



Escopo do Projeto Integrador (Backend) - Desafio 2

Grupo:

Camila Lima
João Victor Pereira Andrade
Karina Camargo Araujo
Luan Lazarine
Pamela Oliveira Silva
Vitor Gabriel Teles de Lima

Data:

25/04/2025

NeoRH - Sistema de Gerenciamento de Recursos Humanos

O projeto refere-se a uma solução de gestão destinada ao departamento de Recursos Humanos. O software possibilita o registro, visualização, edição e remoção de dados fundamentais de funcionários, funções e remunerações de maneira prática e eficaz. A plataforma é direcionada ao uso interno da organização, contribuindo para o controle de informações indispensáveis e promovendo maior organização no ambiente corporativo.

Descrição Geral

O NeoRH – Sistema de Gerenciamento de Recursos Humanos tem como propósito oferecer uma solução eficiente para controle das informações essenciais dos colaboradores de uma empresa. O sistema possibilita o registro, visualização, atualização e remoção de dados de funcionários de maneira simples e estruturada.

A aplicação foi idealizada especialmente para empresas de pequeno porte, visando uma gestão mais eficiente e ágil de seus recursos humanos. O objetivo é facilitar o gerenciamento de equipes, inserção e remoção de funcionários e também automatizar

cálculos financeiros referentes a salários, bônus e aumentos, promovendo maior agilidade, segurança e assertividade dos dados.

Público-alvo:

O público-alvo deste sistema são funcionários e gestores de equipes de RH de empresas de todos os portes, buscando agilizar e tornar o sistema mais eficiente.

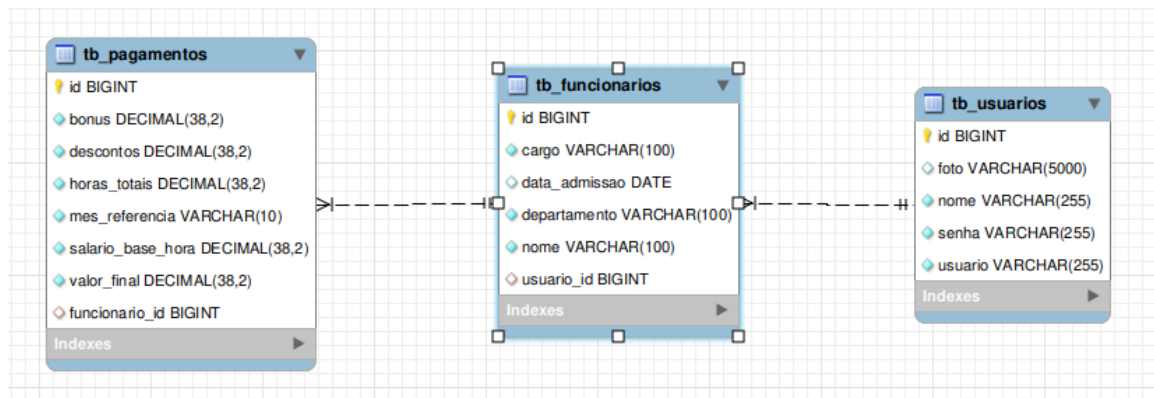
Problema que o projeto visa resolver:

O sistema NeoRH foi desenvolvido para solucionar diversos desafios enfrentados pelas empresas na gestão de recursos humanos. Dentre as principais soluções oferecidas, destacam-se:

- **Centralização e organização de dados:** substitui registros dispersos em arquivos físicos ou planilhas por um banco de dados estruturado, tornando o acesso e o gerenciamento das informações dos colaboradores mais rápido e confiável.
- **Redução de erros e retrabalho:** automatiza processos que antes eram feitos manualmente, minimizando falhas ocasionadas por digitação, perda ou duplicidade de informações.
- **Agilidade nos processos de RH:** Facilita tarefas administrativas diárias, como consulta, edição e exclusão de dados, agilizando a rotina do setor de recursos humanos.
- **Segurança e confidencialidade:** protege informações sensíveis de funcionários, adotando medidas que restringem o acesso a pessoas autorizadas e promovendo maior confiabilidade na guarda dos dados.
- **Facilidade de atualização e manutenção de informações:** permite a alteração dos dados dos colaboradores de forma simples e imediata, mantendo as informações sempre atualizadas.
- **Acessibilidade:** Oferece acesso rápido às informações a partir de qualquer estação de trabalho interna, tornando o RH mais dinâmico e disponível para toda a empresa.
- **Suporte ao crescimento:** proporciona uma base sólida para expansão do negócio, permitindo que a empresa aumente seu quadro de funcionários sem perder a qualidade na gestão de recursos humanos.

