

IFSP - Instituto Federal de Educação, Ciência e Tecnologia  
Câmpus São Paulo

Bruna da Silva Pires	SP3056651
Daniel Roberto Pereira	SP3046702
Igor Nathan de Oliveira Rocha	SP305263X
Leonardo Marques da Silva	SP3052591
Lucas Lima de Santana	SP3046559
Marcelo Carlos Olimpio Junior	SP3046583

**Portal de Vagas de Estágio**

Proposta de projeto para disciplina PI1A5

Professor: Carlos Henrique Veríssimo Pereira

IFSP - Instituto Federal de Educação, Ciência e Tecnologia  
Câmpus São Paulo

Tecnologia em Análise e Desenvolvimento de Sistemas

PI1A5 - Projeto Integrado I

São Paulo - SP - Brasil

2022

# Lista de abreviaturas e siglas

API	<i>Application Programming Interface</i> - Interface de Programação de Aplicativos - Citado em <a href="#">9</a>
LGPD	Lei Geral de Proteção de Dados - Citado em <a href="#">7</a>
SSO	<i>Single Sign-On</i> - Login único - Citado em <a href="#">9</a>

# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>4</b>
<b>1.1</b>	<b>Justificativa</b>	<b>4</b>
<b>1.2</b>	<b>Proposta de solução</b>	<b>4</b>
<b>1.3</b>	<b>Objetivos</b>	<b>4</b>
<b>2</b>	<b>REQUISITOS</b>	<b>6</b>
<b>2.1</b>	<b>Requisitos Funcionais</b>	<b>6</b>
<b>2.2</b>	<b>Requisitos Não-funcionais</b>	<b>6</b>
2.2.1	Requisitos de produtos	7
2.2.2	Requisitos externos	7
<b>2.3</b>	<b>Regras de Negócio</b>	<b>7</b>
<b>3</b>	<b>TECNOLOGIAS</b>	<b>8</b>
<b>3.1</b>	<b>Arquitetura</b>	<b>8</b>
3.1.1	Diagramas de arquitetura	8
<b>3.2</b>	<b>Integrações</b>	<b>8</b>
3.2.1	Login com o Google e LinkedIn	9
3.2.2	Entrar em contato via <i>Whatsapp</i>	9
<b>4</b>	<b>ANÁLISE DE CONCORRENTES</b>	<b>12</b>

# 1 Introdução

Nesse capítulo serão mostrados os principais pontos do nosso projeto, os objetivos e quais os problemas que queremos solucionar com nossa aplicação.

## 1.1 Justificativa

Existe, na contemporaneidade, uma grande dificuldade em adquirir experiência profissional através da prática de estágio, muitas vezes obrigatória no projeto pedagógico de curso das universidades. Tal problema se dá por meio das plataformas que disponibilizam tais vagas, porém com uma certa cobrança injusta em relação a habilidades que o candidato precisa possuir previamente. É também notável que existe uma certa dificuldade de conexão entre a empresa e o candidato, que muitas vezes não obtém o retorno sobre o processo de seleção da vaga.

## 1.2 Proposta de solução

Tendo em vista os problemas anteriormente descritos, o Portal de Vagas de Estágio é um sistema para aproximar novos estudantes e empresas com vagas de estágio disponíveis, de modo que os candidatos possam receber indicações de vagas condizentes com seu perfil e empresas recebam recomendações de candidatos possivelmente adequados às vagas anunciadas.

## 1.3 Objetivos

O objetivo principal da nossa solução é promover um meio de conexão mais direto entre os estudantes em busca de estágio e empresas que buscam interessados em suas vagas de estágio alinhados com o perfil buscado. Através do sistema de recomendações, tanto os estudantes quanto as empresas têm papel ativo no processo de encontrar um(a) estudante/vaga ideal, cujas as competências e perfil sejam condizentes com o que é procurado.

A partir do nosso objetivo principal, podemos listar alguns objetivos mais práticos da nossa solução:

- Ser um *website* de fácil usabilidade, onde os estudantes encontrem vagas sem passar por longos processos seletivos.

