



# WORKR

TeSP em Programação de Sistemas de Informação

Jason Vieira Mendes

Tiago João Ruivo Ramos

Leiria, fevereiro de 2022



# WORK

Jason Vieira Mendes, Nº 2180601

Tiago João Ruivo Ramos, Nº 2180689

Trabalho de Projeto em Sistemas de Informação do curso TeSP de Programação de Sistemas de Informação.

Leiria, fevereiro de 2022

## **Originalidade e Direitos de Autor**

O presente relatório de projeto é original, elaborada/o unicamente para este fim, tendo sido devidamente citados todos os autores cujos estudos e publicações contribuíram para a/o elaborar.

Reproduções parciais deste documento serão autorizadas na condição de que seja mencionado/a o/a Autor/a e feita referência ao ciclo de estudos no âmbito do qual a/o mesma/o foi realizado, a saber, TeSP em Programação de Sistemas de Informação, no ano letivo 2021/2022, da Escola Superior de Tecnologia e Gestão do Instituto Politécnico de Leiria, Portugal, e, bem assim, à data das provas que visaram a avaliação destes trabalhos.

# **Agradecimentos**

Queremos agradecer toda a paciência e a ajuda de todos os docentes das nossas unidades curriculares que nos acompanharam até ao final do nosso projeto. Também queremos agradecer ao coordenador de curso Sílvio Priem Mendes por ter aprovado o nosso projeto e pelas dicas de melhoramento do mesmo.

Sem esquecer também das nossas famílias que nos apoiaram durante o nosso percurso académico e que nunca desistiram da nossa capacidade e empenho.

# **Resumo**

## **Relatório de Projeto – Workr**

**Jason Vieira Mendes**

**Tiago João Ruivo Ramos**

A aplicação Workr é um serviço de procura de trabalho/emprego, para desenvolvedores de software, inspirada em outras já existentes. Esta tem a sua parte web que é ligada à base de dados diretamente e a parte do sistema em android que é ligado por uma API à mesma.

Neste sistema, o utilizador poderá prestar um serviço sendo na aplicação móvel ou no website ou candidatar-se a uma empresa que procura um prestador de serviços. Os utilizadores poderão editar os seus perfis e as empresas poderão efetuar pagamentos aos seus trabalhadores.

**PALAVRAS-CHAVE:** workr, android, trabalho, serviços

# **Abstract**

## **Project Essay – Workr**

**Jason Vieira Mendes**

**Tiago João Ruivo Ramos**

The Workr application is a service, inspired by other applications, where software developers can look for tasks in other companies to maximize their workflow and productivity in exchange of a payment. This application has its website which is connected to a database directly and an android application that is connected to an API to get data from that same database.

In this system, the user can lend their services to a company in need, this can be done either on the website or the mobile application. The users can edit their profiles and the promoter of the company can send payments to the clients for their services.

**Keywords:** workr, android, work, services

# Índice

<b>Originalidade e Direitos de Autor .....</b>	<b>iii</b>
<b>Agradecimentos .....</b>	<b>iv</b>
<b>Resumo .....</b>	<b>v</b>
<b>Abstract .....</b>	<b>vi</b>
<b>Lista de Figuras .....</b>	<b>ix</b>
<b>Lista de tabelas .....</b>	<b>xi</b>
<b>Lista de siglas e acrónimos.....</b>	<b>xii</b>
<b>1. Introdução .....</b>	<b>1</b>
<b>2. Metodologia.....</b>	<b>3</b>
<b>3. Arquitetura do Sistema.....</b>	<b>6</b>
<b>4. Gestão do Projeto .....</b>	<b>7</b>
<b>4.1. Matrizes de Responsabilidade .....</b>	<b>7</b>
4.1.1. Matriz de Responsabilidade de Web .....	7
4.1.2. Matriz de Responsabilidade API .....	8
4.1.3. Matriz de Responsabilidade Android .....	9
<b>5. Análise .....</b>	<b>11</b>
<b>6. Casos de uso e Mockups.....</b>	<b>17</b>
<b>7. Implementação.....</b>	<b>18</b>
<b>8. Testes .....</b>	<b>20</b>
<b>8.1. Desenho dos Testes .....</b>	<b>20</b>
8.1.1. Testes Unitários .....	20
8.1.2. Testes Funcionais .....	20
8.1.3. Testes de Aceitação .....	20
<b>8.2. Objetivo dos Testes.....</b>	<b>21</b>
8.2.1. Testes Unitários .....	21
8.2.2. Testes Funcionais .....	21
8.2.3. Testes de Aceitação .....	22
<b>8.3. Resultados Obtidos.....</b>	<b>22</b>
<b>9. Conclusão e Trabalho Futuro.....</b>	<b>23</b>
<b>Webgrafia .....</b>	<b>24</b>
<b>Anexos.....</b>	<b>25</b>



# **Lista de Figuras**

Figura 1 - Arquitetura do Sistema .....	25
Figura 2 - Casos de Uso da Aplicação.....	26
Figura 3 - Mockup Web Página principal.....	27
Figura 4 - Mockup Web Empresa.....	28
Figura 5 - Mockup Android Página principal.....	29
Figura 6 - Mockup Android filtrar empresa .....	29
Figura 7 - Mockup Android Empresa.....	29
Figura 8 - Teste Unitário Cargo (validação, inserir e editar) .....	30
Figura 9 - Teste Unitário Cargo (eliminar).....	30
Figura 10 - Teste Unitário Contratante (inserir) .....	31
Figura 11 - Teste Unitário Contratante (validação) .....	31
Figura 12 - Teste Unitário Contratante (editar e eliminar) .....	31
Figura 13 - Teste Unitário Empresa (validação).....	31
Figura 14 - Teste Unitário Empresa (inserir e editar) .....	31
Figura 15 - Teste Unitário Empresa (eliminar).....	31
Figura 16 - Teste Unitário Ofertas (validação).....	31
Figura 17 - Teste Unitário Ofertas (inserir e editar) .....	31
Figura 18 - Teste Unitário Ofertas (eliminar).....	31
Figura 19 - Teste Unitário Prestador (validação).....	31
Figura 20 - Teste Unitário Prestador (inserir).....	31
Figura 21 - Teste Unitário Prestador (editar e eliminar).....	31
Figura 22 - Teste Funcional Candidatura .....	31
Figura 23 - Teste Funcional Criar CV .....	31
Figura 24 - Teste Funcional Registrar Contratante .....	31
Figura 25 - Teste Funcional Registrar Empresa.....	31
Figura 26 - Teste Funcional Registrar Prestador .....	31
Figura 27 - Teste Aceitacao Login .....	31

Figura 28 - Página Web ver Empresa (2/2).....	31
Figura 29 - Página Web ver Empresa (1/2).....	31
Figura 30 - Página Web Home.....	31
Figura 31 - Página Web Perfil Prestador.....	31
Figura 32 - Página Web Candidatura .....	31
Figura 33 - Página Web Feedback Empresa .....	31
Figura 34 - Página Web Perfil Contratante .....	31
Figura 35 - Página Web Editar Perfil .....	31
Figura 36 - Página Web Serviços Prestador .....	31
Figura 37 - Página Web Calendário de Entrevistas.....	31
Figura 38 - Página Web Gerir Candidatura.....	31
Figura 39 - Página Web Editar Perfil .....	31
Figura 40 - Página Web Atribuir Serviços .....	31
Figura 41 - Página Web Criar Ratings .....	31
Figura 42 - Página Web Criar Pagamento.....	31
Figura 43 - Página Web BackOffice Resolver Disputas .....	31
Figura 44 - Página Web BackOffice Criar Empresa .....	31
Figura 45 - Página Web Backoffice Home .....	31
Figura 46 - Página Web BackOffice Autorizar Contratante .....	31
Figura 47 - Página Web BackOffice Ratings .....	31
Figura 48 - Página Web BackOffice Histórico entre ambas as partes.....	31
Figura 49 - Burndown Report (Sprint 2).....	31
Figura 50 - Burndown Report (Sprint 1).....	31
Figura 51 - Cumulative Flow Diagram .....	31
Figura 52 - Sprint Burndown Chart (Sprint 2) .....	31
Figura 53 - Sprint Burndown Chart (Sprint 1) .....	31
Figura 54 - Burndown Report (Sprint 3).....	31
Figura 55 - Sprint Burndown Chart (Sprint 3) .....	31

# **Lista de tabelas**

Tabela 1 - Metodologia Web .....	3
Tabela 2 - API Metodologia .....	3
Tabela 3 - Android Metodologia .....	4
Tabela 4 - Matriz de Responsabilidade de Web .....	7
Tabela 5 - Matriz de Responsabilidade API .....	8
Tabela 6 - Matriz de Responsabilidade Android .....	9
Tabela 7 - Requisitos Funcionais .....	11

# **Lista de siglas e acrónimos**

API	Application Programming Interface
ESTG	Escola Superior de Tecnologia e Gestão
YII2	Yes It Is!
DER	Diagrama de Entidade e Relacionamento

# 1. Introdução

Com o avançar do tempo e com a evolução do mundo tecnológico, a procura de desenvolvedores de Software é sempre cada vez mais elevada.

O Workr é uma aplicação móvel com um Website que, tem como objetivo, facilitar a procura de emprego para Desenvolvedores e também facilitar a procura de prestadores de serviços para empresas que os necessitem.

No âmbito das Unidades Curriculares de Plataformas de Sistemas de Informação, Acesso Móvel a Sistmas de Informação e Serviços e Interoperabilidade de Sistemas, esta aplicação terá três vertentes. A primeira é uma aplicação móvel que o utilizador, no Dashboard/Feed poderá se candidatar a uma empresa deslizando o dedo da esquerda para a direita, esta é interligada ao website por uma API feita na Framework Yii2 (Yes It Is!). O website terá uma interface que mostrará as empresas registadas na aplicação e o utilizador poderá procurar por categoria ou nome da empresa e ainda candidatar-se a qualquer empresa que estiver interessado. As empresas ou contratantes também poderão procurar prestadores de serviços para as suas necessidades enquanto empresa. Estes serviços serão renumerados e o pagamento também é feito pela aplicação com uma taxa de pagamento para o Workr.

Na presente secção é feita uma breve descrição dos objetivos de todos os pontos deste relatório. A secção 2, Metodologia, vai demonstrar o tempo estimado, o desvio e a importância de cada caso de uso na implementação das nossas aplicações. A secção 3, Arquitetura do Sistema, é onde vamos apresentar a arquitetura do nosso projeto, explicando-a ao pormenor. A secção 4, Gestão do Projeto, é onde se vai demonstrar os gráficos de implementação do projeto designados por tarefa e tempo e as Matrizes de Responsabilidade para cada tarefa. A secção 5, Análise, é a parte do relatório onde se vai mostrar os Caso de Uso das nossas aplicações do projeto com as suas pré-condições, sequencia de acontecimentos, caminhos alternativos e pós-condições dos mesmos. A secção 6, Desenho, é demonstrado todos os protótipos, Mockups e casos de uso do sistema. A secção 7, Implementação, é explicado ao pormenor todos os passos, comandos e problemas que levaram à conclusão deste projeto. Na secção 8, Testes, mostramos todos os testes criados para o desenvolvimento da nossa aplicação, estes sendo Testes Unitários, Funcionais e de

Aceitação, bem como os mesmos a correr e a sucederem. A secção 9 é usada para concluir este relatório com um breve parágrafo conclusivo.

## 2. Metodologia

Ao longo do projeto introduzido no ponto número um, foi utilizada uma Metodologia Ágil que implicou um desenvolvimento mais rápido e com maior frequência conforme o que era necessário para a continuação do projeto em si.

Neste relatório irão ser indicadas as técnicas usadas para a utilização desta metodologia.

