

# Gestão de Recursos Humanos

Disciplina Técnicas de Programação 1

Eduardo Rocha Biagini – 242032273 – dudurbiagini@gmail.com  
Lucas Centurion Netto – 242003781 – lcenturionnetto@gmail.com  
Filipe Araújo Lopes Grillo – 202023541 – lipe.grillo007@gmail.com  
João Vitor Lopes Rocha – 242014041 – jvitorlocha55@gmail.com

Setembro 2025

## Resumo

Este relatório apresenta o desenvolvimento de um software voltado para a gestão de recursos humanos, cuja necessidade surge da demanda das empresas em organizar cadastros de colaboradores, otimizar processos internos de RH e melhorar o atendimento das demandas administrativas. A aplicação busca oferecer maior eficiência na administração de informações de funcionários, reduzindo falhas manuais e agilizando o acesso a dados relevantes. O objetivo principal do trabalho é criar uma solução capaz de centralizar e automatizar as rotinas de gestão de pessoas, como cadastro de colaboradores, controle de cargos e salários, acompanhamento de férias e geração de relatórios, atendendo às necessidades identificadas e proporcionando uma ferramenta prática de apoio ao setor de RH.

## 1 Introdução

Com o avanço da tecnologia e a crescente complexidade dos processos organizacionais, torna-se cada vez mais necessário o uso de ferramentas digitais que auxiliem na gestão eficiente de recursos humanos. A administração de informações relacionadas aos colaboradores, como dados cadastrais, cargos, salários e benefícios, exige precisão, agilidade e organização. Nesse contexto, o desenvolvimento de sistemas específicos para o setor de RH se apresenta como uma solução estratégica para atender às demandas operacionais e administrativas das empresas. Este relatório aborda a criação de um software voltado para a gestão de recursos humanos, destacando sua importância, funcionalidades e os benefícios esperados com sua implementação.

### 1.1 Contexto do Projeto

A gestão de RH é uma área essencial dentro das organizações, responsável por administrar informações e processos relacionados aos colaboradores. Com o crescimento das empresas e o aumento das demandas administrativas, torna-se cada vez mais complicado manter a organização, a precisão e a agilidade no controle de informações quando se utiliza apenas processos manuais ou ferramentas genéricas. Neste caso, surge a necessidade de desenvolver uma solução tecnológica que centralize e automatize as rotinas do setor de RH, permitindo maior controle, segurança e eficiência nos processos internos. O projeto visa atender a essa demanda, oferecendo uma aplicação prática e funcional para facilitar o gerenciamento de pessoas dentro das organizações.

### 1.2 Objetivos de Desenvolvimento

**Objetivo Geral:** Desenvolver um software de gestão de recursos humanos que permita um controle de dados dos colaboradores e que otimize processos administrativos relacionados a área de RH.

**Objetivos Específicos:**

- Criar uma interface intuitiva para cadastro e consulta dos colaboradores.
- Implementar funcionalidades para controle.
- Automatizar a geração de relatórios administrativos e gerenciais.

- Reduzir erros manuais de dados.
- Melhorar o atendimento a demanda.

### 1.3 Requisitos Funcionais

- Cadastro de colaboradores.
- Gerenciamento de cargo e salário.
- Controle de férias.
- Geração de relatórios personalizados.
- Login e permissões para a autenticação de usuários.

### 1.4 Requisitos não Funcionais

- Segurança dos dados.
- Usabilidade e interface intuitiva.
- Performance do sistema.
- Compatibilidade do software.
- Estabilidade do sistema, prevendo crescimento de aumento de dados.

### 1.5 Regras de negócio

- Um colaborador não pode ter mais de um cargo ativo simultaneamente.
- As férias só podem ser marcadas após 12 meses trabalhados.
- Alterações salariais devem ser registradas com antecedência e justificativa.
- Somente usuários com perfil de administrador ou RH podem editar ou excluir cadastros.
- Relatórios financeiros só podem ser acessados por usuários com permissão.
- Não é permitido cadastrar dois colaboradores com o mesmo CPF.

### 1.6 Estrutura do Relatório

As próximas seções deste relatório detalham a análise e o desenvolvimento do software de gestão de recursos humanos implementado.

- Na **Seção 2**, é apresentada a solução proposta, abordando a modelagem e a arquitetura do sistema. Serão detalhados o design, a descrição das classes que estruturam o sistema, a implementação da lógica de negócio e os mecanismos de persistência de dados.
- Na **Seção 3**, são discutidos os resultados e a análise do software finalizado. Esta seção demonstra as funcionalidades implementadas através de telas do sistema, descreve as tecnologias utilizadas e detalha como técnicas de programação orientada a objetos foram aplicadas para garantir a qualidade da solução.
- Por fim, a **Seção 4** apresenta a conclusão, sintetizando a relevância do sistema de RH desenvolvido e seu impacto no contexto proposto, além de apontar possíveis melhorias e evoluções futuras.

