



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

RELATÓRIO DE ESTÁGIO

Curso Técnico Superior Profissional
em Testes de Software

Maria do carmo da Cruz Claudio

julho | 2018





IPG

Politécnico
da Guarda

Polytechnic
of Guarda

Escola Superior de Tecnologia e Gestão

Instituto Politécnico da Guarda

RELATÓRIO DE ESTÁGIO

MARIA DO CARMO DA CRUZ CLÁUDIO

RELATÓRIO PARA A OBTENÇÃO DO DIPLOMA DE TÉCNICO SUPERIOR PROFISSIONAL

EM TESTES DE SOFTWARE

JULHO DE 2018



Escola Superior de Tecnologia e Gestão
Instituto Politécnico da Guarda

RELATÓRIO DE ESTÁGIO

MARIA DO CARMO DA CRUZ CLÁUDIO

SUPERVISOR (ALTRAN): HUGO PRATA

ORIENTADORA (IPG): NATÁLIA GOMES

JULHO DE 2018

Identificação

Estudante:

Nome – Maria do Carmo da Cruz Cláudio

Morada – Urbanização da Quinta das Bertas Lº47 6300-814 - Guarda

N.º de aluno – 1012368

Curso Técnico Superior Profissional (TeSP) em Testes de Software

Instituto Politécnico da Guarda – Escola Superior de Tecnologia e Gestão

Correio eletrónico – 1012368@sal.ipg.pt / mariacaudio97@hotmail.com

Local de estágio:

Empresa – Altran Portugal SA – Global Delivery Centre, Fundão

Área de atividade – Consultoria de Inovação e Tecnológica

Localização – Centro de Negócios e Serviços, Praça Amália Rodrigues. 6230-350 Fundão

Telefone – +351 210 331 600

Correio eletrónico – info@altran.pt

Site – www.altran.pt

Coordenador na Altran – Eng.º Hugo Miguel Prata (Project Manager)

Correio eletrónico – hugomiguel.prata@altran.pt

Estágio curricular:

Início – 08/03/2018

Fim – 13/07/2018

Orientadora – Prof.ª Natália Fernandes Gomes

Correio eletrónico – ngomes@ipg.pt

Agradecimentos

O estágio curricular que realizei no 2º ano do Curso Testes de Software, na Empresa Altran Fundação foi-me bastante gratificante, pelo simples facto de as pessoas serem muito simpáticas e acolhedoras, o que levou a minha rápida integração no espaço e o que trouxe um ambiente muito agradável para quem trabalha aqui e para quem vem estagiar, uma das vantagens que tive, foi integrar uma equipa fantástica e com um projeto interessante a meu ver.

Considero esta oportunidade uma mais-valia na preparação tanto para o meu futuro profissional como para a minha formação escolar. Deste modo, gostaria de agradecer a várias pessoas que contribuíram em todo este processo.

Em primeiro lugar gostaria de agradecer à minha excelente Orientadora de Estágio – Natália Gomes por me ter facilitado o estágio, que se mostrou uma pessoa muito profissional e empenhada no seu trabalho, o que me motivou bastante para a realização do meu estágio curricular e dos trabalhos propostos durante esta. Além disso, mostrou-se sempre disponível para qualquer dúvida que surgisse, o que fez com que me sentisse mais integrada neste grupo de trabalho.

É também muito importante referir o meu coordenador Hugo Prata, que me acompanhou durante o período de estágio, que foi muito prestável na minha integração e se mostrou sempre disponível para esclarecer qualquer tipo de dúvida, não só a mim como ao meu colega de Estágio.

Também não é de esquecer a equipa no geral que se apresentou sempre disponível para tudo o que fosse necessário. É um privilégio poder fazer parte deste grupo de trabalho, é uma instituição com uma fantástica equipa de profissionais onde dou todo o mérito, principalmente à equipa na qual fui integrada.

