

Um caso de estudo sobre o desenvolvimento de aplicações Web com Angular e .NET em ambiente empresarial.

Miguel dos Santos Serôdio Serrano de Matos

Relatório de Estágio para obtenção do Grau de Mestre em
Engenharia Informática
(2º ciclo de estudos)

Orientador: Nuno Gonçalo Coelho Costa Pombo
Orientador da Latitude: João José Teles Gouveia

Junho de 2024

Declaração de Integridade

Eu, Miguel dos Santos Seródio Serrano de Matos, que abaixo assino, estudante com o número de inscrição M12765 do 2º ciclo de Engenharia Informática da Faculdade de Engenharia, declaro ter desenvolvido o presente trabalho e elaborado o presente texto em total consonância com o Código de Integridades da Universidade da Beira Interior.

Mais concretamente afirmo não ter incorrido em qualquer das variedades de Fraude Académica, e que aqui declaro conhecer, que em particular atendi à exigida referenciação de frases, extratos, imagens e outras formas de trabalho intelectual, e assumindo assim na íntegra as responsabilidades da autoria.

Universidade da Beira Interior, Covilhã 13/06/2024

Agradecimentos

Gostaria de expressar os meus sinceros agradecimentos a todas as pessoas que contribuíram para a realização deste trabalho.

Primeiramente, desejo agradecer ao meu orientador, Nuno Pombo, pela orientação valiosa, apoio incansável e pela partilha do seu conhecimento ao longo deste projeto.

Quero expressar a minha sincera gratidão ao meu supervisor na empresa, João Gouveia. O seu apoio, orientação e vasta experiência foram indispensáveis para o meu sucesso e desenvolvimento enquanto profissional dentro da empresa.

Finalmente, quero manifestar aos meus amigos e à minha família, os meus profundos agradecimentos pelo incentivo constante, compreensão e apoio emocional ao longo desta jornada.

Resumo

Este relatório de estágio foi elaborado no âmbito da cadeira de 2º semestre, Dissertação ou Estágio em Engenharia Informática. O documento dá uma visão geral da jornada do estagiário como um *Full Stack Developer* na empresa Latitudde - Digital Enablers, LDA, descrevendo os obstáculos enfrentados e as lições aprendidas durante o desenvolvimento de aplicações *web*. Durante este estágio Angular foi usado para *frontend*, .NET para *backend*, SQL Server para a base de dados e GitLab para a organização e gestão do código.

Além de uma visão geral dos projetos, o relatório aprofunda-se em aspetos técnicos específicos, como a utilização do ASP.NET Identity para a gestão de utilizadores e o uso de componentes do PrimeNG para a construção de plataformas responsivas e eficientes.

Durante este estágio, procurou-se aplicar os conhecimentos adquiridos ao longo do curso de Engenharia Informática, enfrentando desafios práticos e desenvolvendo competências essenciais na área da engenharia informática.

Palavras-chave

Angular, ASP.NET core, Base de dados, Frontend, Backend.

Abstract

This internship report was written as part of the 2nd semester course, Dissertation or Internship in Computer Engineering. The document gives an overview of the trainee's journey as a *Full Stack Developer* at Latitudde - Digital Enablers, LDA, describing the obstacles faced and lessons learned during the development of web applications. During this internship Angular was used for *frontend*, .NET for *backend*, SQL Server for the database and GitLab for code organization and management.

In addition to an overview of the projects, the report goes into more detail on specific technical aspects, such as the use of ASP.NET Identity for user management and the use of PrimeNG components to build responsive and efficient platforms.

During this internship, the aim was to apply the knowledge acquired during the Computer Science degree, facing practical challenges and developing essential skills in the field of computer engineering.

Keywords

Angular, ASP.NET core, Database, Frontend, Backend.

Contents

1	Introdução	1
1.1	Enquadramento	1
1.2	Objetivos	1
1.3	Desafios Enfrentados	2
1.4	Organização do Documento	3
2	Descrição da Empresa	5
2.1	Introdução	5
2.2	Visão Geral	5
2.3	Serviços e Tecnologias	5
3	Plano de Estágio	7
3.1	Introdução	7
3.2	Plano de Trabalho	7
3.3	Cronograma	7
4	Estado da Arte	9
4.1	Introdução	9
4.2	Frontend	9
4.3	Backend	10
4.4	Full Stack	10
4.5	Tendências Emergentes	11
4.6	Conclusão	11
5	Tecnologias Utilizadas	13
5.1	Introdução	13
5.2	Linguagens	13
5.2.1	HTML	13
5.2.2	CSS	13
5.2.3	JavaScript	13
5.2.4	C#	13
5.2.5	SQL	14
5.2.6	LINQ	14
5.3	Ferramentas	14
5.3.1	SQL Server Managment Studio 19	14
5.3.2	Visual Studio Code	14
5.3.3	Visual Studio 2022	14
5.3.4	Swagger	14
5.3.5	Git	15
5.3.6	GitLab	15

5.3.7	Microsoft Teams	15
5.3.8	Figma	15
5.4	Frameworks	15
5.4.1	Angular	15
5.4.2	ASP.NET Core	15
6	Projetos	17
6.1	Projetos Formativos	17
6.1.1	Angular	17
6.1.2	ASP.NET Core	17
6.2	Projeto Profissional	18
6.2.1	Descrição da Aplicação	18
6.2.2	Estrutura da aplicação:	18
6.2.3	Recolha de Requisitos	19
6.2.4	Arquitetura do sistema	20
6.2.5	Metodologia de Trabalho	21
6.2.6	Desenvolvimento da Aplicação	22
6.2.7	Desafios enfrentados	26
6.2.8	Trabalho Futuro	27
7	Conclusão	29
	Bibliografia	31

List of Figures

2.1	Logo da Latitudde	5
3.1	Cronograma	7
6.1	Estrutura do Projeto	21
6.2	Estrutura da Base de Dados	23
6.3	Página dos Motoristas	24
6.4	Página dos Utilizadores	25
6.5	Página dos Veículos	25
6.6	Página dos Bilhetes	26

List of Tables

6.1	Requisitos Funcionais dos Clientes	20
6.2	Requisitos Funcionais dos Administrador	20
6.3	Requisitos Não Funcionais	20

Lista de Acrónimos

UBI	Universidade da Beira Interior
API	Application Programming Interface
IDE	Integrated Development Environment
HTML	HyperText Markup Language
CSS	Cascading Style Sheets
SQL	Structured Query Language
LINQ	Language Integrated Query
CRUD	Create, Read, Update, and Delete
IA	Inteligência Artificial

Chapter 1

Introdução

1.1 Enquadramento

Durante este estágio, houve uma fase formativa que ocorreu na plataforma Udemy[1]. As plataformas de *e-learning* como a Udemy têm como objetivo proporcionar acesso flexível e acessível à educação, permitindo que os utilizadores adquiram novas competências e conhecimentos em diversas áreas.

Esta experiência permitiu aprofundar os conhecimentos do estagiário nas várias áreas do desenvolvimento *web*, com especial ênfase nas tecnologias Angular e ASP.NET core.

Atualmente, está envolvido na construção de uma plataforma que facilita a gestão e utilização de um sistema de transporte público, tanto para os administradores como para os clientes.

