

## Projeto MVC

- **Objetivo:** Aplicar os conceitos MVC aprendidos durante as semanas de estudo.
- **Escopo:** Criar um sistema para gerenciar funcionários GFT em WA (Sem alocação).
- **Tecnologias utilizadas:**
  1. Spring Boot.
  2. Spring MVC.
  3. Spring Data JPA.
  4. Thymeleaf.

## Telas da aplicação





- **Home:** <http://localhost:8080/wa>



- Esta tela, deverá possuir um botão que popule a base de dados.

- **Listar Funcionário:** <http://localhost:8080/wa/funcionarios>

The screenshot shows a web application interface for managing employees in WA. The header includes a 'GFT' logo and navigation links: Home, Funcionários, Vagas, Alocação, and Histórico. Below the header is a search bar with a magnifying glass icon and the title 'Funcionários em WA'. To the right of the search bar is a pagination control showing page numbers 1, 2, 3, ..., 20, 21, 22, with page 2 selected. The main content area displays a table with the following columns: Nome, Cargo, Tecnologias, Início WA, and Previsão término WA. The table contains two rows of employee data and three rows of placeholder data (indicated by '...'). Below the table is a button labeled 'Cadastrar Funcionário'.

Nome	Cargo	Tecnologias	Início WA	Previsão término WA
Henrique Vergara	Analista de Sistemas	 	03/11/2020	05/01/2021
Clecio Silva	Analista de Sistemas	 	01/10/2020	03/12/2021
...	...	...	...	...
...	...	...	...	...
...	...	...	...	...

[Cadastrar Funcionário](#)

- Nesta tela, deverá ser exibida a listagem de funcionários que constam em WA.
- Deve se implementar um sistema de busca/filtro.
- Criar um botão que redireciona para uma tela de cadastro de Funcionário.
- Implementar paginação da listagem (desafio bônus).

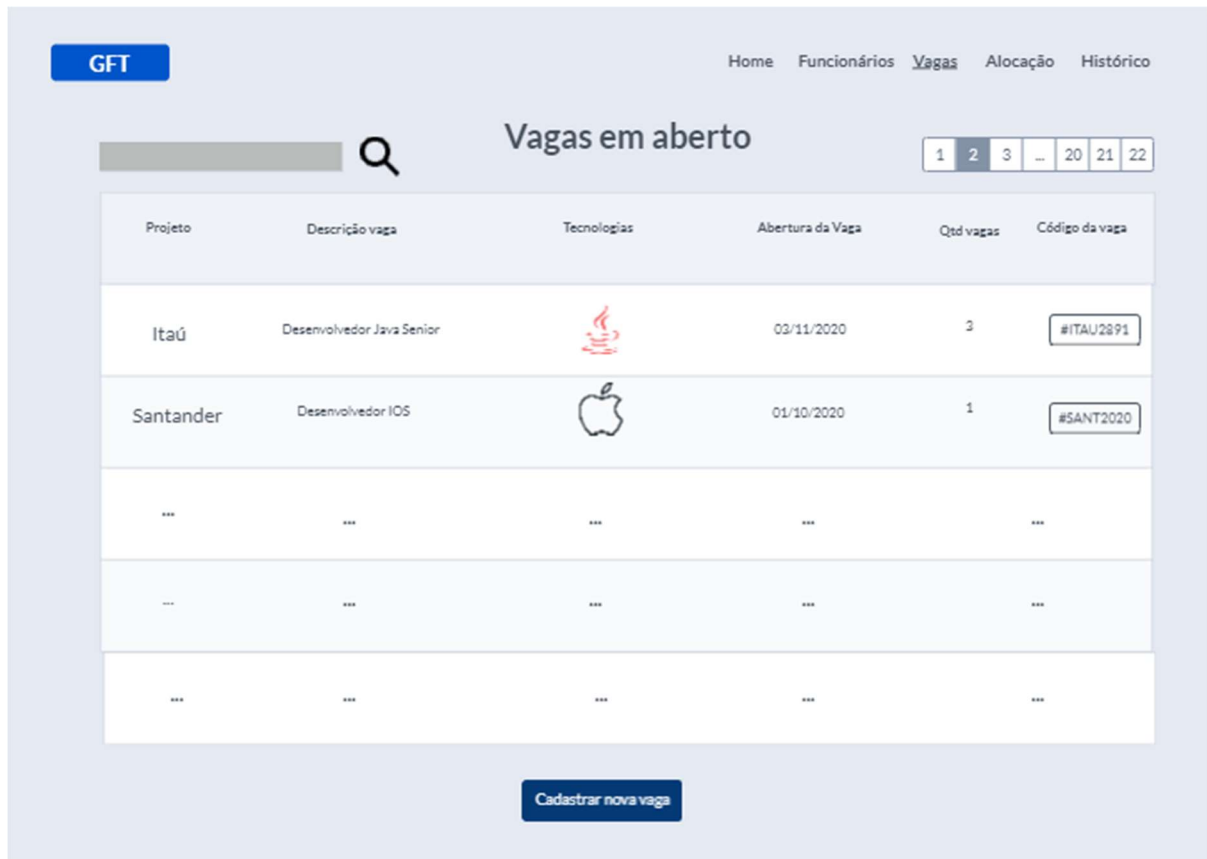
- Caso não possuir funcionário em WA, exibir uma mensagem “Nenhum funcionário cadastrado no sistema.”



