|  |  |  |  |
| --- | --- | --- | --- |
| A picture containing icon  Description automatically generated | | Curso de TeSP em  Programação de Sistemas de Informação  PlatSI | |
|  |  | |  |
| **Ano Letivo 2021/2022** | **Av. Periódica** | | **2º Ano, 1º Semestre** |

|  |
| --- |
| **PROJETO DE PlatSI** |

**Relatório de acompanhamento do**

**Ciclo de Desenvolvimento de Software do Projeto em PlatSI**



|  |  |  |
| --- | --- | --- |
| **Turno**: PL1 | **Grupo**: 1 | **Docente**: Mário Fernandes |
| **Nº 2201126** | Diogo Pereira | |
| **Nº 2202415** | Andreia Agostinho | |
| **Nº 2201131** | Gonçalo Ferreira | |

Instituto Politécnico de Leiria

Torres Vedras

Tesp Programação de Sistemas de Informação

2021-2022

Diogo Pereira

Andreia Agostinho

Gonçalo Ferreira

**Índice**

[1 Introdução 5](#_Toc90720194)

[1.1 Especificação de Requisitos 6](#_Toc90720195)

[1.1.1 Requisitos Funcionais (RF) 6](#_Toc90720196)

[1.2 Especificação de User Stories 10](#_Toc90720197)

[1.3 RBAC 12](#_Toc90720198)

[2 Conclusão 13](#_Toc90720199)

**Índice de tabelas**

[Tabela 1 - Requisitos Funcionais do Back-office da Plataforma Web 6](#_Toc90720188)

[Tabela 2 - Requisitos Funcionais do Front-office da Plataforma Web 8](#_Toc90720189)

[Tabela 3 - Tabela de User Stories do Back-office da Plataforma Web 10](#_Toc90720190)

[Tabela 4 - Tabela de User Stories do Front-office da Plataforma Web 11](#_Toc90720191)

[Tabela 5 - Tabela de Roles do RBAC 12](#_Toc90720192)

[Tabela 6 - Tabela de Rules do RBAC 12](#_Toc90720193)

# Introdução

No âmbito da unidade curricular de PlatSI do 1º Semestre do 2º ano do Curso TeSP de Programação de Sistemas de Informação do Instituto Politécnico de Leiria, foi criado o projeto “CarBuddy”, que consiste numa plataforma de gestão de todo o tipo de veículos, desde as suas reparações às suas manutenções.

Assim, para a Unidade Curricular de PlatSI, o objetivo do projeto é a criação e desenvolvimento do Website, para a Unidade Curricular de SIS, o objetivo é o desenvolvimento de uma API e para a Unidade Curricular de AMSI, o objetivo é a criação de uma aplicação móvel.

O nosso projeto de gestão de reparação de veículos, consiste numa combinação de três componentes, sendo eles:

Um website, onde se fazem gestão de contas de utilizador, onde vão ser vistos/registados os veículos, o seu estado e as suas reparações.

Uma aplicação, onde cada cliente tem listadas todas as reparações e os respetivos veículos e o colaborador/mecânico, inserindo o VIN do carro acede ao histórico de reparações.

Uma API, para ser possível fazer uma interligação de dados entre ambos os componentes referidos anteriormente.

O objetivo, é que mesmo a nível internacional, se possam registar e verificar o histórico de reparações.

O projeto é comum entre ambas as disciplinas, mas as fases que traçam este projeto são divididas pelas três, conforme a sua pertinência.

## Especificação de Requisitos

### Requisitos Funcionais (RF)

Tabela 1 - Requisitos Funcionais do Back-office da Plataforma Web

|  |  |  |
| --- | --- | --- |
| # ID | Descrição | Prioridade |
| RF-01 | O sistema deverá obter os seus dados pela API desenvolvida em SIS e pela base de dados | **Alta** |
| RF-02 | O sistema deverá estar dividido em back e front office | **Alta** |
| RF-03 | O sistema deverá ter Login para todos os utilizadores | **Alta** |
| RF-04 | O sistema deverá utilizar a tecnologia RBAC para a autenticação | **Alta** |
| RF-05 | O sistema deverá ser desenvolvido utilizando a Framework Yii e a linguagem de programação PHP | **Alta** |
| RF-06 | O back-office deverá estar protegido contra acessos de utilizadores com acesso exclusivo ao front-office | **Alta** |
| RF-07 | O sistema deverá ter uma ferramenta de gestão de utilizadores para o administrador | **Alta** |
| RF-08 | O sistema deverá ter uma ferramenta de gestão de empresas para o administrador | **Alta** |
| RF-09 | O sistema deverá confirmar todas ações antes de as executar | **Média** |
| RF-10 | O sistema deverá ter uma resposta de no máximo 2 segundos por cada ação | **Alta** |
| RF-11 | O design do sistema deverá ser simples, agradável e intuitivo | **Alta** |
| RF-12 | O design do sistema deverá ser idêntico ao design do sistema desenvolvido para a unidade curricular de AMSI | **Média** |
| RF-13 | O sistema deverá ser alvo de testes de software (unitários, funcionais e de aceitação) | **Média** |
| RF-14 | O sistema deverá estar disponibilizado e terminado em janeiro | **Alta** |
| RF-15 | O sistema deverá estar publicado num repositório GitHub | **Média** |
| RF-16 | O sistema deverá prevenir hacking SQL Injection | **Baixa** |
| RF-17 | O sistema deverá ser testado utilizando a Framework Codeception | **Média** |
| RF-18 | O sistema deverá encriptar as palavras-passe dos utilizadores | **Alto** |
| RF-19 | O sistema deverá enviar um e-mail ao cliente quando a reparação/manutenção do seu veiculo estiver concluída | **Baixa** |