O projeto abrange duas secções distintas:

- **Backoffice:** Focada na administração interna da plataforma, permitindo a gestão de operações, frotas e utilizadores.
- **Área Cliente:** Direcionada aos utilizadores finais, facilitando a consulta de horários, compra de bilhetes e gestão de contas pessoais.

O estagiário esteve envolvido em todas as fases do ciclo de vida deste projeto, desde a recolha dos requisitos técnicos até ao desenvolvimento e implementação. Esta experiência proporcionou-lhe uma visão abrangente e valiosa do processo de desenvolvimento de software, permitindo compreender a importância da definição clara de requisitos, da estruturação adequada do projeto e da colaboração eficaz entre os diferentes membros da equipa.

1.2 Objetivos

Os objetivos deste estágio foram definidos para garantir um desenvolvimento profissional abrangente. Os objetivos incluem:

- Compreender a cultura organizacional e os processos internos da empresa. Adaptar-se ao ambiente de trabalho e estabelecer boas relações com colegas e superiores.
- Aprender e aplicar novas tecnologias e ferramentas utilizadas pela empresa. Desenvolver habilidades práticas, com foco nas melhores práticas da indústria.

- Participar ativamente nos projetos da empresa, contribuindo com ideias e soluções.
- Demonstrar a capacidade de entregar tarefas e cumprir prazos estabelecidos.
- Comunicar de forma eficaz para garantir que todos os membros da equipa estão alinhados com os objetivos do projeto.
- Adaptar às metodologias de trabalho utilizadas pela empresa, como o *Scrum*.
- Refletir sobre as experiências e desafios enfrentados, identificando lições aprendidas e oportunidades de melhoria.

1.3 Desafios Enfrentados

Durante o estágio, foram enfrentados uma série de desafios que contribuíram significativamente para o desenvolvimento profissional e pessoal do estagiário. Esses desafios variaram desde a gestão do tempo até a adaptação a novas tecnologias e metodologias de trabalho. Abaixo, estão descritos os principais desafios enfrentados:

- **Gestão do Tempo:** Conciliar um emprego a tempo inteiro com os estudos de mestrado foi um dos maiores desafios. A necessidade de equilibrar as tarefas do trabalho, as responsabilidades acadêmicas e descanso, exigiu uma gestão de tempo rigorosa e eficiente. A organização e o estabelecimento de prioridades foram cruciais para garantir que todas as responsabilidades fossem cumpridas com qualidade e dentro dos prazos estabelecidos.
- **Primeiro Trabalho na Área de Informática:** Este estágio representou a primeira experiência profissional na área de informática. A adaptação ao ambiente de trabalho, a familiarização com as práticas e ferramentas utilizadas pela empresa, e a necessidade de aplicar conhecimentos teóricos em situações práticas reais, foram desafios significativos.
- **Domínio de Novas Tecnologias:** Durante o estágio, foi proposto o desafio de dominar novas tecnologias e *frameworks*, como Angular para o desenvolvimento *frontend* e ASP.NET core para o *backend*. Tecnologias de que não tinha conhecimentos prévios.
- **Pressão por Cumprimento de Prazos:** A pressão para cumprir prazos rigorosos foi constante. A necessidade de gerir múltiplas tarefas simultaneamente, resolver problemas técnicos imprevistos e garantir a qualidade do trabalho, exigiu resiliência e eficácia.
- **Adaptação a Novas Metodologias de Trabalho:** A empresa utilizava metodologias ágeis para o desenvolvimento de software, como *Scrum*. A adaptação dessas metodologias, participação no *Sprint Planning*, as *Daily Scrums* e nas *Sprint Reviews* foi um desafio inicialmente, mas que rapidamente se transformou numa valiosa experiência de aprendizagem.

1.4 Organização do Documento

De modo a refletir o trabalho que foi realizado, este documento encontra-se estruturado da seguinte forma:

1. O primeiro capítulo - **Introdução** - apresenta o enquadramento do estágio, os seus objetivos, desafios enfrentados e a organização do documento.
2. O segundo capítulo - **Descrição da Empresa** - apresenta a empresa, sua missão, visão e serviços fornecidos.
3. O terceiro capítulo - **Plano de Estágio** - detalha as atividades previstas e o cronograma delas.
4. O quarto capítulo - **Estado da Arte** - descreve a evolução do desenvolvimento web, abordando tópicos como *frontend*, *backend* e o futuro da área.
5. O quinto capítulo - **Tecnologias Utilizadas** - descreve as ferramentas e tecnologias utilizadas durante o estágio, assim como o contexto em que foram aplicadas.
6. O sexto capítulo - **Projetos** - apresenta os projetos desenvolvidos durante o estágio, incluindo objetivos e metodologias.
7. O sétimo capítulo - **Conclusão** - resume as experiências e aprendizagens do estágio.

Chapter 2

Descrição da Empresa

2.1 Introdução

A Latitudde [2] faz parte do Grupo ReadinessIT, uma empresa multinacional especialista em projetos de Transformação Digital, com mais de 400 pessoas, escritórios em três continentes (Portugal, Chile, Peru e Nova Zelândia) e a trabalhar em projetos internacionais em todo o mundo.



Figure 2.1: Logo da Latitudde

2.2 Visão Geral

Com sede no Fundão, a Latitudde é uma empresa tecnológica dedicada ao desenvolvimento de soluções digitais inovadoras. Criada com a missão de capacitar as organizações na sua transformação digital, a empresa atua como um parceiro estratégico para otimizar processos, melhorar a experiência do cliente e aumentar a eficiência operacional através da tecnologia.

2.3 Serviços e Tecnologias

Os projetos desenvolvidos pela Latitudde focam-se em tecnologias *low-code*, como OutSystems e Appian, além de outras tecnologias populares, como React, Angular, Java, .Net, Flutter e Ionic. Os serviços oferecidos vão desde o desenvolvimento de software personalizado até à consultoria tecnológica, integração de sistemas e soluções de *Cloud Computing*.

Chapter 3

Plano de Estágio

3.1 Introdução

Este capítulo irá detalhar a organização e cronograma do estágio realizado na empresa Lattude - Digital Enablers, de modo a dar um melhor entendimento de como foi estruturado.

3.2 Plano de Trabalho

O estágio foi dividido em duas partes. A primeira parte teve como objetivo formar o estagiário nas tecnologias e ferramentas que iria utilizar dentro da empresa. Isso foi realizado por meio da plataforma *e-learning* Udemy, onde foram feitos dois cursos, um focado no desenvolvimento *frontend* com Angular e outro no desenvolvimento *backend* com ASP.NET Core. A segunda parte consistiu na integração num projeto real onde foram aplicados os conhecimentos adquiridos ao longo do período formativo.

3.3 Cronograma

O estágio teve início no dia 05 de Fevereiro de 2024 e está previsto acabar no dia 04 de Novembro de 2024. O estagiário trabalha das 9h às 18h nos dias úteis da semana.

