



UNIVERSIDADE  
LUSÓFONA

# BestRide Front-end

## Trabalho Final de Curso

### Relatório Final

Aluno 1 : Cláudio Caetano a21902813

Aluno 2 : Pedro Costa a21904825

Orientador : Rui Ribeiro

Trabalho Final de Curso | LEI | 05/06/2022

[www.ulusofona.pt](http://www.ulusofona.pt)

### **Direitos de cópia**

*BestRide*, Copyright de (Cláudio Caetano, Pedro Costa), ULHT.

A Escola de Comunicação, Arquitectura, Artes e Tecnologias da Informação (ECATI) e a Universidade Lusófona de Humanidades e Tecnologias (ULHT) têm o direito, perpétuo e sem limites geográficos, de arquivar e publicar esta dissertação através de exemplares impressos reproduzidos em papel ou de forma digital, ou por qualquer outro meio conhecido ou que venha a ser inventado, e de a divulgar através de repositórios científicos e de admitir a sua cópia e distribuição com objectivos educacionais ou de investigação, não comerciais, desde que seja dado crédito ao autor e editor.

## **Resumo**

Oferecer a melhor qualidade de serviço em animação turística. BestRide tem como objetivo rentabilizar tempo e dinheiro para profissionais e clientes, ser um suporte de gestão, referência cultural, integrar a oferta nacional numa única plataforma, vender, analisar e avaliar a diversidade existente para também criar novos passeios e experiências em áreas por vir explorado dentro e fora das aldeias / cidades, de modo online e offline; trabalhar em parceria com fornecedores de produtos / serviços importantes para estes públicos, contribuir para aumentar as exportações nacionais, desonerar municípios, polícia, proteção civil; integrar a comunidade académica no desenvolvimento e continuidade da plataforma.

# Abstract

Offer the best quality of service in tourist entertainment. BestRide aims to monetize time and money for professionals and customers, be a management support, cultural reference, integrate the national offer in a single platform, sell, analyze and evaluate the existing diversity to also create new tours and experiences in areas to be explored inside and outside villages / cities, online and offline; work in partnership with suppliers of important products / services for these audiences, contribute to increasing national exports, relieve municipalities, police, civil protection; integrate the academic community in the development and continuity of the platform.

# Conteúdo

<b>Resumo</b>	<b>iii</b>
<b>Abstract</b>	<b>iv</b>
<b>Lista de Figuras</b>	<b>vi</b>
<b>1 Identificação do Problema</b>	<b>1</b>
<b>2 Levantamento e análise dos Requisitos</b>	<b>4</b>
<b>3 Viabilidade e pertinência</b>	<b>8</b>
<b>4 Solução Desenvolvida</b>	<b>10</b>
4.1 Ionic . . . . .	10
4.2 Docker . . . . .	10
4.3 AWS - Amazon Web Services . . . . .	11
4.4 Stripe . . . . .	12
4.5 ArcGIS Webservice . . . . .	12
4.6 Arquitetura da solução . . . . .	12
<b>5 Benchmarking</b>	<b>14</b>
<b>6 Método e planeamento</b>	<b>17</b>
<b>7 Resultados</b>	<b>19</b>
<b>8 Conclusão e trabalhos futuros</b>	<b>21</b>
<b>Bibliografia</b>	<b>22</b>
<b>Manual Técnico da Aplicação</b>	<b>23</b>
<b>Anexo 1 - Imagens</b>	<b>25</b>
<b>Anexo 2 - Ecrãs Desenvolvidos</b>	<b>33</b>
<b>Glossário</b>	<b>54</b>

# Lista de Figuras

1	Impacto do Covid nas áreas de negocio. Ver noticia [2]	25
2	Arquitetura de uma aplicação Ionic	25
3	Ficheiro env do Ionic	26
4	Código da lista de roteiros	26
5	Código Docker	27
6	Código do tradutor de a.i	27
7	Arquitetura de publicação das apps	28
8	Código HTML para um formulário do Stripe	28
9	Lista de movimentos no modo de testes no Stripe	29
10	Exemplo de aplicação do ArcGIS	29
11	Diagrama da arquitetura geral do frontend	30
12	Cronograma TFC	30
13	Plano de testes no trello para app turista	31
14	Detalhe de um processo BPMN	31
15	Deploy das aplicações	32
16	Exemplo de configurações para a aplicação do profissional	32
17	Criação conta turista	33
18	Login Turista	34
19	Recuperação Conta Turista	35
20	Mudar Idioma da App	36
21	Eliminar Conta / Alterar Dados	37
22	Listagem de Tours	38
23	Comentário de um Tour	39
24	Formulário para booking do roteiro antes do pagamento	40
25	Formulário para alterar a password antiga	41
26	Pagamento do tour com Stripe	42
27	Formulário para comentar o tour	43
28	Opção de Criar um Condutor ou uma Empresa	44
29	Formulário de Criar um Condutor	45
30	Formulário de Criar uma Empresa	46
31	Login, com opção de usar Facebook ou Google	47
32	Alterar Dados / Eliminar Conta	48
33	Criar e Listar Veículos associados a uma Empresa	49
34	Estatísticas associada a Empresa	50

35	Criação de Roteiros com Pontos de Interesse Associados . . . . .	51
36	Empresa pode associar condutores e veículos . . . . .	52
37	Resposta da aplicação a um ecrã de PC . . . . .	53

# 1 Identificação do Problema

A ideia do projeto, BestRide [3], começou para resolver problemas no negócio dos Tuk Tuk - um negócio nascido na Itália e introduzida em Lisboa para a relevância do turismo no nosso país. Sabemos que o turismo tem grande influência em todos os sentidos socioculturais e, portanto, é uma área de grande desenvolvimento do que podemos revelar aos turistas - pois o negócio é concentrado nos turistas, e pela sua descoberta das grandes áreas de interesse histórico dispersadas pelas cidades de Portugal com maior destaque populacional.

Os tuk tuk nascem assim para oferecer serviços culturais ao longo de um roteiro com a finalidade de mostrar as principais áreas ou pontos históricos pelo qual fazem a cidade sendo conhecida. Oferece também a possibilidade da realização de casamentos e outras temáticas de celebrações ou festas. Com a situação do Covid-19, houve claramente alguns problemas que causaram um grande impacto no negócio. Podemos observar na imagem. Fig. 1

Existem outros problemas a notar nos negócios dos tuk tuk's. Pela seguinte ordem de ideias enumeramos alguns problemas encontrados,

- A inexistência de um sistema de informação para a gestão do negócio pelo que é uma grande vantagem para esta startup.
- O negócio tem épocas pouco produtivas entre o intervalo temporal de novembro e março devido á pouca quantidade de turistas.
- Não existe nenhum programa certificacional para educar os guias / animadores turistas sendo que qualquer pessoa com uma licença de condução válida pelas leis pode ser contratado pelas empresas de Tuk Tuk e as mesmas ensinam unicamente a condução das próprias viaturas.
- Os preços médios não são facilmente conhecidos pelo que envolve alguma investigação por parte dos clientes de forma a perceber quais as empresas cujo preço é mais fiável.
- A existência de uma rotatividade constante de recursos humanos.
- O facto de haver poucos lugares para as viaturas torna-se muito incómodo.
- As empresas de Tuk Tuk tem um controlo limitado no que diz respeito á questão de pagamentos dos roteiros e do número real de viagens feitas por dia.
- Não existe a organização dos roteiros como existe em outras áreas como a reserva num hotel, pelo que, indica então uma pouca procura pela área.

Estes problemas tem então solução na proposta deste TFC por parte do front-end, pelo que auxilia por cada necessidade de uma forma acessível e engenhosa no sentido de se colocar "na

pele"dos turistas e dos guias e empresas, para estudar o melhor procedimento de UI/UX - sabendo que já temos a informação tratada do lado do back-end para construir uma ferramenta de gestão no negócio dos Tuk Tuk vindo pela perspectiva dos dois lados do negócio - em que num lado temos os turistas e por outro lado as empresas que oferecem os seus serviços de animação turista, então exige duas plataformas que embora sejam diferentes em termos de requisitos e funcionalidades encontram sintonia na forma de como os turistas poderão usufruir do sistema de informação quando começarem a fazer reservas de roteiros de uma forma rápida e simples com todas indicações desejadas e as operações que se encontrarem disponíveis.

Falando um pouco sobre o começo do presente TFC, a criação da equipa teve lugar em meados de abril de 2021 em que os alunos Cláudio Caetano e Ricardo Faia foram escolhidos sob recomendação do orientador do presente TFC para começar a estudar o problema e sugerir uma solução tecnológica híbrida-web devido á ausência no mercado de aplicações mobile após o estudo do próprio cliente, Sónia Gomes, embora exista concorrência que faça o mesmo mas não com o detalhe do presente projeto. Logo ambos os alunos trabalharam tanto em front-end como back-end e tinham contacto também com reuniões via zoom de forma a poder ter uma realidade do que o cliente ou parte interessada do negócio tem necessidades que se transforma em requisitos e por consequência devem ser transformados em processos mais diretos como casos de usos(*use cases*).

Portanto o projeto até ao início do terceiro ano teve um excelente avanço não escusando até as pausas do verão que causaram algum atraso, mas com a pandemia foi difícil até mesmo se a aplicação turista estivesse finalizada para poder ser testada por clientes e empresas, seria útil ter a outra aplicação para que faça sentido a lógica do negócio; os turistas compram roteiros customizados por empresas que os publicam com o devido condutor e carro próprio bem como para situações especiais. Concluindo, o projeto teve um excelente avanço por ambos embora que Cláudio se dedicou mais em problemas mais difíceis do sentido de dominação do Angular e do Ionic para a implementação e configuração por exemplo de certas ferramentas como o Stripe e integração das mesmas totalmente na aplicação no qual o próprio se resumiu a investigar muito estes problemas e que continuou no TFC juntamente com Pedro Costa que o auxiliou nesta parte.

Portanto, a criação dos TFC de front-end e back-end acontece no momento em que Pedro Costa e Ângelo Bernardes (colabora com Ricardo Faia no back-end) se juntam ao TFC conforme decidirem em que parte querem ter mais foco e cria-se a equipa de 2 elementos em cada setor, o de apresentação e da lógica de dados e a sua gestão em bases de dados. Nessa altura recolhia-se mais requisitos para finalizar um MVP para a aplicação do turista e começa a recolha da listagem de requisitos para o início do desenvolvimento da aplicação da empresa.

Existem websites de empresas de Tuk Tuk que atualmente fornece os mesmos serviços, mas

sabemos que as apps tendem a auxiliar o utilizador de uma forma simples e rápida e este é o propósito da criação desta solução. Além de haver estas necessidades para resolver, podemos visualizar os problemas que as empresas enfrentam nos Tuk Tuk, estudar a melhor abordagem para cada problema e realizar a implementação nas apps.

Vamos explicar resumidamente qual o objetivo para o turista e para os guias e as respetivas empresas. Começa por haver a parte do registo, com as diferentes necessidades para cada grupo de utilizadores. Realizado o registo, o cliente, têm acesso aos benefícios que a app contém, nomeadamente começa o "Tuking", ou seja, começa a procurar roteiros do seu interesse para reserva, realiza o pagamento dos mesmos numa determinada data e hora; um guia recebe a confirmação de que um ou mais clientes querem fazer este roteiro no qual esta associado. O condutor ou a empresa, adquirem a app, registam-se e tem acesso a um serviço de gestão completo. Conseguem criar roteiros, associar veículos e condutores; podem manter um roteiro em modo de rascunho enquanto estiverem em atualizações. Quando o roteiro ficar finalizado, aparece na lista de roteiros disponíveis.

Ambas as aplicações estão a ser construídas a pensar no modo de como se podem apresentar perante a diversidade dos dispositivos como computadores portáteis ou tablets para ser o mais responsivo possível, não sendo unicamente uma aplicação só para os telefones modernos. A aplicação do profissional tem essa particularidade por exemplo.

Comparando os resultados obtidos perante a finalização do TFC com o proposto anteriormente. O grupo conseguiu com sucesso resolver quase todos os problemas propostos, sendo que a dificuldade na comunicação entre as duas equipas (back-end e front-end), levou a que certas soluções não ficassem de total agrado à equipa, como por exemplo o problema, "A existência da rotatividade constante de recursos humanos", pois faltou acabar a associação entre condutores e empresa. Tendo isto, para uma primeira versão da aplicação/website a equipa do front-end acredita que fez um ótimo trabalho, ficando assim também com uma boa base para que futuros colegas de licenciatura consigam pegar no trabalho e melhorá-lo fazendo eles a segunda versão.

## 2 Levantamento e análise dos Requisitos

Na presente secção listamos os requisitos das apps seja da app do turista seja da app do profissional/empresa pelo qual se focou a solução do desenvolvimento das mesmas. A grande parte dos requisitos foram criados pelos stakeholders de forma a também a equipa de desenvolvimento poder perceber melhor o modelo de negócio a desenvolver.

Devido a ambas as apps serem MVP's os requisitos tiveram alguns degraus de alterações ao longo do tempo, resultado de ideias propostas por Sónia Gomes e outros parceiros de negócios, sejam stakeholders ou empresas reais que testaram as apps.

A listagem apresentada de seguida representa o que deve estar funcional no sentido de preparar as apps para implementação de cenários de testes mais realistas envolvendo empresas e clientes. Para tal, estes requisitos foram propostos para garantir as funcionalidades básicas ou obrigatórias para o começo deste negócio. Podemos começar por descrever os requisitos associados á aplicação do turista.

### **Requisitos funcionais da app turista**

1. O utilizador poderá criar conta de forma a poder ter acesso ás funcionalidades da aplicação. Aceita os termos de condição para criar a conta. Fig. 17
2. O utilizador pode realizar login por email e password ou pelas redes sociais Facebook ou Google. Fig. 18
3. O utilizador pode recuperar a sua password da conta. Fig. 19
4. O utilizador pode alterar a password da conta. Fig. 25
5. O utilizador pode eliminar a conta respeitando a norma RGPD. Fig. 21
6. O utilizador pode visualizar os dados associados a conta e altera-los caso necessário. 21
7. O utilizador pode escolher o idioma que pretende de forma a ser mais acessível para o utilizador. Fig. 20
8. Roteiros são apresentados ao utilizador de forma ao mesmo poder saber mais detalhes sobre o mesmo. Fig. 22
9. O utilizador consegue reservar e agendar uma viagem num dado roteiro. Fig. 22

10. Utilizador pode comentar uma viagem (escolher um valor no ranking e um texto) após finalizado a viagem de forma a outros utilizadores poderem ver o feedback. Fig. 23
11. O utilizador pode visualizar os meios de pagamentos para finalizar a inscrição da reserva num dado roteiro. Fig. 26
12. O utilizador pode caso seja necessário atualizar os meios de pagamentos.
13. Apresentar a média dos comentários associados ao tour. Fig. 20
14. Possibilidade de visualizar roteiros tuk tuks's em outras cidades. Fig. 22

Na aplicação do profissional listamos os requisitos digamos obrigatórios de forma á aplicação ter uma ligação com a aplicação do turista. Aqui são apresentados dois atores, a empresa e o condutor.

