# TokenTravel: Aplicação Web para Auxiliar em Visualização de Rotas e Solicitação de Viagens em Transportes Públicos na Cidade de Capitão Poço-PA

# Lucas de Souza e Souza<sup>1</sup>, Ruan Moraes De Assunção<sup>2</sup>

<sup>1</sup>Campus Capitão Poço – Universidade Federal Rural da Amazônia (UFRA) Capitão Poço – PA – Brasil

{lucas.souza5294,ruanmoraes012}@gmail.com

Resumo: O presente projeto tem por objetivo, auxiliar os usuários interessados em realizar solicitação de transporte público, capaz de locomover até seu destino, por meio de uma aplicação web, responsável por cadastrar e solicitar viagens. Mediante o exposto, esta aplicação tem por finalidade revolucionar o transporte público nas cidades, onde as pessoas por meio de um dispositivo com acesso a internet, poderiam acessar a página web, e por meio da localização, verificar as rotas disponíveis entre as cidades, assim como, acessar cronograma de horários cadastrados.

# 1. Introdução

Os moradores de Capitão Poço utilizam diferentes meios de locomoção intermunicipais, seja por moto, carro, van, ônibus, etc. Dentre os citados, a tecnologia proporcionou uma praticidade no momento de comprar a passagem de ônibus, porém os outros métodos de viagens não obtiveram o mesmo destino. Por essa questão, os moradores que utilizam carros e vans, por exemplo, necessitam embarcar no horário em que os veículos passam ou enviar mensagens nas redes sociais do motorista/cobrador para reservar sua passagem.

Para facilitar o embarque destes cidadãos assim como reservar passagens com antecedência, evitando quaisquer transtornos, é necessário implementar medidas técnicas e estratégicas, na forma de uma aplicação web gratuita, para oferecer aos utilizadores mais confiança, segurança e simplicidade na reserva de passagens, facilitando o acesso e, assim, verificar os horários de viagem de carros e vans e quais veículos se encontram disponíveis para início de trajetória.

#### 1.1. Público alvo

O projeto tem como objetivo atender os moradores e visitantes das cidades próximas a Capitão Poço, que trabalham, estudam fora de suas cidades ou simplesmente precisam viajar para outra cidade, e com isso poderão solicitar pela plataforma viagens programadas em transportes públicos como táxi ou van, com o objetivo de chegar ao destino mais rápido, facilitando acesso aos clientes em encontrar esses profissionais. Muitos cidadãos optam por utilizar serviços de viagens, dessa forma, visando essa oportunidade em realizar um projeto, que ajude a divulgar os serviços e melhorar a comunicação entre cliente e motorista, trazendo novas oportunidades de promover esse trabalho, por meio de uma ponte onde os passageiros que precisam viajar, poderão se conectar aos motoristas responsáveis por realizar esse serviço de locomoção de passageiros, mitigando a demora/atraso ao destino.

# 1.2. Objetivos

A aplicação web tem por objetivo auxiliar os usuários que estejam interessados em solicitar viagens entre os municípios, como por exemplo Capitão Poço e Capanema do estado do Pará, onde essa aplicação irá contribuir com a praticidade de solicitar viagens e garantir boa comunicação entre o solicitante e o solicitado (cliente e motorista), além de proporcionar redução de tempo até o destino desejado, trazendo possibilidades de promover o trabalho desses profissionais.

# 1.2.1. Objetivos Específicos

A aplicação é direcionada especificamente aos cidadãos que precisam viajar entre cidades com taxistas ou vans, que pretendam viajar para outras cidades próximas, ou seja, fazer com que os cidadãos cheguem com maior rapidez ao destino final. Levando em consideração que essa aplicação deve trazer consigo, organização, confiabilidade e segurança aos usuários que utilizam a plataforma web, evitando acontecimentos como, perda de viagens, insegurança e atrasos ocorridos por falta de organização.

#### 2. Funcionalidades

Logo os elementos funcionais da aplicação serão: informar a plataforma web dentro do campo de pesquisa, a localização do ponto de partida e destino final, realizar pesquisas das rotas cadastradas dos motoristas.

