



Departamento de Engenharia Informática e de Sistemas
Instituto Superior de Engenharia de Coimbra
Instituto Politécnico de Coimbra

Licenciatura em Engenharia Informática

Curso LEI

Unidade Curricular de Programação Web

Ano Lectivo de 2020/2021

Tema A

eRenting



Cláudio Filipe Barbosa Figueiredo
Nº- 2016014904

Filipe Alexandre Nunes Silva
Nº- 2016020567

Coimbra, Janeiro de 2021

Índice

- RESUMO ii
- 1. REGRAS DE NEGÓCIO 1
- 2. DESCRIÇÃO DO TEMA ABORDADO NO TRABALHO PRÁTICO 3
 - 2.1. Designação do Tema 3
 - 2.2. Pesquisa e Linha de Pensamento..... 3
- 3. DESCRIÇÃO DO TRABALHO PRÁTICO E METODOLOGIAS UTILIZADAS 5
 - 3.1. Tecnologias e Ferramentas Utilizadas..... 5
 - 3.2. Contextos Utilizados 6
 - 3.3. Pacotes NUGET Utilizados..... 6
- 4. Implementação do Website do Trabalho 7
 - 4.1. Linha de Pensamento..... 7
 - 4.2. Diagrama da Base de Dados..... 7
 - 4.3. Controladores 8
 - 4.4 Vistas Implementadas..... 9
 - 4.5 Templates..... 9
 - 4.6 Outras Funcionalidades..... 9
 - 4.7 Contas Utilizadas..... 10
 - 4.8 Utilização da Aplicação 10
- 5. CONSIDERAÇÕES FINAIS 13

RESUMO

O presente relatório pretende sintetizar o trabalho prático relativamente à disciplina de Programação Web, incluída no 1º semestre do 3º ano da Licenciatura de Engenharia Informática no Instituto Superior de Engenharia de Coimbra.

O objetivo principal deste trabalho prático é disponibilizar uma plataforma que permita, de uma forma simples, empresas de Renting alugarem as suas frotas de veículos, assim como também a gestão dessas mesmas frotas.

Esta aplicação web foi desenvolvida em C# ASP.NET MVC5, onde foram utilizadas tecnologias como SQLServer (LocalDb) e Entity Framework Code First.

(Palavras-Chave: ASP.NET MVC5, CodeFirst, LocalDb, frota, gestão).

1. REGRAS DE NEGÓCIO

O negócio de aluguer de carros já não é novidade no mercado. Alugar um carro é uma ótima opção numa grande diversidade de possibilidades. Um exemplo insere-se nas viagens internacionais ou até mesmo intercidades onde à chegada a uma nova localização, caso o cliente tenha feito a viagem por comboio, avião..., existe a solução do mesmo ter a facilidade de deslocamento através de um veículo alugado. Desta forma, não há a necessidade da utilização de transportes públicos ou esperar que um táxi ou Uber chegue.

As reservas costumam ser feitas numa loja de cada empresa Rent-a-car. Assim, caso o cliente não tenha a certeza de qual o carro pretende terá que ir a várias lojas das diversas empresas para então tomar uma decisão.

Com a situação bem conhecida da pandemia Covid-19 essa possibilidade deixou de ser segura, tanto para o cliente como para os funcionários das diversas empresas que ficam em contacto direto com um número bastante alto de clientes todos os dias.

Deste modo, a aplicação desenvolvida visa solucionar este problema, assim como abre a possibilidade de que as empresas consigam fazer uma gestão simples e eficaz das suas frotas de veículos. Os clientes também ficam a ganhar uma vez que lhes é possível ver uma enorme quantidade de opções, comparar preços e assim tomar a decisão de qual o carro a alugar com apenas alguns cliques e sem sair do conforto da sua casa.

2. DESCRIÇÃO DO TEMA ABORDADO NO TRABALHO PRÁTICO

2.1. Designação do Tema

O trabalho prático foi focado em 3 aspetos principais:

- ➔ Deve ser “user-friendly” para o cliente, uma vez que este aspeto é o que, num cenário real, irá favorecer o projeto desenvolvido em relação à competição existente;
- ➔ Deve ser de simples manutenção;
- ➔ Deve ser coerente ao longo de todo o processo criativo. Ao desenvolver uma aplicação de raiz é bastante comum serem criadas várias incoerências à medida que a aplicação, WebSite..., está a ser criado, tornando-se bastante difícil adicionar novas características à mesma caso haja essa necessidade.

