



IPG Politécnico
da Guarda
Escola Superior
de Tecnologia e Gestão

RELATÓRIO DE ESTÁGIO

Curso Técnico Superior Profissional em
Testes de Software

João Pedro Salvador Lopes

julho | 2016





Escola Superior de Tecnologia e Gestão

Instituto Politécnico da Guarda

RELATÓRIO DE ESTÁGIO

JOÃO PEDRO SALVADOR LOPES

RELATÓRIO PARA A OBTENÇÃO DO DIPLOMA DE TÉCNICO SUPERIOR

PROFISSIONAL EM TESTES DE SOFTWARE

JUNHO/2016

Elementos identificativos

Aluno:

João Pedro Salvador Lopes

Nº 1011853

Tesp: Testes de *software*

Estabelecimento de Ensino:

Escola Superior de Tecnologia e Gestão

Instituto Politécnico da Guarda

Instituição Acolhedora do Estágio:

Nome: Altran - Global Delivery Centre

Morada: Centro de Negócios e Serviços, Praça Amália Rodrigues 6230-350 Fundão

Duração do Estágio:

Início: 1 de Março de 2015

Fim: 30 de Junho de 2016

Orientador de Estágio:

Nome: José Alberto Quitério Figueiredo

Grau académico: Mestre

Agradecimentos

Gostaria de deixar um enorme agradecimento ao Instituto Politécnico da Guarda, mais concretamente á Escola Superior de Tecnologia e Gestão, pela oportunidade proporcionada e pelo enriquecimento que me devolveu a nível pessoal e profissional.

Agradeço, ao meu orientador de estágio Prof José Alberto Quitério Figueiredo pelo seu profissionalismo, simpatia e dedicação, o que representou um elevado marco para que fosse possível a concretização do meu estágio.

Em especial, gostaria de agradecer a toda a minha família, amigos e colegas de curso, porque sem o seu apoio incondicional não seria possível ter atingido o fim deste percurso.

Por último, gostaria de demonstrar o meu apreço pela equipa Daikin iTC. A equipa que me acolheu e integrou no seu projeto, e que me possibilitou trabalhar num ambiente saudável. Por me permitirem aplicar todos os conceitos adquiridos, e enriquecer ainda mais toda a minha experiência.

Plano de Estágio Curricular

Sendo o projeto executado segundo a metodologia Agile, utilizando a técnica *SCRUM*, tive os seguintes objetivos no decurso do estágio:

1. Integração na equipa de projeto;
2. Conhecer os objetivos do projeto;
3. Conhecer os objetivos da equipa de testes no projeto;
4. Criação de planos de testes funcionais (Manuais) de acordo com as metodologias Altran;
5. Criação de relatórios de testes funcionais (Manuais) de acordo com as metodologias Altran;
6. Comunicação de defeitos (erros) de forma estruturada e concisa;
7. Participação na criação de testes automáticos, utilizando a *framework* SpecFlow e a ferramenta Visual Studio 2015.

Índice Geral

Elementos identificativos	1
Agradecimentos.....	2
Plano de Estágio Curricular.....	3
Glossário de Abreviaturas	9
Capítulo I - Introdução	10
1 – Introdução	11
Capítulo II – A instituição: Altran - Global Delivery Centre.....	12
2.1 – Grupo Altran	13
2.2 – Altran em Portugal	15
Capítulo III – Enquadramento teórico.....	16
3.1 – Introdução	17
3.2 – O que são os testes de <i>software</i>	18
3.3 – Tipo de testes de <i>software</i>	19
3.3.1 – Testes Manuais.....	19
3.3.2 – Testes Integração.....	19
3.3.3 – Testes Sistema	19
3.3.4 – Teste de Regressão.....	19
3.3.5 – Testes de unitários.....	19
3.3.6 – Testes de performance.....	19
3.4 – Atividade inerentes aos testes de <i>software</i>	20
3.5 - Ferramentas de teste.....	21
3.5.1 – Benefícios e desvantagens da utilização de ferramentas de teste.....	21
3.5.2 – Tipos de ferramentas de teste	21
3.6 - Metodologias de desenvolvimento de <i>software</i>	22
3.6.1 – Metodologia Ágil	23
Capítulo IV – O projeto: Daikin iTC	25
4.1 – O cliente: Daikin	26

4.2 – Âmbito e descrição do projeto	27
4.3 – Funcionamento interno do projeto	28
4.3.1 – Metodologia de desenvolvimento	28
4.3.2 – Constituição atual da equipa.....	28
4.3.3 – Fluxo de trabalho.....	29
4.3.4 - Fluxo de reportação de defeitos	30
4.4 – Ferramentas utilizadas no âmbito do projeto	31
4.4.1 – Visual Studio Pro 2015	31
4.4.2 – NUnit.....	31
4.4.3 – Jenkins.....	31
4.4.4 – SpecFlow.....	31
4.4.5 – Selenium.....	32
4.4.6 – Jira	32
4.4.7 – Gherkin.....	32
4.4.8 – TICS	33
Capitulo V – Trabalho desenvolvido	34
5.1 – Introdução	35
5.2 – Trabalho desenvolvido	36
5.2.1 – Configuração do Ambiente de teste	36
5.2.2 – Desenho de plano de testes.....	36
5.2.3 – Execução e reportação de falhas	37
5.3 – Ferramentas utilizadas.....	38
5.3.1 - Swagger.....	38
5.3.2 – Visual Studio e SpecFlow	38
5.3.3 – Browser <i>developer</i> tools.....	39
5.4 – Formações e eventos	40
5.4.1 - Induction Day.....	40
5.4.2 – Evento Ignite Your Future.....	41

5.4.3 – Formação linguística em Francês	42
5.4.4 – Formação Java.....	43
Conclusão.....	45
Bibliografia	46
Anexos.....	48
Anexo 1 – Instalações da Altran no Fundão	49
Anexo 2 – Página de login da aplicação	51

Índice de Tabelas

Tabela 1 - Tipos de ferramentas de teste	21
Tabela 2 - Equipa Daikin iTC	28
Tabela 3 - Tabela Resumo Fluxo de Trabalho	29

Índice de Ilustrações

Figura 1- Grupo Altran no mundo [1].....	13
Figura 2 - Domínios tecnológicos Grupo Altran [1]	13
Figura 3 - Setores de atividade predominantes do Grupo Altran [1]	14
Figura 4 - Clientes Altran [1]	14
Figura 5 - Historia da Altran em Portugal [2]	15
Figura 6- Setores de atividade predominantes do Grupo Altran em Portugal [1]	15
Figura 7 - Tempo vs Custo de encontrar erros [3]	18
Figura 8 - Esquema Processo de Teste [4]	20
Figura 9 - Scrum.....	24
Figura 10 - Presença da Daikin no Mundo [5]	26
Figura 11 - Daikin Cloud Service [6].....	27
Figura 12 - Metodologia de desenvolvimento do projeto [7].....	28
Figura 13 - Fluxo de reportação de defeitos	30
Figura 14 - Test plan	36
Figura 15 - Exemplo estrutura relatório de erros	37
Figura 16 – Swagger UI	38
Figura 17 - Visual Studio: SpecFlow	39
Figura 18 - Google Chrome DevTools Window [1]	39
Figura 19 - Cartaz Ignite Your Future.....	41

Glossário de Abreviaturas

TESP – Técnico Superior Profissional

TEST SUIT - Conjunto de casos de teste

TEST DATA – Conjunto de dados de teste

TESTWARE – Conjunto de ficheiros relacionados com a atividade de teste

SCRUM – Modelo iterativa ou incremental da metodologia Ágil

SPRINTS – Representa um ciclo na metodologia Ágil

BACKLOG – Lista de funcionalidades a serem implementadas

SPRINT PLANNING – Reunião de planeamento da *Sprint*

PRODUCT OWNER – Pessoa responsável pela representação do cliente

DAILY SCRUM – Reunião diária

RESTFUL - Representational State Transfer

HTML – Hypertext Markup Language, Linguagem de desenvolvimento Web.