Em segundo lugar, quero agradecer ao IPG – Instituto Politécnico da Guarda – por nos proporcionar estágios de forma a aplicarmos os conhecimentos que adquirimos na escola. É também uma mais-valia para os estagiários, pois dá uma ideia de como é o mundo do trabalho. Desse modo consegui compreender como deve funcionar uma empresa, para além disso apercebi-me como deve ser o ambiente.

Em terceiro lugar gostaria de agradecer aos meus pais, familiares e amigos por todo o apoio demonstrado durante o meu percurso profissional. Aos meus pais que sempre me mostraram que sem esforço não há resultados. E aos meus familiares e amigos pela companhia e pela força que me transmitiram.

Por fim, posso concluir que a integração no ambiente de trabalho também se deveu muito ao encorajamento e incentivo, ao dinamismo, criatividade e empenho por parte de todos os que me integraram na empresa, sobretudo a equipa do OSS e a todos os que por cá conheci.

Índice

Identificação.....	3
Agradecimentos.....	4
Índice de Figuras.....	6
Lista de Abreviaturas e Acrónimos.....	7
1. Introdução.....	8
2. Local de Estágio.....	9
2.1. Identificação da Empresa.....	9
2.2. Altran Fundão.....	10
3. Estágio.....	11
3.1. Atividades Desenvolvidas.....	11
4. Conclusão.....	40
5. Bibliografia.....	41

Índice de Figuras

Figure 1 – Documentação serviços reutilizáveis.....	11
Figure 2 - Esquema WSDL.....	15
Figure 3 - Iniciação de testes no SoapUI.....	17
Figure 4 - Testes no SoapUI - Assertions.....	18
Figure 5 - Testes em execução.....	19
Figure 6 - Teste terminado - pass.....	19
Figure 7 - Testes a falhar.....	20
Figure 8 - Testes concluídos e todos a passar.....	21
Figure 9 - Sumário de testes.....	22
Figure 10 - Início de testes no ReadyApi.....	23
Figure 11 - Detalhes do Teste Suite.....	24
Figure 12- Testes no ReadyApi.....	24
Figure 13 – Resultados do Load teste no ReadyApi.....	25
Figure 14 - XQuery no ReadyApi.....	26
Figure 15 - XPath no ReadyApi.....	27
Figure 16 - Histórico da execução de testes.....	27
Figure 17 - Resultados dos testes com 11 steps que passam.....	28
Figure 18 - LoadTest no SoapUI.....	29
Figure 19 - Dados estatísticos no gráfico - SoapUI.....	30
Figure 20 - Teste no ReadyApi com utilização do Loop.....	31
Figure 21 - Load Test no ReadyApi.....	32
Figure 22 - Resultados dos testes com 46 steps que passam.....	32
Figure 23 - Condição XPath - Goto Condicional.....	33
Figure 24 - Groovy Script.....	34
Figure 25 - Teste Completo.....	35
Figure 26 - Execução de testes terminada.....	35
Figure 27 - Novos testes SoapUI com novo Webservice.....	36
Figure 28 - Cadernos de casos de teste - Excel.....	37

Lista de Abreviaturas e Acrónimos

API - Interface de Programação de Aplicações

HTML - Linguagem de Marcação de Hipertexto.

HTTP - Protocolo de Transferência de Hipertexto

ISIS - Sincronizar Infraestruturas de Integração de Serviços

MTOM - Mecanismo de Otimização de Transmissão de Mensagens

OSS – Operações do Sistema de Suporte

RPC - Chamada de Procedimento Remoto

SOAP - Protocolo Simples de Acesso a Objetos

TCP - Protocolo de Controlo de Transmissão

UDDI - Descrição Universal, Descoberta e Integração

URI - Identificador Uniforme de Recurso

WSDL - Linguagem de Descrição do Serviço da Web

XML – Linguagem de Marcação Extensível

1. Introdução

A realização do presente relatório surge no âmbito do estágio curricular, referente ao 2º ano do Curso Técnico Superior Profissional – Testes de Software do Instituto Politécnico da Guarda. O estágio curricular foi realizado na Empresa Altran Portugal, designadamente na Altran do Fundão no período compreendido entre 8 de Março e 13 de Julho de 2018, tendo sido integrada num Projeto da 3ª maior Operadora de Telecomunicações da França, a Bouygues Telecom.

