**Centro Universitário Estácio do Ceará**

**PARANGABA**

**Projeto Cross Training**

**Igor Stenio Pereira Santos**

**Luan Victor Dias Lopes**  
**Vitor Vinícius Lima Silva**

**Willian**

**Lian**

**Orientador: Cynthia Maia**

**2025**

**Fortaleza/CE**

Sumário

[**1.** Planejamento de requisitos 3](#_Toc199157243)

[1.1. Identificação das partes interessadas e parceiros 3](#_Toc199157244)

[1.2. Problemática e/ou problemas identificados 3](#_Toc199157245)

[1.3. Justificativa 3](#_Toc199157246)

[1.4. Objetivos/resultados/efeitos a serem alcançados (em relação ao problema identificado e sob a perspectiva dos públicos envolvidos) 3](#_Toc199157247)

[2. Design do usuário 3](#_Toc199157248)

[2.1. Protótipo 3](#_Toc199157249)

[3. Construção 3](#_Toc199157250)

[3.1. Refinando o protótipo e criando um entregável 3](#_Toc199157251)

[4. Transição 3](#_Toc199157252)

[4.1. Testes de integração 3](#_Toc199157253)

[4.2. Implementação 3](#_Toc199157254)

# Planejamento de requisitos

## Definição do escopo

O projeto Cross Training se iniciou com o anseio da responsável pelo estabelecimento em criar um sistema que fosse capaz de gerenciar as principais informações do seu negócio em um só lugar, deixando de lado a caderneta de anotações e possibilitando consultas mais eficientes.

Para que este objetivo pudesse ser efetivamente aplicado, foi estabelecido o seguinte escopo do projeto em conjunto a todas os Stakeholder do projeto:

|  |  |
| --- | --- |
| Manter informação de contato de alunos | Um novo aluno é adicionado ao sistema. Manter a seguinte informação de contato dos alunos: Nome, Endereço, Cidade, Estado, Telefone. "Manter" significa que podem ser feitas 4 operações (ou CRUD): Create, Read, Update, Delete. |
| Alterar estado "matriculado" do cadastro | Uma pessoa cadastrada poderá ser aluna ou não. Deve-se manter a data de matrícula e a data em que o aluno deixa de ser aluno, data de vencimento da matrícula (mas sem eliminar o cadastro). Não se deve remover um registro se a pessoa ainda for aluno (tipicamente, você não joga fora seu cadastro, mesmo depois que alguém deixa de ser aluno!). |
| Registrar pagamento | Registrar todos os pagamentos feitos pelo aluno (valor, data, tipo). Cada pagamento tem um tipo. Não precisa fazer cadastro de novos tipos de pagamentos. Os tipos de pagamentos podem ser dinheiro ou cartão. |

# Design do usuário

## Criação do Protótipo

Para o desenvolvimento rápido de um protótipo para a validação do cliente, foi utilizado o modelo de projeto MTV (Model, Templates e View) em conjunto ao framework *Flask* e todas suas dependências, buscando atingir o desenvolvimento rápido do projeto.

Deste modo, foram desenvolvidos seguindo a estrutura de projeto menciona 4 arquivos bases para testar a viabilidade de execução do escopo mencionado no tópico 1.1, sendo 2 destes referente a configuração do servidor *Flask* e a integração das funcionabilidades, estes arquivos foram denominados de *app.py* e *\_\_init\_\_.py,* para o desenvolvimento das funcionabilidades e criação de rotas foram criados mais 2 arquivos denominado *aluno\_view.py* e *aluno\_rotas.py.*

Do ponto de vista técnico, alguns detalhes são relevantes, para a realização de testes foi criado um ambiente de desenvolvimento no *Flask* no *IP local na porta 8080* e instanciado alguns *endpoints* no modelo API que recebe dados em *Json* e retorna respostam também neste modelo.

Após o desenvolvimento do protótipo a partir dos modelos mencionados nos parágrafo anteriores, foi entregue o protótipo do projeto para teste dos Stakeholders, e este já foi capaz de satisfazer duas das três principais exigências mencionadas no escopo do projeto.

# Construção

## Refinando o protótipo e criando um entregável

# Transição

## Testes de integração

## Implementação

**Tipo de projeto: Intensivo;**