## 2 Solução Proposta

### 2.1 Estrutura Inicial

A estrutura do projeto foi organizada para separar o código-fonte da lógica de negócio dos arquivos de interface do usuário. As classes Java foram alocadas no diretório `src/main/java/grupo/trabalho`, enquanto os arquivos FXML foram mantidos em `src/main/java/resources/grupo/trabalho`.

## 2.2 Design e Modelagem

O design do sistema foi orientado a objetos, buscando representar as entidades do domínio de RH de forma clara. A modelagem conceitual foi a base para a implementação das classes, focando em suas responsabilidades e relacionamentos.

### 2.2.1 Descrição das Classes

O sistema foi modularizado em diversas classes com responsabilidades bem definidas, seguindo o padrão Model-View-Controller (MVC).

- **HelloApplication:** Classe principal que inicia a aplicação JavaFX.
- **HelloController:** Controla a tela de login.
- **MainController:** Gerencia o menu principal e a navegação entre módulos.
- **AdmController, FinanceiroController, CandidaturaController e RecrutamentoController:** Controladores das principais subtelas do projeto.
- **Componentes JavaFX:** Utilização de classes como **AnchorPane** para estruturar os elementos visuais.

## 2.3 Implementação da Lógica do Software

A lógica de negócio foi implementada nos métodos das classes controladoras. Para este protótipo, a persistência de dados foi simulada em memória, utilizando coleções Java. O tratamento de exceções foi implementado para lidar com entradas inválidas, exibindo alertas para guiar o usuário.

## 2.4 Integração e navegabilidade

A integração entre a interface gráfica (View) e a lógica de negócio (Controller) é realizada pelo JavaFX através do **FXMLLoader**. A navegabilidade do sistema é garantida por métodos que gerenciam a transição entre as janelas (Stage), proporcionando uma experiência de usuário coesa.

# 3 Resultados e Discussão

## 3.1 Tecnologia Utilizadas

- **Framework:** JavaFX para a construção da interface gráfica.
- **IDE:** IntelliJ IDEA.
- **Ferramenta de UI:** Scene Builder para projetar os arquivos FXML.
- **Controle de Versão:** GitHub.

## 3.2 Demonstração da Solução

A seguir, são apresentadas as principais telas do protótipo funcional, demonstrando o fluxo de navegação do sistema de RH.



Figura 1: Tela inicial de Login do sistema.



Figura 2: Menu Principal, ponto de acesso para os módulos do sistema.