Descreva com clareza, objectividade e detalhes os principais aspectos que foram focados no trabalho prático e o porquê da escolha desses aspectos sobre o tema exposto.

Pesquise e apresente fundamentos teóricos referidos pelos autores neste seu relatório.

Refira os principais linhas de pensamento seguidas no desenvolvimento do trabalho prático!

2.2. Pesquisa e Linha de Pensamento

Foi realizada uma pesquisa antes do início da realização da aplicação onde foi possível retirar algumas ideias de como deveria ser o layout da aplicação e componentes que deveriam estar inseridas na mesma.

Porém, entendeu-se mais tarde, à medida que as várias aulas de Programação Web iam sendo lecionadas que haveria sempre alterações a serem feitas e que haveria a necessidade de fazer alterações desde a base, com a tentativa de tornar a aplicação coesa e sem erros.

Desta forma, foram criados 3 tipos de utilizadores:

- ➔ User – Cliente particular após ser feita a autenticação;
- ➔ Worker – Funcionários de uma dada empresa;
- ➔ Company – Administração da Empresa que possui poder sobre os funcionários da mesma;

Foram colocadas questões como por exemplo: “Será que os funcionários devem também ter acesso a fazer Reservas?”, ou, “Será que os utilizadores devem ter influência nas entregas ou isso é algo que é feito automaticamente?”.

Assim, a Tabela1, representa as várias possibilidades de cada um deste tipo de utilizadores ao utilizar a aplicação desenvolvida:

Tabela 2-1 – Tipos de Utilizadores e Funcionalidades

User	Worker	Company
Ver catálogo de Veículos	Efetuar Reservas de Veículos	Gestão dos Funcionários da Empresa
Efetuar Reservas de Veículos	Gestão das diversas Reservas	Gestão de Veículos da Empresa
Ver Histórico de Reservas	Confirmar Reservas	Gestão de Categorias de Veículos
	Entrega do Veículo ao Cliente	
	Receção do Veículo	
	Verificações após Receção	

Além dos 3 tipos de utilizadores apresentados, também existe o tipo Admin e a possibilidade do utilizador não estar autenticado ao qual foi denominado de “Guest”. O tipo de utilizador Admin destina-se apenas ao administrador da aplicação. Este possui o controlo total da aplicação assim como as suas funcionalidades. Já o “Guest” apenas tem a possibilidade de ver o catálogo de veículos. Deste modo, é necessário o mesmo fazer o registo ou login antes de ter acesso às reservas.

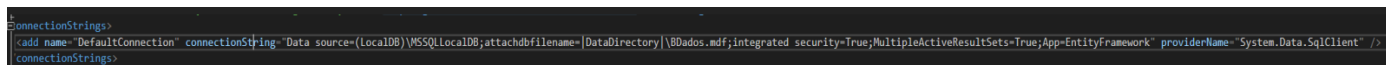
3. DESCRIÇÃO DO TRABALHO PRÁTICO E METODOLOGIAS UTILIZADAS

3.1. Tecnologias e Ferramentas Utilizadas

Para a realização do trabalho prático foram utilizadas diversas tecnologias como por exemplo:

- SQLServer(LocalDb);
- Entity Framework CodeFirst;
- Bootstrap4;
- Template base de Bootstrap;
- IdentityUserRoleManagement;
- ReSharper;

Enquanto tecnologias como o Bootstrap foram utilizadas para que existisse uma melhor experiência do utilizador ao aceder o Website desenvolvido, tecnologias como o SQLServer (LocalDb) permitiram criar uma base de dados de uma forma simples, através de uma conexão denominada por “ConnectionString”. Ao fazer essa conexão, a infraestrutura SQL Server necessária é criada e iniciada, fazendo com que a aplicação consiga usar a base de dados sem serem necessárias configurações complexas. A Figura1 mostra a ConnectionString utilizada no trabalho prático



```
<connectionStrings>  
  <add name="DefaultConnection" connectionString="Data source=(LocalDB)\MSSQLLocalDB;attachdbfilename=DataDirectory\BDados.mdf;integrated security=True;MultipleActiveResultSets=True;App=EntityFramework" providerName="System.Data.SqlClient" />  
</connectionStrings>
```