WEB:

**Tabela 1 - Metodologia Web**

Tarefa	Tempo Estimado	Desvio	Importância
Ver lista de empresas	3 Dias	NA	Elevada
Utilizador gerir o seu próprio perfil	2 Dias	NA	Elevada
Criar Empresa	1 Dia	1 Dia	Elevada
Gerir Empresa	2 Dias	1 Dia	Média
Criar Currículo	1 Dia	NA	Média
Candidatar-se a uma Empresa	2 Dias	2 Dias	Elevada
Gestão de Utilizadores	1 Dia	2 Dias	Média
Fazer Registo	1 Dia	NA	Elevada
Ver lista Empregados	1 Dia	NA	Baixa
Gestão de Empregados	2 Dias	1 Dia	Média
Gestão de Empregadores	2 Dias	1 Dia	Média
Calendário de Entrevistas	1 Dia	NA	Baixa
Autenticação	1 Dia	NA	Elevada
Filtrar empresas de acordo com categoria	2 Dias	1 Dia	Média
Gestão das ofertas de serviços ou posições	2 Dias	2 Dias	Média
Realizar Pagamentos	1 Dia	NA	Elevada
Feedback Contratual público	3 Dias	NA	Elevada

API:

**Tabela 2 - API Metodologia**

Tarefa	Tempo Estimado	Desvio	Importância
Listar Utilizadores	2 Dias	NA	Elevada
Criar Utilizadores	1 Dia	NA	Média
Editar Utilizadores	1 Dia	NA	Média
Eliminar Utilizadores	1 Dia	NA	Média
Registo	2 Dias	NA	Elevada
Login	2 Dias	NA	Elevada
Listar Cargo	2 Dias	NA	Elevada
Criar Cargo	1 Dia	NA	Média
Editar Cargo	1 Dia	NA	Média
Eliminar Cargo	1 Dia	NA	Média

Listar CV	2 Dias	NA	Elevada
Criar CV	1 Dia	NA	Média
Editar CV	1 Dia	NA	Média
Eliminar CV	1 Dia	NA	Média
Listar Disputas	2 Dias	NA	Elevada
Criar Disputas	1 Dia	NA	Média
Editar Disputas	1 Dia	NA	Média
Eliminar Disputas	1 Dia	NA	Média
Listar Empresa	2 Dias	NA	Elevada
Criar Empresa	1 Dia	NA	Média
Editar Empresa	1 Dia	NA	Média
Eliminar Empresa	1 Dia	NA	Média
Listar Feedback	2 Dias	NA	Elevada
Criar Feedback	1 Dia	NA	Média
Editar Feedback	1 Dia	NA	Média
Eliminar Feedback	1 Dia	NA	Média
Listar Localidades	2 Dias	NA	Elevada
Criar Localidades	1 Dia	NA	Média
Editar Localidades	1 Dia	NA	Média
Eliminar Localidades	1 Dia	NA	Média
Listar Ofertas	2 Dias	NA	Elevada
Criar Ofertas	1 Dia	NA	Média
Editar Ofertas	1 Dia	NA	Média
Eliminar Ofertas	1 Dia	NA	Média
Listar Pagamento	2 Dias	NA	Elevada
Criar Pagamento	1 Dia	NA	Média
Editar Pagamento	1 Dia	NA	Média
Eliminar Pagamento	1 Dia	NA	Média
Listar Ratings	2 Dias	NA	Elevada
Criar Ratings	1 Dia	NA	Média
Editar Ratings	1 Dia	NA	Média
Eliminar Ratings	1 Dia	NA	Média
Listar Serviços	2 Dias	NA	Elevada
Criar Serviços	1 Dia	NA	Média
Editar Serviços	1 Dia	NA	Média
Eliminar Serviços	1 Dia	NA	Média
Listar Tipo de Pagamento	2 Dias	NA	Elevada
Criar Tipo de Pagamento	1 Dia	NA	Média
Editar Tipo de Pagamento	1 Dia	NA	Média
Eliminar Tipo de Pagamento	1 Dia	NA	Média

Android:

Tabela 3 - Android Metodologia

Tarefa	Tempo Estimado	Desvio	Importância
Layout Login	1 Dia	NA	Elevada
Layout Registro	1 Dia	NA	Elevada

Layout menu principal Feed	2 Dias	1 Dia	Elevada
Layout Perfil utilizador	3 Dias	NA	Elevada
Layout Swipe	4 Dias	2 Dias	Elevada
Layout criar Empresa	1 Dia	NA	Elevada
Layout Oferta	1 Dia	NA	Elevada
Layout Pagamento	1 Dia	NA	Elevada
Funcionalidade Login	3 Dias	1 Dia	Elevada
Funcionalidade Registo	2 Dias	1 Dia	Elevada
Funcionalidade Registo Anónimo	1 Dia	NA	Baixa
Funcionalidade ver Empresas	1 Dia	NA	Média
Funcionalidade Candidatar empresa	2 Dias	1 Dia	Elevada
Funcionalidade criar Oferta	2 Dias	NA	Elevada
Funcionalidade criar Pagamento	2 Dias	NA	Elevada
Funcionalidade criar Empresa	2 Dias	NA	Elevada
Base de Dados local	5 Dias	NA	Média
Ligaçāo API	4 Dias	2 Dias	Elevada

### 3. Arquitetura do Sistema

A arquitetura do sistema apresentada implica uma ligação Apache com Wampserver64 à base de dados que por sua vez irá fazer a ligação ao website e irá fazer com que os comandos CURL e Custom enviem os dados para a aplicação móvel como demonstra na Figura 1.

Como demonstra na figura, a base de dados faz ligação ao website diretamente e faz a ligação à API que então irá mandar os dados para a aplicação móvel para serem usadas e manipuladas na mesma.

## 4. Gestão do Projeto

Neste tópico são apresentados os gráficos que explicam a gestão do projeto, desde o início ao seu fim (Figuras 49 até 55).

Estes gráficos foram retirados da plataforma Jira de [www.atlassian.net](http://www.atlassian.net).

Também será apresentada a Matriz de Responsabilidade para cada funcionalidade anteriormente apresentada.

### 4.1. Matrizes de Responsabilidade

#### 4.1.1. Matriz de Responsabilidade de Web

Tabela 4 - Matriz de Responsabilidade de Web

Tarefa	Jason Mendes	Tiago Ramos
Registo	2	4
Autenticação	2	4
Candidatar a uma Empresa	1	
Dar Feedback	2	4
Página das Empresas		1
Perfil	1	
Ver trabalhos atribuidos		1
Report		1
Criar CV	1	
Ver Ratings		1
Filtrar empresas por Localidade		1
Editar Empresa	1	

Editar Ofertas	1	
Editar Perfil	1	
Gerir candidaturas	1	
Calendário de Entrevistas	1	
Enviar Pagamentos		1
Criar Ratings		1
Atribuir Serviços		1
Registrar Empresa	2	4
Gerir Reports	2	4
Ver processo histórico entre todas as partes	2	4
Ratings de users (backoffice)	2	4
Aceitar Contratantes	1	
Legenda: 1-100% 2-50% 3-25% 4- Restante		

#### 4.1.2. Matriz de Responsabilidade API

Tabela 5 - Matriz de Responsabilidade API

Tarefa	Jason Mendes	Tiago Ramos
Listar Utilizadores		1
Criar Utilizadores		1
Atualizar Utilizadores		1
Eliminar Utilizadores		1
Registo		1
Autenticação		1
Listar Empresas	2	4
Criar Empresas	2	4

Atualizar Empresas	2	4
Listar Pagamentos		1
Criar Pagamentos		1
Atualizar Pagamentos		1
Eliminar Pagamentos		1
Listar CV	2	4
Criar CV	2	4
Atualizar CV	2	4
Eliminar CV	2	4
Listar Ofertas	2	4
Criar Ofertas	2	4
Atualizar Ofertas	2	4
Eliminar Ofertas	2	4
Listar Serviços	2	4
Criar Serviços	2	4
Atualizar Serviços	2	4
Eliminar Serviços	2	4
Legenda: 1-100% 2-50% 3-25% 4- Restante		

#### 4.1.3. Matriz de Responsabilidade Android

Tabela 6 - Matriz de Responsabilidade Android

Tarefa	Jason Mendes	Tiago Ramos
Registo	2	4
Autenticação	2	4
Gerir Perfil	2	4
Activities	1	

---

Criar CV	2	4
Candidatar a uma Empresa	1	
Filtrar Empresas	1	
Swipe		1
Realizar pagamentos	1	
Layouts	1	
Legenda: 1-100% 2-50% 3-25% 4- Restante		

## 5. Análise

A aplicação anteriormente apresentada tem como objetivo principal criar uma plataforma onde os desenvolvedores de aplicações de informação possam ter uma experiência mais interativa e simples para obter tarefas fora do seu local de trabalho para um rendimento mais elevado vindo de empresas necessitadas de trabalhadores a curto-médio prazo.

A tabela seguinte irá percorrer todos os requisitos funcionais do projeto e explicar detalhadamente as mesmas.

**Tabela 7 - Requisitos Funcionais**

Requisito	Atores	Pré-condições	Seqüência de Acontecimentos	Caminhos Alternativos	Pós-condições
<b>Registo</b>	Utilizador	Para o utilizador ter acesso ao registo terá de abrir a aplicação Web e criar a sua conta no browser	<ul style="list-style-type: none"> <li>• Abrir Browser ou Aplicação;</li> <li>• Preencher campos com os dados pedidos;</li> <li>• Efetuar Registo.</li> </ul>	NA	O registo será efetuado e será redirecionado para a página de login.
<b>Login</b>	Utilizador	Para o utilizador ter acesso ao login deverá abrir o site ou aplicação móvel e ir para a aba Login.	<ul style="list-style-type: none"> <li>• Abrir Browser ou Aplicação;</li> <li>• Preencher o campos de login;</li> <li>• Efetuar Login.</li> </ul>	Caso não tenha conta criada deverá se registrar para criar uma conta e de seguida efetuar login	Caso o login seja efetuado com sucesso irá ser redirecionado para a página principal da aplicação.

<b>Candidatar a Empresa</b>	Prestador de Serviços	Para o prestador de serviços poder candidatar-se a uma empresa terá que estar autenticado e ir à página principal da aplicação	<ul style="list-style-type: none"> <li>• Autenticar-se;</li> <li>• Ir para página principal da aplicação;</li> <li>• Selecionar Empresa;</li> <li>• Clicar “Candidatar a Empresa”. (Web);</li> <li>• Deslizar o dedo da esquerda para a direita no “Card” da empresa (android)</li> </ul>	Caso não esteja autenticado, não deixará candidatar-se a uma empresa	Após sucedido, vai aparecer uma mensagem no browser ou na aplicação android e poderá posteriormente inserir as empresas onde está inserido, no perfil
<b>Filtrar empresas de acordo com categoria</b>	Utilizador	No menu da esquerda (Web), estão lá as cidades de empresas que existem dentro da base de dados onde o cliente poderá clicar e irá filtrar as empresas de acordo com esse clique.	<ul style="list-style-type: none"> <li>• Página principal;</li> <li>• Clicar nos nomes de localidades no menu da esquerda;</li> </ul>	NA	Após clicar, as empresas serão filtradas e poderá então candidatar-se a empresas da localidade desejada.
<b>Perfil</b>	Prestador de Serviços	Na barra de navegação poderá clicar “Ver Perfil” e mostrará todos os seus dados enquanto Prestador de Serviços, os pagamentos feitos a si e os trabalhos atribuídos.	<ul style="list-style-type: none"> <li>• Clicar “Bem Vindo” (Web);</li> <li>• Clicar em “Ver Perfil” (Web);</li> <li>• Clicar em “Perfil” na navegação da aplicação (Android).</li> </ul>	Não estar autenticado	Após clicar no perfil, o Prestador de Serviços poderá ver os seus dados.

<b>Perfil</b>	Contratante	<p>Na barra de navegação poderá clicar “Ver Perfil” e mostrará todos os seus dados enquanto Contratante, as suas Empresas bem como editar os mesmos.</p>	<ul style="list-style-type: none"> <li>• Clicar “Bem Vindo” (Web);</li> <li>• Clicar em “Ver Perfil” (Web);</li> <li>• Clicar em “Perfil” na navegação da aplicação (Android).</li> </ul>	Não estar autenticado	Após clicar no perfil, o Contratante poderá ver os seus dados.
<b>Criar Curriculum Vitae</b>	Prestador de Serviços	<p>Na barra de navegação é só clicar em “Criar CV” e o Prestador de Serviços poderá preencher o formulário e criar o seu currículo.</p>	<ul style="list-style-type: none"> <li>• Clicar “Bem Vindo” (Web);</li> <li>• Clicar em “Criar CV” (Web);</li> <li>• Clicar em “Criar CV” na navegação da aplicação (Android).</li> </ul>	Não estar autenticado	Após criar o Currículo o Prestador de Serviços poderá usá-lo para candidatar-se a uma empresa.
<b>Feedback</b>	Prestador de Serviços	<p>Uma Empresa precisa de estar registada e o prestador autenticado</p>	<ul style="list-style-type: none"> <li>• Clicar numa Empresa no feed;</li> <li>• Clicar em “Dar Feedback”;</li> <li>• Preencher formulário e enviar.</li> </ul>	Não estar autenticado	Após enviar o feedback da empresa, o Prestador de Serviços receberá uma mensagem na página web.
<b>Calendário de Entrevistas</b>	Contratante	<p>Na barra de navegação é só clicar em “Calendário de Entrevistas” e então o contratante poderá criar uma entrevista clicando no dia que pertende.</p>	<ul style="list-style-type: none"> <li>• Clicar em “Bem Vindo” (Web);</li> <li>• Clicar em “Calendário de Entrevista” (Web);</li> </ul>	Não estar autenticado	Após clicar em “Calendário de Entrevistas”, o contratante poderá clicar no dia de entrevista e criar uma entrevista.

