**PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS**

**PUC Minas Virtual**

**Pós-graduação *Lato Sensu* em Engenharia de *Software***

Projeto Integrado

Relatório Técnico

HealthJobs

Rodrigo Gomes Calixto da Silva

Belo Horizonte

Junho/2023

# Projeto Integrado

**Sumário**

Projeto Integrado 3

1. Cronograma de Trabalho 4

2. Introdução 5

3. Definição Conceitual da Solução 6

3.1 Diagrama de Casos de Uso 6

3.2 Requisitos Funcionais 6

3.3 Requisitos Não-funcionais 7

4. Protótipo Navegável do Sistema 7

5. Diagrama de Classes de Domínio 8

6. Arquitetura da Solução 8

6.1 Padrão Arquitetural 8

6.2 C4 model - Diagrama de Contexto 9

7. Frameworks de Trabalho 10

8. Estrutura Base do Front End 10

9. Modelo Relacional ou Projeto de Banco de Dados NoSQL 10

10. Plano de Testes 11

11. Apropriação de Horas no Projeto 11

12. Código da Aplicação 12

13. Avaliação Retrospectiva 12

13.1 Objetivos Estimados 12

13.2 Objetivos Alcançados 12

13.2 Lições aprendidas 12

14. Referências 13

## Cronograma de Trabalho

|  |  |  |  |
| --- | --- | --- | --- |
| **Datas** | | **Atividade / Tarefa** | **Produto / Resultado** |
| **De** | **Até** |
| 26/02/2023 | 26/02/2023 | 1. Criação do relatório técnico, objetivo do sistema, o problema e objetivos do trabalho. | Relatório técnico |
| 27/02/2023 | 27/02/2023 | 2. Criação do relatório técnico, definição conceitual da solução. | Relatório técnico |
| 07/03/2023 | 07/03/2023 | 3. Criação do Caso de Uso | Relatório técnico |
| 09/03/2023 | 09/03/2023 | 4. Criação dos requisitos funcionais | Relatório técnico |
| 10/03/2023 | 10/03/2023 | 5. Criação dos requisitos não funcionais | Relatório técnico |
| 20/03/2023 | 30/03/2023 | 6. Criação do protótipo navegável | Relatório técnico |
| 10/04/2023 | 10/04/2023 | 7. Gravação do vídeo do protótipo | Relatório técnico |
| 15/04/2023 | 25/04/2023 | 8. Criação do diagrama de classe de domínio | Relatório técnico |
| 25/04/2023 | 30/04/2023 | 8. Definição do Padrão Arquitetural do Projeto e Tecnologias | Relatório técnico |
| 01/05/2023 | 02/05/2023 | 9. Criação do Diagrama de Contexto | Relatório técnico |
| 03/05/2023 | 05/05/2023 | 10. Definição de Frameworks e Estrutura Base do Front End. | Relatório técnico |
| 06/05/2023 | 08/05/2023 | 11. Criação do Modelo Relacional do Banco de Dados. | Relatório técnico |
| 10/05/2023 | 20/05/2023 | 12. Criação do Plano e Relatório de Execução de Testes e Apropriação de Horas. | Relatório técnico |
| 21/05/2023 | 27/05/2023 | 13. Finalização do Projeto com as Descrições Finais. | Relatório técnico |

**Observação: acrescente ou retire linhas, caso seja necessário.**

## Introdução

***2.1. Objetivos***

O objetivo deste projeto é através de uma plataforma, facilitar a empregabilidade dentro da área da saúde, onde profissionais poderão buscar por vagas em hospitais e clínicas da sua região ou regiões distantes.

A plataforma será uma aplicação web, que poderá ser acessada pelo profissional em busca de vagas e pelo empregador que postará as vagas.

Transformar o mercado de contratação da área da saúde, com uma aplicação de fácil utilização, agradável e adaptável a diferentes dispositivos.

***2.2. O Problema***

Contratar profissionais é sempre um processo demorado, burocrático e custoso, em muitos casos tempo de ambas as partes é desperdiçado, seja da pessoa que busca a oportunidade ou do empregador que precisa de alguém qualificado para preencher a vaga.