CSS - Cascading Style Sheets, utilizado para definição de folhas de estilo de páginas Web.

JAVASCRIPT - Linguagem orientada a objetos utilizada no desenvolvimento web.

Capitulo I - Introdução

1 – Introdução

O presente relatório de estágio descreve as atividades desenvolvidas no âmbito do estágio curricular do Curso Técnico Superior de Testes de *Software*.

Este estágio realizou-se na Altran - Global Delivery Centre localizada no Fundão entre os dias 1 de março de 2016 e 30 de junho de 2016.

Durante o período de estágio foi possível a aplicação dos conhecimentos adquiridos no curso bem como a aquisição de novas valências técnicas e pessoais.

Capitulo II – A instituição: Altran - Global Delivery Centre

2.1 – Grupo Altran

O Grupo Altran é uma multinacional francesa, presente no mercado a mais de 30 anos.

Com mais de 24.000 colaboradores, 500 contas-chave a nível mundial estando presente em mais de 20 países espalhados por 3 continentes como pode ser observado na Figura 1.



Figura 1- Grupo Altran no mundo [1]

As soluções tecnológicas do Grupo Altran abrangem diversos domínios desde o desenvolvimento de produtos inovadores bem como sistemas de informação, entre outros como demonstra a Figura 2:



Figura 2 - Domínios tecnológicos Grupo Altran [1]

No mercado o Grupo Altran está presente em diversos setores de atividade sendo que o setor com maior ênfase de mercado é o de aeroespacial, defesa & ferrovias, representando cerca 26% dos seus clientes como demonstra a Figura 3.



Figura 3 - Setores de atividade predominantes do Grupo Altran [1]

Dentro do leque de clientes que confiaram os seus projetos ao Grupo Altran podemos encontrar as seguintes empresas representadas na Figura 4, entre outras:



Figura 4 - Clientes Altran [1]

2.2 – Altran em Portugal

O Grupo Altran está presente em Portugal desde 1998, tendo consolidado a sua marca em 2009, com a fusão da Altran CIS e a Altior, como demonstra a Figura 5.



Figura 5 - Historia da Altran em Portugal [2]

Em Portugal o Grupo Altran conta com mais de 890 colaboradores, dispersos por zonas geográficas distintas: Porto, Fundão e Lisboa.

Em Portugal a ênfase de mercado encontra-se no setor das telecoms & media representando cerca de 35% dos seus clientes algo que pode ser observado na Figura 6:

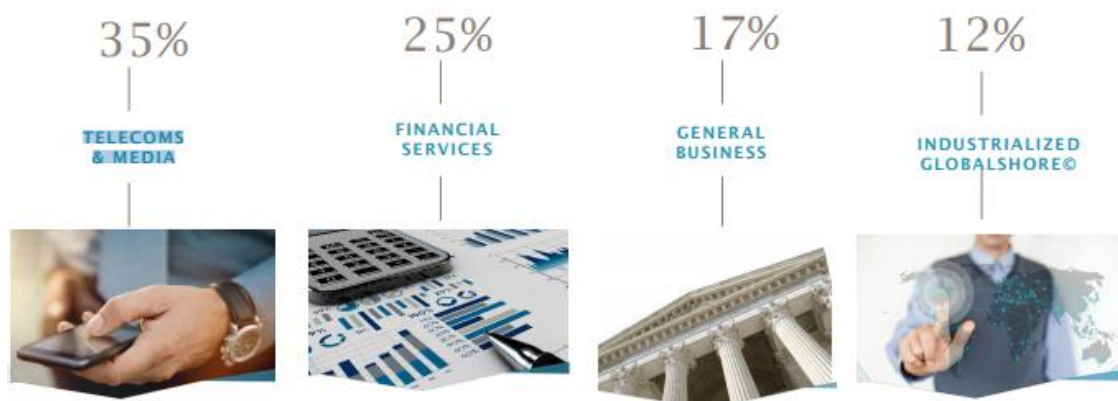


Figura 6- Setores de atividade predominantes do Grupo Altran em Portugal [1]

Capitulo III – Enquadramento teórico

3.1 – Introdução

No presente capítulo passarei a abordar alguns dos conceitos tratados no curso Tesp Testes de Software que considero de maior relevância para um melhor entender das atividades desenvolvidas no âmbito do estágio curricular bem como o presente relatório.

Dentro desses conceitos enquadram-se os seguintes tópicos:

1. O que são testes de software;
2. Tipo de testes de software;
3. Atividade inerentes aos testes de software;
4. Ferramentas utilizadas;
5. Metodologias de desenvolvimento de software.

3.2 – O que são os testes de *software*

Os testes de *software* consistem num processo no qual se demonstra que o *software* desempenha as funções previstas de forma correta, de acordo com os requisitos definidos.

Consequentemente os testes devolvem um valor acrescentado ao *software*, pela identificação e consequente eliminação das falhas encontradas.

As falhas ao serem encontradas previamente á entrega do sistema faz com que o custo do desenvolvimento do sistema seja menor, pois ao serem encontrados precocemente evita que seja necessário refazer o código já desenvolvido.

Um exemplo prático é um erro encontrado durante a fase de análise de requisitos, que possivelmente vai ser facilmente resolvido. É gerado um novo documento com as alterações necessárias para a correção da falha, sem custos de maior para o cliente.

O mesmo já não acontecia caso este erro fosse encontrado apenas depois do lançamento do produto. O custo pode vir a ser descomunal caso este implique alterações ao nível do design, alterações no código, etc.

Pode se concluir que o custo da correção de um defeito é proporcional ao tempo decorrido até que este seja encontrado como é demonstrado na Figura 7.

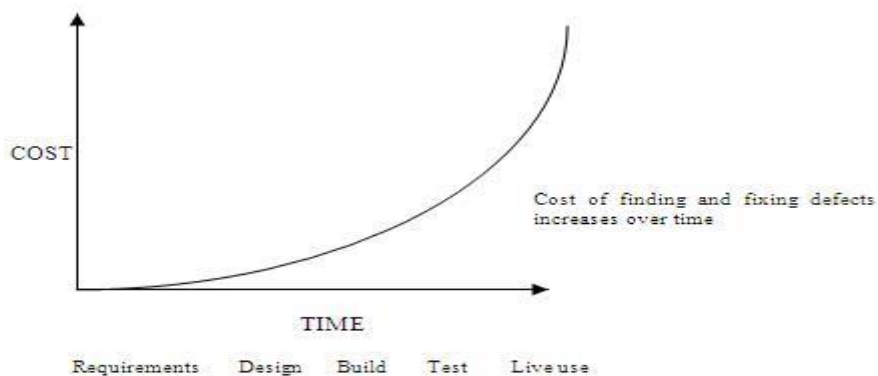


Figura 7 - Tempo vs Custo de encontrar erros [3]

3.3 – Tipo de testes de *software*

3.3.1 – Testes Manuais

São testes que avaliam o comportamento da aplicação, comparando o resultado da sua execução com o resultado esperado.

Este tipo de teste é transversal a todos os tipos de testes (integração, sistema, regressão).

3.3.2 – Testes Integração

São testes que garantem o funcionamento de diferentes componentes do sistema em conjunto conforme descrito nas especificações do sistema.

3.3.3 – Testes Sistema

Testa o sistema como um todo, garantido que o que foi desenvolvido cumpre com o âmbito do sistema.

Este tipo de teste é frequentemente utilizado numa fase de entrega do sistema ou de uma funcionalidade específica, pois garante este vai ser entregue de acordo com as especificações e propósito do mesmo.

3.3.4 – Teste de Regressão

São testes que são executados quando existe alguma alteração no sistema, desde correção de erros, implementação de novas funcionalidades, e outras.

