



**IPG** Politécnico  
da Guarda  
Escola Superior  
de Tecnologia e Gestão

# RELATÓRIO DE ESTÁGIO

Curso Técnico Superior Profissional  
em Desenvolvimento de Aplicações Informáticas

Rui Alexandre Gomes Pina

julho | 2017





Escola Superior de Tecnologia e Gestão  
Instituto Politécnico da Guarda

## Relatório de Estágio

**Rui Alexandre Gomes Pina**

Relatório para a obtenção do diploma de Técnico Superior Profissional em  
Desenvolvimento de Aplicações Informáticas

Julho de 2017

## **Identificação dos Intervenientes no Estágio**

### **Estudante**

Nome: Rui Alexandre Gomes Pina;

Nº de aluno: 1012021;

Curso: TeSP Desenvolvimento de Aplicações Informáticas;

Escola: IPG - Escola Superior de Tecnologia e Gestão.

### **Local de estágio**

Empresa: SASTI, Lda.;

Localização: Rua Duque de Bragança, n8 Cave 6300 - 703 Guarda;

Telefone: 271222301;

### **Estágio curricular**

Início: 01 /03/2017;

Fim: 14/07/2017;

Supervisor: Miguel Valério;

Orientador: Luís Figueiredo;

## Plano de Estágio

- Introdução a CMS (*Content Management System*):
  - Wordpress;
  - Joomla;
- Introdução ao Bootstrap:
  - *Container*;
  - *Grid*;
  - *Text-style*;
  - *Tables*;
  - *Images*;
  - *Buttons*;
  - *Alerts*;
  - *Panels*;
  - *Navbar*;
  - *PopOver*;
  - *Carousel*;
  - Etc...
- Introdução ao CSS:
  - Ionic
    - Criação de Páginas;
    - Navegação por *tabs*;
    - Listas;
    - *Lands*;
    - Botões;
    - Menus;
    - Alertas;
    - Etc...
- Aprendizagem de LaTeX, para fazer relatórios, tutoriais, utilizando código.

## **Resumo**

No âmbito do TesP de Desenvolvimento de Aplicações Informáticas foi realizado na SASTI, entre os dias 1 de Março e 14 de Julho de 2017, o estágio, de modo a colocar em prática todos os conhecimentos teóricos e práticos adquiridos ao longo do curso e adquirir novos conhecimentos. Este estágio teve a duração de 750 horas.

Durante este estágio foram realizadas todas as atividades previstas no plano de estágio.

Foi-me também fornecido pelo supervisor alguns vídeos a ensinar como trabalhar com outros “programas” (joomla, LaTeX, bootstarp), nos quais foram também realizados vários exemplos, para que fica-se a perceber melhor como tais funcionavam.

## **Agradecimentos**

Queria agradecer ao IPG por me ter dado esta oportunidade, pois aprendi varias coisas acerca do mundo do trabalho e também coisas relacionadas ao curso, queria agradecer também ao meu supervisor de estágio (Miguel Valério) agradecer também a todos os outros colegas de trabalho (José Alves, Sara Fernandes, Pedro Morais e Carla) da SASTI, todos contribuíram para o meu bom desempenho no estágio, sempre me ajudaram quando tinha dúvidas na realização de qualquer atividade, e agradecer também ao meu professor orientador Luís Figueiredo.

Gostaria de agradecer a professora Natália Gomes por me ter ajudado com a correção do relatório de estágio.

Por fim, encaminho o meu agradecimento aos meus pais que sempre me acompanharam e apoiaram ao longo desta etapa.

## Índice

|  |    |
|--|----|
| CAPITULO 1 – INTRODUÇÃO.....   | 1  |
| CAPITULO 2 – CARACTERIZAÇÃO DA EMPRESA .....   | 2  |
| CAPITULO 3 – Atividades Desenvolvidas Durante o Estágio .....                                      | 3  |
| 3.1 – Instalação de bastidores .....   | 3  |
| 3.2 – Reparação de Computadores .....  | 3  |
| 3.3- Manutenção preventiva de impressoras no Hospital da Guarda .....                              | 3  |
| 3.4- Visualização de tutoriais para a aquisição de mais conhecimentos na área da programação ..... | 3  |
| 3.4.1-Wordpress.....   | 4  |
| CAPITULO 4 – PROJETO DESENVOLVIDO NO ESTÁGIO.....  | 5  |
| 4.2 – Objetivos .....  | 5  |
| 4.3 – O projeto .....  | 5  |
| CAPITULO 5 – CONCLUSÃO .....   | 16 |
| Bibliografia.....  | 17 |

## Índice de Figuras

|  |    |
|--|----|
| Figura 1 - Logótipo da Empresa. ....                           | 2  |
| Figura 1- Wordpress.....                                       | 4  |
| Figura 2- Criação do projeto.....                              | 5  |
| Figura 3- Criação de páginas.....                              | 6  |
| Figura 4- App Module .....                                     | 6  |
| Figura 5- Tabs Html.....                                       | 7  |
| Figura 6- Tabs.ts .....  | 7  |
| Figura 7- App Component.....                                   | 7  |
| Figura 8- Página Tabs.....                                     | 8  |
| Figura 9- Lista de Compras HTML.....                           | 8  |
| Figura 10- Lista de Compras .....                              | 9  |
| Figura 11- Receitas.....                                       | 10 |
| Figura 12- Nova Receita .....                                  | 11 |
| Figura 13- Firebase.....                                       | 12 |
| Figura 14- Firebase Login .....                                | 12 |
| Figura 15- Codigos de Login da Firebase .....                  | 13 |
| Figura 16- Autenticação.....                                   | 13 |
| Figura 17- PopOver.....  | 14 |
| Figura 18- Código para Carregar dados para a Firebase.....     | 14 |
| Figura 19- Código para Carregar os dados para a Aplicação..... | 14 |



## **CAPITULO 1 – Introdução**

Este relatório surge na sequência do estágio curricular do Curso de Técnico Superior Profissional (TeSP), de Desenvolvimento de Aplicações Informáticas. O estágio realizou-se na SASTI, de 01 de Março a 14 de Julho de 2017. Durante o estágio foram realizadas e implementadas as ferramentas e atividades previstas no plano de estágio.

Para além das atividades desenvolvidas no plano de estágio foram também realizadas outras atividades, tais como reparação de computadores, manutenção de impressoras, instalação de bastidores e visualização de alguns cursos online (wordpress, ionic), nos quais foram realizados vários exemplos de sites/aplicações e criação de tutoriais.

Estes tutoriais serão apresentados neste relatório em Anexos.

O presente relatório divide-se em cinco capítulos incluindo o da Introdução.

O segundo capítulo apresenta a empresa onde foi realizado o estágio, o terceiro capítulo fala das atividades realizadas durante o estágio, o quarto fala do projeto realizado na empresa ao longo do estágio, e por fim no quinto capítulo está a conclusão. No final destes cinco capítulos encontram-se os anexos, aqui (anexos) estão

## CAPITULO 2 – Caracterização da empresa



Figura 1 - Logótipo da Empresa.

Neste capítulo vamos fazer uma breve apresentação da SASTI.

“A SASTI é uma Sociedade Lda. A empresa que atua na área da comercialização de soluções para tecnologias de informação, tira proveito da experiência profissional dos seus colaboradores, os quais detêm bons conhecimentos quer das exigências atuais dos mercados quer ao nível das soluções que adequa e implementa nos seus clientes. A SASTI trabalha constantemente para melhorar a qualidade global dos serviços que presta e as soluções específicas e quase personalizadas que implementa, procurando ser vista pelos seus clientes como o parceiro de confiança para as soluções, que estes procuram, nas áreas das TIC. Está presentemente, muito vocacionada para consultoria e serviços. A sua base de clientes é formada maioritariamente (cerca de 95%) por empresas ou instituições estando estas divididas por PMEs, algumas grandes empresas, e acima de tudo instituições de solidariedade social. A carteira de clientes, justificada também pela juventude da empresa, tem crescido desde a sua criação. Estima-se um crescimento de 10% ao ano a partir de 2014.”

<http://www.sasti.pt/index.php/sasti/historial>

## **CAPITULO 3 – Atividades Desenvolvidas Durante o Estágio**

Para além do plano de estágio, foram realizadas varias atividades na empresa tais como:

- Instalação de bastidores;
- Reparação de Computadores;
- Manutenção preventiva de impressores no Hospital da Guarda;
- Visualização de tutoriais para a aquisição de mais conhecimento na área da programação;

### **3.1 – Instalação de bastidores**

Ao início deste estágio foi-me dito por parte de um dos trabalhadores da empresa que teria que ir com ele para um lar fazer a instalação de bastidores, e a interligação de 3 edifícios com fibra.

### **3.2 – Reparação de Computadores**

Durante este estágio, foi também pedido para reparar alguns computadores de clientes.

### **3.3- Manutenção preventiva de impressoras no Hospital da Guarda**

No decorrer do estágio, foi-me pedido por parte de um colega de trabalhar na empresa para ir com ele fazer a manutenção das impressoras do Hospital da Guarda, para tal foi visto se alguma das impressoras precisava de toner, foi retirada uma folha com o número total de páginas imprimidas em cada impressora, e foram limpas as partes com mais desgaste na impressora, foi também feita a reparação de algumas impressoras, substituindo as peças avarias por peças novas.

### **3.4– Visualização de tutoriais para a aquisição de mais conhecimentos na área da programação**

Ao longo deste estagio foram me fornecidos vários tutoriais para a aquisição de mais conhecimentos nesta área. E dos quais foram realizados vários exemplos de aplicações/websites. Para completar foram criados por mim dois tutoriais, um deles do Bootstrap e outro do Wordpress, os quais podem ser visualizados na secção dos anexos.

### 3.4.1-Wordpress



Figura 2- Wordpress

O WordPress é uma “aplicação” para gerir o conteúdo de um site para web feito em PHP com a base de dados *MySQL*, o WordPress serve para a criação de sites e blogs através da web. O wordpress é uma das ferramentas mais conhecidas para a criação de blogs, porem este é mais utilizado por quem quer um site mais profissional e com mais recursos.

## CAPITULO 4 – Projeto Desenvolvido no Estágio

O projeto proposto pela empresa é criar uma aplicação feita em ionic, esta aplicação será uma lista de compras e receitas, que ira conter um login e base de dados (firebase), para guarda os dados que colocamos na aplicação, caso contrário a aplicação não iria ter os nossos dados gravados quando a mesma fosse reaberta.

### 4.2 – Objetivos

Para este projeto é necessário instalar o node.js.

Depois do node.js instalado vamos instalar o ionic, para tal vamos colocar na linha de comandos o seguinte comando:

`Npm install -g corvoda ionic.`

Para criar o projeto vamos a consola e colocamos o seguinte comando: `ionic start nome_do_projeto blank` (blank é para criar um projeto sem nada).

### 4.3 – O projeto

Para iniciar este projeto teremos de o criar através da linha de comandos, utilizando o comando “Ionic start receitas blank”, como podemos visualizar na Figura 2.

```
C:\AllexPina>ionic start receitas blank
? Creating directory .\receitas - done!
[INFO] Fetching app base
      (https://github.com/ionic-team/ionic2-app-base/archive/master.tar.gz)
? Downloading - done!
[INFO] Fetching starter template blank
      (https://github.com/ionic-team/ionic2-starter-blank/archive/master.tar.gz)
? Downloading - done!
? Updating package.json with app details - done!
? Creating configuration file ionic.config.json - done!
[INFO] Installing dependencies may take several minutes!
> npm install
? Running command - done!
> npm install --save-dev --save-exact @ionic/cli-plugin-ionic-angular@latest
? Running command - done!
> git init
? Running command - done!
> git add -A
? Running command - done!
> git commit -m "initial commit" --no-gpg-sign
? Running command - done!

? ? ? Your Ionic app is ready to go! ? ? ?

Run your app in the browser (great for initial development):
  ionic serve

Run on a device or simulator:
  ionic cordova run ios

Test and share your app on a device with the Ionic View app:
  http://view.ionic.io

? Link this app to your Ionic Dashboard to use tools like Ionic View? Yes

You will need to login in order to link this app. Please run the following commands to do so.
  ionic login - login first
  ionic link - then link your app

Go to your newly created project: cd .\receitas
C:\AllexPina>_
```

Figura 3- Criação do projeto

Para este projeto teremos que criar várias páginas, para tal utilizamos o seguinte comando, descrito na Figura 3.

```
C:\AllexPina\receitas>ionic g page tabs  
[OK] Generated a page named tabs!
```

Figura 4- Criação de páginas

Nesta aplicação foram criadas as seguintes páginas:

- Editar Receita;
- Receita;
- Receitas;
- Lista de Compras;
- Entrar;
- Tabs;

Para que as páginas funcionem, temos de as declarar na nossa app.module.ts e importar, tal como indica a Figura 4.