Nos dias de hoje, estamos em um patamar de evolução tecnológica, seja em internet, servidores e dispositivos, que possuem um alto poder de computação, tornando assim viável a automatização e inovação de muitos processos, a ideia é através de uma aplicação web, agilizar e tronar o mais fácil possível a empregabilidade de profissionais da saúde no Brasil.

***2.3. Objetivo geral***

O objetivo geral deste trabalho é apresentar uma aplicação web, responsiva e performática, direcionada com as boas práticas e padrões da web e da experiência de usuário, para facilitar a empregabilidade dentro da área da saúde.

Objetivos específicos:

* O recrutador poderá postar vagas de trabalho com informações necessárias para que o profissional tome a decisão certa, se pretende se candidatar ou não.
* O profissional poderá buscar por vagas de trabalho, tendo a visualização do que é esperado por ele, seja sobre o trabalho, hospital/clínica em que trabalhará, remuneração, carga horária e benefícios.
* O profissional poderá se candidatar para vagas de trabalho.

## Definição Conceitual da Solução

## Diagrama de Casos de Uso

|  |
| --- |
|  |

## Requisitos Funcionais

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Descrição Resumida** | **Dificuldade (B/M/A)\*** | **Prioridade**  **(B/M/A)\*** |
| RF01 | Candidato deve ser capaz consultar vagas | B | A |
| RF02 | Candidato deve ser capaz filtrar vagas por cidade | B | A |
| RF03 | Candidato deve ser capaz filtrar vagas por cargo | B | A |
| RF04 | Candidato deve ser capaz filtrar vagas por especialidade | B | A |
| RF05 | Candidato deve ser capaz filtrar vagas por regime de trabalho | B | B |
| RF06 | Candidato deve ser capaz de se candidatar para vagas abertas | M | A |
| RF07 | Candidato deve ser capaz de desistir de candidaturas | M | A |
| RF08 | Candidato deve ser capaz de ver candidaturas | A | M |
| RF09 | Candidato deve ser capaz de cadastrar seu currículo | A | M |
| RF10 | Candidato deve ser capaz de visualizar seu currículo | M | B |
| RF11 | Recrutador deve ser capaz de visualizar currículo do candidato | M | B |
| RF12 | Recrutador deve ser capaz de cadastrar empresas | M | B |
| RF13 | Recrutador deve ser capaz de excluir empresas | M | B |
| RF14 | Recrutador deve ser capaz de postar vagas | B | B |
| RF15 | Recrutador deve ser capaz de finalizar vagas | B | B |
| RF16 | Recrutador deve ser capaz de cancelar vagas | B | B |
| RF17 | Recrutador deve ser capaz de aprovar candidato | B | B |
| RF18 | Recrutador deve ser capaz de reprovar candidato | M | M |
| RF19 | Recrutador deve ser capaz de notificar candidato por e-mail | M | M |
| RF20 | Recrutador deve ser capaz de notificar candidato por WhatsApp | M | M |

\* B = Baixa, M = Média, A = Alta.

**Observação: acrescente quantas linhas forem necessárias.**

## Requisitos Não-funcionais

|  |  |  |
| --- | --- | --- |
| **ID** | **Descrição** | **Prioridade**  **B/M/A** |
| RNF01 | O sistema deve apresentar tempo de resposta abaixo de 3 segundos no processamento de 80% das operações de consulta | A |
| RNF02 | O sistema deve estar disponível em qualquer período, do dia e da noite em regime 24/7 | A |
| RNF03 | O sistema Web deve ser responsivo de forma a proporcionar a utilização de qualquer uma de suas funcionalidades em resoluções de 800 pixels até 1080 pixels | M |
| RNF04 | O sistema deve garantir a segurança das senhas dos usuários, criptografando-as ao serem inseridas no banco de dados | A |
| RNF05 | O sistema deve ser fácil de manter e atualizar | M |
| RNF06 | O sistema deve ser capaz de escalar conforme a demanda | B |

