

# Software Developer / Full Stack

Licenciatura em Eng. Informática

Estágio-Projeto 2020/2021



UNIVERSIDADE DE ÉVORA  
ESCOLA DE CIÊNCIAS E TECNOLOGIA

DEPARTAMENTO DE INFORMÁTICA



---

## Dinis Marques dos Matos

Orientador na Empresa: Ricardo Torrão

Orientador no Departamento: Lígia Ferreira

*Participar no desenvolvimento de uma aplicação web (cloud) tendo contacto com o processo de desenvolvimento e tecnologias usadas*

---

---

*Évora, 22 de Junho de 2021*

---

# Tabela de conteúdos

<b>1 Introdução</b>	<b>4</b>
1.1 Enquadramento	4
1.2 Objetivos	4
1.3 Contribuições	4
1.4 Estrutura do documento	4
<b>2 Ambiente empresarial</b>	<b>6</b>
<b>3 Ambiente de desenvolvimento</b>	<b>7</b>
3.1 Ambiente técnico	7
3.1.1 Software	7
3.2 Ambiente aplicacional	8
3.3 Metodologia de trabalho	8
<b>4 Trabalho desenvolvido</b>	<b>9</b>
4.1 Descrição detalhada	9
<b>5 Avaliação Crítica</b>	<b>10</b>
<b>Referências</b>	<b>11</b>

---

# 1 Introdução

Este documento apresenta todos os aspetos sobre o estágio (Enquadramento, Objetivos, Contribuições, Ambiente Empresarial, Ambiente de Desenvolvimento, Trabalho Desenvolvido e Avaliação Crítica) que foi realizado na Licenciatura em Engenharia Informática da Universidade de Évora, no ano letivo 2020/2021.

## 1.1 Enquadramento

O estágio foi realizado na empresa Two Impulse, de modo remoto, e fui integrado numa equipa de projeto que estava a desenvolver um produto para a companhia Swiss Reinsurance Company, mais conhecida por Swiss Re, em que esta é a maior companhia de seguros e resseguros do mundo, sendo operacional em mais de 30 países.

## 1.2 Objetivos

Durante o período de aprendizagem, foi posto um objetivo de trabalho para ajudar a equipa em que fui inserido. Esta meta consistia em participar no desenvolvimento de uma aplicação web (cloud) tendo contacto com o processo de desenvolvimento e tecnologias usadas, e para esse efeito, implementar testes unitários para as aplicações *backend* e *frontend* foi a tarefa atribuída.

## 1.3 Contribuições

No decurso do estágio contribui de forma a conseguir atingir o objetivo, em que este era ajudar a aumentar cobertura do código do projeto estar acima dos 80%, como também reportar alguns *bugs* no código, resolver menores e maiores *bugs* no tempo restante do estágio, tendo em conta toda a informação que a ferramenta SonarQube disponibiliza.

## 1.4 Estrutura do documento

Ao longo do resto do documento, está indicado tópicos relevantes acerca do período de estágio como:

- Ambiente Empresarial - Fala um pouco sobre a empresa sobre a qual o estágio foi aplicado, a equipa em que estava inserido e a minha posição nessa mesma.
- Ambiente de Desenvolvimento - Nesta parcela está descrito como o ambiente de software foi utilizado e a minha apreciação.

- 
- Trabalho Desenvolvido - Esta divisão detalha em relação ao trabalho realizado ao longo do estágio, com a devida ordem respetiva de tempo, e alguns exemplos mais específicos para descrever melhor sobre as tarefas feitas.
  - Avaliação Crítica - Uma última segmentação do documento no qual é dito a minha apreciação relativamente ao estágio sobre o meu aproveitamento, no que diz a conhecimento e métodos de trabalho, e também sobre a empresa na qual eu estava colocado.

---

## 2 Ambiente empresarial

A Two Impulse é uma empresa fundada em 2016 com o objetivo de ajudar empresas a tornarem-se mais digitais, a ser *data-driven*, a tomar melhores decisões, a descobrir novas oportunidades e modelos de negócios.

A empresa é dividida por vários grupos e a equipa em que fiquei colocado estava a fazer um projeto para a companhia Swiss Re, em que esta foi composta por *Project Manager*, *Backend Development*, *Frontend Development* e *Tests*. Nesta última área foi a qual em que eu fui inserido para ajudar.

---

## 3 Ambiente de desenvolvimento

No que se refere ao ambiente de desenvolvimento, neste caso de software, estive relativamente confortável com as ferramentas utilizadas. Apesar de não ter tido qualquer experiência anteriormente com as tecnologias a utilizar, consegui adaptar-me relativamente bem e adorei trabalhar com o Pycharm, tendo em consideração a ajuda que disponibiliza para a linguagem Python.

### 3.1 Ambiente técnico

#### 3.1.1 Software

As ferramentas usadas a nível de software durante a fase de aprendizagem foram:

- Pycharm
- Docker
- PyTest
- Coverage.py
- SonarQube
- Azure DevOps

O Pycharm é um IDE (*integrated development environment*) mais focado para linguagem de programação Python, em que este foi usado para programar.

Em termos simples, Docker é uma plataforma de software que simplifica o processo de construção, execução, gestão e distribuição de aplicativos. Faz o processo ao virtualizar o sistema operacional do computador no qual está instalado e em execução. O Docker foi utilizado para correr uma imagem do ambiente em que o mesmo vai correr.

