

## **Licenciatura em Segurança Informática em Redes de Computadores**

### **Programação em Ambiente Web**

Trabalho Prático

Grupo: de\_pressão2

**Pedro Costa, César Meireles, Marcos Oliveira**

8150245@estg.ipp.pt, 8150399@estg.ipp.pt, 8150173@estg.ipp.pt

## Índice

Introdução.....	4
Objetivos.....	4
Ferramentas utilizadas.....	5
Requisitos de negócio.....	6
Base de Dados.....	7
Preparação do ambiente de execução da aplicação web.....	8
Aplicação web desenvolvida (Backend).....	10
Padrão MVC.....	10
Models.....	10
Views.....	11
Controllors.....	11
MongoDB – Interface Compass.....	12
Aplicação web desenvolvida (Frontend).....	14
Página de autenticação.....	14
Páginas iniciais.....	15
Conclusão.....	19

## Índice de Figuras

Modelo User.....	10
Modelo Telemóvel.....	10
Controller – Criação dos users.....	11
Controller – Criação dos telemóveis.....	11
Base de dados – Tabela telemóveis.....	12
Base de dados – Tabela users.....	13
Página de Login.....	14
Página de Registo.....	15
Página inicial de Funcionário.....	15
Página de validação de leilões.....	16
Página inicial de Utilizador.....	16
Criação de um novo leilão.....	17
Leilões pendentes.....	17
Todos os leilões do Utilizador.....	18
Criação de licitação.....	18

## Introdução

Este documento tem como objetivo a especificação de conteúdo técnico da plataforma de vendas de telemóveis usados com o nome de RevendeTele, onde vai constar o propósito desta plataforma, o seu domínio, os vários requisitos necessários para o desenvolvimento da mesma, os testes efetuados, etc.

## Objetivos

Neste projeto realizado no âmbito da unidade curricular de Programação em Ambiente Web, temos como objetivo desenvolver:

- Conhecer e aplicar o modelo Cliente-Servidor e as principais tecnologias/protocolos associados para o desenvolvimento de aplicações Web;
- Compreender e implementar o padrão MVC;
- Implementar aplicações Web através da manipulação de linguagens e frameworks para o desenvolvimento das componentes de cliente e servidor em aplicações Web
- Desenvolver uma aplicação Web utilizando as linguagens, bibliotecas e frameworks lecionadas;
- Compreender o desenvolvimento de aplicações como fullstack developer.

## Ferramentas utilizadas

MongoDB – Utilizamos como base de dados, e decidimo-nos por esta pois já tínhamos algum conhecimento da Unidade Curricular de Processamento Estruturado de Informação, e era mais direto na parte de trocas de informação entre o servidor e a base de dados.

ExpressJS – Será onde estará o nosso back-end e usaremos para fazer a api REST pedida.

EJS - Framwork do ExpressJS para front-end.

Node.js – Utilizamos para a criação e programação do servidor, escolhemos esta solução pois era a que nos sentíamos mais confortáveis.

Mongoose – É um package para a gestão de Base de dados MongoDB, este permite a gestão da base de dados de forma mais simples e permite a utilização de algumas funcionalidades de presentes em Base de dados relacionais.

jQuery – É uma biblioteca de Javascript que facilita em muito a manipulação de elementos no lado do cliente.

DataTables – É também uma biblioteca Javascript baseada em jQuery para a criação e manipulação de tabelas.

Bootstrap 4 – É uma biblioteca CSS para tornar a aplicação mais visualmente apelativa com grande simplicidade.

Postman – Testes de Back-End;

GitHub Desktop – Auxílio na utilização da plataforma GitHub;

## Requisitos de negócio

Um utilizador com login pode licitar apenas no caso de o valor ser superior ao valor de licitação.

Caso a data para o fim do leilão já tenha expirado deve aparecer menção do vencedor do leilão e impedir o produto de receber mais licitações.

Sempre que foi submetida uma venda, o funcionário tem de validar se o telemóvel pode ser vendido ou não!