- 
- Ser uma aplicação onde de fato os estudantes encontrem vagas que condizem com a realidade de um estagiário.
  - Pensar sempre na experiência dos usuários, de modo que a aplicação seja simples e efetiva ao mesmo tempo.

## 2 Requisitos

Nesse capítulo serão expostos os requisitos funcionais, não-funcionais e regras de negócio que nossa aplicação terá, tais requisitos foram formados a partir de estudos de como irá funcionar os processos de nosso *website*.

### 2.1 Requisitos Funcionais

Durante nossa análise, decidimos que esses seriam os principais requisitos funcionais do nosso projeto:

- Realizar o gerenciamento de vagas entre os candidatos e as empresas de uma forma simplificada;
- Recomendar vagas para estudantes, empresas para estudantes, estudantes para vagas/empresas;
- Manter um histórico de vagas aplicadas pelo estudante;
- Manter um histórico de candidatos aplicados a vaga;
- Exibir uma linha do tempo da situação da vaga;
- Alertar os estudantes aplicados à vaga sobre cada mudança em seu status;
- Possibilitar o gerenciamento da vaga pela empresa que a registrou/publicou;
- Possibilitar que a empresa possa acionar (entrar em contato) com os estudantes recomendados/aplicados à vaga;
- Possibilitar que a empresa realize mudanças no status da vaga;
- Possibilitar que o estudante realize um *feedback* da empresa pós-entrevista, que será visto por outros estudantes.
- Não permitir o registro de vagas cujas horas de atividades ultrapassem a carga horária prevista por lei de acordo com a situação escolar de cada estudante.

### 2.2 Requisitos Não-funcionais

Os requisitos não-funcionais do nosso projeto estão listados abaixo:

### 2.2.1 Requisitos de produtos

- Ser fácil de aprender a usar;
- Estar disponível 24 horas por dia, 7 dias por semana;
- Ter alta escalabilidade;
- Ser fácil de reparar erros;
- Tempo para carregar os dados necessários menor do que 15 segundos;
- Ter uma taxa de falhas menor que 3/1000;

### 2.2.2 Requisitos externos

- O sistema deve estar de acordo com a [Lei Geral de Proteção de Dados \(LGPD\)](#).

## 2.3 Regras de Negócio

- As vagas registradas pelas empresas para estudantes de educação especial e dos anos finais do ensino fundamental, na modalidade profissional de educação de jovens e adultos, não podem exigir mais de 4 horas diárias/20 horas semanais de dedicação para as atividades e no caso de estudantes do ensino superior, da educação profissional de nível médio e do ensino médio regular, 6 horas diárias/30 horas semanais, de acordo com a lei Nº 11.788, de 25 de setembro de 2008.

## 3 Tecnologias

Nesse capítulo serão citadas a arquitetura do nosso projeto com ilustrações demonstrando de forma mais lúdica, as possíveis integrações que nossa aplicação terá com sistemas externos.

### 3.1 Arquitetura

Para o desenvolvimento do projeto, e tendo em vista que será construída uma aplicação web de página única, utilizaremos de ferramentas que cerceiam o ecossistema de *Single Page Applications*. Para isso, teremos a divisão do projeto em *front-end* e *back-end* de modo que eles se comuniquem via protocolo HTTP com requisições e respostas no formato JSON. Para o desenvolvimento do *front-end* utilizaremos Typescript por meio da biblioteca React; o *back-end* será desenvolvido utilizando Java com o micro *framework* Spring Boot. Um módulo de apoio no lado do servidor poderá ser possível, e para ele utilizaremos Python.

Em relação ao deploy das aplicações, o *front-end* será hospedado na plataforma Vercel, que é primariamente voltada para Javascript, proporcionando uma melhor agilidade de desenvolvimento, enquanto o *back-end* será hospedado no Heroku, que é uma plataforma como serviço de fácil manuseio e que nos permitirá ter um maior foco no desenvolvimento do projeto. Através do Heroku podemos também fazer a utilização do banco de dados PostgreSQL por meio do serviço de apoio Heroku Postgres.