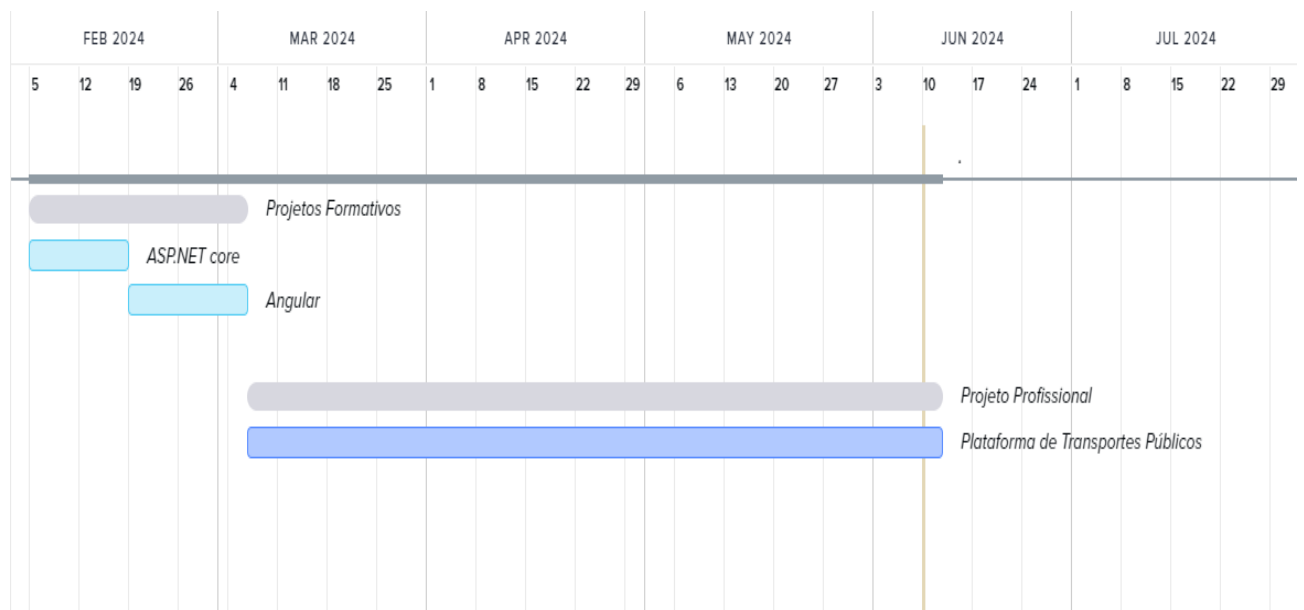


Figure 3.1: Cronograma

Chapter 4

Estado da Arte

4.1 Introdução

Na década de 90 do século passado, mais precisamente em agosto de 1991, Berners-Lee publicou um pequeno resumo do projeto da *World Wide Web* no alt.hypertext newsgroup. Essa data marca o nascimento da Web como um serviço público da Internet[3].

Rapidamente evoluiu de páginas estáticas, com informações fixas e sem capacidade de interação, para páginas dinâmicas, onde o utilizador pode interagir e obter conteúdo personalizado em tempo real. A Internet deixou de ser um ambiente onde a informação era previamente definida e tornou-se um espaço de partilha interativa.

Esta nova fase trouxe uma ênfase significativa na participação do utilizador. Plataformas como blogs e redes sociais tornaram-se populares, permitindo aos utilizadores não só consumir conteúdo, mas também criá-lo e partilhá-lo com outros de forma rápida e fácil.

Isto foi tudo tornado possível devido ao surgimento de novas abordagens e tecnologias de desenvolvimento web. Por exemplo, o uso de linguagens de programação do lado do servidor, como PHP e ASP.NET, permitiu a criação de websites capazes de responder aos pedidos dos utilizadores em tempo real e de fornecer conteúdo personalizado com base nas suas interações.

Além disso, o surgimento de frameworks e bibliotecas como Angular, React e Vue.js, simplificou significativamente o processo de criação de interfaces de utilizador complexas e interativas, permitindo aos programadores concentrar na lógica de negócios.

4.2 Frontend

O *frontend*[4] de uma aplicação web é a parte responsável por tornar a aplicação interativa e funcional. Não só apresenta visualmente o conteúdo, mas também lida com a interatividade e a experiência do utilizador. É a capacidade de transformar simples linhas de código em algo apelativo, atraente e funcional.

Tecnologias como HTML, CSS e JavaScript formam a base sobre a qual as interfaces são construídas. O HTML define a estrutura e o conteúdo da página, enquanto o CSS controla o estilo e a apresentação visual, como cores, fontes e layout. O JavaScript é utilizado para

adicionar interatividade dinâmica, como validação de formulários e manipulação do DOM.

Além das tecnologias fundamentais, as *frameworks* para *frontend* desempenham um papel crucial no desenvolvimento de aplicações web modernas. Algumas conhecidas como React, Vue.js e Angular oferecem conjuntos abrangentes de ferramentas e componentes reutilizáveis que simplificam o desenvolvimento e aceleram a criação de interfaces dinâmicas e responsivas.

4.3 Backend

O *backend* de uma aplicação web desempenha um papel crítico ao lidar com a lógica de negócios e o processamento de dados. Enquanto o *frontend* é responsável por apresentar visualmente a aplicação aos utilizadores, o *backend* opera de forma complementar, fornecendo as funcionalidades essenciais que sustentam as operações da aplicação.

A sua principal responsabilidade é garantir a segurança, escalabilidade e eficiência das operações. Isso significa que para além de receber pedidos do *frontend*, também os processa, manipulando dados, executando operações de autenticação e autorização, e fornecendo respostas adequadas de volta ao utilizador.

As linguagens de programação, tais como C#, Python e Java, são frequentemente escolhidas. No entanto, é crucial realçar que a seleção da linguagem deve ser orientada pelas exigências específicas do projeto, assim como pela experiência e preferências da equipa de desenvolvimento.

Além disso, as *frameworks* de *backend* desempenham um papel crucial na simplificação do desenvolvimento e na construção de sistemas robustos e escaláveis. Elas fornecem estruturas que permitem os programadores se concentrarem na lógica de aplicação, em vez de terem de lidar com detalhes de baixo nível.

4.4 Full Stack

O desenvolvimento *full stack* permite aos programadores dominarem todas as vertentes do desenvolvimento de aplicações web, desde a interface do utilizador até à lógica de negócios e armazenamento de dados. Esta abordagem integral possibilita a criação de soluções completas e funcionais, promovendo a eficiência e a eficácia no processo de desenvolvimento. Estes profissionais estão bem posicionados para enfrentar desafios complexos e oferecer soluções inovadoras que respondem às necessidades dos utilizadores e dos negócios.

4.5 Tendências Emergentes

À medida que o desenvolvimento web evolui, surgem tendências promissoras que moldarão o futuro do setor. Estas tendências não só oferecem novas oportunidades, mas também melhoram significativamente a experiência do utilizador e expandem as capacidades das aplicações.

Uma tendência em destaque é o crescimento das PWAs, *Progressive Web Applications*, que combina o melhor do mundo nativo e mundo web, proporcionando uma experiência de utilizador rápida, confiável e cativante, independentemente do dispositivo ou da conexão à rede. Com recursos como integração nativa, funcionalidade *offline* e notificações, os PWAs estão a tornar-se uma escolha popular para empresas que procuram oferecer experiências de alta qualidade.

Outra tendência é o uso de Inteligência Artificial (IA), que desempenha um papel cada vez mais importante. Desde chatbots de atendimento ao cliente até sistemas de recomendação de conteúdo personalizado, a IA está a transformar a forma como se interage com as aplicações.