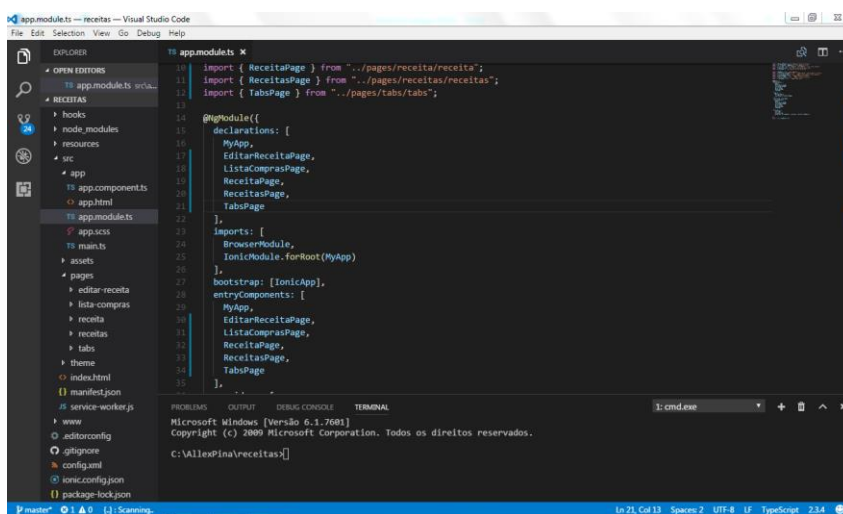


Figura 5- App Module

Depois das páginas criadas vamos começar por implementar a página tabs:

Esta página irá conter dois ícones, as linhas de código da Figura 5 serão utilizadas para tal.

```
<ion-tabs>
  <ion-tab [root]="lcPage" tabIcon="list-box" tabTitle="Lista de Compras"></ion-tab>
  <ion-tab [root]="receitasPage" tabIcon="list-box" tabTitle="Receitas"></ion-tab>
</ion-tabs>
```

Figura 6- Tabs Html

Para que a página funcione é necessário importar as páginas que precisamos para a tabs.ts, como podemos observar na Figura 6.

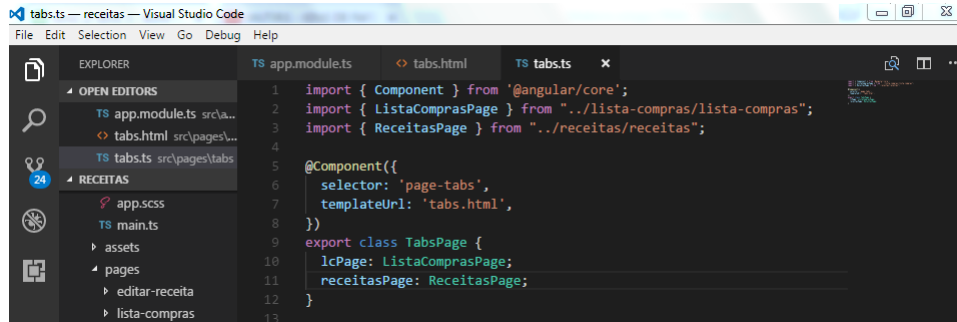


Figura 7- Tabs.ts

De seguida alteramos a nossa página inicial da aplicação para que a partir de agora seja a página tabs, esta alteração será realizada na app.component.ts como podemos verificar na Figura 7.

```
1 import { Component } from '@angular/core';
2 import { Platform } from 'ionic-angular';
3 import { StatusBar } from '@ionic-native/status-bar';
4 import { SplashScreen } from '@ionic-native/splash-screen';
5
6 import { TabsPage } from '../pages/tabs/tabs';
7 @Component({
8   templateUrl: 'app.html'
9 })
10 export class MyApp {
11   rootPage:any = TabsPage;
12
13   constructor(platform: Platform, statusBar: StatusBar, splashScreen: SplashScreen) {
14     platform.ready().then(() => {
15       // Okay, so the platform is ready and our plugins are available.
16       // Here you can do any higher level native things you might need.
17       statusBar.styleDefault();
18       splashScreen.hide();
19     });
20   }
21 }
```

Figura 8- App Component



Figura 9- Página Tabs

A nossa página tabs está completa como podemos ver na Figura 8. Agora iremos passar para a Lista de Compras. Podemos visualizar o código desenvolvido em html para a Lista de Compras na Figura 9.

A screenshot of the Visual Studio Code editor showing the HTML code for a shopping list page. The Explorer sidebar on the left shows the project structure, with the 'shopping-list.html' file selected. The main editor area displays the following HTML code:

```
1 <ion-header>
2 <ion-navbar>
3 <ion-buttons start>
4 <button ion-button icon-only menuToggle>
5 <ion-icon name="menu"></ion-icon>
6 </button>
7 </ion-buttons>
8 <ion-buttons end>
9 <button ion-button icon-only (click)="onShowOptions($event)">
10 <ion-icon name="more"></ion-icon>
11 </button>
12 </ion-buttons>
13 <ion-title>Lista de Compras</ion-title>
14 </ion-navbar>
15 </ion-header>
16 <ion-content padding>
17 <form #f="ngForm" (ngSubmit)="onAddItem(f)">
18 <ion-list>
19 <ion-item>
20 <ion-label fixed>Nome</ion-label>
21 <ion-input type="text" name="ingredientName" placeholder="Milk" ngModel required</ion-input>
22 </ion-item>
23 <ion-item>
24 <ion-label fixed>Quantidade</ion-label>
25 <ion-input type="number" name="amount" placeholder="2" ngModel required</ion-input>
26 </ion-item>
27 </ion-list>
28 <button ion-button type="submit" block [disabled]="!f.valid">Adicionar</button>
29 </form>
30 </ion-content>
31 <ion-list>
32 <ion-item *ngFor="let item of listItems; let i = index" (click)="onCheckItem(i)">
33 <h3 padding>{{ item.name }} ({{ item.amount }})</h3>
34 </ion-item>
35 </ion-list>
```

Figura 10- Lista de Compras HTML

Na pasta que contém a lista de compras iremos ter um cabeçalho (header) com dois botões, um que será o menu e outro com: “...” que ira ter dois botões quando aberto, um para carregar a nossa lista que foi gravada na base de dados e outro botão que ira servir para gravar a lista.



Depois do cabeçalho iremos então ter na nossa página duas labels (Nome/ Quantidade) a frente de cada label teremos um input, que será para colocarmos os nossos itens.

Para finalizar teremos o nosso botão que só será clicável quando colocarmos texto no Nome e Quantidade.

Ao ser adicionado um item, este será visualizado por baixo do botão adicionar, e para remover um item basta clicar nele. Podemos visualizar o resultado final desta página na Figura 10.

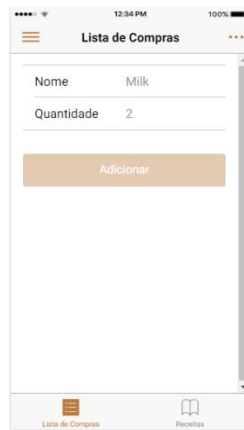


Figura 11- Lista de Compras

Com esta página concluída vamos então começar a trabalhar na página das receitas.

Esta página irá conter um botão para adicionar uma receita e outro para carregar ou gravar as receitas na base de dados, irá conter também outro botão que será o menu, este menu servirá para o logout e para passar para a Lista de compras.

Na Figura 11 podemos visualizar a página das receitas.

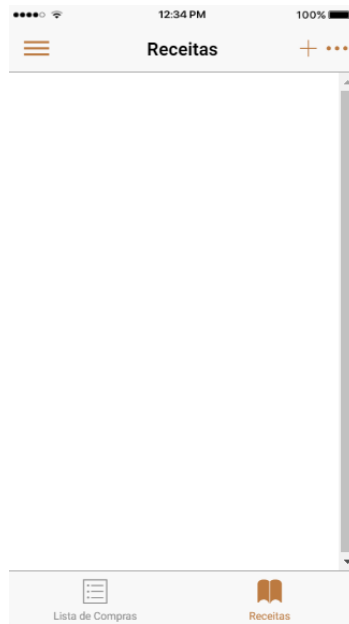


Figura 12- Receitas

Vamos agora criar a página para adicionar receitas na nossa lista. Esta página terá o nome da receita, uma descrição e a dificuldade, para além disto terá também um botão para editar ingredientes, que permitirá adicionar ou eliminar os ingredientes, para finalizar irá ter um botão que fará com que a receita seja adicionada as receitas. Na Figura 12 podemos visualizar a página para criar uma nova receita.

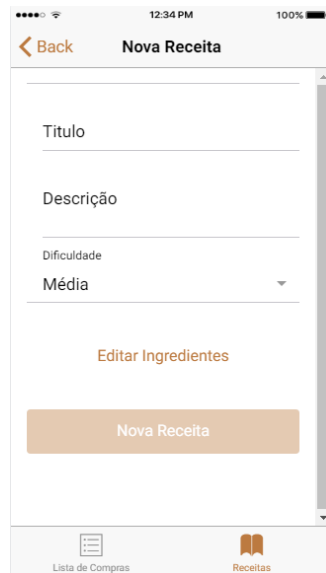


Figura 13- Nova Receita

Depois das páginas criadas vamos passar para a parte da base de dados, e como base de dados iremos utilizar um serviço chamado firebase.

Firebase é uma ferramenta do Google e ao mesmo tempo uma infra-estrutura para desenvolver aplicações com maior rapidez e flexibilidade.

Para ligar a nossa app ao firebase teremos de fazer o login com uma conta google no firebase.

Depois de efetuado o login no site vamos clicar na opção aceder a consola e criar um novo projeto (adicionar projeto).

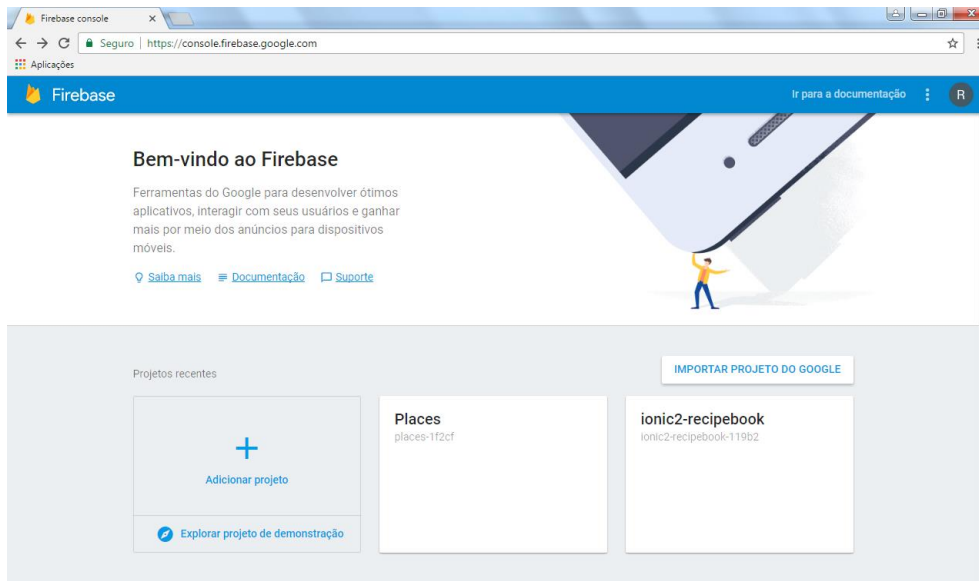


Figura 14- Firebase

Na figura 13 podemos visualizar a página inicial da Firebase.

Ao clicar em adicionar projeto será aberta uma janela, na qual teremos que colocar o nome do projeto e o país onde nos encontramos. Com os campos acima preenchidos vamos clicar em adicionar.

Para o nosso login iremos precisar de autorizar a autenticação, para tal vamos ao menu e clicamos em “authentication” e onde diz e-mail teremos de alterar o status para ativado (Figura 14).