Na página de internet os todos os utilizadores podem ver a lista de telemóveis para venda (todos os leilões), mas podem apenas licitar no artigo com login válido na página.

Um telemóvel pode receber licitações até ao fim do leilão.

Na página de detalhes do artigo deve aparecer uma lista de licitações ordenada pelo valor da licitação.

Caso a data para o fim do leilão tenha já expirado deve aparecer menção do vencedor do leilão e impedir o produto de receber mais licitações.

Os utilizadores com login devem poder submeter artigos para leilão e ver a lista de artigos que já submeteram para leilão.

Um utilizador pode também cancelar um leilão que ainda não tenha terminado.

Para efeitos de negócio considere que os artigos são entregues na instituição antes do final do leilão.

Os trabalhos entregues deverão evitar (se possível) utilizar caminhos absolutos ou endereços específicos, de modo a que possam ser facilmente utilizados em qualquer máquina.

## Base de Dados

Como estamos a utilizar MongoDB a nossa base de dados é orientada por documentos, e esta está dividida em 3 coleções.

Telemóveis: Coleção em que cada documento representa um processo, sendo que as informações do aluno estão também embutidas no mesmo documento, ou seja, o dossier do aluno.

Utilizadores: Coleção que contém todos os utilizadores da Aplicação.

Licitações: Coleção que contém todas as entidades da Aplicação.

## Preparação do ambiente de execução da aplicação web

De início, iremos instalar o npm, nodejs e mongodb:

```
npm install -g  
sudo apt-get install nodejs  
sudo apt install mongodb
```

Criaremos a pasta onde será guardado todos os ficheiros do mongodb.

```
mkdir data/db  
sudo chmod -R 775 data  
sudo apt install mongodb
```

Séa editado agora o ficheiro *mongodb.service*.

```
sudo touch /etc/systemd/system/mongodb.service  
sudo nano /etc/systemd/system/mongodb.service  
"  
[Unit]  
Description=High-performance, schema-free document-oriented database  
After=network.target  
  
[Service]  
User=mongodb  
ExecStart=/usr/bin/mongod --quiet --config /etc/mongod.conf  
  
[Install]  
WantedBy=multi-user.target  
"
```

Iremos dar agora algumas permissões necessárias e será iniciado o programa.

```
sudo chmod -R go+w /data/db  
  
sudo service mongodb start  
sudo service mongodb status  
  
sudo npm i -g nodemon  
  
sudo mongod
```

Num novo terminal, e dentro da pasta do projeto, iremos correr o próximo comando para termos acesso à base de dados.

```
mongoimport --db paw_tp --collection users --file utilizadores.json
```



Para iniciarmos o nosso projeto, basta fazermos o seguinte comando:

```
npm run paw
```

Num browser abrir a URL: <http://localhost:3000/>

Observação: Usamos portas dinâmicas portanto o valor 3000 foi usado apenas na implementação do trabalho.

Iremos criar alguns utilizadores para conseguir demonstrar o correto e completo funcionamento da aplicação.

## Aplicação web desenvolvida (Backend)

Inicialmente, foi discutido qual seria a base de dados e quais as frameworks que iríamos usar o desenvolvimento da nossa plataforma e decidimos, como base de dados, usar MongoDB.

### Padrão MVC

Consequentemente, seguimos o padrão de software MVC e geramos os Models, Views, Controllers e também as Routes.