Ademais, se for necessário o armazenamento de objetos como arquivos ou imagens, utilizaremos a plataforma Cloudinary principalmente por sua fácil integração com a linguagem de programação Java através de bibliotecas.

#### 3.1.1 Diagramas de arquitetura

Os diagramas [Figura 1](#), [Figura 2](#) e [Figura 3](#) ilustram de modo geral a arquitetura pensada para a solução proposta, utilizando das tecnologias já citadas.

### 3.2 Integrações

Nessa seção serão citadas as possíveis integrações que nossa aplicação terá, que foram decididas baseadas em outras aplicações do mercado.



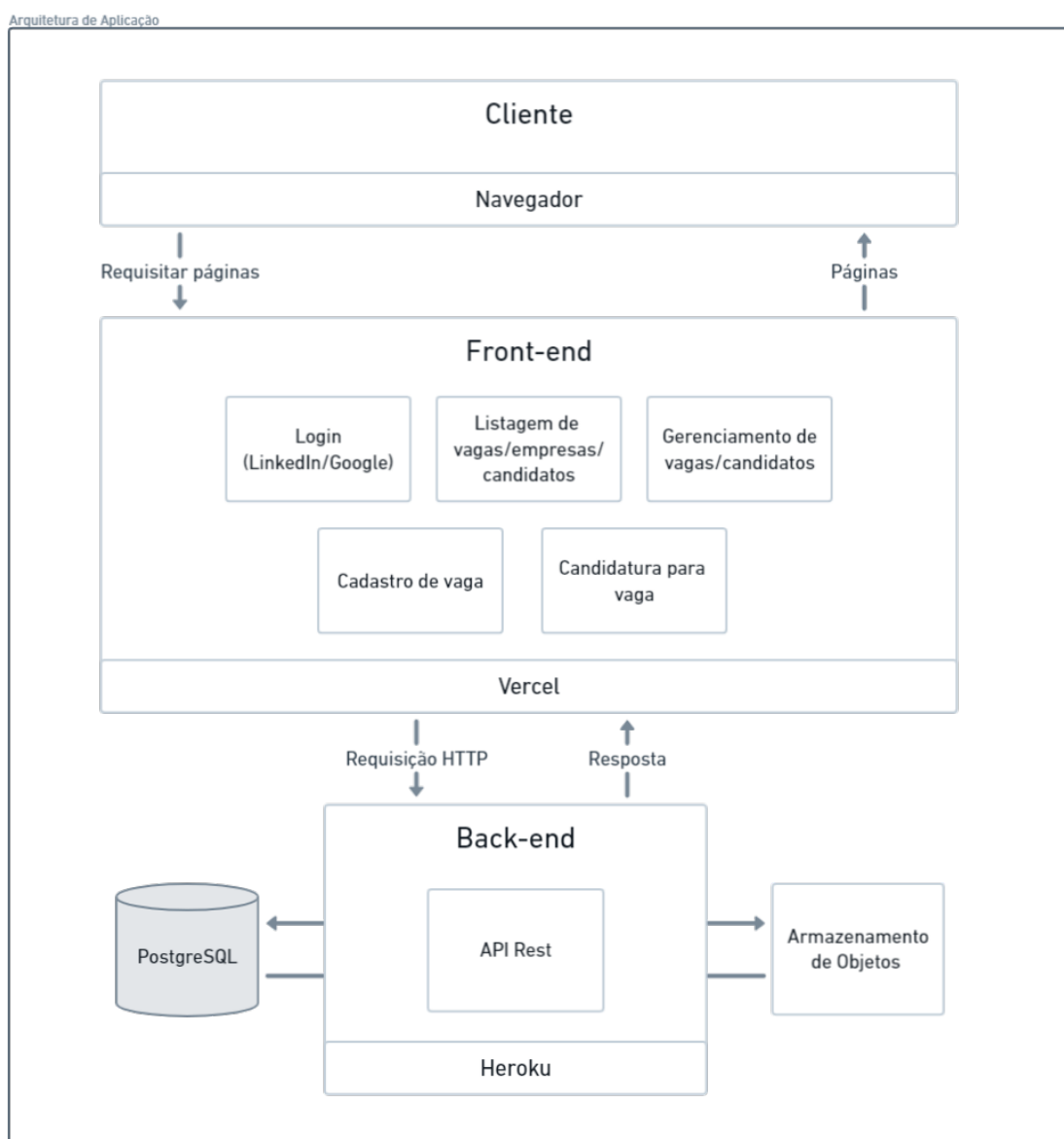