Figura 1 – ConnectionString utilizada

Durante as aulas de Programação Web foram lecionadas duas formas de utilização do Entity Framework. Uma delas é DatabaseFirst, onde é iniciado o projeto começando pela base de dados. A outra é CodeFirst, onde são criados os modelos inicialmente e a base de dados é criada através dos contextos existentes. Assim, o grupo decidiu ir pela vertente CodeFirst, fazendo desta forma uso das migrações e atualizações da base de dados cada vez que existia alguma alteração nos modelos criados.

Ao longo da realização da aplicação foram visitados diversos Websites, slides das aulas de Programação Web como também exercícios das aulas práticas, ou até mesmo vídeos na plataforma Youtube.

Entre uma grande variedade de Websites, é necessário referir obrigatoriamente o site:

- www.stackoverflow.com
- www.reddit.com/r/csharp
- <https://docs.microsoft.com>

Geralmente, foram esses os sites visitados quando existia alguma dúvida, ou quando a aplicação estava a mostrar erros desconhecidos a ambos os autores do trabalho prático.

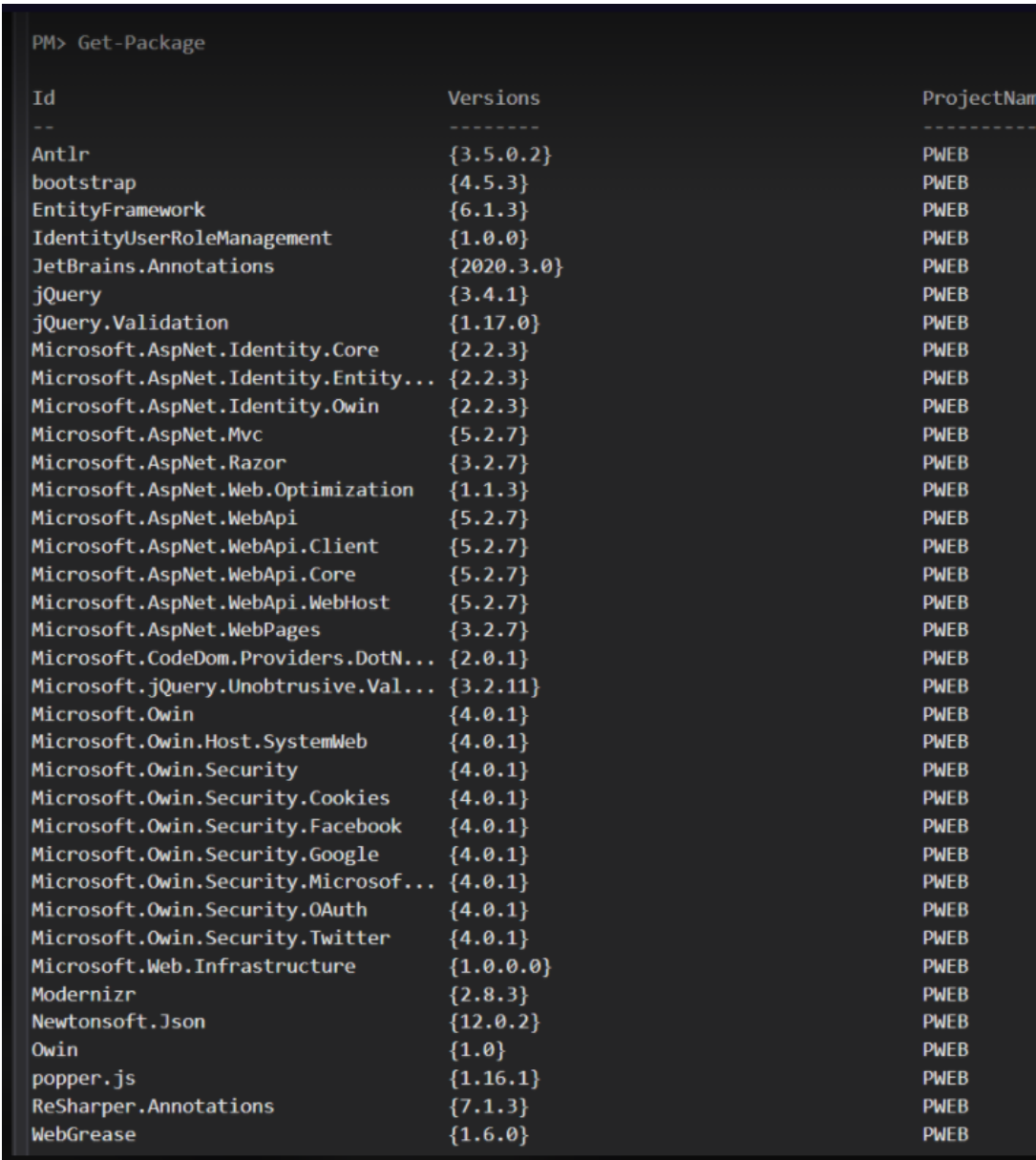
3.2. Contextos Utilizados

Encontram-se na classe IdentityModel.cs os seguintes contextos utilizados na aplicação:

- ➔ Veiculos;
- ➔ Reservas;
- ➔ Entregas;
- ➔ Categorias;
- ➔ Empresas;
- ➔ Defeitos;

3.3. Pacotes NUGET Utilizados

No decorrer do desenvolvimento da aplicação foram utilizados diversos pacotes NUGET que facilitaram a realização do trabalho. A lista seguinte faz referência a todos os pacotes utilizados:



Id	Versions	ProjectName
Antlr	{3.5.0.2}	PWEB
bootstrap	{4.5.3}	PWEB