Este tipo de teste garante que as alterações que foram efetuadas no sistema não provocaram a existência de novos *bugs*.

3.3.5 – Testes de unitários

Este tipo de teste é normalmente escrito e executado pelos desenvolvedores do código. Visão garantir que um pequeno bloco do código está a funcionar corretamente. Isto significa por exemplo que quando é criada uma função com um determinado input esta retorna um valor esperado. Garante que quando o input é inválido a função é capaz de tratar esse erro, etc.

3.3.6 – Testes de performance

Este é um tipo de teste que é normalmente executado antes de uma *release*. Visa garantir que o sistema funciona corretamente em determinadas condições com a carga excessiva no sistema, tráfego excessivo na rede o tempo de resposta, etc.

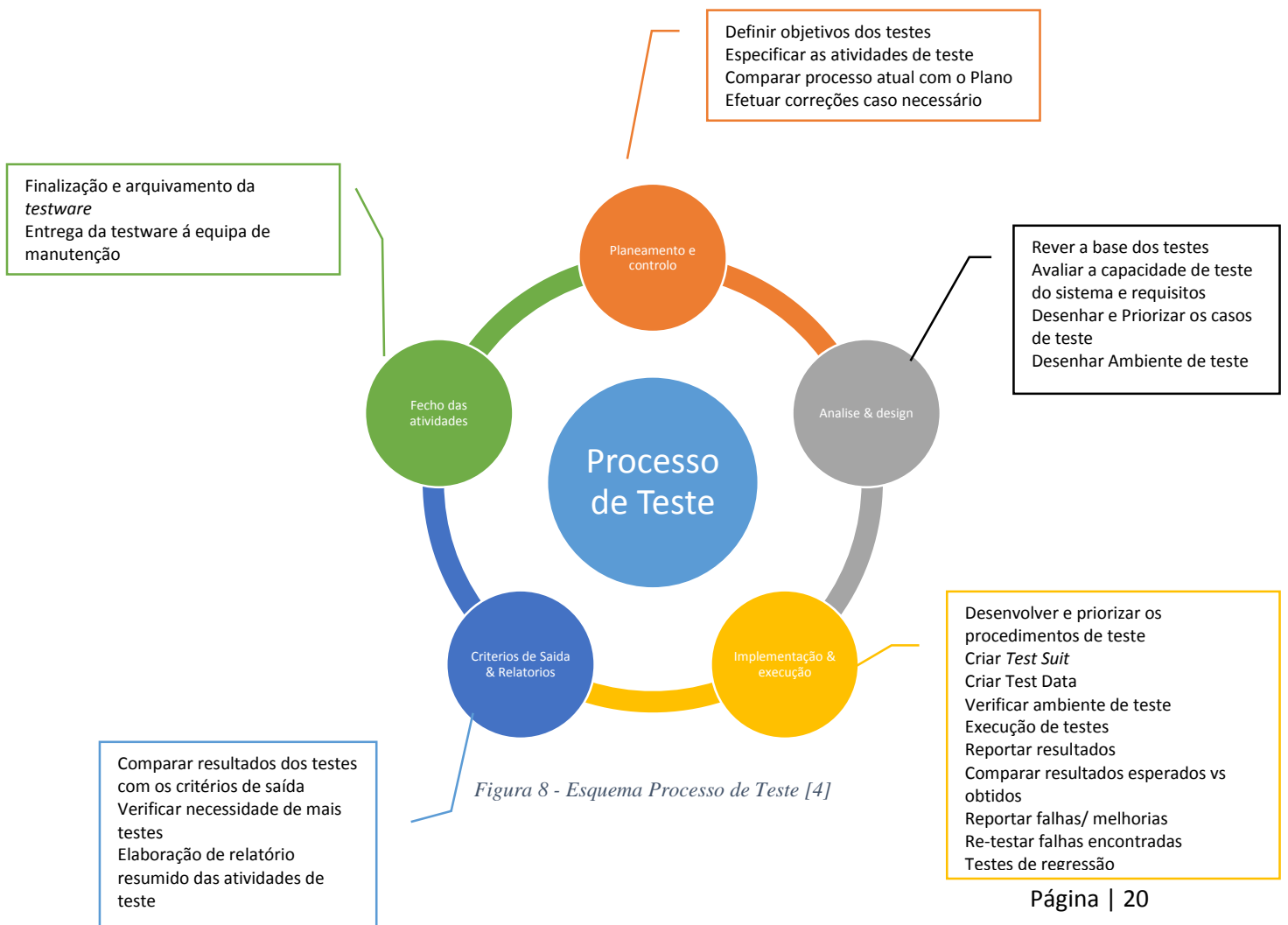
3.4 – Atividade inerentes aos testes de *software*

Os testes de *software* é um processo que envolve diversas atividades, mais do que apenas a sua execução.

Existem atividades de teste, antes e após a execução do teste. Estas atividades podem incluir:

- Planeamento e Controlo;
- Condições de teste (Análise);
- Design de Casos de Teste;
- Execução;
- Verificar resultados;
- Avaliação dos critérios de saída;
- Reportar o sistema de teste usado e as conclusões retiradas;
- Finalização ou fecho das atividades.

O seguinte esquema da Figura 8 sintetiza a informação abordada:



3.5 - Ferramentas de teste

3.5.1 – Benefícios e desvantagens da utilização de ferramentas de teste

A utilização de uma ferramenta de teste dentro de uma organização por acaratar diversos benefícios ou prejuízos para os projetos que vão utilizar. Devido a esse facto é necessário ponderar os prós e contras da utilização da mesma.

Entre as vantagens da utilização da ferramenta podemos encontrar:

- Redução de trabalho repetitivo
- Ajuda na automatização de planos de teste
- Facilidade no acesso á informação sobre os testes
- Facilidade em agendar a execução de casos de teste

Porem entre os pontos menos positivos podemos considerar:

- Expectativas irreais sobre a ferramenta
- Subestimar o tempo, custo e esforço para a introdução inicial de uma ferramenta
 - Subestimar o tempo e o esforço necessários para alcançar vantagens significativas e contínuas a partir da ferramenta

3.5.2 – Tipos de ferramentas de teste

No mercado existe um leque enorme de ferramentas de teste disponíveis, desde ferramentas gratuitas a pagas.

Podemos agrupar as ferramentas em categorias sendo que as principais indicadas na Tabela 1:

Nome	Função	Exemplo
Ferramentas de gestão de testes	Estas ferramentas oferecem interfaces para: <ul style="list-style-type: none">• Registo da execução de testes• Registo de <i>bugs</i>• Gestão de requisitos• Relatórios de execução de testes	HP Quality Center
Ferramentas de gestão de requisitos	Armazenam instruções de requisitos e ajudam a identificar requisitos inconsistentes ou em falta num sistema	Borland - Atlas
Ferramentas de gestão de incidentes	Armazenam e gerem os relatórios de incidentes e ajudam á gestão do ciclo de vida das anomalias	Jira
Ferramentas de execução de testes	Permitem que os testes sejam executados automaticamente, ou semiautomáticos	Borland - Silk Mobile

Tabela 1 - Tipos de ferramentas de teste

3.6 - Metodologias de desenvolvimento de *software*

Existem diversas metodologias de desenvolvimento de *software* na qual os projetos se podem basear. Estas metodologias visão aumentar a velocidade do desenvolvimento do sistema bem como a melhoria contínua do processo, melhorando a comunicação da equipa e a sua organização diária com o objetivo principal de alcançar as metas desejadas de uma forma rápida e eficiente.

Estas metodologias têm vindo a sofrer um grande desenvolvimento ao longo dos tempos, adaptando-se as necessidades tanto das equipas de desenvolvimento bem como aos seus clientes.

Entre estas metodologias as mais conhecidas são o modelo em V, cascata e Ágil modelo na qual passarei a aprofundar posteriormente.