**Observação: acrescente quantas linhas forem necessárias.**

## Protótipo Navegável do Sistema

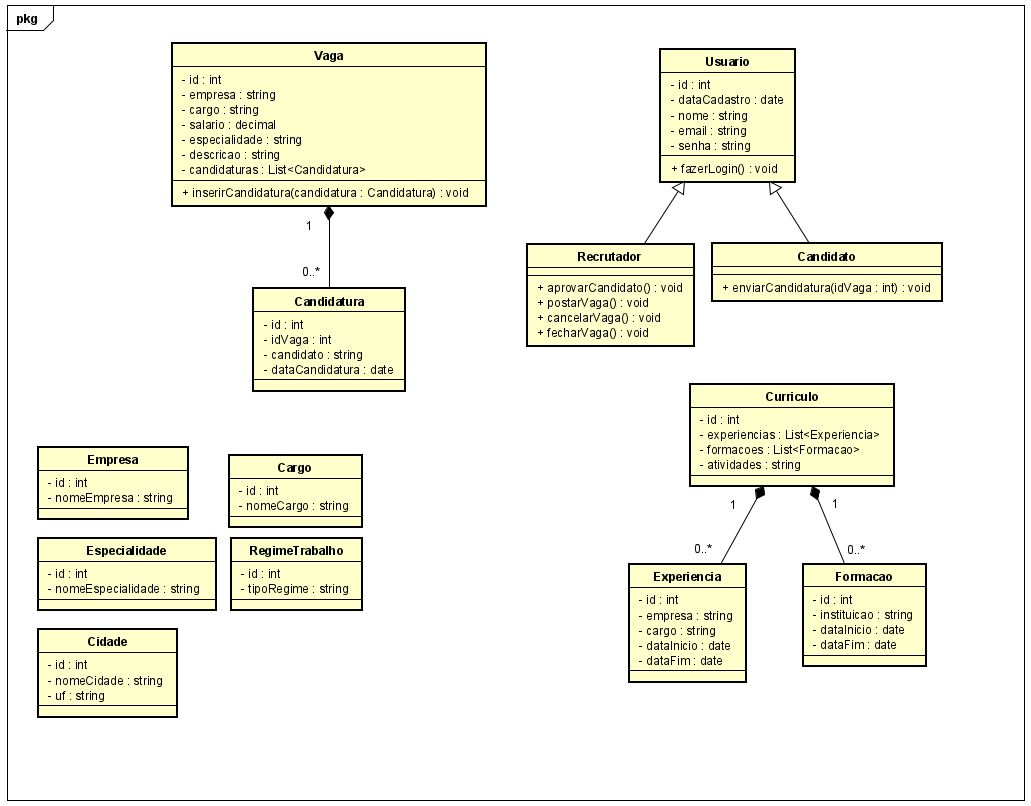
Link do protótipo navegável (Figma): <https://www.figma.com/proto/70v2i8rpZZThpCmMGKE92h/TCC?node-id=507-85&scaling=min-zoom&page-id=0%3A1&starting-point-node-id=507%3A85>

Link do protótipo Wire frames: <https://www.figma.com/file/70v2i8rpZZThpCmMGKE92h/TCC?node-id=0-1&t=aibd8zZTDD4wR0zf-0>

Link do vídeo no youtube: <https://youtu.be/w50k0FSEAZc>

Link do vídeo: <https://github.com/rodrigomes002/tcc-engenharia-software/blob/master/docs/video_prototipo.mp4>

