



Licenciatura em Gestão de Sistemas e Computação

***Trabalho sobre Aplicativo Web JobRadar***

Projeto Final de Licenciatura

Elaborado por João Martins Lima

Aluno nº 20141862

Orientador: Professor Doutor Mário Macedo

Barcarena

dezembro 2020



Atlântica Instituto Universitário

Licenciatura em Gestão de Sistemas e Computação

***Trabalho sobre Aplicativo Web JobRadar***

Projeto Final de Licenciatura

Elaborado por João Martins Lima

Aluno nº 20141862

Orientador: Professor Doutor Mário Macedo

Barcarena

dezembro de 2020

O autor é o único responsável pelas ideias expressas neste relatório

## **Agradecimentos**

Começo por agradecer ao Professor Alexandre Barão que, embora já não lecionasse esta cadeira na Atlântica Instituto Universitário, acompanhou e supervisionou o meu trabalho ao longo do tempo. Pela constante disponibilidade, orientação e profissionalismo. Pelo seu tamanho conhecimento e entusiasmo por esta área que são contagiantes.

Nunca será também demais agradecer ao Professor Mário Macedo que abraçou este meu projeto com toda a disponibilidade e atenção. Dando-me a orientação necessária para o finalizar.

Um obrigado especial à minha namorada pela constante ajuda e apoio incondicional.

Aos meus pais que antes de me verem como filho, vêem-me como cidadão e transmitiram-me sempre os valores e os ideais para conseguir alcançar os meus objetivos.

E, por fim, mas não menos importante, um muito obrigado a todo o grupo docente da Licenciatura de Gestão de Sistemas e Computação da Atlântica Instituto Universitário que contribuíram para a minha evolução profissional.

Com este projeto, fica assim concluída mais uma etapa da minha vida.

## **Resumo**

Este documento foi desenvolvido no âmbito do Projeto Final de licenciatura intitulado *Aplicação Web JobRadar*.

Devido à crescente procura de emprego, falta de qualidade das ofertas e carência de uma *feature* centrada na procura de emprego através da localização geográfica aliada ao facto da existência de um défice, também ele crescente, na comunicação entre candidatos e recrutadores, surgiu a necessidade de desenvolver novas funcionalidades de apoio a candidatos à procura de emprego. Esta aplicação para web está focada na promoção e procura de anúncios de emprego tendo em conta a localização do candidato/ empresa, reduzindo assim a complexidade na procura de emprego e a dificuldade de comunicação entre empresas e candidatos, mantendo também toda a informação e processo centralizado e organizado.

A aplicação Job Radar foi desenvolvida com recurso a base de dados, num modelo cliente-servidor.

Palavras-chave: *JobRadar*, *Job*, Emprego, Ofertas de Emprego, Candidatos a Emprego, Orientação Profissional.

## **Abstract**

### **JobRadar Web Application**

This document was developed within the scope of the Final Degree Project entitled JobRadar Web Application.

Candidacy to the growing job search, lack of quality of offers and lack of a feature focused on job search through geographic location coupled with the fact that there is a deficit, also growing, in the communication between candidates and recruiters, the need arose to develop new qualities of support for job seekers. This web application is focused on the promotion and search of job vacancies taking into account the location of the candidate / company, as well as the complexity of the job search and the difficulty of communication between companies and candidates, also maintaining all the information and centralized process and organized.

The Job Radar application was developed using a database, in a client-server model.

Keywords: JobRadar, Job, Employment, Job Offers, Jobseekers, Career Counseling.

## Índice

Agradecimentos .....	iii
Resumo .....	iv
Abstract.....	v
Índice .....	vi
Índice de figuras.....	viii
Índice de tabelas.....	x
Lista de abreviaturas e siglas .....	xi
1.Introdução .....	1
1.1. Contexto e Motivação.....	1
1.2. Definição do Problema .....	3
1.3. Objetivos de Investigação.....	4
1.4. Método de Investigação .....	5
1.5. Resultados Alcançados .....	7
1.6. Estrutura do Documento .....	7
2. Revisão da Literatura .....	8
2.1. Introdução .....	8
2.2. Questões Éticas do Processo de Investigação.....	9
2.3. Especificação de Requisitos.....	15
2.4. Modelo de Segurança.....	18
2.5. Arquitetura Cliente-Servidor .....	20
3. Especificação de Requisitos.....	24
3.1. Visão Geral .....	24
3.2. Requisitos Funcionais .....	25



3.3. Requisitos Não Funcionais .....	29
4. Arquitetura do Sistema .....	31
4.1. Casos de Uso.....	33
4.2. Diagrama de Atividades.....	37
4.3. Diagrama de Classes .....	39
4.4. Modelo de Domínio .....	40
4.5. Modelos básicos de Base de dados relacional .....	41
4.6. Estrutura da Base de Dados .....	44
4.7. Base de dados JobRadar (MER) .....	46
5. Protótipo do Sistema .....	51
5.1. Introdução .....	51
5.2. Descrição Geral.....	51
5.2.1 Servidor.....	51
5.2.2 Administrador de base de dados .....	52
5.2.3 Front-end e Back-end.....	53
5.2.4 Mockup .....	54
5.2.5 Esquema de Navegação .....	56
6. Interface com o Utilizador .....	58
7. Conclusão.....	84
Bibliografia .....	88

## Índice de figuras

Figura 1 – Modelo de Segurança .....	18
Figura 2 – Modelo Cliente Servidor .....	22
Figura 3 – Canal de Distribuição (Ligação Cliente-Servidor).....	23
Figura 4 – Arquitetura do Sistema JobRadar.....	33
Figura 5 - Atores do sistema JobRadar e principais casos de Uso .....	36
Figura 6 – Diagrama de Atividades JobRadar .....	38
Figura 7 – Classes (Candidato / Candidaturas).....	39
Figura 8 – Diagrama de Classes (Candidato / Candidaturas).....	39
Figura 9 – Modelo de Domínio JobRadar .....	40
Figura 10 – Constituição de Tabelas Relacionais.....	42
Figura 11 – Estrutura Base de Dados JobRadar .....	45
Figura 12 – Modelo Entidade e Relação JobRadar.....	46
Figura 13 – Diretória local Apache - JobRadar .....	51
Figura 14 – Php My Admin JobRadar .....	52
Figura 15 – Esquema Back End e Front End JobRadar.....	53
Figura 17 – Esquema de Navegação JobRadar.....	56
Figura 18 – Página Index JobRadar.....	58
Figura 19 – Página de UserIn JobRadar .....	59
Figura 20 – LogIn1 .....	59
Figura 21 – LogIn2 .....	59
Figura 22 – Registo Empresa.....	60
Figura 23 – Página Candidato JobRadar.....	61
Figura 24 – Motor de Pesquisa - Candidato JobRadar .....	62

Figura 25 – Secção Mapa - Candidato JobRadar.....	63
Figura 26 – Seção Entrevistas - Candidato JobRadar.....	64
Figura 27 – Features Futuras Candidato JobRadar.....	65
Figura 28 – Feedback Candidato JobRadar .....	65
Figura 29 – Favoritos Candidato JobRadar .....	66
Figura 30 – Mercado Candidato JobRadar .....	67
Figura 31 – Dicas Candidato JobRadar .....	68
Figura 32 – Seção Perfil - Candidato JobRadar.....	69
Figura 33 - Dados Candidato JobRadar.....	70
Figura 34 – Experiência Candidato JobRadar .....	71
Figura 35 – Áreas de Interesse Candidato JobRadar .....	72
Figura 36 – Candidaturas Pendentes Candidato JobRadar.....	73
Figura 37 - CV Candidato JobRadar.....	74
Figura 38 – Trabalho de Sonho Candidato JobRadar.....	75
Figura 39 – Contacte-nos Candidato JobRadar .....	76
Figura 40 - Página Empresa JobRadar.....	77
Figura 41 - Perfil Empresa JobRadar.....	78
Figura 42 - Features Futuras Empresa JobRadar .....	79
Figura 43 - Feedback Empresa JobRadar .....	80
Figura 44 – Dados Empresa Empresa JobRadar.....	81
Figura 45 - Contacte-nos Empresa JobRadar .....	82
Figura 46 - Contacte-nos Empresa JobRadar .....	83

## Índice de tabelas

Tabela 1 – Requisito Funcional Utilizador Anónimo (RFUA).....	25
Tabela 2 – Requisito Funcional Utilizador Autenticado Candidato (RFUAC).....	26
Tabela 3 – Requisito Funcional Utilizador Autenticado Empresa (RFUAE).....	27
Tabela 4 – Requisito Funcional Administrador (RFA) .....	28
Tabela 5 – Requisito Funcional Sistema (RFS).....	28
Tabela 6 – Requisitos Não Funcionais (RNF).....	30
Tabela 7 – Intervenientes do Sistema JobRadar .....	35
Tabela 8 – Casos de Uso JobRadar.....	35
Tabela 9 – Tabela da Base de dados Applyer.....	47
Tabela 10 - Tabela da Base de dados Provider.....	47
Tabela 11 - Tabela da Base de dados Jobs.....	48
Tabela 12 - Tabela da Base de dados Announce .....	48
Tabela 13 - Tabela da Base de dados AnnounceFavs.....	49
Tabela 14 - Tabela da Base de dados Categories.....	49
Tabela 15 - Tabela da Base de dados Applyercategories .....	49
Tabela 16 - Tabela da Base de dados Attachments .....	49
Tabela 17- Tabela da Base de dados Applications .....	50

## **Lista de abreviaturas e siglas**

API – Application Programming Interface  
CSS – Cascading Style Sheets  
CTA – Call to action  
DBA – Administrador de Base de Dados  
DER – Diagrama de Entidade e Relação  
DRP – Disaster Recovery Plan  
DSR – Design-Science Research  
FTP – File Transfer Protocol  
GPS – Global Positioning System  
HTML – Hyper Text Markup Language  
HTTP – Hyper Text Transfer Protocol  
HTTPS – Hyper Text Transfer Protocol Secure  
ID – Identity Document  
MER – Modelo Entidade Relação  
PC – Personal Computer  
PSI – Políticas de Segurança Informação  
PHP – Personal Home Page  
RG – Research goal  
SMTP – Simple Mail Transfer Protocol  
SGBD – Sistema de Gestão de Base de Dados  
SI – Sistema de Informação  
SQL – Structured Query Language  
TI – Tecnologia de Informação  
UML – Unified Modeling Language  
URL – Uniform Resource Locator

## **1.Introdução**

### **1.1. Contexto e Motivação**

O mercado de trabalho ocupa uma posição fulcral na maneira como as sociedades funcionam. A privação de emprego é um problema que afeta, sobretudo, a camada jovem adulta que procura o primeiro emprego ou uma nova oportunidade de emprego.

#### **A experiência**

É um ciclo vicioso com uma linha demasiado ténue. As empresas querem trabalhadores com experiência e os candidatos a emprego necessitam de experiência para poderem ser selecionados e, posteriormente, contratados. É legítimo que assim o seja porque o candidato ao ter experiência profissional, obrigatoriamente, tende a saber comportar-se consoante o ambiente em que está inserido. Contudo, o candidato ao não ter experiência profissional não significa que não esteja capacitado para o desafio a que se compromete.

A camada jovem portuguesa à procura do seu primeiro emprego tem também a oportunidade de estar inserida nos programas do IEFP- Instituto de Emprego e Formação Profissional para o incentivo ao emprego, contudo este órgão estatal não tem capacidade de segmentar os anúncios nem tão pouco os candidatos devido à sua quantidade.

#### **O trabalho precário**

São os trabalhos a tempo parcial ou part-time, como são conhecidos no mercado de trabalho, que configuram o maior modelo de trabalho precário em Portugal e com efeitos nefastos.

De acordo com os dados divulgados na comunicação social portuguesa no início do ano de 2019<sup>1</sup>, Portugal tem mais de 130 mil contratos instáveis, de curta duração e com

---

<sup>1</sup> <https://www.dinheirovivo.pt/economia/portugal-e-o-segundo-pais-onde-o-trabalho-precario-mais-subiu-12782109.html>

salários baixos. Comparativamente com a Europa, só a Croácia atingiu um número superior de trabalhadores precários. Os setores da restauração, construção civil e turismo representam os principais contratos a termo e de salários reduzidos.

### **Pós Pandemia**

O Mundo passou por uma revolução, e continua a passar, depois do aparecimento do vírus COVID-19. Foi-nos apresentada uma nova dinâmica que acabou por alterar os nossos hábitos e as nossas rotinas. As soft skills passaram assim a ter um peso ainda maior para os departamentos de recursos humanos das organizações. As soft skills são características comportamentais que qualquer candidato deve conter para assumir uma função numa determinada empresa. Uma vez que as medidas para a prevenção do vírus se centraram no isolamento social muitas empresas viram-se, literalmente, obrigadas a assumir este novo modelo de trabalho. Após tantas mudanças, algumas soft skills ganharam uma notoriedade decisiva. Inteligência emocional e gestão de tempo foram algumas dessas skills selecionadas.

Já pouco importa em que empresa trabalhamos efetivamente. Hoje estamos no escritório, amanhã em homeoffice, depois de amanhã noutra qualquer. Hoje também já ninguém prescinde de ferramentas tecnológicas para encontrar emprego e comunicar com as empresas. As aplicações de emprego rapidamente ocuparam um espaço virtual extremamente chamativo e comunicativo. O JobRadar trará benefícios imediatos para a população e para a disseminação de empregos.

## **1.2. Definição do Problema**

Verificam-se cada vez mais lacunas nos sites de emprego online, nomeadamente na procura e nos processos de recrutamento. O problema identificado em sites como o indeed, netempregos e saop empregos consiste em não existir um acompanhamento por parte do site desde que o candidato submete o seu CV, até ao momento em que recebe a resposta por parte da empresa, quando a recebe.

O único propósito dos sites acima referidos é disponibilizarem gratuitamente anúncios de emprego, ainda que, muitas vezes, obriguem também o utilizador a entrar em outros websites para se conseguir candidatar à vaga que pretende que, em inúmeras vezes, já não se encontra ativa.

Outro problema identificado consiste no facto dos sites não acompanharem o dinamismo que hoje as novas tecnologias têm para oferecer, como por exemplo, a aplicação GoogleMaps. Utilizando o GoogleMaps seria possível procurarmos vagas de emprego sem recorrermos a filtros de pesquisa e usufruindo apenas das funcionalidades que esta aplicação nos disponibiliza. O GoogleMaps permitia assim aos candidatos procurarem o seu emprego através de um mapa real indo de encontro imediato às áreas periféricas à da sua residência. Não há nenhuma solução que tenha os requisitos e funcionalidades equivalentes às do aplicativo JobRadar e que esteja implementada no mercado português.



### 1.3. Objetivos de Investigação

Tendo em conta o contexto supracitado, definiram-se os seguintes objetivos de investigação (RG – *Research Goals*) de modo a resolver a problemática de investigação em cima referida:

RG 1- Realizar a revisão de literatura para a compreensão dos aspetos técnicos inerentes à realização da aplicação web “JobRadar” (por exemplo, conceito *Responsive Web Design*, *Open Source*, Modelo Cliente- Servidor, Modelo de Base de Dados Relacional);

RG 2- Elaborar a especificação de requisitos relativos à aplicação “JobRadar”;

RG 3- Definir a Arquitetura do Sistema;

RG 4- Elaboração do diagrama de atividades;

RG 5- Definir o Modelo de Base de Dados;

RG 6- Testar e validar todas as tecnologias de modo a suportarem o protótipo aplicacional “JobRadar”;

RG 7- Elaboração de *mockup* do futuro protótipo ‘JobRadar’ e esquema de navegação;

RG8- Desenvolver a interface e funcionalidades de acordo com o *mockup* e esquema de navegação desenvolvido;

RG 9- Desenvolver a *feature* Google Maps em conformidade como pretendido.

RG 10- Testar funcionalidades do sistema.

#### **1.4. Método de Investigação**

A metodologia de investigação selecionada, de acordo com as problemáticas supracitadas, é o método *Design Science Research* (DSR). O DSR é um conjunto de métodos de estudo, processos e técnicas selecionadas com a finalidade de criar uma pesquisa de desenvolvimento para um trabalho científico.

Esta metodologia foi criada no âmbito da resolução de problemas atuando com o objetivo de criar inovações que definem ideias, práticas, capacidades técnicas ou produtos que sejam eficazes através da análise do projeto, implementação e em conjunto com a gestão e o uso das tecnologias de informação. (Hevner, Ram, March, & Park, 2004)

As características de um Sistema de Informação (SI) englobam sistemas de trabalho e as suas metodologias envolvidas. Contudo, os fatores que têm maior destaque são os métodos de recolha de dados, os processos de análise e de melhorias, com o objetivo de melhorarem a eficiência dos processos já existentes. (Silver, Markus, & Beath, 1995)

Segundo (Hevner, Ram, March, & Park, 2004), existem sete diretrizes no DSR com o fim de ajudar os investigadores a compreender os requisitos da metodologia. As diretrizes enunciadas no DSR são a avaliação do projeto, a relevância do problema, o rigor, os objetivos, contribuições da pesquisa e o design como um artefacto (seja ele um relatório, uma análise ou até um modelo estrutural). Todas estas diretrizes resultam, por fim, no design como um processo de pesquisa.

O ciclo de pesquisa é um processo muito semelhante ao ciclo de vida e também semelhante ao desenvolvimento de um software, tornando-se assim numa metodologia muito utilizada na realização de pesquisas relacionadas com sistemas de informação.

As Tecnologias de Informação (TI) estão em constante modificação e evolução e, por isso, os conhecimentos técnicos têm de os acompanhar.

Segundo (March & Smith, 1995), foram identificados dois processos de design relevantes que englobam os quatro artefactos de design produzidos pelo *Design Science Research* (DSR). As elaborações, os modelos, os métodos e as instâncias utilizadas. Os artefactos informáticos são definidos como modelos (algoritmos e práticas), resultantes de uma criação de instâncias (protótipos e projeções), que permitem aos investigadores compreender, planear e resolver problemas inerentes ao desenvolvimento e à implementação obtendo assim sucesso, com a certeza, porém, que existem cada vez mais complexidades e exigências nas áreas emergentes.

O DSR é muito referenciado pela frase ‘Design como processo de pesquisa’ pois pretende fazer uma revisão da literatura, no qual, são analisados diferentes projetos de diferentes importâncias, validando conteúdos, funcionalidades e mecanismos oferecidos aos utilizadores. (Hevner, Ram, March, & Park, 2004)

## **1.5. Resultados Alcançados**

Os resultados alcançados procedem da compreensão e do desenvolvimento de estratégias e metodologias usadas com o fim de serem criados e utilizados modelos práticos e técnicos que apoiaram a investigação e definição do modelo.

## **1.6. Estrutura do Documento**

A estrutura do presente documento encontra-se dividida em cinco capítulos que descrevem ao pormenor e em detalhe toda a análise, tanto quanto a investigação realizada para que este projeto fosse desenvolvido e obtivesse sucesso. Assim sendo, de maneira sequencial este documento está dividido em:

Primeiro capítulo- O objetivo do primeiro capítulo é fazer uma introdução e contextualização da problemática dando também a conhecer os objetivos pretendidos, a definição das palavras-chave e a estrutura da dissertação presente.

Segundo capítulo- No segundo capítulo é realizada, detalhadamente, a revisão de literatura ou enquadramento teórico e conceptual. Neste capítulo são apresentadas as várias problemáticas acerca do tema e os conceitos correlacionados com as mesmas.

Terceiro capítulo- A descrição do planeamento realizado e efetuado para o aplicativo centrando-se nomeadamente nos requisitos.

Quarto capítulo- O quarto capítulo descreve a arquitetura do sistema: modelos, casos de uso e diagramas, entre eles, o diagrama de atividades e classes.

Quinto capítulo- O quinto capítulo detalha todo o planeamento efetuado para a conceção e desenvolvimento do protótipo. É neste capítulo que é também apresentado o *mockup*.

Sexto capítulo- O sexto capítulo representa todas as funcionalidades e interfaces criadas no aplicativo web.

Sétimo Capítulo- Por fim, no sétimo capítulo, é dada a importância da problemática e apresentada uma análise crítica às visões preexistentes e trabalho desenvolvido.

## **2. Revisão da Literatura**

### **2.1. Introdução**

Neste capítulo a elaboração da revisão da literatura teve como base a pesquisa, a análise de artigos científico, livros, documentos relacionados com a problemática e as ferramentas utilizadas. Foram definidas palavras-chave e conceitos de forma a dar início ao processo de investigação.

Todos os resultados alcançados foram adquiridos através da pesquisa bibliográfica, do cruzamento dos termos científicos e de todo conteúdo pesquisado. As plataformas mais relevantes foram o Springer, Repositório Científico da Atlântica, Google Scholar e Emerald Insight.