3.6.1 – Metodologia Ágil

É uma metodologia de desenvolvimento de *software* que tem como principal foco o equilíbrio entre as variáveis preço, tempo e qualidade do sistema.

O principal benefício o facto de ser possível entregar pequenas verses do sistema ao cliente, permitindo que o cliente faça uma avaliação contínua do sistema e caso exista a necessidade de alterar algum requisito no sistema essa alteração não seja um fator de bloqueio para todo o projeto.

Uma das técnicas mais conhecidas desta metodologia é o *Scrum*.

No *Scrum* tipicamente os projetos são divididos em *Sprints*, usualmente com uma duração entre as 2 e as 4 semanas.

Cada *Sprint* representa um período de tempo na qual um conjunto de atividades vão ser efetuadas.

As atividades a serem desenvolvidas são mantidas numa lista conhecida como *Backlog*.

No início de cada *Sprint* é feita uma reunião envolvendo o *Product Owner* e a equipa de desenvolvimento e testes.

Nesta reunião é definido quais as funcionalidades que vão ser desenvolvidas durante o decorrer do *Sprint*, sendo estas priorizadas pelo *Product Owner*, e atribuído um grau de complexidade para cada uma das atividades pela equipa de desenvolvimento. Estas atividades vão ser colocadas na *Sprint Backlog*.

Durante a execução da *Sprint*, diariamente é efetuado um *Daily Scrum* com a duração de cerca de 15 minutos na qual a equipa de desenvolvimento e de testes se reúnem para dizer o ponto de situação, referindo o que fizeram no dia anterior, o que vão fazer no dia e quais os impedimentos relativos as suas tarefas.

No final da *Sprint* é feita uma reunião, a *Sprint Review* na qual são apresentadas as funcionalidades implementadas. Por fim é feito uma *Sprint Retrospective* na qual é debatido o que correu bem durante a execução da *Sprint* e que pode ser melhorado na *Sprint* seguinte.

Este ciclo é repetido até que todas as funcionalizações sejam entregues e o projeto seja dado como concluído.

A Figura 9 esquematiza a técnica *Scrum*:

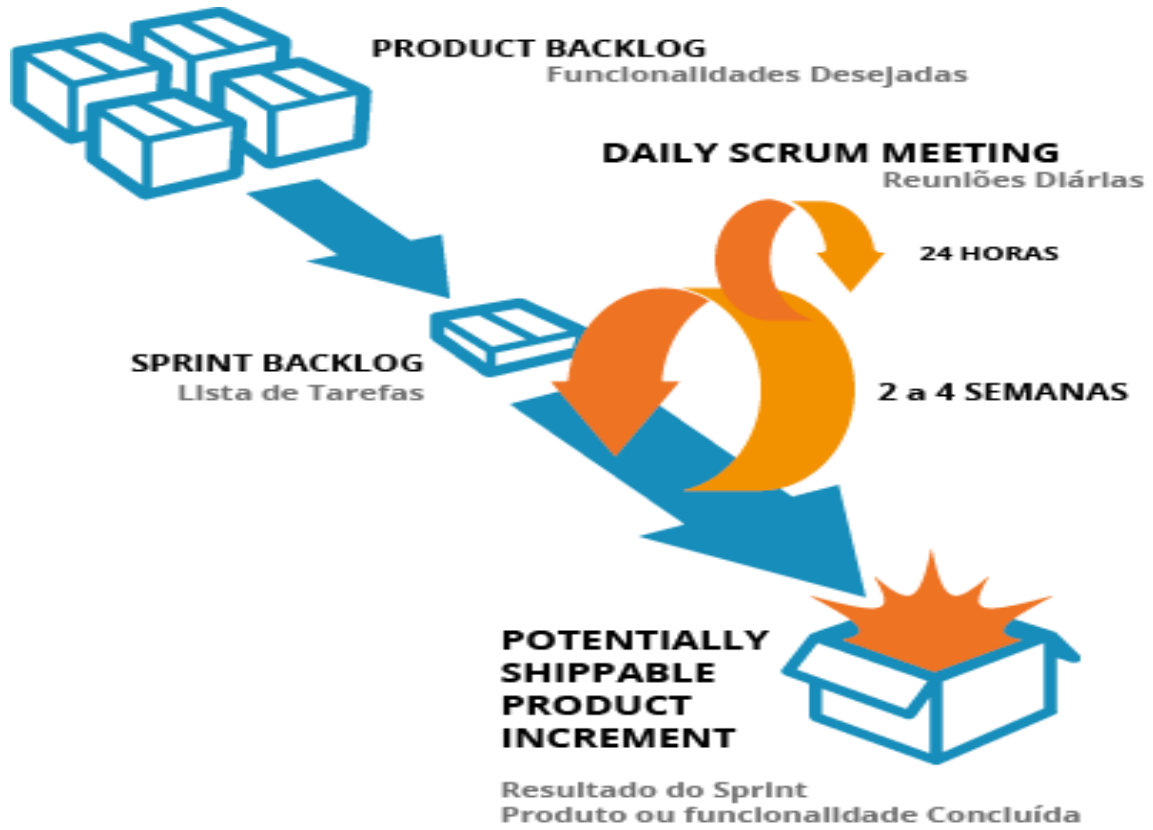


Figura 9 - Scrum

Capitulo IV – O projeto: Daikin iTC

4.1 – O cliente: Daikin

A Daikin é uma empresa multinacional japonesa, fabricante e fornecedora de equipamento AVACR (aquecimento, ventilação, ar condicionado e refrigeração).

A Daikin foi fundada em 1924 pelo senhor Akira Yamada tendo dado-lhe o nome de Osaka Kinzoku Kogyosho Limited, o que em 1923 virá ser chamado de Daikin nome pelo qual hoje é conhecida.

A Daikin está presente um pouco por todo o mundo como demonstra a Figura 10.

Em Portugal a Daikin está presente desde 2004, processo que ocorreu com a compra da Esteproar pela Daikin Europe N.V.



Figura 10 - Presença da Daikin no Mundo [5]

4.2 – Âmbito e descrição do projeto

O projeto Daikin iTC consiste na implementação de um sistema de um produto na *Cloud* para a monitorização e controlo de dispositivos de climatização da marca como demonstrado na Figura 11.

Com este sistema será possível realizar diversas operações que irão interagir com os equipamentos do cliente, como por exemplo:

- O utilizador poderá controlar e monitorizar os seus equipamentos de uma forma simples e estruturada;
- Será possível criar horários para executar diversas operações no tempo definido;
- Os técnicos responsáveis pela reparação dos equipamentos poderão verificar quais os erros que estão a ocorrer remotamente.

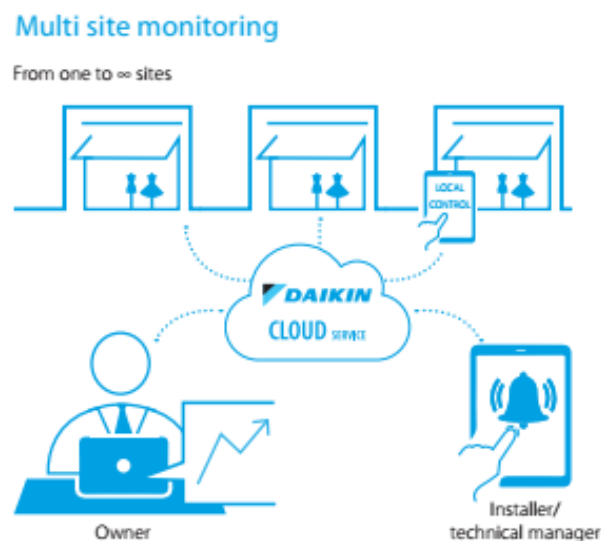


Figura 11 - Daikin Cloud Service [6]

4.3 – Funcionamento interno do projeto

4.3.1 – Metodologia de desenvolvimento

O projeto é executado segundo a metodologia Agile, utilizando a técnica *SCRUM* como demonstra a Figura 12.

