

Serviço Nacional de Aprendizagem Comercial do Rio Grande do Sul
Faculdade Senac Porto Alegre
Curso Superior de Tecnologia em Sistemas para Internet

Guilherme Leindecker dos Santos

RELATÓRIO TÉCNICO-CIENTÍFICO DE FRAMEWORKS E APIS

Módulo Emissão Ordem Serviço

Porto Alegre, 2017

SUMÁRIO

1. INTRODUÇÃO.....	3
2. APLICAÇÃO DESENVOLVIDA.....	4
3. SOLUÇÃO.....	5
4. CONSIDERAÇÕES FINAIS.....	6
Referências Bibliográficas.....	7

1. INTRODUÇÃO

O trabalho consiste em desenvolver uma solução para emissão de ordens de serviços para uma auto peças localizada em Porto Alegre. A emissão de ordens de serviço online consiste em cadastramento de usuários da aplicação, cadastramento e listagem de clientes e emissão da ordem de serviço prestado pela empresa.

Para o desenvolvimento da aplicação em questão foi utilizado MongoDB, ExpressJS, AngularJS e NodeJS, juntamente com Bootstrap.

2. APLICAÇÃO DESENVOLVIDA

A aplicação consiste em auxiliar a empresa JM Auto Peças a emitir ordens de serviços e conta com módulos de: Login, Cadastramento de usuários, Cadastramento de Clientes e Emissão de Ordens de Serviço.

Funcionalidades:

- **Cadastramento de usuário**, a aplicação possui um módulo para cadastramento de usuários a serem utilizados, tornando assim o usuário final mais independente na gestão de usuários com permissões para acessar o sistema. Os usuários cadastrados são salvos na collection: usuarios no banco de dados.

➡ Cadastrar Usuário

Email

Senha

Cadastrar

Já possui Login? [Login](#)

- **Login**, aplicação possui área reservada por meio de login. A validação do login é feita com base nos usuários previamente cadastrados e persistidos na collection usuarios.

➡ Login

Email

Senha

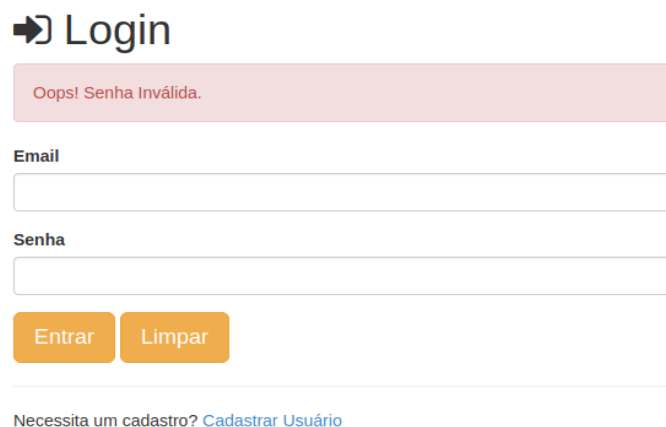
Entrar

Limpar

Necessita um cadastro? [Cadastrar Usuário](#)

O módulo de login possui as seguintes validações:

1. Validação de senha



The image shows a login form titled "Login" with a right-pointing arrow icon. A red error message "Oops! Senha Inválida." is displayed at the top. Below the error message are two input fields: "Email" and "Senha". At the bottom of the form are two orange buttons: "Entrar" and "Limpar". Below the buttons is a link: "Necessita um cadastro? [Cadastrar Usuário](#)".

2. Usuário não localizado



The image shows a login form titled "Login" with a right-pointing arrow icon. A red error message "Usuário não foi localizado." is displayed at the top. Below the error message are two input fields: "Email" and "Senha". At the bottom of the form are two orange buttons: "Entrar" and "Limpar". Below the buttons is a link: "Necessita um cadastro? [Cadastrar Usuário](#)".