Toda a escolha dos artigos utilizados foi baseada na leitura e na interpretação dos conceitos e na informação dos *abstracts*, na data da sua realização.

Encontra-se toda a informação relevante utilizada nos diversos processos da pesquisa bibliográfica na secção de Bibliografia.

Foram também avaliados e comparados sites tanto quanto serviços das empresas mais relevantes no mercado, as funções disponibilizadas e os conceitos semelhantes em vigor.

## 2.2. Questões Éticas do Processo de Investigação

### Boas práticas

Segundo (Carvalho, 2015) concretização deste objetivo passa pelo bom funcionamento da aplicação e dos serviços nela disponibilizados. As aplicações web apresentam-se de forma a simplificar o acesso a serviços ou tarefas relacionadas com um determinado evento ou uma determinada necessidade dos cidadãos ou empresas.

Como boa prática devemos considerar os diversos tipos de aplicações web que podem ser criadas, ou que já existem, e avaliar as suas características. Encontrar vantagens e evitar pontos fracos, de modo a ser possível a distinção positiva comparativamente a outras.

De seguida, serão detalhados quais os tipos de aplicações web que devem ser analisados e quais os requisitos de cada aplicação ou serviços a disponibilizar.<sup>2</sup>

### Responsive web Design

Existem dois conceitos importantes a ter em conta no desenvolvimento de qualquer tipo de web sites e aplicações:

*Aplicações Web Responsive:* São aplicações desenvolvidas e programadas de forma a que os elementos que a compõe se adaptem, sem perderem a sua formatação, de forma automática às dimensões do dispositivo no qual ele está a ser visto.

*Aplicações Web Adaptativas:* São sites que se ajustam ao browser, mas sem abandonar as medidas fixas. Neste caso, os textos e as imagens não sofrem quaisquer alterações, sendo as aplicações web adaptativas apenas desenvolvidas para desktop.

---

<sup>2</sup><https://usabilidade.gov.pt/tipos-de-aplicacoes>

O objetivo de ambas as aplicações é facultarem ao utilizador a sua navegação, independentemente do dispositivo que esteja a usar. Posto isto, as aplicações web *responsive* são uma mais valia nos dias que correm, uma vez que o computador está a perder peso como plataforma de acesso à internet. (Carvalho, 2015)

Este desenvolvimento deverá seguir ainda a estratégia de *Progressive enhancement*.

### ***Progressive enhancement***

Segundo (Gustafson, 2011) ao utilizar o *progressive enhancement* o foco passa por criar um aplicativo web onde o design intuitivo e o seu conteúdo são as prioridades. O benefício é ter uma grande abrangência para os utilizadores que podem e têm acesso ao conteúdo e às funcionalidades base do website, sem exigência no browser a utilizar.

Vantagens de utilizar o *Progressive enhancement*:

- Os custos de desenvolvimento são menores do que no desenvolvimento de uma aplicação nativa porque são utilizadas linguagens de programação comuns de desenvolvimento de aplicações web;
- As atualizações são automaticamente disponibilizadas.

Desvantagens de utilizar o *Progressive enhancement*:

- Não é possível utilizar todas as funcionalidades avançadas de um sistema nativo.

### **Aplicações nativas**

(Gustafson, 2011) Descreve a utilização de uma aplicação nativa, sendo definida pelo desenvolvimento especificamente para dispositivos específicos, neste caso para um

dispositivo móvel. A aplicação nativa não é desenvolvida para facultar a navegação do utilizador no seu *browser*. As aplicações nativas são desenvolvidas para dispositivos móveis, tablets, entre outros.

Existem dois exemplos de sistemas nativos criados para uso no mercado.

O ‘Objective C’ é a linguagem utilizada no sistema operativo iOS da Apple, o Java é a linguagem base de todo o sistema operativo desenvolvido para Android.

Estes dois sistemas apenas funcionam nos dispositivos que estão preparados e otimizados para conseguirem ler o seu código. Por outras palavras, funcionam apenas nos dispositivos com os requisitos necessários para obter esses sistemas em pleno funcionamento.

O Android só ‘corre’ nos dispositivos preparados para tal, como o Sistema iOS só ‘corre’ em dispositivos Apple.

#### **Vantagens da utilização de sistemas nativos:**

- Um melhor desempenho, mais rápido e com um nível de maior fiabilidade dedicado ao hardware disponibilizado no dispositivo;
- Existem funcionalidades bastante claras com ações diretas nos dispositivos móveis, como, por exemplo, a câmara e as suas características particulares, o calendário e as suas ligações, o módulo gps, o touch ID, a capacidade do hardware e a compatibilidade com a versão do software, etc;
- Certas aplicações tem a possibilidade de trabalhar sem o recurso à internet, mas exigem também a capacidade do dispositivo.



### **Desvantagens da utilização de sistemas nativos:**

- O custo;
- A compatibilidade com outros sistemas.

O desenvolvimento de um sistema nativo requer mais complexidade, tempo de execução e conceção e, por isso, o seu valor é superior aos restantes sistemas. Os sistemas nativos também não conseguem obter sucesso sem que lhes sejam feitos testes e são adaptáveis a todos os dispositivos, outras duas razões para que o custo dos mesmos seja superior.<sup>3</sup>

### **Segurança dos dados**

O tema Segurança é um tema sensível quando mencionamos privacidade de dados e segurança da Informação. O tratamento e análise da informação recolhida é sustentado no sistema de Políticas de Segurança da Informação (PSI). O PSI permite estabelecer regras de procedimentos e métodos que têm como objetivo respeitar os dados de todos os utilizadores.

O tema supracitado tem como objetivo estabelecer diretrizes e regular a utilização dos recursos informáticos, bem como atribuir responsabilidades e definir direitos e deveres dos utilizadores. Pretende igualmente gerir expectativas de acesso e utilização, bem como restrições de modo a contribuir para o respeito à utilização e proteção da informação digital. São objetivos a ter em conta quando se pretende agir segundo um regulamento, designadamente: Normalizar as atividades de segurança, fornecer suporte às atividades de segurança que visem garantir a confidencialidade, integridade, disponibilidade e autenticidade das informações e assegurar que os recursos humanos e tecnológicos envolvidos no manuseamento e processamento da informação estão em conformidade com as regulamentações estabelecidas. (Høstland, Enstad, & Øyvind, 2010)

---

<sup>3</sup> <https://usabilidade.gov.pt/tipo-de-aplicacao-recomendada>

## **Hash**

O hash de dados é uma prática comum na ciência da computação em torno da criptografia. As funções de hash criptográficas são muito utilizadas nas Tecnologias de Informação (TI). Podem também ser usadas para assinaturas digitais, códigos de autenticação de mensagem (MACs), entre outras formas de autenticação. São também utilizadas para indexar dados em tabelas hash, impressões digitais, identificação de arquivos e deteção de duplicados, por último, é também possível detetar se um arquivo enviado sofreu ou não corrupção de dados acidental ou intencional.

O seu significado computacionalmente é codificar, e, por isso, é meramente impossível codificar uma palavra e que a mesma possa ter dois resultados diferentes. Algoritmos de hash são funções que geram um resultado de comprimento fixo (o hash ou valor de hash) de uma determinada entrada. (Pieprzyk & Sadeghiyan, 1993)

## **Encriptação de passwords**

O MD5 é um algoritmo que codifica as passwords criadas e utilizadas pelos utilizadores registados até à data no JobRadar. O MD5 foi projetado por Ronald Rivest em 1991 e veio para substituir a função hash anterior, o MD4, criado em 1990.

Quando se faz uma codificação com este hash a mensagem ou a password dos utilizadores pode ter no máximo 128 bits e é convertida em 32 caracteres hexadecimais, como os exemplos em seguida o demonstram. (Rivest, 1992)

Exemplos:

MD5("123456789")

= d41d8cd98f00b204e9800998ecf8427e

MD5("987654321")

= 6ebe76c9fb411be97b3b0d48b791a7c9

MD5(“batatafrita”)

= 85a1bd745fe08481d12effaa5cb728f7

MD5(“fritabatata”)

= 282e3f3db976440959dd3f08c4e3edca

O MD5 mostrou algumas fragilidades por não ser um código inviolável. Mas, mesmo assim, continua a ser globalmente utilizado para determinar e converter palavras chave específicas em base de dados (DBA). (Silver, Markus, & Beath, 1995)

### **Open Source**

Open Source ou Código Aberto é uma ferramenta que disponibiliza uma versão do software ao público, sem qualquer restrição. Têm também uma contribuição fundamental para a inovação fora da indústria de software mostrando que empresas de diferentes ramos têm substituído soluções proprietárias de softwares por Open Source, oferecendo flexibilidade aos seus programadores. Desta forma, contribuem também para a transformação digital, conectando organizações a sistemas ágeis e flexíveis.

O código-fonte de um programa diz respeito à sua estrutura. É um grupo de palavras e símbolos ordenados, escritos em uma linguagem de programação que permite que sejam executados por um computador. Programas de código aberto oferecem uma série de vantagens, tanto para programadores quanto para empresas que apostam nessas soluções. Isso significa que um programador consegue personalizar o software, inserindo ou retirando botões, comandos e outros itens para facilitar seu uso.

Num Mundo cada vez mais digital, a privacidade é uma das maiores preocupações de quem utiliza programas de computador, pois aumenta vulnerabilidade dos utilizadores. Ferramentas open source têm também uma grande vantagem que é poderem-se adaptar com muita facilidade a produtos de diferentes origens. (Weber, 2004)

### **Vantagens:**

- Acesso gratuito;
- Código fonte sem restrições;
- Multifacetado;
- Integridade à responsabilidade do autor do código fonte;
- Sem discriminação pessoal ou étnica;
- Sem discriminação de áreas de atuação;
- Licença renovada automaticamente;
- Licença para qualquer tecnologia.

### **Desvantagens:**

- Não tem garantias de proteção aos dados ou entidades fiscalizadoras de código aberto;
- Pode ter menor segurança devido à livre configuração. Caso não seja bem implementado, pode expor dados financeiros de clientes;
- Exige conhecimentos avançados no domínio da linguagem utilizada e do código-fonte;
- Implica uma constante confirmação a cada linha do software a fim de corrigir falhas;
- Não oferece suporte;
- Coloca toda a atualização, gestão do software e hospedagem nas mãos do programador.

### **2.3. Especificação de Requisitos**

Neste capítulo existem várias definições utilizadas para requisitos e resoluções de problemas. Existem diferentes soluções para abordar um problema e cada solução emprega ferramentas e processos diferentes para os distintos requisitos de software.

Um ponto importante para a resolução de problemas e que deve ser sublinhado é a lógica e a forma com que são conduzidas as atividades de planeamento de forma a encontrar soluções para os problemas encontrados e prevenir situações que possam surgir. Para ser possível planear ou levantar requisitos para o bom funcionamento de um software é necessário ter em conta dois fatores importantes: requisitos funcionais e requisitos não funcionais. (Bourque & Failey, 2014)

- Requisitos funcionais: são conhecidos pelas capacidades ou funcionalidades de um software. Descrevem funções que o software executa, pode também descrever etapas, conjuntos de testes e validar ambientes de atuação.
- Requisitos não funcionais: são aqueles que agem como restrições ou requisitos de qualidade. Podem ser classificados como o desempenho de requisitos, requisitos de manutenção, requisitos de segurança, requisitos de confiabilidade, requisitos de segurança e requisitos de interoperabilidade.

Neste capítulo, a palavra "sistema" significa uma combinação de elementos a cumprir consoante o objetivo definido. Estes objetivos comuns incluem hardware, software, firmware, pessoas, informações, técnicas, instalações, serviços e outros elementos de suporte. (Bourque & Failey, 2014)

Quando falamos em software, sabemos a complexidade que exige o seu planeamento e construção, para facultar estes dois pontos criou-se uma linguagem: *Unified Modeling Language* (UML). Esta Linguagem de Modelação Unificada vem auxiliar na criação e na definição das características dos sistemas, dos requisitos, da estrutura lógica, da dinâmica dos processos e, até mesmo, em relação ao hardware que o sistema poderá precisar. (Guedes, 2008).

É também considerada como uma linguagem de modelação e notação gráfica que pode ser demonstrada em vários formatos, como, por exemplo, no diagrama de casos de uso, no diagrama de atividades ou no diagrama de sequência.

Segundo (Connolly & Begg, 2015), para o desenho de um sistema de informação é de todo o interesse a criação do modelo de dados. Estes tipos de ferramentas são insubstituíveis para o bom funcionamento das aplicações. O objetivo de linguagens como esta é demonstrar como são construídas e organizadas as estruturas de dados e como são definidas definir as interligações dentro da estrutura.

(Connolly & Begg, 2015) propõe também utilizar uma metodologia composta por três tipos de abordagens:

**Modelo Conceptual:** O objetivo do modelo conceptual é uma representação descrita de alto nível (sendo exclusivamente o ponto de vista do arquiteto que planeou os dados e a sua interligação). Tudo tem de ser contextualizado, estruturado e explicativo. É um diagrama de blocos que demonstra as relações entre as entidades e os atributos. Tem a preocupação de captar e retratar a realidade de uma organização ou de um projeto, processo de negócio, setor de atividade, departamento, regras, ferramentas, propriedades, técnicas, processos e mapeamento de domínio do utilizador. Assim sendo, acaba por dar origem a um modelo lógico e físico da base de dados, onde o objetivo é a organização da informação.

**Modelo Lógico:** O modelo lógico agrega detalhes da implementação. É influenciado pelo seu Sistema de Gestão de bases de dados (SGBD) que contém as funções e regras do projeto. O modelo lógico demonstra-nos tabelas e as ligações entre elas, definições de dados como as chaves primárias, chaves estrangeiras e os seus componentes.

**Modelo Físico:** O modelo físico inclui a representação do projeto com a interface gráfica, base de dados e programas utilizados. Este modelo implica a regra de normalização de dados. O arquiteto tem de estar familiarizado com a estrutura do projeto para uma implementação de uma base de dados normalizada, criar tabelas necessárias para um bom funcionamento da base de dados, sendo isto uma exigência para uma boa eficiência e consulta de dados e ter em atenção as principais características e recursos necessários para armazenamento e manipulação das estruturas de dados (conseguir fazer *Select's*,

*Delete's e Insert's* na base de dados). Trata-se de sequências de comandos executados em *Structured Query Language* (SQL) por forma a conseguir gerir, extrair, criar e apagar registos nas tabelas existentes da base de dados.

## 2.4. Modelo de Segurança

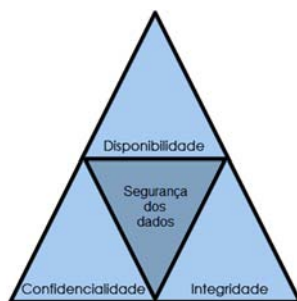
O modelo de Segurança da Informação (SI) é conhecido por ser um processo organizado e estruturado que permite preservar a confidencialidade, integridade e a disponibilidade da informação. Tem como objetivo a prevenção contra ameaças dentro e fora da rede em que o domínio é executado.

Existem três pontos a destacar na segurança dos dados, segundo (Connolly & Begg, 2015) são:

**Disponibilidade:** Diz respeito ao acesso à informação e bens associados por quem está devidamente autorizado. A informação deve estar acessível sempre que necessário sem interrupções.

**Integridade:** Pretende garantir a veracidade e o estado da informação, bem como os seus métodos de processamento. O conteúdo da informação não pode ser modificado de nenhuma forma inesperada.

**Confidencialidade:** A preocupação é assegurar que a informação é acessível somente a pessoas devidamente autorizadas. O acesso à informação é restrito a utilizadores legítimos.



**Figura 1 – Modelo de Segurança**

Disponível em: <[https://www.gta.ufrj.br/grad/12\\_1/seg\\_smartgrid/possiveisataques.html](https://www.gta.ufrj.br/grad/12_1/seg_smartgrid/possiveisataques.html)>

### **Segurança operacional**

(Lenaeus, et al., 2015) refere-se ao processo de atribuir privilégios, funções e permissões para utilizadores ou grupos de utilizadores de maneira a que se consiga fazer a gestão de acesso do domínio. A segurança operacional controla o acesso aos dados, procedimentos e serviços de um domínio informático.

A segurança operacional é constituída por restrições de acessos dos utilizadores utilizando meta-dados com base na função do utilizador.

As funções e permissões de cada utilizador são definidas por ações e restrições em função do tipo de utilizador dentro da organização.

### **Segurança de Infraestrutura**

Segundo (Lenaeus, et al., 2015), a segurança de infraestrutura tem como objetivo proteger toda a rede informática contra acessos e alterações indesejadas.

Este tipo de segurança é constituído por normas e medidas de proteção de dados transmitidos e armazenados num domínio informático. A autenticação de utilizadores e os serviços estabelecidos podem ser um ponto de acesso a acessos indesejados. Foi através da segurança de infraestrutura que as passwords dos utilizadores foram encriptadas com o algoritmo MD5.

As conexões para componentes externos, incluindo aplicações cliente, bases de dados relacionais, ligações origem e destino devem ser estrategicamente bem concebidas para não serem postas de parte vulnerabilidades que possam ser exploradas de forma indesejada.

Para (Lenaeus, et al., 2015) a implementação de um *Disaster Recovery Plan* (DRP) é sempre uma mais valia como meio de prevenção de um suposto ataque ou uma falha existente.

Para isso é preciso passar por 3 etapas:

1. Gestão de crise: contempla as primeiras ações para solucionar o problema;



2. Manutenção da continuidade operacional: é o processo de realização das rotinas para assegurar o funcionamento dos serviços;
3. Recuperação de serviços: volta-se para os itens afetados, como dispositivos que deixaram de funcionar e dados corrompidos.

## 2.5. Arquitetura Cliente-Servidor

A Internet tornou-se uma ferramenta fundamental tanto como trabalho, como para diversão ou comunicação. Acompanhando a evolução da internet, também os browsers evoluíram. Foi devido às evoluções mencionadas anteriormente que o *Hyper Text Markup Language* (HTML) também evoluiu, criando versões mais atualizadas e funcionais, de modo a ser acessado a partir de qualquer browser e a responder às necessidades da Web.

No intervalo de 1993 a 1995, a linguagem HTML teve grandes avanços. Foram implementadas várias versões com melhorias que vieram enriquecer todas as suas funcionalidades e ligações com outras linguagens. No decorrer do ano 1997 foi desenvolvida a versão HTML 3.2 pelo grupo *World Wide Web Consortium* (W3C) de modo a ser aceite como universal (Eis & Ferreira, 2012).

O HTML 4 foi então tornado como uma ferramenta usada mundialmente, contudo foram encontradas algumas necessidades às quais não correspondia. Por essa mesma razão, apareceu o HTML 5 como uma versão mais completa, substituta e corrigida à versão anterior.

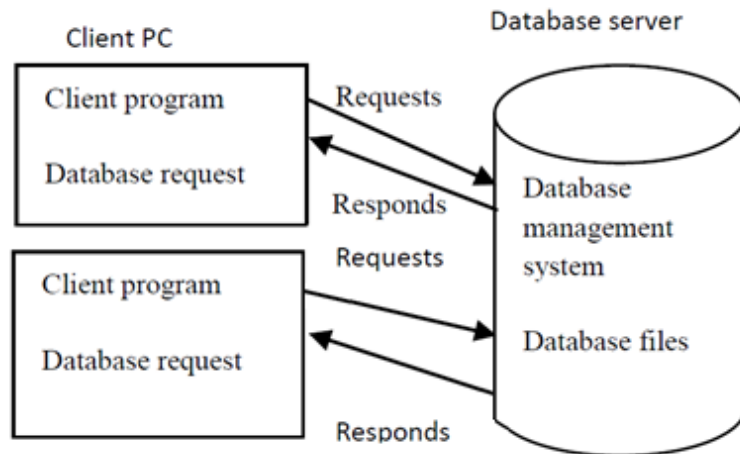
Com isto, o HTML 5 veio fornecer à Web características inexistentes em versões anteriores, como se refere (Eis & Ferreira, 2012).

Das novas características que trouxe, podemos mencionar a criação de tags que permitem incluir secções muito utilizadas como rodapés, cabeçalhos, menus e sidebars, entre outros, de forma automática utilizando o *Cascading Style Sheets* (CSS) como complemento, fazendo do JavaScript uma linguagem mais eficiente e versátil.

Em paralelo com o HTML a possibilidade de se usar o CSS, oferece ainda outro tipo de vantagens, funcionalidades e interação, acabando por ser uma linguagem interligada que controla todo o aspeto visual do HTML. O CSS é considerado uma linguagem prioritária em termos de hierarquia, isto é, sempre que existe uma definição definida no CSS, esta é priorizada pelo browser em relação às definições criadas no HTML, mesmo que o CSS seja considerado uma extensão. O CSS vem formatar características básicas, como cores, tamanho das fontes de letras, estilos, *forms*, margens, botões, entre outros. Finalizando, o CSS 3 veio tornar-se o maior componente web a trabalhar em conjunto com o HTML, tornando mais fácil a criação de designs e layouts como uma linguagem unificada (Eis & Ferreira, 2012).

