

Centro Universitário Estácio Juiz de Fora - campus Rio Branco
Análise e Desenvolvimento de Sistemas – Turma 3001

Projeto de Programação Orientada a Objetos

Trabalho apresentado ao professor Anderson Barboza da Cruz, na disciplina Programação Orientada a Objetos em Java, pelas(os) alunos(as), Alana Karine, Davi Nascimento, Gustavo Rodrigues e Jamil Salomão.

Juiz de Fora, 08 de Outubro de 2024.

Sumário

1. Introdução
2. Objetivo
3. Informações da Empresa
4. Desafios do Desenvolvimento
 - 4.1. Descrição do Problema
 - 4.2. Impacto do Problema
 - 4.3. Cenário Atual
5. Justificativas para Criação do Projeto
 - 5.1. Urgência
 - 5.2. Benefícios
 - 5.3. ROI Tecnológico e Comercial
 - 5.4. Riscos de não agir
6. Requisitos Funcionais
7. Requisitos Não Funcionais
8. Padrão de Projeto
 - 8.1. Orientação a Objetos
9. Padrão de Código
10. Integrantes do Projeto e suas Funções

1. Introdução

O desenvolvimento de uma aplicação para o controle de processos seletivos requer uma abordagem estratégica e cuidadosa, combinando práticas de Engenharia de Software com uma compreensão profunda das necessidades de recrutamento. Este projeto descreve a metodologia e as etapas envolvidas na criação de um sistema projetado para otimizar o processo de seleção e contratação de novos funcionários, substituindo o atual uso de planilhas de Excel.

A aplicação proposta permitirá o gerenciamento eficiente de todas as fases do processo seletivo, oferecendo funcionalidades essenciais como o cadastro e acompanhamento de candidatos e controle das etapas do processo. A plataforma proporcionará uma visão centralizada e organizada das candidaturas, simplificando a visualização das informações e melhorando a comunicação entre recrutadores e candidatos.

A interface será intuitiva e amigável, projetada para facilitar o uso por profissionais de RH e garantir uma experiência positiva para todos os usuários envolvidos.

Neste documento, apresentaremos a visão geral do projeto, detalhando os objetivos, as funcionalidades principais e os benefícios esperados com a implementação da nova solução. Acreditamos que a aplicação proposta irá transformar a forma como os processos seletivos são geridos, oferecendo uma solução robusta e eficiente para os desafios enfrentados com as planilhas atuais.

2. Objetivo

O objetivo deste projeto é desenvolver uma aplicação que modernize e simplifique o controle de processos seletivos para contratação de novos funcionários. O sistema será criado desde a concepção inicial da interface até a estruturação dos dados e a implementação das funcionalidades essenciais.

A abordagem incluirá a modelagem eficiente dos dados dos candidatos e a automação das etapas do processo seletivo, assegurando que a aplicação ofereça uma gestão integrada e simplificada. As funcionalidades principais serão projetadas para facilitar o acompanhamento de candidaturas e a comunicação entre recrutadores e candidatos.

A aplicação será desenvolvida com um foco em usabilidade e segurança, proporcionando uma experiência intuitiva e eficiente para os profissionais de RH. O objetivo é garantir a entrega de um software robusto que otimize o processo seletivo, substituindo o uso de planilhas de Excel e resolvendo os desafios associados a elas.

3. Informações da Empresa

- **Nome da Empresa:** WebCertificados;
- **Setor de Atuação:** Venda de Certificados Digital;
- **Produtos / Serviços Oferecidos:** Criação de Certificados Digitais e Suporte / Atendimento ao cliente;
- **Tecnologias Utilizadas:** Softwares de Gestão, Registro e Automação;
- **Linguagens Utilizadas pela Empresa:** Python, Javascript e Typescript.

4. Desafios do Desenvolvimento

Abaixo listaremos os desafios encontrados para o desenvolvimento com sucesso do projeto proposto.

4.1 Descrição do Problema:

Hoje, existe apenas uma planilha Excel para o gerenciamento de processos seletivos, impossibilitando um bom gerenciamento dos dados e uma visão completa.

4.2 Impacto do Problema:

A situação atual do controle dos processos seletivos da empresa pode gerar uma sobrecarga de dados causando uma dificuldade no manuseio ou então erros de inserção na planilha. Além da falta de segurança encontrada em planilhas de Excel, principalmente por possuírem dados sensíveis dos candidatos.