### 3.2.1 Login com o Google e LinkedIn

Pensando na experiência de usuário, nossa aplicação terá a opção do estudante se logar através do *Single Sign-On (SSO)* dessas empresas. Dessa forma, não será necessário digitar a senha toda vez que o usuário for usar nosso *website*, precisando apenas clicar um botão e fazer o login em uma dessas alternativas.

### 3.2.2 Entrar em contato via *Whatsapp*

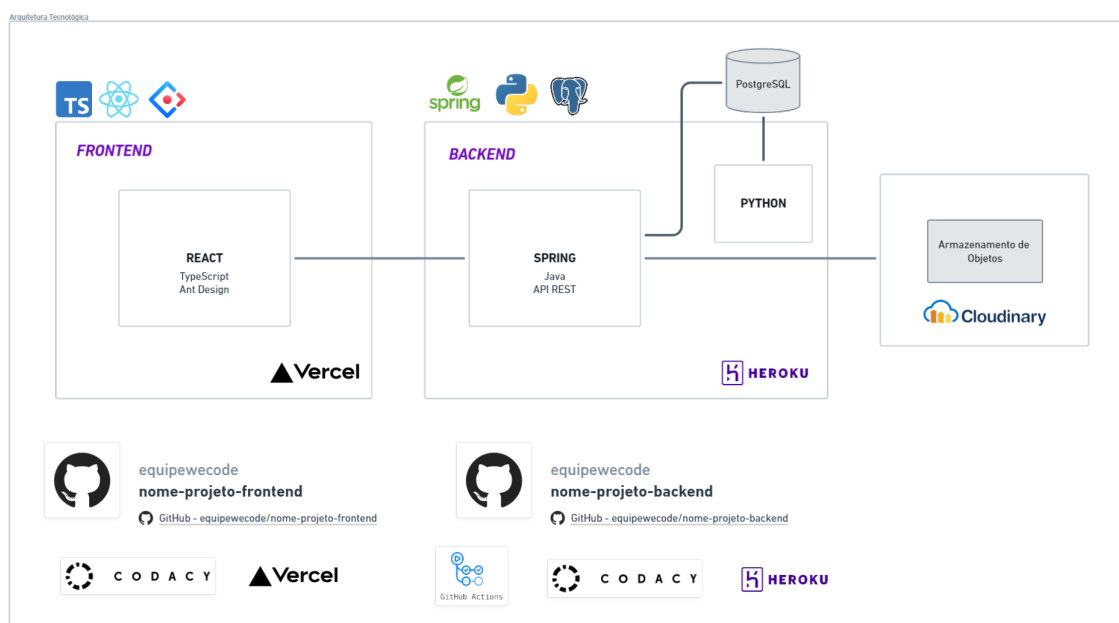
Nossa aplicação terá, também, uma forma da empresa contatar o estudante via *Whatsapp*. Essa integração será feita via *Application Programming Interface (API)* disponibilizada pela própria empresa que mantém o aplicativo (Meta). Dessa forma, com apenas um clique, será possível enviar uma mensagem diretamente ao estudante.