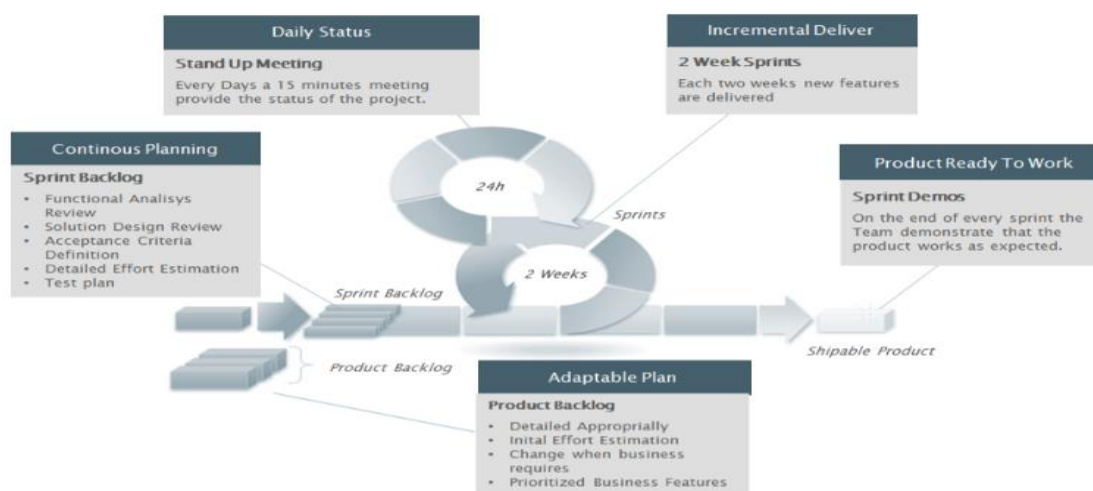


Figura 12 - Metodologia de desenvolvimento do projeto [7]

4.3.2 – Constituição atual da equipa

Função	Responsabilidades
Product Owner - (Daikin Europe)	Representa o cliente final. Responsável pelo <i>Project Backlog</i>
API Architect/Experts - (Daikin Europe)	Responsável pela arquitetura das API's Fornecer informação relativa a API's que vão ser consumidas pela aplicação.
Subject Matter Experts - (Daikin Europe)	Responsável por explicar dados técnicos do trabalho a ser desenvolvido.
Delivery Manager - (Altran BE)	Responsável por garantir o curso normal do projetos nas diferentes fases.
Proxy Product Owner - (Altran BE)	Responsável pela comunicação entre a <i>Scrum Team</i> e o Product Owner.
Scrum Master - (Altran PT)	Ajuda na compreensão da metodologia do projeto. Responsável por definir o <i>Sprint Backlog</i> de acordo com a capacidade da equipa
Web Developer - (Altran PT)	Responsável pelo FrontEnd do sistema. Desenvolve produtos web como: <ul style="list-style-type: none"> • HTML • CSS • JavaScript • AngularJS
.Net Developer - (Altran PT)	Responsável pelo backend da aplicação
Tester - (Altran PT)	Garante a qualidade do sistema. Analisa, desenvolve e executa testes para os produtos entregues.
Quality Manager	Responsável pela qualidade do sistema. Fornece suporte e analisa aspetos relativos ao risco e qualidade do projeto.

Tabela 2 - Equipa Daikin iTC

4.3.3 – Fluxo de trabalho

Cada *Sprint* tem uma duração de duas semanas.

Caso existam novas funcionalidades a serem adicionadas ao *Project Backlog* é feita uma reunião para definir o grau de complexidade de cada funcionalidade.

No início de cada *Sprint* é feito uma reunião de *Sprint* planing. Esta reunião envolve a equipa de desenvolvimento, equipa de testes, o *Proxy Product Owner* e o Gestor do projeto onde é definido o que vai ser implementado ou não na *Sprint*.

Seguidamente a equipa de desenvolvimento e de testes faz uma reunião de *grooming* em separado, mas com o mesmo objetivo: a divisão do trabalho a ser efetuado para a *Sprint* atual.

Diariamente é feito uma *Stand Up Meeting*, com uma duração relativa de 15 minutos, onde é dito o que foi feito no dia anterior e o que vai ser feito no dia atual.

No final da *Sprint* é feita uma reunião, a *Sprint Review* na qual são apresentadas as funcionalidades implementadas.

Por fim é feito uma *Sprint Retrospective* na qual é debatido o que correu bem durante a execução da *Sprint* e que pode ser melhorado na *Sprint* seguinte.

Em resumo podemos sintetizar o fluxo de trabalho na Tabela 3:

Evento	Duração	Quando
<i>Sprint Planning</i>	0,5 Dias	Início da <i>Sprint</i>
<i>Sprint execution</i>	8.5 Dias	Depois da <i>Sprint Planning</i>
<i>Stand-up meetings</i>	15 Minutos	Diariamente
<i>Sprint Review</i>	0.5 Dias	Depois da <i>Sprint execution</i>
<i>Sprint Retrospective</i>	0.5 Dias	Depois da <i>Sprint Review</i>

Tabela 3 - Tabela Resumo Fluxo de Trabalho

4.3.4 - Fluxo de reportagem de defeitos

Ao longo da execução dos testes na aplicação foram muitos os erros e pontos de melhoria encontrados. É importante que estes depois de serem encontrados seja convenientemente reportado para as pessoas responsáveis. A figura 13 demonstra o processo na qual os erros são reportados:

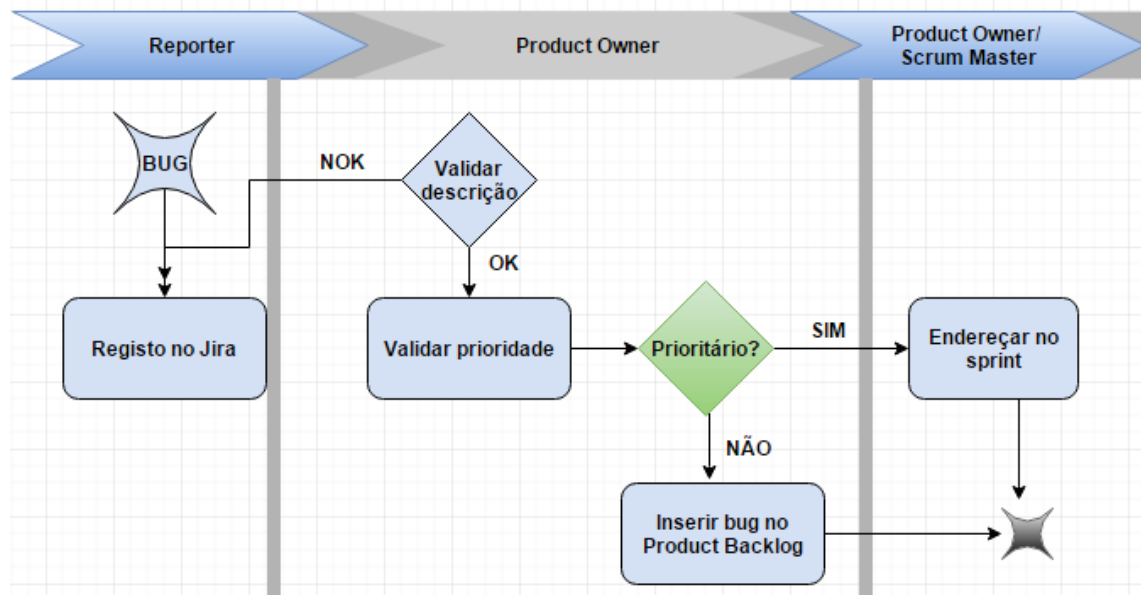


Figura 13 - Fluxo de reportagem de defeitos

No projeto Daikin iTC é utilizada a ferramenta Jira para a gestão de *bugs*. Sempre que é inserido uma nova falha é necessário colocar alguma informação para que esta fique estruturada e concisa:

1. Definição da falha;
2. Paços para obtenção da falha.
3. Nível de prioridade.

4.4 – Ferramentas utilizadas no âmbito do projeto

No início do projeto, na fase de planeamento foi definido pelo arquiteto do sistema um conjunto de ferramentas a serem usadas no âmbito do projeto.