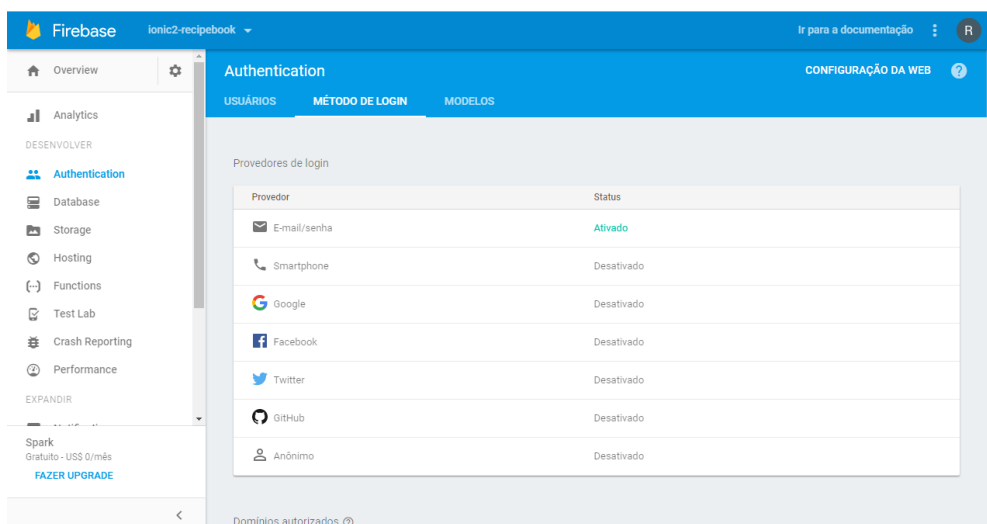


Figura 15- Firebase Login

É necessário instalar o plugin da firebase no nosso projeto para que ele funcione, para tal utilizamos o seguinte comando na cmd:

```
npm install --save firebase
```

Para começar a utilizar este plugin vamos ao firebase (site) na autenticação temos um botão que diz “configuração web”, clicamos nele e podemos ver que é aberta uma janela com código (Figura 15):

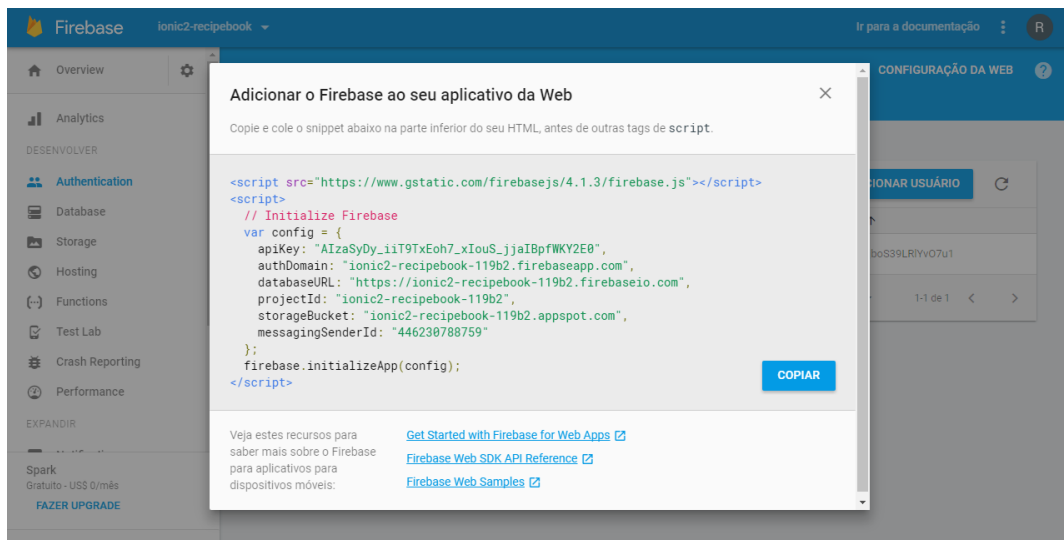


Figura 16- Codigos de Login da Firebase

Para iniciar a firebase vamos copiar apenas a apiKey e a authDomain da figura acima, e colocar no ficheiro dos componentes (app.component.ts), é também necessário fazer o import da firebase no mesmo ficheiro.

Para que a autenticação funcione temos de criar um novo serviço (Figura 16) e, para tal vamos criar um ficheiro com os “comandos” necessários para tal.

Este serviço será utilizado nas páginas Entrar e Criar Conta.

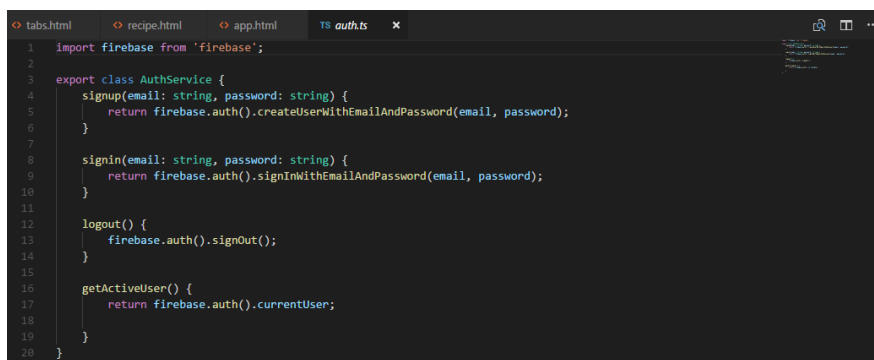


Figura 17- Autenticação

Para guardar os dados colocados na página das receitas ou na da lista de compras é necessário criarmos um popover com dois botões (Figura 17), um deles para carregar a lista contida na base de dados para a aplicação e outro para guardas os dados da aplicação para a base de dados.

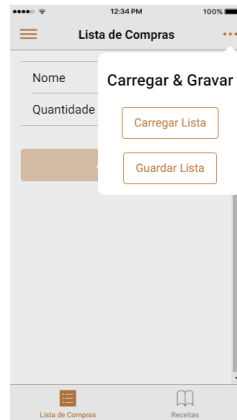


Figura 18- PopOver

Para que este botão (guardar) carregue a informação na base de dados temos de ir buscar um token que é dado por parte da firebase (Figura 18).

```
storeList(token: string) {
  const userId = this.authService.getActiveUser().uid;
  return this.http
    .put('https://ionic2-recipebook-119b2.firebaseio.com/'
      + userId + '/shopping-list.json?auth='
      + token, this.ingredients)
    .map((response: Response) => {
      return response.json();
    });
}
```

Figura 19- Código para Carregar dados para a Firebase

Para que o outro botão carregue os dados para a aplicação temos de ir buscar o token acima e carrega-lo para a aplicação (Figura 19):

```
fetchList(token: string) {
  const userId = this.authService.getActiveUser().uid;
  return this.http.get('https://ionic2-recipebook-119b2.firebaseio.com/'
    + userId + '/shopping-list.json?auth='
    + token)
    .map((response: Response) => {
      return response.json();
    })
    .do((ingredients : Ingredient[])=> {
      if (ingredients) {
        this.ingredients = ingredients
      } else {
        this.ingredients = [];
      }
    });
}
```

Figura 20- Código para Carregar os dados para a Aplicação

Para que a app possa ser executada num dispositivo android teremos de fazer o download e instalação do jdk e do android studio.

Após a instalação dos mesmos, teremos de ir a consola e correr o seguinte comando na pasta do projeto:

```
Cordova platform add android
```

Isto fara com que tenhamos a plataforma android instalada no projeto.

Para compilar o projeto para um apk corremos o seguinte comando:

```
Cordova build android
```

O apk do nosso projeto irá encontrar-se dentro da pasta do projeto:

```
projeto/platforms/android/build/outputs/apk
```

## **CAPITULO 5 – Conclusão**

Com este estágio consegui por em prática muitos dos conhecimentos, que tenho adquirido ao longo do curso, adquiri também novos conhecimentos, como por exemplo, programar em HTML, foi uma boa experiência para mim pois consegui atingir os objetivos propostos.

Ao longo destes quatro meses de estágio, adquiri bastante experiência, que vai ser-me útil no futuro.

É com muito apreço que agradeço ao pessoal da SASTI pelo apoio demonstrado ao longo desta caminhada, e com especial atenção para o meu supervisor de estágio, Miguel Valério pela sua disponibilidade e paciência que demonstrou.



## **Bibliografia**

**UDEMY, "Curso de Ionic", Consultado a 10/05/2017 em**

<https://www.udemy.com/ionic-2-the-practical-guide-to-building-ios-android-apps/>

**Website utilizado para o login e base de dados do projeto. Consultado a 20/06/2017:**

<https://console.firebase.google.com>

**Resumo do que é o Wordpress, Consultado a 06/07/2017:**

<https://pt.wikipedia.org/wiki/WordPress>

**Resumo do que é o Bootstrap, Consultado a 06/07/2017:**

[https://en.wikipedia.org/wiki/Bootstrap\\_\(front-end\\_framework\)](https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework))

## Anexo A

### Tutorial de Bootstrap:



Figura 1 Bootstrap

O Bootstrap (Figura 1) é uma estrutura *web open-source* que tem como objetivo projetar *websites* e aplicações *web*. O Bootstrap é baseado em HTML e CSS, para criar botões, formas e barras de navegação, entre outros. Este (Bootstrap) é bastante responsivo, visto que se adapta a todos os ecrãs, pois podemos aumentar ou diminuir o tamanho da página e tudo o que está dentro dela aumenta ou diminui.

## 1. Container

O *container* cria um pequeno espaço do lado direito e esquerdo da página.

Ex:

```
<div class="container-fluid">
  <header class="jumbotron">
    <h1>Heading</h1>
  </header>
</div>
```

Se colocarmos o *fluid* a frente do *container* vai fazer com que o espaço que fica de cada lado do *header* seja menor.

## 2. Grid System

O bootstrap tem uma *grid* de 12 colunas, porem nem sempre é necessário utilizar todas as colunas.

Com o *grid system* não precisamos de nos preocupar com os padrões, preenchimento, margens...

Ex:

```
<div class="container-fluid">
  <div class="col-md-5" style="background: blue">Hello </div >
  <div class="col-md-2" style="background: orange">Hello </div >
  <div class="col-md-5" style="background: red">Hello </div >
</div >
```

Se a junção de todos os valores que se seguem ao md for maior que 12 vamos visualizar que o ultimo *Hello* irá passar para a linha seguinte.

Ao colocarmos a *grid* dentro de uma "row" (`<div class="row" > . . . </div>`) podemos verificar que esta linha não está dentro do *container*, logo, irá ocupar todo o espaço da linha.

### 3. Tipos de texto

- `<h1></h1>` (Só pode existir um);
- `<h2></h2>`;
- `<h3></h3>`;
- `<h4></h4>`;
- `<h5></h5>`;
- `<h6></h6>`;
- `<blockquote></blockquote>` - No *blockquote* podemos verificar que existe uma linha antes do texto;
- `<abbr></abbr>` - Na abreviatura podemos indicar o seu significado, e para o visualizar mos basta passar o rato em cima da palavra;
- `<mark></mark>` - O texto inserido fica como um fundo mais escuro;
- `<dl clash="dl-horizontal">`
  - `<dt><dt>` - *Titulo*
  - `<dd><dd>` - *Texto*
- `</dl>`;
- `<p></p>` - Paragrafo;
- `<p class "text-success"></p>` - Paragrafo a verde;
- `<p class "text-danger"></p>` - Paragrafo a vermelho;
- `<p class "text-info"></p>` - Paragrafo a azul;
- `<p class "text-primary"></p>` - Paragrafo a azul (um pouco mais claro);
- `<p class "text-muted"></p>` - Paragrafo a cinzento;
- `<p class "text-warning"></p>` - Paragrafo a castanho;
- `<p class "text-success bg-success"></p>` - Paragrafo com o fundo a verde (pode ser outra cor, basta alterar o *success* para outro valor);
- `<p class "text-primary bg-primary"></p>` - Paragrafo com o fundo azul, neste caso visto que as duas cores são primarias o texto fica a branco);
- `<p class "lead"></p>` - Paragrafo com o tamanho de letra maior
- `<p class "small"></p>` - Paragrafo com o tamanho de letra menor
- `<p class "text-right"></p>` - Paragrafo a direita;
- `<p class "text-center"></p>` - Paragrafo ao centro;
- `<p class "text-uppercase"></p>` - Paragrafo com letras maiúsculas;
- `<p class "text-lowercase"></p>` - Paragrafo com letras minúsculas;

## 4. Tabelas

```
<table class="table">
  <thead>(Cabeçalho)
    <tr>
      <th>S. no</th>
      <th>Name</th>
      <th>Paid</th>
      <th>Not Paid</th>
    </tr>
  </thead>
  <tbody>(Corpo do texto)
    <tr>
      <td>1</td> (texto correspondente ao texto da primeira linha/coluna do
cabeçalho)
      <td>Rui</td>
      <td>500</td>
      <td>200</td>
    </tr>
  </tbody>
</table>
```

Podemos visualizar o resultado obtido com estas linhas de código na Figura 22.

| S. no | Name | Paid | Not Paid |
|-------|------|------|----------|
| 1     | Rui  | 500  | 200      |

Figura 2- Tabela Bootstrap

Podemos alterar o formato do texto da tabela da seguinte forma:

- `<table class="table table-striped">`
- `<table class="table table-condensed">` (remove o espaço em excesso)
- `<table class="table table-hover">`
- `<table class="table table-bordered">` (a tabela foca com bordas)

Podemos também alterar a cor de fundo de cada linha/coluna da tabela:

`<tr class="danger">` (neste exemplo estamos a alterar a cor de uma linha inteira)

`<th>S. no</th>`

`<th>Name</th>`

`<th>Paid</th>`

`<th>Not Paid</th>`

`</tr>`

`<td class="active">Rui</td>` (neste exemplo estamos a alterar apenas uma coluna de uma determinada linha)

## 5. Imagens

Para adicionar uma imagem na nossa página é necessário o seguinte código:

- `` (para localizar a nossa imagem temos que indicar a pasta e o nome da imagem)

O código em cima descrito será colocado dentro do `<body> => <div>`;

Para formatar a imagem basta utilizar o seguinte código (ex):

- ``

A `thumbnail` faz com que tenhamos uma borda a volta da imagem, podemos também alterar a `class` onde está escrito `"img-thumbnail"` para `"img-circle"`.