EntityFramework	{6.1.3}	PWEB
IdentityUserRoleManagement	{1.0.0}	PWEB
JetBrains.Annotations	{2020.3.0}	PWEB
jQuery	{3.4.1}	PWEB
jQuery.Validation	{1.17.0}	PWEB
Microsoft.AspNet.Identity.Core	{2.2.3}	PWEB
Microsoft.AspNet.Identity.Entity...	{2.2.3}	PWEB
Microsoft.AspNet.Identity.Owin	{2.2.3}	PWEB
Microsoft.AspNet.Mvc	{5.2.7}	PWEB
Microsoft.AspNet.Razor	{3.2.7}	PWEB
Microsoft.AspNet.Web.Optimization	{1.1.3}	PWEB
Microsoft.AspNet.WebApi	{5.2.7}	PWEB
Microsoft.AspNet.WebApi.Client	{5.2.7}	PWEB
Microsoft.AspNet.WebApi.Core	{5.2.7}	PWEB
Microsoft.AspNet.WebApi.WebHost	{5.2.7}	PWEB
Microsoft.AspNet.WebPages	{3.2.7}	PWEB
Microsoft.CodeDom.Providers.DotN...	{2.0.1}	PWEB
Microsoft.jQuery.Unobtrusive.Val...	{3.2.11}	PWEB
Microsoft.Owin	{4.0.1}	PWEB
Microsoft.Owin.Host.SystemWeb	{4.0.1}	PWEB
Microsoft.Owin.Security	{4.0.1}	PWEB
Microsoft.Owin.Security.Cookies	{4.0.1}	PWEB
Microsoft.Owin.Security.Facebook	{4.0.1}	PWEB
Microsoft.Owin.Security.Google	{4.0.1}	PWEB
Microsoft.Owin.Security.Microsof...	{4.0.1}	PWEB
Microsoft.Owin.Security.OAuth	{4.0.1}	PWEB
Microsoft.Owin.Security.Twitter	{4.0.1}	PWEB
Microsoft.Web.Infrastructure	{1.0.0.0}	PWEB
Modernizr	{2.8.3}	PWEB
Newtonsoft.Json	{12.0.2}	PWEB
Owin	{1.0}	PWEB
popper.js	{1.16.1}	PWEB
ReSharper.Annotations	{7.1.3}	PWEB
WebGrease	{1.6.0}	PWEB

Figura 2 – Pacotes NUGET utilizados

4. Implementação do Website do Trabalho

4.1. Linha de Pensamento

Na realização do projeto houve uma linha de pensamento definida em alguns aspetos em que os elementos do grupo acreditaram ser necessários para o bom funcionamento da aplicação.