- **Página principal**, na index da aplicação possui uma breve introdução explicando o principal escopo do sistema, link para o projeto no GitHub e menu superior de navegação entre os módulos

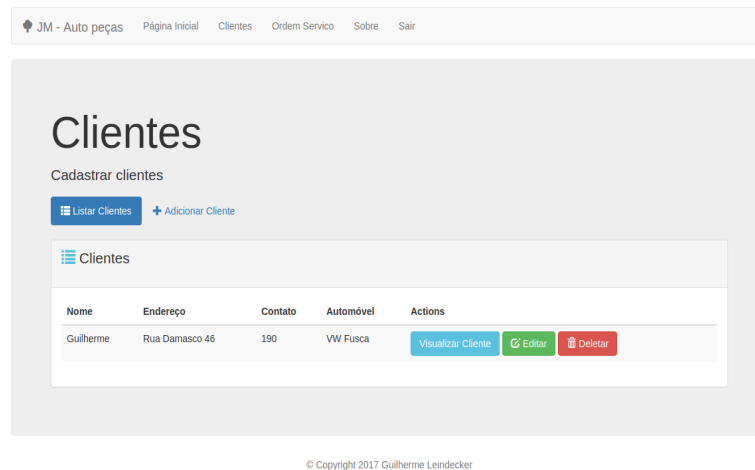


The image shows the top part of a web application. At the top is a navigation bar with the text "JM - Auto peças" and several links: "Página Inicial", "Clientes", "Ordem Serviço", "Sobre", and "Sair". Below the navigation bar is a large gray box containing the title "Emissão de Ordem de Serviço" in a large font. Below the title is a subtitle "Módulo para emissão de ordem de serviço" and a small blue button with the text "Github".

- **Clientes**, o módulo de clientes é dividido em duas partes.

Listagem de clientes, lista todos os clientes cadastrados no sistema, havendo a opção de Visualizar o cadastro do cliente, Edição do cadastro do cliente e exclusão do mesmo.

Listagem Clientes :



JM - Auto peças | Página Inicial | Clientes | Ordem Serviço | Sobre | Sair

Clientes

Cadastrar clientes

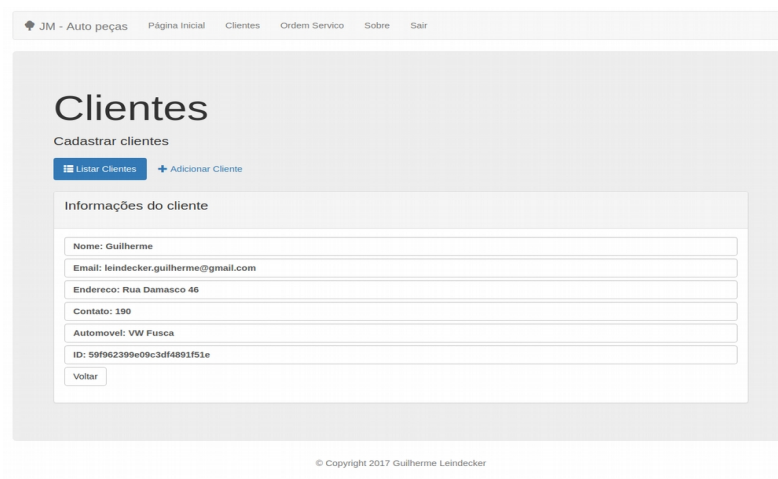
[Listar Clientes](#) [+ Adicionar Cliente](#)

Clientes

Nome	Endereço	Contato	Automóvel	Actions
Guilherme	Rua Damasco 46	190	VW Fusca	Visualizar Cliente Editar Deletar

© Copyright 2017 Guilherme Leindecker

Visualizar Cliente :



JM - Auto peças | Página Inicial | Clientes | Ordem Serviço | Sobre | Sair

Clientes

Cadastrar clientes

[Listar Clientes](#) [+ Adicionar Cliente](#)

Informações do cliente

Nome: Guilherme

Email: leindecker.guilherme@gmail.com

Endereço: Rua Damasco 46

Contato: 190

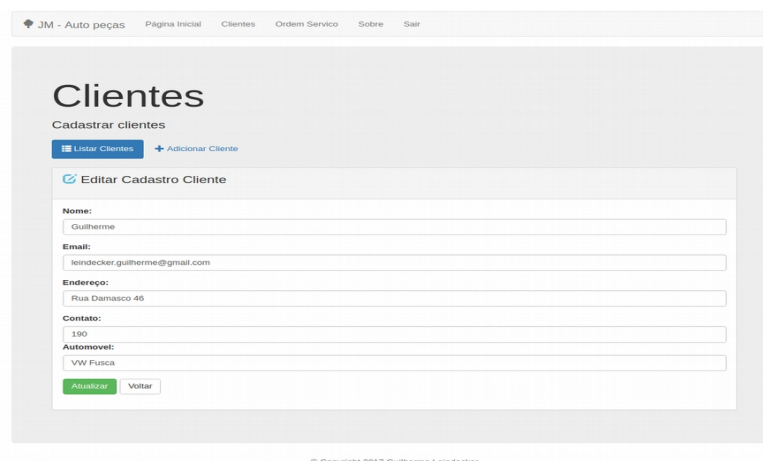
Automóvel: VW Fusca

ID: 59f962399e09c3df4891f51e

[Voltar](#)