Segundo (Norman, 2013), a engenharia da usabilidade serve e tem como objetivo conseguir detetar problemas de design logo na fase inicial do desenvolvimento, poupando recursos e tempo, oferecendo benefícios importantes e evitando a insatisfação dos clientes. Por esta razão, há questões essenciais num ciclo de vida de um produto tecnológico, tais como a tecnologia e a experiência do utilizador que são pilares de sucesso para a sua satisfação.

No mundo da computação, atualmente, alguns dos protocolos padronizados que os clientes e os servidores usam para se comunicarem são o *File Transfer Protocol* (FTP), o *Simple Mail Transfer Protocol* (SMTP) e *Hypertext Transfer Protocol* (HTTP)/(HTTPS). Assim, o sistema cliente-servidor pode ser definido como uma arquitetura de software composta tanto pelo cliente quanto pelo servidor, em que os clientes enviam solicitações e os servidores respondem. O cliente-servidor fornece uma comunicação entre processos porque envolve a troca de dados tanto do cliente como do servidor, sendo desempenhadas funções diferentes. (Sulyman, 2014)



**Figura 2 – Modelo Cliente Servidor**

Disponível em <[https://www.academia.edu/43822833/Client\\_Server\\_Model](https://www.academia.edu/43822833/Client_Server_Model)>

Uma arquitetura de um sistema cliente-servidor é geralmente composta por um servidor de aplicações, um servidor de base de dados e um PC.

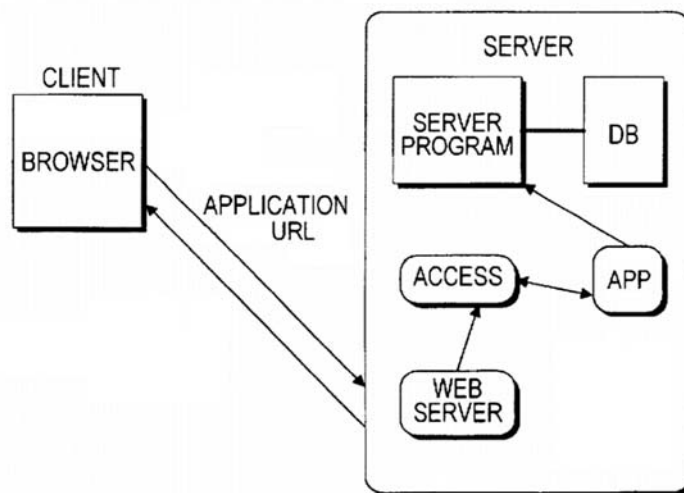
As duas arquiteturas principais são a arquitetura de duas e três camadas. A arquitetura de sistema cliente-servidor de duas camadas é uma arquitetura que envolve apenas o servidor de uma base de dados e um PC de um cliente. Nesta mesma arquitetura, os utilizadores executam as aplicações no PC, que se conecta através da rede ao servidor. Ao contrário do sistema cliente-servidor de três camadas que necessita de um *middleware*, que é um servidor de aplicações onde existe um software que é executado numa máquina à parte, responsável por toda a gestão lógica da aplicação. Neste sistema de três camadas a aplicação cliente executa a codificação lógica de negócios e, em seguida, envia uma resposta para o utilizador. É considerado quando o cliente tem acesso à base de dados diretamente, sem envolver qualquer intermediário. Ainda neste caso, o servidor responsável pela base de dados também executa a parte lógica responsável pela atribuição respetiva de informação a cada um dos clientes. (Sulyman, 2014)

Um sistema orientado a objetos ilustra um mecanismo complexo, como a imagem a baixo mostra. Desenvolver um sistema Web orientado a objetos baseado em componentes,

camadas e interligar aplicações exige modelação de dados, interpretação do problema. Realizar a conexão entre a interface e a base de dados, o web server e a linguagem utilizada, e estruturar os algoritmos de forma a que permitam representar as regras de negócio. Todo este sistema descreve o aplicativo Web JobRadar.

O utilizador ao entrar no URL da aplicação web, faz o login através do browser, e, em seguida, faz um pedido de acesso através do servidor web, que, juntamente com as informações do utilizador contacta o nó de acesso à aplicação. Este nó de acesso contém o programa servidor (Apache) que faz a disponibilização do conteúdo online, interligado com o gerenciador de base de dados PHP My Admin que contém a base de dados (SQL), e, por sua vez, confirma a autenticidade dos dados do utilizador na camada de acesso à aplicação.

O servidor web e os motores de apresentação em HTML e CSS, juntamente com o PHP e JavaScript criam um aplicativo que executa todas as definições para que foi programado. Assim é disponibilizado o conteúdo Web como uma plataforma de aplicação cliente-servidor e canal de distribuição. (Gish, 1998)



**Figura 3 – Canal de Distribuição (Ligação Cliente-Servidor)**

Adaptado através de < <https://patentimages.storage.googleapis.com/ef/59/cc/ac07c47d5fcf5e/US5768510.pdf> >

O Google Maps é um serviço de pesquisa gratuito na web. Permite-nos visualizar mapas e imagens satélite da Terra. Esta aplicação é fornecida e desenvolvida pela empresa Google. Atualmente, o serviço disponibiliza mapas, rotas de um ponto para outro no globo, transportes e horários atualizados diariamente, trânsito, estimativas de viagens e possíveis incidentes no tráfego. Permite verificar a nível geográfico declives, tipografias de terrenos, meios aquáticos ou terrestres. É possível conhecer sítios de interesse, explorar cidades e até o mundo com mapas estáticos e dinâmicos, com imagens de *Street View* e rotações 360°. Para além disto, permite ainda a customização e personalização como guardar locais e pontos de interesse referenciados com a latitude e longitude a nível GPS.

Para a criação e utilização de uma API ao nível de frameworks, existe a necessidade de um registo inerente à criação da chave API. Para se proceder à configuração e autenticação é necessário ter uma conta Google associada ao Google Cloud. Só aí a Chave JavaScript API, constituída por uma strings criptografada permite a ligação com a aplicação criada. A chave usada é associada à faturação da Google de modo a ser possível rastrear todas as solicitações de API associadas. Contudo, todas as chaves têm limitações associadas e não permitem identificar nem as aplicações que a usam como qualquer utilizador a executar a solicitação da API, pelo que não é possível restringir o acesso a utilizadores específicos ou a contas de serviços, sendo recomendável o uso de uma autenticação padrão. Para uma maior customização e utilização da mesma existem planos de upgrades e ferramentas que são possíveis ser acrescentadas para uma maior eficiência.<sup>4</sup>

### **3. Especificação de Requisitos**

#### **3.1. Visão Geral**

De forma a ser possível efetuar-se o levantamento de requisitos de um software, é necessário ter em conta todas as funcionalidades que o sistema deve desempenhar. Esta exigência corresponde à importância de implementar os requisitos desejados e funções suportadas pelas tecnologias adotadas. Os seguintes tópicos têm como objetivo,

---

<sup>4</sup> <https://developers.google.com/maps/documentation/javascript/overview>

identificar as funções, de modo a permitir aos futuros utilizadores da aplicação executar as suas tarefas.

Por conseguinte, cada requisito funcional será representado numa tabela que irá identificar a prioridade de cada requisito, quais os atores intervenientes e quais as tarefas necessárias para a sua concretização.

### 3.2. Requisitos Funcionais

Os requisitos funcionais têm um papel de extrema importância durante o desenvolvimento de um sistema, podendo ser usados como critérios de seleção na escolha de alternativas do projeto e forma de implementação. A informação sobre as arquiteturas existentes aliada à experiência do arquiteto de software tem influência direta na definição de requisitos. Assim sendo, (Bourque & Failey, 2014) define que os requisitos funcionais descrevem funções que tem e devem ser executadas pelo software. Estes requisitos são também conhecidos como capacidades ou características.

As tabelas (seguintes) identificam os principais requisitos funcionais do JobRadar, sendo estes identificados por um código, que está classificado em termos de prioridade (Alta, Média ou Baixa). Cada uma destas classificações está designada como: Alta – Fundamental ao Sistema; Média – Deve ser incremento ao Sistema; Baixa – Opcional ao Sistema.

**Tabela 1 – Requisito Funcional Utilizador Anónimo (RFUA)**

Código	Prioridade	Descrição do requisito
RFUA01	Alta	Permite o acesso à <i>Homepage</i> : JobRadar.
RFUA02	Alta	Permite a visualização de todas as seções e detalhes da <i>Homepage</i> .
RFUA03	Baixa	Apenas permite a visualização da informação à cerca das Dicas e Notícias da plataforma.
RFUA04	Alta	Permite a criação de registo utilizador.
RFUA05	Alta	Permite a criação de registo, e ter a possibilidade de escolher o seu tipo de perfil (Candidatos/Empresas).
RFUA06	Média	Permite pedido de ajuda ou informação na seção de contactos.

**Tabela 2 – Requisito Funcional Utilizador Autenticado Candidato (RFUAC)**

<b>Código</b>	<b>Prioridade</b>	<b>Descrição do requisito</b>
<b>RFUAC01</b>	Alta	Permite o acesso à <i>Homepage</i> : JobRadar.
<b>RFUAC02</b>	Alta	Permite a visualização de todas as seções e detalhes da <i>Homepage</i> .
<b>RFUAC03</b>	Média	Permite a visualização de Dicas e Notícias da plataforma.
<b>RFUAC04</b>	Alta	Permite a criação de registo utilizador.
<b>RFUAC05</b>	Alta	Permite a criação de registo, e ter a possibilidade de escolher o seu tipo de perfil (Candidatos/Empresas).
<b>RFUAC06</b>	Média	Permite pedido de ajuda ou informação na seção de contactos.
<b>RFUAC07</b>	Alta	Permite iniciar e terminar a sessão no aplicativo JobRadar.
<b>RFUAC08</b>	Alta	Permite a visualização de todos os anúncios de emprego disponíveis.
<b>RFUAC09</b>	Alta	Permite filtrar todos os anúncios de emprego disponíveis, pelo nome de empresa, nome do anúncio, empresa ou pela descrição.
<b>RFUAC10</b>	Alta	Permite inserir os seus dados pessoais, fotografia e CV, consultá-los e alterá-los.
<b>RFUAC11</b>	Médio	Permite inserir o seu trabalho de sonho e descrevê-lo.
<b>RFUAC12</b>	Médio	Permite guardar todos os anúncios de emprego que forem do seu agrado nos seus favoritos.
<b>RFUAC13</b>	Alta	Permite visualizar todas as candidaturas enviadas que estejam pendentes de uma revisão, e até eliminá-las.
<b>RFUAC14</b>	Alta	Permite visualizar todas as candidaturas já revistas e qual o seu feedback
<b>RFUAC15</b>	Médio	Permite visualizar notícias recentes com informações relevantes
<b>RFUAC16</b>	Médio	Permite visualizar dicas relevantes em como agir e como se preparar para futuras entrevistas.

**Tabela 3 – Requisito Funcional Utilizador Autenticado Empresa (RFUAE)**

<b>Código</b>	<b>Prioridade</b>	<b>Descrição do requisito</b>
<b>RFUAE01</b>	Alta	Permite o acesso à <i>Homepage</i> : JobRadar.
<b>RFUAE02</b>	Alta	Permite a visualização de todas as seções e detalhes da <i>Homepage</i> .
<b>RFUAE03</b>	Média	Permite a visualização de Dicas e Noticias da plataforma.
<b>RFUAE04</b>	Alta	Permite a criação de registo utilizador.
<b>RFUAE05</b>	Alta	Permite a criação de registo, e ter a possibilidade de escolher o seu tipo de perfil (Candidatos/Empresas).
<b>RFUAE06</b>	Médio	Permite pedido de ajuda ou informação na seção de contactos.
<b>RFUAE07</b>	Alta	Permite iniciar e terminar a sessão no aplicativo JobRadar.
<b>RFUAE08</b>	Alta	Permite disponibilizar anúncios de emprego.
<b>RFUAE09</b>	Alta	Permite visualizar todos os anúncios de emprego disponibilizados.
<b>RFUAE10</b>	Alta	Permite inserir os dados da empresa, consultá-los e alterá-los.
<b>RFUAE11</b>	Alta	Permite visualizar todas as candidaturas enviadas para os anúncios disponibilizados pela própria empresa.
<b>RFUAE12</b>	Alta	Permite analisar, e visualizar toda as candidaturas recebidas e dar feedback aos candidatos.



**Tabela 4 – Requisito Funcional Administrador (RFA)**

Código	Prioridade	Descrição do requisito
RFA1	Alta	Possibilitar o acesso ao Backoffice.
RFA2	Alta	Através do Backoffice, o administrador pode visualizar todas contas criadas, fazer a gestão de utilizadores, fazer a gestão de toda a base e dados, editar informações, design da plataforma e inserir e retirar <i>features</i> .
RFA3	Alta	Através do Backoffice, o administrador pode adicionar, editar e remover funcionalidades.
RFA4	Alta	Através do Backoffice, o administrador pode corrigir bugs.
RFA5	Alta	Através do Backoffice, o administrador pode solucionar e informar utilizadores de condutas e problemas solucionados.
RFA6	Alta	Através do Backoffice, o administrador deve poder adicionar, editar e remover utilizadores.
RFA7	Alta	Permite iniciar e terminar a sessão no aplicativo JobRadar.

**Tabela 5 – Requisito Funcional Sistema (RFS)**

Código	Prioridade	Descrição do requisito
RFS1	Média	O aplicativo Web deve dispor de uma navigation-bar que permite uma navegação rápida para diferentes seções dentro do aplicativo.
RFS2	Baixa	A aplicação deve dispor de um logótipo clicável que permite a navegação instantânea para o topo da homepage, mesmo que o utilizador se localize em qualquer outra seção.
RFS3	Alta	A aplicação dispõe de um Backoffice para a gestão do sistema acedido apenas pelo seu Administrador.
RFS4	Alta	A aplicação deve conter uma seção de contactos para os utilizadores poderem contactar o Administrador.
RFS5	Alta	A aplicação deve ter mecanismos com que haja interação entre os candidatos e as empresas que disponibilizam os anúncios.
RFS6	Alto	A aplicação deve mostrar ao utilizador os anúncios relativos às palavras chave inseridas na seção de pesquisa.

<b>RFS7</b>	Médio	A aplicação deve guardar os anúncios nos favoritos dos candidatos sempre que o candidato carregar no botão para o efeito.
<b>RFS8</b>	Alta	A aplicação deve permitir ao utilizador editar os detalhes do seu perfil sempre que quiser.
<b>RFS9</b>	Alta	A aplicação deve permitir ao utilizador ver, editar e apagar os seus registos e informações sempre que quiser.
<b>RFS10</b>	Média	A aplicação deve guardar as alterações efetuadas na edição do perfil e informações relativas, e o sistema deve exibir uma mensagem a informar caso não tenha sido concretizado.
<b>RFS11</b>	Alta	A aplicação não deve permitir a um utilizador anónimo entrar na zona reservada.
<b>RFS12</b>	Alta	O sistema deve exibir apenas aos utilizadores autenticados as funções permitidas para o seu tipo de conta.

### 3.3. Requisitos Não Funcionais

Contrariamente aos requisitos funcionais, os requisitos não funcionais ou requisitos de qualidade como são conhecidos, restringem a solução (Bourque & Failey, 2014). Podemos dar como exemplo deste tipo de requisitos, requisitos de desempenho, requisitos de manutenção, requisitos de segurança, requisitos de confiabilidade, entre outros.

Para (Arruda, 2020), um requisito não funcional é um conjunto de restrições impostas no sistema de maneira a ser desenvolvido, ditando, por exemplo, a utilidade, atratividade e confiabilidade do requisito.

Os requisitos não funcionais ajudam a definir as funcionalidades do sistema, expressando uma característica do mesmo em forma de qualidade ou restrição.

**Tabela 6 – Requisitos Não Funcionais (RNF)**

Código	Prioridade	Categoria	Descrição do requisito
RNF01	Alta	Implementação	A aplicação deve ter sido implementada com políticas de segurança e mecanismos de proteção de erros.
RNF02	Médio	Implementação	A aplicação deve ser desenvolvida recorrendo a linguagens OpenSource.
RNF03	Alta	Implementação	A aplicação deve ser bem desenvolvida de forma a não haver incompatibilidades de softwares.
RNF04	Alto	Usabilidade	A aplicação deve estar em português.
RNF05	Alta	Usabilidade	A aplicação deve dispor de um design simples e intuitivo.
RNF06	Média	Usabilidade	A aplicação deve dispor de uma unidade de interface <i>responsive</i> e que se ajuste de maneira eficaz ao dispositivo que o utilizador possua.
RNF07	Média	Confiabilidade	A aplicação deve permitir sempre acesso a cada utilizador aos seus dados através de redundância.
RNF08	Alta	Confiabilidade	A aplicação deve estar disponível pelo menos 99% do tempo em todos os dias úteis.
RNF09	Alta	Compatibilidade	A aplicação deve funcionar em qualquer <i>browser</i> .
RNF10	Médio	Compatibilidade	A aplicação deve ser permitir a sua utilização através de qualquer dispositivo.
RNF11	Alta	Disponibilidade	A aplicação deve ser acedida a partir da ligação à <i>internet</i> .
RNF12	Alta	Disponibilidade	A aplicação não pode permitir o acesso de um utilizador, aos dados de outro utilizador.
RNF13	Média	Segurança	O sistema deve bloquear o acesso ao utilizador que após enumeras tentativas falhadas, insista em continuar, deve identificar uma ameaça e impossibilitar o <i>login</i> a esse IP.
RNF14	Alta	Segurança	A <i>password</i> de cada utilizador autenticado deve ser encriptada.
RNF15	Médio	Escalabilidade	A aplicação deve ser escalável de modo a acompanhar novas tendências, e poder evoluir com o mercado emergente.
RNF16	Média	Escalabilidade	A aplicação deve ser escalável de modo a ser possível receber novas <i>features</i> sempre que necessário.
RNF17	Média	Desempenho	Ao iniciar a aplicação, esta não deve demorar mais que 5 segundos a exibir a <i>Homepage</i> .
RNF18	Baixa	Desempenho	A aplicação deve ter uma boa estrutura para não bloquear, não demonstrar bugs ou erros de código.

#### **4. Arquitetura do Sistema**

Neste capítulo é apresentada a arquitetura do sistema desenvolvido para a aplicação “JobRadar”. É apresentada a metodologia e as ferramentas utilizadas, de forma sucinta no decorrer do capítulo.

A arquitetura do sistema foi baseada num modelo de quatro pontos. O modelo escolhido e, por consequência usado, foi o UML que é constituído, tal como o nome indica por uma Linguagem de Modelação Unificada. O UML serviu assim de base para a elaboração do diagrama de uso, do diagrama de classes, do diagrama de objetos e do diagrama de atividades.

Com base neste modelo Servidor-Cliente, as tecnologias utilizadas foram o HTML, CSS, JavaScript, JQuery no Client-Side, PHP e MySQL que foram utilizadas em conjunto com o Web Server Apache no lado do Server-Side. Posto isto:

- HTML5 (Hypertext Markup Language): É a quinta geração do padrão HTML desenvolvido em 2007, sendo oficializado em 2014 na construção de páginas Web;
- CSS3 (Cascading Style Sheets): São páginas de script com o estilo usado no formato de layout de cada site. O CSS3 acaba assim por ser a apresentação gráfica de páginas Web;
- Javascript: Linguagem de script (OpenSource) usada no desenvolvimento web com o propósito de adicionar elementos dinâmicos e interativos aos sites;
- JQuery: É uma biblioteca JavaScript que permite adicionar funcionalidades extras ao código JavaScript dentro do HTML das páginas web;
- PHP: É uma linguagem de scripting incorporada no código HTML de páginas da web. Por ser considerado seguro é utilizado em bases de dados e ligações ao servidor com outras informações;
- MySQL: É o tipo de bases de dados mais comum entre os servidores web sendo um sistema de gestão de bases de dados baseado na linguagem SQL desenvolvida pela Microsoft;

- Bootstrap 3.3: A estrutura popular de desenvolvimento da web usada para criar sites inclui estilos CSS, arquivos HTML, Javascript e JQuery com bibliotecas dinâmicas para a criação de aplicações Web.

