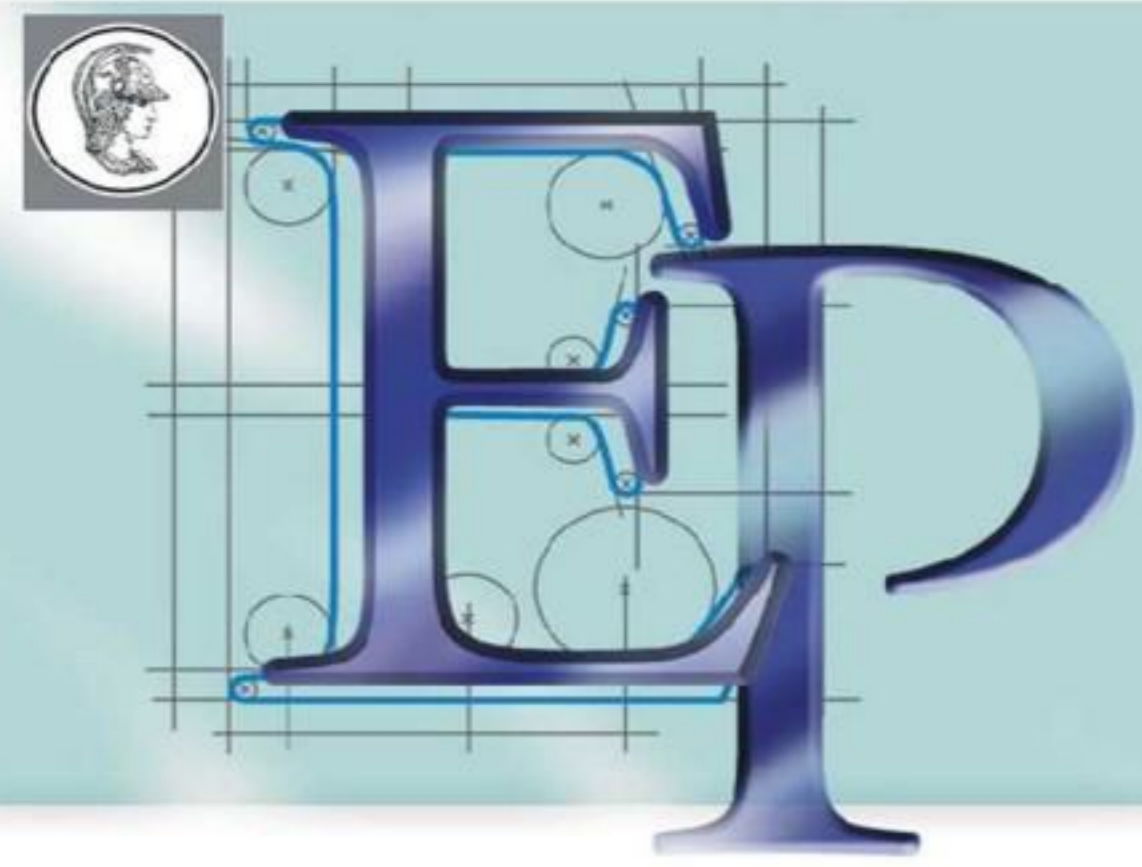


Projeto de Formatura – 2024



PCS - Departamento de Engenharia de Computação e Sistemas Digitais

Engenharia de Computação

Tema: Desenvolvimento de uma solução para monitoramento de presença e engajamento para o Cursinho Popular da Escola Politécnica da Universidade de São Paulo