- **Cadastrar Funcionário:** <http://localhost:8080/wa/funcionarios/cadastrar>

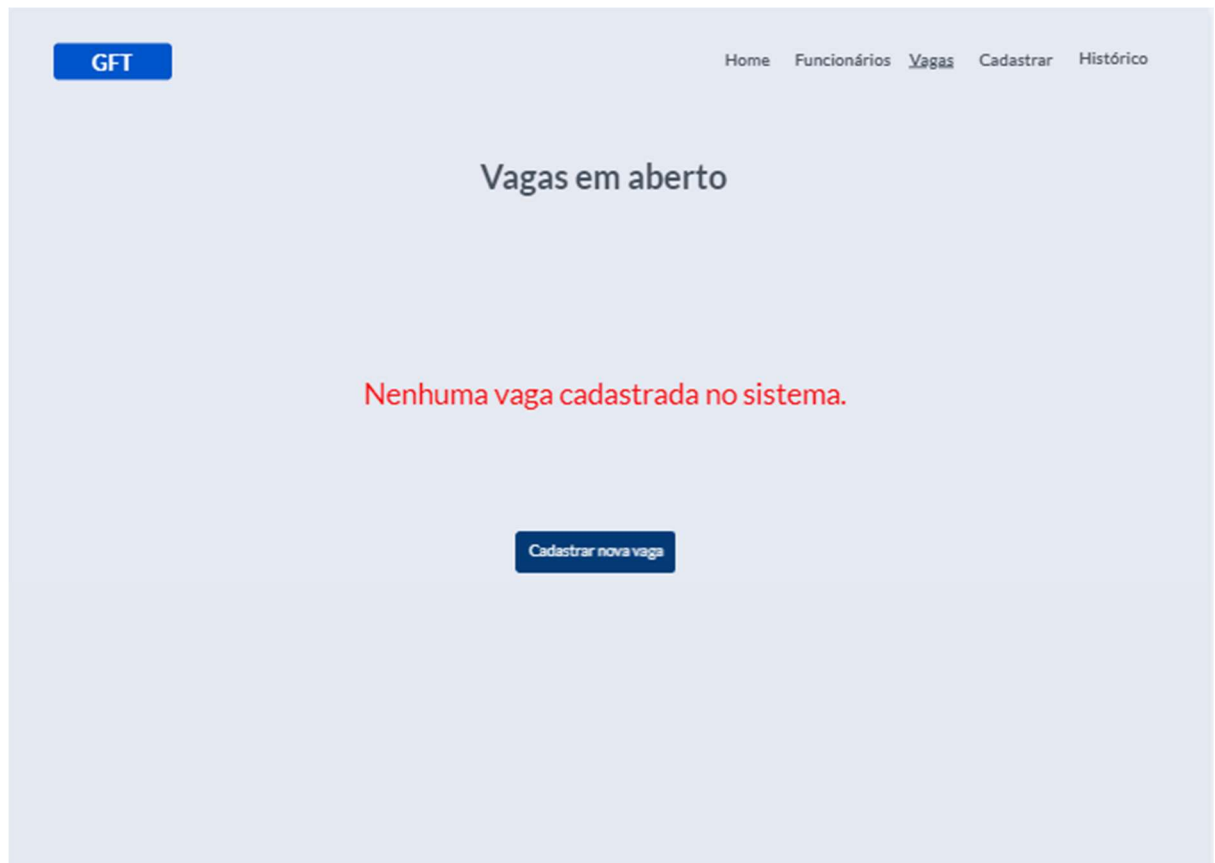
- Nesta tela, deverá ser implementado o cadastro de um Funcionário no sistema de WA.
- Deve se aplicar validações, não aceitar campos em branco ou entrada de dados inválidos.
- Após cadastrar, redirecionar para tela de pesquisa de Funcionários.
- **Obs:** Modelo de dados no final do documento.