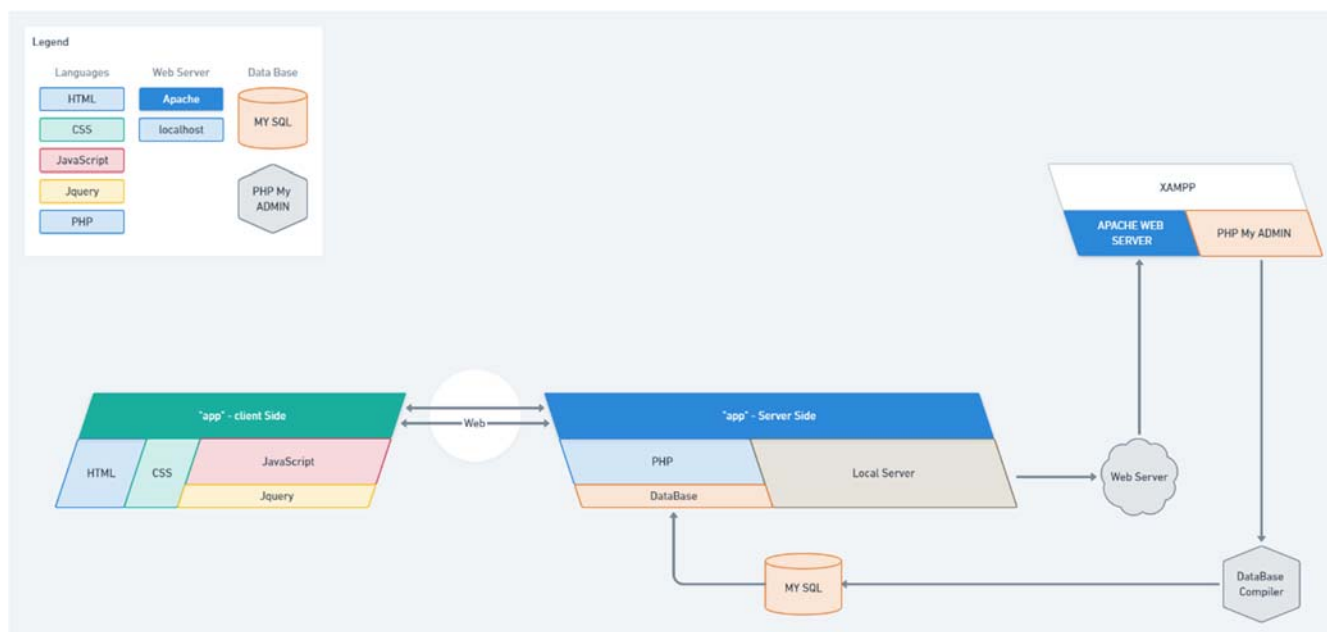
Para o desenvolvimento do aplicativo web JobRadar foi utilizado um Bootstrap *free*, como template, aqui referido<sup>5</sup> de autoria de Paul Silva. Este Bootstrap inclui uma página estática “Index” em HTML e CSS e contém um código fonte que o faz ser web responsive. As alterações efetuadas no template para o desenvolvimento do JobRadar não incluíram a alteração de nenhuma biblioteca.

Este template foi usado para criar um aplicativo web dinâmico, tendo sido necessário alterar todo o seu conteúdo, acrescentando código PHP juntamente com JavaScript. Foi também necessário criar várias páginas e funcionalidades, de forma a ser possível interagir com a base de dados dedicada e com o servidor Apache.

Desta forma, foi possível realizar uma conexão entre tecnologias com o propósito de criar um aplicativo web completo e funcional na gestão dos seus recursos e indo ao encontro das necessidades apresentadas na sua conceção e no seu desenvolvimento, como demonstra a figura 5.

---

<sup>5</sup> <https://templatemag.com/instant-bootstrap-personal-template/>



**Figura 4 – Arquitetura do Sistema JobRadar**

Criado através de <Whimsical.com>

#### 4.1. Casos de Uso

Em todos os processos de desenvolvimento de um software, tem de haver a distinção de diversos os cenários. Os cenários vêm mostrar diferentes realidades num universo de diversas interações no sistema ilustrado.

Segundo (Bornia, 2005), os casos de uso servem para facilitar a visualização da aplicação, ajudando a delimitar o sistema. Servem também de meio de comunicação entre os utilizadores finais por oferecerem uma visão dinâmica do sistema.

Foi necessário identificar os atores intervenientes no sistema “JobRadar”, isto é, distingui-los. A perceção da existência de um Utilizador Anónimo, dois tipos de Utilizadores Registados e o Administrador do Sistema “JobRadar”.

Os casos de uso auxiliam a dimensionar a capacidade de processamento necessária. Servem como base para o desenvolvimento incremental e ajudam a estimar o tamanho de

projetos e recursos necessários. São utilizados como ferramenta para controlar um projeto. (Bornia, 2005)

Para definir os casos de uso é necessário avaliar os requisitos funcionais e as suas definições, como veículo de captura de requisitos.

Utilizando o diagrama UML, o diagrama de casos de uso é composto por:

- Interface: As suas características e o desencadear de ações, consoante cada tipo de interveniente e a sua utilização;
- Utilizadores: Tipos de utilizadores existentes no sistema e a distinção entre acessos;
- Funcionalidades: Ações associadas às permissões, acessos e tipos de utilizadores;

**Tabela 7 – Intervenientes do Sistema JobRadar**

Tabela intervenientes do Sistema		
Tipo	Ator	Descrição
Externo ao sistema	Utilizador anónimo	Qualquer visitante do website, sem registo criado no sistema "JobRadar"
Interno ao Sistema	Utilizador registado	Utilizador/Candidato registado no sistema "JobRadar"
Interno ao Sistema	Utilizador registado	Utilizador/Empresa registado no sistema "JobRadar"
Interno ao Sistema	Administrador do sistema	Administrador do sistema "JobRadar"

Após identificar os atores, foram identificados os principais casos de uso que se encontram na Tabela 1, acima descritos.

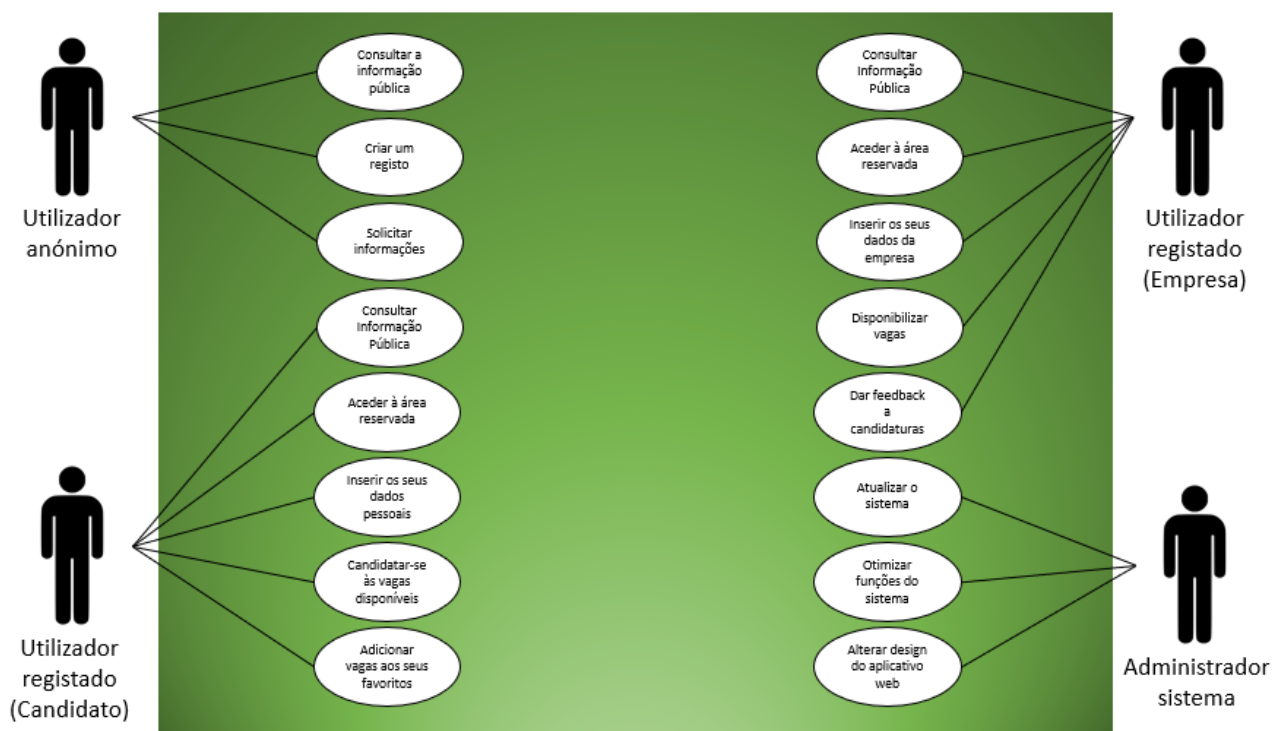
**Tabela 8 – Casos de Uso JobRadar**

Tabela casos de uso	
Ator	Casos de Uso
Utilizador anónimo	Consultar a informação pública Criar um registo Solicitar informações
Utilizador registado (candidato)	Consultar Informação Pública Aceder à área reservada Inserir os seus dados pessoais Inserir as suas áreas de interesse Candidatar-se aos anúncios disponíveis Adicionar anúncios de emprego aos seus favoritos
Utilizador registado (empresa)	Consultar Informação Pública Aceder à área reservada Inserir os seus dados da empresa Disponibilizar anúncios Avaliar candidaturas aos seus anúncios disponibilizadas Dar feedback às candidaturas enviadas por candidatos Agendar entrevistas
Administrador do sistema	Atualizar o sistema Otimizar funções do sistema Alterar design do aplicativo web

A Tabela 2 descreve os atores interventivos no sistema JobRadar.



Por forma a se conseguir visualizar todos os atores intervenientes no sistema “JobRadar” e nos casos de uso foi elaborado o Modelo de Casos de Uso, demonstrado no esquema referido abaixo, Figura 7.



**Figura 5 - Atores do sistema JobRadar e principais casos de Uso**

Adaptado através de < [https://www.researchgate.net/figure/Figura-4-Diagrama-de-caso-de-uso-O-sistema-de-doacoes-e-estruturado-por-tres-classes-que\\_fig4\\_322003640](https://www.researchgate.net/figure/Figura-4-Diagrama-de-caso-de-uso-O-sistema-de-doacoes-e-estruturado-por-tres-classes-que_fig4_322003640) >

## **4.2. Diagrama de Atividades**

Pretende-se com esta representação gráfica conseguir descrever os passos e as etapas sequenciais de cada processo. O primeiro passo para criar o Diagrama de Atividades foi estruturar a problemática e ações, e, dessa forma, criar hipóteses possíveis para obter a finalidade desejada.

Simplificando, cada ação é criada num fluxo de atividades entre cada elemento, acabando por interligar o Candidato e a Empresa em certas ações ou fases dos processos de construção. O JobRadar foi projetado através deste diagrama de atividades.

A Figura 10, que representa o Diagrama de Atividades, descreve o desencadear de hipóteses de decisão que podem ser tomadas no fluxo de ações e no desenrolar da aplicação. As mudanças após cada ação proporcionam hipóteses distintas entre cada utilizador envolvido.

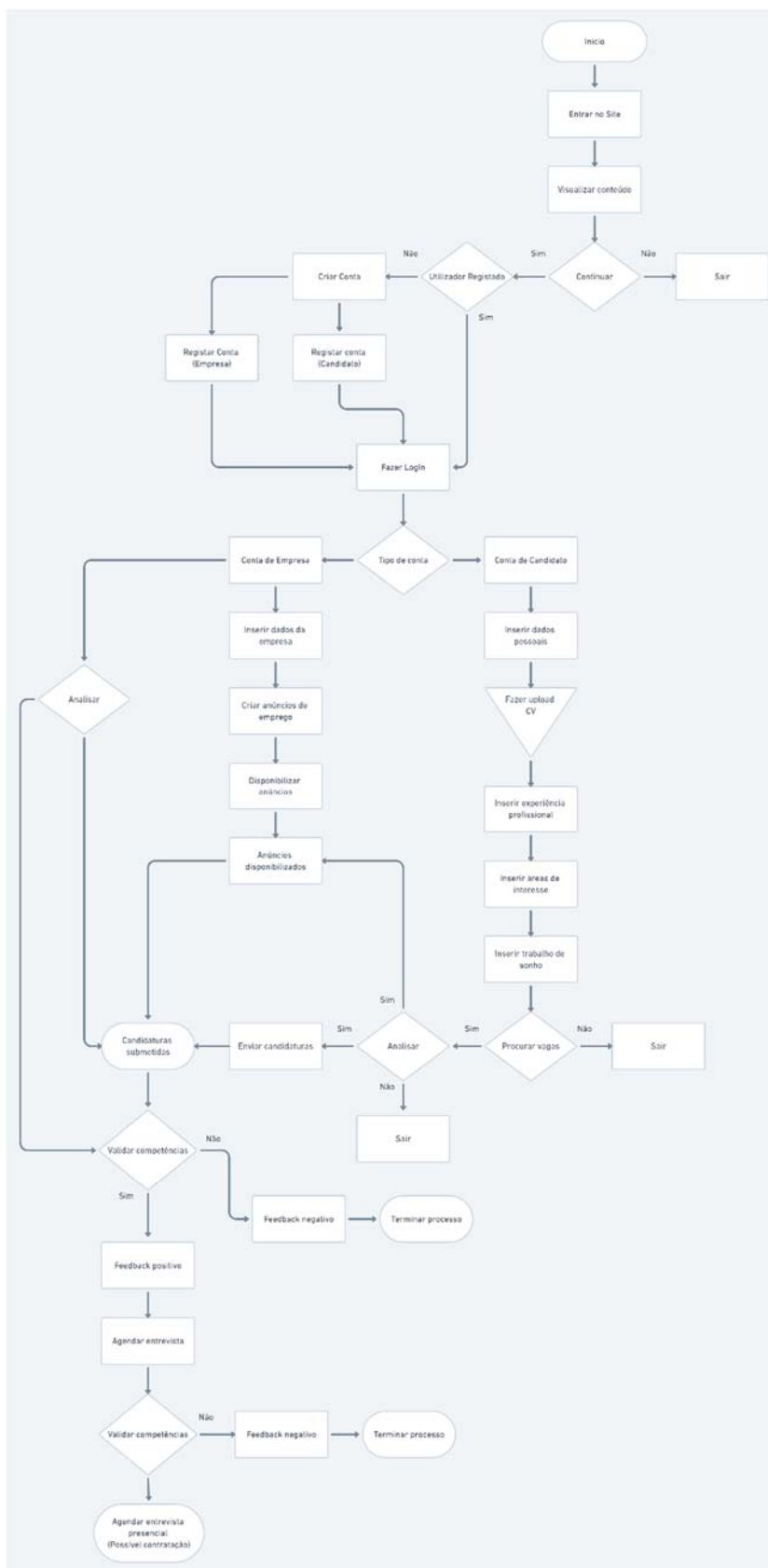


Figura 6 – Diagrama de Atividades JobRadar

Criado através de <Whimsical.com>

### 4.3. Diagrama de Classes

Através da utilização do modelo UML, o modelo de domínio (também designado por modelo conceptual), ilustra a relação estática entre classes.



Figura 7 – Classes (Candidato / Candidaturas)

Demonstrando a relação estática das classes do sistema destes diagramas, é apresentada a diferença a nível de relacionamento e detalhe, como visível na Figura 7 e na Figura 8.

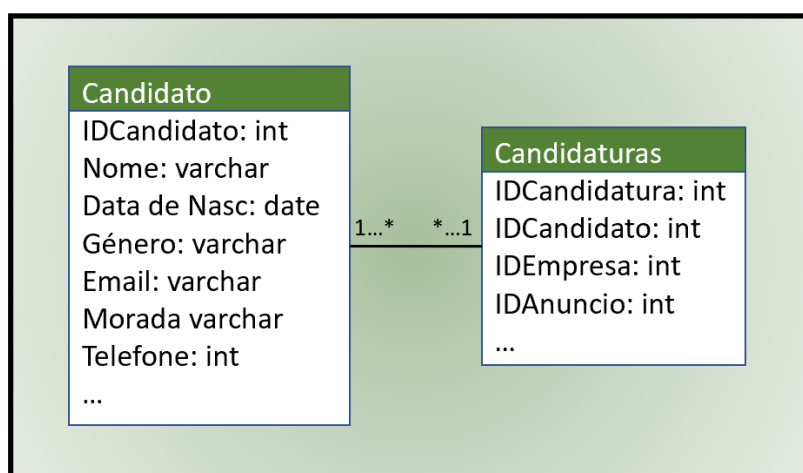


Figura 8 – Diagrama de Classes (Candidato / Candidaturas)

Neste diagrama são apresentadas as relações objetivas entre as principais classes Candidato e Candidaturas e os seus atributos.

#### 4.4. Modelo de Domínio

A melhor forma de criar um modelo de domínio é especificar o domínio em excesso e, posteriormente, o modelo de domínio incluirá as classes conceituais de maior importância.

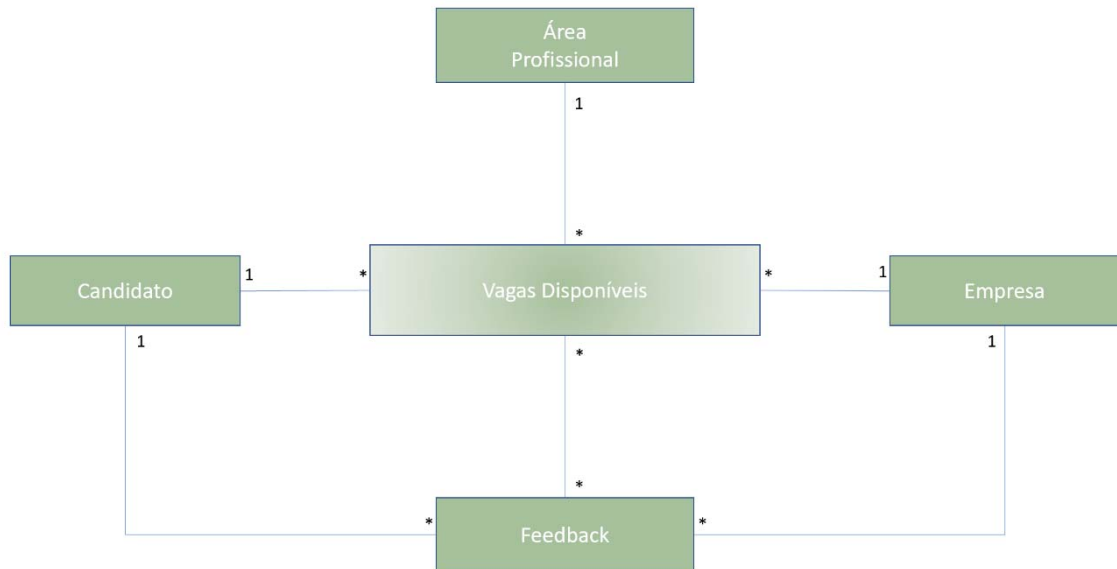


Figura 9 – Modelo de Domínio JobRadar

Criar um modelo de domínio é essencial, representando o artefacto mais importante, e, criando-o durante a análise orientada aos objetos. Desta forma, o modelo de domínio permite ilustrar as classes conceituais significativas no domínio do problema.

Para representar as classes conceituais e os objetos do mundo real, aplica-se o conceito de UML, complementando o modelo de domínio com a notação na forma de diagramas de classe.

#### **4.5. Modelos básicos de Base de dados relacional**

Uma base de dados é um repositório onde são guardadas informações. Todas as informações podem ser consultadas, alteradas e apagadas, na totalidade ou parcialmente, através de uma aplicação conhecida como Sistema de Gestão de Base de Dados (SGBD).

A informação é organizada conforme o parecer do arquiteto. Deve encontrar-se estruturada, facilitando assim a sua consulta, utilização e preservação com o objetivo de manter a longevidade da informação.

