



INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO PIAUÍ
CAMPUS TERESINA CENTRAL
CURSO TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

IZADORA PAMELLA SILVA OLIVEIRA
SHARA CRISTTINA SOUSA LOPES

ANIMALHOTELS:
PLATAFORMA DE GERENCIAMENTO DE TUTORES E ANIMAIS

TERESINA
2025

I. Objetivos

O objetivo principal da atividade é aplicar os conceitos de [ReactJS](#) e [TypeScript](#), desenvolvendo um projeto funcional em dupla ou trio, seguindo boas práticas de componentização, arquitetura, gerenciamento de estado e integração com API. O projeto consiste em simular parte da plataforma AnimalHotels (uma espécie de Airbnb para animais), com foco no módulo de Tutores e Animais, permitindo que usuários cadastrados gerenciem múltiplos tutores e, para cada tutor, diversos animais.

Além disso, o objetivo implícito é reforçar a fixação dos conceitos e da filosofia do ReactJS, consolidando o aprendizado teórico por meio da prática e da aplicação em um projeto realista.

II. Requisitos Gerais do Projeto

Item	Descrição
Autenticação	Sistema de login com token JWT
CRUD Tutor	Nome, email, telefone
CRUD Animal	Nome, espécie (gato, cachorro), raça, idade, tutor, foto (opcional)
Relação Tutor-Animal	Um Tutor pode ter vários Animais
API	Consumo de API real ou mock
Rotas	Públicas e privadas
Gerenciamento de Estado	Local e global (ContextAPI + Reducer)
Hooks & Formulários	useState, useReducer, useMemo, useCallback, useRef; formulários com react-hook-form ou Formik
Organização	Pastas, componentes e estilização

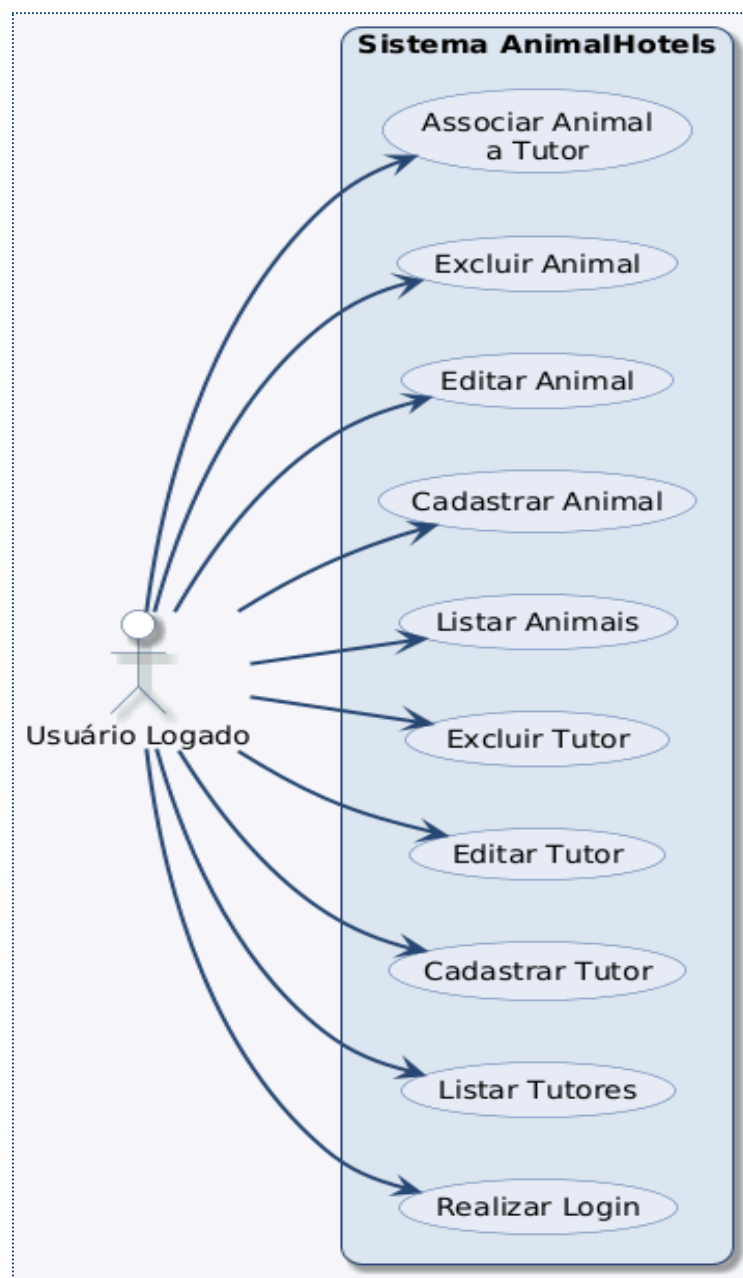
III. Ferramentas e Bibliotecas

Categoria	Descrição
Base	Vite + React + TypeScript
Roteamento	react-router-dom
Requisições HTTP	axios, react-query ou outra lib
Formulários	react-hook-form ou Formik



Gerenciamento de Estado	ContextAPI + Reducer
Hooks Adicionais	useState, useReducer, useMemo, useCallback, useRef
Estilização	TailwindCSS, CSS Modules, Styled-components, shadcn/ui, Next.js
API Mock/Própria	json-server ou criar API com Node/TS, Java ou Python

IV. Diagrama Caso de Uso



V. Relatório Dia 26/11

Na etapa inicial do dia, foi realizada a preparação completa do ambiente de desenvolvimento do projeto *AnimalHotels*. As principais ações executadas foram:

- ❖ Identificação e acesso correto ao diretório raiz do projeto;
- ❖ Reorganização da estrutura de pastas conforme padrões recomendados para aplicações React + TypeScript: `src/components`, `src/routes`, `src/services`, `src/contexts` entre outras.
- ❖ Verificação dos arquivos essenciais do projeto, como: `package.json`, `vite.config.js`, `tsconfig.json`, diretório `src/`.

Durante os testes iniciais, foram identificados erros que impediam a execução do projeto, entre eles:

- ❖ Mensagem de erro: *"Cannot find module 'react-router-dom'"*
- ❖ Falha ao executar o comando `npm run dev`.

As ações realizadas para correção incluíram:

- ❖ Ajuste completo no arquivo `tsconfig.json`, garantindo compatibilidade com React + TypeScript + Vite.
- ❖ Instalação das dependências necessárias para o funcionamento adequado do projeto:
`react`, `react-dom`, `@types/react`, `@types/react-dom`, `react-router-dom`, `axios`, entre outras dependências relevantes.

Após todas as correções, foram realizados testes de execução:

- ❖ O servidor Vite foi iniciado com sucesso através do comando `npm run dev`.
- ❖ O projeto abriu corretamente no navegador, sem erros de compilação.
- ❖ Todas as dependências e configurações foram validadas, garantindo estabilidade inicial para prosseguir com o desenvolvimento.

Foi criada e validada a seguinte estrutura padrão:

```
animalhotels/  
├─ src/  
│   ├── components/  
│   ├── pages/  
│   ├── routes/  
│   ├── contexts/  
│   ├── hooks/  
│   ├── services/  
│   ├── types/  
│   ├── utils/  
│   └─ styles/
```