Os programadores estão a explorar técnicas como *machine learning* e processamento de linguagem natural para criar experiências mais inteligentes e personalizadas. Isto leva a que se desenvolvam receios de que o seu trabalho, *web developers*, seja substituído pela IA.

Desde o início quando a linguagem CSS foi lançada pela primeira vez em 1996, no Internet Explorer que foram aparecendo ferramentas sem código que espalham estes receios. Em 1998 o Homestead permitiu que as pessoas construíssem páginas da web personalizadas sem escrever uma única linha de código. Na década de 2000, era o WordPress. Na década de 2010, era o Webflow, hoje o grande receio é a IA [5].

No entanto, é importante destacar que os programadores têm mostrado uma notável capacidade de adaptação a essas novas tecnologias ao longo do tempo, encontrando maneiras de integra-las nos seus fluxos de trabalho, aumentando sua produtividade e valor.

Estas são apenas algumas das tendências que estão a moldar um futuro onde a inovação tecnológica e a compreensão das necessidades dos utilizadores impulsionam o desenvolvimento web. Estar atento a estas mudanças e adaptar-se proativamente é essencial para garantir o sucesso contínuo neste campo em constante evolução.

4.6 Conclusão

O desenvolvimento de aplicações web é uma área dinâmica e em constante evolução, impulsionada pela necessidade de soluções digitais cada vez mais sofisticadas e interativas. Com a interligação entre *backend* e *frontend* desempenhando um papel central no desenvolvimento

web, é fundamental compreender os princípios e as melhores práticas associadas a cada um desses componentes.

Chapter 5

Tecnologias Utilizadas

5.1 Introdução

Neste capítulo, são examinadas as tecnologias que foram parte integrante dos projetos realizados durante o estágio. Desde as linguagens de programação até às *frameworks*.

5.2 Linguagens

5.2.1 HTML

HyperText Markup Language (HTML), é a espinha dorsal que define a estrutura básica de uma página web. Permite criar elementos como títulos, parágrafos, listas e links, organizando o conteúdo de forma hierárquica e compreensível. Foi utilizado para criar a estrutura fundamental das páginas, definindo como o conteúdo seria apresentado.

5.2.2 CSS

Cascading Style Sheets (CSS), complementa o HTML ao definir o estilo e a apresentação visual das páginas. Permite controlar aspectos como cores, fontes, espaçamento e *layout*, tornando o conteúdo mais atraente e fácil de ler. Ele desempenha um papel crucial ao aplicar estilos, criando uma aparência consistente e agradável em todas as páginas. Isso inclui desde a definição de cores e fontes até o posicionamento de elementos na página, garantindo uma experiência visualmente atraente e coesa.

5.2.3 JavaScript

JavaScript é uma linguagem de programação que permite adicionar interatividade às páginas web. Foi usado para criar funcionalidades dinâmicas e interativas, como validação de formulários e animações.

5.2.4 C#

C#, é uma linguagem de programação orientada a objetos desenvolvida pela Microsoft. O uso desta linguagem permitiu desenvolver a lógica do servidor, especialmente em conjunto com a *framework* ASP.NET Core, permitindo a criação de Application Programming Interface (API) e a implementação da lógica de negócios.

5.2.5 SQL

Structured Query Language (SQL), é uma linguagem padrão para gerir e manipular bases de dados relacionais. Permite realizar operações Create, Read, Update, and Delete (CRUD), em bases de dados de forma eficiente.

5.2.6 LINQ

Language Integrated Query (LINQ), é uma extensão do C# que facilita a integração entre o código e as instruções SQL. Foi usada para escrever consultas de forma intuitiva e legível, que são automaticamente traduzidas para SQL. Isto simplificou a manipulação de dados e permitiu operações CRUD de maneira mais eficiente e integrada.

5.3 Ferramentas

5.3.1 SQL Server Management Studio 19

SQL Server Management Studio é uma ferramenta que foi usada para gerir bases de dados, executar consultas SQL, e analisar o desempenho, garantindo a eficiência e integridade dos dados.

5.3.2 Visual Studio Code

Visual Studio Code serviu para desenvolver o *frontend* das aplicações em Angular. É um Integrated Development Environment (IDE) leve e flexível desenvolvido pela Microsoft. Oferece suporte nativo para TypeScript, a linguagem de programação utilizada no desenvolvimento de aplicações Angular. Com recursos como IntelliSense para sugestões de código proporcionou um ambiente de desenvolvimento eficiente e produtivo.

5.3.3 Visual Studio 2022

Visual Studio 2022 é um IDE completo para a construção de aplicações em várias linguagens, incluindo C#. No entanto, no contexto deste estágio ele foi utilizado para desenvolver as APIs com ASP.NET Core, proporcionando ferramentas avançadas para testes e *deployment*.

5.3.4 Swagger

O Swagger é uma ferramenta para documentação de APIs RESTful. Com ele, foi possível criar documentação interativa, incluindo detalhes sobre *endpoints*, parâmetros, tipos de resposta e exemplos de uso. Além disso, o Swagger permite testar e explorar a API diretamente na interface da documentação, simplificando o desenvolvimento e a integração.

5.3.5 Git

Git é um sistema de controlo de versão distribuído. Neste estágio, ele foi utilizado para rastrear mudanças no código-fonte, facilitar a colaboração entre membros da equipa, e manter um histórico completo das modificações, garantindo a integridade e rastreabilidade do desenvolvimento.

5.3.6 GitLab

O GitLab[6] é uma plataforma DevOps que oferece repositórios Git e ferramentas para gestão de projetos. Por meio das suas Boards, foi possível centralizar a organização dos projetos, ajudando com o acompanhamento das tarefas e dos problemas, desde a identificação de bugs até a implementação de novas funcionalidades. Isso garantiu que todos os membros da equipe estivessem alinhados e atualizados.

5.3.7 Microsoft Teams

Microsoft Teams é uma plataforma de comunicação e colaboração. No contexto deste estágio, esta aplicação foi utilizado para a comunicação entre os membros da equipa, organização de reuniões, e compartilhamento de ficheiros, melhorando a coordenação e eficiência do trabalho.

5.3.8 Figma

O Figma é uma ferramenta de *design* colaborativo baseada na nuvem, usada para criar interfaces de utilizador. Durante este estágio, esta ferramenta foi utilizada para os programadores poderem seguir os *designs* criados para aplicação pela equipa de *design*. Com o Figma, era possível ver e realizar ajustes rápidos e eficientes nos *designs*.

5.4 Frameworks

5.4.1 Angular

Angular[7] é uma framework front-end desenvolvida pela Google para a criação de aplicações web de página única (SPA) dinâmicas. Durante o estágio, Angular foi utilizado para desenvolver as user interfaces (UI), proporcionando uma experiência rica e responsiva.

5.4.2 ASP.NET Core

ASP.NET Core[8] é uma *framework open-source* ideal para a construção de aplicações modernas e APIs. Ela foi utilizada para desenvolver a lógica do servidor e criar endpoints RESTful, proporcionando uma base sólida e eficiente para a comunicação entre o cliente e o servidor.