No modelo relacional são definidas as entidades a usar para cada tabela e para cada atributo que resultam em diferentes domínios. O objetivo é conseguir definir os domínios de modo a ter os atributos bem estruturados. (Coelho, 2011)

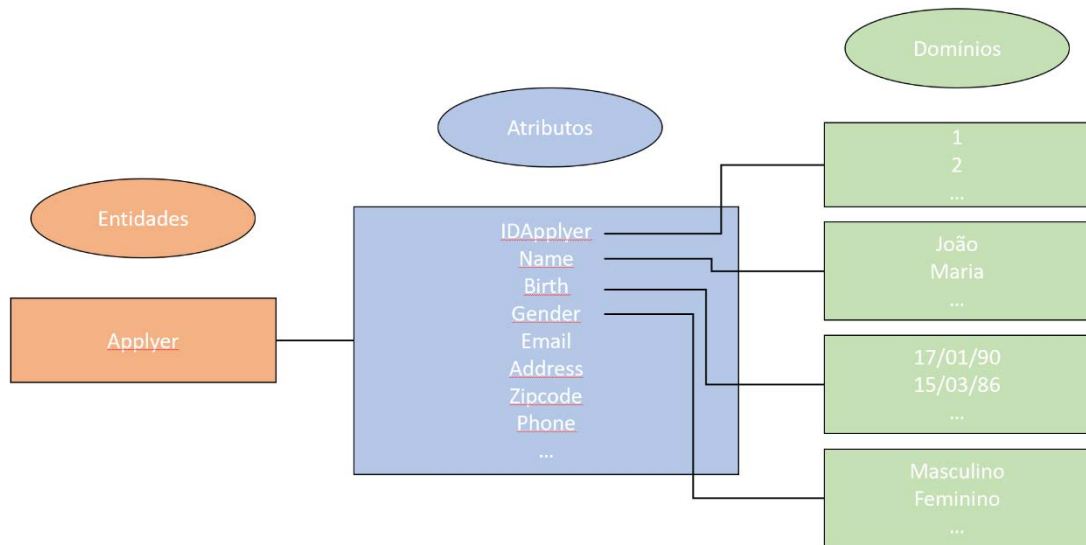
Uma vez analisado o problema é construído o Diagrama de Entidade e Relação (DER) que representa o sistema de informação e, em seguida, procede-se à criação no Sistema de Gestão de Bases de Dados. (Al-Masree, 2015)

Os passos necessários dar início ao processo são:

##### **A Criação das Tabelas**

De forma a implementar cada entidade é criada uma tabela de acordo com as seguintes indicações:

- Dar à tabela o nome da entidade;
- Incluir na tabela um campo por cada atributo associado à entidade;
- Não misturar atributos de diferentes entidades;
- Definir o campo ou campos que serão a chave primária e/ou as chaves estrangeiras;
- Por vezes é necessário criar tabelas intermédias de conexão, apenas constituídas por chaves privadas de diferentes entidades e algum atributo inerente.



**Figura 10 – Constituição de Tabelas Relacionais**

Adaptado de < <https://taniaferreira.wordpress.com/entidades-relacionamento-atributos-e-dominios/> >

## As Relações entre tabelas

De forma a implementar uma Relação (1) / (1) é necessário:

Incluir numa das tabelas envolvidas a relação de um campo que contenha como atributo o mesmo nome que a chave de outra tabela.

De forma a implementar uma Relação (1) / (\*) é necessário:

A tabela que tiver o \* é a tabela que irá conter um campo como valor chave de outra tabela.

De forma a implementar uma Relação (\*) / (\*) é necessário:

Criar uma tabela, com o nome dessa união, com o mínimo de dois campos, um para cada chave das duas tabelas envolvidas, juntamente com os atributos adicionais necessários para a estrutura dessa relação.

## Criação de Chaves

### Chaves Primárias

As chaves primárias (*Primary Key*) são criadas com o objetivo de identificar uma entidade de forma única, enquanto criam relações entre várias tabelas que tenham atributos comuns. Por este motivo, uma chave primária tem de ser única e não pode ter registos nulos. A maioria das vezes é automaticamente incrementada. Cada tabela só pode assim ter uma Chave Primária que, geralmente, é o seu ID.

### Chave Estrangeira

Uma chave estrangeira (*Foreign Key*) tem como objetivo o relacionamento entre tabelas. O nome de chave estrangeira surge do resultado de ser praticamente uma chave primária de outra tabela mas com o propósito de as conectar.

A Chave Estrangeira tem ainda outros propósitos, como por exemplo:

- A impossibilidade de adicionar valores inválidos como um ID de uma tabela;
- A impossibilidade que seja removido indevidamente um registo pela segurança de estar ligado a outra tabela. Será sempre necessário ser removida primeiro na tabela original.

Com isto, chegamos à conclusão que uma tabela pode sempre ter mais do que uma chave estrangeira, ao contrário das chaves primárias. (Coelho, 2011)

Significado das chaves:

Chave Primária: **Identificação**

Chave Estrangeira: **Relacionamento**

## A Criação de Ligações

Uma vez criadas todas as tabelas, através dos princípios enunciados anteriormente, procedem-se à ligação entre eles, usando os atributos criados para o efeito.



Uma vez que cada item de dados é registado apenas uma vez, numa tabela, existe uma menor possibilidade de uma inconsistência. A eficiência das ligações é registar os dados num só local, o que significa que se utiliza menos espaço de armazenamento. Além disto, as tabelas mais pequenas tendem a fornecer dados mais rapidamente do que as tabelas maiores. Por fim, a má utilização das tabelas, e não conseguir uma separação das mesmas para assuntos diferentes, pode originar valores nulos (se existir uma ausência de dados). Esta redundância nas tabelas dificulta assim o desempenho. Para um melhor entendimento, a estrutura de uma base de dados é mais fácil de compreender se os assuntos estiverem adequadamente separados em tabelas. (Coelho, 2011)

### **Modelo Entidade Relação (MER)**

Este modelo Entidade e Relacionamento é utilizado na Engenharia de Software com o objetivo de descrever os objetos (entidades) que refletem o domínio de negócios em que estão inseridos, as características (atributos) e como se relacionam entre si (ligações/relacionamentos).

Num modo geral, este modelo concetual representa de forma abstrata a estrutura que possuirá o base de dados. Toda a base de dados irá conter várias entidades, vários atributos, tabelas intermédias, chaves primárias e estrangeiras. (Al-Masree, 2015)

Este modelo é construído através dos seguintes passos:

Em primeiro, especificar o Sistema de Informação a construir:

O Modelo Entidade Relação ou Modelo Entidade Associação.

Em segundo, em conformidade com o modelo implementado de SI:

Criar o Modelo Físico, ou seja, exemplificar a estrutura da Base de Dados Relacional.

#### **4.6. Estrutura da Base de Dados**

Serão apresentados os objetos criados e utilizados do sistema “JobRadar”.

Objectos/Entidades: Applyer, Provider, Jobs, Announce, Annoucefavs, Categories, Applyercategories, Attachments, Applications.

Todos estes objetos são entidades do Sistema de Informação, formam um conjunto de dados e cada unidade de informação é independente, podendo estar ou não relacionada com outras. Assim, é benévola a solução construída neste sistema pela separação e projeção das entidades.

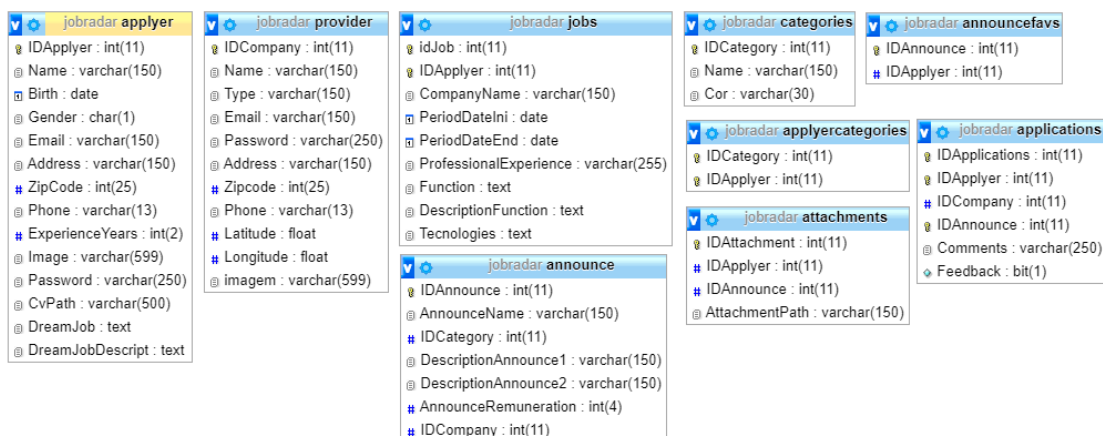


Figura 11 – Estrutura Base de Dados JobRadar

## 4.7. Base de dados JobRadar (MER)

### Modelo Entidade e Relação JobRadar

Implementadas as nove Entidades a baixo identificadas, formam-se as nove tabelas que completam a base de dados do JobRadar.

As colunas e os seus campos resumem os atributos escolhidos que constituem cada entidade, demonstrando como cada uma é constituída. Como referência, está demonstrada a estrutura da base de dados e modelo lógico criada com o Modelo Entidade Relação (MER) apresentando as interligações da aplicação “JobRadar”.

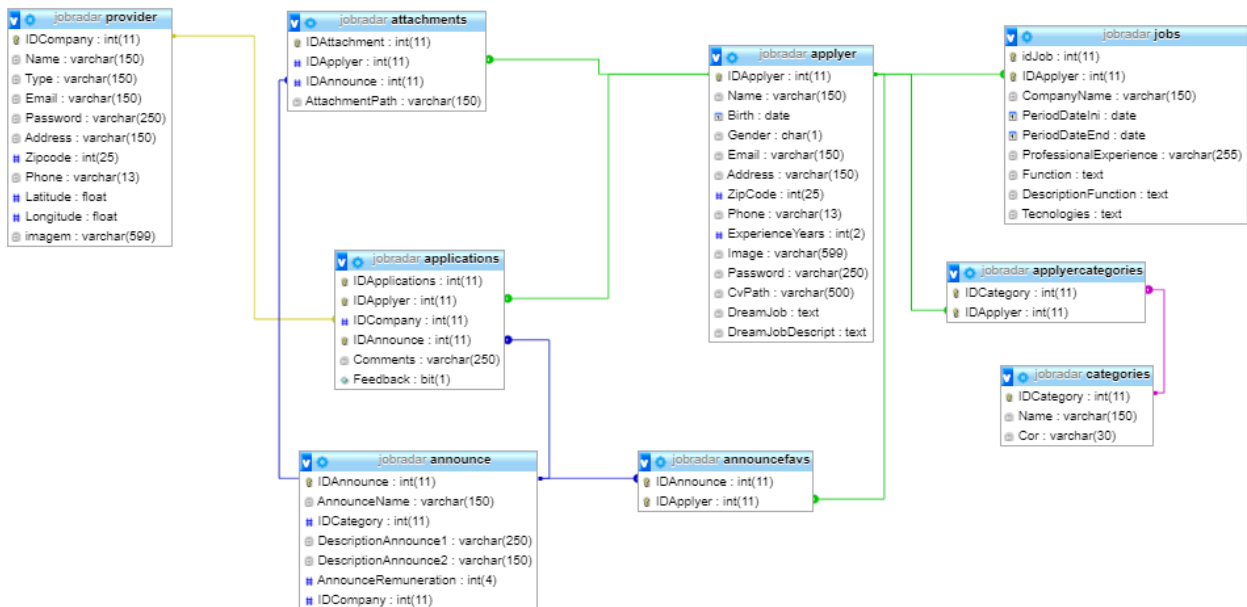


Figura 12 – Modelo Entidade e Relação JobRadar

As bases de dados juntamente com todas as entidades criadas estão descritas com todas as suas características. Foram criadas tabelas ilustrativas dos dados, utilizando o significado que tem no sistema, os seus atributos e das tabelas relacionadas. Encontram-se abaixo as suas apresentações referentes ao detalhe de cada entidade:

**Tabela 9 – Tabela da Base de dados Applyer**

Tabela Applyer					
Nome	Chaves	Tipo de variável	Tamanho	Nulo	Descrição
IDApplyer	Primária	integer	11	Não	Variável com auto-incremento, ID do candidato
Name		varchar	150	Não	Nome do candidato
Birth		date		Não	Data de nascimento
Gender		char	9	Não	Género Feminino / Masculino
Email		varchar	150	Não	Email de registo
Address		varchar	150	Não	Morada
ZipCode		integer	25	Não	Código Postal
Phone		varchar	13	Não	Telefone
ExperienceYears		integer	2	Não	Anos de experiência
Image		varchar	599	Sim	Fotografia Candidato (Opcional)
Password		varchar	250	Não	Senha pessoal
CvPath		varchar	500	Não	Diretoria do CV
DreamJob		text		Não	Trabalho de sonho
DreamJobDescript		text		Não	Descrição de trabalho de sonho

Tabela 9 referente à entidade Applyer, onde são demonstra todas as entidades, atributos e domínios.

**Tabela 10 - Tabela da Base de dados Provider**

Tabela Provider					
Nome	Chaves	Tipo de variável	Tamanho	Nulo	Descrição
IDCompany	Primária	integer	11	Não	Variável com auto-incremento, ID de empresa
Name		varchar	150	Não	Nome da empresa
Type		varchar	150	Não	Tipo de empresa
Email		varchar	150	Não	Email de registo
Password		varchar	250	Não	Senha pessoal
Address		varchar	150	Não	Morada
ZipCode		integer	25	Não	Código Postal
Phone		varchar	13	Não	Telefone
Latitude		float		Sim	Coordenada GPS
Longitude		float		Sim	Coordenada GPS
Imagem		varchar	599	Não	Fotografia empresa (Opcional)

Tabela 10 referente à entidade Provider, onde são demonstradas todas as entidades, atributos e domínios.

**Tabela 11 - Tabela da Base de dados Jobs**

Tabela Jobs					
Nome	Chaves	Tipo de variável	Tamanho	Nulo	Descrição
IDJob	Primária	integer	11	Não	Variável com auto-incremento, ID do cargo
IDApplyer	Secundária	varchar	11	Não	Nome do candidato
CompanyName		varchar	150	Não	Nome da empresa
PeriodDateIni		date		Não	Data de início da experiência
PeriodDateEnd		date		Não	Data do fim da experiência
ProfessionalExperience		varchar	255	Sim	Descrição da experiência
Function		text		Não	Função desempenhada
DescriptionFunction		text		Não	Descrição da função
Tecnologies		text		Não	Tecnologias trabalhadas

Tabela 11 referente à entidade Jobs, onde são demonstradas todas as entidades, atributos e domínios.

**Tabela 12 - Tabela da Base de dados Announce**

Tabela Announce					
Nome	Chaves	Tipo de variável	Tamanho	Nulo	Descrição
IDAnnounce	Primária	integer	11	Não	Variável com auto-incremento, ID do anúncio
AnnounceName		varchar	150	Não	Nome do anúncio
IDCategory		integer	11	Não	ID da categoria
DescriptionAnnounce1		varchar	250	Não	1º Descrição anúncio
DescriptionAnnounce2		varchar	150	Não	2º Descrição anúncio
AnnounceRemuneration		integer	4	Não	Remuneração
IDCompany		integer	11	Não	ID da empresa

Tabela 12 referente à entidade Announce, onde são demonstradas todas as entidades, atributos e domínios.

**Tabela 13 - Tabela da Base de dados AnnounceFavs**

Tabela AnnounceFavs					
Nome	Chaves	Tipo de variável	Tamanho	Nulo	Descrição
IDAnnounce	Primária	integer	11	Não	Variável com auto-incremento, ID do anúncio
IDApplyer	Primária	integer	11	Não	Variável com auto-incremento, ID do candidato

Tabela 13 referente à entidade Announcefavs, onde são demonstradas todas as entidades, atributos e domínios.

**Tabela 14 - Tabela da Base de dados Categories**

Tabela Categories					
Nome	Chaves	Tipo de variável	Tamanho	Nulo	Descrição
IDCategory	Primária	integer	11	Não	Variável com auto incremento, ID do anúncio
Name		varchar	150	Sim	Nome da categoria
Cor		varchar	30	Não	Cor respectiva

Tabela 14 referente à entidade Categories, onde são demonstradas todas as entidades, atributos e domínios.

**Tabela 15 - Tabela da Base de dados Applyercategories**

Tabela Applyercategories					
Nome	Chaves	Tipo de variável	Tamanho	Nulo	Descrição
IDCategory	Primária	integer	11	Não	Variável com auto-incremento, ID da categoria
IDApplyer	Secundária	integer	11	Não	Variável com auto-incremento, ID do candidato

Tabela 15 referente à entidade Applyercategories, onde são demonstradas todas as entidades, atributos e domínios.

**Tabela 16 - Tabela da Base de dados Attachments**

Tabela Attachments					
Nome	Chaves	Tipo de variável	Tamanho	Nulo	Descrição
IDAttachment	Primária	integer	11	Não	Variável com auto-incremento, ID do CV/Upload
IDApplyer	Secundária	integer	11	Não	Variável com auto-incremento, ID do candidato
IDAnnounce	Secundária	integer	11	Não	Variável com auto-incremento, ID do anúncio
AttachmentPath		varchar	150	Não	Diretoria do armazenamento do CV

Tabela 16 referente à entidade Attachments, onde são demonstradas todas as entidades, atributos e domínios.

**Tabela 17- Tabela da Base de dados Applications**

Tabela Applications					
Nome	Chaves	Tipo de variável	Tamanho	Nulo	Descrição
IDApplications	Primária	integer	11	Não	Variável com auto-incremento, ID da candidatura
IDApplyer	Primária	integer	11	Não	Variável com auto-incremento, ID do candidato
IDCompany	Secundária	integer	11	Não	Variável com auto-incremento, ID de empresa
IDAnnounce	Primária	integer	11	Não	Variável com auto-incremento, ID do anúncio
Comments		varchar	250	Sim	Comentários sobre o candidato
Feedback		bit	1	Sim	Feedback

Tabela 17 referente à entidade Applications, onde são demonstradas todas as entidades, atributos e domínios.

## 5. Protótipo do Sistema

### 5.1. Introdução

O protótipo criado foi o primeiro exemplar do modelo. Com este projeto foi possível construir um conceito mais sólido, eliminando imperfeições e melhorando a cada teste executado.

As versões criadas anteriores ao protótipo visavam testar o produto, serviço ou sistema. Um protótipo é construído na fase de testes de um projeto.

### 5.2. Descrição Geral

#### 5.2.1 Servidor

O Apache é um servidor Web OpenSource que contempla quase metade dos sites, de hoje em dia, hospedados na internet. Este servidor permite disponibilizar qualquer conteúdo. É responsável e age como um intermediário. Disponibiliza conteúdo entre o servidor e as máquinas dos visitantes e utilizadores enquadrando-se na estrutura cliente-servidor.

O Servidor utilizado foi o Apache que permite que todo o site esteja localizado remoto no computador onde foi instalado e desenvolvido. A aplicação JobRadar está instalada num localhost mas passará para um webhost quando for publicada e estiver online. Uma das grandes razões para a escolha do Apache foi por ter incluída a melhor taxa de processamento PHP.

## Index of /

<a href="#">Name</a>	<a href="#">Last modified</a>	<a href="#">Size</a>	<a href="#">Description</a>
 <a href="#">jobradar/</a>	2020-10-30 01:58	-	

*Apache/2.4.25 (Win32) OpenSSL/1.0.2j PHP/5.6.30 Server at localhost Port 80*

Figura 13 – Diretória local Apache - JobRadar



## 5.2.2 Administrador de base de dados

Em termos de administrador de base de dados o Php MyAdmin foi o administrador seleccionado pois possibilita a execução de uma série de edições e tarefas de administração de Bases de Dados.

Por ser um compilador da Base de Dados SQL permite criar e remover bases de dados. Permite também alterar, criar, remover tabelas, editar, inserir e remover campos, como também executar códigos SQL manipulando campos, chaves primárias e chaves secundárias utilizando em paralelo a linguagem PHP.