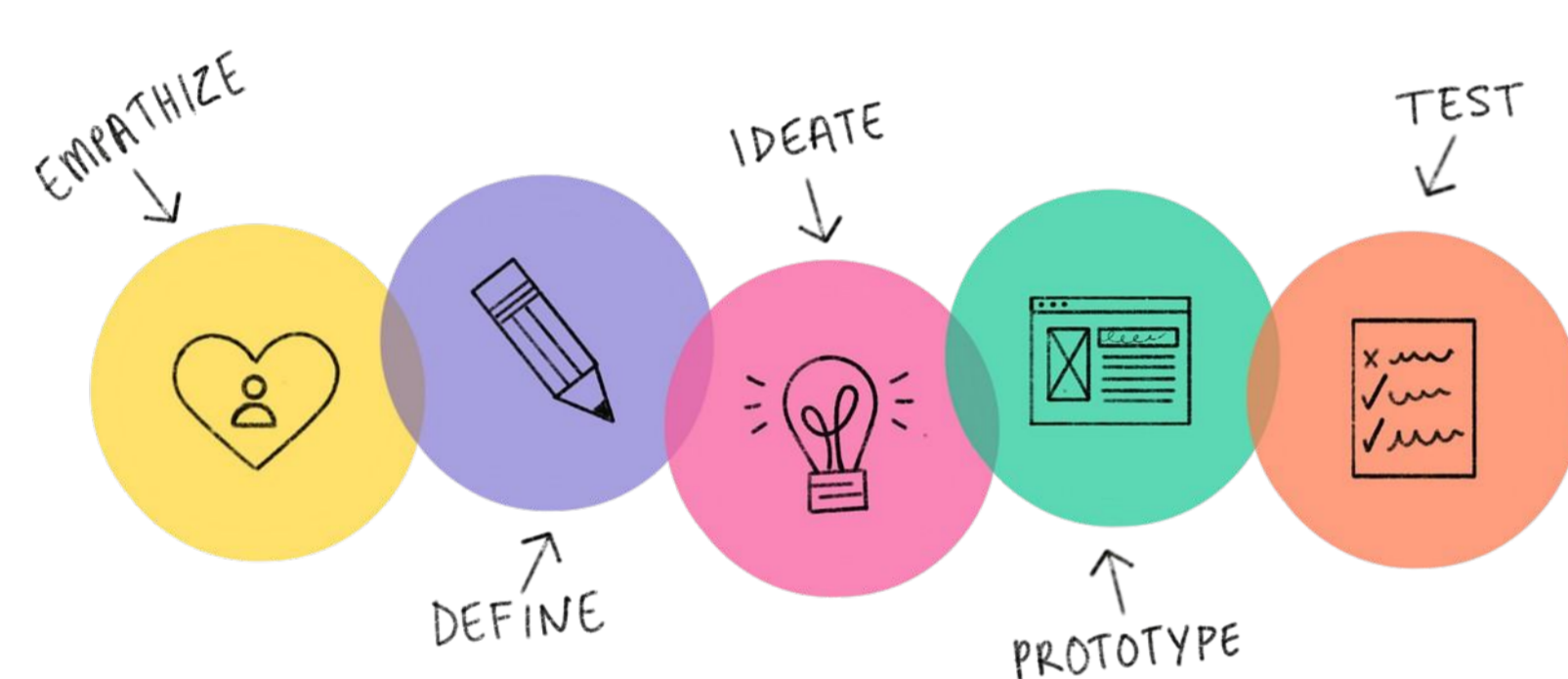
INTRODUÇÃO E OBJETIVOS

O **Cursinho Popular da Poli** é um curso social preparatório para vestibulares com foco em estudantes de baixa renda. Para auxiliá-los, o CP utiliza a presença dos alunos para acompanhar a evasão, por meio de uma lista assinada e passada manualmente para o sistema. Isso gera erros e atrasos de até 2 meses, além de fazer com que a presença das turmas online não seja controlada. Dessa forma, o objetivo é criar um **ambiente de controle de presença** para o CP, buscando automatizar o processo e deixá-lo mais seguro e consistente.

METODOLOGIA

DESIGN DA SOLUÇÃO

O projeto utilizou uma **adaptação**, feita pelo grupo, das duas metodologias a seguir, seguindo a divisão de fases e algumas ferramentas do Design Thinking, mas implementando as instruções e atividades do Design Sprint.