Chapter 6

Projetos

6.1 Projetos Formativos

De modo a preparar o estagiário para projetos reais com clientes, a empresa proporcionou uma fase formativa. Durante este período, a plataforma de *e-learning* Udeemy foi utilizada para adquirir conhecimentos nas tecnologias que seriam usadas, Angular para *frontend* e ASP.NET Core para *backend*. Estas formações foram fundamentais para garantir um desempenho eficaz nos projetos.

6.1.1 Angular

O curso **Angular 2(v17+) Typescript do Básico ao Avançado** de Dener Troquatte oferece uma abordagem detalhada e bem estruturada para quem pretende dominar o desenvolvimento *frontend* com Angular e Typescript. No decorrer das 45 horas de formação, foi obtido uma compreensão sólida nos fundamentos de Angular e Typescript. A participação em projetos práticos foi essencial para reforçar a aprendizagem teórica.

Esta formação foi dividida em módulos que abrangem desde a configuração inicial do ambiente de desenvolvimento até à criação de aplicações simples. São abordados conceitos essenciais como a arquitetura de aplicações Angular, a integração com serviços RESTful, a utilização de guardas de autenticação (*Auth Guards*), a criação de diretivas personalizadas, e a gestão de formulários reativos.

Entre os projetos realizados, destacam-se a criação de uma aplicação de gestão de tarefas e um webiste simples de *e-commerce*. Cada projeto foi concebido para aplicar os conceitos aprendidos em cada módulo, proporcionando uma experiência prática e contextualizada.

6.1.2 ASP.NET Core

O curso **Build ASP.NET Core Web Api - Scratch To Finish (.NET8 API)** de Sameer Saini, oferece uma abordagem abrangente para iniciantes no desenvolvimento *backend* e na tecnologia .NET. Ao longo das 18 horas de formação, foram adquiridos conhecimentos básicos sobre o funcionamento do .NET através da realização de um projeto prático.

Este focou-se na criação de uma plataforma onde os utilizadores têm capacidade de se registar e iniciar sessão. Uma vez dentro da aplicação, podem gerir diversas regiões e os percursos turísticos associados a cada uma delas. Mesmo sendo uma implementação simples, ofereceu uma visão introdutória dos fundamentos de .NET.

6.2 Projeto Profissional

Após a conclusão dos cursos formativos, o estagiário foi integrado num projeto que acabara de ser iniciado. Este proporcionou uma excelente oportunidade para aplicar os conhecimentos adquiridos durante as formações, uma vez que o *frontend* da aplicação seria desenvolvido em Angular e o *backend* em .NET.

O facto de ter participado no projeto desde o início, forneceu uma grande oportunidade de aprendizagem, oferecendo uma compreensão completa do ciclo de vida de desenvolvimento de uma aplicação. Participar desde a fase inicial permitiu entender o contexto e os objetivos do projeto, ver como as ideias se transformam em requisitos técnicos e como estes evoluem.

A equipa, composta por sete programadores, incluía dois profissionais seniores e cinco trainees recém-saídos da fase formativa do estágio. Esta combinação de experiência e juventude proporcionou um ambiente de aprendizagem enriquecedor e colaborativo.

Esta secção será organizada de modo a proporcionar uma compreensão clara e detalhada acerca da aplicação e das atividades realizadas ao longo do seu desenvolvimento.

6.2.1 Descrição da Aplicação

A plataforma desenvolvida é uma ferramenta que facilita a gestão e utilização de um sistema de transporte público, tanto para os administradores como para os clientes.

Do lado do cliente, além das funcionalidades inerentes a uma aplicação destas como a capacidade de um utilizador gerir a sua própria conta e informações associadas, também é possível a consulta de horários e rotas disponíveis, identificando os pontos de partida, chegada e paragens intermédias, bem como a compra e gestão de bilhetes.

Por parte dos administradores, é disponibilizada a capacidade de gerir utilizadores, o que inclui a criação, edição e eliminação de contas, assim como a atribuição de privilégios específicos, garantindo um acesso controlado e seguro ao sistema. Adicionalmente, a plataforma possibilita a gestão da frota de veículos, das paragens e das zonas de serviço, permitindo delimitar áreas geográficas específicas.

6.2.2 Estrutura da aplicação:

Backoffice, lado dos administradores:

- **Página Login:** Permite o acesso à conta.
- **Página dos Utilizadores:** Permite gerir utilizadores e permissões.
- **Página dos Veículos:** Permite gerir a frota de veículos.

- **Página dos Motoristas:** Possibilita a gestão de motoristas e associação a veículos da frota.
- **Página das Zonas:** Permite gerir e delimitar geograficamente as zonas de operação.
- **Páginas das Paragens:** Permite gerir as paragens.

Área Cliente:

- **Página de Registo:** Permite a criação de uma nova conta.
- **Página de Recuperação de Password:** Permite recuperar a senha de acesso.
- **Página Login:** Permite o acesso à conta.
- **Página Inicial:** Apresenta informações sobre a plataforma e permite a pesquisa de rotas em datas específicas.
- **Página de Bilhetes:** Permite a compra de bilhetes.
- **Página Histórico de Viagens:** Exibe as viagens passadas do utilizador.
- **Página Configuração de Conta:** Permite ao utilizador editar ou remover a sua conta.
- **Página de Notificações:** Permite ao utilizador receber notificações sobre atualizações de horários, cancelamentos e outras informações importantes.
- **Página dos Termos e Condições:** Apresenta informações sobre os direitos e responsabilidades do utilizador e do proprietário do site.

6.2.3 Recolha de Requisitos

Para garantir o sucesso e a eficácia da aplicação em desenvolvimento, é essencial compreender detalhadamente as necessidades e requisitos tanto dos utilizadores finais quanto dos administradores. Esta secção descreve os principais requisitos identificados para a plataforma.

6.2.3.1 Requisitos Funcionais:

Cientes:

Requisitos	Descrição
RF01	Permitir a criação, edição e eliminação da própria conta.
RF02	Permitir acesso ao histórico de viagens, incluindo detalhes de viagens passadas e reservadas.
RF03	Permitir a visualização de horários e rotas de transporte.
RF04	Permitir a compra e cancelamento de bilhetes.
RF05	Permitir que os clientes recebam notificações.

Table 6.1: Requisitos Funcionais dos Clientes

Administradores:

Requisitos	Descrição
RF01	Permitir a criação, edição e eliminação de contas de utilizador.
RF02	Atribuir privilégios específicos aos utilizadores.
RF03	Permitir a gestão dos veículos disponíveis para os serviços de transporte.
RF04	Permitir a gestão das paragens onde os veículos param.
RF05	Permitir a gestão das zonas de operação e suas delimitações geográficas específicas.
RF06	Permitir a gestão de rotas.

Table 6.2: Requisitos Funcionais dos Administrador

6.2.3.2 Requisitos Não Funcionais

Requisitos	Descrição
RNFO1	Garantir a segurança dos dados dos utilizadores através de medidas de encriptação e proteção contra acessos não autorizados.
RNFO2	Manter uma resposta rápida e eficiente da plataforma, mesmo em períodos de elevado tráfego de utilizadores.
RNFO3	Desenvolver uma interface intuitiva e de fácil utilização para clientes e administradores.
RNFO4	Garantir a compatibilidade da plataforma com uma variedade de dispositivos e navegadores web comuns, incluindo computadores desktop, tablets e smartphones.