### Models

#### User

O modelo do User tem a seguinte estrutura:

```
//schema user
const userSchema = new Schema({
  username: {type: String, required:true},
  nome: {type: String, required:true},
  password: {type:String, required:true},
  tipo: {type: String, default:"utilizador"},
  ....
});
```

Modelo User

#### Telemóvel

Por último, o modelo do Telemóvel tem a seguinte estrutura:

```
//schema de um telemovel
let telemoveis = new Schema({
  marca: {type: String, required:true},
  modelo: {type: String, required:true},
  descricao: {type: String, required:true},
  preçoInicial: {type: Number, min: 0, require:true},
  user:{type: String},
  dataFim:{ type: Date, default: (new Date()), required:true},
  imagem: {type: String},
  estado: {type: String, required:true},
  licitacoes:[{
    licitacao:{type: Number, min: 0, require:true},
    user:{type: String}
  }],
  ultimaLct:{type:Number,min:0, require:true}
});
```

Modelo Telemóvel

## Views

Tendo em conta que o desenvolvimento da parte Front-End do projeto foi toda desenvolvida com EJS, não foi necessário a criação de vistas.

## Controllers

Os controllers que temos implementados são:

- usersController
- telemovelController

```
//Criar novo User
UserController.createUser = function (req, res, next) { →
  const {username, nome, password, tipo} = req.body;
  let errors = [];
  →
  //verificar campos obrigatorios
  if(!username || !nome || !password){
    errors.push({msg: 'Preencha todos os campos!'});
  }
  //verificar tamanho da pass
  if(password.length < 6){
    errors.push({msg: 'password tem de conter 6 caracteres!'});
  }
}
```

Controller – Criação dos users

```
//Criar telemovel
telemovelController.registerTelemovel = function (req, res, next) {
  const { marca, modelo, descricao, preçoInicial, dataFim, estado } = req.body;
  var imagem = req.body.imagem;
  let errors = [];

  //get user logged in
  const user = req.user.username;
  //console.log(req.user);

  //instanciar array de licitacoes
  //var licitacoes = {[ ]}

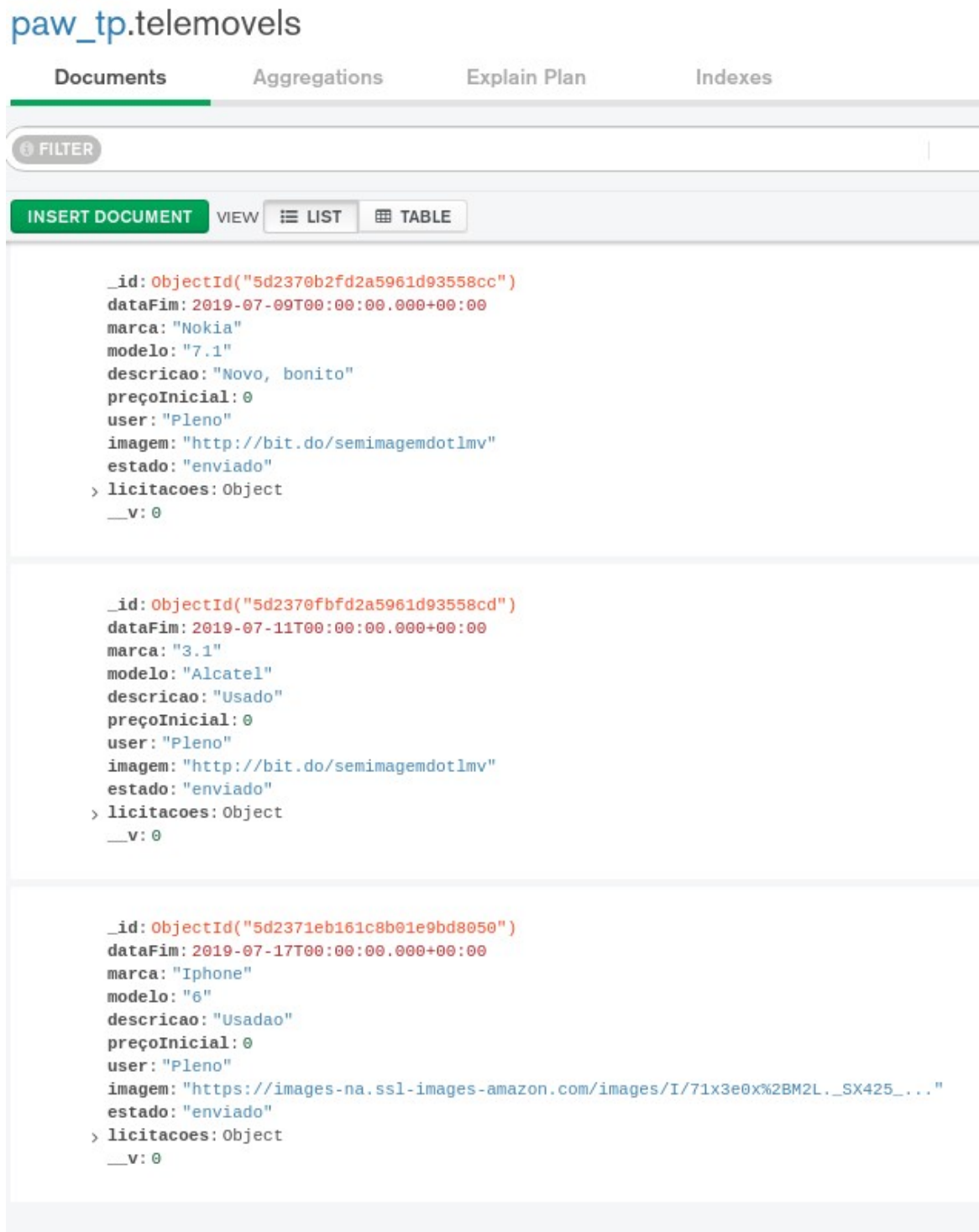
  //verificar se os campos não estão vazios
  if (!marca || !modelo || !descricao || !preçoInicial || !user || !dataFim || !
estado) {
    errors.push({ msg: 'Não está a introduzir dados!' });
    //req.flash('error', 'Não está a introduzir dados!');
  }

  //se nao tiver imagem, guarda uma imagem a dizer "sem imagem"
  if (!imagem) {
    imagem = 'http://bit.do/semimagemdtmlmv';
  }
}
```