As características não funcionais da aplicação serão: apresentar a numeração da rota de viagem que se faz próxima do ponto de partida, apresentar a distância do local da pessoa até o seu destino, informar ao passageiro as opções de dia e horários registrados que mais se adequa ao seu destino e informar ao usuário o valor cobrado e se tem assentos disponíveis com um determinado motorista.

#### 3. Recursos

#### **Git Commit**

- O Git Commit é um comando fundamental no sistema de controle de versão Git, que permite ao usuário salvar as alterações feitas em um repositório.
- O Git mantém um histórico de todos os commits realizados em um repositório, permitindo que os usuários revertam as alterações para uma versão anterior do código, comparam versões diferentes e visualizem quem fez cada alteração e quando.
- O Commit foi realizado com sucesso, onde a alteração do código e a adição/exclusão de arquivos, está sincronizada com o projeto inserido no repositório do GitHub.

#### Git Push

O comando Git Push é utilizado para enviar as alterações realizadas em um repositório local para um repositório remoto. O push permite que outros colaboradores tenham acesso às alterações e trabalhem em conjunto no mesmo projeto.

O push é uma operação importante no fluxo de trabalho do Git, pois permite a colaboração entre os membros de um projeto. No entanto, é importante ter cuidado ao enviar alterações para o repositório remoto, pois elas podem afetar o trabalho de outros colaboradores. Por isso, é importante sempre realizar testes e revisões antes de realizar o push.

O Push foi configurado com sucesso, onde a ferramenta é capaz de enviar os dados da máquina para o repositório do GitHub.

#### Git Pull

Git Pull é uma opção do Git que permite atualizar a versão local de um repositório a partir de uma versão remota. Essa opção é responsável por sincronizar o repositório local com as alterações realizadas em um repositório remoto.

O Pull foi utilizado para sempre que for necessário recuperar um código, quando necessário o responsável pode baixar os dados do repositório remoto do GitHub.

#### Git Merge

O Git Merge é uma ferramenta que unifica o histórico bifurcado de ramificações geradas pelo Git Branch. Essa opção é utilizada para mesclar o conteúdo de duas ou mais ramificações diferentes em uma única ramificação. Ambas as opções são úteis para trabalhar com o controle de versão de projetos em equipe.

O Git Merge é bastante eficaz por evitar conflitos de versão, garantindo que as alterações feitas em diferentes ramificações sejam mescladas corretamente. Além disso, oferece várias opções, como mesclagem automática e mesclagem recursiva, para facilitar o processo de mesclagem de ramificações.

# 3.1. Páginas HTML

#### Quantidade de Páginas

Inicialmente, o projeto TokenTravel possui em torno de sete arquivos de extensão HTML, para apresentação visual do que o projeto irá realizar. Ao decorrer da disciplina e no aprendizado adquirido externamente, serão geradas novas páginas HTML que complementarão a aplicação WEB, a fim de tornar a mesma simples e de fácil manuseio do usuário. Até o presente momento, o projeto possui as páginas: Base, Home, Login, Cadastro, Menu, Pesquisa e Portfólio.

Método Base: Este arquivo é responsável por criar métodos estáticos que fazem parte do pacote dj\_static. Dentro dessa página, foram criados três métodos: {% block menu%} (que se refere à uma página de menu, que inicialmente apresentará links para Login, Cadastro e Pesquisa), {% block body %} (este criará uma padronização do corpo do HTML e suas tags encorpadas, a fim de evitar a desconfiguração das mesmas) e {% block footer %} (neste método, os rodapés que serão posteriormente inseridos, também irão incorporar a padronização e sendo possível inserir arquivos estáticos no diretório local).



Figura 1. Print de base.html.

Método Home: Esse arquivo tem como objetivo inicial, apresentar a página inicial que irá conter alguns elementos como por exemplo, um ícone de um ônibus por meio de um método {% static 'img/TokenTravel Logo.ico' %}, que fará o ícone aparecer ao lado do título da página.



Figura 2. Print de home.html.

Método Menu: Nesse arquivo, apresentará uma lista não numerada dos métodos de encaminhamento, cadastro, login e pesquisa, assim como, essa página HTML, herda por meio de herança {% extends 'base.html' %}, a funções atribuídas da página HTML base.