### **Requisitos funcionais da app profissional**

1. O utilizador pode criar conta de empresa ou de condutor. Fig. 28, 29, 30
2. O utilizador pode realizar login com Facebook ou Google. Fig. 31
3. O condutor pode recuperar, atualizar e eliminar a conta associada. Fig. 32
4. O utilizador pode ver as suas informações da conta para verificar se esta tudo correto na sua criação. Fig. 32
5. O utilizador empresa pode criar veículos para depois associar a condutores. Fig. 33
6. O utilizador empresa pode associar condutores e veículos. Fig. 36
7. O utilizador condutor pode se associar a uma empresa existente.
8. O utilizador empresa pode visualizar estatísticas com base nas informações dos condutores, roteiros e outros dados. Fig. 34
9. O utilizador empresa pode criar roteiros com os devidos pontos de interesse de forma aos utilizadores poderem ter uma noção do que vão cruzar ou ver ao longo do percurso. Fig. 35
10. O utilizador empresa pode guardar um roteiro como rascunho de forma a estar visível ao público quando finalizado. Fig. 35
11. A aplicação deve ser responsiva pelo qual deve adaptar o seu tamanho para os demais ecrãs de forma a não estar muito limitado ao número de dispositivos. Fig. 37

De seguida apresentamos a listagem dos requisitos não funcionais que podem ser aplicadas a ambas as plataformas.

### **Requisitos não funcionais**

- As aplicações são desenvolvidas usando a framework Ionic com a plataforma Angular.
- O processamento máximo das funcionalidades como apresentação dos conteúdos entre outros deve demorar 2 segundos.
- A implementação do ambiente para o alojamento das aplicações. num servidor npm dever ser feita em várias máquinas virtuais com o sistema operativo Linux e com uma memoria no máximo 2 GB. Usar a plataforma Docker para alocar as aplicações.
- Para produção as máquinas virtuais têm de ter no máximo 16 GB de forma a poderem correr software para gerar as respetivas aplicações para Android e IOS. Nota: Para IOS o sistema operativo tem de ser o Mac Os X.
- Verificar as normas de proteção de dados RGPD.

Alguns exemplos dos requisitos já desenvolvidos encontram-se em anexo na página 33. A ausência dos requisitos ou a incompleta finalização acontece com relevância de problemas técnicos sendo que a maiorias dos requisitos presentes não foram totalmente concluídos devido á falta de apoio e esforço por parte dos nossos colegas do back-end.

A presente aplicação continua como proposta para candidatos futuros pelo qual pensa-se em implementar alguns requisitos que façam sentido quando a aplicação estiver um pouco mais completa. De notar que os requisitos funcionais e não funcionais descritos podem sofrer alterações mas formam a base de futuras funcionalidades

1. Na aplicação do profissional/empresa, apresentar avisos de trânsito ou perigos vários para o utilizador;
2. Avisar o utilizador com um pop-up que tem um dado roteiro a visitar naquele dia;
3. Ter disponível um manual de formação para os condutores de forma a poderem estar preparados para um certo roteiro; o documento em formato .pdf pode ser descarregado pela app do profissional.
4. Implementação de um algoritmo de ranking de uma empresa com bases nos condutores e nos roteiros de forma a dar destaque aos melhores roteiros e ás empresas associadas.

5. Implementação de um sistema de segurança de autenticação mais complexo envolvendo biometria, ex: Touch sign in.
6. O utilizador pode ter acesso no mapa a viagem do roteiro em tempo real e verificar em que ponto do roteiro se apresenta e todo o itinerário destacado.
7. O utilizador profissional ou empresa tem um ecrã com estatísticas de dados relevantes. ex: quantos tour's feitos nos últimos sete dias. - Notar que o ecrã encontra-se feito mas os gráficos tem dados de testes, ou seja, ainda não são recebidos do webservice.
8. Garantir proteção nos pedidos do webservice no sentido de haver um token para autenticar que realiza o pedido pois é uma aplicação que tem progresso para a produção e tem uma grande falha de segurança.
9. De forma a garantir o bom funcionamento das autenticações de terceiros como o Facebook e o Google, temos de implementar um certificado SSL nos urls aonde temos as aplicações ionic. Isto deve-se ao facto de o cognito exigir um url protegido quanto não estamos em modo de desenvolvimento para os logins de terceiros.

O processo de levantamento de requisitos funciona para perceber as necessidades do negócio a implementar, sendo que nunca representa um comportamento estático e constante; como continuidade do presente TFC espera-se à medida que a aplicação vai sendo desenvolvida com o esforço da equipa, melhores ideias e mais suscetíveis pela sua lógica obtenham uma garantia de implementação. Seria útil ter também algum feedback de utilizadores que obteriam a app de forma a saber as dificuldades sentidas ou erros durante o processo.

### 3 Viabilidade e pertinência

O projeto BestRide apresenta uma solução para um negócio em constante crescimento - Tuk Tuk - pelo qual temos a presença do negócio em muitas cidades de Portugal e a criação de uma solução para a melhor gestão do negócio em questão iria proporcionar uma enorme automatização de várias operações que as empresas enfrentam a cada dia. Depois acrescenta então a criatividade e outras melhorias que se produtivas em termos económicos podem ser desenvolvidas no sistema.

A verdade é que no mercado, poucas soluções existem pelo que não se trata de um projeto exclusivamente académico - o projeto nasceu para ser parte da solução nos problemas tratados da secção anterior - pelo que podemos garantir que se trata de uma oportunidade e necessidade para todos os trabalhadores da animação turística e também do turismo de Portugal.

A existência de algumas ideias como a criação de uma loja/Marketplace no website do BestRide, um espaço eLearning para dar mais formação aos animadores turísticos, a possibilidade de haver "Tuking" em modo virtual como acontece nos videojogos acreditam um projeto completo e que tem muitas ideias ainda por desenvolver quer na carreira académica quer como na empresa em si. A CEO da bestRide, Sónia Gomes, trabalhou durante muitos anos na área dos tuk tuk em Portugal, tem a sabedoria do negócio logo conhece os problemas da vida profissional de quem trabalha nesta profissão pelo que a contribuição das apps resolve o problema não só dos profissionais como também por parte dos clientes terem na "palma das suas mãos" uma app que possa realizar o "tuking" de uma forma acessível, rápida e com toda a fiabilidade possível.

Abordando o resultado final do TFC, o cliente tinha dois métodos diferentes de realizar testes a aplicação sendo eles os URLs onde podiam ser acedidos em qualquer browser sendo o dispositivo mobile ou não. E no último mês versões android de cada aplicação (ver Manual técnico da aplicação). Tendo este acesso as aplicações o cliente neste caso a Sónia Gomes, quase diariamente realizava testes aleatórios as funcionalidades de cada aplicação. Se encontra algo erro ou falta de funcionalidade, tirava uma captura de ecrã ao problema ou ao ecrã onde se encontrava o mesmo. Com isto a mesma criava um "cartão" no nosso Trello, com a captura de ecrã e uma breve descrição do problema.

A partir daqui a equipa dividia tarefas consoante a aplicação, ficando os bugs da aplicação da empresa para o Pedro Costa e os da aplicação do turista para o Cláudio Caetano. Claro que com cooperação já que as aplicações foram feitas pelos dois. Os bugs maioritariamente consistiam em problemas de formulário como por exemplo: o campo IBAN na criação do driver ter tamanho máximo e não aceitar caracteres especiais, a bugs de design como por exemplo: mudar o logótipo da top bar da aplicação para um já oficial, pois até aqui tínhamos várias possibilidades de logótipos.

Corrigindo os bugs marcávamos como resolvidos lançávamos uma nova versão da aplicação (explicado melhor no capítulo de "Método e planeamento" como se faz). Usando as vantagens do Ionic o cliente tinha logo de imediato a versão mais recente disponível sendo nunca necessário instalar novos apks - daí a vantagem das frameworks das arquiteturas híbridas-web como o Ionic - ficando assim pronto para verificar se os problemas foram realmente corrigidos a seu gosto e possivelmente outros novos bugs.

## 4 Solução Desenvolvida

Na presente secção vamos apresentar a solução desenvolvida pelo front-end para desenvolver as duas apps.

### 4.1 Ionic

O Ionic é uma framework utilizada para desenvolver aplicações híbrido-web que utiliza HTML5, Typescript e CSS/SASS. No Javascript temos duas plataformas nomeadamente o Angular e o React. Utiliza capacitor para aceder a funcionalidades mais avançadas que dependem dos recursos do sistema e vai atualmente na versão 6. Usamos o Angular e implementamos também o Angular Material para melhorar a interação com o utilizado devido a aspetos técnicos como a facilidade de criar modelos customizáveis e com menos linhas de código como uma caixa de texto para a password, ou seja, simplifica alguns obstáculos de desenhos de formulários. Fig. 2

A maneira como acedíamos aos serviços do back-end era feito por duas fases. Definimos num ficheiro específico do Ionic o endpoint da API para termos consistência em cada pedido feito no que dizia ao endpoint. Fig. 3. De seguida, para cada função fazíamos os pedidos HTTP do tipo GET ou POST. Sempre dividindo os ficheiros pelo qual todas as operações envolvendo a API criávamos ficheiros de serviço próprios do Ionic para manter sempre uma arquitetura de organização com boas práticas.

A escolha também do Ionic deve-se ao facto de as exigências dos componentes gráficos revelarem um aspeto artístico logo tínhamos de saber um grande conhecimento de CSS para aplicar tanto a botões como listas e até mesmo texto, o que o designer que a Sónia Gomes contratou, tinha desenhado e enviado por email para os programadores implementarem a cada ecrã. Fig. 4

Na aplicação do profissional/Empresa foi implementado também uma técnica de responsividade para suportar a app para mobile e ecrãs de maior dimensão; de notar que talvez no futuro não seja preciso pois o ideal da aplicação é só mesmo para smarthphones a partir de o download pelas respetivas stores.

### 4.2 Docker

O Docker é uma plataforma de automação de tarefas que usa a virtualização para executar aplicações num espaço chamado "container" de forma a isolar o sistema e a aplicação para entregar software em pacotes. A utilização do Docker foi para elaborar um sistema de execução com a ajuda

das ferramentas da AWS, poderem ter um url de forma a que os clientes/partes interessadas possam testar a aplicação e consultar os requisitos pretendidos.

Usamos o Docker para virtualizar a aplicação num container numa máquina com o sistema operativo, Amazon Linux 2, num EC2 com um servidor para correr a aplicação num link de forma a poder entregar as soluções aos clientes. Com a ajuda do professor Rui Santos, conseguimos implementar um mecanismo na AWS designado de pipelines para de uma forma geral pegar no código atualizado sempre que haja um novo git push, atualizar o código no EC2 e depois voltar a correr o serviço do Docker com uma nova imagem. Fig. 5

Era enviado sempre o IP público da instância EC2 para o cliente poder testar. Caso houvesse algum problema podíamos ver os registos (logs) do que se estava a passar na máquina do Docker.

### 4.3 AWS - Amazon Web Services

AWS é uma plataforma de serviços cloud que fornece os seus serviços unicamente em serviços de nuvens, no qual dispõe soluções robustas, de alta qualidade e desempenho para o desenvolvimento de software nas indústrias, sendo muito procurada e adquirida por tais motivos.

O AWS foi usado para implementação dos servidores de gestão das bases de dados como também para disponibilizar as aplicações em ionic ao público para a realização de testes por parte dos clientes - não só sendo uma aplicação com testes locais -, autenticação de utilizadores por email/password e redes sociais como Facebook ou Google, repositório de imagens, tradutor de linguagens com A.I e também para armazenar e gerir a base de dados. Usamos o serviço S3 para armazenar as imagens dos roteiros e no futuro para outra informação que tenha sentido armazenar. Fig. 6

Foi usado serviços do CodeCommit, Deploy, Pipeline para poder atualizar a imagem do Docker que corre em cada máquina EC2 com base nas alterações de código; de seguida ilustramos a arquitetura desta solução e no futuro espera-se resolver fazer o deploy automaticamente pois cada vez que quisermos atualizar os sistemas, temos de fazer a operação manualmente. Fig. 7

De notar que quando as apps forem para produção e com a propaganda para vários utilizadores testarem da app, o sistema não vai ser adequado quando atingir um certo número de utilizadores produzindo assim uma grande carga perante os sistemas e logo existe um risco para o negócio dos serviços ficarem bloqueados e inoperáveis. Para isso deve podemos pensar num sistema de distribuição de carga e para tal o AWS fornece um serviço designado por Distributed Load Testing<sup>1</sup>. Porém, o modelo atual situa-se bastante das nossas necessidades pois ainda estamos em modo de

---

<sup>1</sup>Consulte: <https://aws.amazon.com/pt/solutions/implementations/distributed-load-testing-on-aws/>

testes.

Concluindo, a integração dos serviços da AWS nas nossas aplicações agilizou e acelerou bastante o desenvolvimento. Foi proposto pelo orientador quando um dos alunos estava a programar a criação de contas, e a recuperação com o mecanismo de enviar email com um código. Tudo que pode ser resolvido ao máximo com o Cognito. Agilizar o desenvolvimento e criar as nossas infraestruturas desta startup foi um passo elegante para a empresa BestRide.