4.3 Cenário Atual:

O atual cenário que se encontra é uma planilha em excel que possui os dados de processos seletivos anteriores bastante desorganizados, com dados antigos e sem utilidade comercial. Além disso as categorias ficam divididas em células relacionadas a cada ano, sendo a principal a que possui as entrevistas, etapas relacionadas ao mês atual, nome e situação do candidato, não sendo muito usual.

5. Justificativas para a Criação do Projeto

Abaixo listaremos as justificativas que levaram a necessidade da do projeto proposto.

5.1 Urgência:

O nível de urgência foi considerado Médio pelo cliente, visto que eles já possuem hoje um controle dos processos em planilhas de Excel. Porém, é de se lembrar que pode ocorrer problemas no atual estado a qualquer momento.

5.2 Benefícios:

Melhor organização dos dados coletados e utilização do sistema com uma interface gráfica mais fácil e amigável para usuários, além de maior segurança e diminuição na ocorrência de erros.

5.3 ROI Tecnológico e Comercial:

A implementação da nossa aplicação para o controle de processos seletivos terá um impacto significativo na eficiência e precisão das contratações. A nova solução proporcionará uma série de benefícios que otimizarão o processo de recrutamento e seleção.

Primeiramente, ao substituir as planilhas de Excel por uma plataforma integrada, a aplicação reduzirá significativamente a ocorrência de erros humanos e proporcionará um gerenciamento mais preciso das candidaturas. Isso não só melhora a assertividade dos dados, mas também economiza tempo e recursos ao automatizar tarefas e simplificar a comunicação entre recrutadores e candidatos.

Além disso, a aplicação contribui para a imagem da empresa, demonstrando um compromisso com a inovação e a responsabilidade social. A modernização do processo seletivo evidencia a adoção de práticas avançadas e eficientes, fortalecendo a reputação da empresa no mercado.

A aplicação adotará padrões rigorosos de Confiabilidade, Integridade e Disponibilidade (CID) na segurança da informação e seguirá as diretrizes da Lei Geral de Proteção de Dados (LGPD), garantimos que as informações registradas na aplicação estarão protegidas contra riscos e acessos não autorizados. Isso proporciona um ambiente seguro e confiável tanto para a empresa, quanto para os candidatos.

Por fim, ao evitar a necessidade de soluções comerciais externas ou o desenvolvimento interno de sistemas complexos, a aplicação representa uma economia substancial de custos futuros. Com uma solução interna bem projetada, a empresa estará mais bem posicionada para gerenciar seus processos de recrutamento de forma eficaz e sustentável, garantindo benefícios a longo prazo.

5.4 Riscos de Não Agir:

Os grandes riscos são com manuseio dos dados ou erros. Além disso, muitos dados podem com o tempo se tornar inúteis para a empresa, visto que não há uma análise dos dados através da planilha, podendo também gerar grandes riscos de vazamento de dados dos candidatos pela falta de segurança que possui nas planilhas utilizadas atualmente

6. Requisitos Funcionais

A partir dos requisitos funcionais, o sistema oferecerá ao usuário uma experiência clara e organizada, integrando funcionalidades essenciais para o gerenciamento dos processos seletivos. Isso tornará mais fácil entender e utilizar o sistema finalizado. As principais funcionalidades do sistema incluirão:

Página Inicial:

- **Sobre:** Aba com descrição a fim de introduzir ao usuário sobre informações básicas da área de atuação e cultura da empresa;
- **Contato:** Acesso a canais de contato com a empresa (telefones, e-mail, WhatsApp, entre outros);
- **Suporte:** Acesso ao suporte do site caso o usuário precise de algum auxílio durante a navegação.

Página de Login/Cadastro:

- **Cadastro de Novos Usuários:** Permite que novos usuários se registrem, criando um perfil com suas credenciais;
- **Login e Logout:** Acesso seguro através de login com usuário e senha, e opção para logout;
- **Recuperação de Senha:** Função para redefinir senha caso o usuário a esqueça.