- **Listar Vagas:** <http://localhost:8080/wa/vagas>



- Nesta tela, deverá ser exibida a listagem de vagas disponíveis.
- Deve se implementar um sistema de busca/filtro.
- Criar um botão que redireciona para uma tela de cadastro de Vagas.
- Implementar paginação da listagem (desafio bônus).

- Caso não possuir Vagas, exibir uma mensagem “Nenhuma vaga cadastrada no sistema.”

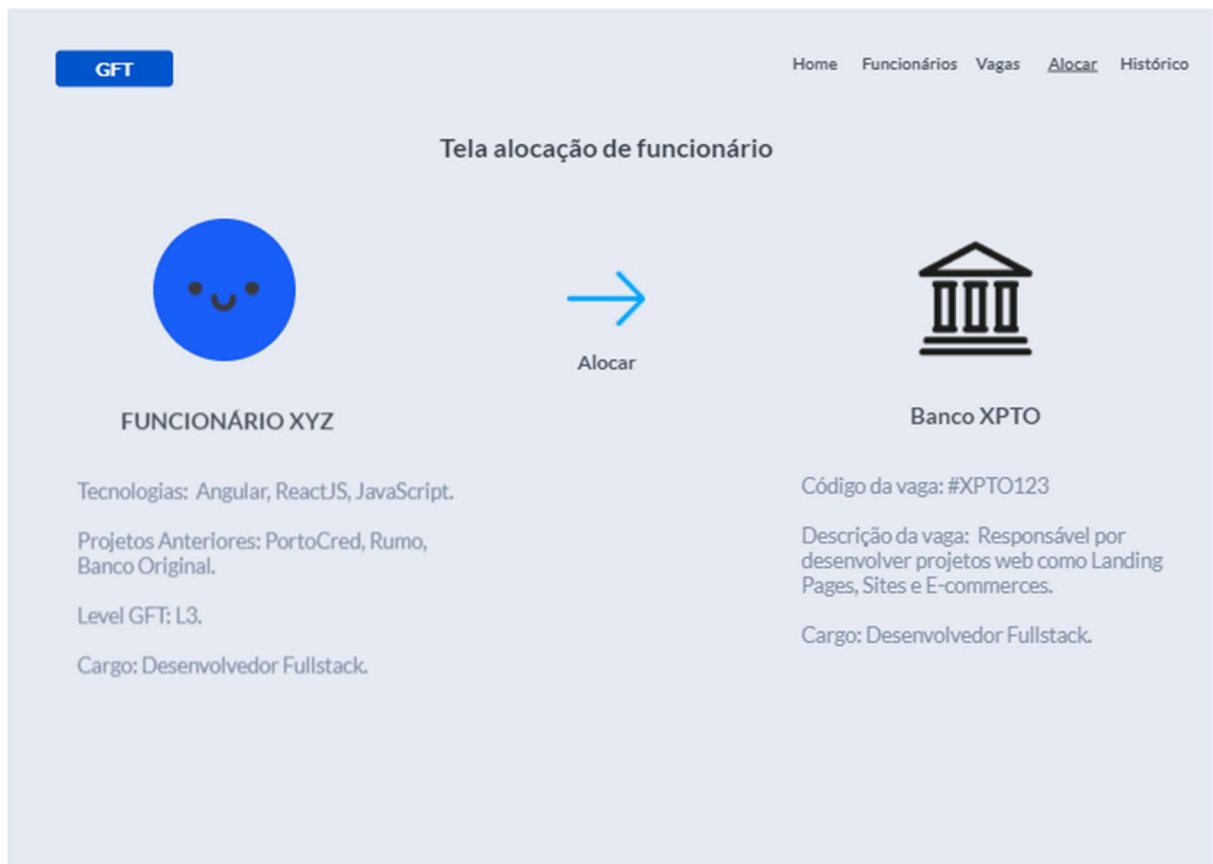


- **Cadastrar Funcionário:** <http://localhost:8080/wa/vagas/cadastrar>

- Nesta tela, deverá ser implementado o cadastro de uma vaga no sistema de WA.
- Deve se aplicar validações, não aceitar campos em branco ou entrada de dados inválidos.
- Após cadastrar, redirecionar para tela de pesquisa de vagas.
- **Obs:** Modelo de dados no final do documento.



- **Alocar em projeto:** <http://localhost:8080/wa/alocar>



- Nesta tela, deverá ser implementada uma lógica para alocar um funcionário em uma vaga (implementar o modelo de dados para esta lógica).
- Após ser alocado, o funcionário deverá ser removido da lista de "Funcionários em WA".
- Caso a vaga possua uma quantidade maior que 1, deverá diminuir após a alocação de um funcionário, caso a vaga tenha apenas quantidade = 1, deverá ser removida da listagem.