Dessa forma, o NeoRH automatiza e moderniza o setor de RH, proporcionando um ambiente corporativo mais eficiente, seguro e preparado para os desafios do mercado atual.

Entidade e Atributos da Entidade



Banco de dados: db_sistema_rh

Entidade Principal: Funcionário (tb_funcionarios)

Atributos:

- **id**: Identificador único do funcionário
- **nome**: Nome completo do funcionário
- **cargo**: Cargo do funcionário na empresa
- **departamento**: Departamento em que o funcionário atua
- **data_admissao**: Data de admissão do funcionário
- **usuario_id**: Referência ao usuário do sistema associada ao funcionário (relacionamento)

Entidade: Usuário (tb_usuarios)

Atributos:

- **id**: Identificador único do usuário
- **nome**: Nome do usuário do sistema
- **foto**: Foto do usuário
- **senha**: Senha de acesso ao sistema
- **usuario**: Nome de usuário/login

Entidade: Pagamentos (tb_pagamentos)

Atributos:

- **id:** Identificador único do pagamento
 - **bonus:** Valor de bônus do pagamento
 - **descontos:** Valor de descontos do pagamento
 - **horas_totais:** Total de horas consideradas no pagamento
 - **mes_referencia:** Mês de referência do pagamento
 - **salario_base_hora:** Salário base por hora
 - **valor_final:** Valor final do pagamento
 - **funcionario_id:** Referência ao funcionário associado ao pagamento
-

Funcionalidades Principais (CRUD)

O sistema contará com funcionalidades básicas de CRUD (Create, Read, Update, Delete) para as entidades Usuários, Funcionários e Pagamento, permitindo o gerenciamento completo das informações cadastradas.

Entidade Usuário:

- **Criar Usuário:** Cadastrar um novo usuário no sistema com dados como nome, usuário, senha e foto.
- **Autenticar:** Autenticação do usuário criando uma chave de acesso ao sistema.
- **Listar Usuários:** Exibir todos os usuários cadastrados, com opção de filtro por ID.
- **Atualizar Usuários:** Editar os dados de um paciente já existente.

Entidade Funcionários:

- **Criar Funcionário:** Cadastrar um novo funcionário com informações como nome, cargo, departamento.
- **Listar Funcionários:** Exibir todos os funcionários cadastrados, com possibilidade de busca por nome e ID.
- **Atualizar Funcionário:** Alterar dados de um funcionário registrado.
- **Excluir Funcionário:** Deletar o cadastro de um funcionário da base de dados.

Entidade Pagamentos:

- **Criar Pagamento:** Cadastrar um novo pagamento com informações como descrição, salário base por hora, mês de referência, horas totais, bônus, descontos, valor final.
- **Listar Pagamentos:** Exibir todos os pagamentos cadastrados, com possibilidade de busca por mês de referência e ID.
- **Atualizar Pagamentos:** Alterar dados de um pagamento registrado.
- **Excluir Pagamentos:** Deletar pagamento da base de dados.

Essas funcionalidades serão acessadas por meio de endpoints RESTful, testados com o Insomnia, garantindo o funcionamento adequado da API.

Tecnologias Utilizadas (Banco de Dados e Backend)

O projeto foi desenvolvido em **Java**, utilizando o framework **Spring Boot**, que facilita a criação de aplicações web com uma configuração simples e organizada. Ele permite uma estrutura baseada em camadas (Controller, Service e Repository) e integração fácil com o banco de dados, além do gerenciamento de dependências com o **Maven**.

O banco de dados utilizado será o **MySQL**, um sistema relacional amplamente usado, confiável e compatível com o ecossistema Java.

Para testar as requisições HTTP da aplicação, como criação, edição e exclusão de dados, será utilizada a ferramenta **Insomnia**, que permite simular e validar chamadas à API de forma prática e eficiente.

Além disso, estamos utilizando o **Trello** para o **gerenciamento de tarefas**, organização das entregas e acompanhamento do progresso do projeto em equipe.

Resumo das tecnologias:

- **Linguagem:** Java
- **Framework:** Spring Boot
- **Banco de Dados:** MySQL
- **Ferramenta de Teste:** Insomnia
- **Gerenciador de Dependências:** Maven
- **JPA/Hibernate:** Para mapeamento objeto-relacional (ORM)
- **Gerenciamento de Tarefas:** Trello