<b>Criar Pagamentos</b>	Contratante	Para ter acesso aos pagamentos basta ir à barra de navegação e clicar em “Pagamentos”.	<ul style="list-style-type: none"> <li>• Clicar em “Bem Vindo” (Web);</li> <li>• Clicar em “Pagamentos” (Web);</li> <li>• Preencher o formulário e enviar (Web);</li> <li>• Clicar em “Pagamentos” (Android);</li> <li>• Preencher com os dados (Android).</li> </ul>	Não estar autenticado ou não ter Prestadores a trabalhar para Empresas suas.	Após criar pagamento, irá aparecer uma mensagem no browser e o Prestador será também notificado na sua página de perfil.
<b>Visualizar Ratings</b>	Prestador de Serviços	Para visualizar os seus ratings, o Prestador de Serviços deverá estar autenticado e deverá ter ratings atribuídos.	<ul style="list-style-type: none"> <li>• Clicar em “Bem Vindo”;</li> <li>• Clicar em “Ver os meus Ratings”.</li> </ul>	Não estar autenticado ou não ter ratings atribuídos.	NA
<b>Criar Ratings</b>	Contratante	Para criar um rating, o Contratante deverá ter um prestador atribuído a uma Empresa sua.	<ul style="list-style-type: none"> <li>• Clicar em “Bem Vindo”;</li> <li>• Clicar em “Ratings”;</li> <li>• Preencher o formulário e enviar.</li> </ul>	Não estar autenticado ou não ter Prestadores atribuídos à sua empresa.	Após criar um Rating, aparecerá uma mensagem no website e o Prestador poderá ver os seus Ratings na sua página.
<b>Atribuir Serviços</b>	Contratante	Para atribuir Serviços o Contratante deverá ter um Prestador a trabalhar para si.	<ul style="list-style-type: none"> <li>• Clicar em “Bem Vindo”;</li> <li>• Clicar em “Atribuir Serviços”;</li> <li>• Preencher o formulário e enviar.</li> </ul>	Não estar autenticado ou não ter Prestadores atribuídos à sua empresa.	Após atribuir um Serviço, aparecerá uma mensagem no website e o Prestador poderá ver os seus Serviços no seu perfil.

<b>Criar Disputa</b>	Prestador de Serviços	Para criar uma disputa, o Prestador de Serviços deverá ter um pagamento feito a si.	<ul style="list-style-type: none"> <li>• Clicar em “Bem Vindo”;</li> <li>• Clicar em “Ver Perfil”;</li> <li>• No perfil terá um botão a dizer “Report”;</li> <li>• De seguida deverá escrever uma breve descrição da disputa e enviar.</li> </ul>	Não estar autenticado ou não ter nenhum pagamento efetuado a si.	Após criar uma Disputa, o Prestador de serviços terá uma mensagem no website e terá que esperar que o mediador (Admin) resolva a disputa entre si e a Empresa atribuída.
<b>Aceitar Contratantes</b>	Mediador (Admin)	Para aceitar contratantes, o Mediador deverá ter acesso ao BackOffice.	<ul style="list-style-type: none"> <li>• Autenticar no BackOffice;</li> <li>• Clicar em “Aceitar Contratantes”;</li> <li>• Clicar em “Autorizar”.</li> </ul>	Não ter acesso ao BackOffice.	Após aceitar um Contratante, o mediador terá uma mensagem na página web e posteriormente, o Contratante, poderá autenticar-se na aplicação.
<b>Ratings dos Users</b>	Mediador (Admin)	Para visualizar os ratings, deverá haver ratings feitos a Prestadores de Serviços.	<ul style="list-style-type: none"> <li>• Autenticar no BackOffice;</li> <li>• Clicar em “Ratings de Users”;</li> <li>• (Opcional) Clicar em “Delete”.</li> </ul>	Não ter acesso ao BackOffice.	Após eliminar um Rating, o Mediador será notificado na página web e o rating desaparecerá da lista.
<b>Resolver Disputa</b>	Mediador (Admin)	Para resolver disputas, o Mediador deverá ter acesso ao BackOffice.	<ul style="list-style-type: none"> <li>• Autenticar no BackOffice;</li> <li>• Clicar em “Reports”;</li> <li>• Clicar em “Resolver”.</li> </ul>	Não ter acesso ao BackOffice.	Após resolver uma Disputa, o mediador terá uma mensagem na página web e posteriormente, o Prestador de Serviços será notificado.

<b>Ver Pagamentos</b>	Mediador (Admin)	Para ver o histórico de pagamentos de ambas as partes, o Mediador deverá ter acesso ao BackOffice.	<ul style="list-style-type: none"> <li>• Autenticar no BackOffice;</li> <li>• Clicar em “Pagamentos”.</li> </ul>	Não ter acesso ao BackOffice.	NA
<b>Registrar Empresa</b>	Mediador (Admin)	Para criar uma Empresa, o Mediador deverá ser notificado pelo Contratante já registado.	<ul style="list-style-type: none"> <li>• Autenticar no BackOffice;</li> <li>• Clicar em “Registrar Empresa”;</li> <li>• Preencher os Formulário.</li> </ul>	Não ter acesso ao BackOffice ou não ser notificado pelo Contratante.	Após Registar uma Empresa, aparecerá uma mensagem na página web e de seguida a empresa será automaticamente atribuída ao Contratante.

## 6. Casos de uso e Mockups

No início do projeto foram feitas mockups e os casos de uso com as nossas visões iniciais de como seria o projeto, naturalmente ao longo do desenvolvimento do projeto foram adicionadas e removidas funcionalidades e aspetos visuais tanto da componente de web como a de aplicação.

As mockups e casos de uso são as da visão final do projeto, estas serão apresentadas nos [Anexos](#). (Figuras 2 até 7)

## 7. Implementação

Neste ponto vai ser explicado detalhadamente o processo de implementação do projeto apresentado.

No começo do projeto, foi criado em papel, um esboço, modelo lógico e DER da base de dados a implementar para manipular os dados inseridos. De acordo com o esboço realizado, foi implementada a base de dados com algumas modificações do esboço original.

Para implementar a parte web deste projeto, foi concebido um template advanced da framework Yii2 com Composer utilizando o comando “composer create-project --prefer-dist yiisoft/yii2-app-advanced workr” no git Bash, uma aplicação para manipular software. De seguida foram criados os modelos, controladores e vistas utilizando a ferramenta “gii” da framework, também como as migrações para que se pudesse desenvolver o registo e autenticação, bem como as permissões em RBAC de cada funcionalidade e o backend para o administrador (que tem todas as permissões do projeto). Posteriormente foram feitos os perfis dos utilizadores, a página principal onde são expostas as empresas aceites pelo administrador, bem como as suas ofertas para os prestadores de serviços e o resto das funcionalidades do FrontOffice e do BackOffice. Foi decidido que na página principal da aplicação web poderá se filtrar as empresas por localidade pela barra de navegação. Só o administrador e o contratante pertencente poderão manipular as empresas bem como aceitar algum prestador de serviços na empresa. O prestador de serviços pode candidatar-se a uma empresa onde este vai ser atribuído uma tarefa, onde poderá ser pago pelo contratante por tal serviço. Este também pode criar um currículo, observar os pagamentos já feitos ao mesmo, bem como reportar algum problema com o pagamento e também observar as tarefas atribuídas a si. O administrador é o único utilizador que poderá criar uma Empresa e atribuí-la a um contratante. Este também pode observar o processo histórico entre todas as partes, ver os ratings dos utilizadores bem como eliminar algum à sua escolha, resolver disputas criadas entre o prestador de serviços e as suas empresas e também aceitar contratantes para poderem entrar na aplicação.

Para criar uma ligação entre a base de dados e a aplicação móvel, foi criada uma API, que é implementada dentro dos ficheiros do website. Como o website, com o auxílio de “gii” foram criados os modelos e de seguida os CRUD’s (modelos, controladores e vistas). Foram implementadas as ações cURL (client URL) para todos os controladores que fazem o

trabalho de ir buscar (GET), inserir (POST), alterar (PUT) e eliminar (DELETE). Além dos comandos cURL já implementados, foram ainda criados comandos “custom” que podem fazer ações criadas pelo utilizador no controlador.

No ponto anterior foram introduzidos os Mockups utilizados para a aplicação móvel, com o auxílio desses Mockups foram criados os Layouts da aplicação móvel. Foi implementado o sistema de registo e autenticação buscando dos dados da API. Ao entrar na aplicação, o utilizador poderá ver as empresas registadas na aplicação, bem como as ofertas das mesmas. O utilizador poderá se candidatar a qualquer empresa e esperar que seja aceite pelo contratante. Este também tem acesso ao seu perfil onde poderá ver as empresas onde presta serviços, bem como os pagamentos feitos a si. O utilizador pode também criar um currículo na aplicação móvel.

## 8. Testes

### 8.1. Desenho dos Testes

No desenvolvimento do projeto Workr foram utilizados testes unitários, testes funcionais e testes de aceitação.

#### 8.1.1. Testes Unitários

- Teste ao Cargo - backend/tests/unit/CargoTest.php;
- Teste ao Contratante - backend/tests/unit/ContratanteTest.php;
- Teste à Empresa - backend/tests/unit/EmpresaTest.php;
- Teste às Ofertas - backend/tests/unit/OfertasTest.php;
- Teste ao Prestador de Serviços - backend/tests/unit/PrestadorTest.php;

(Figuras 8 até 21).

#### 8.1.2. Testes Funcionais

- Teste à Candidatura - frontend/tests/functional/CandidaturaCest.php;
- Teste ao Criar CV - frontend/tests/functional/CriarCVCest.php;
- Teste ao Registar Contratante - frontend/tests/functional/RegisterContratanteCest.php;
- Teste ao Registar Prestador de Serviços - frontend/tests/functional/RegisterPrestadorCest.php;
- Teste ao Registar Empresa - backend/tests/functional/RegistrarEmpresaCest.php;

(Figuras 22 até 26).

#### 8.1.3. Testes de Aceitação

- Teste ao Login - frontend/tests/acceptance/HomeCest.php;

(Figura 27).

## 8.2. Objetivo dos Testes

### 8.2.1. Testes Unitários

Teste ao Cargo:

- Validar a inserção e inserir o cargo na base de dados;
- Mudar o nome do cargo;
- Eliminar o cargo.

Teste ao Contratante:

- Validar a inserção e inserir o Contratante na base de dados;
- Mudar o nome do Contratante;
- Eliminar Contratante.

Teste à Empresa:

- Validar a inserção e inserir a Empresa na base de dados;
- Mudar o nome da Empresa;
- Eliminar Empresa.

Teste às Ofertas:

- Validar a inserção e inserir a Oferta na base de dados;
- Mudar o regime da Oferta;
- Eliminar Oferta.

Teste ao Prestador de Serviços:

- Validar a inserção e inserir o Prestador de Serviços na base de dados;
- Mudar o nome do Prestador;
- Eliminar Prestador.

### 8.2.2. Testes Funcionais

Teste à Candidatura:

- Candidatar-se a uma Empresa.

Teste ao Criar CV:

- Criar um Currículo para o Prestador de Serviços.

Teste ao Registar Contratante:

- Registar um Contratante.

Teste ao Registar Prestador de Serviços:

- Registar um Prestador de Serviços.

Teste ao Registar Empresa:

- Registar uma Empresa.

### 8.2.3. Testes de Aceitação

Teste ao Login:

- Efetuar Login.

## 8.3. Resultados Obtidos

ID	Cenário	Modelo	Pass/Fail
#1	Validar a inserção e inserir o Prestador de Serviços na base de dados	Prestador	Pass
#2	Validar a inserção e inserir o Contratante na base de dados	Contratante	Pass
#3	Validar a inserção e inserir o Cargo na base de dados	Cargo	Pass
#4	Validar a inserção e inserir a Empresa na base de dados	Empresa	Pass
#5	Validar a inserção e inserir Oferta na base de dados	Ofertas	Pass

ID	Cenário	Página	Pass/Fail
#1	Candidatar-se a uma Empresa	frontend/empresa/candidatura	Pass
#2	Criar um Currículo para o Prestador de Serviços.	frontend/cv/create	Pass
#3	Registrar um Contratante.	frontend/Signup	Pass
#4	Registrar um Prestador de Serviços	frontend/Signup	Pass
#5	Registrar uma Empresa.	backend/empresa/create	Pass

ID	Cenário	Pass/Fail
#1	Login na página principal	Pass

## 9. Conclusão e Trabalho Futuro

Depois de alguns meses de trabalho árduo, damos por concluído o desenvolvimento do Workr.

Tivemos várias adversidades ao longo do nosso projeto, tais como, um dos nossos colegas de grupo decidiu desistir do curso a meio do primeiro semestre o que deixou o Tiago e o Jason a fazerem as componentes todas sozinhos. Com este obstáculo, decidimos fazer com que as nossas funcionalidades tivessem todas a trabalhar como devido e deixámos a parte do layout das aplicações por terminar.