Table 6.3: Requisitos Não Funcionais

6.2.4 Arquitetura do sistema

A estrutura do projeto é composta por três componentes principais como está ilustrado na figura 6.1.

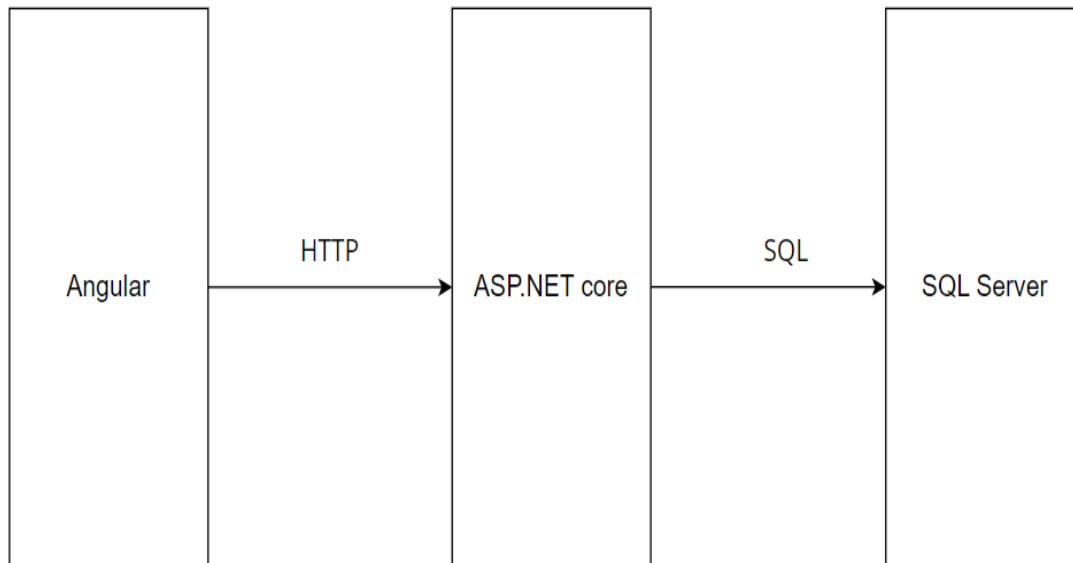


Figure 6.1: Estrutura do Projeto

Neste contexto o Angular é responsável por fornecer uma interface interativa ao utilizador, enviando e recebendo pedidos HTTP para e do *backend*.

O ASP.NET Core atua como o núcleo da aplicação, processando estes pedidos, executando a lógica de negócios necessária e interagindo com o SQL Server para operações de leitura e escrita de dados.

O SQL Server, é onde todos os dados da aplicação são armazenados de forma persistente, garantindo a sua integridade e segurança.

6.2.5 Metodologia de Trabalho

Durante o estágio, a metodologia *Scrum*[9], que é uma das abordagens do *Agile*, um conjunto de práticas para gestão de projetos, foi usada para gerir e realizar o projeto de forma eficiente. O *Scrum* é conhecido pela sua abordagem iterativa e incremental, que permite entregar produtos de alta qualidade de forma rápida e flexível.

Durante a realização deste projeto foram realizadas *sprints*, ciclos de trabalho curtos e definidos, normalmente entre duas a quatro semanas, dependendo da complexidade das tarefas. Os *sprints* começavam com o *Sprint Planning*, uma reunião onde eram delineadas as funcionalidades a serem implementadas, com base nas prioridades do *backlog* do produto. As tarefas eram atribuídas e os objetivos eram estabelecidos.

Durante o *Sprint*, as *Daily Scrums*, reuniões diárias, eram realizadas às 9h30min. Nelas era partilhado o que já tinha sido feito, o que estava por fazer e os desafios que tinham sido enfrentados, além de esclarecer dúvidas.

No final da *sprint*, era realizada uma *Sprint Review*, na qual o trabalho realizado era avaliado e recolhido o *feedback*, permitindo assim ajustes e melhorias contínuas.

6.2.6 Desenvolvimento da Aplicação

Antes de iniciar o desenvolvimento das funcionalidades específicas da aplicação, foi necessário estabelecer uma base sólida tanto para o *frontend* como para o *backend*. No *frontend*, definiu-se a estrutura de pastas e módulos, garantindo uma arquitetura escalável e de fácil manutenção. Esta etapa incluiu a configuração inicial do ambiente, a criação de componentes reutilizáveis e a implementação de serviços para a comunicação com o *backend*.

No *backend*, desenvolvido com ASP.NET Core, o foco inicial foi na definição da estrutura da base de dados. Optou-se por uma abordagem *Code First*, onde todas as entidades (*Entities*) foram definidas através de classes C#. Nesta abordagem, as classes representam as tabelas da base de dados, e os seus atributos representam as colunas dessas tabelas.

Após a definição das entidades, procedeu-se à criação das migrações necessárias para refletir estas estruturas na base de dados. As migrações são um mecanismo que permite gerir e aplicar alterações ao esquema da base de dados ao longo do tempo. Cada migração captura as alterações feitas às entidades e gera o código necessário para atualizar a base de dados de acordo com essas alterações. Desta forma, é possível evoluir a estrutura de forma controlada e incremental.

A estrutura resultante da base de dados está representada na Figura 6.2.