#### 4.4 **Stripe**

O Stripe é uma plataforma de pagamento online sendo considerado o novo PayPal. A sua framework pode ser adicionada em qualquer aplicação web para consumir os seus recursos. As suas principais funções são receber pagamentos, transferências, doações e qualquer tipo de transação financeira que seja comum a e-commerces ou a qualquer tipo de site da web. Em termos de segurança e como estamos a falar de transações bancárias, o mesmo dispõe de ferramentas de A.I que analisam comportamentos criminosos de forma a proteger os clientes e consumidores. Fig. 8

#### 4.5 **ArcGIS Webservice**

ArcGIS é um webservice público que disponibiliza serviços avançados de geolocalização cuja utilidade é notável perante alguns problemas e até mesmo agiliza processos que trabalhem com geolocalização. Foi usado o *ArcGIS Developer*<sup>2</sup> para melhorar uma questão na aplicação do profissional relativamente a um requisito, o de criar o ponto de interesse para o roteiro. Fig. 10

Foi uma dica proposta pelo professor Rui Ribeiro para auxiliar o utilizador a escolher a localização do ponto de interesse sabendo a morada do sítio em questão.

#### 4.6 **Arquitetura da solução**

Iremos apresentar a arquitetura final da elaboração do presente TFC. Fig. 11

Notando de uma forma geral que foi adotado o conceito de Cloud Computing da AWS e no qual elevou várias vantagens para a elaboração das várias tecnologias tanto em front-end como em back-end em termos de desempenho, autenticação, tradutor inteligente para o suporte de vários idiomas na app e todo o que podia ser feito no ambiente local é realizado pelos serviços concedidos por uma plataforma de serviços de computação na nuvem.

---

<sup>2</sup>Consulte : <https://developers.arcgis.com/features/>

O 'core' da aplicação funciona com o protocolo HTTP e o ionic fornece uma facilidade para aplicações exigentes neste sentido com base em módulos que tratam dos pedidos HTTP. Para isso criamos serviços no ionic que são ficheiros exclusivamente para o estabelecimento de uma conexão com cada serviço do servidor.

Do lado do front-end, só precisávamos dos serviços que os alunos responsáveis do back-end implementavam e testavam. Porém a documentação não foi realizada e dificultou muito o nosso trabalho porque não sabíamos que dados tínhamos de enviar e o que podíamos receber; sabendo que o Django contém frameworks para documentar a API, os colegas do back-end não fizeram essa parte pelo qual tínhamos de conversar sempre mais a fundo para saber.

Para terminar esta secção, listamos as disciplinas que causaram impacto na solução na solução proposta:

1. Engenharia de Software - Desenho da arquitetura e modelação dos componentes;
2. Engenharia de Requisitos e Testes - Definição dos requisitos;
3. Algoritmia e Estruturas de Dados - Desenvolvimento com o máximo Performance;
4. Linguagens de Programação I e II - Usar boas práticas de POO e manutenção do software mais dinâmico;
5. Computação Móvel - Desenvolvimento focado em sistemas moveis e as suas funcionalidades;
6. Computação Distribuída - A comunicação entre o Front-end e o Back-end;
7. Sistemas de Informação na Nuvem - Integrar a AWS de forma a beneficiar com os seus serviços;
8. Iteração Humano-Máquina - Boas práticas de UI/UX e acessibilidade.

As disciplinas que foram mais elaboradas no presente TFC, por outras palavras, as disciplinas cujo conhecimento foi sempre mais presente podemos destacar Interação Humano-Máquina e valeu o esforço porque aprendeu-se a desenhar sistemas gráficos para auxiliar os utilizadores, Computação Móvel para analisar alguns aspetos técnicos da aplicação como não sobrecarregar logo a aplicação com dados ou até mesmo a aplicação não causar um desempenho capaz de atingir a carga do telemóvel, entre outros. Computação distribuída, pois, é um típico modelo de cliente servidor e logo foi o mais importante de todos e por fim a área científica de Cloud Computing para aproveitar os serviços da AWS e como podem 'acelerar' o time to market de uma aplicação.

## 5 Benchmarking

No mercado foi feita uma análise de existência em relação a concorrência. Não foi encontrado uma aplicação de Tuk Tuk o que representa uma grande vantagem neste TFC. No entanto, existem soluções que oferecem o mesmo serviço, entre as quais destacamos as empresas TripAdvisor e AirBnB por exemplo. A aplicação da BestRide tornam-se, portanto, um projeto inovador na área de turismo principalmente para a animação turística e apresentamos soluções que realmente podem auxiliar quem trabalha no ramo. Pela mesma ordem de ideias, a aplicação para os turistas deve ser muito acessível, simples e elegante de forma a ser fácil fazer tuk tuk's numa cidade e não deve ser um processo demoroso e cansativo. Vamos apresentar algumas aplicações de turismo que representam uma concorrência ao presente projeto.

### **Tours & Travel**

É uma aplicação de viagens com várias atividades para os turistas. Permite observar mapas próprios para os transportes públicos por exemplo, itinerários diários para os principais destinos da cidade, passeios e atividades que podem ser feitos ao redor da proximidade do utilizador e com avaliações validadas e confiáveis. Concluindo é certamente uma aplicação muito completa em termos de funcionalidades e esta a tornar-se bastante popular.

### **Walkbox - Roteiros Autoguiados**

A aplicação Walkbox é uma aplicação muito completa e diz respeito a publicar aos turistas os locais de interesse no local onde se encontra com várias opções de escolha para roteiros. Apresenta também vários detalhes dos pontos de interesse como a história associada, galeria de fotos. Os mapas são simples e com um design ótimo pelo qual parece dinâmica, atrativa e de fácil acessibilidade. Existe também a opção para a partilha de um dado roteiro nas redes sociais. A aplicação apresentada e considerada como sendo uma das principais. Já discutido a competitividade, descrevemos

de que forma a nossa aplicação possa ser diferente de outras aplicações que possam estar a ser desenvolvidas. Como pode ser inovadora no mercado exige também que a arquitetura da aplicação seja a mais correta possível e validada com o máximo de testes possíveis por parte dos clientes. Em relação á usabilidade o objetivo é a simplicidade na escolha de roteiros customizáveis. Proporcionar uma educação para os condutores de forma a usufruir melhor do serviço e facultar melhores experiências aos turistas. Como as aplicações foram pensadas por um animador turístico - Sónia Gomes - as aplicações garantem não só um produto pensado ao detalhe por cada necessidade, ou seja, implementar soluções que auxiliem as empresas e o seus condutores como também pela parte dos turistas.

A existência de um website proporciona uma forma de explicar aos interessados como e que funciona as aplicações da BestRide como também os recursos que presta. Relevar a imagem e o "Brand" da empresa é um dos objetivos quando estiver na fase de produção e como se trata de uma "Brand" inovadora torna-se popular para procura ou partilha de quem quer fazer tuk tuk's com o melhor desempenho possível em Portugal.

## **TripAdvisor**

O TripAdvisor é uma plataforma web e mobile para avaliar os melhores custos da demais oferta de gama na área do turismo como hotéis, restaurantes e entre outros. O TripAdvisor apresenta um algoritmo de ranking que se baseia nas informações como avaliações positivas/negativas e com bases no volume das respostas apresenta então os seus resultados para destacar se o dado estabelecimento pode ser recomendado ou não.

Para o presente TFC existe diferenças óbvias pois a BestRide pretende só classifica os roteiros criados e não um hotel ou algum estabelecimento do género, porem a aplicação TripAdvisor torna-se uma vantagem por exemplo na escolha na altura de reserva para alojamento.

## **Airbnb**

O Airbnb é uma plataforma para o agendamento/reserva de um alojamento. Escolhendo a cidade pretendida o utilizador escolhe o alojamento conforme as suas necessidades. A plataforma é muito popular na altura de agendar as férias e nesse mesmo instante também ocorre os passeios de Tuk Tuk's pelo qual o Airbnb apresenta também além dos seus serviços, esta componente do negócio do presente TFC. Mas uma vantagem que se pode propor neste caso seria o Airbnb poder mostrar os roteiros de uma empresa na listagem dos seus serviços caso o utilizador quisesse o Tuk Tuk's e o próprio Airbnb podia propor a reserva na aplicação que esta a ser desenvolvida.

A publicidade para publicar a app seja em jornais eletrónicos ou em qualquer meio comunicacional vai destacar as aplicações e garantir muito mais a sua procura. A parceria com agências de viagens, companhias de transporte, hotéis e relevantes negócios que tem uma ligação primordial com turismo alimenta a presença da BestRide para os interessados que queiram conhecer vários pontos da cidade que estão a visitar.

Existe também websites que divulgam a ideia do negócio e proporcionam os seus próprios recursos como roteiros, comentários, galerias, etc. São neste caso empresas que tem o website como gestão do seu negócio e no futuro pretende-se que tais empresas possam gerir todo o seu negócio com a distância de um clique com as duas aplicações desenvolvidas. Listamos aqui algumas empresas como exemplo:

- <https://tukdreams.pt/>

- <https://www.tukonme.pt/>
- <https://www.lisbon-tuk-tours.com/>

Com isto, ao longo da realização do TFC, como já falamos não há uma concorrência direta com a ideia proposta, ou seja, não há uma aplicação dedicada a Tuk Tuks, o que levou a que não houvesse a necessidade de se ajustar a novas ideias da concorrência.

Em resumo, podemos separar as aplicações acima descritas com o que as aplicações representam: tuks tuks é o foco principal do negócio logo tudo o que envolve o negócio de tuk tuks's, os turistas e profissionais podem e devem pensar em usar estas aplicações. Porém podemos pensar nas outras aplicações em fontes de inspiração para ideias que se tornam na informática relativamente a apps em funcionalidades; usar um modelo de IA por exemplo como usa o TripAdvisor por exemplo.

## 6 Método e planeamento

No início do TFC, a aplicação do turista estava praticamente feita e resolveu-se então prosseguir para a fase de testes. Nesta fase começamos a usar o Trello para organizar o trabalho da equipa com um modelo mais simples do estilo kanban. Na altura da escolha do TFC e da junção de mais dois elementos, começou a pensar-se no desenvolvimento da aplicação do profissional, pelo qual foi levantado os requisitos ao cliente. Fig. 12

Porém, enquanto implementávamos os requisitos a seu tempo, dávamos oportunidade ao cliente para testar nem que fosse pequenas coisas que podiam fazer a diferença, portanto ambas as aplicações forem sempre vítimas de testes de modo a poder publicar a aplicação para produção, isto é, lançar o produto a várias pessoas para poderem usar as apps sejam turistas ou empresas como a Tuk Tuk Lisboa.

As reuniões no Google Meet com os clientes também foram muito produtivas pois conseguíamos ter 'feedback' dos testes realizados pelos clientes. Havia também sempre espaço para um 'brain storming' para novas ideias de funcionalidades que faltava implementar.

Os testes eram colocados em cada Trello criada para cada aplicação para manter um nível de organização favorável para a equipa e de modo a não haver confusão. Foi configurado também papéis para auxiliar a pergunta de quem é que tem de resolver este problema logo havia sempre uma ótima organização e colaboração da Sónia Gomes e das partes interessadas para garantir que os requisitos eram cumpridos. Fig. 13

De forma a perceber melhor cada processo que tinha de ser implementado em ambas as aplicações - primeiramente foi usado para a aplicação do profissional - foi criado vários diagramas do estilo *BPMN*<sup>3</sup> para os programadores do front-end terem uma ideia melhor do fluxo de execução de cada modelo de negócio específico que iria ser implementado na app. Desta forma também garantimos uma documentação para uma futura equipa perceber melhor as componentes implementadas e o resultado de cada evento o que produz por exemplo. Fig. 14

Os clientes ao testarem a aplicação, tinham de aceder aos links que a equipa enviava num grupo de WhatsApp para poderem usar as aplicações. Como funcionava esse mecanismo e como dávamos deploy para os clientes terem versões atualizadas? Primeiramente tínhamos os repositórios no serviço CodeCommit que é um serviço de repositório de código como o Github. Com a ajuda do

---

<sup>3</sup>Bussiness Process Model and Notation

professor Rui Santos configuramos no CodePipeline, três etapas em que a primeira retirava a última atualização de commit no código do CodeCommit, compilava o projeto e depois era atualizada a imagem do Docker que estava numa instância EC2. Bastava clicar em *Release change* para que o processo descrito anteriormente automatize as tarefas que definimos. Fig. 15

Esperamos no futuro, sugerimos uma melhoria crítica nesta tarefa seja obtida quando fizemos commit e automaticamente o próprio recebe uma notificação de alteração no *branch* principal e faça esta automatização sem estarmos sempre a clicar no botão para fazer deploy.

## 7 Resultados

Antes de falar dos resultados obtidos neste TFC, o grupo quer explicar a diferença entre trabalho de front-end e o projeto num todo. O front-end de um programa de software ou website é toda a interface gráfica com que o utilizador interage. Do ponto de vista do utilizador, o front-end é um sinónimo de UI/UX. Do ponto de vista de um desenvolvedor, é o design e a programação da interface que fazem a mesma funcionar.

A razão desta pequena explicação serve apenas com o propósito de que quando o grupo considera o requisito como realizado significa que esse ecrã está desenvolvido/criado na aplicação, tendo a sua funcionalidade, ou seja, back-end feita ou não.

Presenteada esta pequena explicação inicial em vez de expor todos os requisitos realizados e repetir basicamente o que já está na secção Levantamento e análise dos Requisitos, decidimos apenas por os dois únicos requisitos incompletos.

1. O utilizador turista pode caso seja necessário atualizar os meios de pagamentos.
2. O utilizador condutor pode se associar a uma empresa existente.

De forma a que os utilizadores possam aceder ao requisitos realizados e funcionalidades, foi implementada várias formas de aceder a aplicação, isto é possível graças a utilização da framework Ionic, estas podem ser todas acedidas ou executadas na secção - Manual Técnico da Aplicação - 23.

1. Browser - Versão web, com compatibilidade e responsiva com todos os dispositivos, sejam eles tablets, smartphones android/iOS ou computadores.
2. APK - Versão instalável em smartphones android, como o Ionic usa sintaxe de browser como por exemplo HTML, CSS a versão para android é em suma um "mini-browser" onde o mesmo vai conectar ao URL em questão. Ficando assim com duas apk uma para cada aplicação.

O defeito é que esta sempre dependente de uma ligação a internet. A vantagem é que não é necessário gerar uma nova apk cada vez que há uma atualização, pois é dependente do URL e basta atualizar a versão do URL, ou seja, a versão browser para que todos os utilizadores fiquem com a última versão mais recente.