Apesar destas dificuldades, conseguimos atingir os nossos objetivos enquanto aplicação. Por falta de tempo e de mão de obra não conseguimos implementar a principal funcionalidade da aplicação móvel, sendo esta, ao deslizar o dedo da esquerda para a direita, o utilizador conseguiria candidatar-se à sua empresa desejada para uma dinâmica maior entre o utilizador e a aplicação.

Com trabalho para o futuro irão ser implementadas as seguintes funcionalidades:

- Deslizar o dedo da esquerda para a direita para se candidatar a uma empresa, ou ao contrário para passar para a empresa seguinte;
- Layout completo da página web, bem como na aplicação móvel;
- Filtrar empresas de acordo com categoria ou natureza do trabalho (mobile);

Concluindo, podemos dizer que o Workr foi um desafio e uma mais-valia para o nosso conhecimento e aprendizagem enquanto programadores juniores e futuros desenvolvedores de aplicações. Foram partilhados métodos de como resolver problemas pelos membros do grupo que enriqueceram a química dos mesmos.

## Webgrafia

Guias da framework Yii2 em: <<https://www.yiiframework.com/>>.

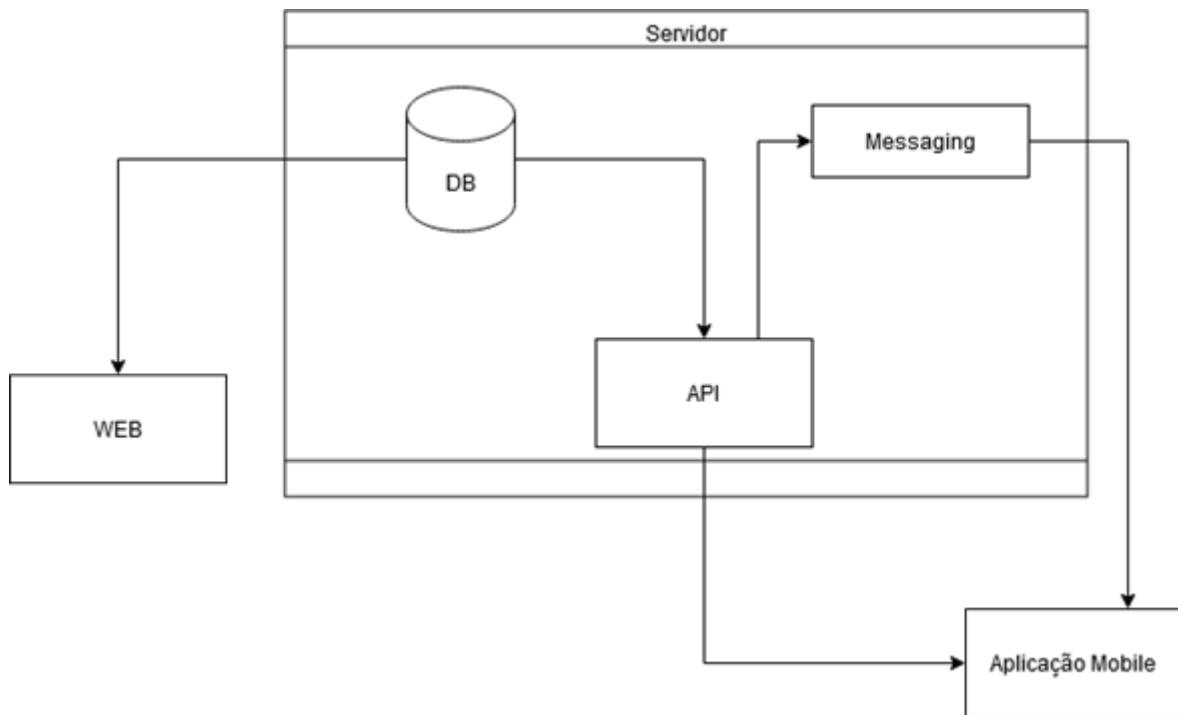
Guias de codeception em: <<https://codeception.com/>>.

Correção de bugs no sistema em: <<https://pt.stackoverflow.com/>>

Guias de Android Studio em: <<https://developer.android.com/docs>>

Guias para Layout e CSS em: <<https://getbootstrap.com/docs/5.1/getting-started/introduction/>>

## Anexos



**Figura 1 - Arquitetura do Sistema**

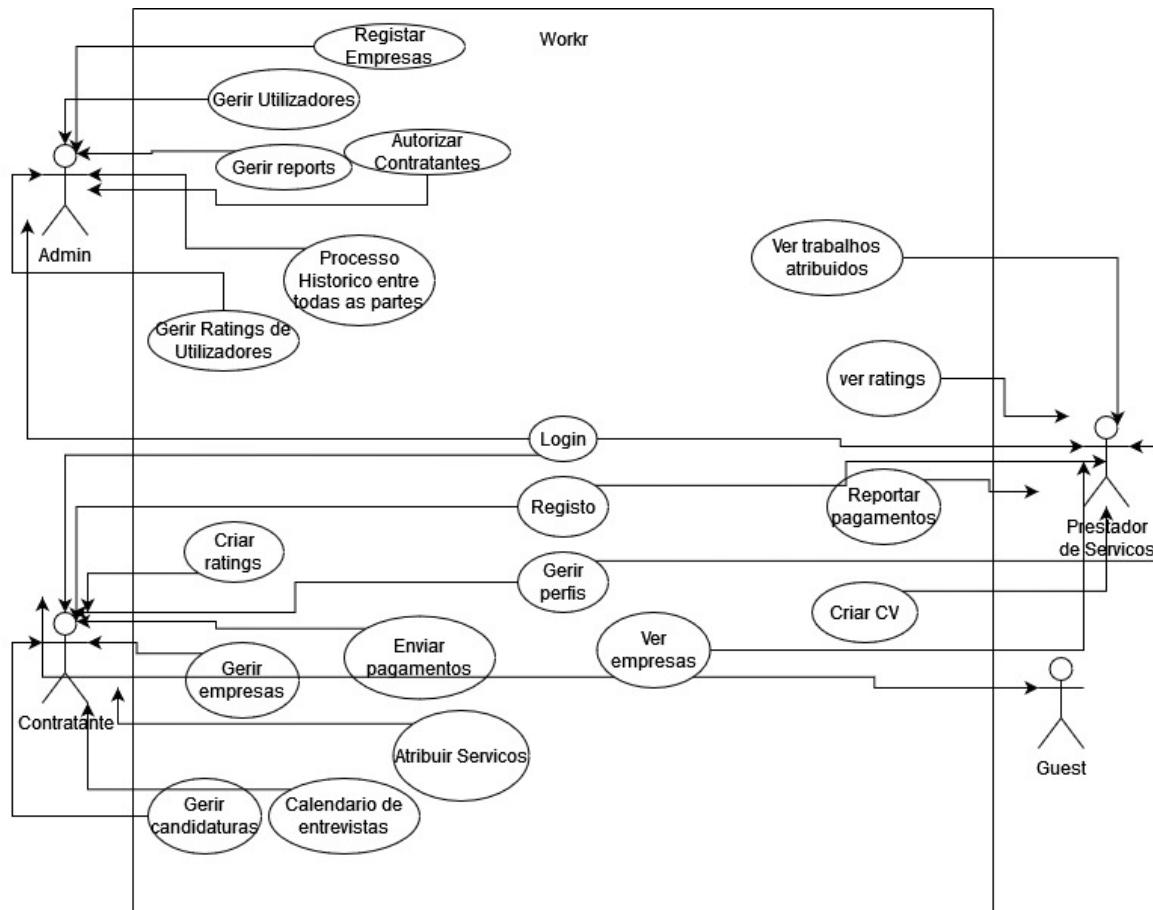


Figura 2 - Casos de Uso da Aplicação

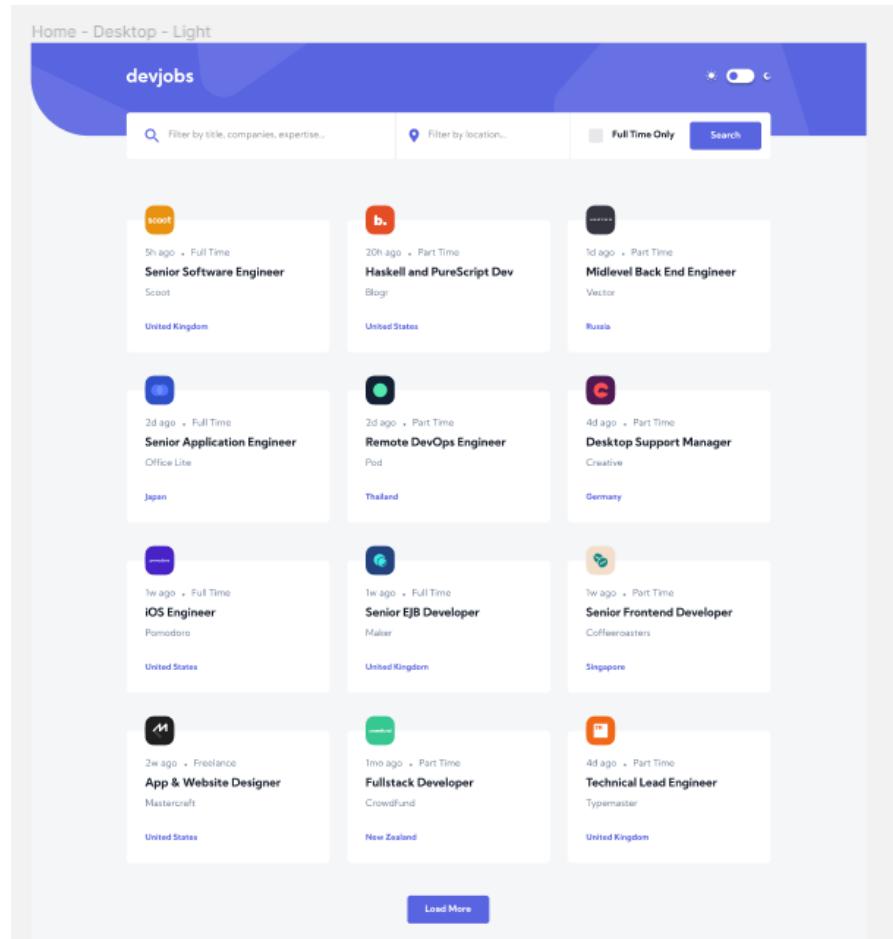
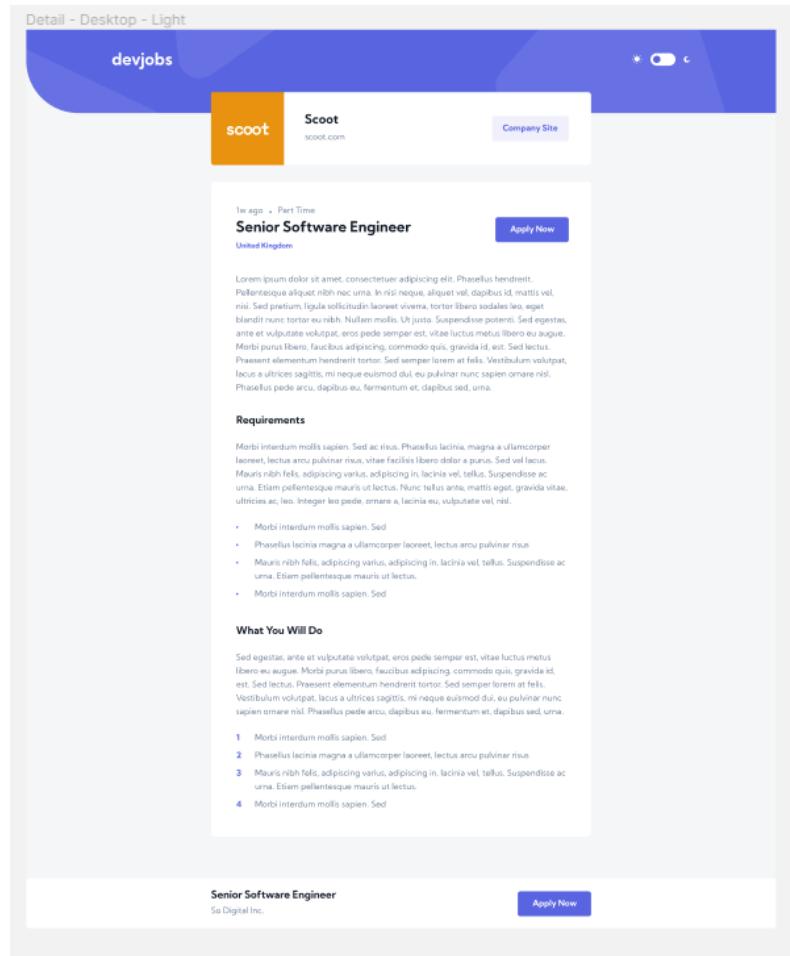