Tabela 2 - Requisitos Funcionais do Front-office da Plataforma Web

|  |  |  |
| --- | --- | --- |
| # ID | Descrição | Prioridade |
| RF-01 | O sistema deverá obter os seus dados pela API desenvolvida em SIS e pela base de dados | **Alta** |
| RF-02 | O sistema deverá estar dividido em back e front office | **Alta** |
| RF-03 | O sistema deverá ter Login para todos os utilizadores | **Alta** |
| RF-04 | O sistema deverá ter uma ferramenta de gestão de veículos para os colaboradores | **Alta** |
| RF-05 | O sistema deverá permitir ao colaborador criar reparações e manutenções | **Alta** |
| RF-06 | O sistema deverá ter uma página onde conste os detalhes de um veiculo selecionado | **Alta** |
| RF-07 | O sistema deverá permitir ao colaborador concluir reparações e manutenções, indicando a descrição das mesmas | **Alta** |
| RF-08 | O sistema deverá utilizar a tecnologia RBAC para a autenticação | **Alta** |
| RF-09 | O sistema deverá ser desenvolvido utilizando a Framework Yii e a linguagem de programação PHP | **Alta** |
| RF-10 | O cliente deverá ter acesso a uma página onde conste os detalhes de um veiculo selecionado e as respetivas manutenções e revisões | **Alta** |
| RF-11 | O sistema deverá confirmar todas ações antes de as executar | **Média** |
| RF-12 | O sistema deverá ter uma resposta de no máximo 2 segundos por cada ação | **Alta** |
| RF-13 | O design do sistema deverá ser simples, agradável e intuitivo | **Alta** |
| RF-14 | O design do sistema deverá ser idêntico ao design do sistema desenvolvido para a unidade curricular de AMSI | **Média** |
| RF-15 | O sistema deverá ser alvo de testes de software (unitários, funcionais e de aceitação) | **Média** |
| RF-16 | O sistema deverá estar disponibilizado e terminado em janeiro | **Alta** |
| RF-17 | O sistema deverá estar publicado num repositório GitHub | **Média** |
| RF-18 | O sistema deverá prevenir hacking SQL Injection | **Baixa** |
| RF-19 | O sistema deverá ser testado utilizando a Framework Codeception | **Média** |
| RF-20 | O sistema deverá encriptar as palavras-passe dos utilizadores | **Alto** |

## Especificação de User Stories

Um user story é uma explicação informal e geral de um recurso de software escrito a partir da perspetiva do utilizador final. O objetivo é identificar todas as funcionalidades necessárias e envolvidas no projeto e explicitar o seu modo de funcionamento. Assim sendo, realizamos duas tabelas, uma para o back-office e outra para o front-office.

Tabela 3 - Tabela de User Stories do Back-office da Plataforma Web

|  |  |  |
| --- | --- | --- |
| # ID | Descrição | Prioridade |
| US-01 | Como Administrador da plataforma devo conseguir iniciar sessão | **Alta** |
| US-02 | Como Administrador da plataforma devo conseguir ver e gerir todos os utilizadores registados na plataforma | **Alta** |
| US-03 | Como Administrador da plataforma devo conseguir adicionar, ver e gerir todas as empresas associadas | **Alta** |
| US-04 | Como Administrador da plataforma devo conseguir adicionar, ver e gerir todos os colaboradores associados | **Alta** |
| US-05 | Como Administrador da plataforma devo conseguir ver gráficos, onde constem informações pertinentes sobre a plataforma | **Média** |