## Diagrama de Classes de Domínio



## Arquitetura da Solução

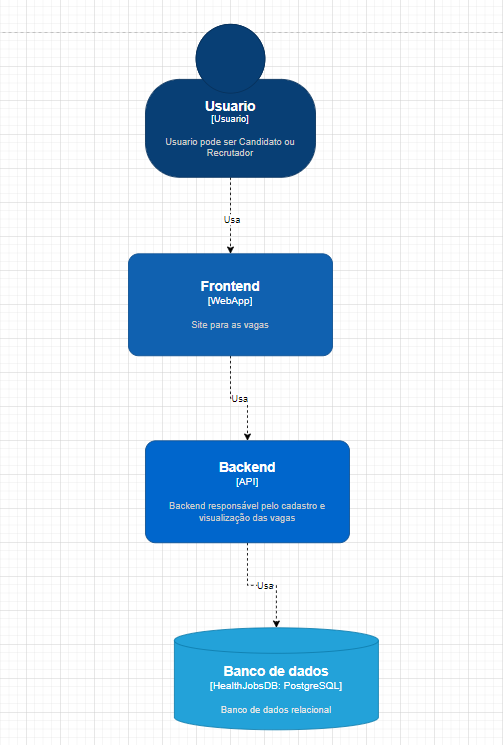
## Padrão Arquitetural

O padrão arquitetural utilizado foi o MVC mesclado com boas práticas e padrões como Repository e filosofias do Domain Driven Design, onde está organizado em camadas, a camada Model corresponde ao Domain, e por ser uma web API não será necessário a camada View e sim teremos Controllers retornando json para o frontend que será uma SPA separada do Backend.

## C4 model - Diagrama de Contexto

Abaixo temos um simples Diagrama de Contexto da aplicação, onde é apresentado a aplicação.

Temos um Usuário que pode ser um Candidato ou um Recrutador, que acessa um WebApp desenvolvido em Angular que é Frontend, que se comunica com o Backend desenvolvido em .NET6, que fornece todos os recursos de uma API, que por sua vez se comunica com a Base de Dados relacional, onde é persistido todas as informações referentes a Aplicação.



## Frameworks de Trabalho

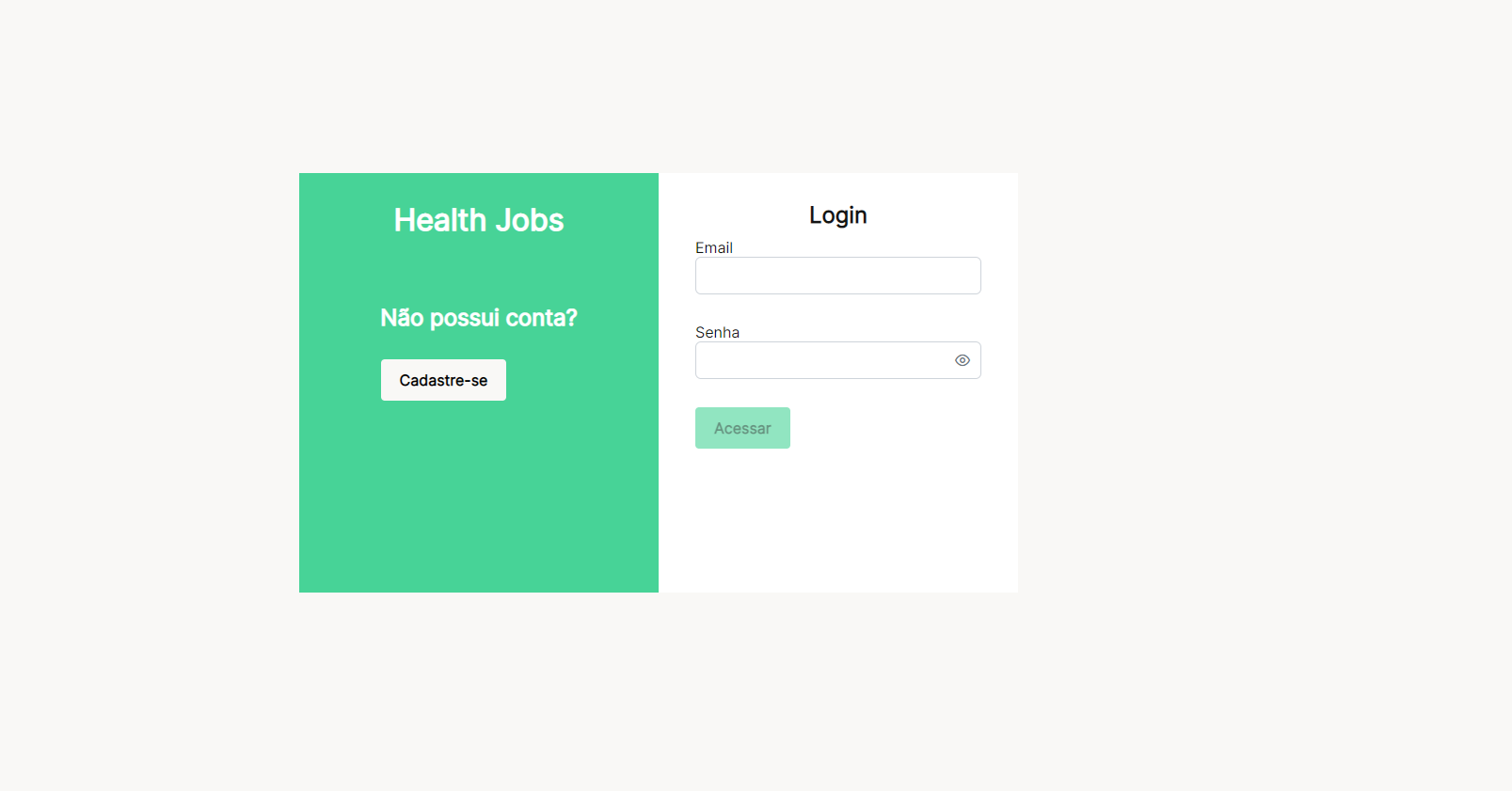
**Frontend**:

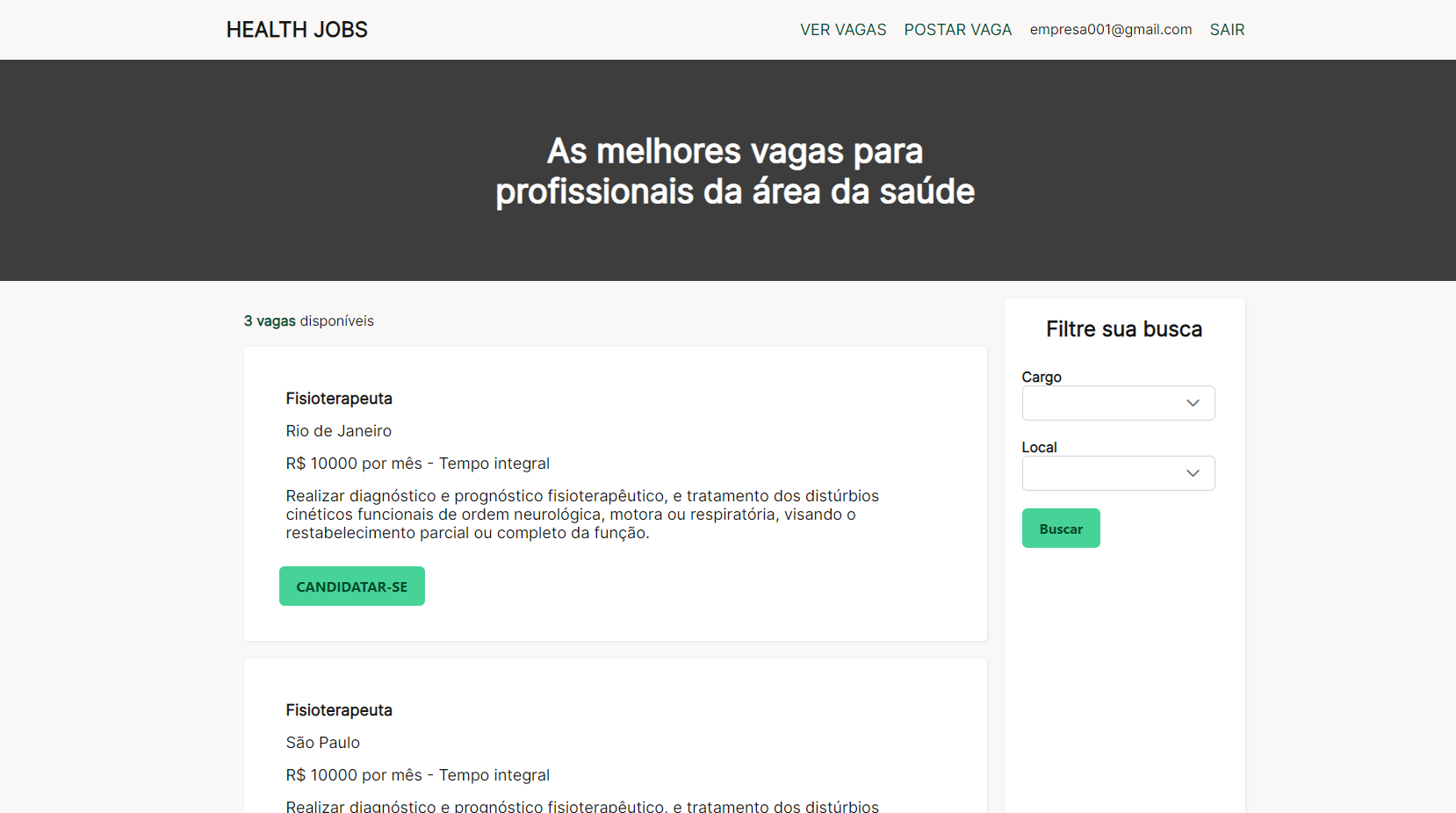
* Angular2+
* NG PrimeFaces
* Typescript
* SCSS