Se alterarmos a `"img-thumbnail"` para apenas `"thumbnail"` só iremos visualizar essa imagem nessa mesma linha.



Figura 3- Imagens Bootstrap

## 6. Botões

Para adicionar um botão na nossa página é necessário o seguinte código:

```
<button>Click Me</button>
```

Existem vários tipos de botões (Figura 4), como por exemplo:

- `<button class="btn">Click Me</button>`
- `<button class="btn btn-default">Click Me</button>`
- `<button class="btn btn-primary">Click Me</button>`
- `<button class="btn btn-info">Click Me</button>`
- `<button class="btn btn-success">Click Me</button>`
- `<button class="btn btn-danger">Click Me</button>`
- `<button class="btn btn-link">Click Me</button>`
- `<button class="btn btn-danger">Click Me</button>`
- `<button class="btn btn-danger btn-block">Click Me</button>` (faz com que o botão ocupe a linha inteira)
- `<button class="btn btn-danger disabled">Click Me</button>` (o botão fica desativo)
- `<button class="btn btn-danger active">Click Me</button>` (o botão fica ativo)

Podemos também alterar o tamanho do botão:

- `<button class="btn btn-default btn-lg">Click Me</button>` (large)
- `<button class="btn btn-default btn-sm">Click Me</button>` (small)
- `<button class="btn btn-default btn-md">Click Me</button>` (medium)
- `<button class="btn btn-default btn-xs">Click Me</button>` (extra small)

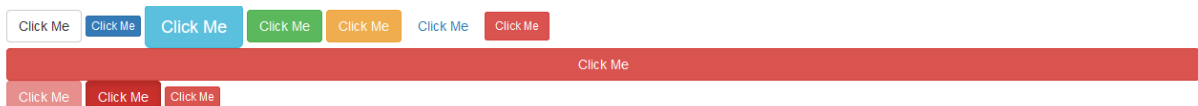


Figura 4 - Botões Bootstrap

### 6.1. Grupo de botões (*button-group*)

Para agrupar botões (Figura 5) é necessário colocar o seguinte código:

```
<div class="btn-group">  
  <button class="btn btn-info">Home</button>  
  <button class="btn btn-info">Home</button>  
  <button class="btn btn-info">Home</button>  
  <button class="btn btn-info">Home</button>  
</div>
```



Figura 5- Grupo de Botões

Para colocar os botões na vertical (Figura 6) basta colocar o seguinte:

```
<div class="btn-group-vertical">  
  <button class="btn btn-info">Software</button>  
  <button class="btn btn-info">Software</button>  
  <button class="btn btn-info">Software</button>  
</div>
```

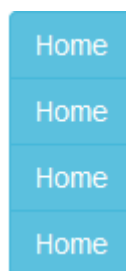


Figura 6- Grupo de Botões na Vertical

Para que os botões que estão agrupados ocupem uma linha inteira (Figura 7) é necessário colocar um `<div>` `</div>` para cada botão, caso contrário cada botão irá ocupar uma linha.



Figura 7 - Grupo de Botões em Linha



## 6.2. Dropdown Button

Ao clicar num dropdown button (Figura 8) ele abre um “menu” com outros botões, para o criar será necessário o seguinte código:

```
<div class="dropdown">  
  <button class="btn btn-primary dropdown-toggle" data-toggle="dropdown">  
Article <span class="caret"></span>  
  <ul class="dropdown-menu"> (é criado um menu com vários botões)  
    <li><a href="">Technology</a></li>  
    <li><a href="">Science</a></li>  
    <li><a href="">Healt</a></li>  
    <li class="divider"></li> (cria uma divisão entre os botões criados anteriormente  
e os que são criados de seguida)  
    <li class="dropdown-header">Personal</li>  
    <li><a href="">Logout</a></li>  
  </ul>  
</div>
```

Para que a lista de botões fique à direita basta alterar mos o seguinte código:

```
<ul class="dropdown-menu dropdown-menu-right">
```

Para que os botões sejam visualizados para cima basta alterar o “dropdown” para “dropup”, como podemos visualizar a seguir:

```
<div class="dropdown"><div class="dropup">
```



Figura 8- Dropdown Button

## 7. Glyphicons

Para adicionar mos um *glyphicon* utilizamos o seguinte código:

```
<span class="glyphicon glyphicon-camera"></span>
```

Para colocar um *glyphicon* num botão:

```
<button class="btn btn-success "><span class="glyphicon glyphicon-search"></span>Search</button>
```

Cada *glyphicon* tem um código, e podemos visualiza-los no seguinte *website*:

<http://glyphicons.bootstrapcheatsheets.com/>

## 8. Cabeçalho e Jumbotron

```
<div class="container-uid">  
  <div class="jumbotron">  
    <h1>The Jumbotron</h1>  
  </div>  
  <h1>The heading without Jumbotron</h1>  
</div>
```

Neste caso existem dois “<h1></h1>”, podemos utilizar os dois, pois cada um se encontra numa classe diferente.

## 9. Janelas

Para criar uma “janela” (Figura 9) utilizamos o seguinte código.

```
<p class="well">This is the well</p>
```

Ex:

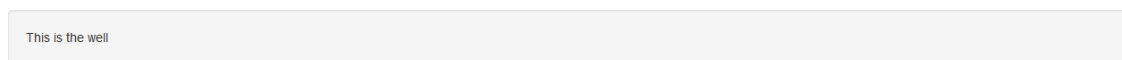


Figura 9- Janela Bootstrap

## 10. Alertas

Para criar mos um alerta basto utilizar o seguinte código:

```
<div class="alert alert-danger fade in"><a href="" class="close" data-dismiss =  
"alert" area-label="close" >& times;</a>The file will be deleted! </div>
```



Figura 10- Alertas Bootstrap

## 11. Labels e Badges

*Badge:*

```
<a href="">Messages <span class="badge">6</span></a>
```

*Badge dentro de um botão:*

```
<button class="btn btn-primary">Click to see the notifications <span class="badge">  
310 </span></button>
```

*Label:*

```
<div class="page-header"><h2>Downloads <span class="label label-default"> Beta  
</span></h2>
```

Podemos visualizar o resultado dos códigos anteriores na Figura 11.

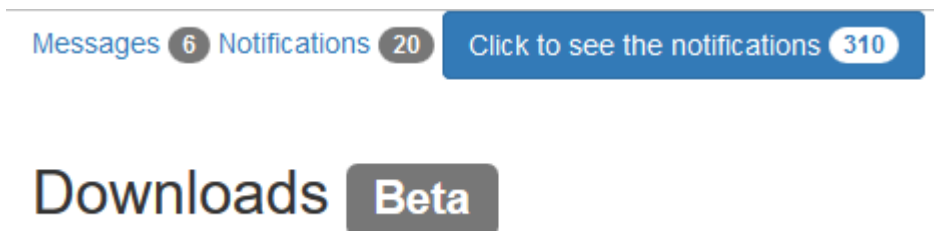


Figura 11- Labels e Badges

## 12. Painel

O seguinte código tem como objetivo criar um painel (Figura 12).

```
<div class="panel panel-default">
  <div class="panel-heading">
    <h4>Panel Heading</h4>
  </div>
  <div class="panel-body">
    <p>This is the Panel Body</p>
  </div>
  <div class="panel-footer">Panel Footer</div>
</div>
```

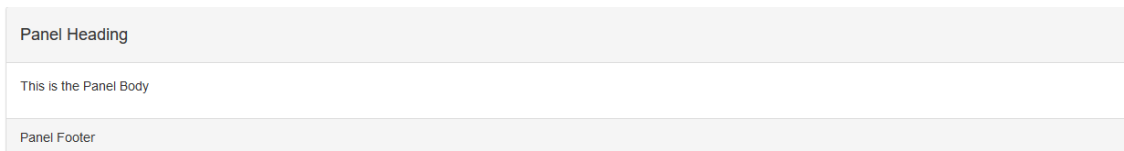


Figura 12- Painel Bootstrap

## 13. Collapsible

Podemos utilizar um *collapsible* (Figura13) num botão como podemos verificar no exemplo seguinte:

```
<button class="btn btn-success" data-target="#cla" data-toggle="collapse">Click to Collapse</button>
<div id="cla" class="collapse in">This is the Collapsible Element</div>
```



Figura 13- Collapsible Bootstrap

### 13.1. Collapsibles em Painéis

Os *collapsibles* podem também ser utilizados dentro de painéis como por exemplo:

```
<div class="panel panel-success">
  <div class="panel-heading">
    <h4><a href="#cla3" data-toggle="collapse">Panel Heading</a></h4>
  </div>
  <div id="cla3" class="collapse panel-collapse">
    <div class="panel-body">
      <p>This is the Panel Body</p>
    </div>
  </div>
</div>
```

Podemos observar que no exemplo anterior, quando clicamos no "Panel Heading" ele abre um *collapsible* (Figura 14):

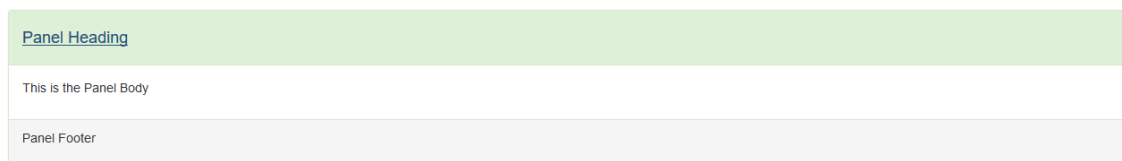


Figura 14- Collapsible Panel

### 14. List Group

Podemos criar uma *list group* (Figura 15) da seguinte forma:

```
<ul class="list-group">
  <li class="list-group-item">This is the list Element</li>
  <li class="list-group-item list-group-item-danger">This is the list Element</li>
  <li class="list-group-item">This is the list Element</li>
</ul>
```

Para criar uma lista ordenada basta substituir o `<ul></ul>` por `<ol></ol>`



Figura 15- List Group

## 15. Collapsibles, List Groups e Accordion

```
<div class="panel panel-default">
  <div class="panel-heading">
    <h4 class="panel-title">
      <a href="#cla5" data-toggle="collapse">Panel group heading</a>
    </h4>
  </div>
  <div class="panel-collapse collapse" id="cla5">
    <ul class="list-group">
      <li class="list-group-item">List no1</li>
      <li class="list-group-item">List no2</li>
      <li class="list-group-item">List no3</li>
      <li class="list-group-item">List no4</li>
    </ul>
    <div class="panel-footer">Panel Footer</div>
  </div>
</div>
```

Ao executarmos o código acima podemos verificar na figura 16 que existe um painel e ao clicar nele podemos visualizar a *list group* com vários números.

|                     |
|---------------------|
| Panel group heading |
| List no1            |
| List no2            |
| List no3            |
| List no4            |
| Panel Footer        |

Figura 16- Collapsibles, List Group e Accordion

## 16. Accordion

```
<div class="panel-group" id="accordion1">
  <div class="panel panel-default">
    <div class="panel-heading">
      <h4 class="panel-title">
        <a href="#" class="collapsed" data-toggle="collapse" data-parent="#
accordion1">Collapsible Accordion</a>
      </h4>
    </div>
    <div class="collapse panel-collapse" id="cla6">
      <div class="panel-body">This is the accordion area</div>
    </div>
  </div>
  <div class="collapse panel-collapse" id="cla6">
    <div class="panel-body">This is the accordion area</div>
  </div>
</div>
```

Se o código acima for duplicado podemos verificar que ao clicarmos no primeiro *panel group* ele expande, e se de seguida clicarmos no segundo *panel group* verificamos que o painel anterior “minimiza”, a isto chama-se *colapsible accordion* (Figura 17).