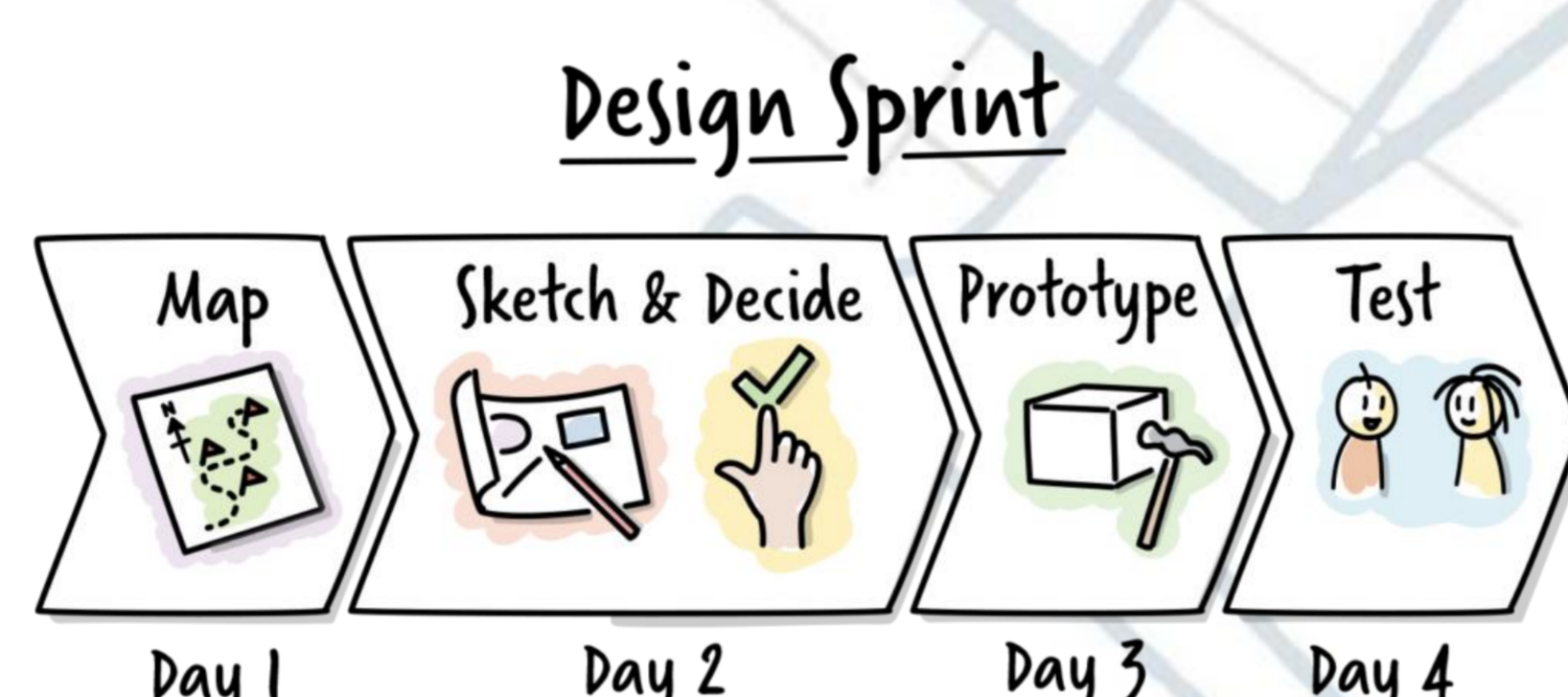


• Design Thinking

Série de conceitos e ferramentas usados para identificar problemas e criar ideias e soluções centradas na **necessidade do usuário** [1].

• Design Sprint

Aplicação do Design Thinking para responder **questões críticas de negócio** em curto período [2].