- **Histórico:** <http://localhost:8080/wa/historico>

The screenshot shows a web application interface for 'Histórico de WA'. At the top left is a blue button labeled 'GFT'. To its right is a navigation menu with links: 'Home', 'Funcionários', 'Vagas', 'Alocação', and 'Histórico' (which is underlined). Below the navigation menu is a search bar with a magnifying glass icon. The main title 'Histórico de WA' is centered. To the right of the title is a pagination control showing a sequence of numbers: 1, 2, 3, ..., 20, 21, 22. The number 2 is highlighted. Below this is a table with five columns: 'Funcionário', 'Cliente', 'Código da vaga', 'Início WA', and 'Início Alocação'. The first row of data shows 'Funcionário XYZ', 'Banco XPTO', '#XPTO123', '03/11/2020', and '01/12/2020'. The following four rows contain placeholder text '\*\*\*' in each column.

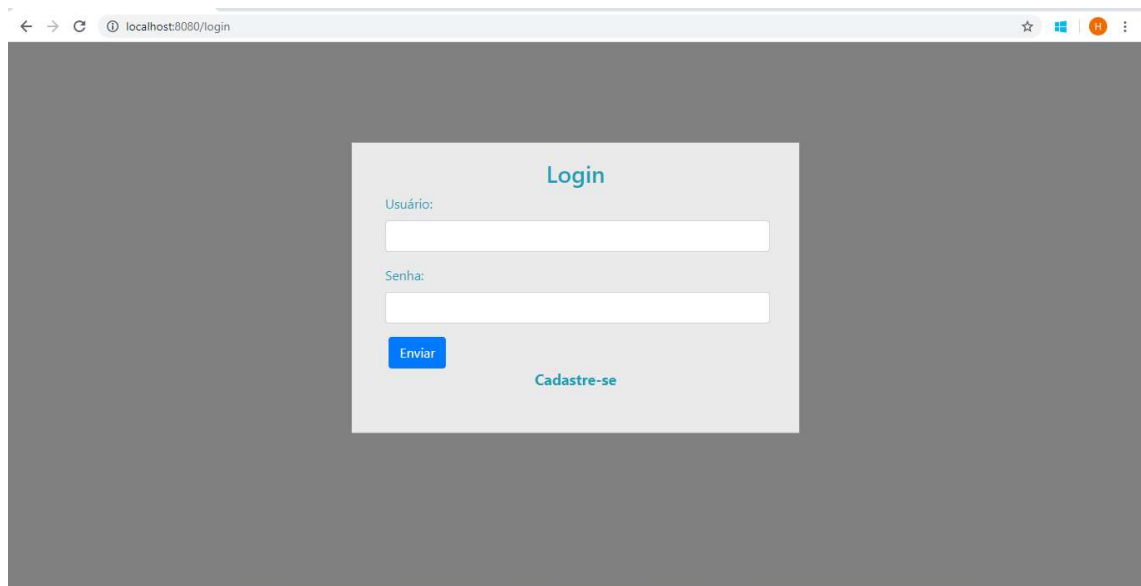
Funcionário	Cliente	Código da vaga	Início WA	Início Alocação
Funcionário XYZ	Banco XPTO	#XPTO123	03/11/2020	01/12/2020
***	***	***	***	***
***	***	***	***	***
***	***	***	***	***
***	***	***	***	***

