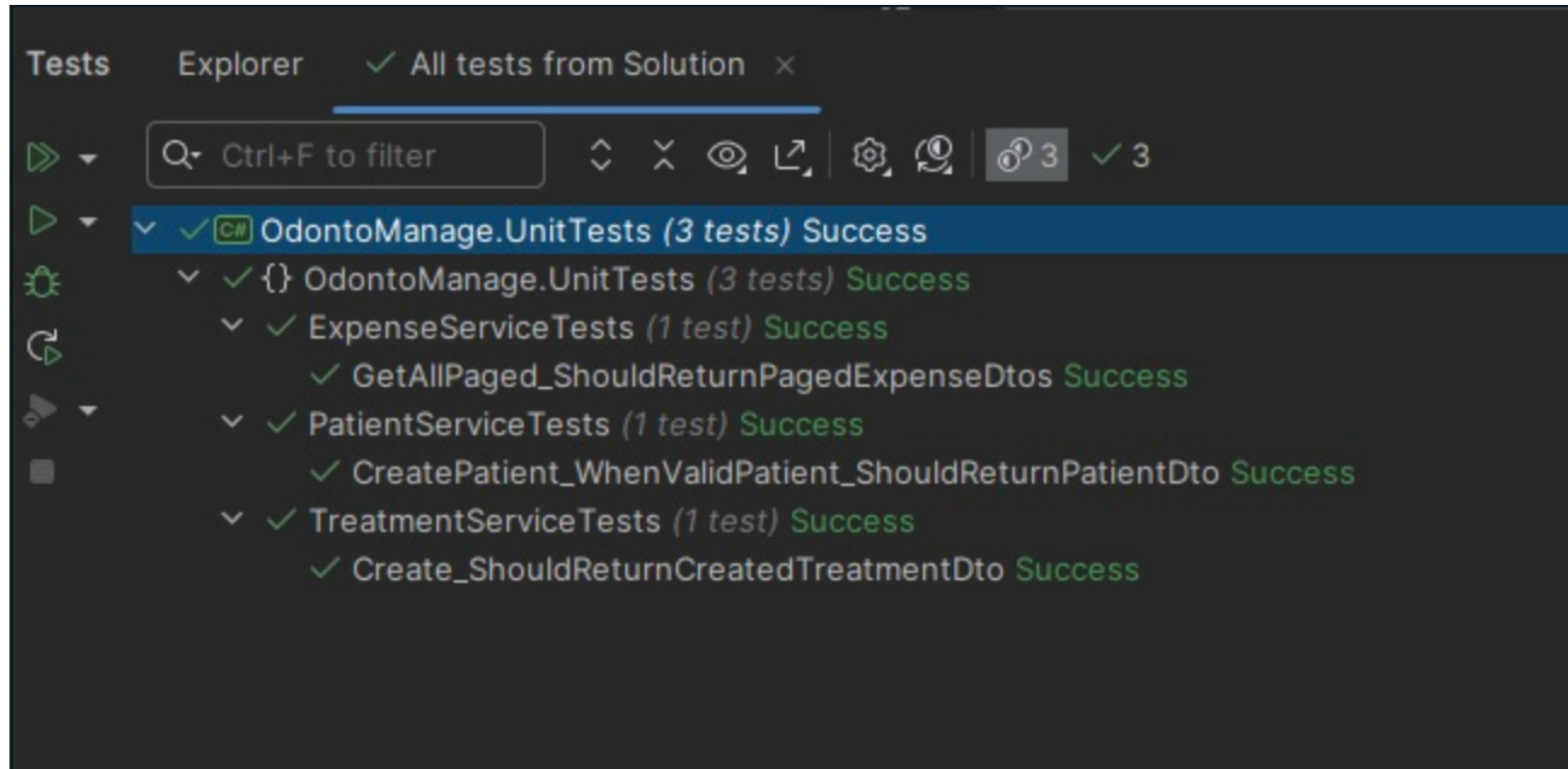
Automated Builds

Manually pushing images to Hub? Connect your account to GitHub or Bitbucket to automatically build and tag new images whenever your code is updated, so you can focus your time on creating.

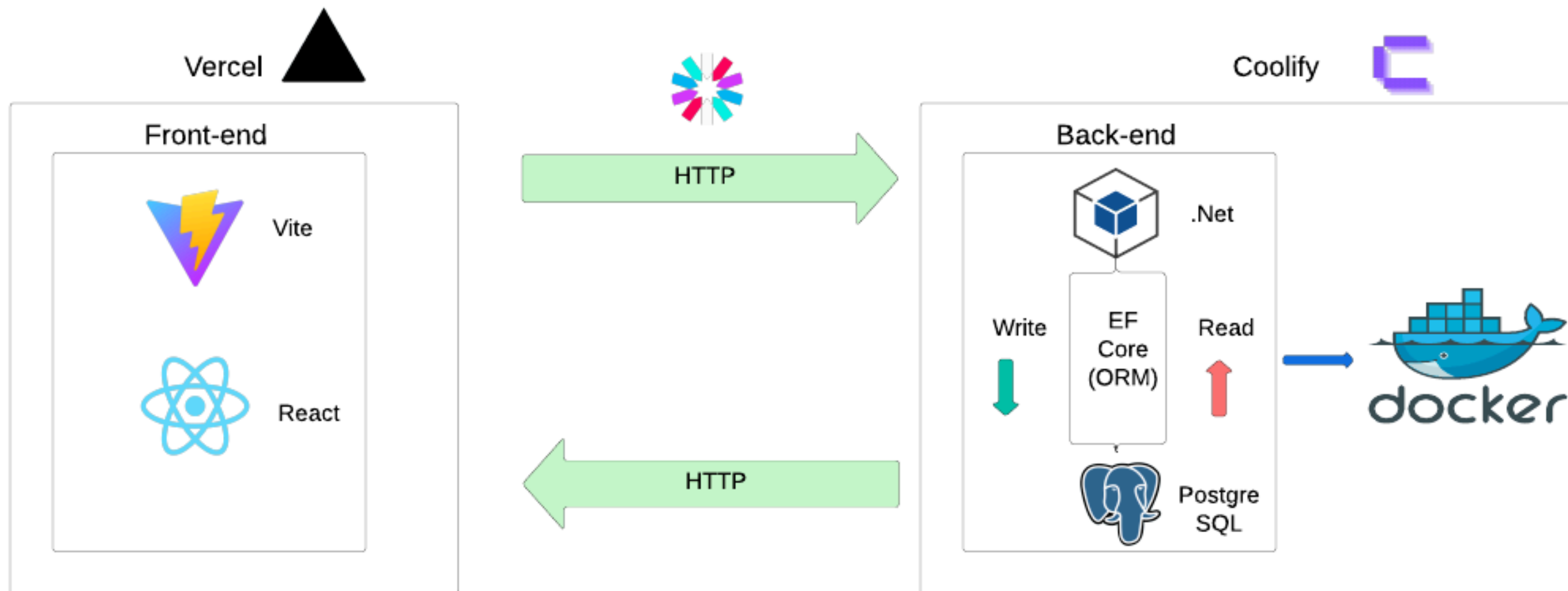
Available with Pro, Team and Business subscriptions. [Read more about automated builds](#) .

Upgrade

# Testes



# Arquitetura do Sistema



# Testes



**Cypress**

Testes no Front-End



**NUnit**

Testes no Back-End





# GitHub



## BACK-END

<https://github.com/AntDeivid/OdontoManage>



## FRONT-END

<https://github.com/JorgeEdu1/ProjetoESFront>