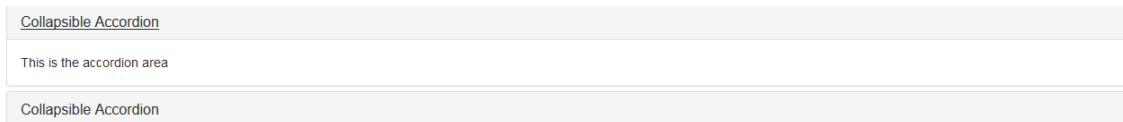


Figura 17- Accordion

## 17. Aba

Para criarmos uma aba (Figura 18) basta utilizar o seguinte código:

```
<div class="container-uid">
  <ul class="nav nav-tabs"> (aba de navegação)
    <li><a href="#">Home</a></li>
    <li class="active"><a href="#">Software</a></li> (ao adicionar uma
class="active" podemos ver que a aba Software fica selecionada)
    <li><a href="#">Mobile</a></li>
    <li><a href="#">Technology</a></li>
  </ul>
</div>
```



Figura 18- Aba Bootstrap

Podemos também adicionar um menu para uma aba (Figura 19):

```
<li class="active"><a href="#">Software</a></li>
<li class="dropdown">
  <a class="dropdown-toggle" href="#" data-toggle="dropdown">Mobile<span
class="caret"></span></a>
  <ul class="dropdown-menu">
    <li><a href="#">Mobile Info</a></li>
    <li><a href="#">Mobile Tech</a></li>
    <li><a href="#">Mobile Articles</a></li>
  </ul>
</li>
```

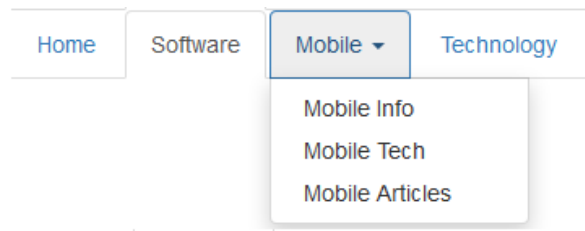


Figura 19- Menu Tab

### 18. Pills

É igual a *tabs* (aba), porém com estilo diferente, como vemos na Figura 20:



Figura 20- Pills

### 19. Dynamic Pills and Tabs

Igual aos anteriores, porém é mais dinâmico, tal como diz o nome.

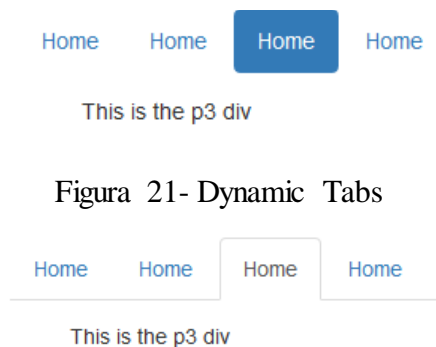


Figura 21- Dynamic Tabs

Figura 22- Dynamic Pills



## 20. Páginas e Paginação

Podemos utilizar o seguinte código para avançar para a próxima página ou retroceder para a anterior:

```
<ul class="pager">
  <li class="previous"><a href="#">Previous</a></li>
  <li class="next"><a href="#">Next</a></li>
</ul>
```

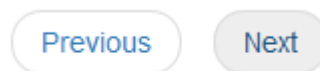


Figura 23- Páginas

No código seguinte podemos visualizar que a página onde nos encontramos:

```
<div class="text-center">
  <ul class="pagination pagination-sm">
    <li><a href="#">1</a></li>
    <li><a href="#">2</a></li>
    <li class="active"><a href="#">3</a></li>
    <li><a href="#">4</a></li>
    <li><a href="#">5</a></li>
  </ul>
</div>
```

O resultado obtido é o seguinte (Figura 24):

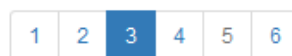


Figura 24- Paginação

## 21. Barra de Navegação (Navbar)

Para criar uma barra de navegação (Figura 25) podemos utilizar o seguinte código:

```
<nav class="navbar navbar-inverse navbar-fixed-top">  
  
  <div class="container">  
  
    <div class="navbar-header">  
  
      <a class="navbar-brand" href="#">Bootstrap</a>  
  
    </div>  
  
    <div>  
  
      <ul class="nav navbar-nav">  
  
        <li><a href="#">Home</a></li>  
  
        <li><a href="#">Software</a></li>  
  
        <li><a href="#">Music</a></li>  
  
        <li><a href="#">Contact Us</a></li>  
  
      </ul>  
  
    </div>  
  
  </div>  
  
</nav>
```

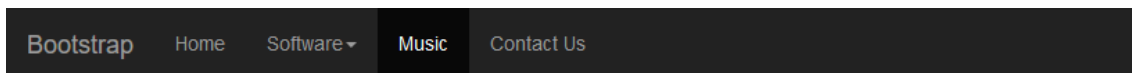


Figura 25- Navbar

## 22. Bootstrap Form

Para criar uma *form group* basta utilizar o seguinte código:

```
<form>
  <div class="form-group">
    <label for="name">Your Name</label>
    <input type="text" class="form-control">
  </div>
</form>
```

Podemos alterar o *type* para outro tipo de caracteres como por exemplo, email, password...

Podemos também alterar a *form* para um *design* diferente:

```
<form class="form-inline"> (As form group ficam todas na mesma linha)
<form class="form-horizontal">
```



Figura 26- Bootstrap Form

## 23. Estilos de validação

Existem vários estilos de validação como por exemplo:Correto, errado...

```
<form class="form">
  <div class="form-group has-feedback has-warning">
    <label>Name</label>
    <input type="text" class="form-control"><span class="glyphicon glyphicon-
warning-sign form-control-feedback"></span>
  </div>
</form>
```

Podemos alterar neste simples código a cor da *form* e o ícone conforme a validação esteja certa ou errada. Na Figura 47 podemos ver o resultado do código anterior.



Figura 27- Validação

## 24. Modals

Inicialmente iremos criar um botão, com o seguinte texto: “*Click to View Modals*”  
Depois do botão criado iremos criar o *modal*, *modal-header*, *modal-body* e o *modal-footer*, dentro do *modal-header* iremos colocar um botão para fechar a janela aberta.  
No *modal-body* iremos colocar um pequeno texto, e por fim no *modal-footer* iremos inserir um botão para fechar a janela aberta.

Exemplo (Figura 28):

```
<button class="btn btn-success" data-toggle="modal" data-target="#modal1">Click  
to View Modal</button>  
<div class="modal fade" id="modal1">  
  <div class="modal-dialog">  
    <div class="modal-content">  
      <div class="modal-header">  
        <button class="close" data-dismiss="modal">&times;</button>  
        <h4>Modal Header</h4>  
      </div>  
      <div class="modal-body">  
        <p>This is the Modal's Body</p>  
      </div>  
      <div class="modal-footer">  
        <button class="btn btn-default" data-dismiss="modal">Close</button>  
      </div>  
    </div>  
  </div>  
</div>
```



Figura 28- Modal

## 25. Carrossel

Inicialmente são criados os slides para cada imagem ser exibida, depois criamos o *carousel-inner*, que contém as imagens que serão exibidas nos slides, por fim iremos criar os botões para poder avançar para a próxima imagem:

```
<div id="carousel1" class="carousel slide" data-ride="carousel">
  <ol class="carousel-indicators">
    <li data-target="#carousel1" data-slide-to="0" class="0"></li>
    <li data-target="#carousel1" data-slide-to="1"></li>
    <li data-target="#carousel1" data-slide-to="2"></li>
  </ol>
  <div class="carousel-inner">
    <div class="item active">
      
    </div>
    <div class="item">
      
    </div>
    <div class="item">
      
    </div>
  </div>
  <a href="#carousel1" class="left carousel-control" data-slide="prev">
    <span class="glyphicon glyphicon-chevron-left" area-hidden="true"></span>
  </a>
  <a href="#carousel1" class="right carousel-control" data-slide="next">
    <span class="glyphicon glyphicon-chevron-right" area-hidden="true"></span>
  </a>
</div>
```

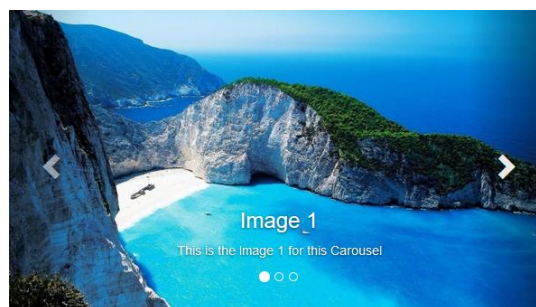


Figura 29- Carrossel

Podemos também adicionar texto para as nossas imagens, colocando o seguinte código:

```
<div class="carousel-caption">
  <h3>Image 1</h3>
  <p>This is the Image 1 for this Carousel</p>
</div>
```

Este código é colocado dentro do *<div>*, depois de colocarmos a imagem (Figura 29).

## 26. Scroll Spy

Neste passo iremos aprender a criar um *scrollspy*.

Em primeiro lugar iremos criar uma *navbar*:

```
<nav class="navbar navbar-inverse navbar-fixed-bottom">
  <ul class="nav navbar-nav">
    <li><a href="#home">Home</a></li>
    <li><a href="#software">Software</a></li>
    <li><a href="#music">Music</a></li>
    <li><a href="#contact">Contact Us</a></li>
  </ul>
</nav>
```

De seguida iremos então criar as secções para cada botão da navbar: É necessário ligar cada botão a cada secção correspondente, para tal é necessário adicionar um id a cada secção.

```
<section id="home" style="background: lightgreen">
  <h1>The Home Section</h1>
</section>
<section id="software" style="background: lightblue">
  <h1>The Software Section</h1>
</section>
<section id="music" style="background: orange">
  <h1>The Music Section</h1>
</section>
<section id="contact" style="background: grey">
  <h1>The Contact Section</h1>
</section>
```

## 27. Tooltip

Para utilizar uma *tooltip* (Figura 30) podemos utilizar o seguinte código:

```
<div class="container"><div style="height: 50px">12</div>  
  
  <button data-toggle="tooltip" data-placement="right" title="This is the button  
  "class="btn btn-success">Tooltip element</button>  
  
</div>
```

O script seguinte serve para criar a nossa tooltip:

```
<script>  
  
  $(document).ready(function(){  
  
    $('[data-toggle="tooltip"]').tooltip();  
  
  });  
  
</script>
```



Figura 30- Tooltip

## 28. PopOver

Neste primeiro *popover* (Figura 31) é necessário clicar nele, tanto para abrir como para fechar:

```
<a href="#" data-toggle="popover" title="The heading" data-content="This is the popOver Content">The PopOver</a>
```

No segundo, clicamos para abrir, porem para fechar podemos clicar fora a área:

```
<href="#" " data-toggle="popover" title="The heading" data-placement="left" data-trigger="focus" data-content="This is the popOver Content">The PopOver</a>
```

No seguinte, basta passar o rato em cima da nossa “janela” que ele irá abrir.

```
<a href="#" data-toggle="popover" title="The heading" data-placement="left" data-trigger="hover" data-content="This is the popOver Content">The PopOver</a>
```

Este é o script necessário para o PopOver:

```
<script>
  $(document).ready(function(){
    $('[data-toggle="popover"]').popover();
  });
</script>
```

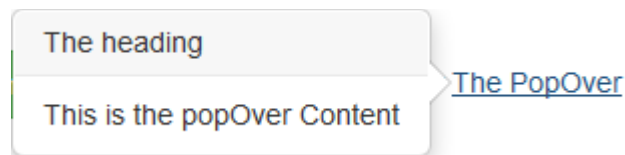


Figura 31- PopOver



## 29. Portfolio

### 29.1. Navigation and Home Area

Vamos começar por criar uma *navbar* para este projeto:

```
<nav class="navbar navbar-inverse navbar-_xed-bottom">
  <ul class="nav navbar-nav">
    <li><a href="#home">Home</a></li>
    <li><a href="#work">What I do</a></li>
    <li><a href="#">What about me</a></li>
    <li><a href="#">Contact me</a></li>
  </ul>
</nav>
```

Após ser criada a *navbar*, vamos criar as secções:

```
<section id="home" style="background: url(images/home.jpg); background-size:
100% 100%"></section>
```

Dentro da secção podemos criar um título:

```
<h1>My Portfolio</h1>
```

Podemos também adicionar mais algum texto:

```
<p class=""lead">Welcome!</p>
```

Podemos ver o resultado obtido na Figura 32.