Em alguns casos estas ferramentas podem ser alteradas caso existam imposições por parte do cliente, algo que não veio a suceder neste caso particular.

No projeto Daikin iTC são utilizadas as seguintes ferramentas:

4.4.1 – Visual Studio Pro 2015

O Visual Studio é um pacote de programas da Microsoft para desenvolvimento de software.



4.4.2 – NUnit

É uma *framework* de testes unitários para todas as linguagens .Net



4.4.3 – Jenkins

É uma ferramenta multiplataforma de integração contínua. Depois do Jenkins ser configurado, a cada alteração no repositório de código as tarefas são executadas e em caso de erros compilação ou falhas nos testes automatizados os elementos do projeto serão alertados.



4.4.4 – SpecFlow

E uma ferramenta .net que permite a escrita de especificações no formato Gherkin.

SpecFlow visa colmatar o défice de comunicação entre os especialistas da matéria e a equipas de desenvolvimento.



4.4.5 – Selenium

O Selenium é uma ferramenta de automação de testes de aplicações web.

Os testes são executados diretamente num *browser*, passo a passo como se estivessem a ser executados manualmente. No final da execução dos testes é gerado um ficheiro de *Logs* na qual esta descrito os passos que o robô seguiu bem como a descrição dos erros encontrados caso estes existam.



4.4.6 – Jira

É uma ferramenta desenvolvida pela Atlassian.

O Jira permite agrupar num único local os dados relativos a tarefas que estão ou não executadas, os bugs relacionados com cada tarefa e a quem estão associados, etc.

É uma ferramenta ideal para um gestor de projeto pois com esta ferramenta possui com um conjunto de ferramentas administrativas como criação/edição de fluxos de trabalho, configurações de permissões, geração de gráficos entre outras que facilitam o dia a dia de toda a equipa.



4.4.7 – Gherkin

Gherkin é uma linguagem corrente para descrição e automação de testes.

Com o Gherkin é possível associar código que é executado nos testes automáticos a frases correntes que descrevem a funcionalidade do sistema. Esta linguagem favorece tanto o cliente como toda a equipa do projeto pois todos falam a mesma linguagem.



4.4.8 – TICS

O TICS é uma framework que analisa um conjunto de dados relativos ao código do projeto e que através de uma aplicação web, o TICS Viewer mostra diversos gráficos relativos á qualidade do código.

O TICS atribui um nível á qualidade do código que vai de A a F sendo o A o nível mais alto e o F o nível mais baixo. Este nível é gerado perante a análise de fatores como a cobertura dos testes unitários, duplicação de funções, entre outros fatores.



Capitulo V – Trabalho desenvolvido

5.1 – Introdução

Ao longo dos quatro meses de estágio curricular na empresa Altran foi me proporcionado a aplicação dos conhecimentos adequados no âmbito do curso Testes de Software no seio de um dos seus projetos, a Daikin iTC.

Para além desta oportunidade ainda tive a possibilidade de aprimorar os meus conhecimentos linguísticos participando em formações da área, sendo no final emitido um certificado com o nível adquirido.

Foi ainda possível a participação em alguns eventos como foi o caso do Induction Day ou o Ignite.

Estas são algumas das atividades na qual passarei a aprofundar no presente capítulo.

5.2 – Trabalho desenvolvido

No âmbito do projeto Daikin tive a oportunidade de desenvolver diversas atividades na qual passarei a enumerar cronologicamente:

5.2.1 – Configuração do Ambiente de teste

Esta foi uma das primeiras atividades efetuadas na qual consistia na configuração do Visual Studio 2015 de modo a que suportasse o desenho e execução de testes automáticos.

Foi necessário adicionar diversos packages como o Nunit Test Adapter, Specflow entre outros.

5.2.2 – Desenho de plano de testes

Sempre que é iniciada uma nova *Sprint* é feito um documento com a mesma estrutura da figura 14.

Test Cases					
WITCA	#	Test case	Preconditions	Steps	Expected result
WITCA-11	1	As an user (any role) I want to associate Management Points to an Area so that they can be controlled/monitored through the area		Scenario: Add a management point to a zone Given I am an user (any role) And a site is assigned to me And I have a zone already created When I drag and drop a management point into the created zone Then the zone has this management point associated	
WITCA-11	2	As an user (any role) I want to associate Management Points to an Area so that they can be controlled/monitored through the area		Scenario: Remove a management point from a zone Given I am an user (any role) And a site is assigned to me And I have a zone already created And I drag and drop a management point into the created zone When I drag and drop a management point out of the zone Then the zone does not have this management point associated	
WITCA-11	3	As an user (any role) I want to associate Management Points to an Area so that they can be controlled/monitored through the area		Scenario: Remove a zone with associated management points Given I am an user (any role) And a site is assigned to me And I have a zone already created And I drag and drop a management point into the created zone When I delete the created zone Then the management point is assigned to the top level zone	
WITCA-11	4	When there is no zones the MP's are allocated to the top level	1- Logged in as an user (any role) and site associated with MP 2- The user is in Building Administration page 3- Delete all zones	- Go to Equipment List and count the MP's - Go to Zones page - Check that the Management Point's on the top level are the same on the equipment list	All the MP's are allocated to the top level
WITCA-11	5	Check that one MP is present only in one zone	1- Logged in as an user (any role) and site associated with MP 2- The user is in Building Administration page 3- Delete all zones	- Go to Building Administration - Go to Zones page - Open a created parent area and associated him one Management Point	The MP associated is only present in new parent area

Figura 14 - Test plan

Este documento contem todos manuais que serão posteriormente executados.

5.2.3 – Execução e reportação de falhas

Por fim, quando os testes estão desenhados e a equipa de desenvolvimento muda o estado das suas historias para Tests então são iniciadas as atividades de execução dos testes.

Ao longo da execução dos testes caso seja encontrada alguma falha é feito uma descrição do problema, com os paços para obter o mesmo erro, a hora em que ocorreu o erro, etc como demonstra a figura 15.

A	B	C	D	E	F	G	H	I	J	K
Assigned	ID_BUG	ID_USJ	bug id in jira	Platform	Version	State	Title	Description	step-by-step	Priority
JL		WITCA-82		Chrome, Firefox, IE11		CLOSED	Forced stop MSG don't have translation	When a controller is at force stop, the MSG displayed are only available in English		
TC		WITCA-83		Chrome, Firefox, IE11		CLOSED		The icon and text of MP in maintenance mode is not as described		
AE		WITCA-11		Chrome		CLOSED	Remove a child with MP	When delete one child with MP he dont appears anymore only after refresh. It first made a Get is done and then the delete		
AE		WITCA-11		Chrome		CLOSED	Remove a zone with associated management points	When delete one zone with MP he dont appears anymore only after refresh.		
JL		WITCA-147		Chrome, Firefox, IE11		CLOSED	The Remoon restrictions don't have translation	The options available in the Remoon restrictions don't have translations (only english)		

Figura 15 - Exemplo estrutura relatório de erros

5.3 – Ferramentas utilizadas

Durante o decorrer das atividades inerentes ao estágio foram utilizadas diversas ferramentas nas quais facilitam a execução de algumas das tarefas que me foram atribuídas.

De seguida descrevemos algumas das ferramentas utilizadas:

5.3.1 - Swagger

O swagger é uma *framework* utilizada na descrição, consumo e visualização de serviços *RESTful*. Tem como base ficheiros JSON mas suportando também XML.