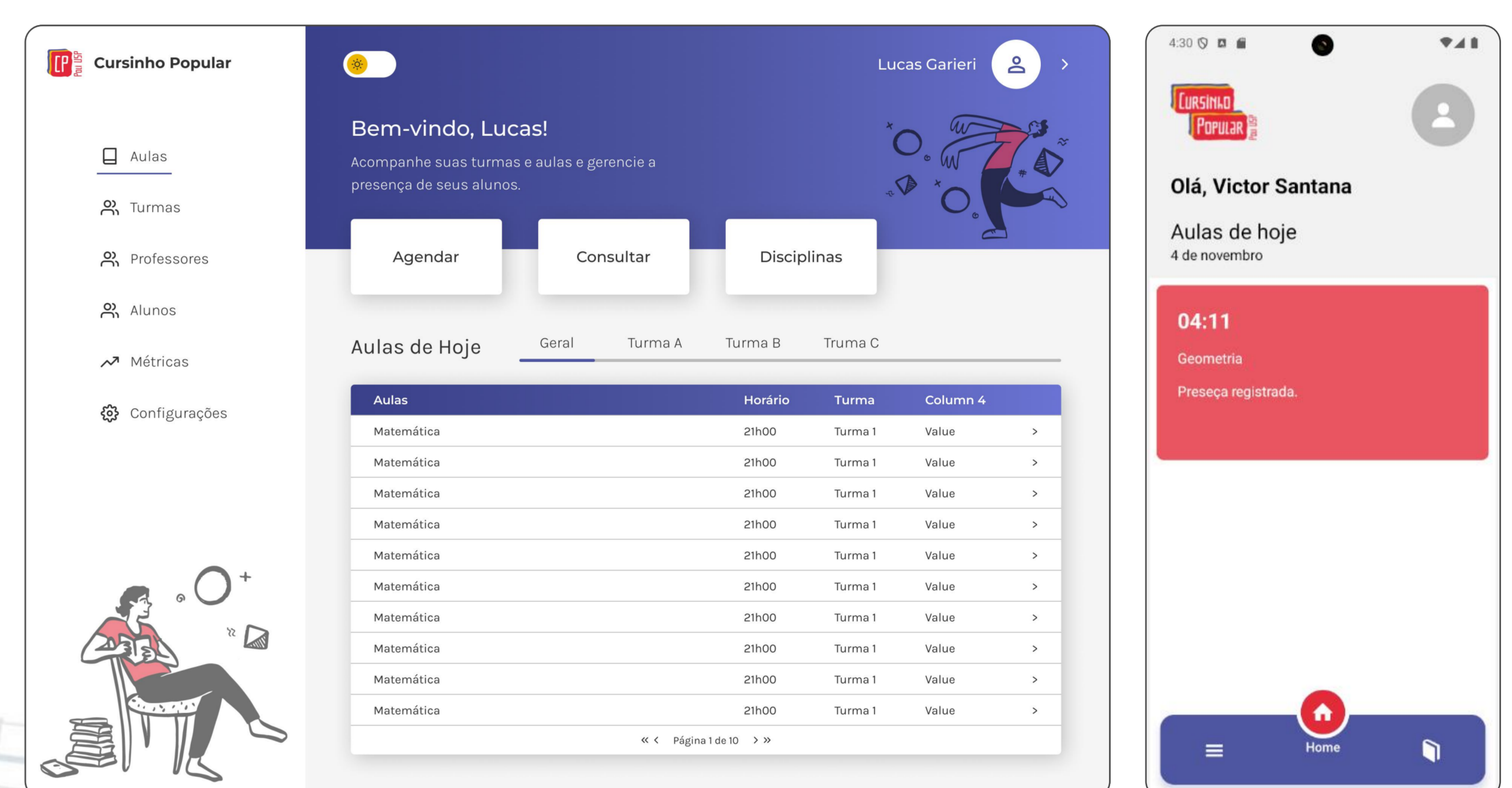


Assim, focou-se primeiro em mapear os usuários do sistema e suas necessidades, entender o problema, desenvolver ideias, prototipá-las e testá-las juntamente ao cursinho. Decidiu-se, então, por um ambiente composto por **dois sistemas**, um mobile e um web, com a marcação de presença por meio de **palavras-chave**. Além disso, o projeto também tem, como requisito, permitir o cadastro de alunos, criação de turmas e matérias, gerenciamento de aulas e acompanhamento de métricas, permitindo centralizar as informações do cursinho.