Durante este período, tive também a oportunidade de frequentar a formação de Língua Francesa – Iniciação de nível A1, promovida pela Altran Fundão em parceria com o IEFPP, tendo tido início no dia 21 de Março e concluída no dia 8 de Junho de 2018, tendo sido aprovada com nota final de 13 valores.

O estágio tem como objetivo a inserção do aluno no mercado de trabalho, proporcionando as condições adequadas à consolidação e aprofundamento dos conhecimentos teóricos adquiridos, conjugados com a aplicação prática da atividade da organização em que se realiza o estágio designadamente o grupo Altran. Deste modo, pretende-se proporcionar ao estudante um

contacto privilegiado com as realidades existentes, como complemento da formação académica, sendo indispensável o acompanhamento durante todo o período de estágio.

No estágio curricular foram desenvolvidas as tais como o desenvolvimento de competências linguísticas, no meu caso de Francês, o uso e a aprendizagem do programa SoapUI com um *Webservice* funcional para desenvolver casos de testes, criar um sistema de testes SoapUI e definir para cada caso de teste as assertions¹ necessárias.

A estrutura do presente relatório de estágio divide-se em 5 capítulos sendo o primeiro a presente introdução, na qual estão descritos os objetivos gerais e específicos do estágio curricular efetuado, o segundo capítulo uma breve apresentação do local de estágio, o terceiro capítulo num corpo central no qual se relatam sumariamente as atividades desenvolvidas no decurso do estágio e o quarto capítulo uma conclusão indicadora da pertinência dos conhecimentos adquiridos, o quinto capítulo é referentes às referências bibliográficas.

¹ As Assertions serão detalhadas no capítulo 3 – Assertions SoapUi e Assertions ReadyApi

2. Local de Estágio

O estágio curricular foi realizado na Empresa Altran Portugal, designadamente na Altran do Fundão no período compreendido entre 8 de Março e 13 de Julho de 2018.

2.1. Identificação da Empresa

O grupo Altran é uma multinacional francesa, líder global em consultoria de engenharia e R&D (pesquisa e desenvolvimento), a Altran posiciona-se como uma empresa inovadora no desenvolvimento de produtos e serviços à medida das necessidades dos clientes. Neste sentido, a Altran acompanha os clientes em cada etapa do projeto, desde o planeamento estratégico à fase de produção. Há mais de 30 anos no mercado global, a Altran capitaliza a experiência e *know-how* em setores chave como *Aerospace*, *Automotive*, *Defence*, *Energy*, *Finance*, *Life Sciences*, *Railway* e *Telecoms*. Com mais de 45 mil colaboradores e presente em mais de 30 países, o grupo Altran gerou, em 2017, receitas de 2,9 mil milhões de euros.[1]

Este grupo internacional opera em mais de vinte países da Europa, Ásia e Américas, oferecendo aos seus clientes um suporte global do projeto, e garantindo simultaneamente um nível consistente de serviço. Os principais domínios tecnológicos abrangidos pelo Grupo Altran consistem no desenvolvimento de produtos inovadores, sistemas inteligentes, experiência e ciclos de vida, sistemas de informação e engenharia mecânica.[1]

Com aproximadamente 1.200 colaboradores, a Altran Portugal é atualmente um dos principais intervenientes na área de Consultoria de Inovação e Tecnológica em Portugal. A Altran está presente no mercado nacional há mais de 15 anos, encontrando-se presente em diversos setores de atividade. A sua sede localiza-se na cidade de Lisboa, sendo que a empresa conta ainda com duas sucursais situadas nas cidades do Porto e do Fundão.[2]

2.2. Altran Fundão

A Altran Fundão, constitui um dos maiores centros de *Global Delivery Center* da Europa, visando a transferência de processos de negócio e de projetos para empresas de países próximos. Os princípios desta abordagem centram-se na existência de um local de entrega comum, na reutilização, em processos padronizados e certificados, e, por último, no desenvolvimento de um plano de melhoria contínua.[3]