Um das principais vantagens da utilização do Swagger é que este permite gerar documentação automática, tendo como base as anotações presentes no código.

A aplicação possui um módulo UI que permite a interação com as *API's* de forma intuitiva sem a necessidade de ter grandes conhecimentos técnicos.

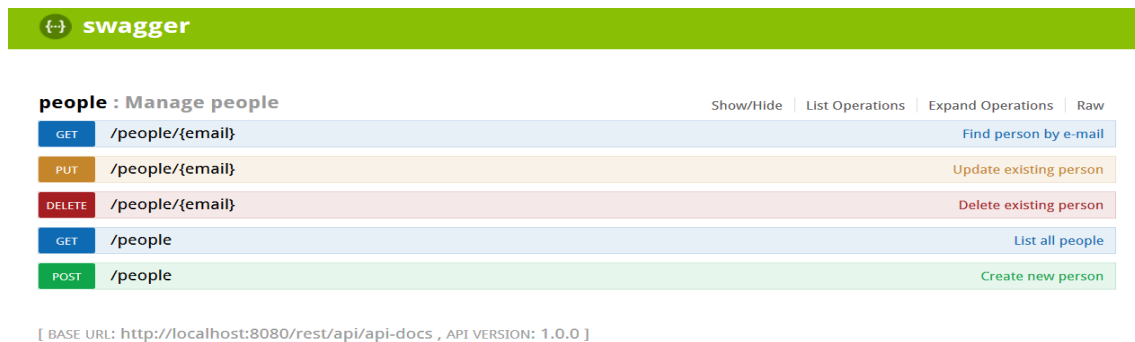


Figura 16 – Swagger UI

5.3.2 – Visual Studio e SpecFlow

O Visual Studio é programa da Microsoft que inclui um conjunto completo de ferramentas para construir aplicativos desktop e aplicativos Web ao nível empresarial e pessoal.

O Visual Studio permite a interação com outras ferramentas como o é o caso do SpecFlow.

O SpecFlow é uma ferramenta .NET de código aberto que permite escrever especificações usando a linguagem Gherkin, tendo como base a metodologia Behavior Driven Development (BDD). A Figura 17 demonstra a interação entre o Visual Studio e o SpecFlow.

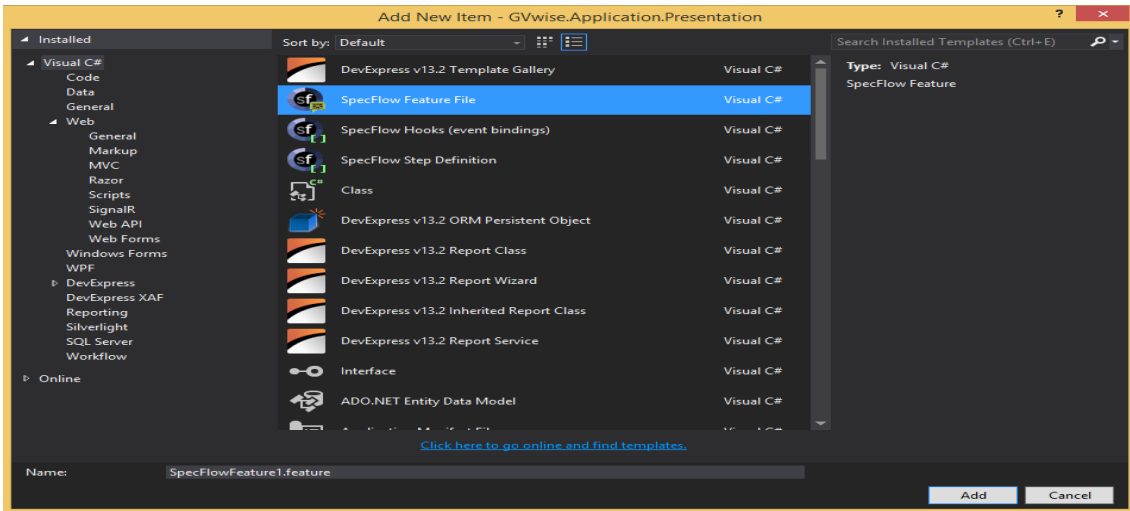


Figura 17 - Visual Studio: SpecFlow

5.3.3 – Browser developer tools

Atualmente todos os Browsers vêm equipados com esta funcionalidade.

Esta ferramenta permite uma análise de todos os elementos carregados pela página web desde Html, Css, JavaScript, permitindo a sua edição diretamente no browser.

Permite também fazer uma análise de performance da página. Analisando o tempo que cada elemento leva até ser carregado, o tempo que demora a página renderizar, entre outras funcionalidades como demonstra a figura 18.

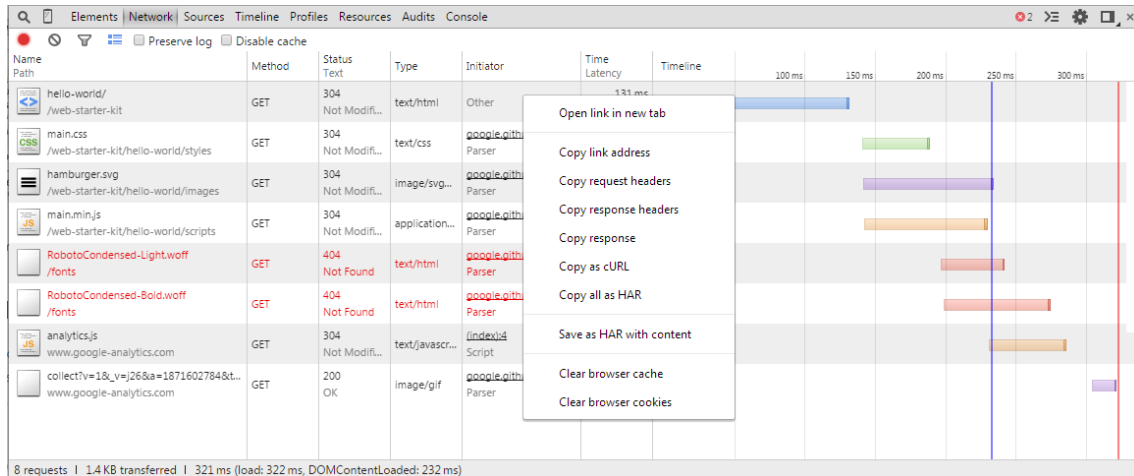


Figura 18 - Google Chrome DevTools Window [1]

5.4 – Formações e eventos

5.4.1 - Induction Day

Este é um evento na qual todos consultores recém-chegados a Altran efetuam.

Esta formação tem como principais objetivos o conhecimento do mundo Altran, a sua cultura, os seus valores, o que faz, como cresceu e progrediu na carreira, etc.

Esta sessão enquadrou-se no âmbito do programa Action4, um programa Altran que define os objetivos estratégicos a longo prazo baseados em 4 grandes eixos: *Employee Differentiation, Growth Engine, Profitability Engine e Customer Differentiation*.

A agenda do evento foi a seguinte:

Agenda:

- Abertura
- Grupo ALTRAN (história, cultura, valores, linhas de oferta)
- Estratégia Action4
- ALTRAN em Portugal
- Comunicação interna
- Ser Consultor (testemunhos)
- Almoço (a convite da Altran)
- Programas Altran
- Conduta Altran
- Quiz Altran
- Papel do consultor na vida da empresa

FAQ's

5.4.2 – Evento Ignite Your Future

O Ignite Your Future é um evento de cariz tecnológico promovido pela Universidade da Beira Interior, Município do Fundão, e pela Altran com a duração de cinco dias. A Figura 19 apresenta o cartaz do evento.

Este evento é especialmente dirigido a estudantes do ensino secundário com apetência e vontade de adquirir competências no domínio das novas tecnologias, nomeadamente nas áreas das TIC (Informática e Comunicações).