DESENVOLVIMENTO

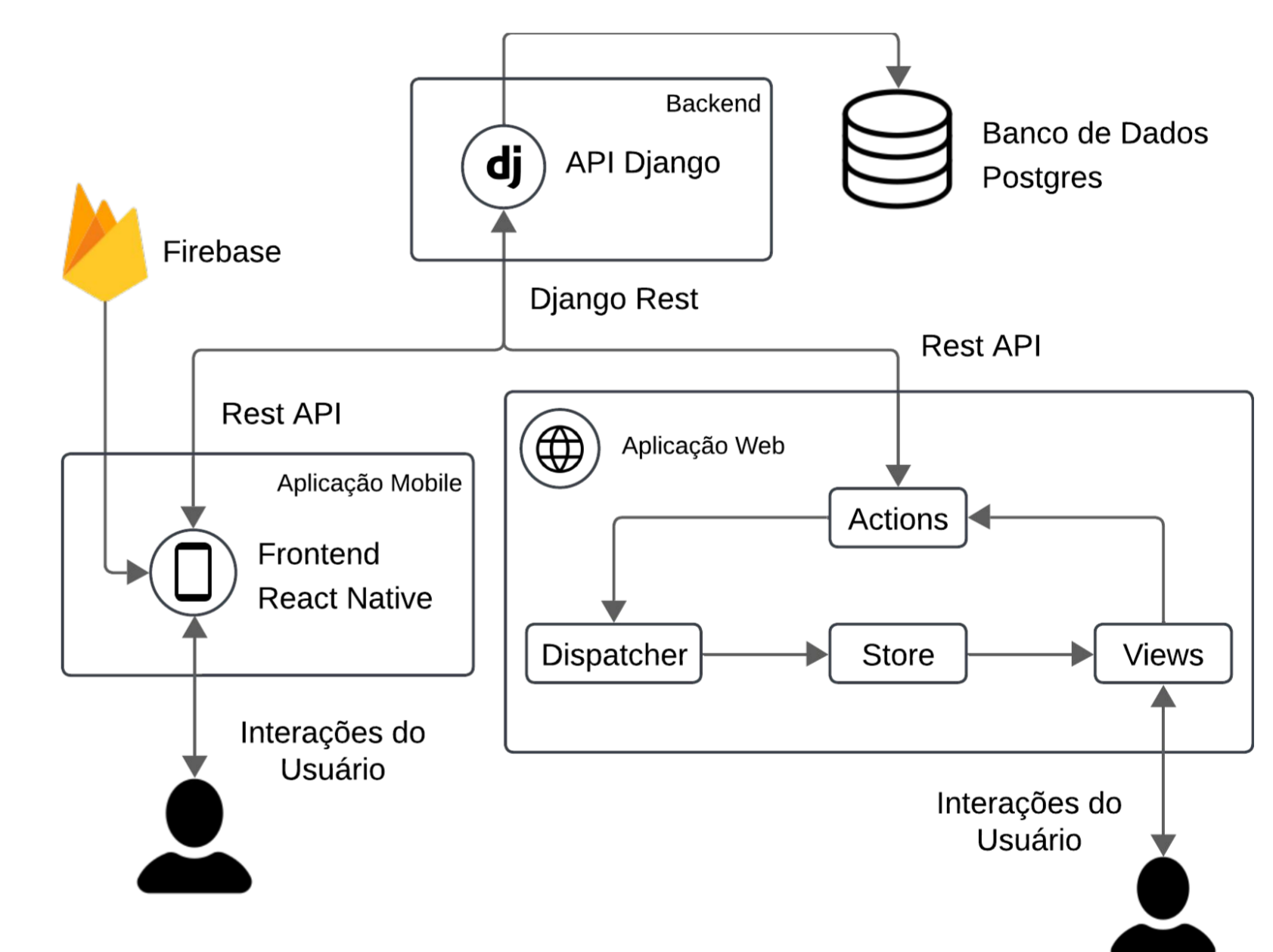
GERENCIAMENTO E TECNOLOGIAS

Seguiu-se a metodologia **Scrum** [3], permitindo a divisão em sprints de 2 semanas com entregas incrementais. Além disso, foram utilizadas as tecnologias **Django** para o backend, **React** e **React Native** para o frontend, **PostgreSQL** para o banco de dados e **Redux** para o gerenciamento de estados.



ARQUITETURA

O sistema segue a arquitetura **MVT** [4], implementada pelo Django. Além disso, o sistema mobile implementa **clean architecture** [5], permitindo um código mais modular. Já o web usa a **flux architecture** [6] para gerenciar os estados, a lógica de atualização e as requisições do sistema, dada sua maior complexidade.



RESULTADOS E CONCLUSÕES

A testagem do sistema totalizou um mês e contou com **testes de usabilidade**, uso em **ambiente controlado** e **pesquisas de satisfação**. Concluiu-se que:

- Foi possível **zerar o atraso** no lançamento das presenças;
- Os alunos passaram a **controlar melhor** suas presenças;
- O processo ficou mais **transparente** e **confiável**;

REFERÊNCIAS

- [1] LEWRICK, M.; LINK, P.; LEIFER, L. **The design thinking playbook**. Hoboken, New Jersey: John Wiley & Sons, Inc, 2018.
- [2] KNAPP, J.; ZERATSKY, J.; KOWITZ, B. **Sprint: How to Solve Big Problems and Test New Ideas in Just Five days**. New York: Simon & Schuster, 2016.
- [3] SCHWABER, K.; SUTHERLAND, J. **Scrum Guide**. Disponível em: <<https://scrumguides.org/scrum-guide.html>>.
- [4] DJANGO SOFTWARE FOUNDATION. **Django documentation | Django documentation**. Disponível em: <<https://docs.djangoproject.com/en/5.1/>>.
- [5] MARTIN, R. C. **Clean Architecture: A Craftsman's Guide to Software Structure and Design**. Boston: Prentice Hall, 2017.
- [6] FACEBOOK. **Flux: An Application Architecture for React**. 2014. Disponível em: <<https://facebookarchive.github.io/flux/docs/in-depth-overview>>.

Integrantes:

- Lucas Pavan Garieri
- Pedro Henrique Rodrigues de Viveiros
- Victor de Almeida Santana

Professor Orientador: Prof. Dr. Fábio Levy Siqueira