A Altran Fundão, junto com a autarquia, disponibiliza um conjunto de condições competitivas que fazem desta localidade uma alternativa viável à alocação de determinados projetos, designadamente:

- Melhoria de tempo de resposta;
- Redução de custos;
- Maior flexibilidade na gestão de capacidade;
- Serviços uniformizados;
- Capacidades linguísticas;
- Proximidade Europeia;

3. Estágio

Neste capítulo irei apresentar as atividades desenvolvidas ao longo do período de estágio, que como já foi referido decorreu entre o dia 8 de Março e 13 de Julho, tendo sido integrada num Projeto da terceira maior Operadora de Telecomunicações de França, a Bouygues Telecom.

Neste projeto tive a oportunidade de lidar com duas equipas diferentes a primeira foi a ISIS que significa *Infrastructures Synchronize d'intégration de Service* e a segunda a OSS que é *Operation Support System*, ambas equipas integram o projeto da Bouygues.

3.1. Atividades Desenvolvidas

Ao longo deste percurso, tive também a oportunidade de desenvolver alguns casos de teste, na área de Testes, que envolvem a instituição e o grupo de trabalho na qual fui integrada, a Bouygues Telecom.

De forma breve podemos definir um Caso de Teste como sendo um conjunto de condições usadas para teste de *software*. Ele pode ser elaborado para identificar defeitos na estrutura interna do *software* por meio de situações que exercitem adequadamente todas as estruturas utilizadas na codificação, ou ainda, garantir que os requisitos do sistema que foi construído sejam plenamente atendidos.

Não obstante, antes da integração na Bouygues Telecom foi-me proposto analisar e apreender informação sobre diferentes aplicações, sistemas de testes e tecnologias que passo a descrever.

Parte ISIS

A primeira semana fui integrada na equipa do ISIS a qual está mais direcionada para a componente de testes mobile que significa *Infrastructures Synchronize d'intégration de Service*, a mesma pertence ao projeto da Bouygues. Nesta primeira semana acompanhei a equipa deste projeto na realização de testes, e como forma de integração também estive a ler documentação sobre o projeto de forma a adquirir um maior número de conhecimentos. A documentação foi-me entregue em francês, o que inicialmente me originou alguma dificuldade na compreensão. Esta dificuldade foi posteriormente colmatada.

