



UNIVERSIDADE FEDERAL RURAL DO SEMI-ÁRIDO  
CENTRO MULTIDISCIPLINAR DE PAU DOS FERROS  
BACHARELADO EM ENGENHARIA DE COMPUTAÇÃO

**KAROLAYNE SANTOS DE AZEVEDO**

## **Relatório de Estágio Supervisionado**

Pau dos Ferros-RN

2019

**KAROLAYNE SANTOS DE AZEVEDO**

# **Relatório de Estágio Supervisionado**

Relatório apresentado a Universidade Federal Rural do Semi-Árido – UFERSA, *Campus* Pau dos Ferros, para a obtenção do título de Bacharel em Engenharia de Computação.

Orientador: Reudismam Rolim de Sousa

Pau dos Ferros-RN

2019

# Relatório de Estágio Supervisionado

Relatório apresentado a Universidade Federal Rural do Semi-Árido – UFERSA, *Campus* Pau dos Ferros, para a obtenção do título de Bacharel em Engenharia de Computação.

APROVADO EM \_\_\_/\_\_\_/\_\_\_\_\_

## BANCA EXAMINADORA

---

Prof. (Orientador)

---

Prof. (1º Membro)

---

Prof. (2º Membro)

## **Resumo**

Neste trabalho são descritas as atividades desenvolvidas na empresa Soirtec localizada na cidade de Pau dos Ferros – RN com relação ao estágio supervisionado obrigatório, requisito parcial para obtenção do título de Bacharel em Engenharia de Computação pela Universidade Federal Rural do Semi-Árido – UFERSA. O estágio foi realizado durante o período de Setembro á Dezembro de 2019, tendo como supervisor Emanuel Lucas Rodrigues Costa e orientador Reudismam Rolim de Sousa. As atividades realizadas buscaram aplicar os conceitos adquiridos durante o curso e foram desenvolvidas utilizando as linguagens HTML, CSS, BOOTSTRAP, JAVASCRIPT e VUEJS, a fim de desenvolver um aplicação web de Gerenciamento de Ordem de Serviço, que será descrita a seguir.

**Palavras-chave:** Estágio, Gerenciamento de Ordem de Serviço, Aplicação Web.

## Sumário

|   |           |
|---|-----------|
| 1.0 INTRODUÇÃO.....   | 6         |
| 2.0 CAMPO DE ESTÁGIO .....  | 7         |
| 3.0 ATIVIDADES REALIZADAS .....   | 8         |
| <b>3.1 Descrição do Projeto Proposta .....</b>  | <b>8</b>  |
| <b>3.2 Levantamento de Requisitos e Construção do Banco de Dados.....</b>                                   | <b>9</b>  |
| <b>3.3 Implementação do <i>Designer</i> Proposto Utilizando Bootstrap e CSS .....</b>                       | <b>10</b> |
| <b>3.3 Construção do Sistema Proposto Utilizando o <i>Framework</i> Vue.js e a Biblioteca Vuetify .....</b> | <b>15</b> |
| <b>3.4 Comunicação entre o <i>Front-end</i> com o <i>Back-end</i> .....</b>                                 | <b>21</b> |
| 4.0 RELAÇÃO TEÓRICA E PRÁTICA .....   | 22        |
| 5.0 CONSIDERAÇÕES FINAIS .....  | 23        |
| 6.0 REFERÊNCIAS BIBLIOGRÁFICAS .....  | 24        |

## **1.0 INTRODUÇÃO**

Um dos grandes obstáculos encontrados pelos alunos de graduação é a dificuldade em aliar a teoria e a prática, adquiridos durante o curso, dentro do campo profissional, por exemplo. Neste sentido, o estágio supervisionado oferece oportunidade de colocar em prática o conhecimento construído nas aulas teóricas e práticas, sob a supervisão de um profissional da área, que irá orientar e corrigir o estagiário em todas as atividades desenvolvidas, como aponta a legislação de estágio em vigor, Lei nº 11.788, de 25 de setembro de 2008, na qual apresenta base jurídica para que o estágio permaneça vinculado ao processo educativo evitando que esta atividade curricular se torne uma forma de precarização nas relações de trabalho [1, 2].

O estágio supervisionado foi realizado na empresa Soirtec Tecnologia e Serviços de Informática, no período de 25/10/2019 até 20/12/2019, de segunda-feira à sexta-feira, nos horários de 18h00min as 22h00min, totalizando 4 horas diárias e 20 horas semanais. A orientação do estágio foi realizada pelo Prof. Dr. Reudismam Rolim de Sousa, por parte da instituição de ensino, e por Emanuel Lucas Rodrigues Costa, Engenheiro de Computação, supervisor por parte da empresa concedente.

Neste sentido, o presente relatório, tem como objetivo, descrever as atividades desempenhadas, durante o período de estágio, buscando alinhar e agregar conhecimentos obtidos ao longo no curso de Engenharia de Computação da UFERSA – Pau dos Ferros. Para tanto, foram desenvolvidas atividades relacionadas ao desenvolvimento de software, utilizando assim conceitos das áreas de Engenharia de Software, Banco de Dados e Programação.