Ao longo do desenvolvimento da aplicação, o PHP torna mais benévolo e funcional a interação com o script e as suas necessidades. Assim, acaba ser uma ferramenta de suporte à criação e ao acesso simplificado a bancos de dados MySQL e de aplicações.<sup>6</sup>

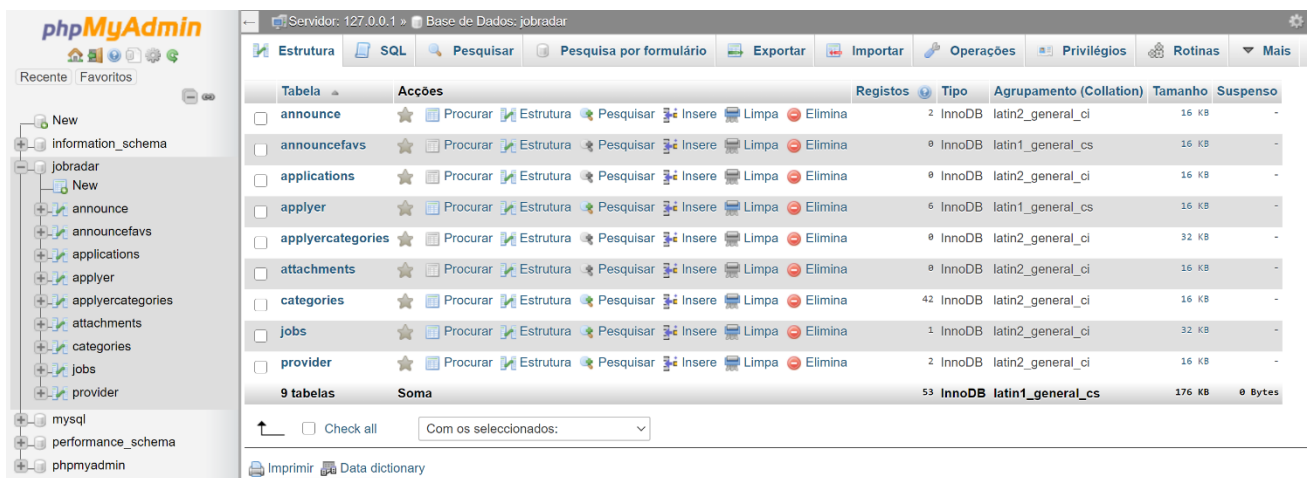
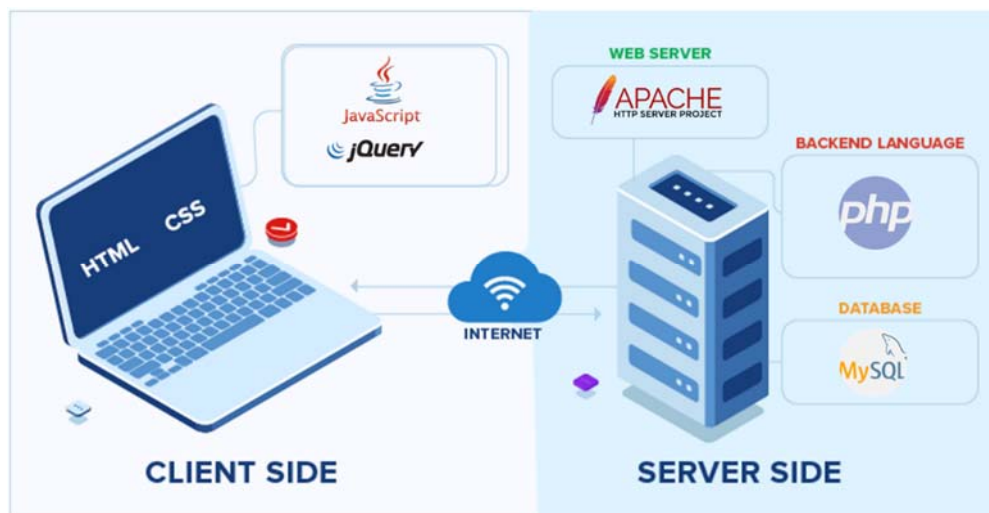


Figura 14 – Php My Admin JobRadar

<sup>6</sup> <https://rockcontent.com/br/blog/phpmyadmin/>

### 5.2.3 Front-end e Back-end

No que diz respeito ao Front End e Back End, os mesmos podem ser separados da seguinte forma no JobRadar:



**Figura 15 – Esquema Back End e Front End JobRadar**

Adaptado de <<https://www.ironhack.com/br/desenvolvimento-web/front-end-x-back-end-qual-e-a-diferenca>>

#### Front-end

O front-end é todo o código da aplicação responsável pela apresentação e parte visual como o HTML, o CSS e o JavaScript, além das frameworks e bibliotecas.

Geralmente, é o código que corre nos browsers sempre que é aberto um site ou uma aplicação web, sendo sempre mais conhecido por *client-side*.

#### Back-end

O back-end, habitualmente, significa trabalhar diretamente com bancos de dados como o MySQL, acessos, segurança e escalabilidade, servidores complexos e aplicativos web.

O back-end é o software que corre do lado do servidor onde se insere o Apache, por essa mesma razão é conhecido como *server-side*, utilizando diversas linguagens de programação. Neste âmbito apenas foi utilizado o PHP como recurso.

#### 5.2.4 Mockup

Foi criado um Mockup, no site Axure, em modelo de escala referente ao protótipo JobRadar.

Link Mockup JobRadar: <https://5ghs1u.axshare.com>

Ao entrar no link, o primeiro passo é alterar a escala de “Default scale” para “Scale to Width” para ser possível observar todo o seu conteúdo.

O programa Axure tem algumas limitações e, por esta razão, não nos é permitido recorrer à função de scroll down da navigation bar, como no protótipo.

No canto superior esquerdo, o botão “Log In” redirecciona-nos, apenas com um clique, para a página onde nos é possível inserir as nossas credenciais de utilizador.

Na página “Log In” existem três hiperligações, o “Ainda não se registou?”, o “Entrar” e o “Voltar atrás”.

O CTA “Ainda não se registou?” reencaminha-nos para a página onde é possível fazermos o primeiro registo, ou como Candidatos, ou como Empresa.

Como não é possível validar os emails das contas registadas, por ser apenas um mockup e não estar conectado com a base de dados, não é possível distinguir se é um candidato ou uma empresa. Assim sendo, o Axure assume por defeito, ao clicarmos no botão enter, a abertura da Homepage do candidato.

Na secção “Entrevistas” e “Perfil” ao clicarmos nas imagens existentes, as mesmas redireccionam-nos para as páginas correspondentes. Em todas estas páginas existe um botão designado com a letra “X”, no canto superior direito, que nos permite sair ou regressar à página anterior. No final de cada página existe também o botão “Voltar” que permite ao utilizador realizar as mesmas ações.

Por fim, existe então o botão “LogOut”, na Home do Candidato, situado no canto superior direito, onde o utilizador pode terminar a sua sessão.

### 5.2.5 Esquema de Navegação

O Esquema de Navegação, ou também conhecido por esquema navegacional, permite visualizar todas as páginas criadas no mockup e, por consequência, no protótipo. Como referido no capítulo 5.2.4, existe um link onde nos é permitido entrar e visualizar o estudo das páginas ilustrativas à aplicação JobRadar, como também uma breve explicação de como navegar no mesmo.

O Mockup tem como intenção recriar e representar a disposição e o design de todos os botões e features projectadas, em forma de modelo à escala, para o protótipo.

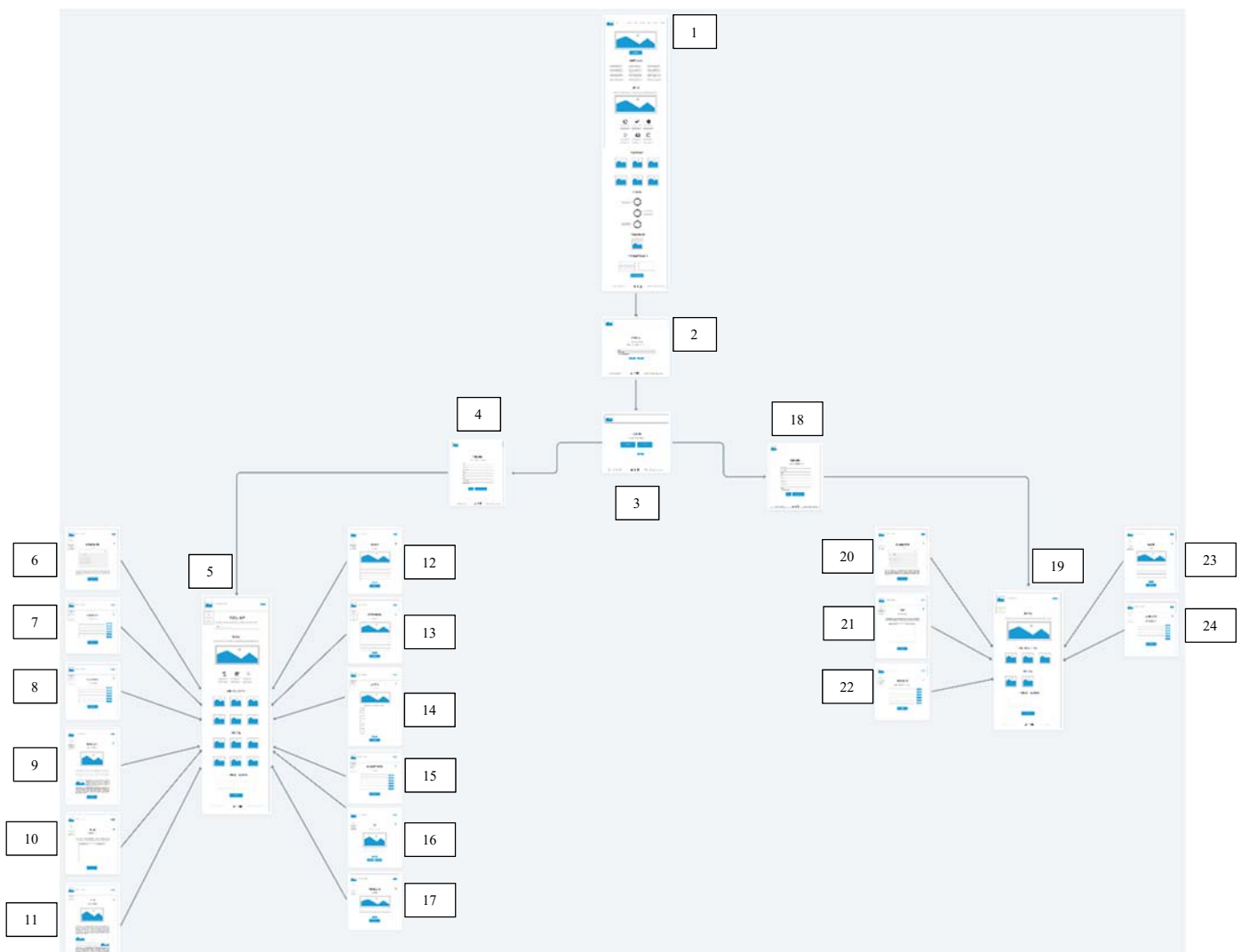


Figura 16 – Esquema de Navegação JobRadar

## Legenda do Esquema Navegacional

1. Página Index
2. Página LogIn
3. Página Novo Utilizador (Candidato/Empresa)
4. Página Preenchimento de informação Candidato
5. Página Home Candidato
6. Página Agendamentos/Entrevistas
7. Página FeedBack
8. Página Favoritos
9. Página Oportunidades de Mercado
10. Página Chat
11. Páginas Dicas
12. Página Dados Pessoais
13. Página Experiência Profissional
14. Página Áreas de Interesse
15. Página Candidaturas Enviadas
16. Página Upload CV
17. Página Trabalho de Sonho
18. Página Preenchimento de informação Empresa
19. Página Home Empresa
20. Página Agendamentos/Entrevistas
21. Página Chat
22. Página FeedBack
23. Página Dado da Empresa
24. Página Anúncios Disponibilizados

## 6. Interface com o Utilizador

PrintScreens do JobRadar e explicação dos botões e funcionalidades.

A imagem relativa à figura em baixo apresenta-nos a página Inicial do “JobRadar”. Todo o conteúdo desta página pode ser visualizado por um utilizador anónimo, isto é, um utilizador que ainda não tenha feito o seu registo no site.



Figura 17 – Página Index JobRadar

1. O botão JobRadar possibilita o Scroll-Up para o topo da página, sempre que o utilizador se encontre em qualquer uma das seções do ponto 3.
2. O botão “Log In” permite ao utilizador realizar o seu login ou, caso ainda não se tenha registado, criar uma nova conta.
3. A barra de menu permite ao utilizador saltar para qualquer uma das seções que pretender instantaneamente.
4. O botão CTA “Ver mais” permite ao utilizador fazer o Scroll-Down para a primeira seção da *navigation bar*, “Serviços”.



Figura 18 – Página de UserIn JobRadar

1. As duas caixas de texto permitem que o utilizador insira o seu Email e respetiva Password, caso a sua conta já esteja previamente criada.
2. No caso do utilizador ainda não estar registado, pressionando a frase “Ainda não se registou?” será, automaticamente, encaminhado para a página “Userin” de forma a poder completar o seu registo, escolhendo o tipo de conta que pretender.
3. O botão “Entrar” permitirá ao utilizador entrar na sua página de acesso após a validação das suas credenciais. Caso não tenha preenchido o email ou a password aparecerá uma mensagem “Preencha este campo”, se preencher os dois e a conta não existir aparecerá uma mensagem “A conta não existe”.



Figura 19 – LogIn1



Figura 20 – LogIn2

4. O botão “Back” ao ser pressionado redireciona automaticamente o utilizador para a página anterior “Index”.





Figura 19 – Página Usercomp JobRadar

1. O botão "Candidato" fará sentido para quem procura emprego. Ao pressionar o botão mencionado, automaticamente, será redirecionado para a página com o formulário a preencher de novo utilizador- Candidato. Ao preencher todos os dados e pressionar OK receberá uma mensagem de confirmação e terá de efetuar o Login.

Figura 21 – Registo Empresa

2. O botão "Provedor/Serviço" fará sentido a alguma entidade que pretenda disponibilizar anúncios de emprego. Ao pressionar o botão, automaticamente, será redirecionado para a página com o formulário a preencher para um novo utilizador- Provedor/ Serviço. Ao preencher todos os dados e pressionar OK receberá uma mensagem de confirmação e terá de efetuar o Login.

3. O botão ao ser pressionado automaticamente voltará para a página anterior "UserIn".

Figura 20 – Registo Candidato



Figura 22 – Página Candidato JobRadar

1. O botão JobRadar possibilita o Scroll-Up para a o topo da página, sempre que o utilizador se encontre em qualquer uma das secções referidas no ponto 2.
2. A tabela lateral permite ao utilizador saltar automaticamente para qualquer secção, instantaneamente, sem ter de percorrer a página até à secção pretendida.
3. A barra de pesquisa permite ao utilizador pesquisar anúncios, através de palavras chave, como o nome da função pretendida ou nome da empresa, como representado na figura 23, abaixo.
4. O Botão “Logout” permite terminar a sessão e regressar à página Index como utilizador anónimo.



Figura 23 – Motor de Pesquisa - Candidato JobRadar

1. Neste caso, ao pesquisarmos a palavra “Informática” surgem todos os anúncios com a palavra inserida, seja no nome do Anúncio, seja no nome da Empresa ou na sua Descrição.
2. O botão “Guardar Favoritos” permite ao utilizador adicionar o anúncio à sua lista pessoal de Favoritos como visível na figura 28.
3. O botão “Enviar Candidatura” permite ao utilizador submeter os seus dados pessoais e o seu CV (previamente inserido).

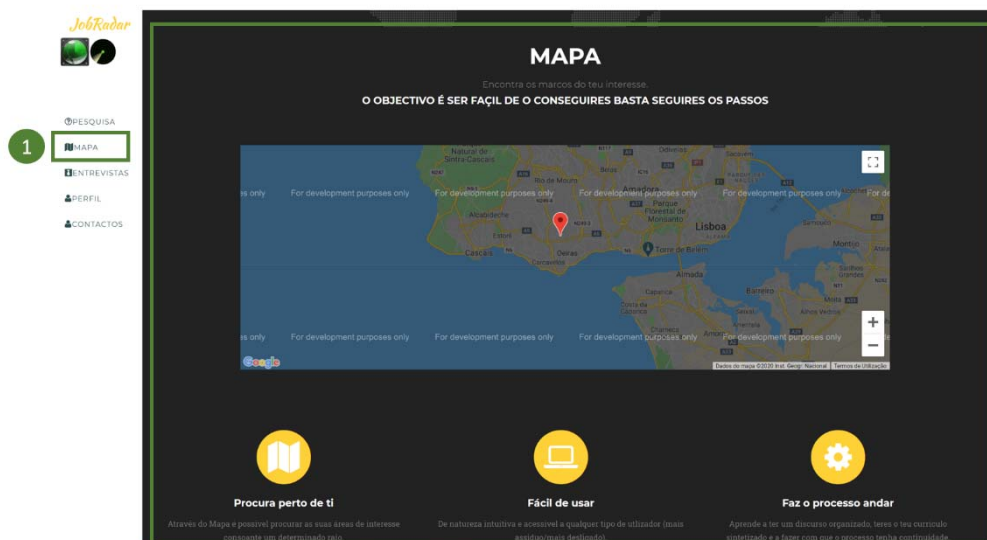


Figura 24 – Secção Mapa - Candidato JobRadar

1. A Secção Mapa é o espaço reservado para o utilizador poder fazer a sua pesquisa de anúncios de emprego uma forma mais interativa e intuitiva, adaptada às novas tecnologias. Esta funcionalidade faz parte das melhorias futuras propostas a implementar.



Figura 25 – Seção Entrevistas - Candidato JobRadar

1. A Seção Entrevistas é o espaço reservado para o utilizador poder fazer a gestão das suas entrevistas.

2. O Calendário pertence às *features* a serem implementadas no futuro. Terá como objetivo permitir ao candidato a verificação dos seus eventos, como os dias e horas das entrevistas agendadas. Representado na figura 26.

3. O Feedback é o espaço onde o candidato pode verificar os comentários que foram feitos às suas candidaturas. Assim, o candidato pode obter a verificação se o CV foi considerado válido para uma entrevista futura. Representado na figura 27.

4. Os Favoritos permitem ao candidato ver a lista pessoal de anúncios de trabalho guardados. Representado na figura 28.

5. O Mercado é o espaço que contém oportunidades escolhidas e tidas como pertinentes para qualquer candidato. Disponibilizando notícias, tendências, curiosidades e oportunidades que se destacaram ultimamente. Representado na figura 29.

6. O Chat pertence às *features* a serem implementadas no futuro. Terá como objetivo criar um mecanismo de comunicação entre os candidatos e as empresas. Representado na figura 26.

7. As Dicas é o espaço onde se disponibilizam conselhos, guidelines, perguntas e respostas em como agir, que postura ter. Este espaço tem como objetivo preparar os candidatos para entrevistas de sucesso. Representado na figura 30.

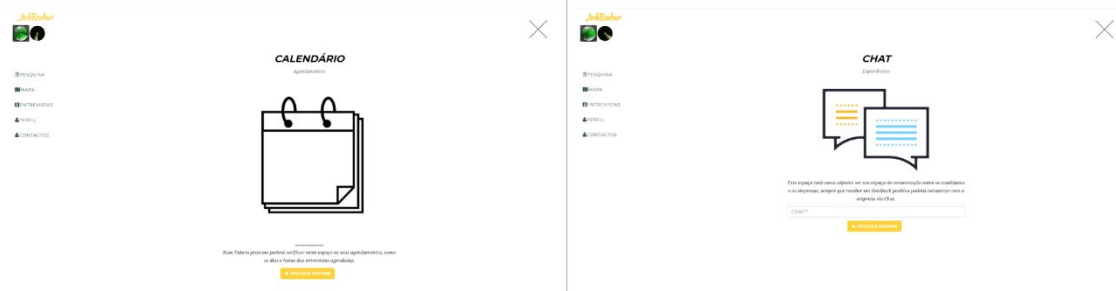


Figura 26 – Features Futuras Candidato JobRadar

Na figura 26 estão representados os dois espaços onde irão estar disponibilizadas as duas *features* planejadas para o futuro.

**CANDIDATURAS**  
Feedback Positivo

1

Anúncio	Empresa	Descrição	Remuneração	Comentários	Aceite
Consultor analista de sistemas de gestao	Samsung	em áreas como a Informática, Gestao, Engenharia Sistemas	1100	Enquadra-se na vaga candidatada.	Sim
Consultor de Infraestruturas Microsoft	Microsoft	Experiencia profissional mínima de 3 anos na área de business intelligence Experiencia mínima de 3 anos em administração de Sistemas Microsoft: Windows Server, Exchange Server, Active Directory, Hyper-V, SQL Server, Office 365, SCCM; Licenciatura na área das Tecnologias de Informática	900	Não tem os requisitos pretendidos.	Não

2

**FECHAR PÁGINA**

1. A tabela do Feedback é constituída pelo Nome do Anúncio, Nome da Empresa, as Descrições das Notícias, a Remuneração e a coluna Aceite (onde o candidato pode verificar se a candidatura foi aceite ou não).

Desta forma, sempre que uma candidatura for submetida é automaticamente enviada para a empresa que disponibilizou o anúncio de emprego.

As candidaturas têm três estados, “Pendente”, “Aceite” e “Não Aceite”.


Sempre que uma candidatura for revista, o seu estado passa de “Pendente” para “Aceite” ou “Não Aceite”. O Feedback está interligado com o motor de busca, representado na Figura 23.

Sempre que uma candidatura estiver no estado “Pendente” estará representada na área das Candidaturas Pendentes. (Figura 35).

2. O Botão “Fechar Página” permite sair e voltar à página de perfil do candidato.

Figura 27 – Feedback Candidato JobRadar

**FAVORITOS**  
Lista Pessoal



Neste menu podes consultar todos as vagas que guardas-te nos teus favoritos

Anuncio	Empresa	Descrição	Remuneração	
Designer, Gráfico Ou de Comunicação e Multimédia (M/F)	Samsung	<p>académico concluído há, no máximo, 2 anos - Domínio da língua inglesa (avançado) - Conhecimentos de informática na ótica do utilizador, incluindo Word, Excel e PPT</p> <p>12 ano num Curso Informática - Nível 4 (Preferencial)</p>	800	APAGAR
Gestor de Cliente - Gestor de Produto	Nokia	<p>Desenvolver relações de confiança com clientes institucionais, adquirindo um conhecimento aprofundado sobre o seu contexto e necessidades, para criar oportunidades de crescimento de negócio;</p> <p>Formação académica superior em Gestao ou similar; Experiencia profissional no setor das tecnologias de informacao entre 3 a 7 anos;</p>	1000	APAGAR

Figura 28 – Favoritos Candidato JobRadar

1. A tabela dos Favoritos é constituída pelo Nome do Anúncio, Nome da Empresa, as Descrições das Notícias e a Remuneração.