Página do Usuário:

- **Interface Amigável:** Interface amigável e intuitiva visando facilitar o uso da aplicação e manter o usuário confortável durante a navegação;
- **Menu Burguer:** Será implementado um Menu Burguer na lateral da página abrindo as opções: Exibição de Vagas Abertas, Inscrição da Vaga, Exibição das Inscrições Realizadas, Alteração de Dados, Logout;
- **Exibição de Vagas Abertas:** Exibirá todas as vagas de emprego que estão em aberto para se candidatar;

- **Inscrição da Vaga:** Será exibido um forms solicitando o envio do currículo do candidato em PDF. Após enviado o arquivo em PDF, o candidato receberá um e-mail contendo um link de um forms para que responda uma lista de perguntas elaboradas pelo RH da empresa;
- **Exibição das Inscrições Realizadas:** Exibirá o status, data/hora e etapa atual do processo seletivo de todas as vagas em que o candidato se inscreveu;
- **Alteração de Dados:** Será apresentado ao usuário um forms para colher as informações que ele deseja alterar em seu cadastro;
- **Logout:** Opção extra de logout, dentro do Menu Burguer.

Página do Admin:

- **Interface Amigável:** Interface amigável e intuitiva visando facilitar o uso da aplicação e manter o usuário confortável durante a navegação;
- **Menu Burguer:** Será implementado um Menu Burguer na lateral da página abrindo opções de acesso a outras páginas;
- **Opção 1 do Menu Burguer:** Exibirá opções de filtro (mês, ano, todos, vaga, etapa) e pesquisa para as tabelas abaixo:

Primeira Tabela:

- Opção de edição/visualização dos dados do processo seletivo de cada usuário através de um formulário;

Ex de base para montar:

<https://docs.google.com/spreadsheets/d/1AuZ2i6hxf0TJso1N7god1w18HrdhRIBVV2OACOUjeI/edit?gid=1321250139#gid=1321250139>

- No exemplo acima, será necessário incluir na tabela uma coluna para verificar: Se o forms está respondido, avaliação do forms do candidato (Ideal, Mediano, Ruim), currículo do candidato (ao clicar nessa opção se dará início ao download do arquivo do currículo do servidor para o admin) e qual a vaga que o candidato se inscreveu.

Segunda Tabela:

- Exemplo de exibição das colunas e dados da tabela (id, email, admin, dataCriação, nome, cpf, telefone, endereço, dataNascimento, sexo).

- Tabela com todos os usuários e seus dados, implementar funcionalidade para tornar admin ou voltar usuário normal por meio de um modal;

- **Opção 2 do Menu Burguer:** Exibirá as opções: Exibição de Vagas Abertas, Criação de Novas Vagas (Será enviado um e-mail para os usuários já cadastrados informando sobre a nova vaga disponível), Modificação dos Dados da Vaga, Encerramento da Vaga.
Obs: As vagas criadas não serão deletadas, apenas serão modificados seus status para “Encerradas”, para que não sejam mais exibidas para os candidatos;
- **Opção 3 do Menu Burguer:** Exibição de um forms para alteração de dados;
- **Opção 4 do Menu Burguer:** Opção de logout.

7. Requisitos Não Funcionais

No que engloba os requisitos não funcionais teremos nossas principais ferramentas de trabalho, softwares organizacionais de comunicação e planejamento, juntamente com linguagens de programação a serem utilizadas, banco de dados, a ferramenta para modelar classes e modelo lógico dele, finalizando com o servidor local de alocação para o projeto, também versionando o projeto. Abaixo detalhamos os elementos principais:

Ferramentas de Desenvolvimento:

- **Visual Studio Code:** Utilizado como ambiente de desenvolvimento integrado (IDE) para escrever e depurar o código;
- **Figma:** Ferramenta para design de interfaces e prototipagem rápida;
- **MySQL:** Será utilizado o MySQL na Hostinger para alocação, administração e design do banco de dados;
- **Postman:** Utilizado para testes da API;
- **PHPadmin:** Para criação e testes do Banco de Dados;
- **Pandas:** Biblioteca de código aberto;
- **Jupyter:** Para fazer análise de dados.

Softwares Organizacionais:

- **Discord:** Plataforma para comunicação em tempo real, facilitando reuniões e discussões sobre o projeto. Além disso, será utilizado para planejamento e acompanhamento do progresso do projeto, permitindo a visualização das tarefas e etapas concluídas;
- **GitHub:** Serviço para controle de versão e gerenciamento do código, proporcionando acesso colaborativo e facilitando o versionamento;
- **Microsoft Word:** Para criação da documentação do projeto.

Linguagens de Programação e Frameworks:

- **React/JS e Node.JS:** Para a construção da interface do usuário e lógica de front-end;
- **Python / Flask:** Para o desenvolvimento do back-end, gerenciamento de dados e lógica de negócios;
- **React Bootstrap:** Utilizado para o design responsivo e estilização das interfaces.