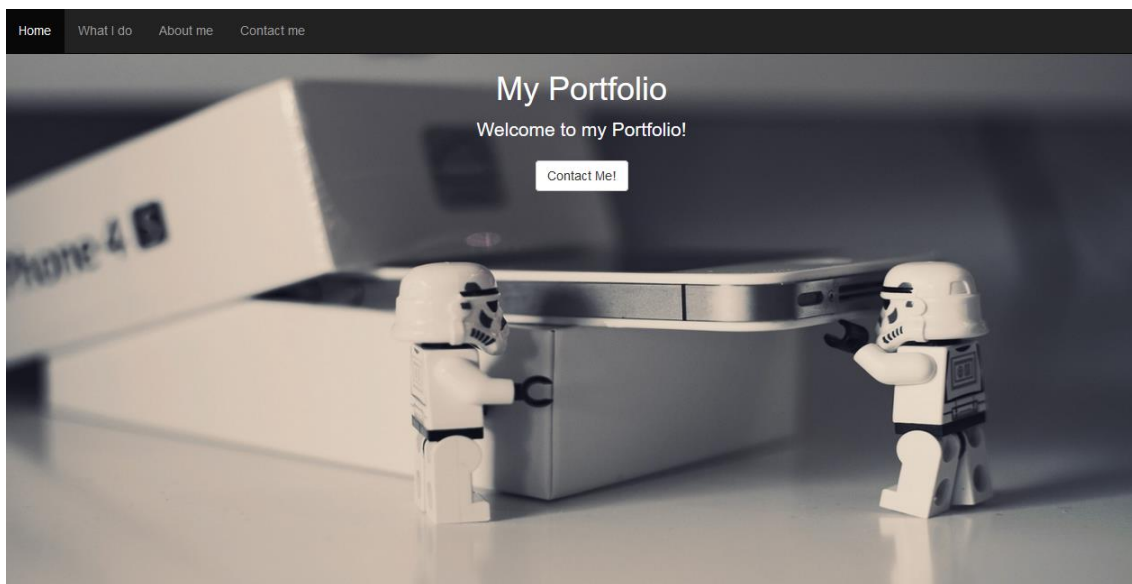


Figura 32- Home Area

## 29.2. *Work Area*

Neste passo iremos criar a nossa segunda secção (Figura 33), denominada de “*Work Area*” Vamos então criar a nossa nova secção:

```
<section id="work" class="container">  
  <div class="page-header">  
    <h1>My Work</h1>  
  </div>  
</section>
```

Para a nossa “*Work Area*” ficar mais bonita podemos adicionar algumas imagens:

```
  
  
  
  

```

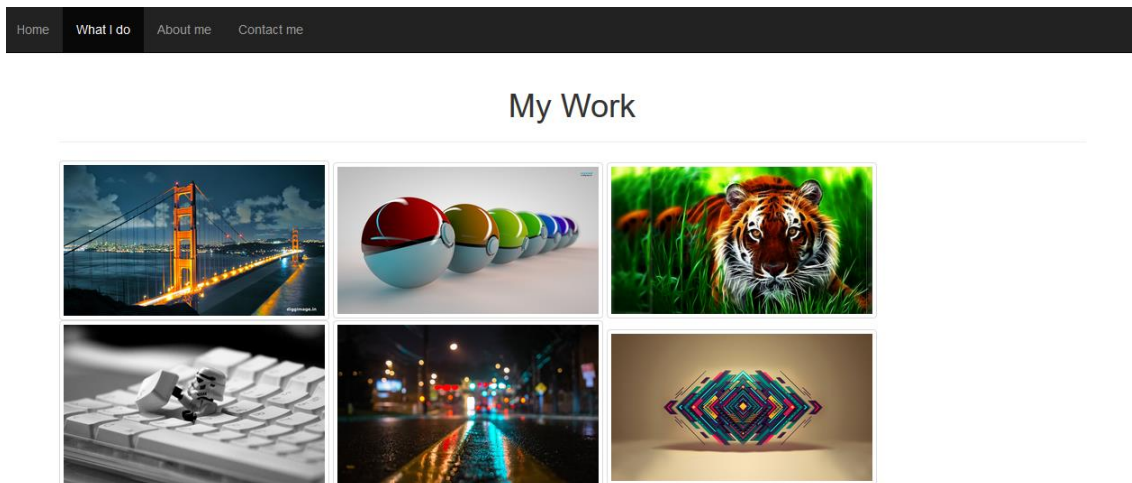


Figura 33- Work Area

### 29.3. About Area

Nesta área podemos então colocar um pequeno texto criado por nos, ou podemos ir a internet ao seguinte *website*: *lipsum.com* e retirar algum texto aleatório.

```
<section id="about" class="container">
```

```
  <div class="col-text">
```

```
    ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation
```

```
  </div>
```

```
</section>
```

Podemos também adicionar um painel com algum texto e/ou imagens:

```
<div class="row">
```

```
  <div class="col-md-6">
```

```
    <div class="panel panel-primary">
```

```
      <div class="panel-heading">
```

```
        <h4>Programming</h4>
```

```
      </div>
```

```
      <div class="panel-body">
```

```
        
```

```
        <p class="lead text-center">I do programming</p>
```

```
      </div>
```

```
    <div class="panel-footer"></div>
```

```
  </div>
```

```
</div>
```

```
</div>
```

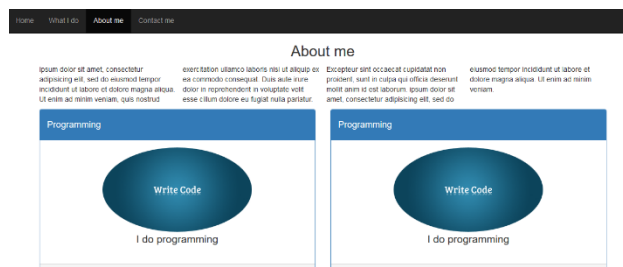


Figura 34- About Area

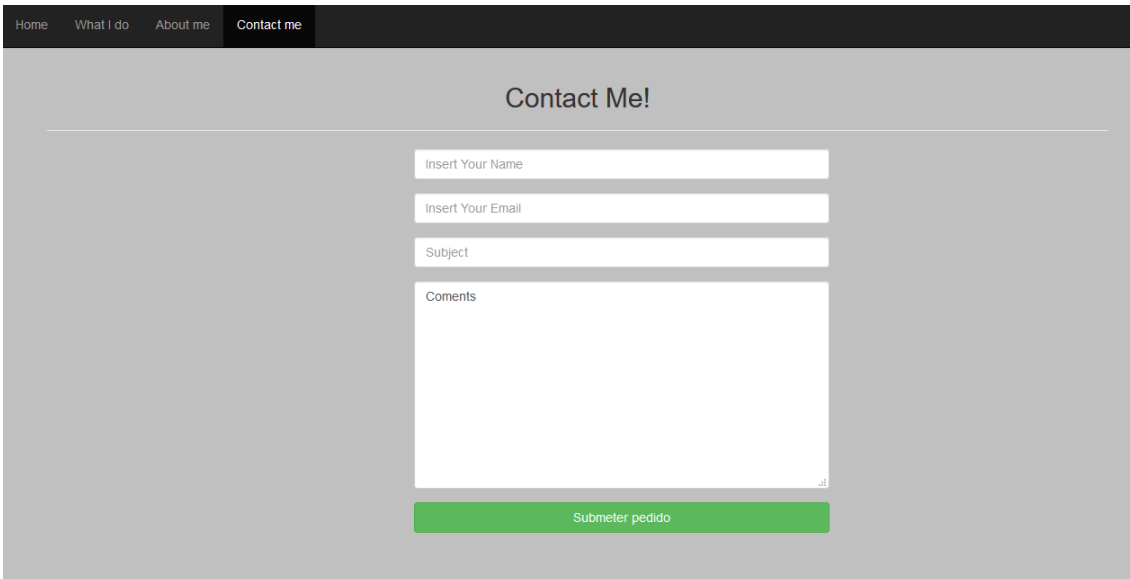
## 29.4. Contact Area

Na nossa *Contact Area* (Figura 35) vamos criar algumas caixas de texto:

```
<form class="col-md-5 col-md-o_set-4">
  <div class="form-group">
    <input class="form-control" placeholder="Insert Your Name" type="text">
  </div>
  <div class="form-group">
    <input class="form-control" placeholder="Insert Your Email" type="email">
  </div>
  <div class="form-group">
    <input class="form-control" placeholder="Subject" type="text">
  </div>
  <div class="form-group">
    <textarea class="form-control" rows="10">Coments</textarea>
  </div>
</form>
```

E por fim vamos criar um botão para submeter:

```
<div class="form-group">
  <input class="btn btn-success btn-block" type="submit">
</div>
```



The screenshot displays a web page with a dark navigation bar at the top containing links for 'Home', 'What I do', 'About me', and 'Contact me'. The main content area has a light gray background and is titled 'Contact Me!'. Below the title, there is a contact form with four input elements: three text boxes labeled 'Insert Your Name', 'Insert Your Email', and 'Subject', and a larger text area labeled 'Coments'. At the bottom of the form is a prominent green button with the text 'Submeter pedido'.

Figura 35- Contact Area

## 30. Galeria de Fotos

### 30.1. Barra de Navegação

Neste novo projeto vamos começar por criar uma *navbar* (Figura 36):

```
<nav class="navbar navbar-inverse">
  <div class="container">
    <div class="navbar-header">
      <div class="navbar-brand">Photo Gallery</div>
    </div>
    <ul class="nav navbar-nav">
      <li><a href="#home">Home</a></li>
      <li><a href="#tech">Technology</a></li>
      <li><a href="#stance">Stance</a></li>
    </ul>
  </div>
</nav>
```

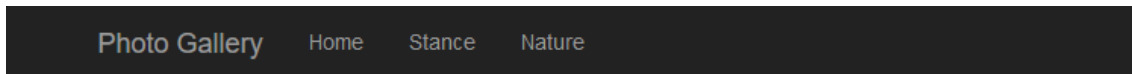


Figura 36- Navbar Galeria Fotos

### 30.2. Carrossel e Slider

Depois de criarmos a nossa *navbar*, vamos então criar um *Carousel Slider* (Figura 37):

```
<div class="container">
  <div class="col-md-10 col-md-o_set-1">
    <div class="carousel slide" data-ride="carousel" id="slider">
      <ol class="carousel-indicators">
        <li data-target="#slider" data-slide-to="0" class="active"></li>
        <li data-target="#slider" data-slide-to="1"></li>
      </ol>
      <div class="carousel-inner">
        <div class="item active">
          
        </div>
        <div class="item">
          
        </div>
      </div>
    </div>
  </div>
</div>
```

Podemos adicionar também os botões para avançar ou voltar para a imagem anterior:

```
<a href="#slider" data-slide="prev" class="left carousel-control">
  <span class="glyphicon glyphicon-chevron-left"></span>
</a>
<a href="#slider" data-slide="next" class="right carousel-control">
  <span class="glyphicon glyphicon-chevron-right"></span>
</a>
```



Figura 37-Carrossel Slider

### 30.3. Thumbnail

Neste passo vamos criar as *thumbnails* (Figura 38) para algumas imagens:

```
</div>
<div class="page-header">
  <h3>Photo Gallery</h3>
</div>
<div class="row">
  <div class="col-md-4">
    <a href="#" class="thumbnail">
      
    </a>
  </div>
  <div class="col-md-4">
    <a href="#" class="thumbnail">
      
    </a>
  </div>
</div>
</div>
</div>
```

Para que todas as imagens fiquem com o tamanho igual, teremos que criar um novo *style*:

```
<style type="text/css">
  .thumbnail img {
    width: 350px;
    height: 200px;
  }
</style>
```

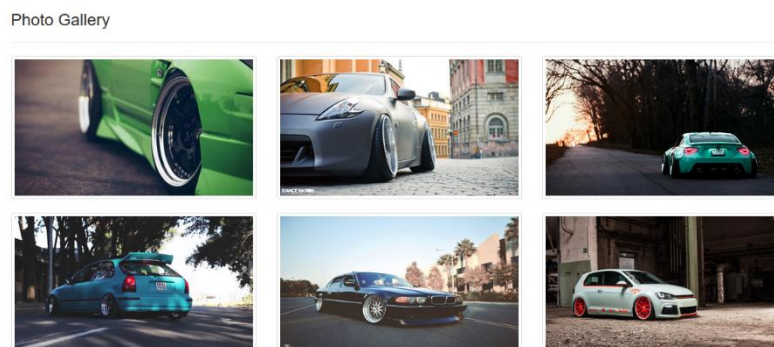


Figura 38- Thumbnails

### 30.4. Rodapé

Podemos criar um rodapé (Figura 59) para o nosso *website*:

```
<footer class="navbar navbar-default navbar-fixed-bottom">  
  <div class="container">  
    <p class="text-center" style="padding: 10px;">Copyrigh Reserved By Rui Pina  
</p>  
  </div>  
</footer>
```

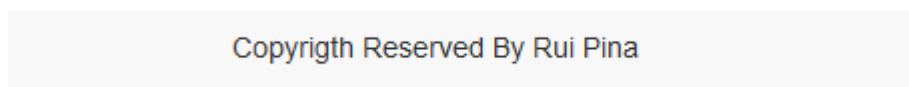


Figura 39- Rodapé

### 30.5. Modal

Para cada uma das imagens colocadas como *thumbnail* vamos criar um modal (Figura 40), para que ao clicar em cada *thumbnail* a imagem abrir em "full size".

```
<div class="modal fade" id="m2">  
  <div class="modal-dialog">  
    <div class="modal-content">  
      <div class="modal-header">Image 2</div>  
      <div class="modal-body">  
          
      </div>  
      <div class="modal-footer">This is the second Image</div>  
    </div>  
  </div>  
</div>
```

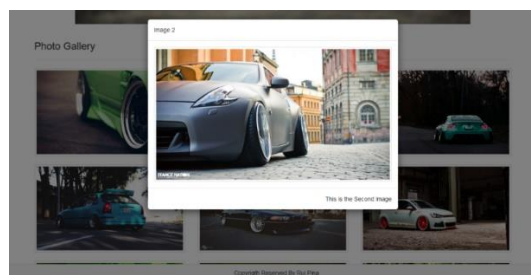


Figura 40- Modal



### 30.6. *Lightbox Plugin*

Neste passo iremos utilizar um novo *plugin*: Lightbox

Para tal teremos que fazer o download do seguinte link:

<http://lokeshdhakar.com/projects/lightbox2/>

A seguir vamos colocar o plugin no nosso html:

```
<link rel="stylesheet" href="css/lightbox.css">
```

```
<script src="js/lightbox.js"></script>
```

Com isto temos o nosso plugin dentro da nossa página.