O programa do evento contou com leque diversificado de atividades desde: competições tecnológicas, momentos de lazer, ações de team building, conferências, visitas a empresas e universidade, desporto e aventura.

O meu enquadramento neste evento vem do facto de ser um dos monitores de um dos muitos exercícios propostos. O exercício tinha como âmbito a introdução aos testes automáticos, apresentação da ferramenta *Robot Framework*.

O exercício era dividido em três partes:

1. Pesquisar por Gmail no Google e chegar a página de Login do Gmail;
2. Efetuar Login;
3. Efetuar Log out;

Todos estes pontos teriam que ser efetuados utilizando a ferramenta Robot Framework, tendo como suporte alguma documentação relativa aos comandos que podiam utilizar na ferramenta.



Figura 19 - Cartaz Ignite Your Future

5.4.3 – Formação linguística em Francês

Durante o período de estágio frequentei uma formação linguística em francês, pós-laboral com uma duração de 50 horas. Esta formação era de nível básico (A1), realizada nas instalações da Altran sendo esta dada por um formador do IEFP.

Sendo a Altran uma multinacional Francesa e muitos dos seus projetos serem para clientes de origem francesa esta formação demonstrou ser muito positiva pois permitiu a aquisição de novas competências que certamente vai ser útil no futuro.

5.4.4 – Formação Java

Esta formação foi uma oportunidade que me foi dada pelo gestor de projeto, na qual tinha como principal objetivo a aquisição de novas competências ao nível de programação que poderão vir a ser úteis no âmbito deste projeto bem como outros.

Esta formação teve uma duração de 168 horas, ou seja, 21 dias.

Nesta foram abordados diversos temas tais como:

- 36 horas (4,5d) – Formação *Soft Skills*
- 4 horas (0,5d) – Formação de Processos
- 40 horas (5d) – Formação tecnológica Java SE 8
- 80 horas (10d) – Formação tecnológica Java Enterprise Edition
- 8 horas (1 d) – Avaliação final

Durante esta formação foram realizados diversos trabalhos, sendo que dou maior relevância dois de cariz técnico com o objetivo de aplicar os conceitos lecionados ao longo da formação:

1. Projeto Java SE

Conceitos aplicados:

- Herança;
- Leitura de dados da consola (stdin) para ler os valores a guardar nos objectos;
- Collections
- Leitura e escrita de todos os objetos definidos de/para ficheiro
- Ligação a BD (No Netbeans)
- Introdução, consulta e manipulação de dados na BD

Descrição do sistema desenvolvido:

Gestão de livros de uma Biblioteca na qual permite:

- Registo de conteúdo (Livro, Revista, DVD...);
- Registo de utilizadores;
- Empréstimo de livro a utilizador;
- Devolução de Livro;
- Consulta de informação.

2. Projeto Java EE

Conceitos aplicados:

- Arquitetura cliente/servidor.
- Gestão de dados pelo servidor
- Aplicativo standalone ou web
- Aplicativo (web) cliente.

Descrição do sistema desenvolvido:

Sistema de gestão de inquéritos na qual permite:

- Definir inquéritos;
- Efetivar o registo dos utilizadores que, via o aplicativo cliente, acedem ao sistema
- Calcular as estatísticas em tempo real das respostas recebidas para cada inquérito
- Criar e gerir a base de dados que arquivará toda a informação própria ao sistema.
- Criar e potenciar o perfil de administrador do sistema
- Prever, se oportuno, o perfil de gestor de inquéritos que terá a sua responsabilidade a
- criação e a gestão dos inquéritos que criar.
- Disponibilização de informação através dum Serviço de Web (SOA) simples ou Restful –
- 8 Mecanismo de agendamento de Backups da Base de Dados na componente da administração.

Esta formação foi importante pois pude fortalecer os meus conhecimentos relacionados com Soft Skills tais como a importância do trabalho em equipa e comunicação, bem como a aquisição de novas valências no que toca á programação.

Tecnicamente esta formação permitiu-me ter uma visão geral da linguagem Java aplicada ao nível pessoal e empresarial, algo que vai ser certamente positivo para o meu desenvolvimento profissional.

Conclusão

A realização do estágio curricular na Altran - Global Delivery Centre foi uma experiência muito positiva pois permitiu a aplicação dos conhecimentos adquiridos no âmbito do curso Tesp Teste de Software bem como a aquisição de nas novas competências profissionais e pessoais.

O facto da Altran possuir múltiplos projetos de grande dimensão ao nível nacional e internacional possibilitou-me ter uma maior precessão de como são aplicados os processos de trabalho a uma grande escala bem como é feita a gestão das diversas equipas de trabalho.

A formação linguística mostrou ser um fator decisivo para o sucesso dentro da empresa, pois em diversos projetos existem elementos de diferentes nacionalidades e é fundamental existir uma boa comunicação entre os diversos intervenientes, motivo este pelo qual a empresa aposta na formação dos seus consultores, algo que me veio a ser possibilitado durante o decorrer do estágio.

Em resumo a realização do estágio na Altran foi uma experiência muito enriquecedora na qual tentei tirar o melhor partido da mesma.

Bibliografia

- [1] Altran Portugal, “A Altran no Mundo e em Portugal,” Altran, 2016.
- [2] Altran, “História - Altran Portugal,” [Online]. Available: http://www.altran.pt/sobre-nos/altran-portugal/historia.html#.V21hQ_krLIU. [Acedido em 10 junho 2016].
- [3] [Online]. Available: <http://istqbexamcertification.com/wp-content/uploads/2011/12/cost-of-defects.jpg>. [Acedido em 10 junho 2016].
- [4] J. Lopes, *Esquema Processp de Teste*, 2016.
- [5] [Online]. Available: <http://polymer-additives.specialchem.com/storefronts/daikin>. [Acedido em 10 junho 2016].
- [6] Daikin, Daikin, 2016.
- [7] “Projeto Daikin iTC,” 2016.
- [8] “Google,” [Online]. [Acedido em 20 junho 2016].
- [9] Altran Portugal, “A Altran no Mundo e em Portugal,” Altran, 2016.
- [10] [Online]. Available: <https://www.dotnetcodegeeks.com/wp-content/uploads/2015/05/visual-studio-2013-logo.jpg>. [Acedido em 10 junho 2016].
- [11] [Online]. Available: https://upload.wikimedia.org/wikipedia/commons/b/b8/Nunit_logo_250.png. [Acedido em 11 junho 2016].
- [12] [Online]. Available: <https://wiki.jenkins-ci.org/download/attachments/2916393/logo-title.png?version=1&modificationDate=1302753947000>. [Acedido em 11 junho 2016].
- [13] [Online]. Available: <https://xebialabs.com/assets/files/plugins/specflow.jpg>. [Acedido em 11 junho 2016].
- [14] [Online]. Available: <https://upload.wikimedia.org/wikipedia/en/5/5c/Seleniumlogo.png>. [Acedido em 11 junho 2016].

[15] [Online]. Available:

https://upload.wikimedia.org/wikipedia/en/thumb/b/bf/JIRA_logo.svg/800px-JIRA_logo.svg.png. [Acedido em 11 junho 2016].

[16] [Online]. Available: <https://avatars0.githubusercontent.com/u/320565?v=3&s=400>.

[Acedido em 11 junho 2016].

[17] [Online]. Available: <http://tics.tiobe.com/viewerC/pub/cfg/logo.png>. [Acedido em 11 junho 2016].

[18] *Browser DevTools*, 2016.

Anexos


Anexo 1 – Instalações da Altran no Fundão





Anexo 2 – Página de login da aplicação



The Smart Control no matter where you are

Username 

Password 

English 

[Forgot password](#) [Forgot Password](#)

[Login](#)

Dashboard



Automatic Control



This website uses cookies to ensure you get the best experience on our website. [Learn more](#)

Accept