Figura 3. Print de menu.html.

Método Cadastro: Nesta página, o usuário que não tiver nenhum dado presente no banco de dados, deverá se cadastrar no site para poder ter acesso às funcionalidades da aplicação. Os dados necessários para um registro correto serão: Nome completo, nome social (como o indivíduo gostaria de ser chamado), CPF, email, telefone,

endereço, senha, entre outros dados. Após essa etapa, o usuário será capaz de acessar as páginas de pesquisa, visualização de dados pessoais, entre outras opções.

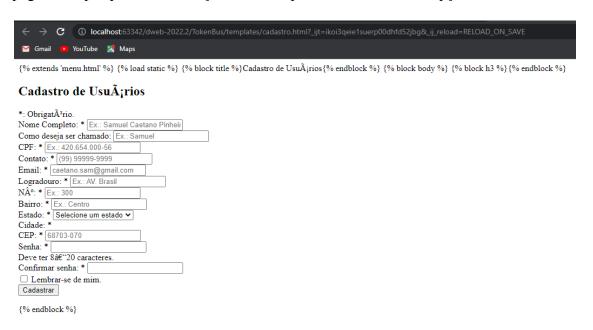


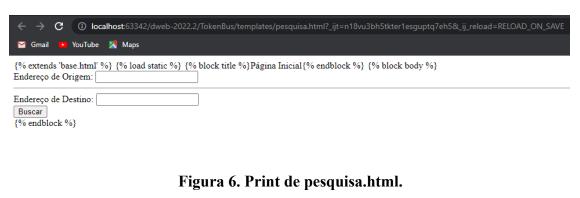
Figura 4. Print de cadastro.html.

Método Login: Nesta página HTML, será necessário para os usuários que já estejam cadastrados no banco de dados, dessa forma, a página login contém um método de acesso, que o usuário deve fornecer informações pessoais do seu cadastro por meio da função input, que é uma forma de interação com o usuário, id de identificação "login", senha de acesso "password" e um botão para confirmar as informações de acesso do usuário.



Figura 5. Print de login.html.

Método Pesquisa: No momento, a aplicação possui um método HTML inacabado, mas temos o objetivo de por meio desse método realizar uma pesquisa de localização em tempo real, no qual tem por objetivo encontrar localização dos dispositivos conectados.



Método Portfólio: Nesta página HTML, tem por objetivo realizar uma apresentação sobre o site, quais os objetivos, público alvo, missão, valores e etc. Assim como, uma breve explicação de como funciona os nossos serviços e benefícios.

como, uma breve explicação de como funciona os nossos serviços e benefícios.
{% extends 'menu.html' %} {% load static %} {% block title %}Portfólio {% endblock %} {% block body %} {% block h3 %} {% endblock %} Guia do Portfólio
Sobre a empresa     Como funciona     Beneficios
Sobre a empresa
Foto do autor
Sobre a empresa
A TokenTravel é uma empresa comprometida em simplificar e facilitar a vida das pessoas através de sua plataforma de agendamento de vans e táxis com motoristas confiáveis. Com a missão de oferecer um serviço simples e prático, a empresa se dedica a proporcionar uma experiência de transporte seguro, eficiente e confortável aos seus clientes. Além disso, a TokenTravel valoriza a transparência e a confiança em suas relações com motoristas e usuários, visando sempre a satisfação de todos os envolvidos em sua rede de serviços. Através d tecnologia de ponta e uma equipe de profissionais altamente capacitados, a empresa está comprometida em proporcionar uma experiência de transporte inigualável aos seus clientes.
Como funciona
Projeto 1
Cadastro e busca de motoristas
A plataforma TokenTravel é fácil de usar, exigindo apenas o cadastro do usuário e a busca por motoristas cadastrados. Os usuários podem agendar uma viagem com antecedência.
₽Projeto 2
Qualidade do serviço
Todos os motoristas da plataforma são avaliados pelos usuários, garantindo a qualidade do serviço. Além disso, a plataforma oferece recursos de segurança, como verificação de antecedentes dos motoristas e acompanhamento ao vivo do trajeto do táxi pelo usuário.
Projeto 3
Pagamento e preços competitivos
Os usuários podem escolher entre várias formas de pagamento aceitas na plataforma TokenTravel. Além disso, os preços oferecidos pela plataforma são competitivos em relação aos serviços de táxi tradicionais.
Beneficios
Foto do autor
Beneficios
Beneficios de utilizar a plataforma TokenTravel, como segurança, facilidade de uso e economia de tempo.
Desenvolvido por Lucas de Souza e Souza; Ruan Moraes de Assunção — ©Copyright, 2023. Contato: (91) 99826-4357; (91) 98281-7656. E-mail: lucas souza5294@gmail.com; ruanmoraes012@gmail.com. Projeto TokenTravel. UFRA — Universidade Federal Rural da Amazônia. Endereço: Travessa Pau Amarelo, Bairro Vila nova, Capitão Poço — PA.
{% endblock %}