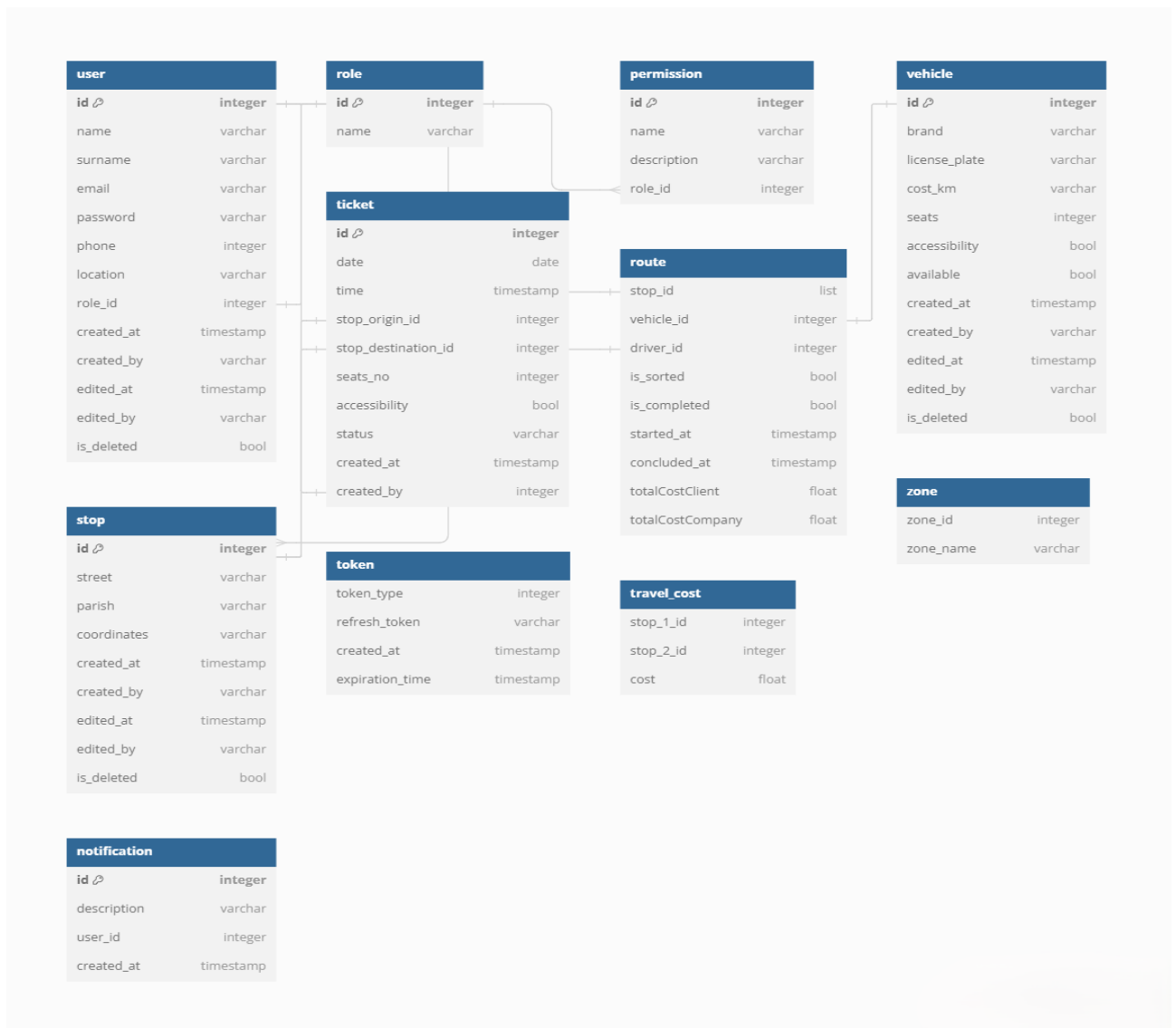


Figure 6.2: Estrutura da Base de Dados

A próxima etapa no desenvolvimento da aplicação foi construir a lógica necessária para suportar utilizadores. De modo a simplificar o processo, ASP.NET Identity[10] foi utilizado para ajudar a criar, gerir e armazenar contas, *roles*, permissões e *tokens*. Além disso, esta ferramenta abrange tanto a autenticação quanto a autorização. Autenticação refere-se ao processo de verificar a identidade do utilizador, garantindo que ele é quem diz ser, enquanto a autorização determina quais ações e que recursos ele tem permissão para aceder. Com isso, o ASP.NET Identity assegura uma gestão eficaz e segura de acessos na aplicação.

Após a conclusão da base da aplicação, procedeu-se a uma divisão mais clara das funcionalidades, onde cada membro da equipa ficou encarregado de desenvolver o *backend* e o *frontend* associado a uma página do *Backoffice*.

O estagiário ficou responsável pela **Página dos Motoristas**, como mostrado na Figura 6.3, onde era possível a criação, edição e remoção de motoristas, bem como a associação a veícu-

los da frota e o armazenamento de documentos importantes.

Nome	Morada	Email	Contacto	Carta	Estado
Pessoa6	Rua cidade da covilhã	peessoa6@latitudde.com	912006135	A1,B,D	Ativo
Pessoa7	Rua cidade da covilhã	peessoa7@latitudde.com	912006135	A,A2,D	Ativo
Pessoa5	Rua cidade da covilhã	peessoa5@latitudde.com	912006135	A,D,D1	Ativo
Pessoa3	Rua cidade da covilhã	peessoa3@latitudde.com	912006135	A,A1,C1,D,D1	Ativo
Pessoa2	Rua cidade da covilhã	peessoa2@latitudde.com	912006135	A1,B,C1,D1	Ativo
Pessoa10	Rua cidade da covilhã	peessoa10@latitudde.com	912006135	A2,D1	Ativo
Pessoa1	Rua cidade da covilhã	peessoa1@latitudde.com	912006135	A,A2,B	Ativo
Pessoa4	Rua cidade da covilhã	peessoa4@latitudde.com	912006135	A,A2	Ativo
Pessoa8	Rua cidade da covilhã	peessoa8@latitudde.com	912006135	A,A2	Ativo
Pessoa9	Rua cidade da covilhã	peessoa9@latitudde.com	912006135	C,D	Ativo

Figure 6.3: Página dos Motoristas

Durante o desenvolvimento do *frontend* desta e das outras páginas houve vários tipos de cuidados a ter, como a integração do módulo de internacionalização do Angular, **i18n**[11], para dar capacidade à aplicação de suportar vários idiomas, nomeadamente o Inglês do Reino Unido e o Português. Além disso, foi utilizada a biblioteca **PrimeNG**[12], que disponibiliza componentes pré-construídos para facilitar a construção das páginas.

Depois da finalização da Página dos Motoristas, a próxima tarefa consistiu em uniformizar o CSS do *Backoffice*. O facto de cada membro da equipa estar a trabalhar em páginas diferentes e os *designs* serem semelhantes entre elas, como podemos observar nas figuras 6.4 e 6.5, levou a que numa fase inicial houvesse muitas inconsistências.

Utilizadores

Q Procurar + Novo Utilizador

Filtro Estado Função

Nome	Morada	Email	Contacto	Função
Pessoa6	Rua cidade da covilhã	peessoa6@latitudde.com	912006135	Motorista
Pessoa7	Rua cidade da covilhã	peessoa7@latitudde.com	912006135	Admin , Operador , Motorista
Pessoa5	Rua cidade da covilhã	peessoa5@latitudde.com	912006135	Motorista
Jose	Rua cidade da covilhã	zeeeeeee@latitudde.com	912006135	Cliente
Pessoa3	Rua cidade da covilhã	peessoa3@latitudde.com	912006135	Motorista
Pessoa2	Rua cidade da covilhã	peessoa2@latitudde.com	912006135	Motorista
Pessoa10	Rua cidade da covilhã	peessoa10@latitudde.com	912006135	Motorista
asdasd	asdasd	miguelserranomatos@gmail.com	912006177	Cliente
Miguel Matoss	Rua cidade da covilhã	pt-update@latitudde.com	912006135	Admin
Miguel Matos	Rua cidade da covilhã	miguel.matos@latitudde.com	912006131	Admin , Cliente

<< < 1 2 > >> 10 v

Figure 6.4: Página dos Utilizadores

Veículos

Q Procurar + Novo Veiculo

Filtro Estado Acessibilidade

Matrícula	Motorista	Modelo	Ano	Registo	Capacidade	Estado
IU-99-UB	Miguel Matos	opel	2000		1	Ativo
IU-99-UT	Miguel Matos	opel	2000		1	Ativo
WW-99-WP	Pessoa7	Mercedes	2000	aaaaaaaaaaaaaaaa	2	Ativo

<< < 1 > >> 10 v

Figure 6.5: Página dos Veículos

No final desta tarefa, o estagiário ficou responsável por testar o *Backoffice* a fundo e resolver qualquer problema que surgisse, enquanto que os restantes membros da equipa avançavam para a construção da área do cliente.

Por final ficou-lhe atribuída a página da compra dos bilhetes na área do cliente, como visto na figura 6.6.