A tabela Favoritos está interligada com o motor de busca, representado na Figura 23.

2. O Botão “Apagar” retira o anúncio da lista do Candidato.

3. A página também conta com o botão “Fechar página” que redirecionará o utilizador para a página de perfil do candidato.

## MERCADO

Oportunidades



Conhece as tendências do mercado laboral em Portugal

Descobre as tendências do mercado laboral em ano de pandemia e alguns dados relevantes, de 2019, em diversas áreas e setores empresariais.

A melhor forma de te adaptares ao mercado de trabalho em Portugal é ficares a conhecê-lo.

Através de um estudo feito pela Hays, *Guia do Mercado Laboral*, podemos tirar algumas conclusões rápidas: os trabalhadores do sector do Retail, Engenharia, Office Support, Contabilidade e Finanças, IT e Turismo são os que tem maior disponibilidade para a mudança, quer isto dizer que os colaboradores destes sectores revelaram um maior interesse em mudar de área. As razões que se destacam são, sobretudo, insatisfação salarial e poucas perspetivas de progressão de carreira.

As vagas que existem com maior oferta de empregabilidade são: Área Comercial, IT, Engenharia, Marketing e Comunicação, Logística e Finanças. As competências mais desejadas por todas as empresas são também unânimes, entre elas, Iniciativa e Inteligência emocional destacam-se. Não nos podemos esquecer que entrámos numa nova era laboral, pós Covid-19, o que obriga a que muitas das tuas soft skills passem também por uma consolidação clara do teletrabalho.

Atualiza-te, sempre que possível, através destas notícias.

Aqui, podes encontrar toda a atualidade, em notícias, sobre o mercado de trabalho em Portugal.

### **COVID-19. NORTE COM MAIS 28.685 DESEMPREGADOS INSCRITOS COMPARATIVAMENTE A FEVEREIRO**

Mais de metade dos inscritos estão registados há menos de um ano no Instituto de Emprego e Formação profissional. Dos 153.022 pessoas inscritas no desemprego, 58% são mulheres.

### **JOBSHOP'20 DIGITAL APRESENTA MAIS DE 400 OFERTAS DE EMPREGO E ESTÁGIOS**

A Faculdade de Direito da Católica organiza, nos dias 25 e 26 de novembro, a 18ª edição do JobShop, o maior evento de empregabilidade na área do Direito em Portugal. Este ano num formato 100% digital, o JobShop mantém o dinamismo das edições anteriores.

1

Figura 29 – Mercado Candidato JobRadar

O Mercado é o espaço que contém oportunidades selecionadas e tidas como pertinentes para qualquer candidato.

Disponibilizando notícias atuais, tendências, curiosidades e oportunidades.

1. São escolhidas notícias específicas sobre o mercado de trabalho em Portugal, de modo a que o candidato se mantenha atualizado sobre o que se passa.



## DICAS

Aprende a conquistar nas entrevistas



Preparámos quatro dicas/ perguntas mais frequentes que te podem fazer numa entrevista! Estás pronto para excederes as expectativas do recrutador?

### FALA SOBRE TI

Muitas vezes o recrutador vai gular a tua entrevista com início nesta afirmação. Trata-se de um convite para partilhares com ele as tuas experiências. Podes falar sobre os teus hobbies, sobre histórias que achas que fazem sentido e resumir os aspetos que definiram a tua carreira profissional. Tens de ser direto e objetivo. Salienta sempre os aspetos que te podem valorizar profissionalmente.

### O QUE É SABES SOBRE A EMPRESA? O QUE É QUE TE LEVA A QUERER TRABALHAR AQUI?

Aqui o teu trabalho de casa vai ser posto à prova. Antes de ires para uma entrevista de emprego, tens de fazer uma pesquisa avançada sobre a empresa e como a mesma pode ser um benefício na tua carreira e para todos os profissionais. Consulta o site institucional, pesquisa sobre a presença da empresa nas redes sociais, etc. Uma coisa imprescindível quando fores responder a esta pergunta: Evidência conhecimento e confiança. Podes ainda acrescentar que os teus objetivos profissionais vão de encontro à filosofia do projeto.

### QUAL É A EXPECTATIVA SALARIAL?

Tens de responder cuidadosamente a esta pergunta, é um assunto delicado. Para responder da forma mais correta, informa-te quais são as remunerações praticadas no teu sector profissional. Se não quiseres dar um valor concreto, menciona um mínimo e um máximo. Podes também frisar que estás disponível a negociar os valores e outros benefícios.

### DÚVIDAS?

De uma maneira geral todas as entrevistas terminam com esta questão. Tens aqui mais uma oportunidade para mostrares todo o interesse à vaga à qual te estás a candidatar. Tenta obter mais informações sobre o empresa ou questiona sobre como é o dia a dia, por exemplo.

✖ FECHAR PÁGINA

1

Figura 30 – Dicas Candidato JobRadar

1. As Dicas é o espaço onde se disponibilizam conselhos, guidelines, perguntas e respostas em como agir, que postura ter, com o objetivo de preparar os candidatos para entrevistas de sucesso.
2. Ao pressionar o botão “Fechar página”, o utilizador fecha a janela Dicas e é redirecionado para a página de perfil do candidato.



Figura 31 – Seção Perfil - Candidato JobRadar

1. A Secção Perfil é o espaço reservado para o utilizador poder fazer a gestão dos seus dados pessoais.
2. OS Dados Pessoais é o espaço reservado para o utilizador inserir todos os seus dados. Representado na figura 32.
3. A Experiência Profissional é o espaço onde o candidato insere todos os seus dados relativos à sua última experiência profissional. Representado na figura 33.
4. As Áreas de Interesse disponibilizam uma lista de áreas laborais, permitindo ao candidato escolher as três com maior relevância para o mesmo. Tem como propósito serem utilizadas na *feature* futura do Mapa Google Maps. Representado na figura 34.
5. As candidaturas Pendentes é o espaço onde o candidato pode visualizar as candidaturas que ainda estão por analisar. Representado na figura 35.
6. O CV é o espaço onde o candidato pode fazer o Upload do seu *Curriculum Vitae*. Representado na figura 36.
7. O Trabalho de Sonho é o espaço onde o candidato pode inserir a função que sempre desejou desempenhar ou até a sua profissão de sonho. Em complemento tem também a possibilidade de poder descrever essa mesma função ou profissão. Representado na figura 37.

## DADOS

Pessoais



João Martins Lima
1990-01-17
M
joomartinslima@gmail.com
Rua Rosmaninho
2785
918307749

**2**

**3**

**Figura 32 - Dados Candidato JobRadar**

1. Neste campo é pretendido que o candidato possa inserir os seus dados pessoais ou atualizá-los.
2. O candidato ao pressionar o botão “Guardar”, guarda as alterações feitas.
3. Ao pressionar o botão “Fechar página”, o utilizador fecha a janela Dados Pessoais e é redirecionado para a página de perfil do candidato.

# EXPERIÊNCIA

Profissional



1

Nome Empresa *
Função *
Descrição*
Tecnologias *
dd/mm/aaaa <input type="calendar"/>
dd/mm/aaaa <input type="calendar"/>
O que gostou mais de fazer *

2

GUARDAR

3

\* FECHAR PÁGINA

Figura 33 – Experiência Candidato JobRadar

1. Neste campo é pretendido que o candidato possa inserir os dados da sua última experiência profissional ou atualizá-los.
2. O candidato ao pressionar o botão “Guardar” guarda as alterações feitas.
3. Ao pressionar o botão “Fechar página”, o utilizador fecha a janela Experiência Profissional e é redirecionado para a página de perfil do candidato.

## AREAS

de maior interesse



Escolhe 3 das suas áreas de maior interesse.

1

- Agricultura
- Arquitectura
- Artes
- Banca
  
- Beleza
- Call Center
- Comercial
- Comunicação Social
  
- Construção Civil
- Contabilidade
- Desporto
- Direito
  
- Educação
- Engenharia Civil
- Engenharia Electronica
- Engenharia Mecânica
  
- Engenharia Quimica
- Farmácia Biotecnologia
- Gestao de Empresas
- Gestao RH

Figura 34 – Áreas de Interesse Candidato JobRadar

1. Nas Áreas de Interesse é pretendido que o candidato possa escolher até três opções, com o intuito de serem utilizadas posteriormente na *feature* futura Mapa do Google. Desta forma, o mapa permitirá ao utilizador ter três áreas de incidência na sua procura, distinguidas com pins de cores diferentes. O utilizador também tem como opção desseleccionar e voltar a seleccionar outras áreas do seu interesse e, desta forma, atualizá-las.

2. A página também conta com o botão “Guardar” para que o candidato guarde as alterações efetuadas.

3. A página também conta com o botão “Fechar página”, que redirecionará o utilizador para a página de perfil do candidato.



Figura 35 – Candidaturas Pendentes Candidato JobRadar

1. A tabela Candidaturas Pendentes é constituída pelo Nome do Anúncio, Nome da Empresa, as Descrições das Notícias e a Remuneração. Permite visualizar todas as candidaturas que ainda não foram revistas e, por essa razão, ainda não se encontram na tabela de Feedback.

A tabela Candidaturas Pendentes está interligada com o motor de busca, representado na Figura 23.

2. O Botão “Cancelar” retira o anúncio da lista do Candidato, eliminando também a sua candidatura.

3. Ao pressionar o botão “Fechar página”, o utilizador será redirecionado para a página de perfil do candidato.

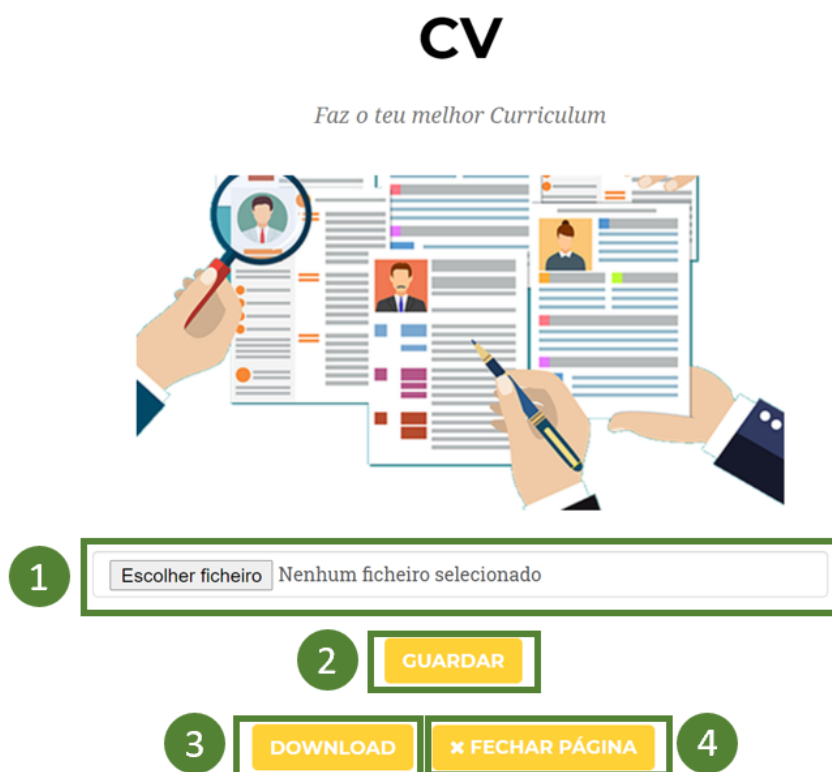


Figura 36 - CV Candidato JobRadar

1. Neste campo é pretendido que o candidato possa escolher a diretoria onde se encontra o seu *Curriculum Vitae* e fazer o upload do documento para que o mesmo possa ser enviado como anexo aos seus dados pessoais nas candidaturas a que se candidatar.
2. O candidato ao pressionar o botão “Guardar” efetua o upload e é direcionado para a página de perfil do candidato.
3. Ao pressionar o botão “Download”, o candidato efetua o download do último CV inserido na sua ficha.
4. Ao pressionar o botão “Fechar página”, o utilizador fecha a janela CV e é redirecionado para a página de perfil do candidato.

# TRABALHO

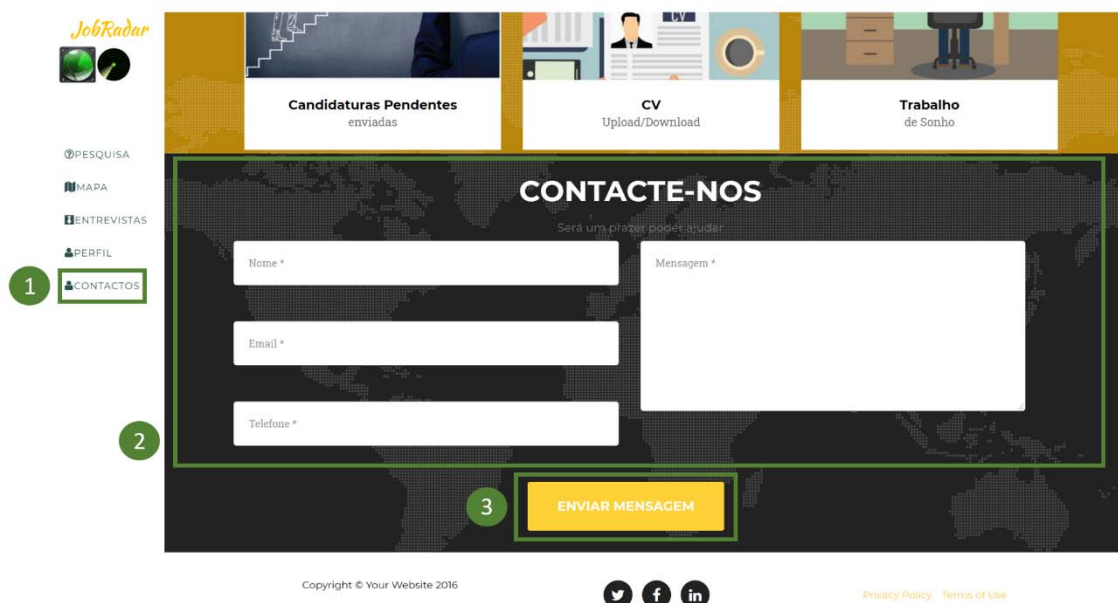
*de Sonho*



**Figura 37 – Trabalho de Sonho Candidato JobRadar**

1. Neste campo é pretendido que o candidato possa inserir a sua função ou profissão de sonho e respetiva descrição.
2. O candidato ao pressionar o botão “Guardar”, guarda as alterações feitas.
3. Ao pressionar o botão “Fechar página”, o utilizador fecha a janela Trabalho de Sonho e é redirecionado para a página de perfil do candidato.





**Figura 38 – Contacte-nos Candidato JobRadar**

1. A Secção Contactos é o espaço reservado para o utilizador poder fazer pedidos de ajuda ou esclarecimentos a cerca do aplicativo Web JobRadar.
2. Neste campo é pretendido que o candidato possa inserir os seus dados, Nome, Email e Telefone juntamente com uma mensagem descritiva do assunto em questão. Receberá um email de resposta automática, pré-criado, confirmando a receção do seu pedido.
3. Ao pressionar o botão “Enviar Mensagem”, o utilizador enviará a mensagem para o Administrador com toda a informação inserida.

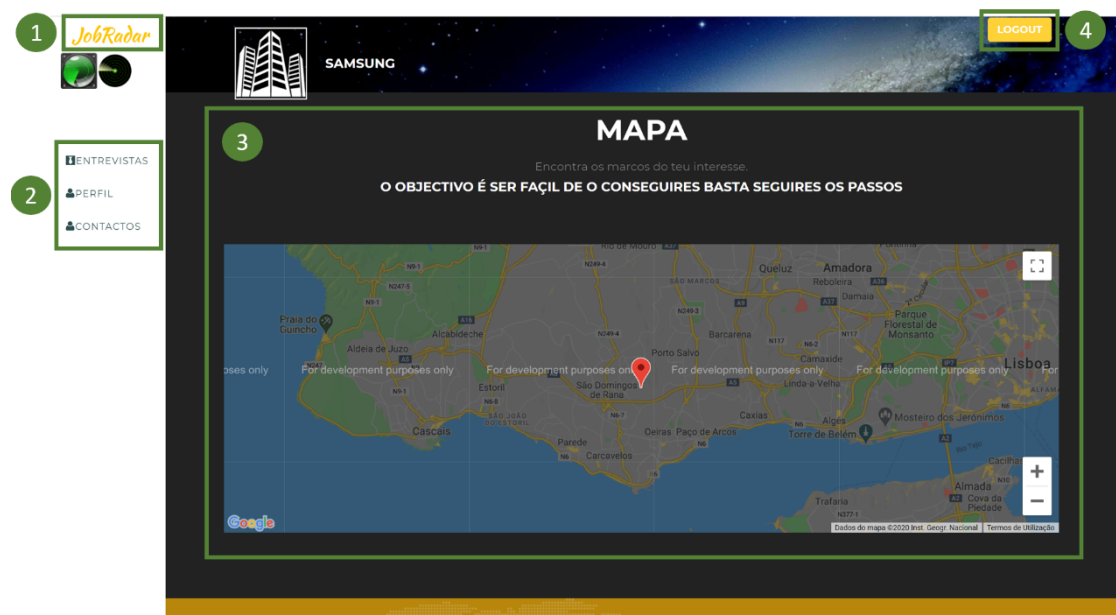


Figura 39 - Página Empresa JobRadar

1. O botão JobRadar possibilita o Scroll-Up para a o topo da página, sempre que o utilizador se encontre em qualquer uma das secções referidas no ponto 2.
2. A tabela lateral permite ao utilizador saltar automaticamente para qualquer secção, instantaneamente, sem ter de percorrer a página até à secção pretendida.
3. A secção demonstrada é o local onde o mapa será disponibilizado e a empresa poderá consultar os candidatos com preferência pela sua área de atuação.
4. O Botão “Logout” permite ao utilizador terminar a sessão e voltar para a página Index como utilizador anónimo.



Figura 40 - Perfil Empresa JobRadar

1. A Secção inclui as Entrevistas e o Perfil e o espaço é reservado para a empresa poder fazer a gestão dos seus dados, anúncios e candidaturas recebidas.
2. A Secção Entrevistas inclui o Calendário, o Chat e o Feedback das candidaturas. Representados nas figuras 41, 42.
3. A Secção Perfil inclui os Dados e os Anúncios Disponibilizados Representados nas figuras 43 e 44.



**Figura 41 - Features Futuras Empresa JobRadar**

Na figura 41 estão representados os dois espaços onde irão estar disponibilizadas as duas *features* planeadas para o futuro.



Figura 42 - Feedback Empresa JobRadar

1. A tabela Feedback apresenta as candidaturas recebidas, neste caso a imagem mostra um anúncio disponibilizado sem qualquer candidatura submetida.
2. Neste caso a tabela representa um anúncio disponibilizado já com uma candidatura submetida, sendo esta candidatura constituída pelo Nome do Candidato, Email, Contacto, CV, Comentários e Feedback.
3. Existe a possibilidade de analisar o CV do candidato fazendo o download do mesmo.
4. O utilizador responsável pela conta de Empresa no JobRadar tem a possibilidade de inserir o seu parecer em relação às competências requisitadas e demonstradas pelo candidato numa caixa de texto em aberto, tendo também a possibilidade de alterar o estado da candidatura no campo de Feedback, passando de “Pendente” para “Sim” (Aprovado) ou “Não” (Não Aprovado).
5. Ao pressionar o botão “Guardar”, o utilizador guarda as alterações feitas.
6. Ao pressionar o botão “Fechar página”, o utilizador fecha a janela Feedback e é redirecionado para a página de perfil da empresa.