Explicando um pouco como fazer a criação de cada uma basta usar o comando:

```
ionic cap build android -prod
```

Este deteta o ficheiro *capacitor.config.ts*, que contem as nossas configurações para gerar a apk. Fig. 16

O comando ira verificar se há uma instalação do Android SDK e do Android Studio, se sim

abre o mesmo. A partir daqui basta irmos a Build -> Build Bundle(s) / APK(s) -> Build APK(s), e temos a apk feita.

3. Local - A versão local não é a melhor para utilizadores finais, já que os mesmos não vão ter acesso ao código fonte, mas para futuros desenvolvedores ou professores é a forma mais fácil de trabalhar no mesmo, pois basta apenas menos de meia dúzia de comandos para ter a aplicação pronta a usar ou a modificar.

Desta forma, os clientes podem testar os resultados dos requisitos implementados e avaliar se estão conforme pretendido e discutido nas reuniões. o front-end realizou um ótimo trabalho na elaboração dos ecrãs cumprindo sempre que possível boas normas. Até ao momento não foi possível elaborar um ambiente de testes para o sistema iOS, pelo qual, espera-se no futuro um avanço nesta etapa.

## 8 Conclusão e trabalhos futuros

O presente TFC foca-se num negócio muito produtivo na altura do verão, altura aonde turistas visitam Portugal por um tempo limitado de férias. Para saberem a história ou pontos culturais de uma cidade, os tuks tuks's realizam viagens ao longo de um roteiro pré-definido aonde passam por pontos relevantes, também chamados de pontos de interesse em cada cidade. Sónia Gomes, a CEO da BestRide trabalhou no ramo e compreende as necessidades dos animadores turísticos.

Fizemos o estudo de mercado e não havia uma aplicação focada unicamente na apresentação deste negócio, por esta ordem de palavras, estas aplicações serão muito úteis neste ramo pois a mestria é focada unicamente no negócio de tuks tuks's e, portanto, uma espécie de 'must-have' para os interessados em tuk tuk's.

As aplicações têm um potencial crescimento para outros países numa versão mais avançada, seja de protótipo das aplicações seja até da própria empresa BestRide. Para tal ainda existe um trabalho grande para alcançar, mas com equipas certas e motivadas pode ser possível. Proponha-se até apresentar as aplicações e o modelo de negócio em eventos como o WebSummit para conseguir apoio de investimento.

Podemos também realçar que o presente TFC tem continuidade académica e empresarial pois os alunos estão a fazer uma aplicação académica no âmbito da disciplina do TFC e ao mesmo tempo um produto empresarial.

Foi falado muito da falta de colaboração dos elementos do back-end para a auxiliar a equipa do front-end no desenvolvimento dos requisitos. Os alunos Ricardo e Ângelo do back-end demonstraram pouco interesse em realizar este TFC e tinham muitas dúvidas em trabalhar com o Django. Podiam, no entanto, adquirir um curso no Udemy aonde tinham cursos para desenvolver API no Django. Tudo isto não foi feito pelo qual implica depois consequências para este lado do front-end que se esforçou.

Tanto os alunos Cláudio e Pedro ganham assim experiência com frameworks usadas para desenvolvimento front-end e percebem a agilidade que os serviços da *Cloud Computing* causa impacto nas aplicações. Cumprem assim o TFC e deixam um legado para futuros alunos com o máximo sucesso possível.

# Bibliografia

## Referências

- [1] DEISI, Regulamento de Trabalho Final de Curso, Set. 2021.
- [2] Rita Neto, *Eco Sapo*, <https://eco.sapo.pt/2020/04/22/virus-esta-a-atacar-o-turismo-veja-o-colapso-em-4-graficos/>, Acedido em outubro 2021.
- [3] Rede Dinâmica XXI, <https://new.rededinamicaxxi.pt/bestride/>, Acedido em 22 de outubro 2021.

# Manual Técnico da Aplicação

## APKs

Se pretender a versão instalada no seu telemóvel android: OneDrive Folder

## URLs

Se pretender a versão pelo browser pode usar estes urls:

- **Turista:** `http://18.224.189.162/`
- **Profissional:** `http://18.217.130.28/`

## Download das Ferramentas Necessárias para Correr Localmente

Para correr este trabalho iremos necessitar de 5 softwares diferentes, 4 para correr o projeto e o outro para visualizar o código do mesmo.

- **Node.js** para interagir com o ecossistema do Ionic. Download da versão mais recente aqui.
- **Um editor de código** (se pretender ver o código). Nós usamos e recomendamos o Visual Studio Code.
- **Linha de Comandos**, usar o terminal embutido no Visual Studio Code, CMD ou Powershell CLI.
- **Git**, para clonar os repositórios. Pode fazer o download aqui.
- **Browser**, pode utilizar qualquer browser que tenha instalado no seu sistema operativo.

## Correr o Projeto

1. Ir para a pasta onde pretende fazer o download do código, e abrir um terminal.
2. Clonar o repositório da aplicação pretendida.

```
git clone https://github.com/claudio-21902813/TFC_BestRide_Frontend_Tourist_Academic.git
```

ou

```
git clone https://github.com/claudio-21902813/TFC_BestRide_Frontend_Professional_Academic.git
```

3. Instalar Ionic CLI.

```
npm install -g @ionic/cli
```

4. Instalar dependências.

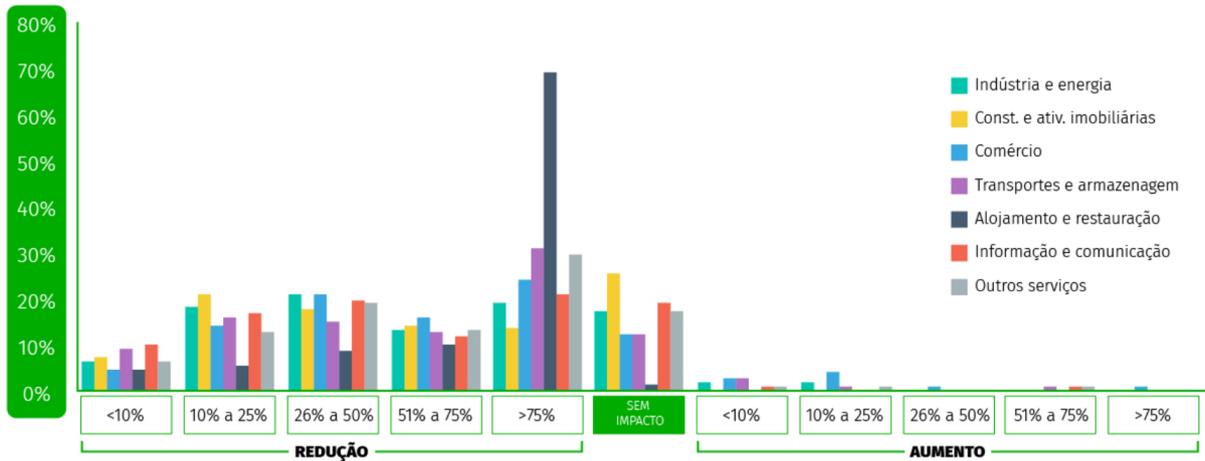
```
npm install
```

5. Por fim basta correr o programa em si.

*ionic s*

Se tudo correu como esperado, o browser ira ter um link com a aplicação a correr localmente por exemplo no localhost:8000. Ambas as aplicações são responsivas pelo qual se quiser experienciar a versão mobile deve ir às ferramentas de programador do browser que esta a utilizar e encontrar o processo para fazer essa transformação.

# Anexo 1 - Imagens



Impacto da atual crise no volume de negócios | INE e Banco de Portugal, 13 a 17 de abril.

Figura 1: Impacto do Covid nas áreas de negocio. Ver noticia [2]

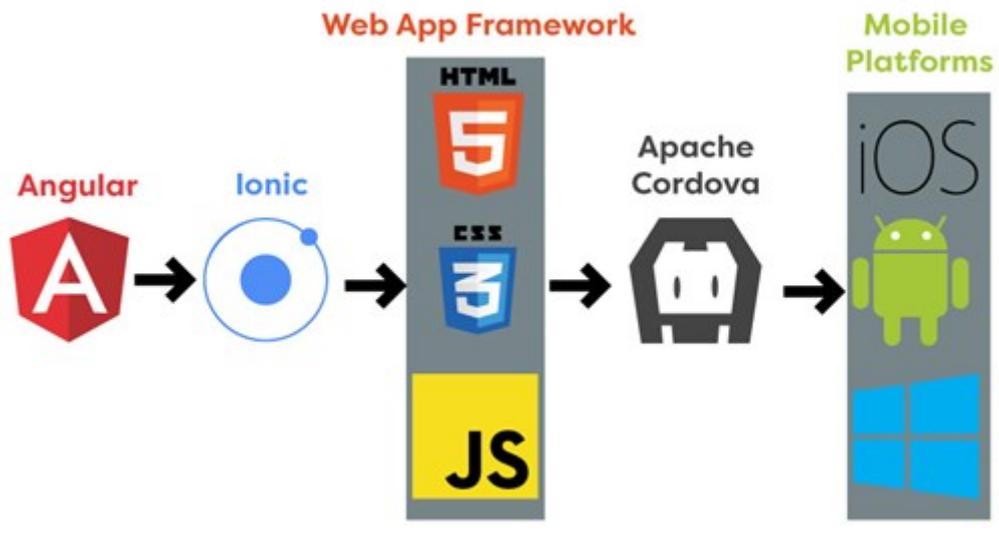


Figura 2: Arquitetura de uma aplicação Ionic

```
export const environment = {
  production: true,
  apiUrl: 'http://18.117.62.47:8000',
```

Figura 3: Ficheiro env do Ionic

```
<ion-list *ngFor="let road of searchedItem">
  <ion-card-header>
    <ion-card-title>{{ road.title| customTranslate | async}}</ion-card-title>
    <ion-card-subtitle>
      {{ "Duration : " + road.duration| customTranslate | async}}
    </ion-card-subtitle>
  </ion-card-header>
  <ion-card-content>
    <!-- 
    <ion-button (click)="showRoteiro(road)" expand="block" color="secondary"> {{ 'Book Tour' | customTranslate |
      async}}
    </ion-button>
    <ion-button (click)="trip_map_details(road)" expand="block" color="primary"> {{ 'See Tour' |
      customTranslate | async}}
    </ion-button>
    <ion-button (click)="comments(road)" expand="block" color="danger"> {{ 'See Comments' |
      customTranslate | async}}
    </ion-button>
  </ion-card-content>
</ion-list>
```

Figura 4: Código da lista de roteiros

```
FROM public.ecr.aws/f217q0b4/node:14

RUN npm install -g @ionic/cli
WORKDIR /app
COPY ./ /app
RUN npm install
EXPOSE 80
CMD ["ionic", "serve", "--external", "--prod", "--port=80"]
```

Figura 5: Código Docker

```
export class CustomTranslateService {
  public currentLang: BehaviorSubject<string> = new BehaviorSubject<string>(
    'en'
  );

  constructor(
    private http: HttpClient,
    public alertController: AlertController
  ) {}

  public translateText(text: string): Observable<any> {
    return this.http.post<any>(`${environment.apiUrl}/translate/`, {
      text,
      outputLang: this.currentLang.value,
      sourceLang: 'en',
    });
  }
}
```

Figura 6: Código do tradutor de a.i

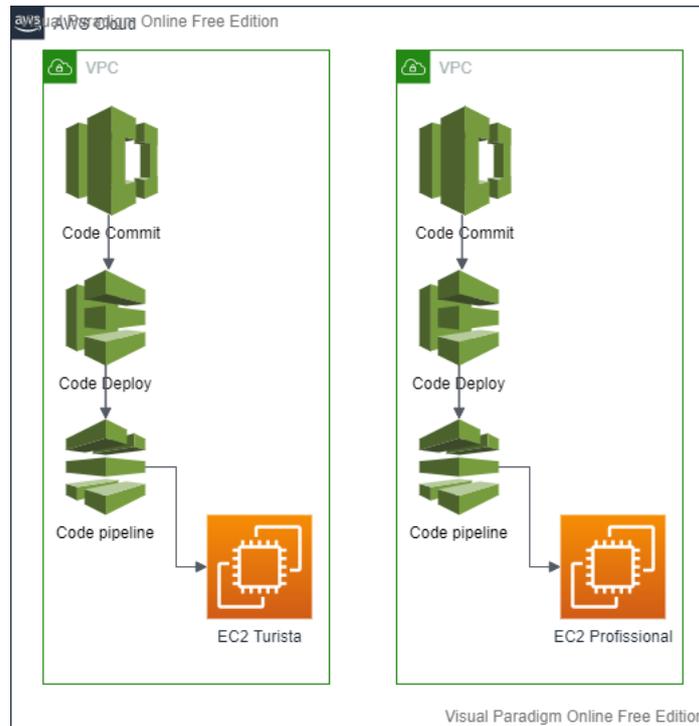


Figura 7: Arquitetura de publicação das apps

```

<ion-content color="warning">
  <ion-grid text-center>
  </ion-grid>
  <ion-card class="welcome-card">
    <ion-card-header>
      <ion-card-title>Payment</ion-card-title>
      <ion-row>
        <h2>{{ tripName | uppercase }} {{ paymentAmount | currency : 'EUR' }}</h2>
      </ion-row>
    </ion-card-header>
    <ion-card-content>
      <form action="/" method="post" id="payment-form">
        <div>
          <div id="card-number">
          </div>
          <div id="card-expiry">
          </div>
          <div id="card-cvc">
          </div>
          <!-- Used to display Element errors -->
          <div id="card-errors" role="alert"></div>
        </div>
        <ion-button type="submit" color="warning" expand="full">Pay Trip</ion-button>
      </form>
    </ion-card-content>
  </ion-card>
</ion-content>

```

Figura 8: Código HTML para um formulário do Stripe

The screenshot shows the Stripe dashboard interface. At the top, there is a search bar and navigation links like 'Criar', 'Ajuda', and 'Modo de teste'. The main section is titled 'Pagamentos' and shows a list of transactions. The table has columns for 'VALOR', 'DESCRIÇÃO', 'CLIENTE', and 'DATA'. Each row represents a payment of 0.60 euros, described as 'Example charge', with various dates in May, November, and October 2021. A sidebar on the left lists categories like 'Fraude e risco', 'Faturas', 'Assinaturas', 'Cotações', 'Links de pagamento', and 'Pedidos'.