Assim, todas as tabelas existentes possuem um identificador único que permite a manipulação de dados entre controladores. Um exemplo disso é a relação existente entre Veículos, Reservas e Entregas. Uma entrega está sempre associada a uma reserva através dos Identificadores de cada uma das mesmas. Já a tabela de veículos interage com as tabelas Reserva,AspNetUsers e Categorias. Uma reserva não é feita sem existir um veículo, e o mesmo está ligado ao dono da empresa que adicionou esse veículo. Além disso o veículo possui uma categoria e apenas uma, visto que não faz sentido um veículo possuir mais que uma categoria.

4.2. Diagrama da Base de Dados

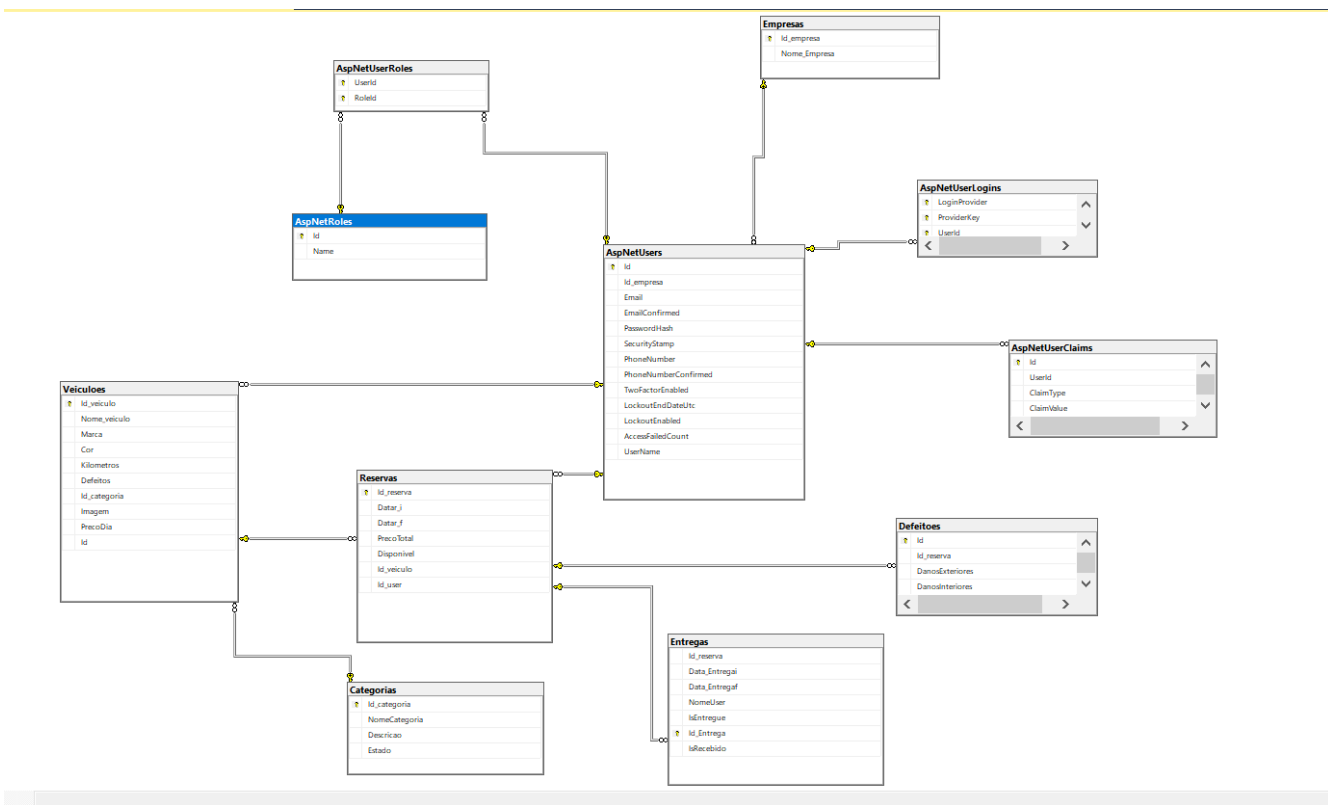


Figura 3 – Diagrama da Base de Dados

No diagrama apresentado em cima é possível ver 11 Tabelas, das quais 5 foram criadas automaticamente com o Entity Framework e com o IdentityUserRolesManagement.