Figura 7. Print de portfolio.html.

# 3.2. Recursos de layouts e estilo CSS

No projeto, foram criados três arquivos com extensão CSS: o primeiro denominado de style.css, o segundo e terceiro são, respectivamente, para as páginas de login (login.css), cadastro (cadastro.css) e portfolio (portfolio.css).

# **ARQUIVO STYLE.CSS:**

É um arquivo de estilos para uma página web. O código começa com as definições para os elementos "h2" e "h3", incluindo a cor, fonte, tamanho e alinhamento de texto. Em seguida, o código define a cor de fundo para o corpo da página.

Em seguida, o código define estilos para um formulário de busca, especificando a largura do campo de entrada e a posição do botão de envio.

O código também define estilos para um banner, incluindo uma imagem de fundo e texto sobreposta. O texto é estilizado com uma fonte personalizada, tamanho, cor, sombra de texto e margens. Além disso, o código define um botão com estilo semelhante, incluindo fonte, tamanho, cor, posicionamento e efeito hover. Estas alterações são para a página inicial, na qual o usuário deve entrar para acessar as funcionalidades da aplicação.

Por fim, o código define estilos para uma barra de navegação, incluindo a cor, tamanho e posição do logotipo e links. Os links da barra de navegação são estilizados com uma transição de cor, tamanho de fonte, posição e efeito hover. O código também define um estilo para uma frase única com uma fonte personalizada, tamanho, cor e alinhamento de texto.



Figura 8. Print da página home.html.

# **ARQUIVO LOGIN.CSS:**

O arquivo login.css define o estilo do formulário de login na página:

- O label tem a cor #040c21 (cor azul marinho):
- A classe .botao-login define o estilo do botão de login, incluindo cor de fundo, cor do texto e tamanho da fonte;

- A classe .caixa-formulario-login define a posição, tamanho, cor de fundo, borda e cor do texto do formulário;
- A classe form#login button define a posição do botão dentro do formulário;
- A tag h3 está oculta (display: none), pois como a página login.html herda as características de menu.html, ela acaba por apresentar esta tag indevidamente.

# **ARQUIVO CADASTRO.CSS:**

O arquivo cadastro.css define o estilo da página de cadastro. Ele contém as seguintes definições de estilo:

- A classe .caixa-formulario-cadastro: Define o estilo da caixa de formulário de cadastro, incluindo a sua posição, borda, cor de fundo, tamanho, cor do texto e fonte;
- A classe .botao-cadastro: Define o estilo do botão de cadastro, incluindo a sua posição, fonte, cor do texto, cor de fundo e borda;
- A classe .texto h2: Define o estilo do título da página de cadastro, incluindo a sua posição, cor do texto, fonte e alinhamento;
- A classe .cadastro-botao-confirmar: Define a posição do botão de confirmação de cadastro;
- A classe .cadastro-fonte: Define a fonte usada para o texto do formulário de cadastro;
- A classe .cadastro-informativo: Define a cor do texto informativo de erro na página de cadastro.

# **ARQUIVO PORTFOLIO.CSS:**

O arquivo portfolio.css define o estilo da página do portfólio. Ele contém definições de estilo. Ele contém um conjunto de regras CSS que se aplicam a um portfólio da aplicação.