PyTest é uma estrutura de teste que permite aos utilizadores escrever código de teste com a linguagem de programação Python. Ajuda a escrever casos de teste simples e escaláveis para bancos de dados, APIs ou UI.

Coverage.py é uma ferramenta para medir a cobertura de código de programas Python ao analisar quais as linhas de código que foram executadas e aquelas que não foram utilizadas.

SonarQube é uma plataforma de código aberto desenvolvida pela SonarSource para inspeção contínua da qualidade do código para realizar revisões automáticas com análise

---

estática do código para detectar *bugs*, código duplicado e vulnerabilidades de segurança em mais de 20 linguagens de programação.

Azure DevOps é uma plataforma de Software como serviço (SaaS) da Microsoft que fornece um conjunto de ferramentas DevOps de “ponta a ponta” para o desenvolvimento e implementação de software. Também integra-se com a maioria das ferramentas líderes no mercado e é uma excelente opção trabalhar com uma cadeia de ferramentas DevOps.

## 3.2 Ambiente aplicacional

O trabalho desenvolvido foi produzido para o projeto SDG Calculator, uma ferramenta que está a ser desenvolvida para pontuar e avaliar o progresso da seguradora em relação aos SDGs (Sustainable Development Goals). O papel estabelecido para desempenhar nesta proposta foi ajudar a criar testes unitários, a validar o código do projeto e para esse efeito foram atribuídos objetivos para o estágio.

## 3.3 Metodologia de trabalho

A organização de trabalho adotada pela empresa foi bastante estruturada para o estágio. Tendo em conta o horário definido, todas as semanas participava numa reunião em que eram descritas novidades e atualizações do que foi feito por cada colega da equipa para existir uma boa organização de trabalho e do resto dos colegas da mesma equipa estarem a par do progresso nas restantes partes do projeto.

Implementar testes unitários segundo a metodologia TDD (*Test Driven Development*) seguindo o plano definido para as features e user stories do projeto a cada sprint era o plano inicial, porém devido ao código já criado e às prioridades estabelecidas, esta metodologia foi aplicada apenas uma vez.

Recebi tarefas em que grande parte destas atribuídas a mim foram feitas pelo responsável da empresa, porém alguns colegas com quem estava a trabalhar, aconselharam certas atribuições para obter um maior proveito durante o estágio.

O método utilizado para contribuir para a empresa foi através de *pull requests* pois não só é a maneira mais prática como também é mais fácil para rever o código que foi desenvolvido. Durante o estágio, tive dificuldades em assuntos específicos, contudo os restantes membros da equipa de trabalho sempre ajudaram-me quando precisei.



---

## 4 Trabalho desenvolvido

Quando entrei para a empresa, estive em formação durante 2 semanas para entender sobre o projeto e quais as ferramentas que iria precisar e trabalhar na qual eu seria inserido para ajudar. Durante esse intervalo, escrevi um documento cujo explicava como instalar e executar o projeto, do jeito que me foi ensinado, para que a próxima pessoa que necessitar de formação possa ler o documento já criado e poupar tempo, tanto para a pessoa que dá como para a que recebe preparação.

Depois da fase de treinamento, estive a produzir testes unitários para a validação de código produzido até conseguir o objetivo ao qual me foi imposto pela empresa. Logo após o cumprimento do objetivo ainda resolvi menores e maiores bugs que tinham sido reportados para serem consertados.

### 4.1 Descrição detalhada

De um ponto de vista mais detalhado, seguem-se alguns exemplos na qual possam ser entendidos melhor como:

- Documento para formação - Documento composto com pré-requisitos, como instalar o software requerido e como utilizá-lo, com devidas imagens e legendas de modo a ser mais perceptível.
- Teste unitário - Teste que valida se um método, tendo em conta um *input* dado, ao executar devolve um *output* esperado. Se houver dependências de outros métodos, esses serão simulados através de *mocks*.

---

## 5 Avaliação Crítica

Apesar de saber que testes unitários eram importantes para a validação de código, não tinha noção do quanto eram úteis e essenciais como sei agora. Ao trabalhar com uma equipa grande fez-me entender que testes são tão necessários quanto o próprio código, e que não só valida o código em si, como também poupa bastante tempo, evita *bugs*, erros dispensáveis e melhora o fluxo de trabalho de todos que estão dependentes do código pois tudo encontra-se mais organizado.

Relativamente a tópicos como trabalho em equipa e a comunicação, do ponto de vista de alguém que não tinha qualquer experiência sobre trabalhar em uma equipa grande, foi excelente. Não só a organização de trabalho como também o ambiente de trabalho em que estive envolvido, pois quando necessitasse de algo, tivesse uma dúvida ou alguma curiosidade sobre o projeto, era sempre bem recebido.

---

## Referências

- [1] Pycharm. <https://www.jetbrains.com/pycharm>  
[Online; accessed 22-June-2021]
- [2] Coverage.py. <https://coverage.readthedocs.io/en/coverage-5.5/>  
[Online; accessed 22-June-2021]
- [3] SonarQube. <https://www.sonarqube.org/>  
[Online; accessed 22-June-2021]
- [4] Azure DevOps. <https://azure.microsoft.com/pt-pt/services/devops/>  
[Online; accessed 22-June-2021]
- [5] Docker. <https://www.docker.com/products/docker-desktop>  
[Online; accessed 22-June-2021]
- [6] PyTest. <https://docs.pytest.org/en/6.2.x/>  
[Online; accessed 22-June-2021]