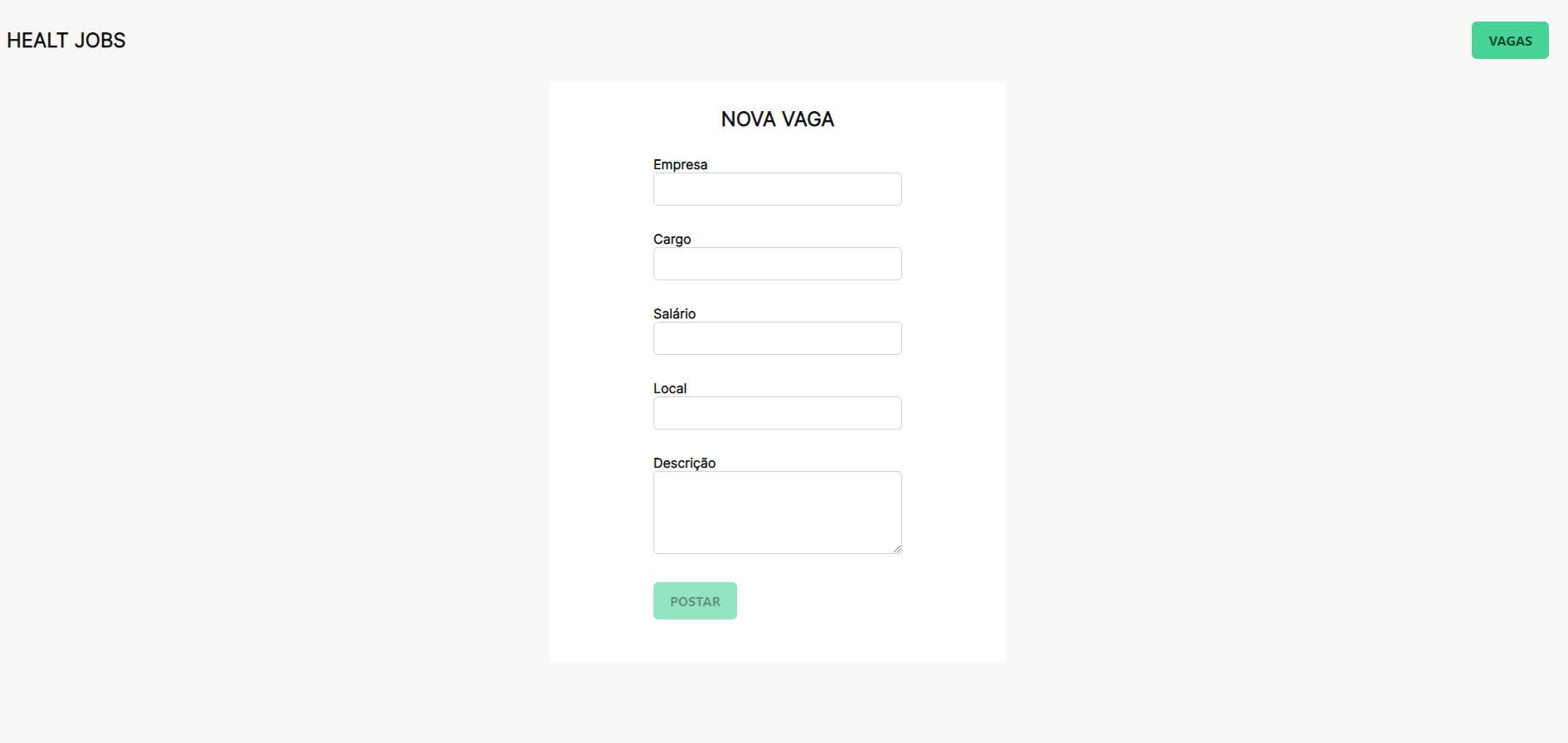
**Backend**:

* AspNet Core Web Api
* C#
* PostgreSQL
* Entity Framework Core

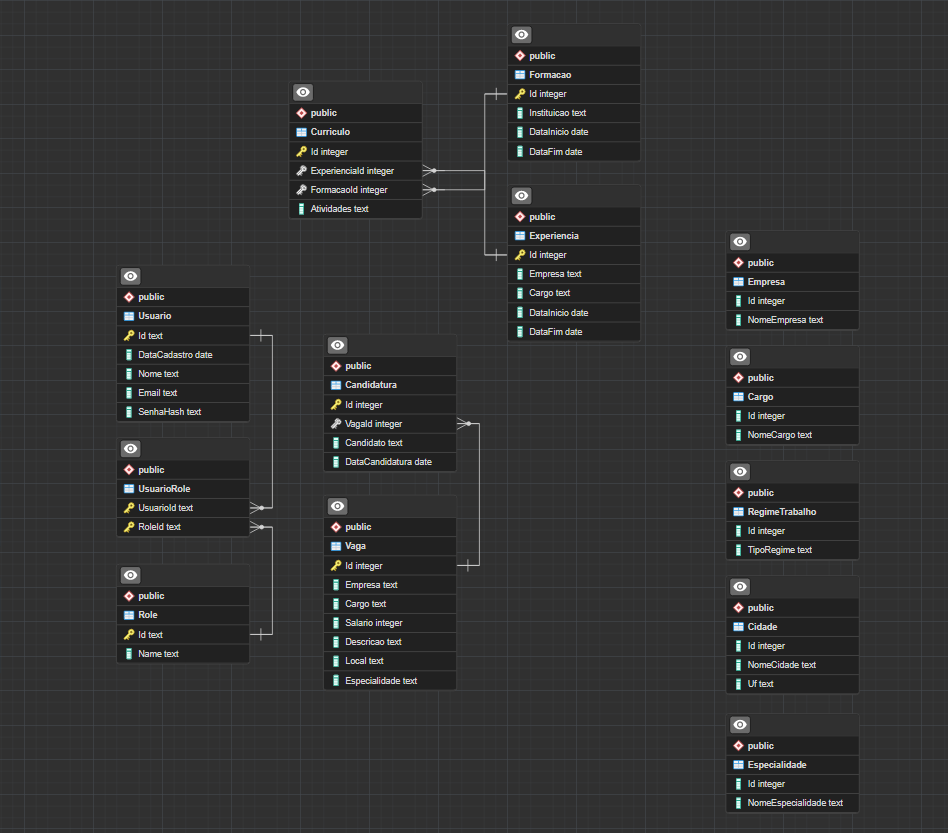
## Estrutura Base do Front End







## Modelo Relacional ou Projeto de Banco de Dados NoSQL



## Plano de Testes

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Número** | **Caso de uso** | **Objetivo do caso de teste** | **Entradas** | **Resultados esperados** |
|  | RF01 | Candidato deve ser capaz consultar vagas | Preenche filtros ou não e clica no botão Buscar | O sistema exibe as vagas abertas |
|  | RF01 | Candidato não consegue consultar vagas | Preenche filtros ou não e clica no botão Buscar | O sistema não exibe as vagas abertas. |
|  | RF06 | Candidato deve ser capaz de se candidatar para vagas abertas | Clicar no botão “CANDIDATAR-SE”. | O Botão de “CANDIDATAR-SE” não é mais exibido e no lugar fica a informação de ‘CANDIDATURA ENVIADA” |
|  | RF06 | Candidato não consegue se candidatar para vagas abertas | Clicar no botão “CANDIDATAR-SE”. | O sistema exibe alguma mensagem de erro durante a candidatura. |
|  | RF14 | Recrutador deve ser capaz de postar vagas | Clicar em “POSTAR VAGA” e preencher o formulário. | O sistema exibe a mensagem “Vaga Postada com Sucesso” . |
|  | RF14 | Recrutador não consegue postar vagas | Clicar em “POSTAR VAGA” e preencher o formulário. | O sistema exibe uma mensagem de erro ao enviar o formulário preenchido. |

## Apropriação de Horas no Projeto

|  |  |  |
| --- | --- | --- |
| **Histórico de apropriação de horas** | | |
| **Data do registro** | **Atividade** | **Quantidade de horas** |
| 26/02/2023 | Criação do relatório técnico, objetivo do sistema, o problema e objetivos do trabalho. | 3 |
| 27/02/2023 | Criação do relatório técnico, definição conceitual da solução. | 1 |
| 07/03/2023 | Criação do Caso de Uso | 5 |
| 09/03/2023 | Criação dos requisitos funcionais | 3 |
| 10/03/2023 | Criação dos requisitos não funcionais | 2 |
| 20/03/2023 | Criação do protótipo navegável | 15 |
| 10/04/2023 | Gravação do vídeo do protótipo | 1 |
| 15/04/2023 | Criação do diagrama de classe de domínio | 5 |
| 25/04/2023 | Definição do Padrão Arquitetural do Projeto e Tecnologias | 20 |