## 2.0 CAMPO DE ESTÁGIO

A Soirtec é uma empresa que oferece os mais diversos serviços e soluções em desenvolvimento de aplicações em Web e Mobile, dispondo de soluções simples como portais e páginas para divulgação até sistema mais complexos como gerenciamentos de clientes, automação comercial.

Sua sede, está localizada no estado de São Paulo, entretanto as atividades de estágio foram realizadas em sua filial na cidade de Pau dos Ferros, localizada na Rua João de Aquino, número 90, Centro. Neste escritório, a empresa conta com cerca de 7 funcionários, nos quais estão divididos entre: desenvolvedores *back-end*, desenvolvedores *front-end*, desenvolvedores *mobile* e *designers*. Pôde-se contar com a supervisão do Engenheiro da Computação Emanuel Lucas Rodrigues Costa durante o período de estágio.

Na filial de Pau dos Ferros, para definir as atividades para os estagiário no período de execução em questão, a empresa aplicou um processo para definir a formação das equipes dado o número de candidatos disponíveis na seleção. Inicialmente, foram selecionados cerca de 21 candidatos para estagiar na empresa. Cada equipe deveria ser formada por em torno de sete pessoas. Assim, os candidatos selecionados, foram divididos em três equipes. Cada equipe trabalhou no desenvolvimento de um sistema com os mesmos requisitos. As equipes deveriam se organizar de forma que cada membro executasse uma função em particular dentro da equipe, tais como *designer*, desenvolvedor *front-end* (lado cliente, que interage com o usuário por meio de interfaces), desenvolvedor *back-end* (lado servidor, que realiza a lógica da aplicação). Diante deste contexto, nossa equipe apresentou dois componentes para o desenvolvimento do *back-end*, três componentes para o desenvolvimento do *front-end* e um componente *mobile*. Vale salientar que dentro da equipe, fui designada a fazer parte da equipe do *front-end*. Esta por sua vez também foi dividida internamente, a

fim de otimizar as tarefas que o *front-end* deveria desempenhar. Assim teríamos um componente responsável por desenvolver os *cruds* da aplicação ( tarefa desempenhada por mim), outro componente ficaria responsável por autenticar e validar os formulários, e outro componente realizaria a estilização da aplicação, bem como a hospedagem do mapa. O sistema proposto, foi destinado ao Controle de Ordem de Serviços. Através desta aplicação, os operadores fariam a gestão e monitoramento de seus prestadores de serviço, por meio de uma aplicação Web e, seus prestadores, teriam acesso às atividades na qual deveriam executar através de uma aplicação *Mobile*.

### **3.0 ATIVIDADES REALIZADAS**

Como mencionado anteriormente, a maior parte das atividades realizadas pela empresa estão voltadas ao desenvolvimento *Mobile* e Web. Neste sentido, foram realizadas atividades voltadas a estes tipos de aplicações, nas quais serão descritos com detalhes a seguir.

#### **3.1 Descrição do Projeto Proposta**

Uma ordem de serviço pode ser descrita como um documento que formaliza um serviço prestado por uma empresa a um cliente. Um sistema de ordem de serviço é um sistema que auxilia na organização da quantidade e a qualidade das ordens de serviço dos seus clientes e funcionários, sendo de extrema importância para que o operador mantenha o controle dos processos, bem como o monitoramento de suas equipes em campo [3].

Neste sentido, o sistema teria que conter, no mínimo, os seguintes requisitos:

- I. Cadastrar prestadores de serviços externos e operadores do sistema.
- II. Cadastrar relatórios dinâmicos onde estes prestadores podem incluir texto e fotos.
- III. Cadastrar atividades realizadas pelos prestadores de serviço, e vincular um ou mais relatórios a estas atividades.
- IV. Monitorar em um mapa as posições dos prestadores de serviço e os serviços já prestados por estes usuários incluindo a rota utilizada.