Figura 3 - Mockup Web Página principal



**Figura 4 - Mockup Web Empresa**

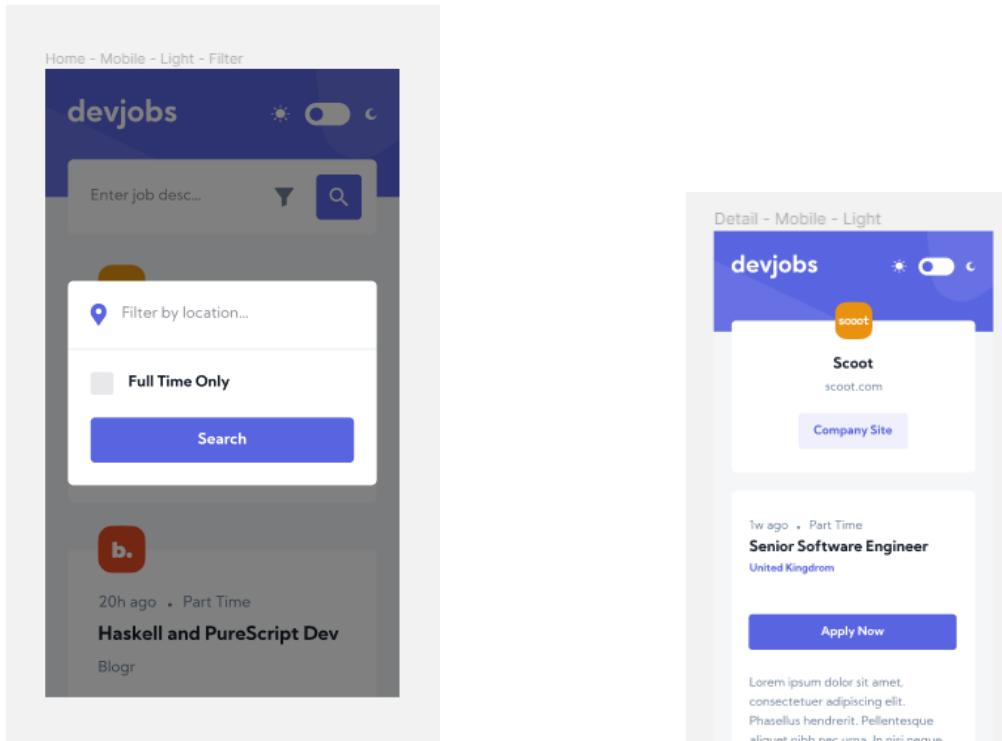


Figura 6 - Mockup Android filtrar empresa

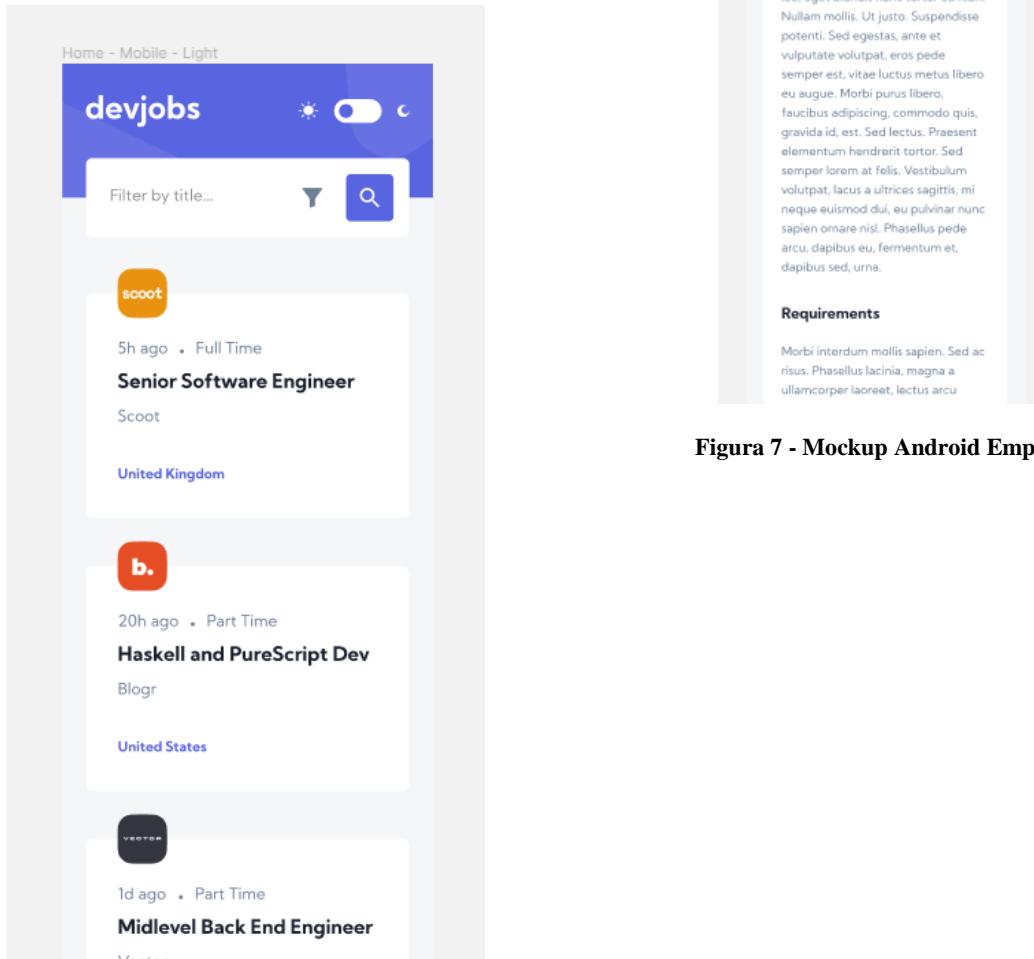


Figura 5 - Mockup Android Página principal

```

22     public function testSomeFeature()
23     {
24         $cargo = new Cargo();
25
26         $cargo->cargo = null;
27         $this->assertFalse($cargo->validate(['cargo']));
28         $cargo->cargo = 'fghnvghtooloooongnafghnvghtoolasdddddddddssadsadsadsadsadsadsadsadsadsads';
29         $this->assertFalse($cargo->validate(['cargo']));
30         $cargo->cargo = 'CargoTeste';
31         $this->assertTrue($cargo->validate(['cargo']));
32     }
33
34     function testSavingUser()
35     {
36         $cargo = new Cargo();
37         $cargo->cargo = ('CargoTest');
38
39         $cargo->save();
40         $this->tester->seeInDatabase('cargo', ['cargo' => 'CargoTest']);
41     }
42
43     function testNameCanBeChanged()
44     {
45         $id = $this->tester->grabRecord('backend\models\Cargo', ['cargo' => 'CargoTest']);
46
47         $cargo = Cargo::findOne($id);
48         $cargo->cargo = ('CargoUpdate');
49         $cargo->save();
50
51         $this->tester->seeRecord('backend\models\Cargo', ['cargo' => 'CargoUpdate']);
52         $this->tester->dontSeeRecord('backend\models\Cargo', ['cargo' => 'CargoTest']);
53     }

```

**Figura 8 - Teste Unitário Cargo (validação, inserir e editar)**

```

55     function testUserDeleted()
56     {
57         $id = $this->tester->grabRecord('backend\models\Cargo', ['cargo' => 'CargoUpdate']);
58
59         $cargo = Cargo::findOne($id);
60         $cargo->delete();
61
62         $this->tester->dontSeeRecord('backend\models\Cargo', ['cargo' => 'CargoUpdate']);
63     }
64 }

```

**Figura 9 - Teste Unitário Cargo (eliminar)**

```
public function testSomeFeature()
{
    $contratante = new Contratante();

    $contratante->nome = null;
    $this->assertFalse($contratante->validate(['nome']));
    $contratante->nome = 'fghnvghtooloooongnafghnvghtooloasdddddddddssadsadsadsadsadsa';
    $this->assertFalse($contratante->validate(['nome']));
    $contratante->nome = 'Jason';
    $this->assertTrue($contratante->validate(['nome']));

    $contratante->sexo = null;
    $this->assertFalse($contratante->validate(['sexo']));
    $contratante->sexo = 'fghnvghtooloooongnafghnvghtooloasdddddddddssadsadsadsadsadsa';
    $this->assertFalse($contratante->validate(['sexo']));
    $contratante->sexo = 'Masculino';
    $this->assertTrue($contratante->validate(['sexo']));

    $contratante->avatar = 'fghnvghtooloooongnafghnvghtooloasdddddddddssadsadsadsadsad';
    $this->assertFalse($contratante->validate(['avatar']));
    $contratante->avatar = '/workr/frontend/web/avatar/default-avatar.png';
    $this->assertTrue($contratante->validate(['avatar']));

    $contratante->datanascimento = null;
    $this->assertFalse($contratante->validate(['datanascimento']));
    $contratante->datanascimento = 'fghnvghtooloooongnafghnvghtooloasdddddddddssadsadsadsa';
    $this->assertFalse($contratante->validate(['datanascimento']));
    $contratante->datanascimento = '2021-11-23';
    $this->assertTrue($contratante->validate(['datanascimento']));
}
```

**Figura 11 - Teste Unitário Contratante (validação)**

```
function testSavingUser()
{
    $user = new User();
    $contratante = new Contratante();
    $user->username = 'testecontratante2';
    $user->email = 'testecontratante2@teste.com';

    $user->setPassword('testecontratante');
    $user->generateAuthKey();
    $user->generateEmailVerificationToken();
    $user->status = 10;

    $user->save();

    $contratante->nome = 'TesteContratante';
    $contratante->sexo = 'M';
    $contratante->avatar = "avatar.png";
    $contratante->datanascimento = '2021-11-10';
    $contratante->IDUser = $user->id;
    $contratante->IDCargo = 1;
    var_dump($contratante);
    ob_flush();
    $contratante->save();

    $this->tester->seeInDatabase('contratante', ['nome' => 'TesteContratante']);
}
```

**Figura 10 - Teste Unitário Contratante (inserir)**

```

function testNameCanBeChanged()
{
    $contratanteid = $this->tester->grabRecord('backend\models\Contratante', ['nome' => 'TesteContratante']);

    $contratante = Contratante::findOne($contratanteid->IDContratante);
    var_dump($contratante);
    ob_flush();
    $contratante->nome = ('TesteUpdated');
    $contratante->save(false);

    $this->tester->seeRecord('backend\models\Contratante', ['nome' => 'TesteUpdated']);
    $this->tester->dontSeeRecord('backend\models\Contratante', ['nome' => 'TesteContratante']);
}

function testUserDeleted()
{
    $contratante = $this->tester->grabRecord('backend\models\Contratante', ['nome' => 'TesteUpdated']);

    $contratante->delete();

    $this->tester->dontSeeRecord('backend\models\Contratante', ['nome' => 'TesteUpdated']);
}

```

**Figura 12 - Teste Unitário Contratante (editar e eliminar)**

```

// tests
public function testSomeFeature()
{
    $empresa = new Empresa();

    $empresa->nome = null;
    $this->assertFalse($empresa->validate(['nome']));
    $empresa->nome = 'dsadsadsadsadsasaaaaafdsafdsaffghjkcxhsshfgvhjvgjhgjhgjhgj';
    $this->assertFalse($empresa->validate(['nome']));
    $empresa->nome = 'EmpresaTest';
    $this->assertTrue($empresa->validate(['nome']));

    $empresa->descricao = null;
    $this->assertFalse($empresa->validate(['descricao']));
    $empresa->descricao = 'fghnvghtooloooongnafghnvghtoolaosdddddddddssadsadsadsadsadsads';
    $this->assertFalse($empresa->validate(['descricao']));
    $empresa->descricao = 'Masculino';
    $this->assertTrue($empresa->validate(['descricao']));

    $empresa->avatar = 'fghnvghtooloooongnafghnvghtoolaosdddddddddssadsadsadsadsadsads';
    $this->assertFalse($empresa->validate(['avatar']));
    $empresa->avatar = '/workr/frontend/web/avatar/default-avatar.png';
    $this->assertTrue($empresa->validate(['avatar']));

    $empresa->estado = 'Texto';
    $this->assertFalse($empresa->validate(['estado']));
    $empresa->estado = '1';
    $this->assertTrue($empresa->validate(['estado']));
}