Assim, tem-se:

- ➔ Veiculos – Tabela que guarda informação relativa aos Veículos das várias empresas;
- ➔ Reservas -Tabela responsável por armazenar conteúdo das reservas;
- ➔ Categorias – Tabela referente às várias Categorias que um carro possa ter;
- ➔ Entregas – Esta tabela é responsável por guardar informação relativa às diversas entregas. É dependente da tabela Reservas;
- ➔ Defeitos – Esta tabela tem como intuito guardar informação relativa aos vários defeitos que um veículo possa apresentar depois que o carro seja devolvido à empresa;

Para a realização destas tabelas foi necessário a utilização dos seus respetivos modelos, sendo estes:

- ➔ Veiculo.cs;
- ➔ Empresa.cs;
- ➔ Reserva.cs;
- ➔ Entrega.cs;
- ➔ Defeito.cs;

Além destes, também houve alterações nos modelos pré-definidos, sendo eles:

- ➔ IdentityModel.cs;
- ➔ AccountViewModel.cs;

Foram também implementados outros modelos, que se encontram em Models/Validations, onde estão classes de validação que foram adicionadas como data annotations às respetivas variáveis nos modelos.

Entre elas está a classe StartReservationDataCheck, EndReservationDataCheck e Veiculos Usage.

4.3. Controladores

Para a parte lógica da aplicação foi necessário adicionar diversos controladores. Além dos já pré-definidos no início do projeto, foram implementados:

- ➔ CategoriasController;
- ➔ EmpresasController;
- ➔ EntregasController;
- ➔ DefeitosController;
- ➔ FuncionariosController;
- ➔ VeiculosController;
- ➔ ReservasController;

Como é possível observar, grande parte dos controladores existentes são relativos aos seus respetivos modelos. Assim, funções de cada uma das funcionalidades estarão no seu respetivo controlador. O controlador FuncionariosController é dos únicos que não

possui qualquer modelo associado. Desta forma, o mesmo faz uso do modelo AccountViewModel. Lá estará o RegisterFuncionarioViewModel, que por sua vez irá permitir a criação de funcionários.

Controladores como o AccountController também foram alterados para satisfazer as necessidades da aplicação.

4.4 Vistas Implementadas

Ao longo da aplicação foram implementadas vistas de modo a que o utilizador tenha uma experiência favorável e que o WebSite funcione como desejado.

Deste modo, foram implementadas vistas para os controladores existentes para que fosse possível satisfazer as funcionalidades pretendidas. Destas vistas, a Create serve para criar novos veículos, reservas, ou o que representar a respetiva vista (Create na vista Veiculos criará novos veículos, enquanto que Create na vista Reservas criará novas reservas). O mesmo acontece para as vistas Index, Edit e Details.

Estas vistas mostram apenas conteúdo aos tipos de utilizador desejados. É feito uso de Authorization, presente nos controladores assim como a condição: `if(User.IsInRole())`, presente nas vistas, com o intuito da aplicação mostrar apenas o que deve para cada tipo de utilizador e nada mais (Não faz qualquer sentido um utilizador do tipo User/particular ter a possibilidade de adicionar um veículo).

Vistas como Account/Register, Account/Login e Shared/Layout também foram alteradas de modo a satisfazer as necessidades da aplicação.

4.5 Templates

Para a realização da aplicação foi utilizado o template SB Admin2. Foi implementado na pasta Content.

Foram adicionadas 3 pastas: css, vendor e js.

O link para aceder ao template original é o seguinte:

Link: <https://startbootstrap.com/theme/sb-admin-2>

4.6 Outras Funcionalidades

Foram implementadas na aplicação funcionalidades como a de recuperação de palavra-pass no login incluindo também a funcionalidade de confirmação de e-mail. Porém, como grande parte dos e-mails das contas utilizadas são fictícios, optou-se por permitir que o utilizador entre mesmo sem confirmar o e-mail, tornando apenas impossível a recuperação da palavra-pass. Para esta funcionalidade foi criado um serviço SMTP com um e-mail associado à conta Google que envia mensagens de confirmação de e-mail, assim como de

recuperação. Este e-mail encontra-se definido no ficheiro Web_Config e fez-se uso de excertos de código já construídos (ForgotPassword, ConfirmEmail).