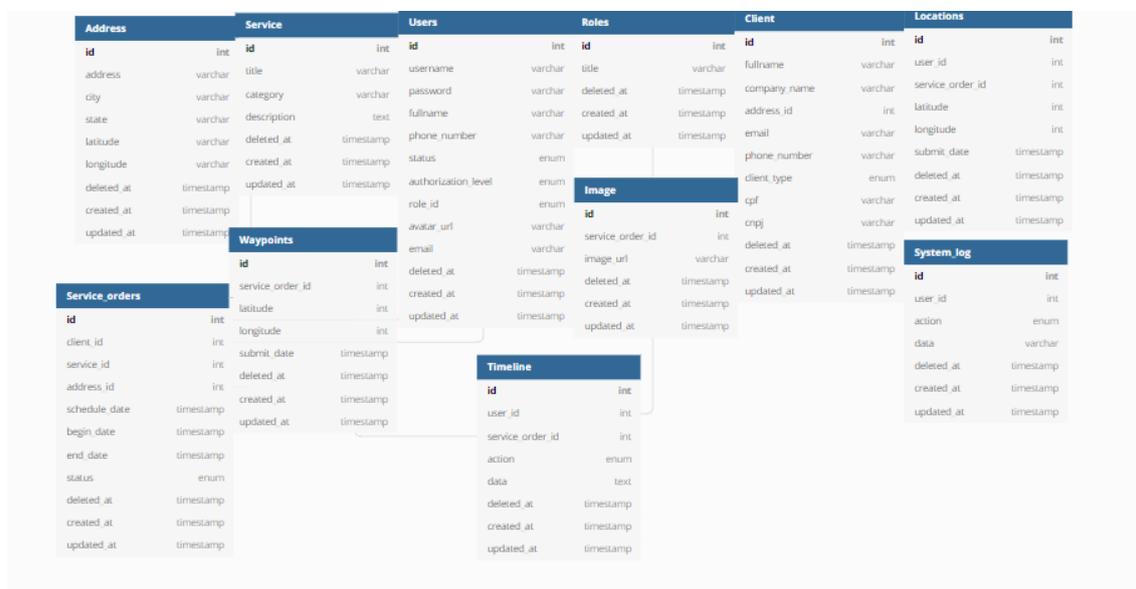
- V. Monitorar em tempo real estado do prestador (em atendimento, deslocamento e disponível para novo serviço).

### 3.2 Levantamento de Requisitos e Construção do Banco de Dados

Antes de iniciar o desenvolvimento do sistema proposto, as duas primeiras semanas do estágio foram dedicadas a realizar tarefas relacionadas a engenharia de software, mais precisamente no levantamento dos requisitos funcionais e não funcionais do sistema, para que os membros da equipe obtivessem uma melhor compreensão do funcionamento do sistema proposto. Paralelamente ao levantamento dos requisitos do sistema, foi realizada a modelagem do banco de dados, com o objetivo de identificar as principais entidades relacionadas ao sistema.

Durante a etapa de modelagem do banco dados, foi criado um modelo lógico do sistema, evidenciando as tabelas que compõem o banco de dados, conforme ilustra a Figura 1.

Figura 1 - Modelo lógico do banco de dados



Fonte: Autor

O sistema oferece a possibilidade de realizar operações de CRUD (*Create, Read, Update e Delete*) de forma a criar, ler, atualizar e deletar itens do banco de dados.

### **3.3 Implementação do *Designer* Proposto Utilizando Bootstrap e CSS**

A segunda atividade solicitada, foi a implementação de uma aplicação que permitisse aos membros da empresa o acesso e monitoramento a documentos e formulários de ordens de serviços, a partir da realização de consultas no sistema interno da empresa, bem como a localização dos prestadores da empresa. O objetivo é aumentar a eficiência com relação a pesquisa de serviços prestado, permitindo a obtenção de documentos de forma simples e rápida.

Vale salientar que, o sistema a ser desenvolvido , não foi destinado a nenhuma empresa específica. Deste modo, deveria-se implementá-lo de maneira genérica, atendendo apenas, os requisitos descritos na seção 3.1.

De modo geral, a aplicação deveria conter dois módulos: Um módulo *Mobile* e um módulo *Web*. O módulo *Mobile*, era destinado aos prestadores de serviço da empresa. A principal função deste módulo seria enviar sua localização como também preencher relatórios de ordens de serviços.

O módulo *Web*, é utilizado pelos operadores da empresa. Estes, seriam responsáveis por acompanhar a localização dos prestadores de serviços, como também realizar cadastros de clientes (físicos ou jurídicos), cadastros de usuários do sistema e cadastro de novas ordens de serviços. Neste caso, os clientes da empresa, não teriam acesso a nenhum módulo.

Para o desenvolvimento do sistema proposto, foi necessário o uso de tecnologias atuais de mercado, os quais não foram o foco durante o curso de graduação. Desta