- Nesta tela, deverá ser exibida a relação de funcionários alocados, detalhando o cliente e a vaga no qual o funcionário foi alocado.
- Deve se implementar um sistema de busca/filtro.
- Implementar paginação da listagem (desafio bônus).

- **Desafio Bônus:**

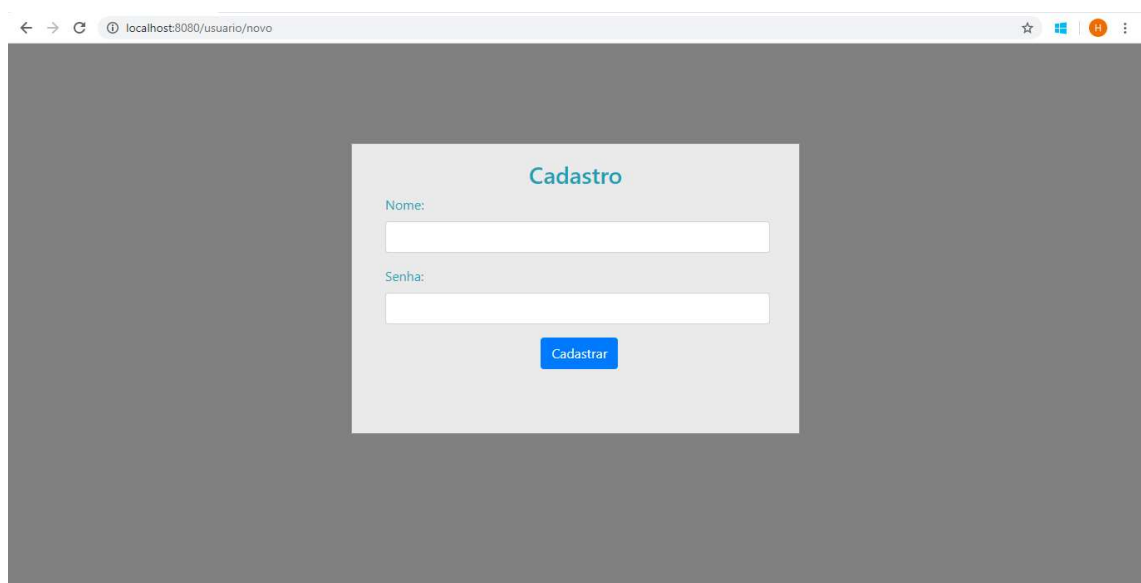
- 1. Implementar login utilizando Spring Security.**

- **Login:**



A screenshot of a web browser window displaying a login form. The browser's address bar shows 'localhost:8080/login'. The form is centered on a dark gray background and is titled 'Login' in blue text. It contains two input fields: 'Usuário:' and 'Senha:'. Below the 'Usuário:' field is a blue button labeled 'Enviar'. To the right of the 'Enviar' button is a blue link labeled 'Cadastre-se'.

- **Cadastro:**



A screenshot of a web browser window displaying a registration form. The browser's address bar shows 'localhost:8080/usuario/novo'. The form is centered on a dark gray background and is titled 'Cadastro' in blue text. It contains two input fields: 'Nome:' and 'Senha:'. Below the 'Senha:' field is a blue button labeled 'Cadastrar'.

## **2. Implementar funcionalidades de acordo com a “role”.**

- **Se o usuário possuir role de “user”, ele apenas poderá visualizar os funcionários e vagas.**
- **Se o usuário possuir role de “admin”, ele terá acesso total, podendo cadastrar, deletar e atualizar dados de funcionários, vagas e visualizar o histórico de alocação.**