Foi implementada a funcionalidade de enviar fotografias no ato de entrega nas verificações assim como a possibilidade de adicionar uma fotografia ao veículo(Thumbnail) de forma a que o utilizador tenha a possibilidade de ver o carro.

Foi considerada a possibilidade de pagamento no ato de entrega do veículo.

Foi também criada uma vista denominada por “not_found”. Esta vista foi criada de modo a mitigar o crash da aplicação. Em ambiente real, caso o utilizador se deparasse com um erro 404 – NotFound, o resultado iria crashar o site sendo necessário reativá-lo novamente. Assim, e de forma a prevenir ataques de SQL Injection foi implementada esta vista que possibilita o utilizador voltar atrás e mitigando assim a quebra da aplicação.

4.7 Contas Utilizadas

Foram utilizadas as seguintes contas:

- ➔ Admin: Admin@isec.pt;
- ➔ Company/Empresa: Company@isec.pt;
cfbuffer@gmail.com;
- ➔ User/Cliente: particular@isec.pt;
- ➔ Funcionário/Worker da empresa CFL: funcionariocfl@gmail.com;
- ➔ Funcionário/Worker da empresa Company: workercompany@isec.pt;

Nota: Todos os e-mails listados acima possuem a palavra-pass: 1qazZAQ!

4.8 Utilização da Aplicação

A utilização da aplicação depende do tipo de utilizador utilizado. Por exemplo, se for um cliente anónimo, é possível observar o catálogo de veículos por parte das várias empresas.

Se o utilizador for um cliente registado os passos passam pelo seguinte:

- ➔ O cliente irá à área de reservas onde poderá escolher a data de reserva pretendida assim como o carro entre os disponíveis.
- ➔ Após a criação da reserva ter sido bem-sucedida, o cliente terá de esperar pela aprovação do funcionário da empresa ao qual o veículo pertence.
- ➔ A parte da entrega é feita automaticamente como é possível entender mais à frente.

Se o utilizador for um funcionário este terá de confirmar a entrega, através da funcionalidade de Editar Reserva.

Nesta funcionalidade apenas a possibilidade de ativar a reserva está permitida. Após a aceitação de reserva é criada automaticamente uma entrega associada à reserva em questão. Na altura da entrega do devido veículo por parte do cliente à empresa existe a funcionalidade de verificação de defeitos com recurso a imagens.

Já se o utilizador for uma empresa/ administrador da empresa, poderá adicionar viaturas novas, editá-las ou apagá-las. Pode gerir os funcionários existentes na empresa podendo adicionar um novo através da funcionalidade de Registo.

Entretanto o Admin, ou seja, o Administrador da aplicação tem controle de todo o site, podendo gerir as Categorias dos diversos veículos ou as contas existentes na aplicação.

Esta escolha foi feita por ambos os integrantes do grupo uma vez que não faz qualquer sentido o Administrador não possuir controle total. Por exemplo, poderia haver um utilizador que se registasse como empresa e colocasse veículos falsos à venda. Nesta situação é imperativo que o Administrador consiga apagar a contra em questão o mais brevemente possível de modo a diminuir os danos.

5. CONSIDERAÇÕES FINAIS

Considerando toda a aplicação desenvolvida os elementos do grupo podem afirmar que grande parte das funcionalidades do trabalho prático foram realizadas com sucesso.

Uma das únicas funcionalidades que não foi realizadas foi a parte de a empresa não conseguir definir as diversas verificações no ato de entrega.

Outro problema que acontece é o facto que, ao registar um utilizador, o mesmo vai entrar na conta. Porém o menu de opções aparece como se o utilizador fosse anónimo. Este problema é resolvido fazendo Logout e Login novamente.

Para o trabalho realizado o grupo considerou que o pagamento poderia ser feito pessoalmente, uma vez que o utilizador terá de ir buscar o carro. Uma das possibilidades de trabalho futuro será criar uma forma personalizada de pagamento online onde o utilizador poderá escolher o tipo de pagamento.