VALOR	DESCRIÇÃO	CLIENTE	DATA
€ 0,60	OK ✓ Example charge		6 de mai. 21:25
€ 0,60	OK ✓ Example charge		6 de mai. 21:25
€ 0,60	OK ✓ Example charge		6 de mai. 09:55
€ 0,60	OK ✓ Example charge		10 de nov. de 2021 17:43
€ 0,60	OK ✓ Example charge		10 de nov. de 2021 17:38
€ 0,60	OK ✓ Example charge		31 de out. de 2021 21:59
€ 0,60	OK ✓ Example charge		27 de out. de 2021 12:41
€ 0,60	OK ✓ Example charge		27 de out. de 2021 12:39
€ 0,60	OK ✓ Example charge		23 de out. de 2021 10:12
€ 0,60	OK ✓ Example charge		23 de out. de 2021 05:37

Figura 9: Lista de movimentos no modo de testes no Stripe

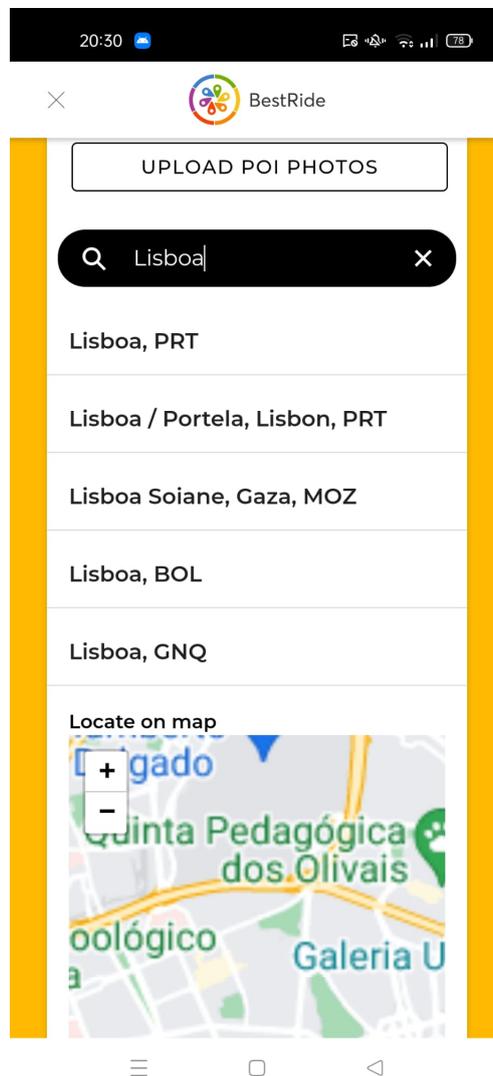


Figura 10: Exemplo de aplicação do ArcGIS

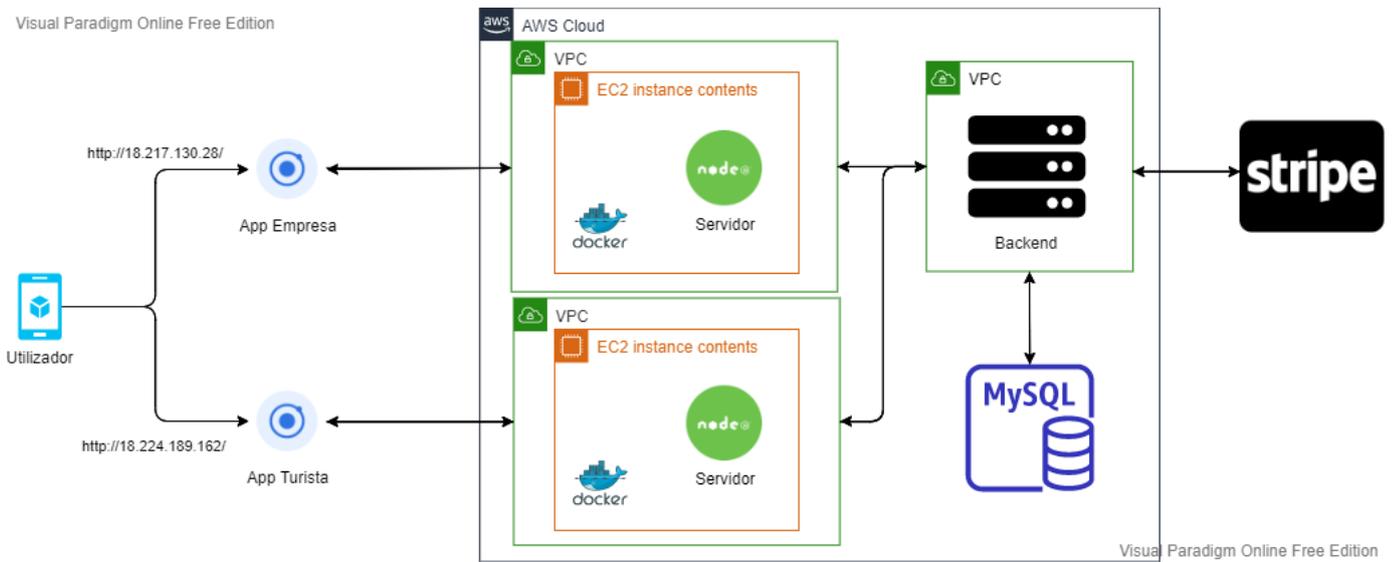


Figura 11: Diagrama da arquitetura geral do frontend

## Calendário TFC

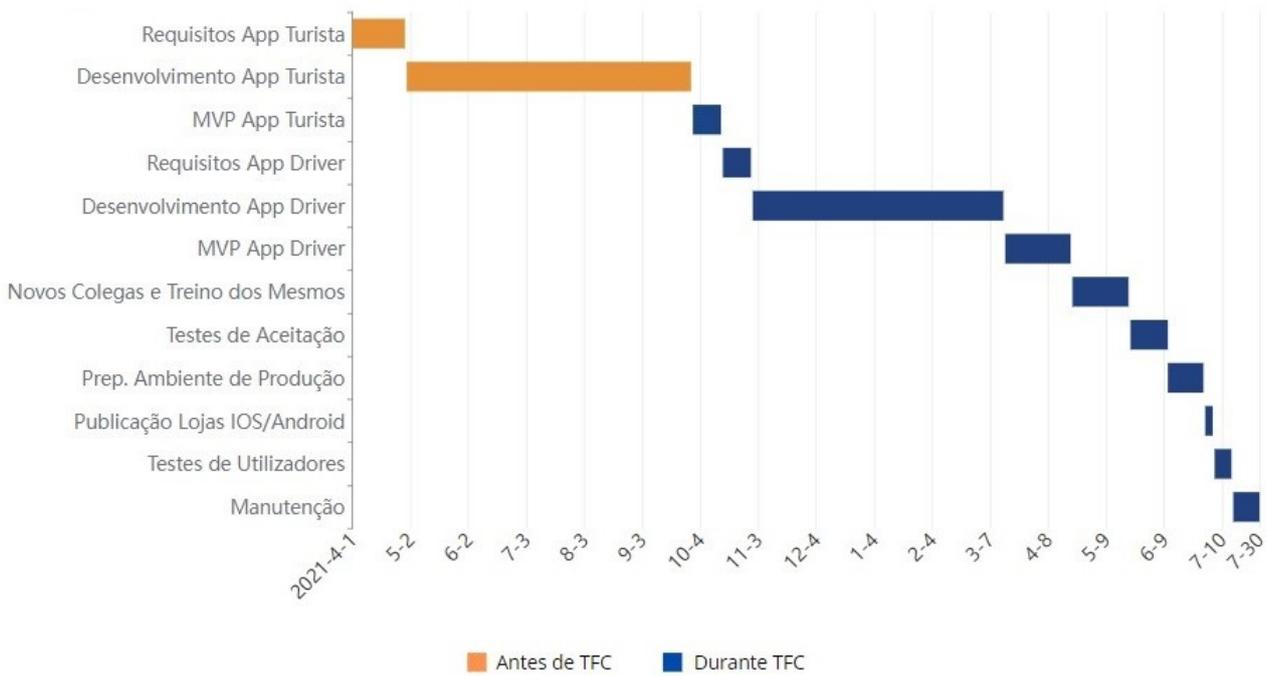


Figura 12: Cronograma TFC

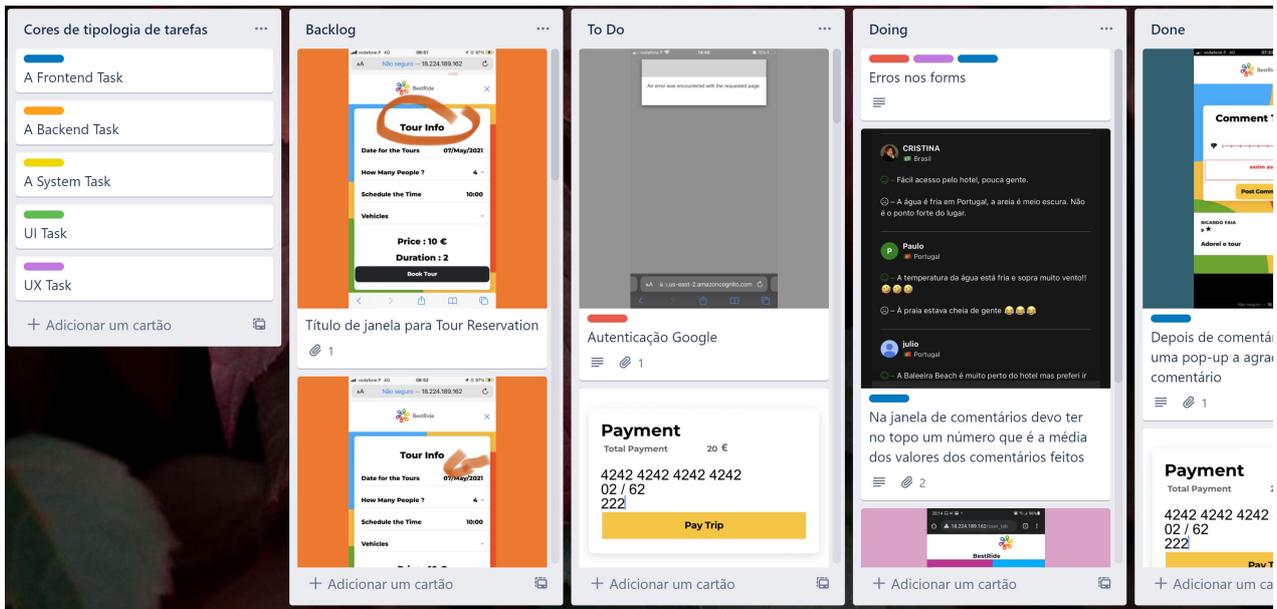


Figura 13: Plano de testes no trello para app turista

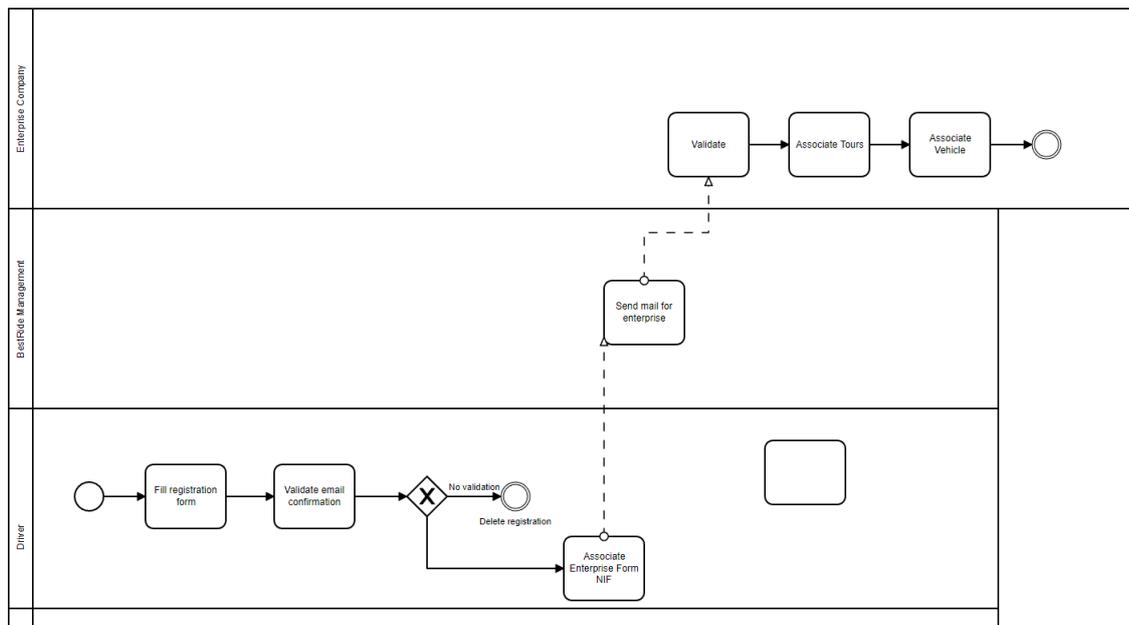


Figura 14: Detalhe de um processo BPMN

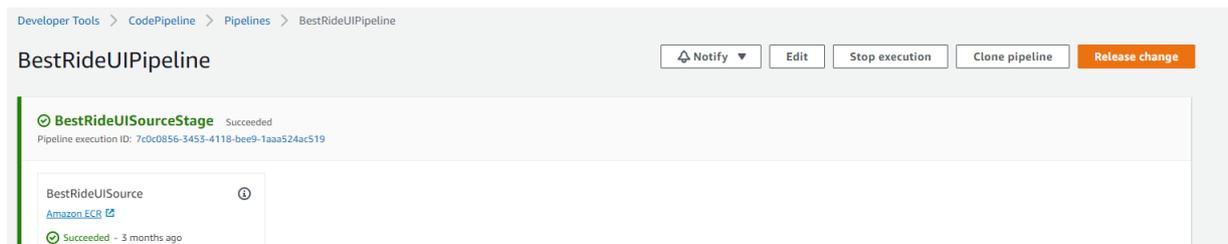


Figura 15: Deploy das aplicações

```
capacitor.config.ts
capacitor.config.ts > ...
1  import { CapacitorConfig } from '@capacitor/cli';
2
3  const config: CapacitorConfig = {
4    appId: 'bestride.professional',
5    appName: 'BestRide Professional',
6    webDir: 'www',
7    bundledWebRuntime: false,
8
9    server: {
10     url: "http://18.217.130.28",
11     cleartext: true
12   }
13 };
14
15 export default config;
16
```

Figura 16: Exemplo de configurações para a aplicação do profissional

## Anexo 2 - Ecrãs Desenvolvidos

### Criação de conta de turista

The image displays two sequential screenshots of the 'Create Account' form in the BestRide application. The left screenshot shows the top portion of the form, including fields for Name, Date of Birth, Country (Portugal...), Number, Postal Code, Gender, Street Address line 1, City, Email, and Password. The right screenshot shows the bottom portion of the form, including fields for Postal Code, Gender, Street Address line 1, City, Email, Password, Repeat password, a Terms and Conditions button, a checkbox for 'I agree with the terms and conditions', and a 'Create Account' button.