- #itens: Define a altura da linha entre cada item dentro do portfólio.
- #barra-portfolio: Define a largura da barra de navegação do portfólio.
- .dropdown: Define a posição fixa e a largura da caixa de seleção do menu dropdown. Além disso, define a fonte Roboto para o texto dentro da caixa.
- .card-text: Define o alinhamento justificado para o texto dentro das cartas do portfólio.
- .lead mb-4: Define o alinhamento justificado para o texto no bloco de introdução do portfólio.
- #portfolio01: Define a largura e altura do item de portfólio na primeira posição. Define a posição absoluta e a distância da parte superior e esquerda da página.
- #sobre-a-empresa: Define a margem inferior de 700px para o bloco "Sobre a empresa" no portfólio.

#### 3.3. Ferramentas do bootstrap utilizadas

#### **BOOTSTRAP**

Esse framework foi utilizado em cinco páginas html: menu, login, cadastro e portfólio. Utilizando estruturas de código que foram adaptadas, como navbar e container.

#### PÁGINA HTML MENU:

Foi utilizado uma estrutura de código navbar "barra de navegação" na página html menu. O código fornecido pelo framework Bootstrap, adaptado manualmente, foi capaz de definir uma barra de navegação para o usuário navegar livremente entre as páginas do site.

A <nav> sendo o elemento que identifica a barra de navegação, incluindo elementos que irão personalizar essa barra, como por exemplo, a classe navbar-expand-lg fornecerá para os dispositivos uma navbar que poderá se expandir, dependendo do dispositivo. A classe bg-body-tertiary irá definir uma cor no fundo da navbar e a classe "navbar-light" define a cor do texto da barra de navegação como "light", para ter contraste nas palavras.

Dentro da navbar, há um <a> que é o link para a página inicial e contém a logo da empresa. A classe navbar-brand indica que é a logo da navbar. Assim como, contém um botão (navbar-toggler), tornando compactar de forma organizada a barra de navegação em dispositivos menores, como exemplo deixando oculto os elementos cadastrar, login, portfólio, sendo possível ao click no menu expandir os seus elementos que ficam definidos em <div class="collapse navbar-collapse" id="navbarNav">.

O formulário de pesquisa está incluído no código e consiste em três campos de entrada (um para a cidade de origem, um para a cidade de destino e um terceiro para a data) e um botão de pesquisa. A classe d-flex é usada para definir a exibição como flexbox e o atributo ms-auto é usado para alinhar os links à direita.

Por fim, temos uma lista não ordenada () dentro da barra de navegação que irá contém páginas que irão direcionar o usuário para outra páginas contidas na aplicação, como "Sobre", "Portfólio", "Login" e "Cadastrar". Cada link é definido como um item da lista () com a classe nav-item e a classe nav-link é usada para estilizar os links como links de navegação.



Figura 9. Print da página menu.html.

# PÁGINA HTML LOGIN:

Nesse código HTML foi utilizado uma estrutura do Bootstrap para criar um formulário de login, onde o usuário poderá fornecer suas informações de acesso aos serviços da plataforma.

A classe "container" irá envolver todo o conteúdo que está contido em outra <div "caixa-formulario-login">, que tem por finalidade agrupar todos os elementos do formulário. Após a criação do formulário usando a tag é a tag HTML usada para criar um formulário. É atribuída a classe "row gy-2 gx-3 align-items-center", que usa o sistema de grid do Bootstrap para definir a largura e o espaçamento das colunas. A classe "col-auto" faz parte de grid do Bootstrap tem por finalidade definir a largura de uma coluna de forma automática, levando em consideração o conteúdo contido nela.

O formulário apresenta apenas dois campos, endereço de e-mail e senha de autenticação, com um atributo "required" que torna o campo obrigatório. Há também um checkbox para "Lembrar-se de mim" que é definido com a tag <input> do tipo "checkbox". O botão de login é criado com a tag <button> e tem a classe "btn btn-primary botao-login".