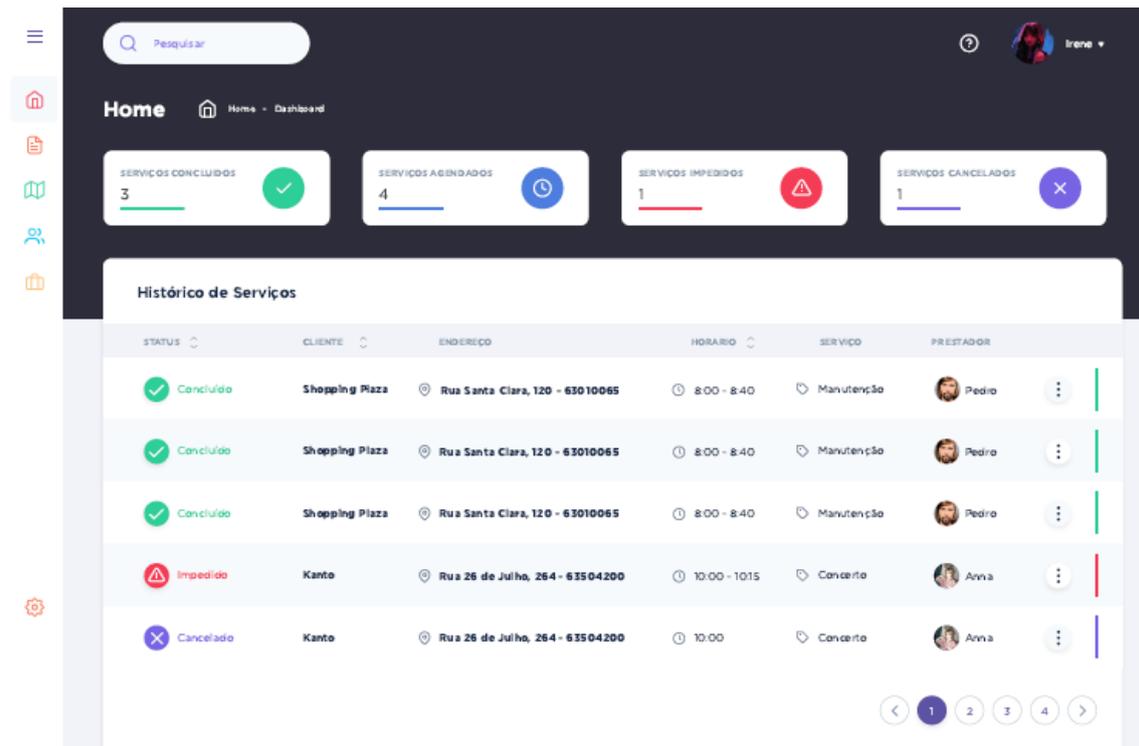
forma, foi necessário um estudo mais detalhado a respeito da linguagem de marcação CSS, do *framework* Bootstrap, e a linguagem JavaScript, a fim de auxiliar nas atividades referentes ao *front-end* da aplicação. Um *framework* pode ser entendido como uma estrutura-base que contém um conjunto de funções e componentes pré-definidos, como componente a fim de agilizar o processo de desenvolvimento, tornando os aplicativos mais rápidos, mais atraentes e mais fácil de usar [4,5].

Segundo o W3SCHOOLS [6], o Bootstrap é um *framework front-end* que veio para facilitar e agilizar o trabalho do desenvolvedor, oferecendo padrões para HTML, JavaScript e CSS, tornando o site responsivo para dispositivos móveis. O Bootstrap conta com uma série de classes em CSS prontas, além de *plugins* em JavaScript (jQuery) que implementam recursos como: formulários, menus de navegação entre outros.

O JavaScript fornece às páginas Web a possibilidade de programação, transformação e processamento de dados enviados e recebidos, interagindo com a marcação e exibição de conteúdo da linguagem HTML e CSS, proporcionando estilização de cores, fontes e posicionamentos de elementos nessas páginas. O seu estudo serviu como base para a utilização do *framework* Vue.js, que será descrito com mais detalhes posteriormente [7, 8].

Após a análise das tecnologias a serem empregadas no sistema, foi iniciado o desenvolvimento de um protótipo do projeto, a fim de colocar em prática os conhecimentos adquiridos ao longo da disciplina de programação Web e os estudos realizados paralelamente. O *design* do sistema proposto foi projetado de forma a ser consistente, atrativo e intuitivo, a fim de facilitar o uso do sistema. A tela principal do sistema é mostrada na Figura 2.

Figura 2 - Tela principal do sistema proposto proposta pelo *designer* do time.



Fonte: Autor.

A aplicação conta com a barra de navegação lateral expansível que redireciona o usuário de acordo com a ação selecionada, a qual apresenta os serviços que são oferecidos pelo sistema sendo: *Home*, *Serviços*, *Clientes*, *Usuários* e *Mapa*.

A seguir, tem-se a página inicial do sistema (*Home*), na qual são apresentadas informações gerais dos serviços agendados, concluídos, impedidos e em andamentos, bem como informações mais específicas como prestador e cliente.

Figura 3 - Implementação da página principal da aplicação, conforme *design* fornecido.

The screenshot displays a dashboard with four status cards at the top: 'SERVIÇOS CONCLUÍDOS' (green checkmark), 'SERVIÇOS AGENDADOS' (blue clock), 'SERVIÇOS IMPEDIDOS' (red warning triangle), and 'SERVIÇOS IMPEDIDOS' (blue close button). Below these is a table with the following data:

| Status      | Cliente | endereço | Horário | Serviço | Prestador |
|-------------|---------|----------|---------|---------|-----------|
| Agendamento | Mark    | Otto     | 15:30   | Mark    | Pedro     |
| Andamento   | Jacob   | Thornton | 15:30   | Mark    | Pedro     |
| Andamento   | Larry   | the Bird | 15:30   | Mark    | Pedro     |

Fonte: Autor.

Nas páginas de Serviços, Clientes e Usuários, o sistema deve oferecer suporte a operações de CRUD, ou seja, permitir cadastrar seus respectivos dados, realizar consultas, atualizar dados e deletar dados. A aplicação deve permitir a realização dessas operações da melhor forma possível. Por exemplo, ao se realizar uma consulta por uma entidade, essas informações são apresentadas em tabelas, cada qual com suas informações específicas.