Figura 1 – Arquitetura de Aplicação



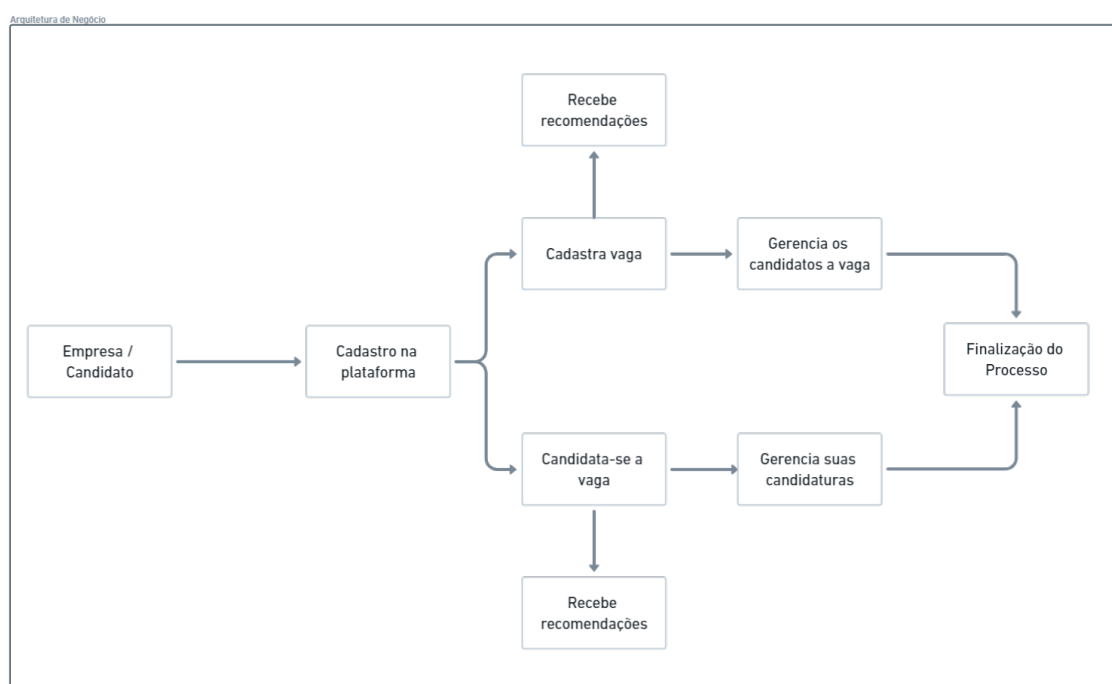
Fonte: Produzido pelos autores utilizando a ferramenta *Whimsical*

Figura 2 – Arquitetura Tecnológica



Fonte: Produzido pelos autores utilizando a ferramenta *Whimsical*

Figura 3 – Arquitetura de Negócios



Fonte: Produzido pelos autores utilizando a ferramenta *Whimsical*

## 4 Análise de Concorrentes

Para a elaboração da proposta, foram verificadas algumas soluções já existentes no mercado. A partir disso, as soluções que mais se assemelham com a proposta foram o *Companhia de Estágios*, *Cia de Talentos* e *Nube*. Com base neste levantamento, podemos observar algumas intersecções de funcionalidades oferecidas. O quadro 1 permite a melhor visualização deste levantamento.

Quadro 1 – Comparação dos aplicativos concorrentes.

Funcionalidades	Companhia de Estágios	Cia de Talentos	Nube	Nosso Projeto
Login/Cadastro.	x	x	x	x
Aplicar em uma vaga.	x	x	x	x
Notificação a cada mudança do status no processo seletivo.			x	x
Recomendação de vagas e/ou empresas aos estudantes de acordo com as suas características.				x
Recomendação de estudantes mais compatíveis com as vagas registradas pelas empresas, de acordo com as características da vaga e da empresa.				x
Simplificação de contato via <i>WhatsApp</i> .				x
Denúncias de vagas incoerentes com a realidade.				x
<i>Feedback</i> de empresas pós-entrevista.				x

Fonte: Os Autores