De forma geral, se trata de um formulário básico usando um pouco de Bootstrap, o mesmo contém campo para o endereço de e-mail e senha, um checkbox para lembrar-se de mim e um botão de login.

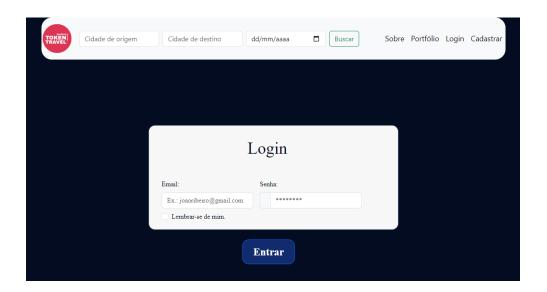


Figura 10. Print da página login.html.

# PÁGINA HTML CADASTRO:

O código do arquivo cadastro.html é um formulário para cadastro de usuários que contém campos para o preenchimento de informações pessoais, como nome completo, CPF, telefone, e-mail, endereço e senha. Abaixo está a explicação de cada item do código:

- A tag div inclui a classe "container" do Bootstrap e ela define uma seção para exibir o conteúdo do formulário.
- A tag div utiliza a classe "caixa-formulario-cadastro" para definir que a caixa de exibição do formulário esteja dentro da seção container.

- A tag h2 usa a classe "texto" para inserir o título do formulário com o texto "Cadastro de Usuários".
- A tag form faz uso da classe "row g-3" e do id "cadastro-formulario" respectivamente, como tamanho do campo de inserir dados e como identificador para uma função em JavaScript posteriormente. Ela define o formulário e atribui um id "cadastro-formulario". O atributo "class" define um layout de grade Bootstrap de 3 colunas.
- A tag span possui a classe "form-text cadastro-informativo" para exibir informações importantes para o preenchimento dos campos, como "Obrigatório".
- A tag div utiliza a classe "col-md-6" para definir uma coluna de tamanho médio para o campo.
- A tag label usa o parâmetro for="cadastro-nome-completo" e a classe "form-label" para definir uma legenda para o campo "Nome Completo".
- A tag input utiliza o tipo "text", da classe "form-control", do identificador id="cadastro-nome-completo", do exemplo de campo placeholder="Ex.: Samuel Caetano Pinheiro" e da função required para definir um campo de entrada de texto para o nome completo com um placeholder e o atributo "required" que torna o campo obrigatório.
- A tag select faz uso do identificador id="cadastro-estado", da classe "form-select" da função onchange="mostrarCidades()" e da função required para definir um campo de seleção para o estado do usuário com uma opção padrão e o atributo "required" que torna o campo obrigatório. A função "mostrarCidades()" é chamada quando o usuário seleciona um estado e exibe um campo de seleção para as cidades do estado.
- A tag select usa o identificador id="cadastro-cidade", o estilo style="display: none" e o required para definir um campo de seleção para a cidade do usuário com uma opção padrão e o atributo "required" que torna o campo obrigatório. Este campo é exibido quando o usuário seleciona um estado e a função "mostrarCidades()" é chamada.
- A tag input faz uso do tipo type="password", da classe "form-control", do identificador id="cadastro-senha" e do required para definir um campo de entrada de senha com o atributo "required" que torna o campo obrigatório. A senha é exibida como asteriscos no formulário.

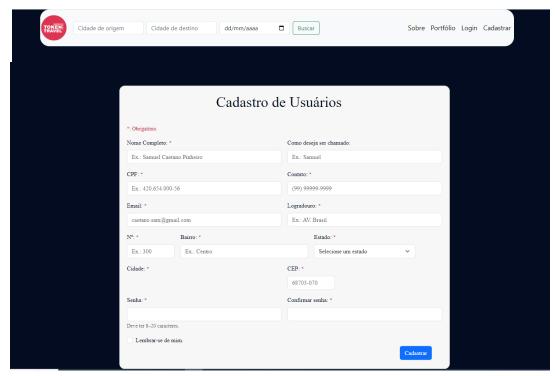


Figura 11. Print da página cadastro.html.