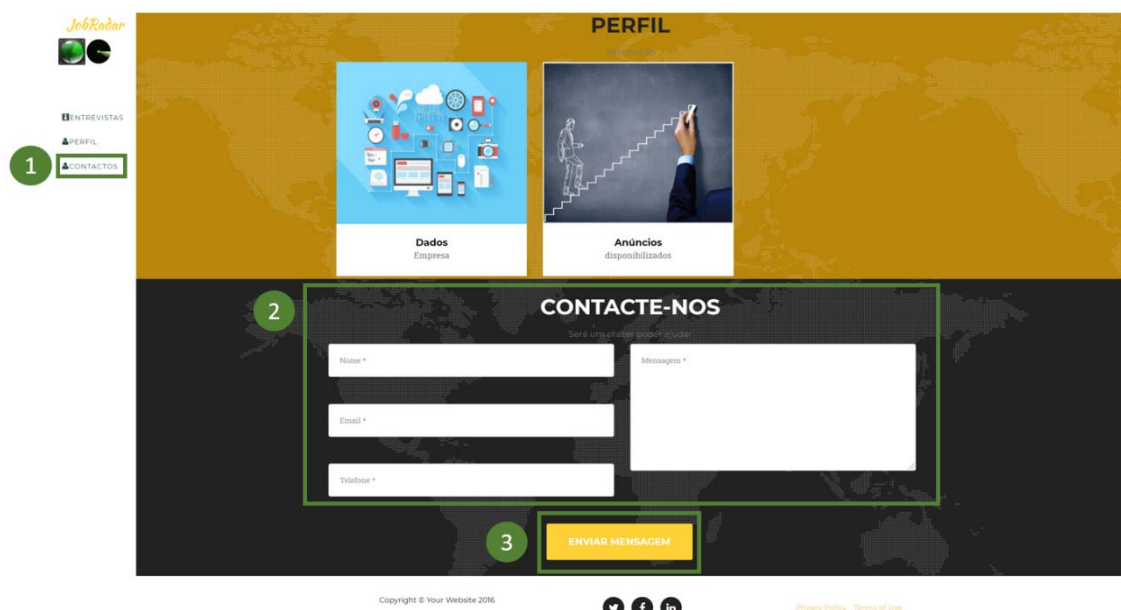
Figura 43 – Dados Empresa Empresa JobRadar

1. Neste campo é pretendido que a empresa possa inserir os seus dados ou atualizá-los.
2. O utilizador da conta de empresa, no JobRadar, ao pressionar o botão “Buscar Coordenadas” faz uma triangulação, de forma a descobrir as coordenadas de onde foi criada a conta.
3. O utilizador da conta de empresa, no JobRadar, ao pressionar o botão “Guardar”, guarda as alterações feitas.
4. Ao pressionar o botão “Fechar página”, o utilizador fecha a janela Dados Empresa e é redirecionado para a página de perfil da empresa.



Figura 44 - Contacte-nos Empresa JobRadar

1. A tabela permite criar anúncios de forma a serem visualizados do lado dos Candidatos. Permite descrever Nome do anúncio, a Categoria, a Função, os Requisitos e o Vencimento Bruto.
2. Ao pressionar o botão “Guardar”, o utilizador publica o anúncio com as informações preenchidas.
3. Neste caso, a tabela apresenta um anúncio disponibilizado anteriormente. Sendo o anúncio constituído por Nome, Descrição, Categoria e Remuneração.
5. Ao pressionar o botão “Fechar página”, o utilizador fecha a janela Feedback e é redirecionado para a página de perfil da empresa.



**Figura 45 - Contacte-nos Empresa JobRadar**

1. A Secção Contactos é o espaço reservado para a empresa poder fazer pedidos de ajuda ou esclarecimentos à cerca do aplicativo Web JobRadar.

2 Neste campo é pretendido que a Empresa possa inserir os seus dados, Nome, Email e Telefone, juntamente com uma mensagem descritiva do assunto em questão. Receberá um email de resposta automática, pré-criado, confirmando a receção do seu pedido.

3. Ao pressionar o botão “Enviar Mensagem”, o utilizador enviará a mensagem para o Administrador com toda a informação inserida.



## 7. Conclusão

Com base na sua definição inicial:

O problema identificado consiste em não existir um acompanhamento dado ao candidato durante todo o seu processo de recrutamento. Outro problema identificado foi o facto de os sites existentes não acompanharem o dinamismo e a evolução dos SI, não usando features intuitivas e que facilitem a navegação ao utilizador.

Research Goals predefinidos no início do trabalho de pesquisa:

RG 1- Realizar a revisão de literatura para a compreensão dos aspetos técnicos inerentes à realização da aplicação web JobRadar (por exemplo, conceito *Responsive Web Design*, *Open Source*, Modelo Cliente- Servidor, Modelo de Base de Dados Relacional);

RG 2- Elaborar a especificação de requisitos relativos à aplicação JobRadar;

RG 3- Definir a Arquitetura do Sistema;

RG 4- Elaboração do Diagrama de atividades;

RG 5- Definir o Modelo de Base de Dados;

RG 6- Testar e validar todas as tecnologias de modo a suportarem o protótipo aplicacional JobRadar;

RG 7- Elaboração de *mockup* do futuro protótipo JobRadar e esquema de navegação;

RG 8- Desenvolver a interface e funcionalidades de acordo com o *mockup* e esquema de navegação desenvolvido;

RG 9- Desenvolver a *feature* Google Maps em conformidade com o pretendido;

RG 10- Testar funcionalidades do sistema.

Assim, relativamente ao cada um dos Research Goals:

RG 1- A revisão da literatura encontra-se no capítulo 2;

RG 2- A especificação dos requisitos encontra-se no capítulo 3;

RG 3- A arquitetura do sistema encontra-se no capítulo 4;

RG 4- O Diagrama de atividades encontra-se no capítulo 4.2;

RG 5- A estrutura da Base de dados encontra-se no capítulo 4.6;

RG 6- A forma utilizada para testar e validar todas as tecnologias de modo a suportar funcionamento do aplicativo JobRadar encontra-se ao longo do capítulo 5;

RG 7- O protótipo do sistema, mockup e o esquema navegacional encontram-se no capítulo 5;

RG 8- A interface e as suas funcionalidades encontram-se especificadas no capítulo 6;

RG 9- Não foi atingido o objetivo de implementar a feature do Google Maps, ficando na *wishlist* como *feature* prioritária;

RG 10- Todos os testes e validações foram realizados ao longo de todo o desenvolvimento do aplicativo.

Tendo em conta os RG mencionados anteriormente, na conclusão preliminar considera-se que o problema identificado foi pertinente e que necessita de uma solução um tanto quanto ambiciosa como o JobRadar. O objetivo passa por criar uma plataforma de suporte e de apoio aos candidatos. As empresas também terão um suporte específico. A plataforma JobRadar vem sensibilizar os candidatos a emprego para a maneira como agem e como se devem preparar para qualquer tipo de entrevista/ função. Fornece também a atualidade do mercado de trabalho, em Portugal, em formato de blog de notícias e, não menos importante, é um como se de um motor de busca de empregos se tratasse. O JobRadar propõem-se em continuar a sua implementação e melhoria de *features* com a ambição de conseguir facilitar a utilização dos candidatos na plataforma e melhorar a qualidade de vida da população alvo.

Em relação às empresas que disponibilizam oportunidades/anúncios de emprego, tem uma maior autonomia e a oportunidade de contactar em tempo real com qualquer

candidato que seja do seu interesse. Por sua vez, o JobRadar compromete-se ainda, em continuar a promover os seus valores éticos e as suas boas práticas associadas à redução da taxa de desemprego que, infelizmente, sofreu um aumento exponencial, nos últimos meses, devido à situação atual do Mundo.

No início desta dissertação foi tido como objetivo fazer a revisão da literatura, que incluiu a revisão de um conjunto de conhecimentos que permitiu criar uma melhor perceção das áreas envolvidas.

Preparar uma análise e especificar os requisitos de todo o desenvolvimento, fez com que fosse realizada uma análise crítica, utilizando a metodologia *Design Science Research* que permitiu criar guidelines para o desenvolvimento do mockup e respetivo protótipo.

Ao longo da pesquisa foi necessário proceder à compreensão do sistema para projectar uma solução, com os requisitos específicos para esta aplicação. A criação de diagramas, Diagrama de uso, Diagrama de classes e Diagrama de atividades tiveram um papel fundamental nesta fase, tal como na arquitetura de todo o aplicativo Web.

De forma a comprovar o seu conceito, o Job Radar compromete-se futuramente a disponibilizar a *feature* que capacita a procura de anúncios de emprego e também de empresas, a todos os candidatos, através de um mecanismo geográfico, um mapa intuitivo, que funcionará através da API do Google Maps. Tomando por base as três áreas de maior interesse de cada candidato, o mapa possibilitará ao mesmo executar procuras, a partir de um ponto, até um raio de ‘x’ quilómetros.

Em relação às empresas, o Job Radar proporcionar-lhes-à num futuro próximo a possibilidade de agendarem entrevistas diretamente com os candidatos pretendidos, como de entrarem em contacto com os mesmos através de um chat privado.

Por último, toda a aplicação foi concebida para o mercado português, e, por esse motivo, apenas existe uma opção de idioma, o português. Porém, existe a possibilidade de

expandir o aplicativo para mercados internacionais e isso, por sua vez, obrigará a que estejam também disponíveis mais idiomas.

## Bibliografia

- Al-Masree, H. K. (2015). *Extracting Entity Relationship Diagram (ERD) From Relational*. Obtido em 6 de Setembro de 2020, de <https://pdfs.semanticscholar.org/1698/cce228b07f416285bfb4afeef99e0858c28b.pdf>
- Arruda, D. F. (2020). *Requirements Engineering in the Context of Big Data Software*. Western University. Obtido em 11 de Novembro de 2020, de <https://ir.lib.uwo.ca/cgi/viewcontent.cgi?article=9262&context=etd>
- Bergantin, C. E. (2019). *Análise de Boas Práticas para o Desenvolvimento de Web Apps*. AMA - Agência para a Modernização Administrativa, IP; TicAPP - Centro de Competências Digitais da Administração Pública. Obtido em 3 de Outubro de 2020, de <https://usabilidade.gov.pt/documents/20142/33332/Guia+de+boas+pr%C3%AAticas+de+usabilidade+para+aplica%C3%A7%C3%B5es+m%C3%B3veis.pdf/88bcf81b-f7db-4e18-1e32-b42f885e9e4b>
- Black, J., Cochran, M., & Highland, T. (2006). *A Study of the MD5 Attacks: Insights and Improvements*. Obtido em 24 de Setembro de 2020, de [https://link.springer.com/chapter/10.1007/11799313\\_17](https://link.springer.com/chapter/10.1007/11799313_17)
- Bornia, G. S. (2005). *Estruturação de Descrições de Casos de Uso através de Mecanismos de Extensibilidade da UML*. Obtido em 1 de Setembro de 2020, de <http://ucdesigner.sourceforge.net/dissertacao-2007-09-23.pdf>
- Bourque, P., & Failey, R. E. (2014). *Guide to the software engineering body of knowledge-SWEBOKv3.0*. IEEE Press. Obtido em 27 de Setembro de 2020, de <https://cs.fit.edu/~kgallagher/Schtick/Serious/SWEBOKv3.pdf>
- Brown, D. K., Rolander, T. A., Silberstein, R. D., & Wein, J. (2002). *Client Server Model For Synchronization of Files*. Obtido em 27 de Novembro de 2020, de <https://patentimages.storage.googleapis.com/e4/a5/c4/6eb95fa4bf1065/WO2002075539A3.pdf>
- Carvalho, R. N. (2015). *Usabilidade, Acessibilidade e Qualidade da Web da Administração Pública Portuguesa*. Escola de Tecnologias e Arquitectura (ISTA) Departamento de Ciências e Tecnologias de Informação. Obtido em 18 de Outubro de 2020, de [https://repositorio.iscte-iul.pt/bitstream/10071/10911/1/Dissertacao\\_Ricardo\\_Carvalho\\_Completa\\_v1.pdf](https://repositorio.iscte-iul.pt/bitstream/10071/10911/1/Dissertacao_Ricardo_Carvalho_Completa_v1.pdf)

- Coelho, J. (2011). *Introdução à Base de Dados*. Universidade Aberta. Obtido em 25 de Outubro de 2020, de <https://repositorioaberto.uab.pt/bitstream/10400.2/3462/1/Introdu%C3%A7%C3%A3o%20%C3%A0%20Base%20de%20Dados.pdf>
- Connolly, T., & Begg, C. (2015). *Database Systems - A Practical Approach to Design, Implementation, Sixth edition*. University of Paisley, Paisley University. Pearson. Obtido em 22 de Outubro de 2020, de <https://dl.ebooksworld.ir/motoman/Pearson.Database.Systems.A.Practical.Approach.to.Design.Implementation.and.Management.6th.Global.Edition.www.EBooksWorld.ir.pdf>
- Damghd, I. B. (1990). *A Design Principle for Hash Functions*. Obtido em 16 de Setembro de 2020, de [file:///C:/Users/joao\\_/Downloads/Damg%C3%A5rd1990\\_Chapter\\_ADesignPrincipleForHashFuncio.pdf](file:///C:/Users/joao_/Downloads/Damg%C3%A5rd1990_Chapter_ADesignPrincipleForHashFuncio.pdf)
- Eis, D., & Ferreira, E. (2012). *HTML5 e CSS3 com farinha e pimenta*. Lulu.com. Obtido em 3 de Novembro de 2020, de [https://www.academia.edu/9413581/HTML5\\_e\\_CSS3\\_com\\_farinha\\_e\\_pimenta](https://www.academia.edu/9413581/HTML5_e_CSS3_com_farinha_e_pimenta)
- Gehani, N. H., Jagadish, H. V., & Shmueli, O. (1992). *Composite Event Specification in Active Databases: Model & Implementation*. Obtido em 6 de Novembro de 2020, de <https://pdfs.semanticscholar.org/7f64/72973aa166870dd0b0f197666a61b51393f1.pdf>
- Gish, S. L. (1998). *Object-oriented system, method and article of manufacture for a client-server application enabler system*. Obtido em 23 de Setembro de 2020, de <https://patentimages.storage.googleapis.com/ef/59/cc/ac07c47d5fcf5e/US5768510.pdf>
- Guedes, G. T. (2008). *UML 2 – Uma Abordagem Prática 2ª Edição*. Obtido em 5 de Novembro de 2020, de <https://s3.novatec.com.br/capitulos/capitulo-9788575222812.pdf>
- Gustafson, A. (2011). *Adaptive Web Design Crafting Rich Experiences with Progressive Enhancement*. Krista Stevens. Obtido em 20 de Setembro de 2020, de [file:///C:/Users/joao\\_/Desktop/JobRadar%20Pappers/Adaptive-Web-Design.pdf](file:///C:/Users/joao_/Desktop/JobRadar%20Pappers/Adaptive-Web-Design.pdf)

- Hevner, A. R., Ram, S., March, S. T., & Park, J. (2004). *Design Science in Information - Systems Research*. Obtido em 17 de Outubro de 2020, de [https://wise.vub.ac.be/sites/default/files/thesis\\_info/design\\_science.pdf](https://wise.vub.ac.be/sites/default/files/thesis_info/design_science.pdf)
- Høstland, K., Enstad, P. A., & Øyvind. (2010). *Information Security Policy - Best Practice Document*. UNINETT. Obtido em 13 de Novembro de 2020, de [https://sigarra.up.pt/up/pt/web\\_gessi\\_docs.download\\_file?p\\_name=F96095589/no-uninett-terena-information-security-policy-best-practice-document-gn3-na3-t4-ufs126.pdf](https://sigarra.up.pt/up/pt/web_gessi_docs.download_file?p_name=F96095589/no-uninett-terena-information-security-policy-best-practice-document-gn3-na3-t4-ufs126.pdf)
- Lenaeus, J. D., O'Neil, L., Leitch, R. M., C. Glantz, G. P., Bryant, J. L., Lewis, J., . . . Johnson, C. W. (2015). *How to Implement Security Controls for an Information Security Program at CBRN Facilities*. Pacific Northwest. Obtido em 11 de Outubro de 2020, de [https://www.pnnl.gov/main/publications/external/technical\\_reports/PNNL-25112.pdf](https://www.pnnl.gov/main/publications/external/technical_reports/PNNL-25112.pdf)
- March, T. S., & Smith, G. F. (1995). *Design and natural science research on information technology*. Obtido em 26 de Setembro de 2020, de <http://www.siks.nl/RM%20M%20nov%202006/research%20%20paper%20marchsmithdssarticle.pdf>
- Mihir Bellare, R. C. (1996). *Keying Hash Functions for Message Authentication*. Obtido em 22 de Setembro de 2020, de <https://cseweb.ucsd.edu/~mihir/papers/kmd5.pdf>
- Montanheiro, L. S. (2018). *Primeiros Passos para o Desenvolvimento Seguro de Aplicações Web*. Obtido em 15 de Setembro de 2020, de <https://repositorio.ifgoiano.edu.br/bitstream/prefix/1266/1/TCC-Montanheiro-v9-FichaOK%20lucas.pdf>
- Norman, D. (2013). *The Design of Everyday Things - Revised & Expanded Edition*. The Perseus Books Group. Obtido em 10 de Novembro de 2020, de [http://www.sunyoungkim.org/class/old/hci\\_f18/pdf/The-Design-of-Everyday-Things-Revised-and-Expanded-Edition.pdf](http://www.sunyoungkim.org/class/old/hci_f18/pdf/The-Design-of-Everyday-Things-Revised-and-Expanded-Edition.pdf)
- Pedro, A. G. (2014). *Design de Interfaces - O Relevo da Coerência Gráfica do Desenvolvimento das Interfaces de uma Marca*. (F. d. Lisboa, Ed.) Obtido em 3 de Novembro de 2020, de [file:///C:/Users/joao\\_/Downloads/Andr%C3%A9%20Gavino%20%E2%80%94%20Design%20de%20Interfaces-%20O%20relevo%20da%20coer%C3%Aancia%20gr%C3%A1fica%20no%20desenvolvimento%20das%20interfaces%20de%20uma%20marca.pdf](file:///C:/Users/joao_/Downloads/Andr%C3%A9%20Gavino%20%E2%80%94%20Design%20de%20Interfaces-%20O%20relevo%20da%20coer%C3%Aancia%20gr%C3%A1fica%20no%20desenvolvimento%20das%20interfaces%20de%20uma%20marca.pdf)

- Pieprzyk, J., & Sadeghiyan, B. (1993). *Design of Hashing Algorithms*. Obtido em 17 de Setembro de 2020, de [https://doc.lagout.org/science/0\\_Computer%20Science/2\\_Algorithms/Design%20of%20Hashing%20Algorithms%20%5BPieprzyk%20%26%20Sadeghiyan%201993-11-23%5D.pdf](https://doc.lagout.org/science/0_Computer%20Science/2_Algorithms/Design%20of%20Hashing%20Algorithms%20%5BPieprzyk%20%26%20Sadeghiyan%201993-11-23%5D.pdf)
- Pompeu, G., Azevedo, G., Victorino, M., & Pacote, M. (2012). *Tecnologia da Informação*. Elsevier Editora Ltda. Obtido em 5 de Novembro de 2020, de <https://docero.com.br/doc/5sxscn>
- Rivest, R. (1992). *The MD5 Message-Digest Algorithm*. Obtido em 18 de Setembro de 2020, de <https://www.rfc-editor.org/rfc/pdf/rfc1321.txt.pdf>
- Rumbaugh, J., Jacobson, I., & Booch, G. (1999). *The Unified Modeling Language Reference Manual*. ADDISON WESLEY. Obtido em 25 de Novembro de 2020, de [http://www.temida.si/~bojan/IPIT\\_2014/literatura/UML\\_Reference\\_Manual.pdf](http://www.temida.si/~bojan/IPIT_2014/literatura/UML_Reference_Manual.pdf)
- Silver, M. S., Markus, M. L., & Beath, C. M. (1995). *The Information Technology Interaction Model: A Foundation for the MBA*. Obtido em 8 de Novembro de 2020, de [https://www.academia.edu/28401505/The\\_Information\\_Technology\\_Interaction\\_Model\\_A\\_Foundation\\_for\\_the\\_MBA\\_Core\\_Course](https://www.academia.edu/28401505/The_Information_Technology_Interaction_Model_A_Foundation_for_the_MBA_Core_Course)
- Stevens, M. (2012). *Single-block collision attack on MD5*. Obtido em 1 de Novembro de 2020, de <https://marc-stevens.nl/research/md5-1block-collision/md5-1block-collision.pdf>
- Streeter, D. C. (2013). *Operational Security and Cyber Security*. Liberty University. Obtido em 11 de Outubro de 2020, de [https://www.academia.edu/5490787/Operational\\_Security\\_and\\_Cyber\\_Security?auto=download](https://www.academia.edu/5490787/Operational_Security_and_Cyber_Security?auto=download)
- Sulyman, S. H. (2014). *Client-Server Model*. Obtido em 19 de Setembro de 2020, de [https://www.academia.edu/43822833/Client\\_Server\\_Model](https://www.academia.edu/43822833/Client_Server_Model)
- Weber, S. (2004). *The Success of Open Source*. Obtido em 4 de Novembro de 2020, de <https://epdf.pub/queue/the-success-of-open-source.html>