Por fim, basta associar o lightbox com as nossas imagens:

```
<div class="col-md-4">
```

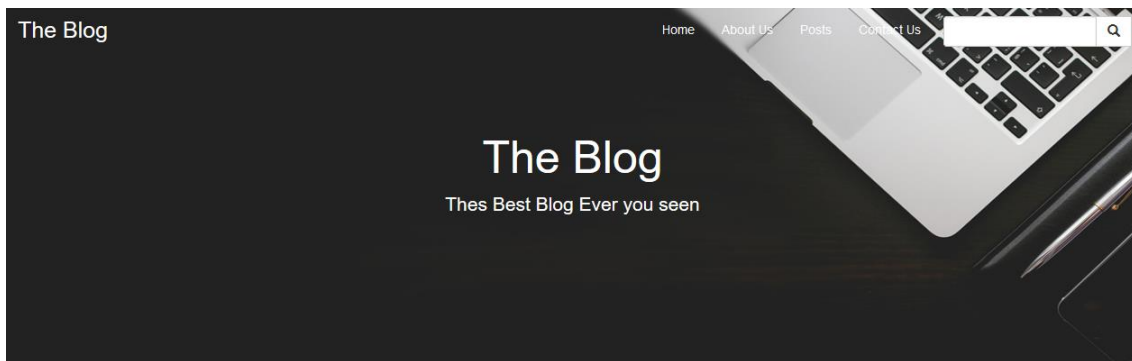
```
  <a href="images/img1.jpg" data-lightbox="gallery" class="thumbnail">
```

```
    
```

```
  </a>
```

```
</div>
```

## 31. Blog



### Water on Mars

A new research has found that There's the Water on Mars

10-March-2017 | Rui Pina

### New Stance Event in LA

A new stance event will happen in LA next week

Figura 41- Blog

Vamos iniciar um novo projeto Blog (Figura 41), para tal iremos criar uma nova pasta com o nome "Blog", e após a pasta estar criada vamos copiar os nossos ficheiros do Bootstrap para dentro dela.

As pastas a ser copiadas são: *css*; *fonts*; *js*;

Por fim iremos criar o nosso `index.html` com o seguinte código:

```
<!DOCTYPE html>
<html>
  <head>
    <title>My Blog</title>
    <link rel="stylesheet" href="css/bootstrap.css">
    <script src="js/jquery.js"></script>
    <script src="js/bootstrap.js"></script>
  </head>
  <body>
  </body>
</html>
```

### 31.1. Barra de Navegação personalizada

Após iniciar este novo projeto vamos então criar uma barra de navegação customizada (Figura 42):

```
<nav class="navbar">
  <div class="container-uid">
    <div class="navbar-header">
      <a href="#" class="navbar-brand">Blog</a>
    </div>
    <ul class="nav navbar-nav">
      <li><a href="#">Home</a></li>
      <li><a href="#">About Us</a></li>
      <li><a href="#">Contact Us</a></li>
    </ul>
  </div>
</nav>
```

Para personalizar a nossa barra de navegação iremos criar um novo ficheiro “*style.css*” com o seguinte:

```
.navbar a{
  color:white;
}
.nav li a:hover{
  background: orange;
}
```

Depois de criado o ficheiro “*index.css*”, vamos aplica-lo no nosso “*index.html*”:

```
<link rel="stylesheet" href="css/style.css">
```

Por fim iremos adicionar uma imagem á nossa barra de navegação:

```
<header style="background: url(images/header.jpg)">
  <div class="text-center">
    <h1 class="text-center">The Blog</h1>
    <div class="lead">Thes Best Blog Ever you seen</div>
  </div>
</header>
```

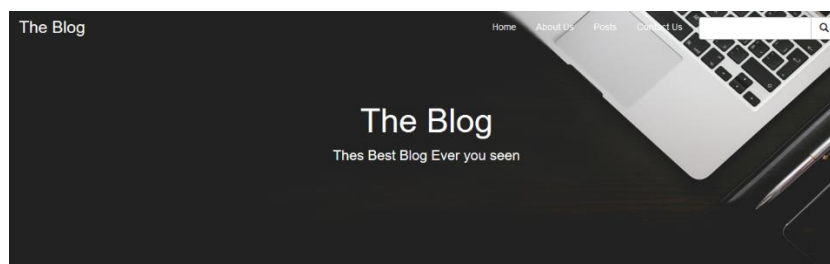


Figura 42- Navbar Personalizada

### 31.2. Barra de Pesquisa

Para criar a nossa Barra de Pesquisa (Figura 43) vamos utilizar o seguinte código:

```
<div class="nav navbar-nav form-inline navbar-right" style="padding: 10px">
  <div class="input-group">
    <input type="text" class="form-control">
    <div class="input-group-btn">
      <button class="btn btn-default"><i class="glyphicon glyphicon-search"></i></button>
    </div>
  </div>
</div>
```



Figura 43- Barra de Pesquisa

### 31.3. Área de Posts

Para o nosso blog ficar mais completo iremos criar uma área de *posts* (Figura 44):

```
<div class="row">
  <div class="col-md-8 col-md-o_set-2">
    <div id="posts">
      <h1><a href="#">New Stance Event in LA</a></h1>
      <p class="lead">A new stance event will happen in LA next week</p>
    </div>
    <div id="posts">
      <h1><a href="#">Technology</a></h1>
      <p class="lead">Technology is moving to its final round</p>
    </div>
  </div>
</div>
```

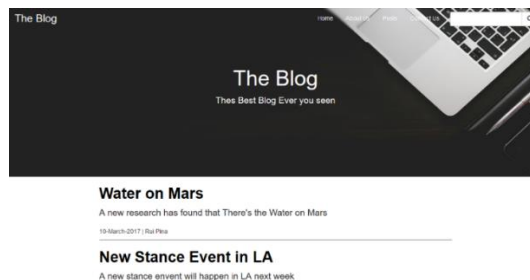


Figura 44- Area de Posts

### 31.4. Rodapé

```
<div class="container-uid">  
<hr>  
<ul class="nav navbar-nav nav-center">  
  <li><a href="#">Home</a></li>  
  <li><a href="#">Technology</a></li>  
  <li><a href="#">Ciencia</a></li>  
</ul>  
</div>
```

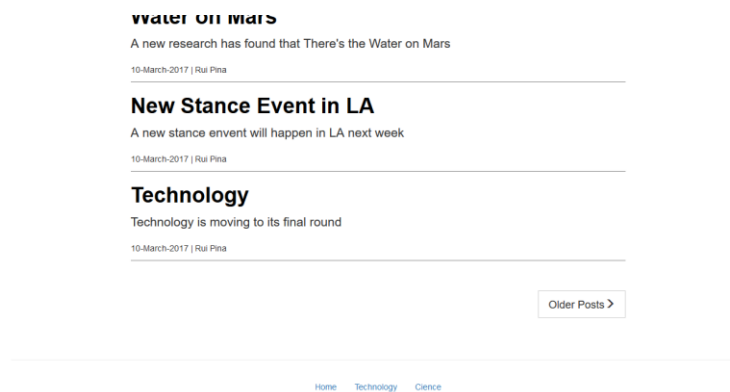


Figura 45- Rodapé Blog

### 31.5. Página de Posts

Inicialmente será duplicado o nosso “*index.html*” e será alterado o nome do duplicado para “*posts.html*”, após a criação do “*posts.html*” vamos alterar o id do *posts* na nossa *navbar*. Por fim vamos alterar o texto que esta depois do titulo e adicionar alguma imagem acerca do *post*. Podemos também passar a data para a linha a seguir ao título para ficar com melhor aspeto.



Figura 46- Página de Posts

## 32. Website Comercial

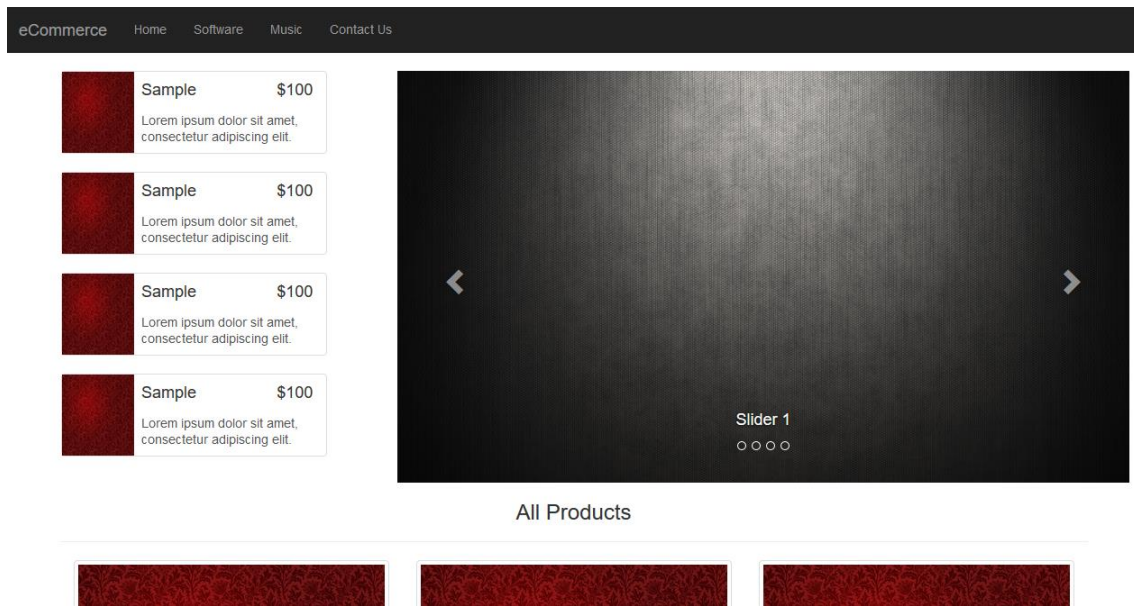


Figura 47- Site Comercial

### 32.1. Barra de Navegação

Depois de o nosso novo projeto estar criado, vamos criar uma *navbar*:

```
<nav class="navbar navbar-inverse">
  <div class="container-uid">
    <div class="navbar-header">
      <a class="navbar-brand" href="#">eCommerce</a>
    </div>
    <ul class="nav navbar-nav">
      <li><a href="#">Home</a></li>
      <li><a href="#">Software</a></li>
      <li><a href="#">Music </a></li>
      <li><a href="#">Contac Us</a></li>
    </ul>
  </div>
</nav>
```

## 32.2. Carrossel

Após a *navbar* estar concluída vamos seguir para o próximo passo, que é o carrossel (Figura 48):

```
<header class="col-md-8">
  <div class="carousel slide" data-ride="carousel" id="c1">
    <ol class="carousel-indicators">
      <li data-target="#c1" data-slide-to="0"></li>
      <li data-target="#c1" data-slide-to="1"></li>
    </ol>
    <div class="carousel-inner">
      <div class="item active">
        
        <div class="carousel-caption">
          <h4>Slider 1</h4>
        </div>
      </div>
      <div class="item">
        
        <div class="carousel-caption">
          <h4>Slider 2</h4>
        </div>
      </div>
    </div>
    <a href="#c1" class="left carousel-control" data-slide="prev">
      <span class="glyphicon glyphicon-chevron-left"></span>
    </a>
    <a href="#c1" class="right carousel-control" data-slide="next">
      <span class="glyphicon glyphicon-chevron-right"></span>
    </a>
  </div>
</header>
```

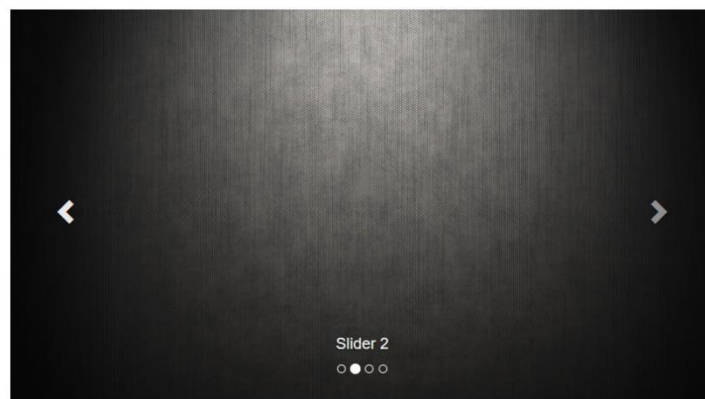


Figura 48- Carrossel

### 32.3. *Aside*

Neste passo vamos criar o *aside* (Figura 49):

```
<aside class="col-md-4">
  <div class="col-md-10 col-md-o_set-1">
    <div class="list-group">
      <a href="#" class="list-group-item" style="padding:0; margin: 0;">
        <div class="col-md-3" style="padding: 0; margin: 0;"><img src=
"images/img1.jpg" style="width: 80px; height:90px;"></div>
        <div class="col-md-9">
          <div class="list-group-item-heading">
            <h4 class="pull-right">$100</h4>
            <h4 class="pull-left">Sample</h4>
            <div class="clear-fix"></div>
          </div>
          <p class="list-group-item-text">Lorem ipsum dolor sit amet, consectetur
adipiscing elit. </p>
        </div>
      </a>
    </div>
  </div>
</aside>
```