A opção “Serviços”, além de conter as informações apresentadas na tabela, possui também a opção de cadastrar um novo serviço. Na Figura 4 é mostrada a tela de Serviços da aplicação.

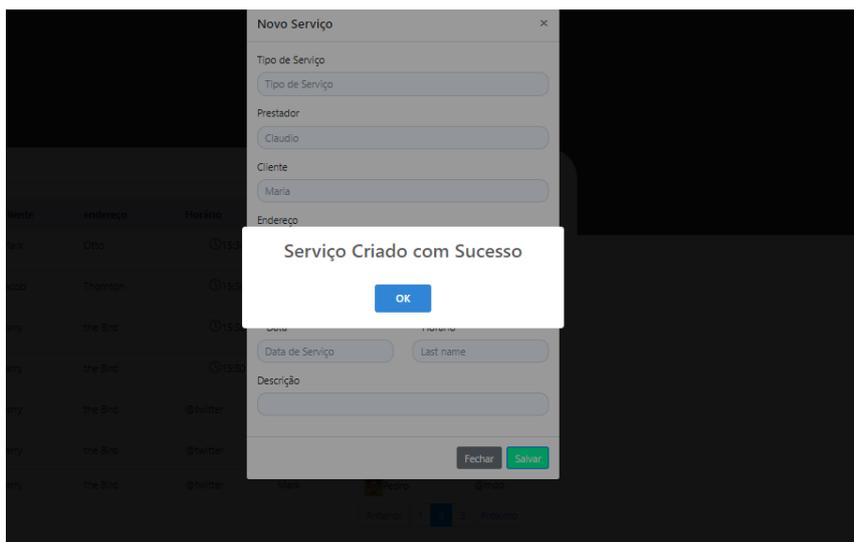
Figura 4 - Tela Serviços do sistema proposto

| Status      | Cliente | endereço | Horário | Serviço | Prestador |
|-------------|---------|----------|---------|---------|-----------|
| Agendamento | Mark    | Otto     | 15:30   | Mark    | Pedro     |
| Andamento   | Jacob   | Thornton | 15:30   | Mark    | Pedro     |
| Andamento   | Larry   | the Bird | 15:30   | Mark    | Pedro     |

Fonte: Autor

Ao clicar no botão “Novo Serviço”, é aberto um formulário com campos nos quais o operador deverá preencher. Após preencher todos os campos e clicar no botão “Salvar”, uma mensagem é mostrada informando se o serviço foi cadastrado com sucesso ou não. Após o cadastro de um novo serviço, uma nova linha deve ser apresentada na tabela apresentada na página “Serviços”. Na Figura 5 é mostrada a implementação do formulário de cadastro de um novo serviço e sua respectiva mensagem com o *status* da realização do cadastro.

Figura 5 - Tela de cadastro de um novo serviço, juntamente com sua respectiva mensagem de *status*.



Fonte: Autor

Todo esse protótipo foi construído utilizando as linguagens HTML, CSS, JavaScript e o *framework* Bootstrap. Vale salientar que esta aplicação ainda não está se comunicando com o *back-end*.

### 3.3 Construção do Sistema Proposto Utilizando o *Framework* Vue.js e a Biblioteca Vuetify

Como segunda atividade de implementação, foi designada a construção da mesma aplicação, utilizado, dessa vez, um *framework* para o desenvolvimento do *front-end* da aplicação. O *framework* utilizado para auxiliar o desenvolvimento do *front-end* foi o Vue.js, que é baseado em JavaScript e recomendado pelos integrantes da própria empresa. Dentre as diversas vantagens citadas pela equipe e exposta a literatura, o Vue.js, é considerado mais simples, flexível e ágil, em comparação com outros *frameworks*, além de possuir uma documentação atualizada e detalhada [9, 10].

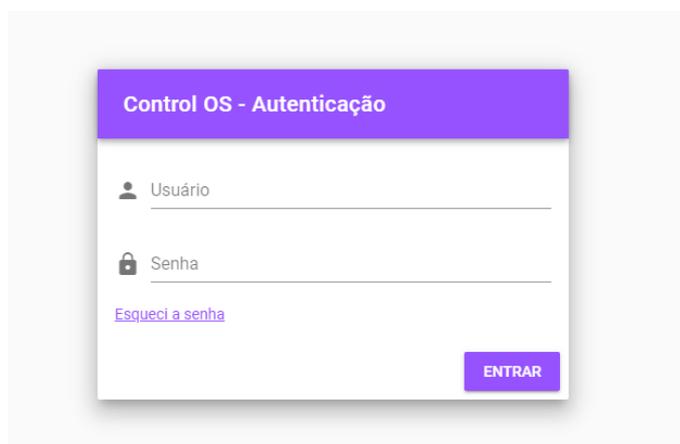
O Vue.js, possui uma infinidade de bibliotecas que auxiliam no desenvolvimento. Dentre os mais diversos recursos e bibliotecas utilizadas para a construção de aplicações, pode-se citar o Vuetify como uma das principais bibliotecas do Vue.js. O Vuetify é um *framework* responsivo em Vue.js, baseado no *Material*

*Design* no qual, possui uma grande quantidade de componentes, ferramentas e *plug-ins*, além uma documentação sólida [11].