© Copyright 2017 Guilherme Leindecker

Editar Cliente :



JM - Auto peças | Página Inicial | Clientes | Ordem Serviço | Sobre | Sair

Clientes

Cadastrar clientes

[Listar Clientes](#) [+ Adicionar Cliente](#)

Editar Cadastro Cliente

Nome: Guilherme

Email: leindecker.guilherme@gmail.com

Endereço: Rua Damasco 46

Contato: 190

Automóvel: VW Fusca

[Atualizar](#) [Voltar](#)

© Copyright 2017 Guilherme Leindecker

- **Ordem de Serviço**, o módulo Ordem de Serviço, bem como o módulo de clientes é dividido em duas partes:

Listagem de Ordem de Serviço :

Nome	Contato	Automóvel	Serviço Realizado	Data Retirada	Total OS	Actions
Guilherme Santos	91521321	Civic Si	troca de óleo	03/11/17	R\$987,33	OS Detalhada Editar Deletar
Guilherme Leindecker	91521321	Civic Si	troca de lâmpada	02/11/17	R\$54,32	OS Detalhada Editar Deletar

© Copyright 2017 Guilherme Leindecker

Visualizar Ordem de Serviço :

Ordem de Serviço Detalhada

Imprimir OS Gerar PDF

Nome: Guilherme Santos
 Email: leindecker.guilherme@gmail.com
 Endereço: Rua Damasco 46
 Contato: 91521321
 Automóvel: Civic Si
 Serviço: troca de óleo
 Data Entrega Veículo: 31/10/17
 Data Retirada Veículo: 03/11/17
 Forma de Pagamento da OS: dinheiro
 Total da OS: R\$987,33
 ID: 59f94a9eb49605d53261eea0
 Voltar

© Copyright 2017 Guilherme Leindecker

Emissão de Ordem de Serviço :

Ordem de Serviço

Emitir OS

Ordem Serviço + Emitir Ordem Serviço

Nome:
 Email:
 Endereço:
 Contato:
 Automóvel :
 Serviço Realizado
 Data Entrega
 Data Retirada
 Forma Pagamento :
 Total Serviço :
 R\$ 0.00
 Salvar Limpar

© Copyright 2017 Guilherme Leindecker

3. SOLUÇÃO

A solução utilizada para o desenvolvimento da aplicação em questão seguiu o padrão do MEAN Stack, da qual utiliza framework front-end (AngularJS), framework back-end (NODEJS e Express, como módulo do NODE) e para a persistência dos dados foi utilizado o MongoDB, banco de dados não-relacional.

Seguiu-se o padrão MVC do AngularJS para implementação das features, foi utilizado o padrão de uso de templates para as views, para assim tornar o código mais legível e ter um ganho em manutenção/refatoração, utilizando o serviço de rotas e criando web api servindo como um middleware para buscar e inserção de dados na base de dados, foi utilizada também a funcionalidade de data binding do próprio angular para buscar os dados do banco de dados de forma rápida e consistente.

Portanto as principais funcionalidades utilizadas na aplicação são:

- ➔ Uso de templates para os arquivos de visualização (Views)
- ➔ MVC suportado pelo AngularJS
- ➔ Uso de rotas
- ➔ Uso de WebApi
- ➔ Injeção de dependências

4. CONSIDERAÇÕES FINAIS

Nesse trabalho foram utilizadas tecnologias de ponta para o desenvolvimento web (MEAN Stack). Grandes empresas estão adotando o Javascript como sua principal linguagem pois nos últimos anos essa tecnologia teve um crescimento muito grande tanto para o desenvolvimento front-end como back-end.

Não obtive muitas dificuldades para localizar material de apoio na internet, visto que como falado acima essas tecnologias estão cada vez crescendo e tendo mais adeptos.

REFERÊNCIAS BIBLIOGRÁFICAS

- MongoDB: <http://www.mongodb.org/>
- ExpressJS: <http://expressjs.com/>
- AngularJS: <https://angularjs.org/>
- NodeJS: <https://nodejs.org/en/>

Meu projeto no Github: <https://github.com/leindecker/ordemServico>