```

**Figura 13 - Teste Unitário Empresa (validação)**

```

function testSavingUser()
{
    $empresa = new Empresa();

    $empresa->nome = 'EmpresaTest';
    $empresa->descricao = 'DescricaoTest';
    $empresa->avatar = 'AvatarTest';
    $empresa->estado = 1;
    $empresa->IDContratante = 1;
    $empresa->IDOferta = 1;
    $empresa->IDLlocalidade = 1;

    $empresa->save(false);

    $this->tester->seeInDatabase('empresa', ['nome' => 'EmpresaTest']);
}

function testNameCanBeChanged()
{
    $empresaid = $this->tester->grabRecord('backend\models\Empresa', ['nome' => 'EmpresaTest']);

    $empresa = Empresa::findOne($empresaid->IDEmpresa);
    $empresa->nome = 'EmpresaUpdate';
    $empresa->save(false);

    $this->tester->seeRecord('backend\models\Empresa', ['nome' => 'EmpresaUpdate']);
    $this->tester->dontSeeRecord('backend\models\Empresa', ['nome' => 'EmpresaTest']);
}

```

Figura 14 - Teste Unitário Empresa (inserir e editar)

```

function testUserDeleted()
{
    $id = $this->tester->grabRecord('backend\models\Empresa', ['nome' => 'EmpresaUpdate']);

    $empresa = Empresa::findOne($id);
    $empresa->delete();

    $this->tester->dontSeeRecord('backend\models\Empresa', ['nome' => 'EmpresaUpdate']);
}

```

Figura 15 - Teste Unitário Empresa (eliminar)

```
// tests
public function testSomeFeature()
{
    $ofertas = new Ofertas();

    $ofertas->regime = null;
    $this->assertFalse($ofertas->validate(['regime']));
    $ofertas->regime = 'dsadsadsadsadsasaaaafdsafdsaffghjkcxddsadsadsadasasasasasasahsshfgvhjvgjhgjh';
    $this->assertFalse($ofertas->validate(['regime']));
    $ofertas->regime = 'RegimeTest';
    $this->assertTrue($ofertas->validate(['regime']));

    $ofertas->requisito = null;
    $this->assertFalse($ofertas->validate(['requisito']));
    $ofertas->requisito = 'dsadsadsadsadsasaaaafdsafdsaffghjkcxddsadsadsadasasasasasasahsshfgvhjvgjhg';
    $this->assertFalse($ofertas->validate(['requisito']));
    $ofertas->requisito = 'RequisitoTest';
    $this->assertTrue($ofertas->validate(['requisito']));

    $ofertas->vagas = null;
    $this->assertFalse($ofertas->validate(['vagas']));
    $ofertas->vagas = 'Texto';
    $this->assertFalse($ofertas->validate(['vagas']));
    $ofertas->vagas = '1';
    $this->assertTrue($ofertas->validate(['vagas']));
}
```

**Figura 16 - Teste Unitário Ofertas (validação)**

```
function testSavingUser()
{
    $ofertas = new Ofertas();

    $ofertas->regime = 'RegimeTest';
    $ofertas->requisito = 'RequisitoTest';
    $ofertas->vagas = 1;
    $ofertas->save();

    $this->tester->seeInDatabase('ofertas', ['regime' => 'RegimeTest']);
}

function testNameCanBeChanged()
{
    $id = $this->tester->grabRecord('backend\models\Ofertas', ['regime' => 'RegimeTest']);

    $ofertas = Ofertas::findOne($id);
    $ofertas->regime = ('RegimeUpdate');
    $ofertas->save();

    $this->tester->seeRecord('backend\models\Ofertas', ['regime' => 'RegimeUpdate']);
    $this->tester->dontSeeRecord('backend\models\Ofertas', ['regime' => 'RegimeTest']);
}

function testUserDeleted()
```

**Figura 17 - Teste Unitário Ofertas (inserir e editar)**

```

function testUserDeleted()
{
    $id = $this->tester->grabRecord('backend\models\ofertas', ['regime' => 'RegimeUpdate']);

    $ofertas = Ofertas::findOne($id);
    $ofertas->delete();

    $this->tester->dontSeeRecord('backend\models\ofertas', ['regime' => 'RegimeUpdate']);
}

```

**Figura 18 - Teste Unitário Ofertas (eliminar)**

```

public function testSomeFeature()
{
    $prestador = new Prestador();

    $prestador->nome = null;
    $this->assertFalse($prestador->validate(['nome']));
    $prestador->nome = 'fghnvghtooloooongnafghnvghtoolasdddddddddssadsadsadsadsads';
    $this->assertFalse($prestador->validate(['nome']));
    $prestador->nome = 'Jason';
    $this->assertTrue($prestador->validate(['nome']));

    $prestador->sexo = null;
    $this->assertFalse($prestador->validate(['sexo']));
    $prestador->sexo = 'fghnvghtooloooongnafghnvghtoolasdddddddddssadsadsadsadsads';
    $this->assertFalse($prestador->validate(['sexo']));
    $prestador->sexo = 'Masculino';
    $this->assertTrue($prestador->validate(['sexo']));

    $prestador->avatar = 'fghnvghtooloooongnafghnvghtoolasdddddddddssadsadsadsadsads';
    $this->assertFalse($prestador->validate(['avatar']));
    $prestador->avatar = '/workr/frontend/web/avatar/default-avatar.png';
    $this->assertTrue($prestador->validate(['avatar']));

    $prestador->datanascimento = null;
    $this->assertFalse($prestador->validate(['datanascimento']));
    $prestador->datanascimento = 'fghnvghtooloooongnafghnvghtoolasdddddddddssadsadsadsad';
    $this->assertFalse($prestador->validate(['datanascimento']));
    $prestador->datanascimento = '2021-11-23';
    $this->assertTrue($prestador->validate(['datanascimento']));

    $prestador->nif = null;
    $this->assertFalse($prestador->validate(['nif']));
    $prestador->nif = 'Texto';
    $this->assertFalse($prestador->validate(['nif']));
    $prestador->nif = '123456789';
}

```

**Figura 19 - Teste Unitário Prestador (validação)**

```

function testSavingUser()
{
    $user = new User();
    $prestador = new Prestador();
    $user->username = 'testeprestador4';
    $user->email = 'testeprestador4@teste.com';

    $user->setPassword('testeprestador');
    $user->generateAuthKey();
    $user->generateEmailVerificationToken();
    $user->status = 10;

    $user->save();

    $prestador->nome = 'Teste';
    $prestador->sexo = 'M';
    $prestador->avatar = "avatarteste.png";
    $prestador->datanascimento = '2021-11-10';
    $prestador->nif = 123456789;
    $prestador->num_tele = 123456789;
    $prestador->IDUser = $user->id;

    $prestador->save();

    $this->tester->seeInDatabase('prestador', ['nome' => 'Teste']);
}

```

Figura 20 - Teste Unitário Prestador (inserir)

```

function testNameCanBeChanged()
{
    $prestadorid = $this->tester->grabRecord('backend\models\Prestador', ['nome' => 'Teste']);

    $prestador = Prestador::findOne($prestadorid->IDPrestador);
    var_dump($prestador);
    ob_flush();
    $prestador->nome = 'TesteUpdate';
    $prestador->save(false);

    $this->tester->seeRecord('backend\models\Prestador', ['nome' => 'TesteUpdate']);
    $this->tester->dontSeeRecord('backend\models\Prestador', ['nome' => 'Teste']);
}

function testUserDeleted()
{
    $prestador = $this->tester->grabRecord('backend\models\Prestador', ['nome' => 'TesteUpdate']);

    $prestador->delete();

    $this->tester->dontSeeRecord('backend\models\Prestador', ['nome' => 'TesteUpdate']);
}

```

Figura 21 - Teste Unitário Prestador (editar e eliminar)

```

class CandidaturaCest
{
    public function tryToTest(FunctionalTester $I)
    {
        //Main Page
        $I->amOnPage(\yii::$app->homeUrl);
        $I->see('Lista de Empresas Disponíveis:');

        //Fazer Login
        $I->click('Registrar/Login');
        $I->click('Fazer Login');
        $I->see('Login:');
        $I->fillField('Username', 'prestashop100');
        $I->fillField('Password', '123123123');
        $I->see('Remember Me');
        $I->click('login-button');
        $I->see('Lista de Empresas Disponíveis:');

        //Clicar num Anúncio
        $I->see('Google');
        $I->click('Google');
        $I->see('Candidatar-se');
        $I->click('Candidatar-se');
        $I->see('Candidatura para: Google');

        //Enviar Candidatura
        $I->fillField('Texto', 'Texto para o CV');
        $option = $I->grabTextFrom('select option:nth-child(2)');
        $I->selectOption("select", $option);
        $I->click('Candidatar-se');
        $I->see('Google');
    }
}

```

Figura 22 - Teste Funcional Candidatura

```

class CriarCVCest
{
    public function tryCriarCV(FunctionalTester $I)
    {
        //Main Page
        $I->amOnPage(\yii::$app->homeUrl);
        $I->see('Lista de Empresas Disponíveis:');

        //Fazer Login
        $I->click('Registrar/Login');
        $I->click('Fazer Login');
        $I->see('Login:');
        $I->fillField('Username', 'prestashop100');
        $I->fillField('Password', '123123123');
        $I->see('Remember Me');
        $I->click('login-button');
        $I->see('Lista de Empresas Disponíveis:');

        //Criar CV
        $I->click('Bem Vindo prestador100');
        $I->click('Criar CV');
        $I->see('Create Cv');
        $I->fillField('Nomeficheiro', 'CVTest');
        $I->fillField('Experiencia', 'CVTest');
        $I->fillField('Educacao', 'CVTest');
        $I->fillField('Comp Linguisticas', 'CVTest');
        $I->fillField('Comp Trabalho', 'CVTest');
        $I->see('Save');
        $I->click('Save');
    }
}

```

Figura 23 - Teste Funcional Criar CV

```

class RegisterContratanteCest
{
    public function tryCreateUser(FunctionalTester $I)
    {
        // Navegação desde Index até Registo
        $I->amOnPage(\yii::$app->homeUrl);
        $I->see('Lista de Empresas Disponíveis:');
        $I->click('Registar/Login');

        // $I->click('Convidado');
        $I->click('Signup Contratante');

        // Inserção de Dados
        $I->see('Signup Contratante');
        $I->fillField('Username', 'testefuncionalcontratante');
        $I->fillField('Email', 'testefuncionalcontratante@mailteste.com');
        $I->fillField('Password', 'testefuncionalcontratante');
        // $I->fillField('Nome', 'NomeContratanteTest');
        $option = $I->grabTextFrom('select option:nth-child(2)');
        $I->selectOption("select", $option);
        $I->fillField('DataNascimento', '1990-01-01');

        // ERRO NA ESCOLHA DE CARGO !!!
        $option = $I->grabTextFrom('select option:nth-child(3)');
        $I->selectOption("select", $option);

        $I->attachFile('#signupformcontratante-avatar', 'logo.png');

        //Final Create
        $I->click('#btn-signup');
        $I->see('Thank you for registration. Please check your inbox for verification email.');
    }
}

```

Figura 24 - Teste Funcional Registar Contratante

```

class RegistrarEmpresaCest
{
    public function tryRegistrarEmpresa(FunctionalTester $I)
    {
        //Login
        $I->amOnPage(\yii::$app->homeUrl);
        $I->see('Sign in to start your session');

        $I->fillField('LoginForm[username]', 'admin');
        $I->fillField('LoginForm[password]', '123123123');
        $I->click('Sign In');
        $I->see('Congratulations!');

        //Registrar Empresa
        $I->see('Registrar Empresa');
        $I->click(['id' => 'registaremepresa']);
        $I->amOnPage('empresa/create');
        $I->see('Create Empresa');
        $I->fillField('Nome', 'EmpresaTeste');
        $I->fillField('Descricao', 'EmpresaTeste');
        $option = $I->grabTextFrom('select option:nth-child(2)');
        $I->selectOption("select", $option);
        $I->fillField('Avatar', 'EmpresaTeste');
        $I->fillField('Estado', '1');
        $option = $I->grabTextFrom('select option:nth-child(2)');
        $I->selectOption("select", $option);
        $I->fillField('Regime', 'EmpresaTeste');
        $I->fillField('Requisito', 'EmpresaTeste');
        $I->fillField('Vagas', '1');
        $I->click('Save');

        $I->see('Empresas');
    }
}

```

Figura 25 - Teste Funcional Registrar Empresa

```

class RegisterPrestadorCest
{
    public function tryCreateUser(FunctionalTester $I)
    {
        // Navegação desde Index até Registo
        $I->amOnPage(Yii::$app->homeUrl);
        $I->see('Lista de Empresas Disponíveis:');
        $I->click('Registar/Login');

        // $I->click('Convidado');
        $I->click('Signup Prestador');

        // Inserção de Dados
        $I->see('Signup Prestador');
        $I->fillField('Username', 'testefuncional');
        $I->fillField('Email', 'testefuncional@emailteste.com');
        $I->fillField('Password', 'testefuncional');
        $I->fillField('Nome', 'NomeTeste');
        $option = $I->grabTextFrom('select option:nth-child(2)');
        $I->selectOption("select", $option);
        $I->fillField('DataNascimento', '1990-01-01');
        $I->fillField('Num Tele', '912345678');
        $I->fillField('Nif', '123456789');
        $I->attachFile('#signupformprestador-avatar', 'logo.png');

        //Final Create
        $I->click('signup-button');
        $I->see('Thank you for registration. Please check your inbox for verification email.');
    }
}