Os componentes ofertados por esta biblioteca são semelhantes ao que foi proposto pelo *designer* da equipe, o que facilitou e aperfeiçoou a construção da aplicação.

A primeira atividade desta etapa foi à construção da tela “Cadastro/Login”, mostrada na Figura 6. Esta tela seria utilizada por administradores do sistema, possibilitando-os realizar o monitoramento e cadastro dos requisitos da aplicação, sendo apresentada logo a seguir.

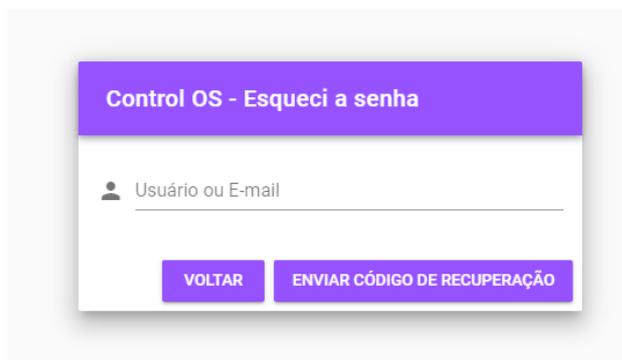
Figura 6 - Tela de login da aplicação

A imagem mostra uma interface de login para uma aplicação chamada "Control OS - Autenticação". O título da interface está em uma barra superior azul com o texto "Control OS - Autenticação" em branco. Abaixo do título, há dois campos de entrada: o primeiro é rotulado "Usuário" com um ícone de pessoa à esquerda, e o segundo é rotulado "Senha" com um ícone de cadeado à esquerda. Abaixo do campo de senha, há um link azul com o texto "Esqueci a senha". No canto inferior direito da interface, há um botão azul com o texto "ENTRAR" em branco.

Fonte: Autor.

Esta tela, conta com os campos “Usuário” e “Senha”, que seriam cadastrados na opção “usuários”. Ao clicar no Campo “Esqueci a senha”, o administrador é levado a página da Figura 8.

Figura 8 - Tela esqueci senha



Fonte: Autor.

Após preencher o seu e-mail no campo apropriada e clicar em “Enviar Código de Recuperação”, o usuário receber um código via e-mail, que lhe fornece acesso temporariamente ao sistema, permitindo realizar a mudança da senha. Vale salientar que, as respectivas telas, foram devidamente autenticadas e validades e já estavam se comunicando com o *back-end*.

Ao clicar no botão de “Entrar” na tela de Login, o usuário é levado a página inicial do sistema, especificamente a página “Ordem de Serviços”, no qual o usuário, pode realizar o cadastro de uma ordem de serviço, verificar o *status* do prestador, se o mesmo está disponível ou não, bem como: Endereço, Data de agendamento, Data de início e Data de Finalização do serviço. Vale salientar que os campos Data de Início e Data de Finalização estão em branco, porque são dados que dependem da aplicação *Mobile*, a ser desenvolvida por outro membro do time de desenvolvimento e por isso não detalhada ao longo do relatório de estágio.

Figura 9 - Página Ordem de Serviço

| Status | Cliente | Endereço | Serviço | Data de Agendamento      | Data de início | Data de Finalização |
|--------|---------|----------|---------|--------------------------|----------------|---------------------|
|        | 8       | 20       | 47      | 2019-10-06T22:41:21.403Z |                |                     |
|        | 8       | 9        | 47      | 2019-11-30T22:59:06.128Z |                |                     |
|        | 8       | 9        | 47      | 2019-11-30T22:59:06.128Z |                |                     |
|        | 8       | 10       | 47      | 2019-11-30T22:59:06.128Z |                |                     |
|        | 8       | 10       | 47      | 2019-11-30T22:59:06.128Z |                |                     |

Fonte: Autor

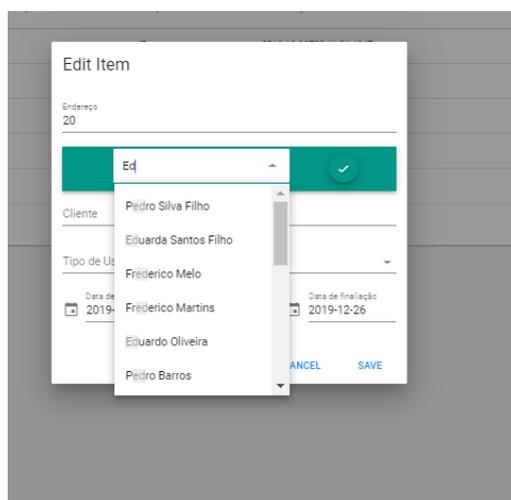
No lado direito da tabela, pode-se observar dois ícones, os quais possibilitam a edição e exclusão dos elementos da tabela. Desta forma, ao clicar na opção editar, um formulário é aberto com as devidas informações do campo selecionado da tabela, e o usuário poderá realizar as devidas alterações nos campos do formulário, alterando o item da tabela. Na Figura 10 é mostrada a tela de edição de um elemento na tabela de serviços.