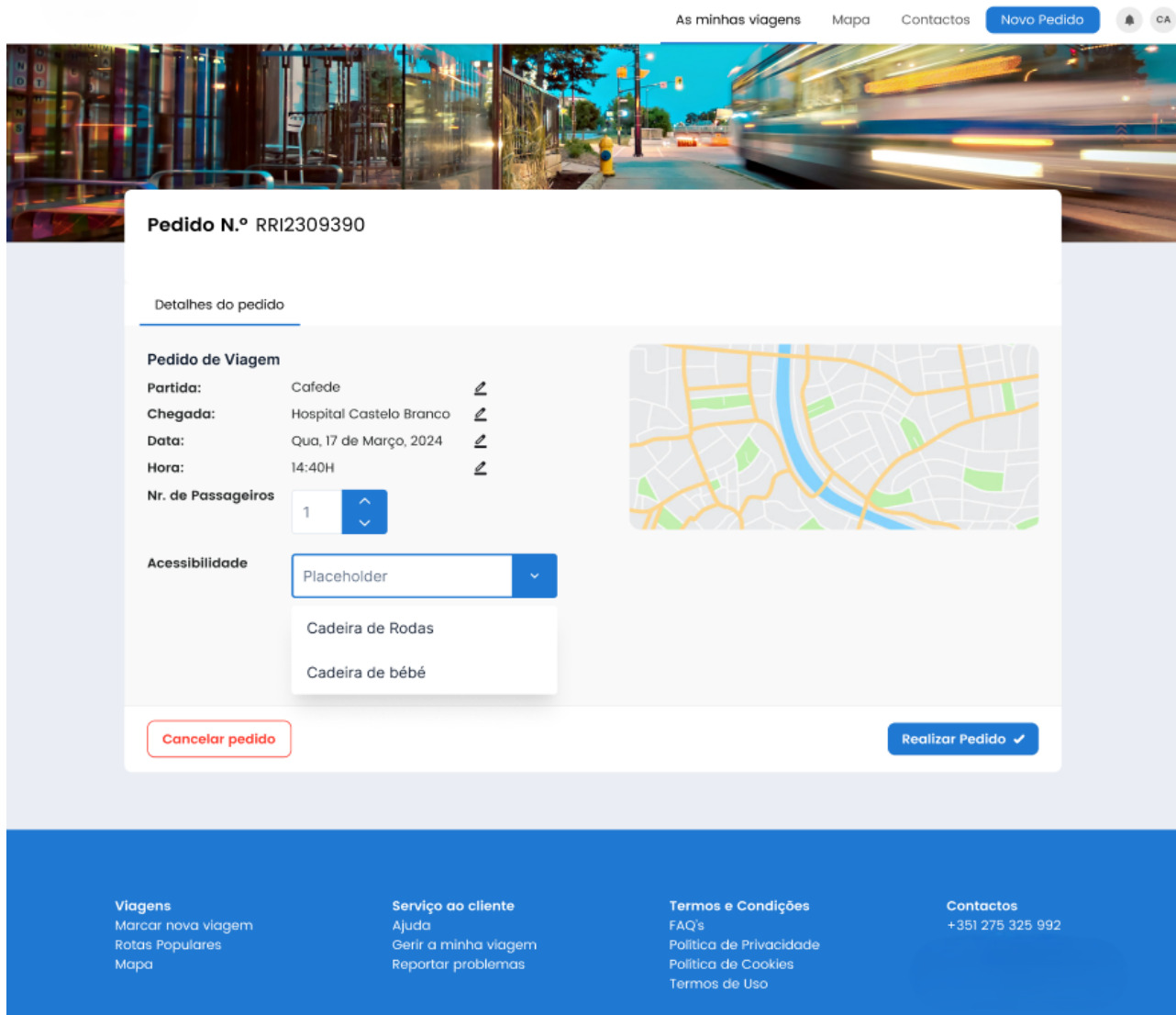


Figure 6.6: Página dos Bilhetes

6.2.7 Desafios enfrentados

É comum que surjam vários tipos de problemas quando uma equipa inexperiente realiza um projeto, especialmente quando se trata do seu primeiro projeto em contexto profissional e nunca tiveram contacto prévio com tecnologias como Angular e ASP.NET Core.

Alguns exemplos desses problemas incluem:

- Falta de comunicação.
- Código de baixa qualidade, resultando em mais tempo gasto em refatoração.
- Trabalho perdido devido à desorganização e à pouca experiência com ferramentas como Git.
- Problemas em cumprir os prazos propostos.

Ao longo do projeto, a equipa enfrentou vários desafios devido à sua inexperiência, mas teve capacidade de aprender com os erros e de os resolver.

6.2.8 Trabalho Futuro

Embora o projeto tenha enfrentado diversos desafios e a equipa tenha aprendido muito ao longo do processo, ainda há muito a ser feito para refinar a aplicação. Desde de implementar métodos de pagamento para os bilhetes até desenvolver um algoritmo de otimização de rotas, de forma a maximizar a cobertura da rede de paragens disponíveis para os utilizadores finais. Existe um longo caminho pela frente até a aplicação estar pronta para os clientes finais.

Chapter 7

Conclusão

O estágio realizado na Latitudde - Digital Enablers foi uma experiência enriquecedora que proporcionou um ótimo ambiente para o desenvolvimento profissional e pessoal. Ao longo do estágio, foi possível obter uma integração eficaz no mercado de trabalho, aplicando e melhorando conhecimentos técnicos.

A experiência adquirida durante este estágio não só consolidou o meu interesse pela área de tecnologia, como também aumentou as minhas perspectivas profissionais. Através dos desafios enfrentados, sinto-me preparado para continuar a construir uma carreira promissora como programador *full stack*, contribuindo para projetos inovadores e impactantes.

Em suma, este estágio foi uma etapa fundamental no início da minha trajetória profissional, oferecendo uma base sólida para futuros desafios e oportunidades no setor tecnológico.

Bibliography

- [1] Udemy. [Online]. Available: <https://www.udemy.com> 1
- [2] Latitudde - digital enablers, lda. [Online]. Available: <https://latitudde.com> 5
- [3] “História da world wide web.” [Online]. Available: https://pt.wikipedia.org/wiki/História_da_World_Wide_Web 9
- [4] GeeksforGeeks, “Front-end x back-end,” 2024. [Online]. Available: <https://www.geeksforgeeks.org/frontend-vs-backend/> 9
- [5] J. Comeau, “The end of front-end development,” 2024. [Online]. Available: <https://www.joshwcomeau.com/blog/the-end-of-frontend-development/> 11
- [6] Gitlab. [Online]. Available: <https://about.gitlab.com> 15
- [7] Angular. [Online]. Available: <https://angular.dev> 15
- [8] Asp.net core. [Online]. Available: <https://dotnet.microsoft.com/en-us/apps/aspnet> 15
- [9] Atlassian, “What is scrum and how to get started.” [Online]. Available: <https://www.atlassian.com/agile/scrum> 21
- [10] Asp.net identity. [Online]. Available: <https://learn.microsoft.com/en-us/aspnet/identity/overview/getting-started/introduction-to-aspnet-identity> 23
- [11] Angular i18n internationalization. [Online]. Available: <https://lokalise.com/blog/angular-i18n/> 24
- [12] Primeng. [Online]. Available: <https://primeng.org> 24