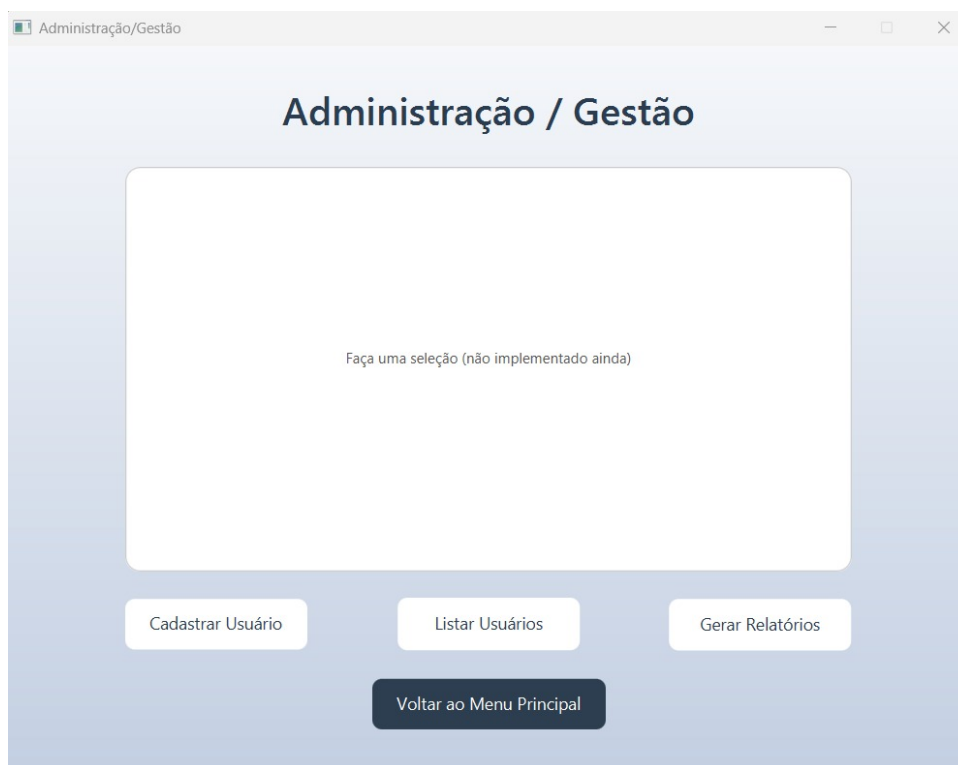


Figura 3: Módulo de Administração / Gestão.



Figura 4: Módulo de Candidatura.



Figura 5: Módulo de Recrutamento.



Figura 6: Módulo Financeiro.

### 3.3 Técnicas de programação Aplicadas

A base do projeto foi construída sobre os princípios de Programação Orientada a Objetos (OOP). Um exemplo prático é a navegação entre telas, que encapsula a lógica de transição de janelas em um método, como demonstrado na Figura 7.

```

@FXML @Eduardo Rocha
private void goToCandidatura() throws IOException {
    Stage prevStage = (Stage) candidaturaButton.getScene().getWindow();
    prevStage.close();

    Stage candidaturaStage = new Stage();
    FXMLLoader candidaturaFXMLLoader = new FXMLLoader(getClass().getResource("/grupo/trabalho/candidatura-view.fxml"));
    Parent candidaturaRoot = candidaturaFXMLLoader.load();

    CandidaturaController candidaturaController = candidaturaFXMLLoader.getController();
    candidaturaController.setMainController(this);

    Scene scene = new Scene(candidaturaRoot);
    candidaturaStage.setTitle("Candidatura");
    candidaturaStage.setScene(scene);
    candidaturaStage.setResizable(false);
    candidaturaStage.show();
}

```

Figura 7: Exemplo de método para navegação entre telas.

Outra técnica foi a injeção de dependência com a anotação `@FXML`, que vincula os componentes da interface (View) aos objetos no controlador (Controller), permitindo a manipulação dos elementos, como visto na Figura 8.

```

@FXML
Button cadastrarCandidatosButton;

@FXML
Button listarCandidatosButton;

@FXML
Button statusDaCandidaturaButton;

@FXML
Button menuButton;

```

Figura 8: Vinculação de componentes FXML ao controlador via anotação.

### 3.4 Desafios Encontrados e Soluções Adotadas

O principal desafio técnico foi implementar a transição entre telas. A solução foi criar uma lógica de navegação centralizada em métodos específicos que gerenciam o ciclo de vida das janelas (Stage).

### 3.5 Análise Crítica da Qualidade da Solução

É importante ressaltar que este é um protótipo inicial. Atualmente, o protótipo ainda não tem quesito nenhum de segurança, a navegação é básica entre as telas e a confiabilidade é quase nula, pois não há persistência real de dados. Contudo, é um bom começo para duas semanas de projeto, estabelecendo uma base sólida sobre a qual funcionalidades mais robustas podem ser construídas.

## 4 Conclusão e Evolução Futura

Este trabalho resultou no desenvolvimento de um protótipo funcional de um sistema de gestão de RH. Como evoluções futuras, sugere-se:

- Implementar um banco de dados para a persistência de dados.
- Desenvolver um sistema de autenticação e controle de acesso seguro.
- Expandir os módulos existentes com mais funcionalidades.
- Melhorar o tratamento de erros para aumentar a robustez do software.

## Referências

- [1] DEITEL, Paul; DEITEL, Harvey. *Java: Como Programar*. 10<sup>a</sup> ed. São Paulo: Pearson Prentice Hall, 2016.
- [2] MARTIN, Robert C. *Código Limpo: Habilidades Práticas do Agile Software*. Rio de Janeiro: Alta Books, 2009.
- [3] GAMMA, Erich; HELM, Richard; JOHNSON, Ralph; VLISSIDES, John. *Padrões de Projeto: Soluções Reutilizáveis de Software Orientado a Objetos*. Porto Alegre: Bookman, 2000.
- [4] ORACLE. *Getting Started with JavaFX*. Documentação Oficial. Disponível em: <https://openjfx.io/openjfx-docs/>. Acessado em set. 2025.
- [5] BLOCH, Joshua. *Java Efetivo*. 3<sup>a</sup> ed. Rio de Janeiro: Alta Books, 2018.