Figura 10 - Tela para editar um elemento da tabela de serviços.

Fonte: Autor.

Este formulário é composto pelos componentes *Selects* e *Pickers* (componentes que permitem ao usuário selecionar um item dentre uma lista pré-definida). Assim, o usuário poderá pesquisar e selecionar os seus Clientes e Tipos de serviços previamente cadastrados no sistema. Na Figura 11 é mostrado uma pesquisa por um item de interesse previamente cadastrado.

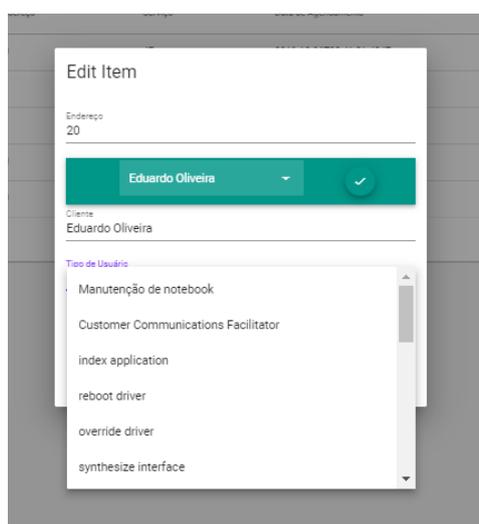
Figura 11 - Detalhe a pesquisa por um serviço previamente cadastrado no sistema.



Fonte: Autor

A Figura 12 e Figura 13 são referentes ao uso e demonstração dos campos deste formulário.

Figura 12 - Demonstração da tela editar item, como escolher o tipo de Usuário.



Fonte: Autor

Figura 13- Demonstração da tela editar item, devidamente preenchida.

Fonte: Autor

Como mencionada anteriormente, outra funcionalidade do sistema é a opção Usuários. Nesta página, mostrada na Figura 14, o operador do sistema, pode observar dados dos usuários do sistema, seja Web ou *Mobile*, assim como seus respectivos nomes, função desempenhada e nível de autorização com relação sistema. Assim como a tela “Ordem de Serviço”, esta funcionalidade possibilita o usuário editar, excluir e adicionar dados no sistema.

Figura 14 - Página de usuário

| status  | Nome ↑                | E-mail                      | Nível de Autorização | Função  |
|---------|-----------------------|-----------------------------|----------------------|---|
| offline | Administrator         | admin@admin.com             | admin                | <a href="https://icon-library.net/images/person-png-icon/person-png-icon-29.jpg">https://icon-library.net/images/person-png-icon/person-png-icon-29.jpg</a> |
| offline | Alessandra Batista    | ub@emes.us                  | admin                | <a href="https://s3.amazonaws.com/u/aces/aces/twitter/petrangr/128.jpg">https://s3.amazonaws.com/u/aces/aces/twitter/petrangr/128.jpg</a>                   |
| offline | Alessandro Moraes     | Modesta57@gmail.com         | admin                | <a href="https://s3.amazonaws.com/u/aces/aces/twitter/tereshenkov/128.jpg">https://s3.amazonaws.com/u/aces/aces/twitter/tereshenkov/128.jpg</a>             |
| offline | Alexandre Macedo      | Harmon_Prozacco@hotmail.com | admin                | <a href="https://s3.amazonaws.com/u/aces/aces/twitter/danthms/128.jpg">https://s3.amazonaws.com/u/aces/aces/twitter/danthms/128.jpg</a>                     |
| offline | Antônio Martins Filho | Melvina91@gmail.com         | admin                | <a href="https://s3.amazonaws.com/u/aces/aces/twitter/sandywoodruff/128.jpg">https://s3.amazonaws.com/u/aces/aces/twitter/sandywoodruff/128.jpg</a>         |

Fonte: Autor

Ao clicar na opção Novo Usuário um formulário, mostrado na Figura 15, é aberto a fim do usuário preencher seus campos. Logo, ao pressionar o botão “Salvar”, todos os dados preenchidos serão armazenados no banco de dados.

Figura 15 - Tela de cadastro de um novo cliente

The image shows a web form titled "Novo Cliente" (New Client). The form is centered on a dark gray background. It contains the following fields from top to bottom: "NOME COMPLETO" (text input), "Avatar" (text input), "Username" (text input), "Password" (text input), "E-mail" (text input), "Telefone" (text input), "Tipo de Usuário" (dropdown menu), and "Função" (dropdown menu). At the bottom right of the form, there are two buttons: "CANCELAR" and "SALVAR".

Fonte:Autor

As opções “Serviços”, “Funções na Empresa” e “Clientes”, seguem nos padrões das telas apresentadas acima, diferenciando-se dos seus conteúdos contidos nos seus formulários.

A validação e autenticação de alguns campos dos formulários foram implementadas com os próprios recursos da biblioteca Vuetify, bem como JavaScript e a biblioteca Vuelidate disponibilizada pelo Vue.js.