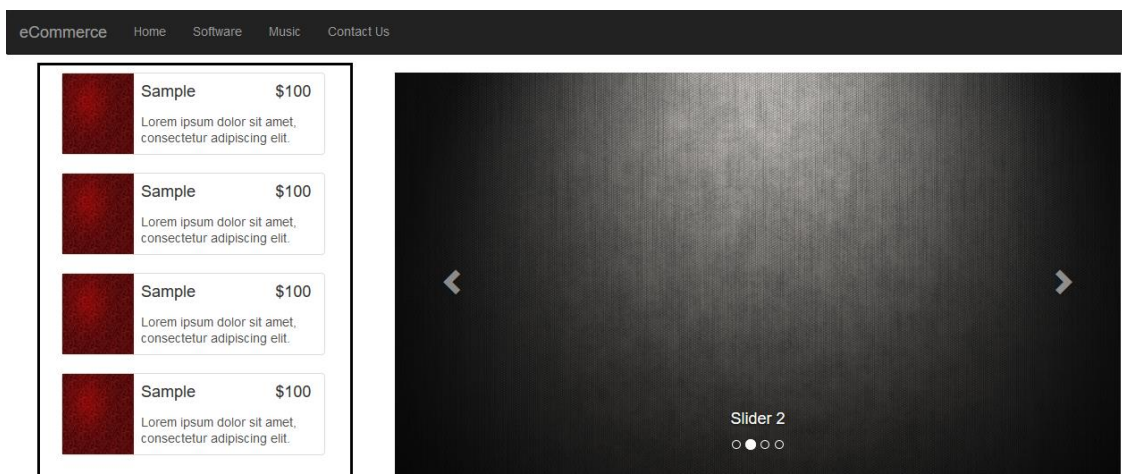


Figura 49- Aside



## 32.4. Área de Produtos

Depois de criada a nossa *aside*, vamos passar para o passo seguinte:

```
<div class="container">
  <div class="col-md-4">
    <div class="thumbnail ">
      
      <div class="caption">
        <h4 class="pull-right">$100</h4>
        <h4><a href="#">Sample Product</a></h4>
        <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod
tempor incididunt ut labore et dolore magna aliqua.Lorem ipsum. </p>
      </div>
    </div>
  </div>
</div>
```

Para criar vários produtos, basta duplicar este código. O resultado obtido é o seguinte (Figura 50):

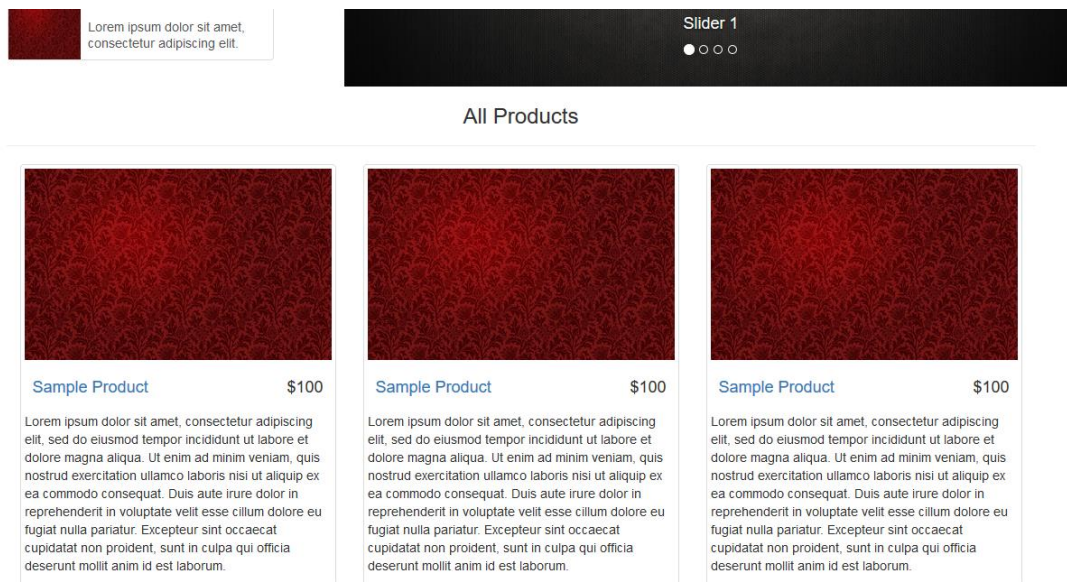


Figura 50- Área de Produtos

### 32.5. Página de Produtos

Para criar a página dos produtos teremos que criar uma nova página html, para isso podemos duplicar o nosso index.html e fazer as nossas alterações.

Em primeiro lugar teremos que alterar o id dos nossos produtos para product.html, que foi o nome dado a nossa nova página html.

Depois de alterar todos os *id's*, vamos eliminar o nosso *slider*, que já não é necessário nesta página.

Vamos também eliminar os "*All Products*" e criar uma nova `<header></header>`, com apenas um produto.

Para este produto podemos adicionar um botão para comentários, e outro para efetuar a compra.

#### Water on Mars

10-March-2017 | Rui Pina

"At vero eos et accusamus et iusto odio dignissimos ducimus qui blanditiis praesentium voluptatum deleniti atque corrupti quos dolores et quas molestias excepturi sint occaecati cupiditate non provident, similique sunt in culpa qui officia deserunt mollitia animi, id est laborum et dolorum fuga. Et harum quidem rerum facilis est et expedita distinctio. Nam libero tempore, cum soluta nobis est eligendi optio cumque nihil impedit quo minus id quod maxime placeat facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum hic tenetur a sapiente delectus, ut aut reiciendis voluptatibus maiores alias consequatur aut perferendis doloribus asperiores repellat."



Figura 51- Página de Produtos

### 32.6. Modals

Para finalizar iremos criar o botão para efetuar a compra.

Depois de ter o botão criado só falta criar o modal:

```
<div class="modal fade" id="buy modal">
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <button class="close" data-dismiss="modal">&times;</button>
        <h4 class="modal-title">Buy the Product</h4>
      </div>
      <div class="modal-body">
        <form>
          <div class="form-group">
            <input type="text" class="form-control input-lg" placeholder="Insert
your Name">
          </div>
          <div class="form-group">
            <input type="email" class="form-control input-lg" placeholder="Insert
your Email">
          </div>
          <div class="form-group">
            <input type="text" class="form-control input-lg" placeholder="Insert
your Mobile No">
          </div>
          <div class="form-group">
            <textarea class="form-control" rows="5" style="font-size:20px;
color:grey;">Address..</textarea>
          </div>
        </form>
      </div>
      <div class="modal-footer">
        <button class="btn btn-danger" data-dismiss="modal">Submit the
Order</button>
      </div>
    </div>
  </div>
</div>
```

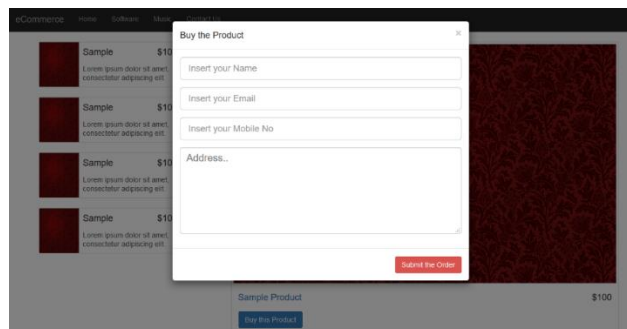


Figura 52- Modal

## Anexo B

### Tutorial de LaTeX

Neste tutorial iremos aprender a utilizar o latex.

O latex é melhor do que o word em varias coisas, como por exemplo, ser mais rápido a responder, pode também ser compatível com Windows, Mac OS e Linux.

Para utilizar este programa é necessário instalar o texmaker.

A maior diferença do latex para o word é que o latex utiliza código para escrever todo o texto.

#### 1. Secções e Margens

Cada secção contém um título, e dentro de cada secção escrevemos tudo o que queremos que apareça no ecrã, como por exemplo: texto, imagens...

Para criar uma nova secção basta utilizar o seguinte código:

```
\section{titulo}
```

Depois da secção criada podemos escrever qualquer coisa dentro dela, o resultado será o seguinte (Figura 1):

#### 1 Introduction

Neste tutorial iremos aprender a utilizar o latex.

O latex é melhor do que o word em varias coisas, como por exemplo, ser mais rápido a responder, pode também ser compatível com Windows, Mac OS e Linux. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Figura 1- Exemplo Secção

Para alterar as margens do texto vamos utilizar um `\usepackage{geometry}`, onde podemos definir as margens do texto.

Esse `\usepackage{geometry}` vai ser inserido no inicio no nosso documento depois da nosso class:

```
\documentclass{article}
```

Podemos ter as duas margens iguais ou podemos ter uma maior que a outra.

Ex:

```
\usepackage[margin=1in,includefoot]{geometry}
```

Para que uma das margens seja diferente basta colocar, por exemplo: *margin-left=1.5in*.

## 2. Cabeçalho e Rodapé

Podemos alterar o cabeçalho e rodapé á nossa maneira, para tal iremos importar um *usepackage*, que é o *fancyhdr* e alterar o estilo:

```
\usepackage{fancyhdr} \pagestyle{fancy}
```

Depois do package estar no nosso ficheiro, vamos eliminar o cabeçalho e o rodapé antigo:

```
\fancyhead{}
```

```
\fancyfoot{}
```

Com o cabeçalho e o rodapé eliminados vamos ter que criar um novo:

```
\renewcommand{\headrulewidth} {0pt}
```

```
\renewcommand{\footrulewidth} {0pt}
```

Para que o número da página apareça no fim utilizamos o seguinte código:

```
\fancyfoot[R]{\thepage}
```

## 3. Subsecção

Para criar uma subsecção, é necessário criar uma secção com o que queremos escrever nela e depois criar a nossa: *\subsection*

Ex:

"A introdução está na página *\pagerefsec:intro*."

## 4. Índice

Para a criarmos o nosso índice temos de colocar o seguinte código depois da "*\titlepage*":

Cria o índice: *\tableofcontents*

Estilo: *\thispagestyleempty*

Faz com que a página em questão não contenha mais nada: *\cleardoublepage*

## 5. Imagens

Para poder adicionar imagens no nosso texto é necessário fazer vários *imports*:

```
\usepackage{graphicx}
```

Este package permite importar as imagens.

```
\usepackage{float}
```

Este package permite controlar o *float*.

Para adicionar uma imagem vamos utilizar o seguinte código:

```
\begin{Figure}[H]
```

```
\centering
```

```
\includegraphics[height=3in]{C:/Users/Estagiario/Desktop/texMaker/images/img.jpg}
```

```
\caption[Option caption]{Real, local caption}
```

```
\label{fig:img}
```

```
\end{figure}
```

```
Figure \ref{fig:img} Stance
```

O resultado obtido é o seguinte (Figura 2):



Figura 2- Imagem

## 6. Índice de Imagens

Para criar um Índice para as Imagens:

```
\listoffigures
```

```
\addcontentsline{toc}{section}{\numberline{}List of Figures}
```

```
\cleardoublepage
```

Este código será introduzido depois do `\end{titlepage}`

## 7. Tabelas

Para criar uma tabela podemos utilizar o código do exemplo anterior:

```
\begin{table}[H]
  \centering
  \label{tab:Meetings}
  \caption[This is optional caption, without reference]{Stance Events}
  \begin{tabular}{l c r} \bfseries{Date} & \bfseries{Local} & \bfseries{Time} \\ \hline
  April, 20 & LA & Day \\
  June, 15 & Chicago & Day \\
  December, 5 & Boston & Night \\
\end{tabular}
\end{table}
```

Para criar um Índice para as Tabelas:

```
\listoftables
\addcontentsline{toc}{section}{\numberline{}List of Tables}
\cleardoublepage
```

## 8. Referencias

Podemos adicionar referências para os nossos índices, permitindo assim clicar no nome de uma das secções que está no índice e automaticamente redirecionar-nos para a página pretendida. Para tal é necessário fazer o seguinte *import*:

```
\usepackage[hidelinks]{hyperref}
```