Figura 17: Criação conta turista

## Login Turista

The image shows a mobile login form for 'BestRide'. At the top, there is a logo consisting of a colorful circular icon with four petals and the text 'BestRide' to its right. Below the logo is a white rectangular box with a colorful border (yellow, orange, green, red) containing the title 'Login'. Inside this box are two input fields: 'Email' and 'Password'. The 'Password' field has a small eye icon on the right side. Below the input fields is a red button with the text 'Login'. Below the white box are two more buttons: 'Sign up with Google' with a Google 'G' logo and 'Sign up with Facebook' with a Facebook 'f' logo. At the bottom of the form, there is a link that says 'Dont have an account? Sign up' and another link that says 'Forgout your password?'. The background of the form is a mix of orange and blue colors with a large blue circular shape on the right side.

Figura 18: Login Turista

## Recuperar Conta

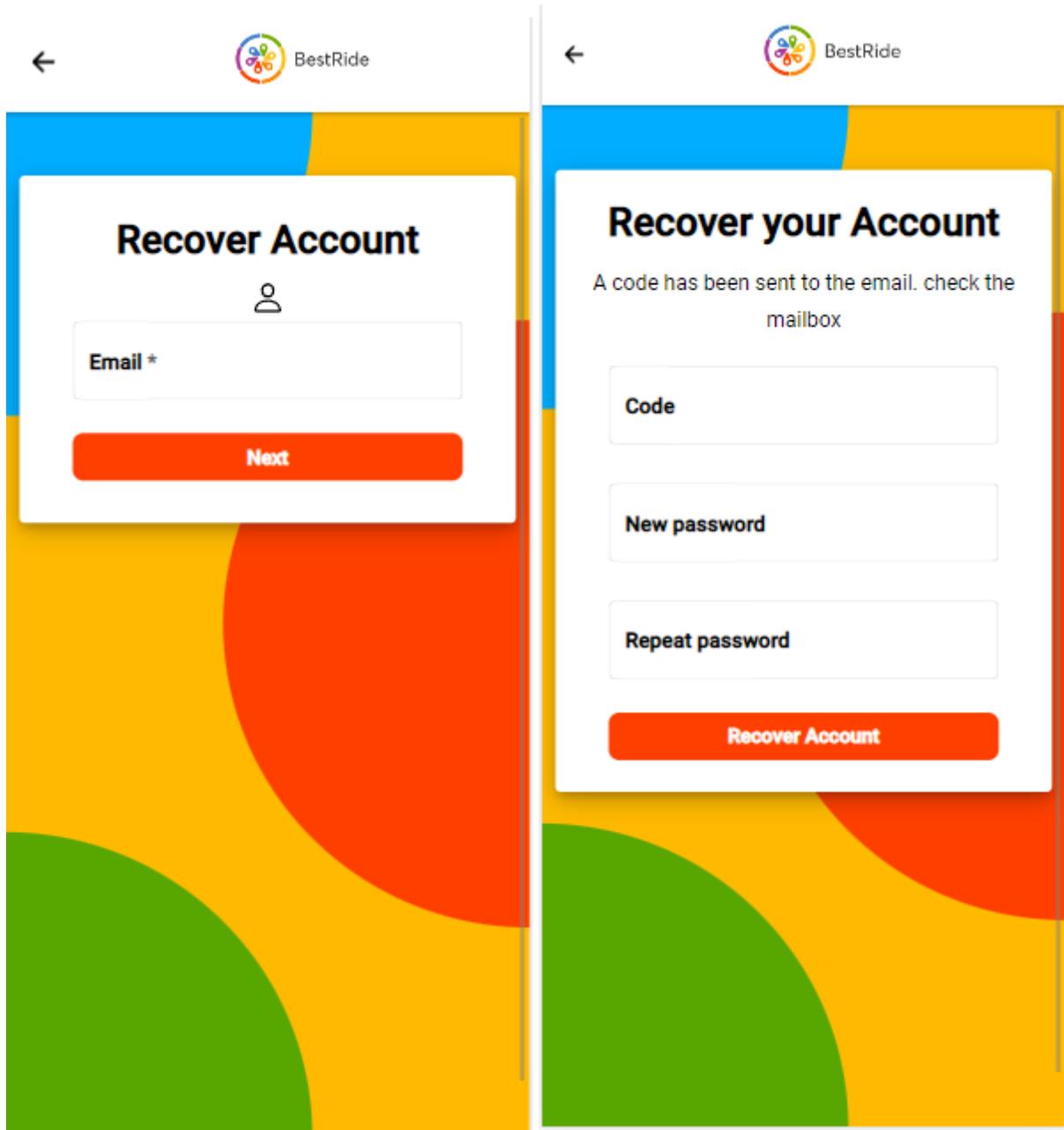


Figura 19: Recuperação Conta Turista

## Ecrã definições

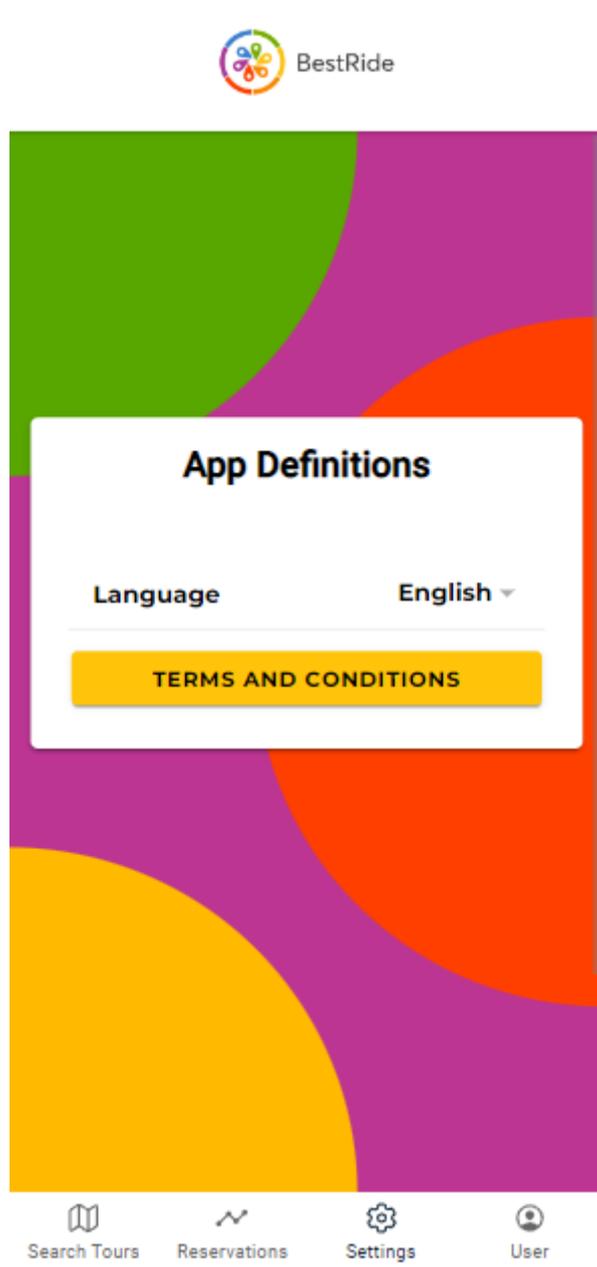


Figura 20: Mudar Idioma da App

## Alterar Dados e Eliminar Conta

The screenshot shows the 'USER DEFINITIONS' form in the BestRide app. The form is titled 'USER DEFINITIONS' and contains the following fields and buttons:

- Name:** Pedro Costa
- City:** Teste City
- Email:** pedrocostaalves@live.com.pt
- Phone Number:** 193182202
- Street address:** Adress Teste

Below the fields are two buttons:

- Update Account** (orange button)
- Change Password** (blue button)

At the bottom of the form are two circular icons:

- A red circle with a black trash can icon (delete).
- A yellow circle with a black arrow icon (edit).

The bottom navigation bar contains four icons and labels:

- Search Tours
- Reservations
- Settings
- User

Figura 21: Eliminar Conta / Alterar Dados

## Lista de Tours



Figura 22: Listagem de Tours

## Comentário do Tour

✕



Historical Tour

Rating 4.6

**Ricardo Faia**



Portugal

---

**Adorei o tour**

---

**Rui**



Portugal

---

assim assim

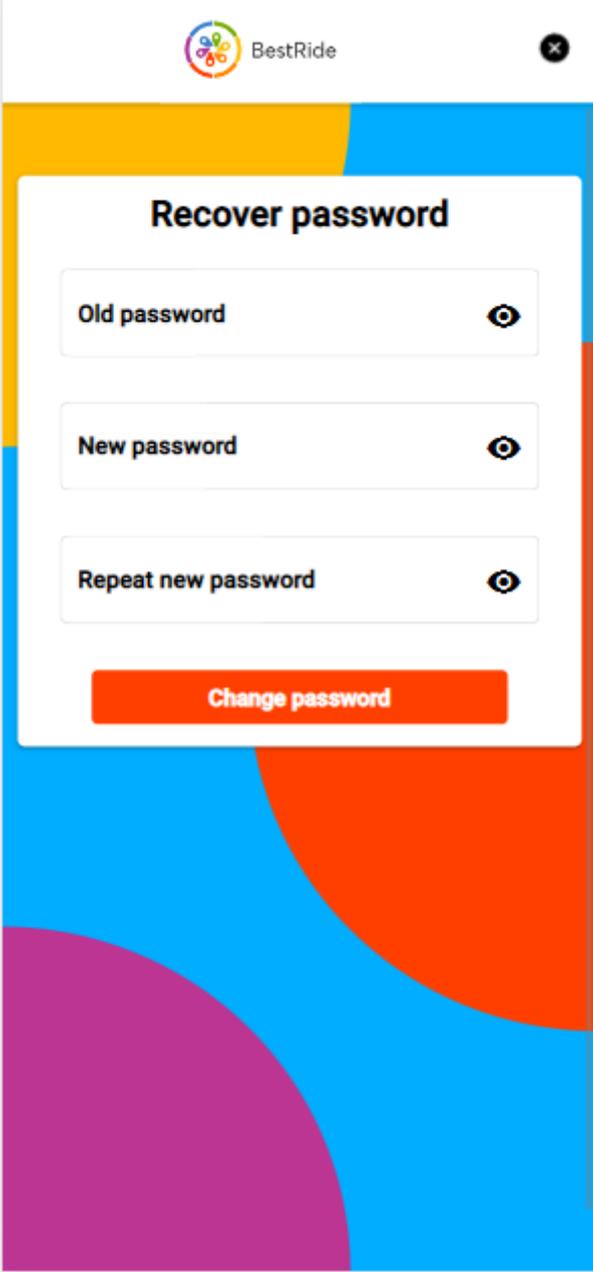
Figura 23: Comentário de um Tour

## Booking do Tour

The image shows a mobile application interface for booking a tour. At the top, there is a header with the 'BestRide' logo and a close button. The main content is a white card with a blue and yellow background. The card is titled 'Book Tour' and 'Historical Tour'. Below the title, there are four input fields: 'Date for the Tours', 'How Many People?' (with a dropdown arrow), 'Schedule the Time', and 'Vehicles' (with a dropdown arrow). Below these fields, the price is listed as 'Price : 10 €' and the duration as 'Duration : 2'. At the bottom of the card is a red button labeled 'Next'.

Figura 24: Formulário para booking do roteiro antes do pagamento

## Alterar password



The image shows a mobile application interface for changing a password. At the top, there is a logo for 'BestRide' and a close button. The main content is a white card titled 'Recover password'. It contains three input fields: 'Old password', 'New password', and 'Repeat new password'. Each field has a small eye icon to the right, indicating a visibility toggle. Below the fields is an orange button labeled 'Change password'. The background of the app is a colorful abstract design with yellow, blue, orange, and purple shapes.