Controller – Criação dos telemóveis

## MongoDB – Interface Compass

Enquanto implementávamos a nossa Base de Dados, decidimos usar o Compass do MongoDB para obtermos uma interface gráfica para a visualização dos dados inseridos.



Base de dados – Tabela telemóveis

paw\_tp.users

Documents Aggregations Explain Plan Indexes

FILTER

INSERT DOCUMENT VIEW LIST TABLE

```
{
  "_id": ObjectId("5d237022fd2a5961d93558cb"),
  "tipo": "utilizador",
  "username": "Pleno",
  "nome": "Marcos Oliveira",
  "password": "$2a$12$63Xpf60Zk6tdy9F8ZUNBX.XwZWsk2DQDueYwaccAZjXpRm6xQR../w",
  "__v": 0
}
```

Base de dados – Tabela users

## Aplicação web desenvolvida (Frontend)

Tendo avaliado em grupo qual a framework que usaríamos para o Frontend, chegamos a conclusão que a melhor opção seria suportar o desenvolvimento desta parte da plataforma do projeto em EJS, pois consideramos a ferramenta mais intuitiva e fácil de usar.

EJS é uma framework client-side, ou seja, executa o seu código do lado do cliente. Isto implica que qualquer conteúdo dinâmico necessita de ser pedido a recursos na web, tendo como exemplo o facto de a comunicação com um servidor ou com base de dados ter de ser feita através de pedidos a cada um destes recursos na web. Esta framework funciona por completo com a linguagem JavaScript.

### Página de autenticação

A interface gráfica da aplicação tem por base um conjunto de páginas em formato EJS (que em tempo real é convertido para HTML), sendo que a página de entrada é uma página para a autenticação do utilizador, em que deverá ser fornecido o seu ID e a sua password, sendo que esta deverá ter pelo menos 6 caracteres.

## RevendeTele

Seja bem-vindo

Efetue login ou faça o seu registo!