Tabela 4 - Tabela de User Stories do Front-office da Plataforma Web

|  |  |  |
| --- | --- | --- |
| # ID | Descrição | Prioridade |
| US-01 | Como Utilizador Guest da plataforma devo conseguir iniciar sessão | **Alta** |
| US-02 | Como Utilizador Guest da plataforma devo conseguir registar-me | **Alta** |
| US-03 | Como Utilizador Guest da plataforma devo conseguir ver a página inicial onde constem informações pertinentes | **Média** |
| US-04 | Como Cliente da plataforma devo conseguir adicionar, ver e gerir todas os meus veículos e ver as reparações dos mesmos | **Alta** |
| US-05 | Como Cliente da plataforma devo conseguir adicionar, ver e gerir todas os meus agendamentos realizados | **Alta** |
| US-06 | Como Cliente da plataforma a minha página inicial deve conter os meus veículos | **Média** |
| US-07 | Como Cliente da plataforma devo conseguir mudar a minha password e gerir a minha conta | **Alta** |
| US-08 | Como Cliente da plataforma devo conseguir ver e utilizar página de contacto | **Média** |
| US-09 | Como Colaborador da plataforma devo conseguir adicionar, ver e gerir todas as reparações realizadas por mim em qualquer veiculo | **Alta** |
| US-10 | Como Colaborador da plataforma devo conseguir ver e gerir todos os agendamentos relacionados com a minha empresa | **Alta** |
| US-11 | Como Colaborador da plataforma devo ter acesso a todas as funcionalidades do Cliente | **Alta** |
| US-12 | Como Colaborador da plataforma a minha página inicial deve conter todas as reparações realizadas por mim | **Média** |
| US-13 | Como Administrador da plataforma devo ter acesso a todas as funcionalidades do Cliente | **Alta** |

## RBAC

O RBAC (Role-based Acess Control) é uma abordagem de segurança para restringir o acesso de utilizadores a uma dada funcionalidade. Assim o RBAC permite ou não a um utilizador aceder a uma dada página da nossa plataforma. Assim sendo, realizamos duas tabelas, uma para as Roles e outra para as Rules da configuração do nosso RBAC. Podemos também afirmar que, os administradores e colaboradores têm acesso a funcionalidades exclusivas, no entanto tantos os administradores, os colaboradores e os clientes têm acesso às funcionalidades do cliente.

Tabela 5 - Tabela de Roles do RBAC

|  |  |  |
| --- | --- | --- |
| Nome | Descrição | ID |
| admin | Tipo de utilizador utilizado para o Administrador | **1** |
| collaborator | Tipo de utilizador utilizado para o Colaborador | **2** |
| client | Tipo de utilizador utilizado para o Cliente | **3** |

Tabela 6 - Tabela de Rules do RBAC

|  |  |
| --- | --- |
| Nome | Descrição |
| backendCrudCompany | Rule associada ao Administrador, que permite aceder a todo o CRUD de empresas do backend |
| backendCrudContributor | Rule associada ao Administrador, que permite aceder a todo o CRUD de contribuidores do backend |
| backendCrudUser | Rule associada ao Administrador, que permite aceder a todo o CRUD de utilizadores do backend |
| frontendCrudVehicle | Rule associada ao Cliente, que permite aceder a todo o CRUD de veículos do frontend |
| frontendCrudRepair | Rule associada ao Colaborador, que permite aceder a todo o CRUD de reparações do frontend |
| frontendCurdScheculesCollaborator | Rule associada ao Colaborador, que permite aceder a todo o CRUD de gestão de agendamentos do frontend |
| frontendCurdScheculesClient | Rule associada ao Cliente, que permite aceder a todo o CRUD de agendamentos do frontend |
| frontendReadRepair | Rule associada ao Cliente, que permite apenas ler as reparações no frontend |

# Conclusão

Conclusão da Fase 1

Com a elaboração deste projeto, cujo tema seria a criação de uma aplicação web onde cada cliente tem listadas todas as reparações e os respetivos veículos, o administrador gere todo o tipo de empresas e colaboradores e o colaborador gere todo o tipo de reparações, conseguimos desenvolver a prática da programação web.

Até ao momento, ainda não tivemos quaisquer dificuldades na elaboração deste projeto, uma vez que se tratou apenas da análise e criação da sua interface, mas prevemos grandes desafios na segunda parte do projeto o que esperamos que nos faça desenvolver bastante as nossas capacidades na área da programação computacional e na segurança computacional.

Enquanto grupo estamos a adorar realizar este projeto e estamos bastante empolgados para concluirmos este trabalho e sentirmo-nos concretizados por mais uma etapa ultrapassada.

Conclusão da Fase 2

Enquanto grupo, o trabalho foi sempre divido e muito bem gerido entre nós, podendo assim, dizer que ambos os três elementos do grupo trabalharam de igual modo. O grupo fica contente por conseguir concluir assim um projeto com esta grandeza, pois sentimos evolução pessoal e profissional ao realizar este projeto.