Figura 25: Formulário para alterar a password antiga

## Ecrã pagamento do tour

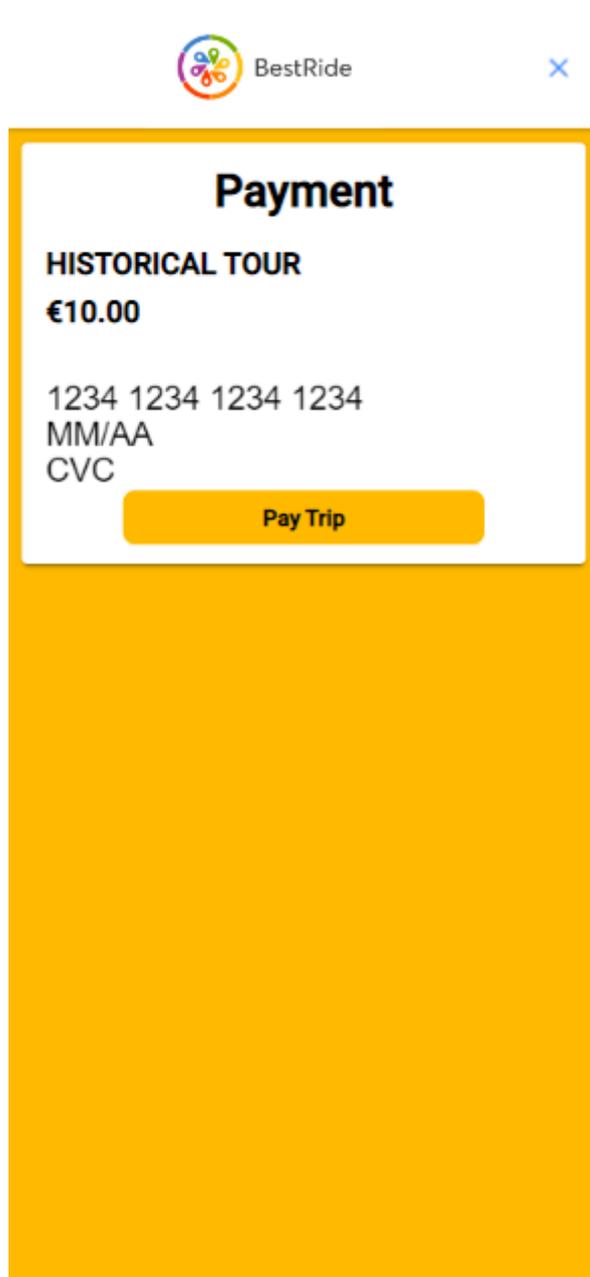
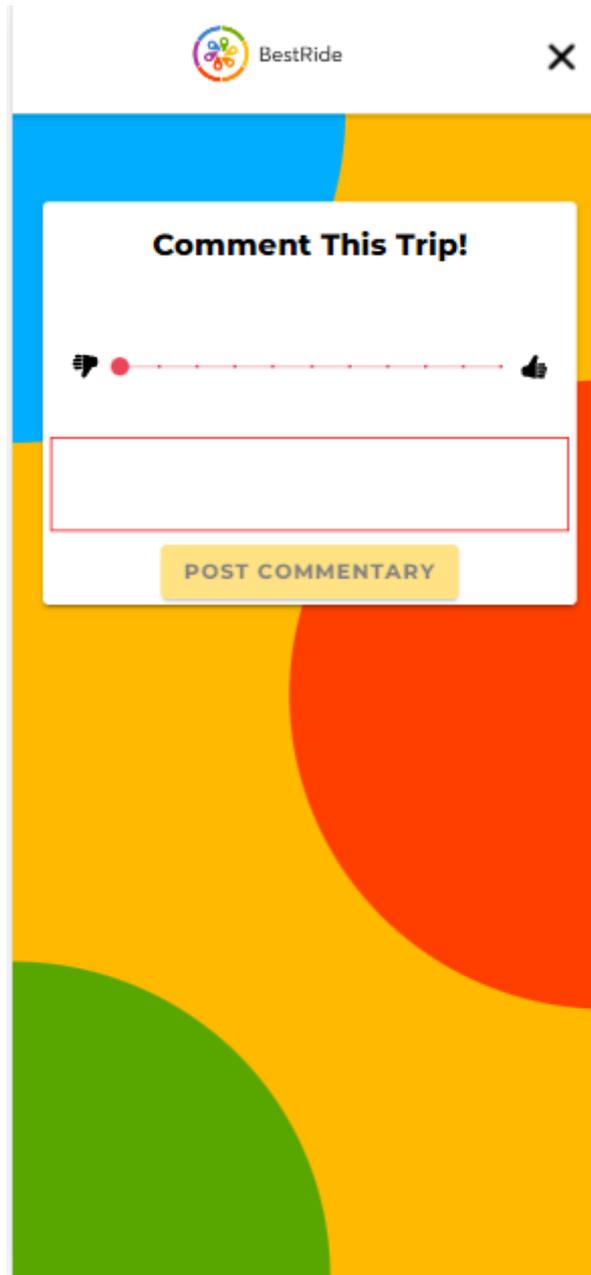


Figura 26: Pagamento do tour com Stripe

## Ecrã para comentar tour



The image shows a mobile application interface for commenting on a trip. At the top, there is a logo for 'BestRide' and a close button (X). The main content area has a white background with a blue and yellow header. The title 'Comment This Trip!' is centered. Below the title is a progress indicator consisting of a horizontal line with a red dot and thumbs-up/down icons. A text input field is positioned below the progress indicator, and a yellow button labeled 'POST COMMENTARY' is at the bottom of the form. The background of the app is decorated with large, colorful abstract shapes in blue, yellow, orange, and green.

Figura 27: Formulário para comentar o tour

## Criar Empresa ou Condutor

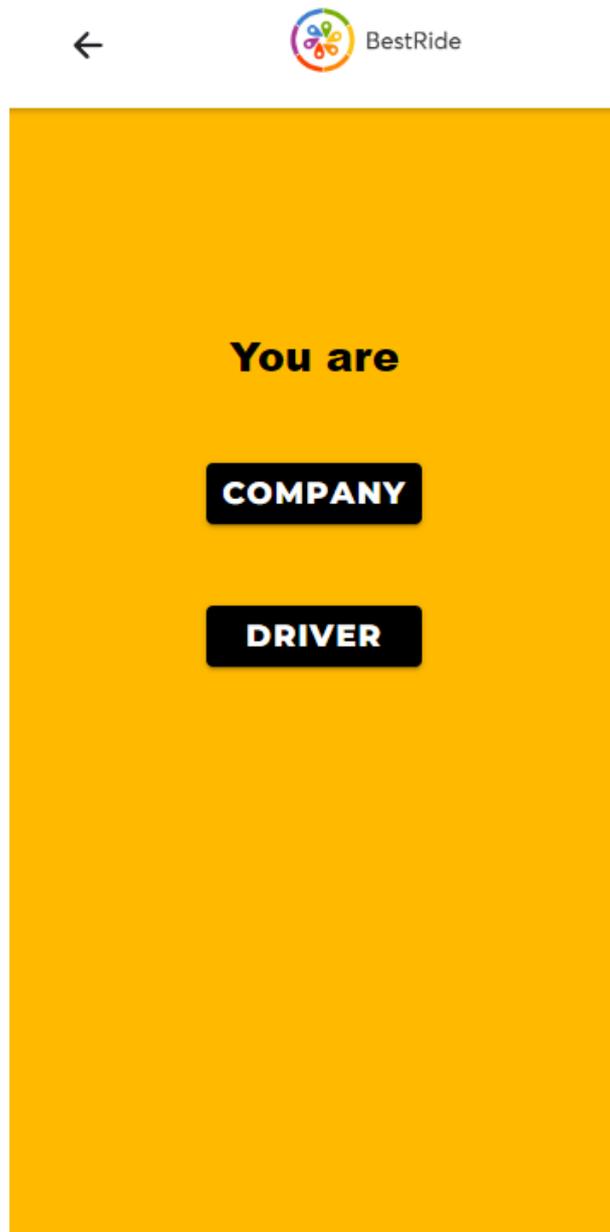


Figura 28: Opção de Criar um Condutor ou uma Empresa

The image displays two side-by-side screenshots of the 'Driver Form' in the BestRide application. Both screenshots feature the BestRide logo and a back arrow at the top.

**Left Screenshot: Driver Form**

- First and Last Name
- Email
- Password (with an eye icon for visibility toggle)
- Confirm Password (with an eye icon for visibility toggle)
- Gender (dropdown menu)
- Birth Date (with a calendar icon)
- Guide People with special needs (checkbox)

**Right Screenshot: Driver Form**

- UPLOAD BI DOCUMENT (blue button)
- NIF
- Citizen Card
- Nationality
- Tourism Courses or Certifications
- Driver Licence
- ANCAT Licence Number

Figura 29: Formulário de Criar um Condutor

The image displays two mobile application screens for creating a company account, set against a red background. Both screens feature the BestRide logo and a back arrow at the top.

**Left Screen: Create Company Account**

- Title: **Create Company Account**
- Field: **Email**
- Button: **NEXT**

**Right Screen: Company Form**

- Field: **Company Name**
- Section: **Rnatt Number**
- Fields: **Code** / **Year**
- Field: **Address**
- Field: **City**
- Fields: **Postal** and **Country** (dropdown menu showing **Portugal**)
- Fields: **Phone** (dropdown menu showing **Port...**) and **Phone**

Figura 30: Formulário de Criar uma Empresa

## Login Empresa/Condutor



### Welcome Back!

Don't have an account? [Sign up](#)  
[Forgot your password?](#)