### **3.4 Comunicação entre o *Front-end* com o *Back-end***

Outra tarefa designada foi à comunicação entre o *front-end* e o *back-end* da aplicação. No *front-end*, como mencionado, foram utilizados os componentes do *Vuetify*, e o *Back-end*, onde foi o utilizado o *Node.js*. Para tanto, fizemos o uso de outra biblioteca, denominada *Vue-Axios*. O *Axios* é utilizado por uma aplicação *Web* quando se necessita salvar e exibir dados de uma aplicação diretamente do banco de dados, [9,10].

Os métodos *get*, *post* e *put e delete* servem respectivamente para: Exibir, salvar, editar e deletar dados do banco de dados via *HTTP*. Logo esses métodos, são

implementados dentro dos *Controllers*, já que adotou-se o padrão MVC, sendo apresentado na Figura 16.

#### **4.0 RELAÇÃO TEÓRICA E PRÁTICA**

Como mencionado na seção um, a realização do estágio requer do estagiário domínios teóricos e práticos adquiridos ao longo do curso. Desta forma pode-se relacionar diversos conceitos adquiridos durante o curso de Engenharia de Computação que auxiliaram diretamente e indiretamente nas atividades desempenhadas durante o estágio, como conceitos das disciplinas de: Banco de Dados, Paradigmas de Programação, Engenharia de Software e Algoritmos e Programação e Programação Web tiveram maiores aplicações nesse estágio. Além dos conhecimentos adquiridos durante o curso, também foi necessário aprender outros conceitos de grande importância nas aplicações desenvolvidas, tais como JavaScript, Vue.js e protocolo HTTP (*axios*).

## 5.0 CONSIDERAÇÕES FINAIS

O estágio supervisionado, configura-se como uma etapa importante do estudante de graduação não apenas pelo fato de se poder aplicar os conhecimentos adquiridos ao longo do curso, como também, proporciona o aluno integrar-se de forma concreta com o que o mercado proporciona, como o gerenciamento de metas e prazos, bem como administração de conflitos e contratempos que possam surgir quando se trabalha em grupo e relaciona-se diretamente com o cliente.

Diante as atividades propostas, apenas um requisito não foi atendido, devido à problemas como o tempo hábil para desenvolver todos os requisitos e principalmente, desfalque na equipe de desenvolvimento do *front-end*.

O fato de se trabalhar com desenvolvimento Web, possibilitou o conhecimento e busca de linguagens, ferramentas e tecnologias não vistas dentro do curso de Engenharia de Computação, forçando um aprofundamento detalhado nesta área de atuação, na qual se encontra em constante ascensão dentro do mercado de trabalho.

Portanto, é possível concluir que o estágio atingiu seu objetivo, visto que foi possível aplicar e associar os conhecimentos adquiridos ao longo da jornada acadêmica, além de aprender coisas novas, atendendo de maneira satisfatória as solicitações das atividades da empresa.

## 6.0 REFERÊNCIAS BIBLIOGRÁFICAS

- [1] I. M. Colombo e C. . M. Ballão, “Histórico e aplicação da legislação,” *Educar em Revista*, vol. 53, pp. 171-186, 2014.
- [2] I. C. SCALABRIN e A. M. C. MOLINARI, “A IMPORTÂNCIA DA PRÁTICA DO ESTÁGIO SUPERVISIONADO,” *Revista Unar*, vol. 7, n. 1, 2013.
- [3] G. Rodrigues, “Sistema de ordem de serviço: como funciona.,” 06 fevereiro 2019. [Online]. Available: <https://www.blog.auvo.com/post/sistema-ordem-servico>. [Acesso em 20 Dezembro 2019].
- [4] R. Jaques, “PHPIT,” 07 01 2016. [Online]. Available: <http://www.phpit.com.br/artigos/o-que-e-um-framework.phpit>. [Acesso em 20 Dezembro 2019].
- [5] M. Guedes, “TreinaWeb,” 05 Abril 2019. [Online]. Available: <https://www.treinaweb.com.br/blog/para-que-serve-um-framework/>. [Acesso em 22 Dezembro 2019].
- [6] “w3schools.com,” [Online]. Available: <https://www.w3schools.com/bootstrap/default.asp>. [Acesso em 22 Dezembro 2019].
- [7] “DevMedia,” [Online]. Available: <https://www.devmedia.com.br/guia/javascript/34372>. [Acesso em 23 Dezembro 2019].
- [8] “JavaScript Introduction,” [Online]. Available: [https://www.w3schools.com/js/js\\_intro.asp](https://www.w3schools.com/js/js_intro.asp). [Acesso em 18 Dezembro 2019].
- [9] “Vue.Js Brasil,” 14 Julho 2016. [Online]. Available: <http://vuejs->

brasil.com.br/por-que-vuejs-e-uma-boa-opcao/. [Acesso em 25 Dezembro 2019].

[10]“Vue.js,” [Online]. Available: <https://vuejs.org/>. [Acesso em 25 Dezembro 2019].

[11]“Vuetify,” [Online]. Available: <https://vuetifyjs.com/pt-BR/getting-started/quick-start>. [Acesso em 25 Dezembro 2019].