```

Figura 26 - Teste Funcional Registar Prestador

```

class HomeCest
{
    public function checkHome(AcceptanceTester $I)
    {
        $I->amOnPage('workr/frontend/web/');
        $I->wait(2); // wait for page to be opened

        $I->see('Lista de Empresas Disponíveis');

        $I->see('Registar/Login');
        $I->click('Registar/Login');
        $I->wait(2); // wait for page to be opened
        $I->click('Fazer Login');
        $I->wait(2); // wait for page to be opened
        $I->see('Username');
        $I->fillField('LoginForm[username]', 'contratante');
        $I->wait(1); // wait for page to be opened
        $I->fillField('LoginForm[password]', '123123123');
        $I->wait(1); // wait for page to be opened
        $I->click('login-button');
        $I->wait(5); // wait for page to be opened
        $I->see('Lista de Empresas Disponíveis');

    }
}

```

Figura 27 - Teste Aceitacao Login

**Localidades:**

- Lisboa
- Leiria

### Lista de Empresas Disponíveis:

---



**Google**  
 Sexo: Google  
 Data Nascimento: Google  
 Localidade: Lisboa  
 Cargo Disponível: Full-Time

---



**EmpresaTestUpdate**  
 Sexo: EmpresaTestUpdate  
 Data Nascimento: EmpresaTestUpdate  
 Localidade: Leiria  
 Cargo Disponível: Full-Time

---

—

**Testes**  
 Sexo: Testes  
 Data Nascimento: Testes  
 Localidade: Lisboa

**Figura 30 - Página Web Home**

**Workr** Bem Vindo tiago ▾ Logout ▾ Logout (tiago) Sobre Contacto



**Google**  
  
 Nome do Contratante: Contratante 1  
 Estado: **Disponível**  
 Localidade: Lisboa

**Descrição:**  
Google LLC é uma empresa multinacional de serviços online e software dos Estados Unidos. O Google hospeda e desenvolve uma série de serviços e produtos baseados na internet e gera lucro principalmente através da publicidade pelo AdWords.

**Ofertas:**  
 Regime: Teste  
 Requisitos da Empresa: Teste  
 Vagas Disponíveis: 30

[Candidatar-se](#) [Dar Feedback](#)

**Membros:**



**Prestador**  
 Sexo: M  
 Data Nascimento: 2021-11-10

**Figura 29 - Página Web ver Empresa (1/2)**



**tiago**  
 Sexo: M  
 Data Nascimento: 1998-06-24

**Feedback anónimo da empresa:**

**★ Rating: 4/10 ★**  
 Texto: Empresa buéda fixe

**Figura 28 - Página Web ver Empresa (2/2)**

The screenshot shows a web page titled "Candidatura para: Google". It includes fields for "Nome" (Name) containing "tiago", "Texto" (Text), "Id Cv" (CV ID) with a dropdown menu showing "Selecionar o CV" (Select CV), "Id Empresa" (Company ID) with a dropdown menu showing "1", and a green "Candidatar-se" (Apply) button.

**Figura 32 - Página Web Candidatura**

The screenshot shows a web page titled "Feedback a: Google". It includes fields for "Rating" (Rating), "Texto" (Text), "Id Empresa" (Company ID) with a dropdown menu showing "1", and a green "Candidatar-se" (Apply) button.

**Figura 33 - Página Web Feedback Empresa**

The screenshot shows a user profile for "tiago". It displays basic information: Sexo: M, Data Nascimento: 1998-06-24, Email: tiago@gmail.com, Número de Telemóvel: 123456799, and NIF: 123456787. There are buttons for "Editar Perfil" (Edit Profile) and "Ver Trabalhos Atribuidos" (View Assigned Tasks). Below this, a table titled "Pagamentos:" (Payments) lists ten payment entries:

ID Pagamento	Valor	Data do Pagamento	Tipo de Pagamento	Nome do Contratante	Ação
5	3000	2022-01-02 21:23:06	Multibanco	Contratante 2	<button>Report</button>
7	100	2022-01-02 21:23:06	Multibanco	Contratante 2	<button>Report</button>
8	100	2022-01-02 21:23:06	Multibanco	Contratante 2	<button>Report</button>
9	100	2022-01-02 21:23:06	Paypal	Contratante 2	<button>Report</button>
10	1234	2022-01-02 21:28:27	Paypal	Contratante 2	<button>Report</button>

**Figura 31 - Página Web Perfil Prestador**

**Figura 36 - Página Web Serviços Prestador**
**Figura 35 - Página Web Editar Perfil**
**Figura 34 - Página Web Perfil Contratante**

**Update Contratante: 2**

Nome  
Contratante 2

Sexo  
Masculino

Avatar  
/workr/frontend/web/avatar/default-avatar.png

Datanascimento  
2021-11-2

Id User  
32

Id Cargo  
1

Figura 39 - Página Web Editar Perfil

**Candidaturas**

ID Candidatura	Nome do Prestador	Texto	Nome do Ficheiro	Empresa	Ação
16	tiago	Gostava bué de trabalhar aqui, juro!	Curriculo Tiago Ramos	EmpresaTestUpdate	<span style="background-color: green; color: white; padding: 2px;">Aceitar</span> <span style="background-color: red; color: white; padding: 2px;">Recusar</span>

Figura 38 - Página Web Gerir Candidatura

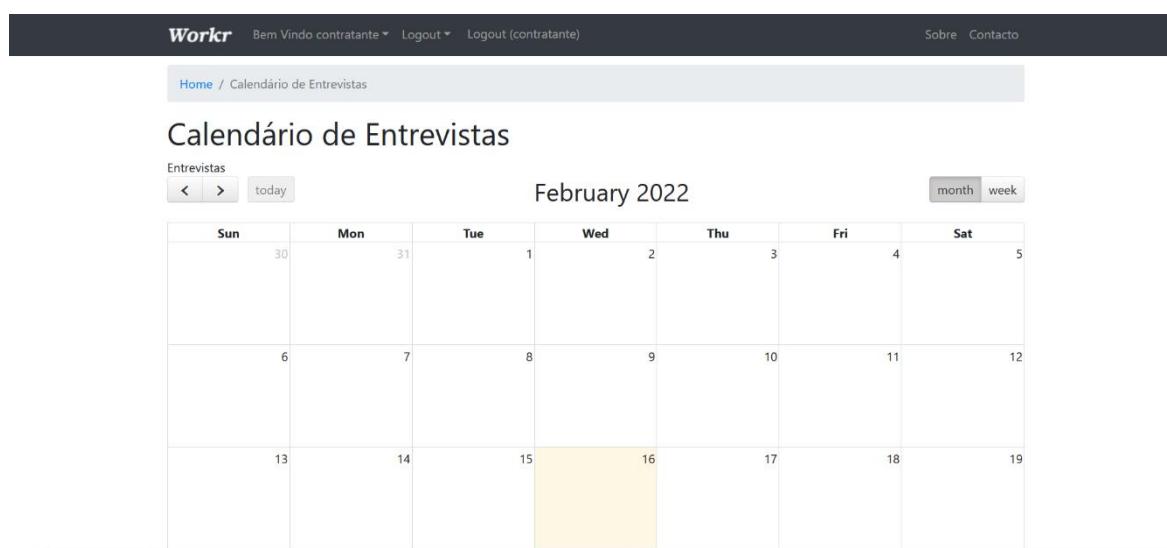


Figura 37 - Página Web Calendário de Entrevistas

Bem Vindo contratante ▾ Logout ▾ Logout (contratante) Sobre Contacto

Home / Pagamentos / Create Pagamento

## Create Pagamento

Valor

Escolher Tipo de Pagamento

-- Escolher Tipo de Pagamento --

Prestador de Servicos

-- Escolher Prestador de Servicos --

**Save**

**Figura 42 - Página Web Criar Pagamento**

Bem Vindo contratante ▾ Logout ▾ Logout (contratante) Sobre Contacto

Home / Ratings / Create Ratings

## Create Ratings

Valor

Descrição

Prestador da Empresa

-- Escolher Prestador da Empresa --

ID do Contratante

2

**Save**

**Figura 41 - Página Web Criar Ratings**

Bem Vindo contratante ▾ Logout ▾ Logout (contratante) Sobre Contacto

Home / Servicos / Create Servicos

## Create Servicos

Requerimento

Data Inicio

Data Fim

Prestador de Servicos

-- Escolher Prestador de Servicos --

**Save**

**Figura 40 - Página Web Atribuir Serviços**

**Starter Page**

Alert! Congratulations!

I am a danger callout!  
There is a problem that we need to fix. A wonderful serenity has taken possession of my entire soul, like these sweet mornings of spring which I enjoy with my whole heart.

Users Registrados: 26 | Contratantes Registrados: 6 | Prestadores Registrados: 5

Quantidade de Ratings Feitos: 5 | Empresas Criadas: 3 | Candidaturas Enviadas: 6

**Registrar Empresa** **Reports** **Ver Pagamentos**

Figura 45 - Página Web Backoffice Home

**Create Empresa**

**Create Empresa**

Nome:

Descricao:

Id Localidade:

Selecione a localidade

Avatar:

Estado:

0  
(0 = Não aparece na lista de procura | 1 = Aparece na lista de procura)

Id Contratante:

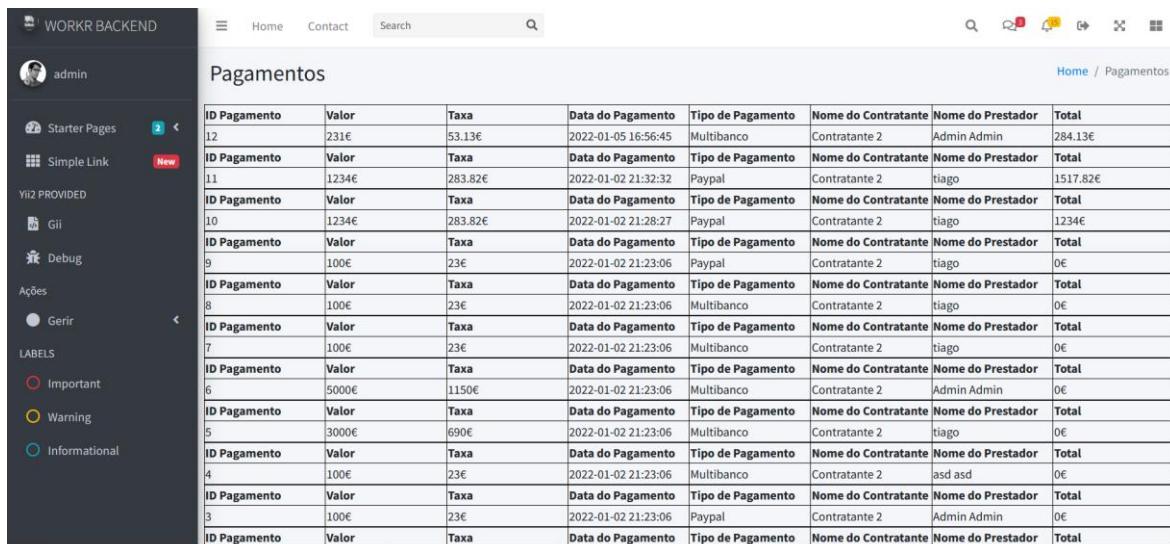
Selecione o Contratante

Figura 44 - Página Web BackOffice Criar Empresa

**Disputas**

ID Disputa	Descrição	ID Pagamento	Contratante	Prestador	Ação
1	Fui mal pago, juro	5	Contratante 2	tiago	Resolver

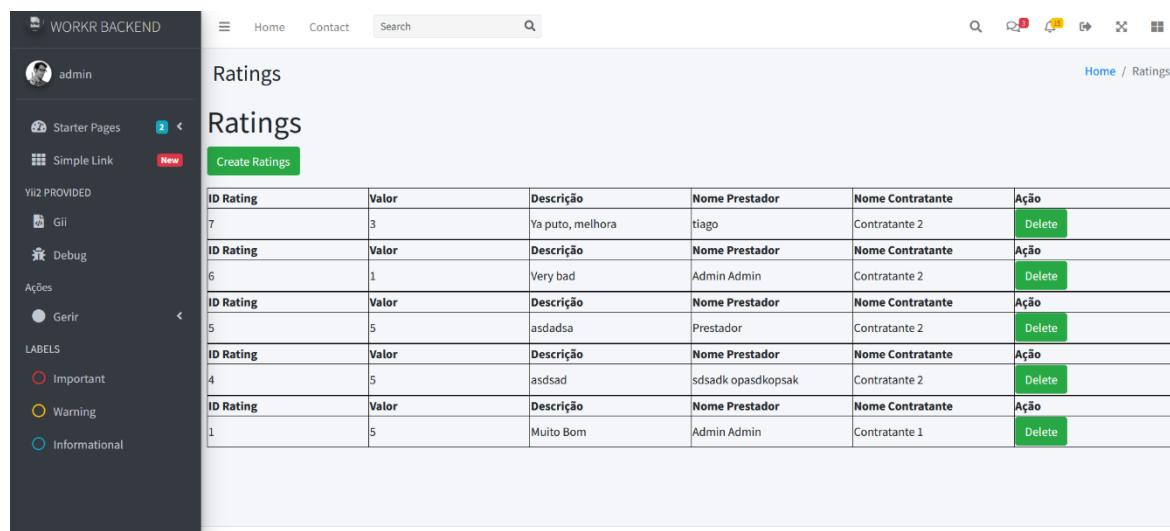
Figura 43 - Página Web BackOffice Resolver Disputas



The screenshot shows the 'Pagamentos' (Payments) section of the Workr BackOffice. The left sidebar includes 'Starter Pages' (2), 'Simple Link' (New), 'Yii2 PROVIDED', 'Gii', 'Debug', 'Ações' (Actions), 'Gerir' (Manage), and 'LABELS' (Important, Warning, Informational). The main content area displays a table with columns: ID Pagamento, Valor, Taxa, Data do Pagamento, Tipo de Pagamento, Nome do Contratante, Nome do Prestador, and Total. The table lists 17 rows of payment data.