| 01/05/2023 | Criação do Diagrama de Contexto | 5 |
| 03/05/2023 | Definição de Frameworks e Estrutura Base do Front End. | 10 |
| 06/05/2023 | Criação do Modelo Relacional do Banco de Dados. | 8 |
| 10/05/2023 | Criação do Plano e Relatório de Execução de Testes e Apropriação de Horas. | 12 |
| 21/05/2023 | Finalização do Projeto com as Descrições Finais. | 15 |
| 25/05/2023 | Revisão do conteúdo, ajustes no projeto e documentação. | 20 |

## Código da Aplicação

Repositório do GitHub: <https://github.com/rodrigomes002/tcc-engenharia-software>

Endereço do site da aplicação: <https://effervescent-eclair-085083.netlify.app/>

Credenciais de acesso:

Recrutador: recrutador1@email.com / Senha: 123Abc@@@

Candidato: candidato1@email.com / Senha: 123Abc@@@

Link do Vídeo de Apresentação: <https://youtu.be/P0-GisPLJOk>

## Avaliação Retrospectiva

Finalizar este curso me proporcionou muitos momentos de acertos e erros, neste projeto pude aplicar muitas lições aprendidas durante as aulas, cada módulo foi um novo desafio vencido, montar o design e criar as interfaces foram os maiores desafios juntamente de seguir o cronograma proposto.

## Objetivos Estimados

Os objetivos estimados foram montar, cumprir o cronograma e entregar um projeto dentro do tempo planejado.

## 13.2 Objetivos Alcançados

Os objetivos alcançados foram montar, cumprir o cronograma e entregar um projeto com os requisitos solicitados, dentro do tempo planejado e com bom funcionamento.

## Lições aprendidas

|  |  |  |
| --- | --- | --- |
|  | **Retrospectiva (Lições Aprendidas)** | |
|  | **Descrição da Lição** | **Classificação** |
| 1 | Montar e seguir um cronograma | Positiva |
| 2 | Aplicar conhecimentos adiquiridos no curso | Positiva |
| 3 | Melhorar a escrita de documentos | Positiva |
| 4 | Aplicar ideias para resolver problemas reais | Positiva |
|  |  |  |

## Referências

Evans, Eric. **Domain-Driven Design: Atacando as complexidades no coração do software.** Alta Books, 2016

Rafael , André. **Curso** **Web Design para Iniciantes,** Origamid, 2023.

Macoratti. **Curso Web API ASP .NET Core Essencial (.NET 6),** Udemy, 2023.