Registo

Login


Todos os leilões



Iphone 6  
**14€**

Página de Login

Se ainda não tiver criado conta, será apresentado com esta página para o efetuar.

 **Registo**

Preencha todos os campos!

password tem de conter 6 caracteres!

Username

Introduza o seu username

Nome completo

Introduza o seu nome

Password

.....

Tipo:

utilizador

Registrar

Já tem conta? [Login](#)

[Voltar](#)

Página de Registo

## Páginas iniciais

Depois da autenticação surge uma página inicial para cada tipo de utilizador (Administrador e utilizador “normal”), cada página inicial tem um conjunto de divisórias com determinadas funções, cada divisória será explicada a seguir.

## Dashboard - Funcionário

Validar leilões

Terminar leilões

Logout

Leiloes concluidos

Página inicial de Funcionário

Esta página permite não só o acesso as várias informações disponíveis para um Administrador assim como acesso à edição dos produtos que aguardam a validação de se podem ou não ser vendidos.

## Validar leiloes pendentes

	
Nokia 7.1	3.1 Alcatel
0€	0€
Vendedor: Pleno	Vendedor: Pleno
Estado: enviado	Estado: enviado

Página de validação de leilões

## Dashboard - Utilizador

Bem-vindo Pleno

Valor angariado:

Número de lances em leilões:

Número de leilões ganhos:

Criar leilao

Ver leiloes pendentes

Ver os meus leiloes

Licitações

Leiloes que licitei


Logout

Leiloes ativos

Página inicial de Utilizador

Esta página permite não só o acesso as várias informações úteis para um utilizador com conta criada assim como acesso à criação de licitações e de telemóveis para leilão. A tabela que é inicialmente apresentada contém todos os dados atribuídos ao utilizador que fez a autenticação.



 **Leilao novo**

**Marca**

**Modelo**

**Descrição**

**Preço**

**User**

**Data do fim do leilao**

**Imagem do telemóvel**

**Estado:**

[Voltar](#)

Criação de um novo leilão




O utilizador poderá assim, criar um novo leilão, verificar quais os leilões que estão a aguardar a validação por parte de um funcionário, ver o estado dos seus leilões que foram validados e ainda pode licitar em diferentes leilões criados por outros utilizadores.

## Leiloes pendentes

		
Nokia 7.1	3.1 Alcatel	Iphone 6
0€	0€	0€
Vendedor: Pleno	Vendedor: Pleno	Vendedor: Pleno
Estado: enviado	Estado: enviado	Estado: enviado

Leilões pendentes

## Os meus leiloes

 Nokia 7.1 <b>0€</b> Vendedor: Pleno	 3.1 Alcatel <b>0€</b> Vendedor: Pleno	 Iphone 6 <b>0€</b> Vendedor: Pleno Descrição: Usada Termina: Wed Jul 17 2019 01:00:00 GMT+0100 (Western European Summer Time) Estado: enviado <a href="#">Todas as licitações</a>
--	--	--

Todos os leilões do Utilizador

### Nova Licitação

Valor

User

Leilao

[Licitar](#)

[Voltar](#)

Criação de licitação

## Conclusão

Na introdução ao trabalho deste relatório, tínhamos definido, como objetivos principais, conhecer e aplicar de forma concreta tudo o que tínhamos aprendido ao longo do semestre nas aulas desta unidade curricular; conhecer e aplicar métodos que tornaria mais fácil a compreensão do nosso trabalho prático.

Apesar de toda a ajuda que íamos tendo ao longo do tempo, a curva de aprendizagem e a quantidade de matéria lecionada nas aulas estenderam-se ao longo do tempo e associado á dificuldade e extensão do trabalho, foram-nos criados empecilhos que tivemos que aprender a ultrapassar.

Chegando agora á conclusão, e refletindo sobre as dificuldades geradas na criação deste trabalho, não podíamos deixar de destacar a aprendizagem que todo este projeto nos proporcionou, porém, não foi necessário concluir o trabalho com sucesso.