ID Pagamento	Valor	Taxa	Data do Pagamento	Tipo de Pagamento	Nome do Contratante	Nome do Prestador	Total
12	231€	53.13€	2022-01-05 16:56:45	Multibanco	Contratante 2	Admin Admin	284.13€
11	1234€	283.82€	2022-01-02 21:32:32	Paypal	Contratante 2	Tiago	1517.82€
10	1234€	283.82€	2022-01-02 21:28:27	Paypal	Contratante 2	Tiago	1234€
9	100€	23€	2022-01-02 21:23:06	Paypal	Contratante 2	Tiago	0€
8	100€	23€	2022-01-02 21:23:06	Multibanco	Contratante 2	Tiago	0€
7	100€	23€	2022-01-02 21:23:06	Multibanco	Contratante 2	Tiago	0€
6	5000€	1150€	2022-01-02 21:23:06	Multibanco	Contratante 2	Admin Admin	0€
5	3000€	690€	2022-01-02 21:23:06	Multibanco	Contratante 2	Tiago	0€
4	100€	23€	2022-01-02 21:23:06	Multibanco	Contratante 2	asd asd	0€
3	100€	23€	2022-01-02 21:23:06	Paypal	Contratante 2	Admin Admin	0€
2	100€	23€	2022-01-02 21:23:06	Paypal	Contratante 2	Tiago	0€

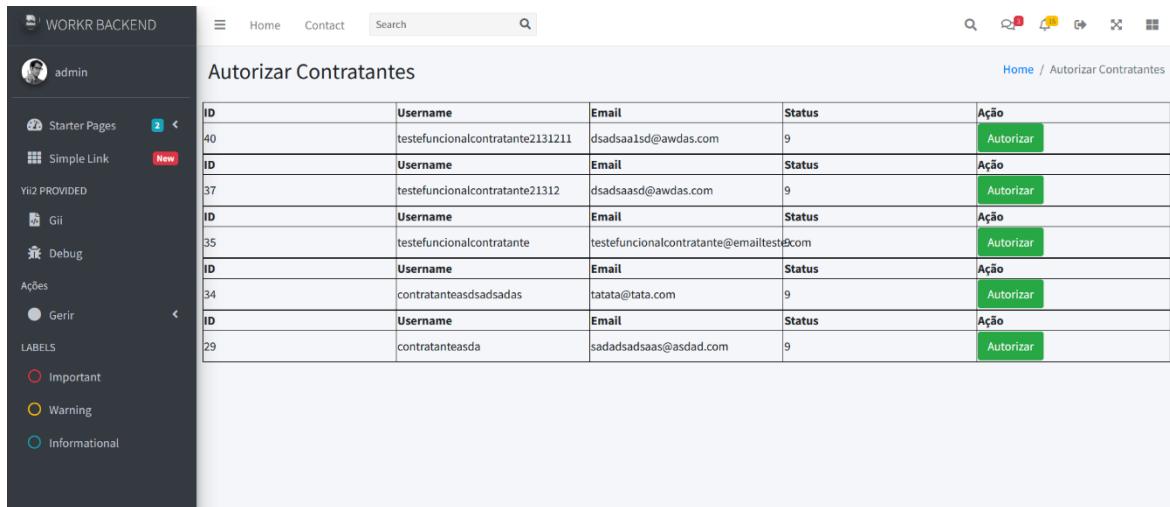
Figura 48 - Página Web BackOffice Histórico entre ambas as partes



The screenshot shows the 'Ratings' section of the Workr BackOffice. The left sidebar includes 'Starter Pages' (2), 'Simple Link' (New), 'Yii2 PROVIDED', 'Gii', 'Debug', 'Ações' (Actions), 'Gerir' (Manage), and 'LABELS' (Important, Warning, Informational). The main content area displays a table with columns: ID Rating, Valor, Descrição, Nome Prestador, Nome Contratante, and Ação (Action). The table lists 7 rows of rating data.

ID Rating	Valor	Descrição	Nome Prestador	Nome Contratante	Ação
7	3	Ya puta, melhora	tiago	Contratante 2	Delete
6	1	Very bad	Admin Admin	Contratante 2	Delete
5	5	asdadsa	Prestador	Contratante 2	Delete
4	5	asdsad	sdsadk opasdokpsak	Contratante 2	Delete
1	5	Muito Bom	Admin Admin	Contratante 1	Delete

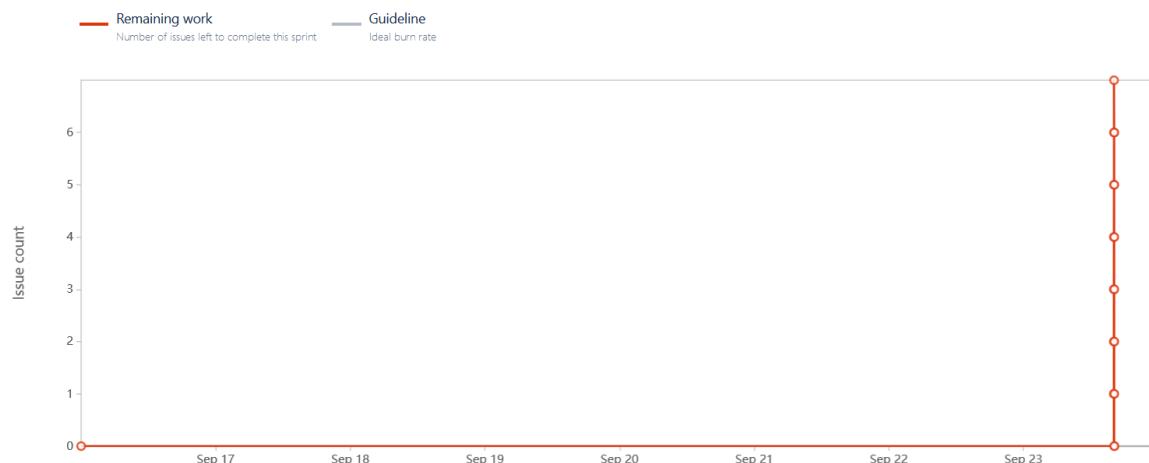
Figura 47 - Página Web BackOffice Ratings



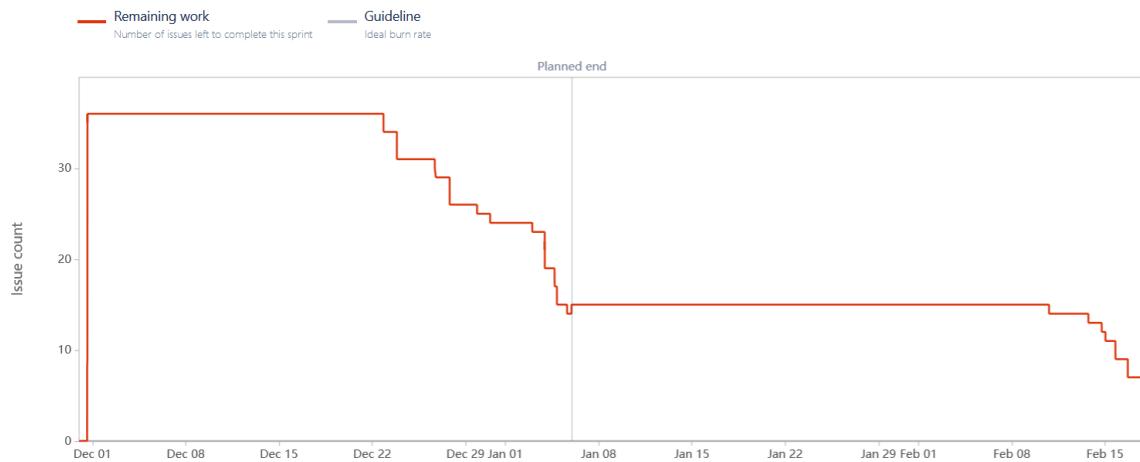
The screenshot shows the 'Autorizar Contratantes' (Authorize Contractors) section of the Workr BackOffice. The left sidebar includes 'Starter Pages' (2), 'Simple Link' (New), 'Yii2 PROVIDED', 'Gii', 'Debug', 'Ações' (Actions), 'Gerir' (Manage), and 'LABELS' (Important, Warning, Informational). The main content area displays a table with columns: ID, Username, Email, Status, and Ação (Action). The table lists 9 rows of contractor authorization data.

ID	Username	Email	Status	Ação
40	testefuncionalcontratante2131211	dsadsaa1sd@awdas.com	9	Autorizar
37	testefuncionalcontratante21312	dsadsaasd@awdas.com	9	Autorizar
35	testefuncionalcontratante	testefuncionalcontratante@emailteste.com	9	Autorizar
34	contratanteadsadsadas	tatata@tata.com	9	Autorizar
29	contratanteadsa	Sadadsadsaas@asdad.com	9	Autorizar

Figura 46 - Página Web BackOffice Autorizar Contratante

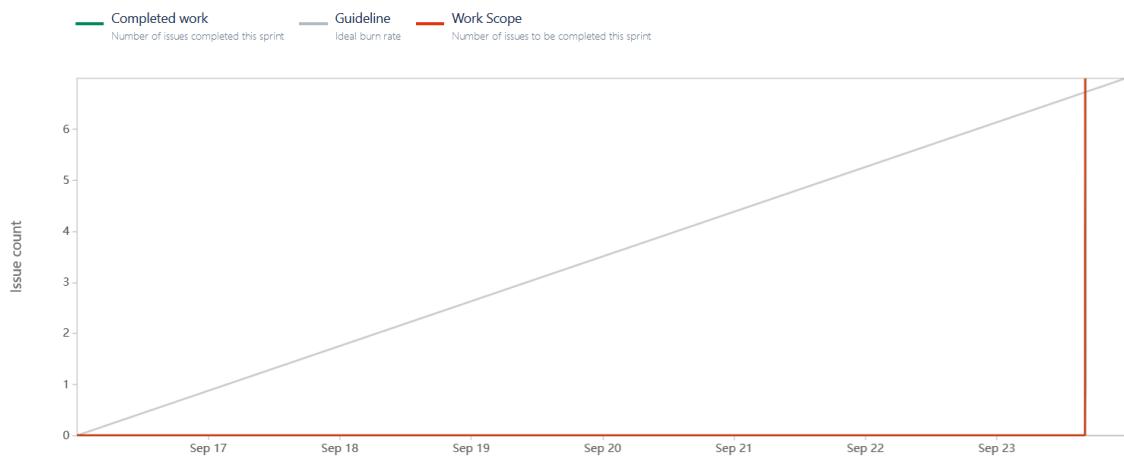
**Figura 51 - Cumulative Flow Diagram****Date** - 16 September 2021 - 23 September 2021**Figura 50 - Burndown Report (Sprint 1)****Date** - 27 September 2021 - 11 October 2021**Figura 49 - Burndown Report (Sprint 2)**

**Date** - 30 November 2021 - 6 January 2022



**Figura 54 - Burndown Report (Sprint 3)**

**Date** - 16 September 2021 - 23 September 2021



**Figura 53 - Sprint Burndown Chart (Sprint 1)**

**Date** - 27 September 2021 - 11 October 2021



**Figura 52 - Sprint Burndown Chart (Sprint 2)**

Date - 30 November 2021 - 6 January 2022



Figura 55 - Sprint Burndown Chart (Sprint 3)