Figura 31: Login, com opção de usar Facebook ou Google

### Detalhes/Modificar/Editar Conta

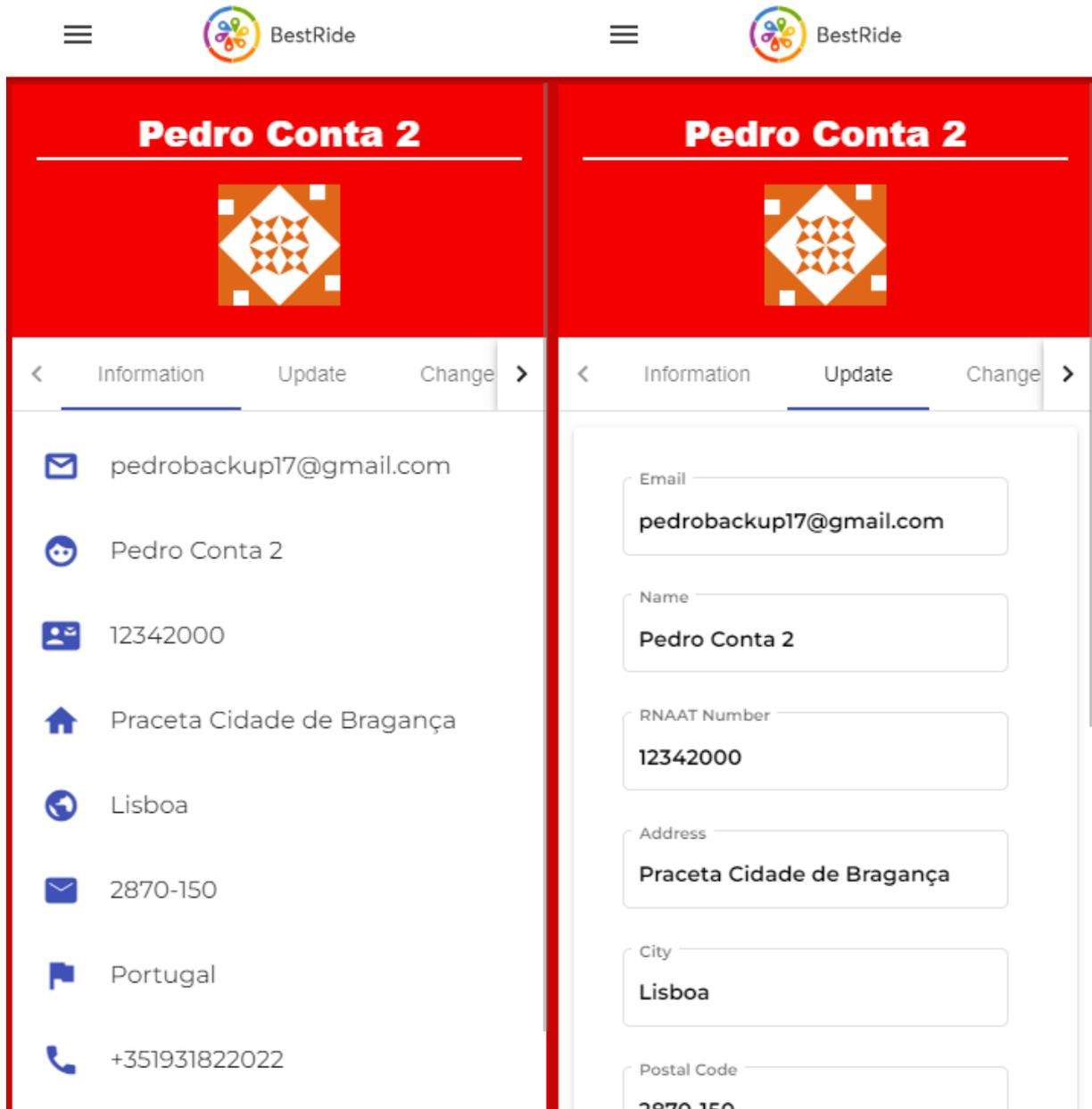


Figura 32: Alterar Dados / Eliminar Conta

## Criar e Listar Veículos

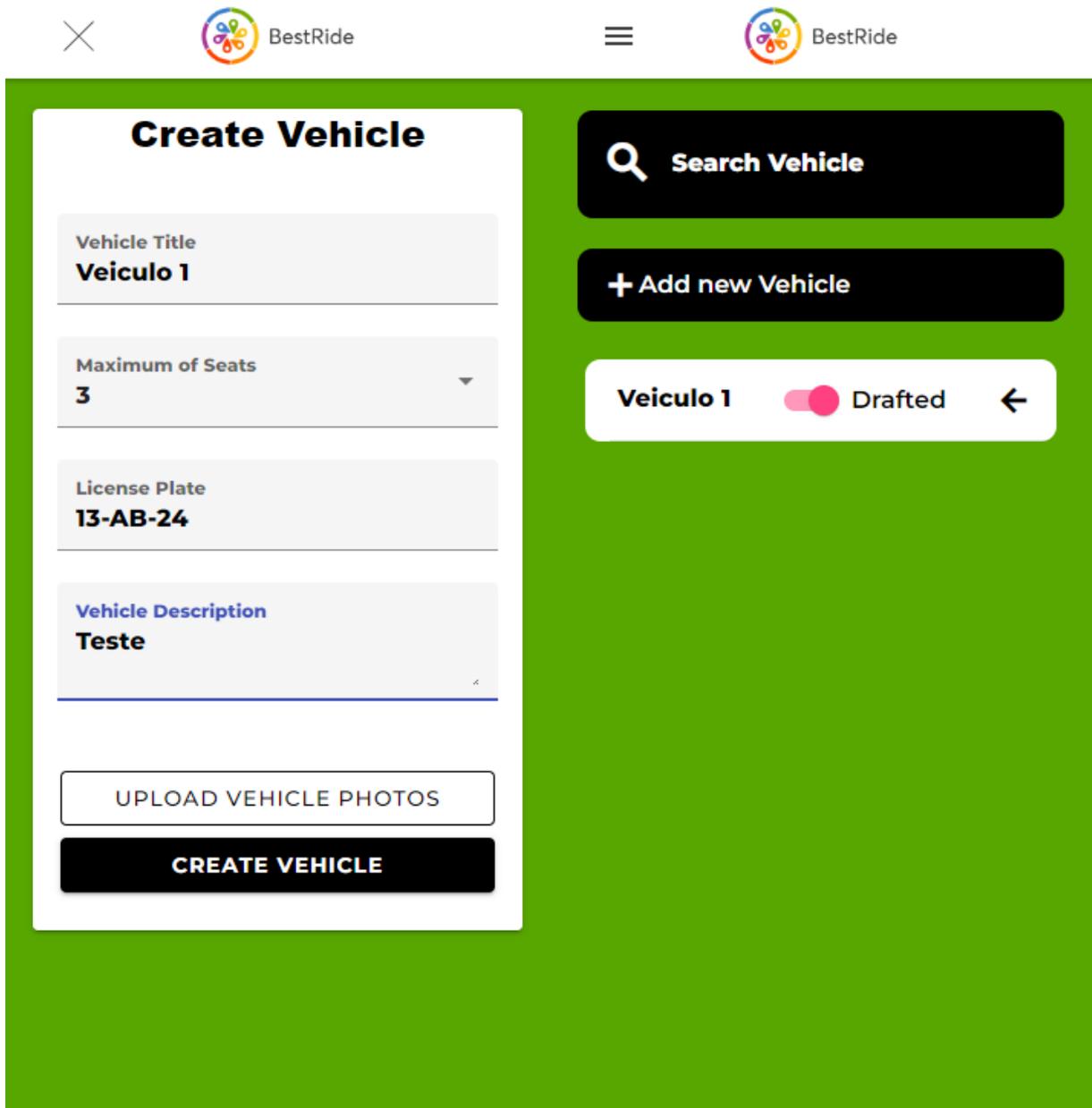


Figura 33: Criar e Listar Veículos associados a uma Empresa

## Estatísticas da Empresa

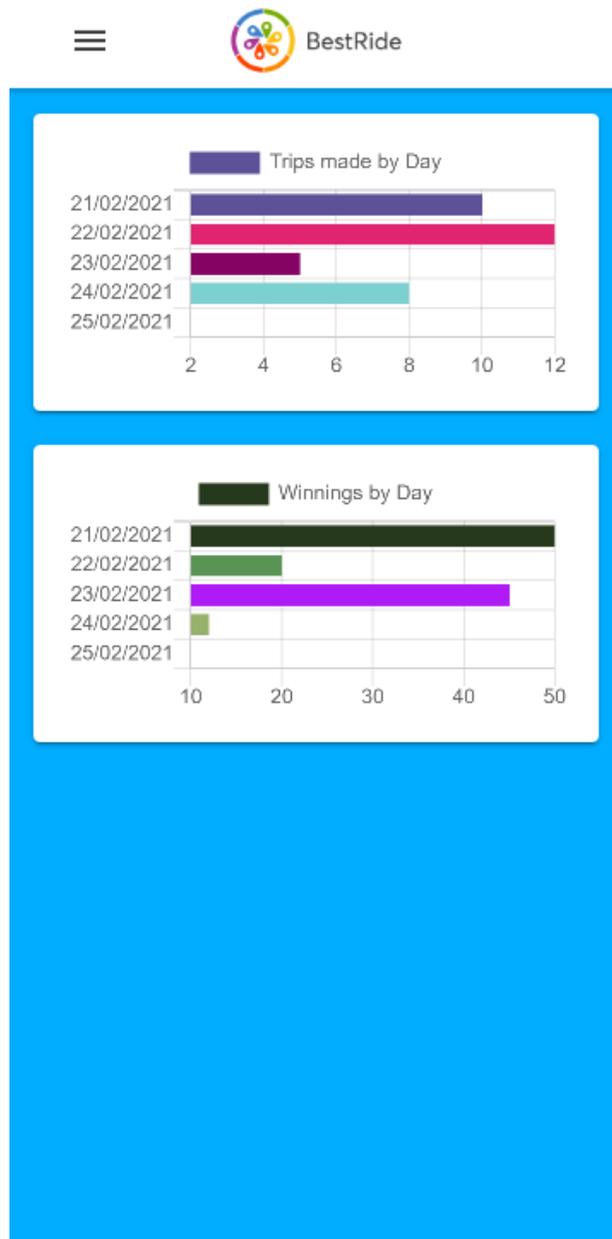


Figura 34: Estatísticas associada a Empresa

## Criação de Roteiros

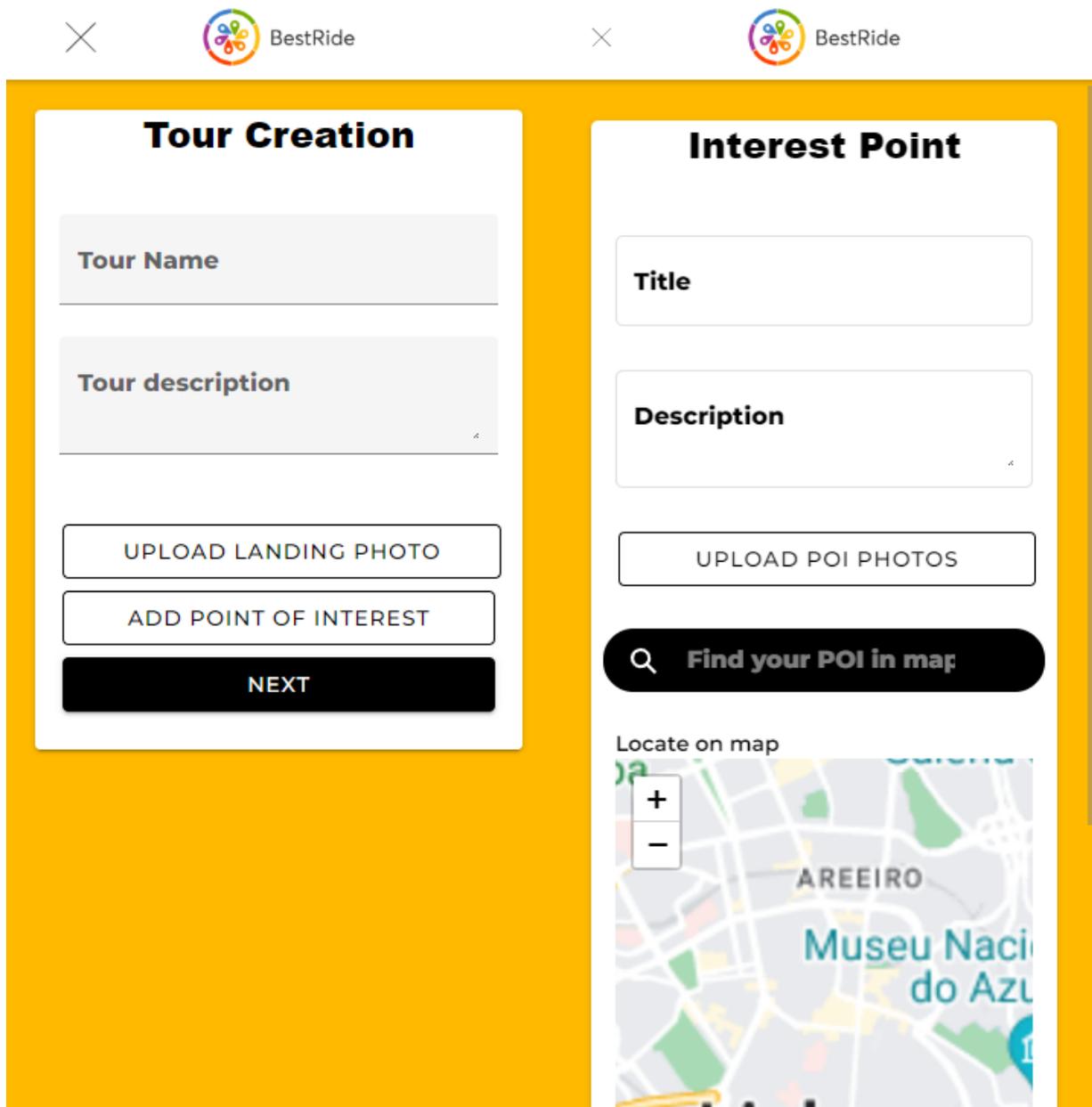


Figura 35: Criação de Roteiros com Pontos de Interesse Associados

## Associar Condutores e Veículos

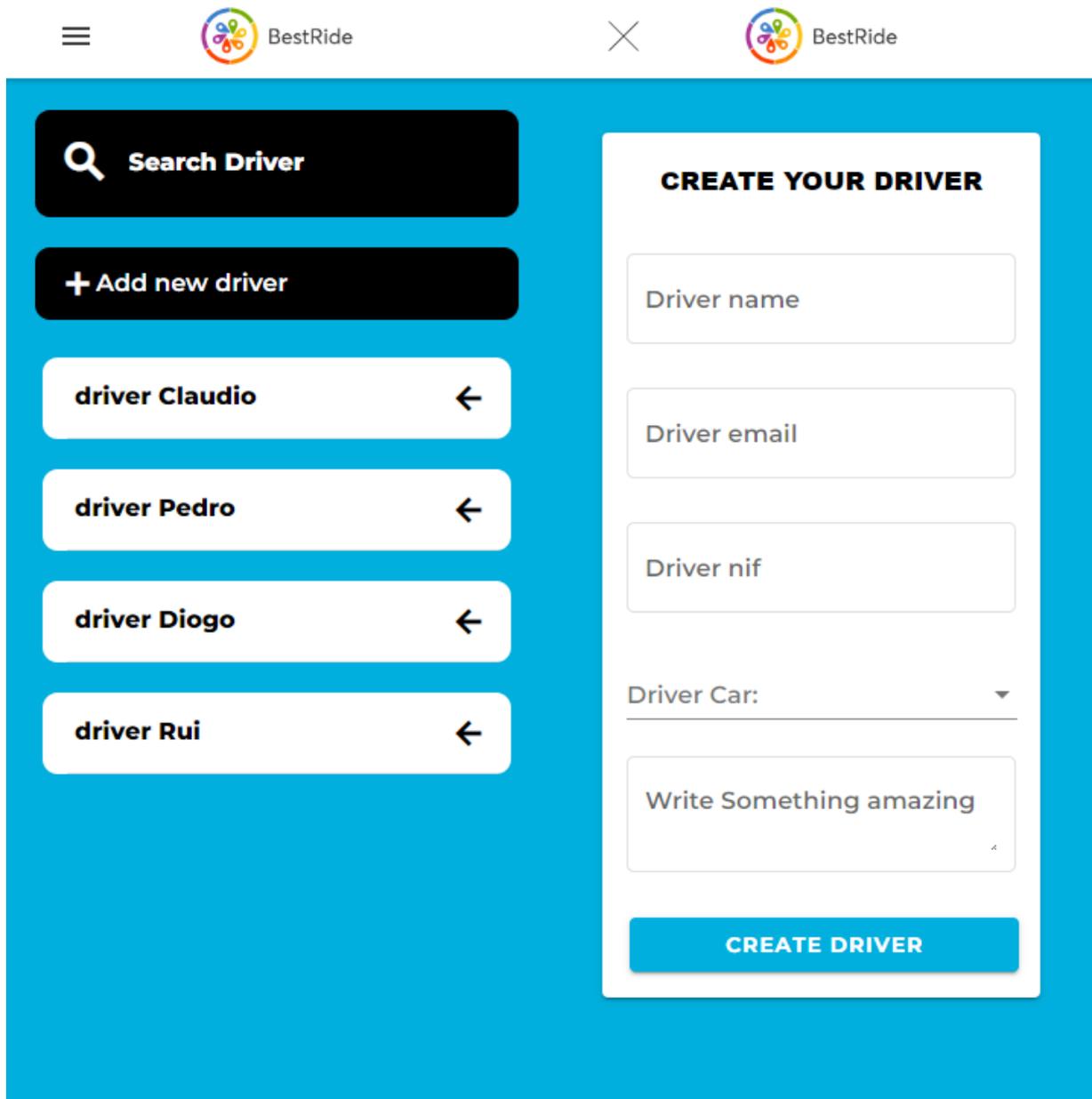


Figura 36: Empresa pode associar condutores e veículos

## Aplicação deve ser Responsiva

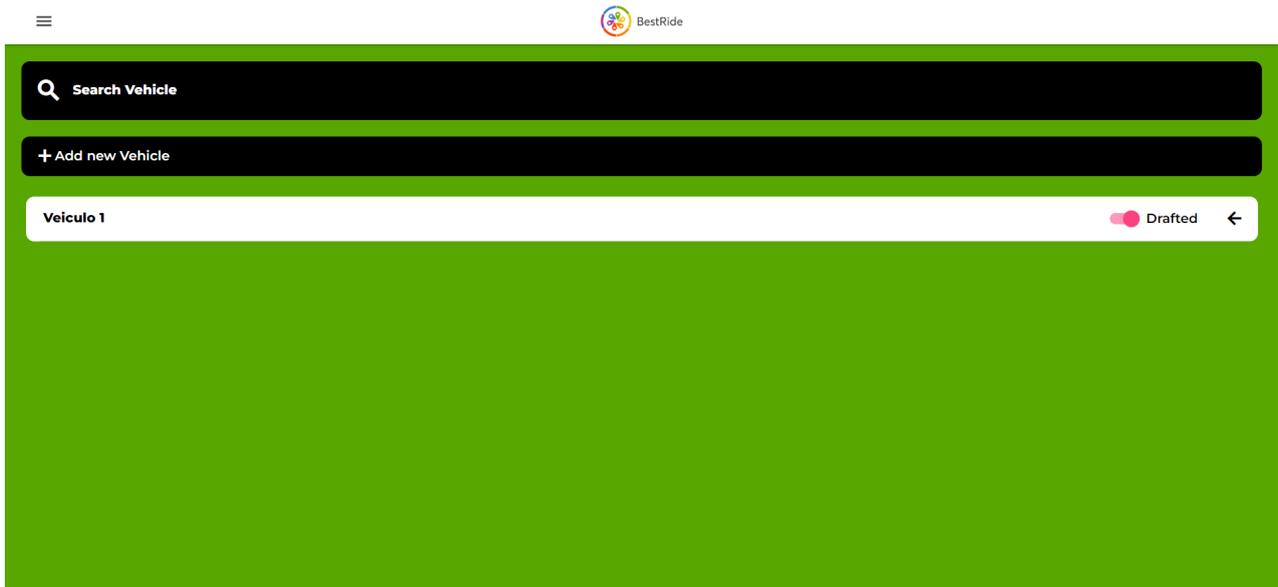


Figura 37: Resposta da aplicação a um ecrã de PC

# Glossário

**App** Aplicação/Software mobile. 54

**AWS** Amazon Web Services. 54

**CSS** Cascading Style Sheets. 54

**HTML** HyperText Markup Language. 54

**JS** Javascript. 54

**LEI** Licenciatura Engenharia Informática. 54

**TFC** Trabalho Final de Curso. 54