Servidores e Banco de Dados:

- **Servidor Flask:** Servidor para API Rest do sistema;
- **React/JS e Node.JS:** Utilizado para o desenvolvimento rápido e eficiente do front-end;
- **Servidor MySQL na Hostinger:** Para alocação do Banco de Dados;
- **Servidor WAMP:** Para realização de testes com MySQL.

Modelagem:

- **App.diagrams:** Ferramenta utilizada para modelar diagramas de classes, bem como os modelos conceitual e lógico do banco de dados.

Links das Ferramentas Utilizadas:

Tecnologias Principais:

- **Flask:** <https://flask.palletsprojects.com/en/3.0.x/>
- **React:** <https://react.dev/>
- **React Bootstrap:** <https://react-bootstrap.netlify.app/>
- **Node.js:** <https://nodejs.org/en>
- **Wamp:** <https://sourceforge.net/projects/wampserver/>
- **App.diagrams:** <https://www.drawio.com/>
- **Postman:** <https://www.postman.com/downloads/>
- **Bootstrap:** <https://getbootstrap.com/>

Tecnologia Extra:

- **Biblioteca de Gráfico:** <https://getbootstrap.com/>
- **React Charts:** <https://react-charts.tanstack.com/docs/overview>
- **Pandas:** <https://pandas.pydata.org/>
- **React Forms:** <https://marmelab.com/react-admin/EditTutorial.html>

8. Padrão do Projeto

O padrão de projeto Model-View-Controller (MVC) organiza a aplicação em três componentes principais, separando a lógica de negócios da interface do usuário e facilitando a manutenção e a escalabilidade do sistema. Essa estrutura será aplicada tanto no front-end, utilizando HTML, CSS e JavaScript/React, quanto no back-end, com Python/Flask. A orientação a objetos será um princípio central em todo o desenvolvimento, garantindo modularidade e reutilização de código.

Model:

Model é responsável pela lógica de dados e regras de negócio. Ele define como os dados são estruturados, manipulados e armazenados. No back-end, Python/Flask será usado para gerenciar os modelos e interagir com o banco de dados.

View:

View é responsável pela apresentação dos dados e pela interface com o usuário. No front-end, React gerenciará as views, criando componentes reutilizáveis para a interface gráfica.

Controller:

Controller atua como intermediário entre o Model e a View, processando entradas, atualizando modelos, e retornando respostas. Ele coordena as interações entre o usuário e o sistema.

8.1. Orientação a Objetos:

Todo o projeto será desenvolvido utilizando **orientação a objetos (OO)**. Isso significa que tanto no front-end quanto no back-end, as funcionalidades serão implementadas em classes e objetos, promovendo encapsulamento, modularidade e reutilização de código.

- **No Back-end:** As entidades do banco de dados serão representadas como classes, com métodos para operações de CRUD/API REST e lógica de negócios;
- **No Front-end:** Componentes de interface e lógica serão estruturados como classes ou componentes funcionais com hooks, permitindo a criação de componentes reutilizáveis e bem-organizados.

9. Padrão de Código

Para garantir a consistência, legibilidade e manutenibilidade do código, adotaremos um conjunto de diretrizes de codificação. Este padrão de código facilitará a colaboração e a revisão do código, além de garantir que todos os desenvolvedores sigam práticas uniformes. A seguir, são apresentadas as diretrizes para a implementação tanto no front-end quanto no back-end.

- **Indentação:** Use 1 tab para indentação;
- **Aspas:** Use aspas duplas " para strings e JSONs, exceto onde aspas simples ' , são necessárias;
- **Nomeação de Variáveis e Funções:** Use snake_case para variáveis e funções, PascalCase para nomes de componentes e classes.
- **Comentários:** Use # para comentários de linha única e para blocos de comentários (Utilizar um # em cada linha). Comente o código para descrever a lógica complexa ou funções importantes.

10. Integrantes do Projeto e suas Funções

- **Alana Karine:** Desenvolvedora Frontend e Estatística;
- **Davi Nascimento:** Desenvolvedor Backend e criação/controle do Banco de Dados;
- **Gustavo Rodrigues:** Realização de Testes e criação/controle da Documentação do Projeto;
- **Jamil Salomão:** Desenvolvedor Frontend e Prototipação do Projeto.