# PÁGINA HTML PORTFÓLIO:

O código do arquivo portfolio.html apresenta uma página que descreve a empresa TokenTravel, seus benefícios, e como sua plataforma funciona. A página possui três seções: "Sobre a empresa", "Como funciona" e "Benefícios".

Para criar a interface da página, o Bootstrap foi utilizado, com suas classes de CSS pré-definidas e um sistema de grid para tornar o layout responsivo. A classe "dropdown" é usada para criar o botão "Guia do Portfólio" que expande um menu ao ser clicado. As classes "card" e "card-body" são usadas para criar cartões que exibem as informações sobre a empresa e seus serviços. As classes "display-4", "mb-4", "py-5" e "col-md-4" são usadas para definir o estilo de elementos como títulos, margens, e larguras de colunas.

O Bootstrap também fornece scripts JavaScript para recursos interativos, como o menu suspenso ("dropdown-menu") que é ativado pelo botão "Guia do Portfólio". O script também é usado para criar um efeito de slide suave entre as seções quando o usuário clica em um link do menu. A biblioteca JavaScript do Bootstrap é referenciada no código-fonte por meio de um link para um arquivo CDN.



Guia do Portfólio ▼



#### Sobre a empresa

A TokenTravel é uma empresa comprometida em simplificar e facilitar a vida das pessoas através de sua plataforma de agendamento de vans e taixis com motoristas confláveis. Com a missão de oferecer um serviço simples e prático, a empresa se dedica a proporcionar uma experiência de transporte seguro, eficiente e confortivel aos seus cilentes. Além disso, a TokenTravel valoriza a transparência e a conflança em suas relações com motoristas e usuários, visando sempre a satisfação de todos os envolvidos em sua rede de serviços. Através de tecnologia de ponta e uma equipe de profissionais altamente capacitados, a empresa está comprometida em proporcionar uma experiência de transporte inigualável aos seus clientes.

# Não possui um cadastro?

#### CAPASTRE-SE

Seja mais um a utilizar nosso serviço sofisticado!

Escolha a data Escolha o destino

#### Cadastro e busca de motoristas

A plataforma TokenTravel é fácil de usar, exigindo apenas o cadastro do usuário e a busca por motoristas cadastrados. Os usuários podem agendar uma viagem com antecedência.

# PRATICIDADE

#### QUALIDADE

#### SEGURANÇA

Nossa aplicação web oferece praticidade, qualidade e segurança em um só lugar.

#### Qualidade do servico

Todos os motoristas da plataforma são avaliados pelos usuários, garantindo a qualidade do serviço. Além disso, a plataforma ofercer recursos de segurança, como verificação de antecedentes dos motoristas e acompanhamento ao vivo do trajeto do táxi pelo usuário.

# CUSTO BENEFÍCIO

Sobre Portfólio Login Cadastrar

Preços acessíveis alinhados com o mercado.

# Pagamento e preços competitivos

Os usuários podem escolher entre várias formas de pagamento aceitas na plataforma TokenTravel. Além disso, os preços oferecidos pela plataforma são competitivos em relação aos serviços de táxi tradicionais.

# SEQURANÇA FACILIDADE ECONOMIA

# Beneficios

Beneficios de utilizar a plataforma TokenTravel, como segurança, facilidade de uso e economia de tempo.

Figura 12. Print da página portfolio.html.

# 3.4. Recursos com javaScript

# 3.5. Banco de dados

# **3.6. CRUD**

# **Outras**

Apresentar outros recursos que não tenham sido abordados nas aulas, mas foram implementados no app, podendo ganhar ponto extra.

# Link da página web

<a href="https://github.com/ruanmoraes018/dweb-2022.2">https://github.com/ruanmoraes018/dweb-2022.2</a>

# Referências

Boulic, R. and Renault, O. (1991) "3D Hierarchies for Animation", In: New Trends in Animation and Visualization, Edited by Nadia Magnenat-Thalmann and Daniel Thalmann, John Wiley & Sons ltd., England.

Dyer, S., Martin, J. and Zulauf, J. (1995) "Motion Capture White Paper", http://reality.sgi.com/employees/jam\_sb/mocap/MoCapWP\_v2.0.html, December.