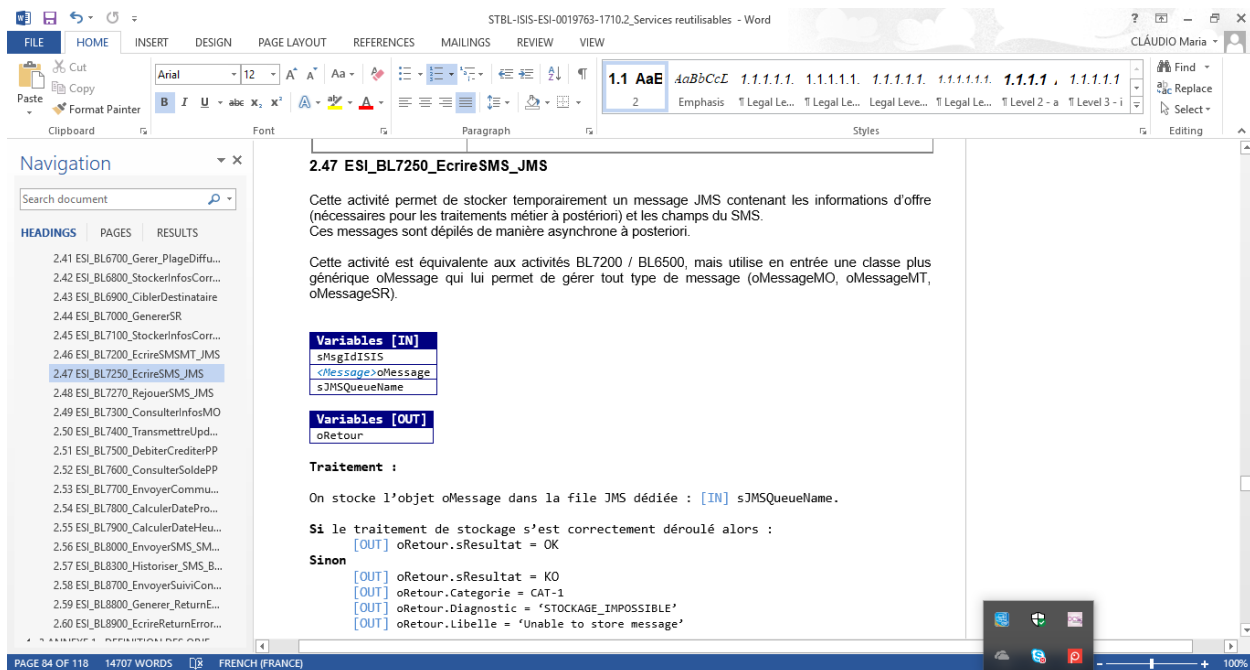


Figure 1 – Documentação serviços reutilizáveis

A figura 1, representa um exemplo da documentação do projeto, isto é informação sobre o projeto a nível mobile. A imagem representa a receção de mensagens no telemóvel sobre ofertas disponíveis para o cliente.

Parte OSS

Após uma semana após o início do estágio mudei de equipa e passei para a parte do OSS (*Operation Support System*). Esta está mais direcionado para as funcionalidades *Fixe*, quer isto dizer que se relaciona com a parte de Internet, Telefone e Televisão, ou seja, ofertas para casa do cliente.

Após a realização de uma reunião com o meu coordenador, ficou definido que ali ficaria a ler tutoriais e ver vídeos sobre o SoapUI, ferramenta *open source*. Após analisar a documentação sobre esta ferramenta procedi à instalação do programa em si, mas na versão free uma vez que a versão pro necessita de licença, a qual a Altran não dispunha.

O objetivo pretendido com esta tarefa foi a de conhecer e explorar uma nova ferramenta muito utilizada no projeto. Após a análise e compreensão desta ferramenta desenvolveu-se, então, o caderno de testes e criou-se o sistema de testes SoapUI para assim definir as suas *assertions*² necessárias.

Esta tarefa permitiu-me conhecer a ferramenta de um modo geral, mas de modo a melhorar os meus conhecimentos foi também necessário saber um pouco mais também sobre o Soap³. De modo a esclarecer o leitor apresento, de forma breve, em que consiste o protocolo Soap.

² Funções que verificam se o estado da aplicação está de acordo com o esperado/definido.

³ Protocolo para troca de informações estruturadas.

SOAP

O SOAP [4], é um protocolo para troca de informação num sistema distribuído e descentralizado. Este é baseado em XML e consiste em 3 partes: um envelope onde se descreve o que esta na mensagem e como processá-la; as regras para representar chamadas e respostas RPC (*Remote procedure call*); um cabeçalho onde são colocadas informações adicionais (por exemplo informação sobre autenticação) e um corpo que contém a informação a ser transportada. A vantagem do SOAP é a de poder ser usado numa grande variedade de protocolos de transporte de dados como por exemplo HTTP, TCP.

Devido às suas características o SOAP é bastante utilizado para a criação de serviços Web permitindo controlar, administrar ou simplesmente obter informações a partir de máquinas remotas ligadas via rede de computadores. Assinala-se como grande desvantagem o facto do SOAP usar XML uma vez que a sua utilização faz com que se perca um pouco de velocidade de processamento face a outras tecnologias.

SoapUI

O SoapUI [5], é uma ferramenta *open source* escrita em Java cuja principal função é consumir e testar *Web Services*. *Web Services*. É uma tecnologia baseada em XML e HTTP e cuja principal função é disponibilizar serviços interativos na Web que podem ser acedidos por qualquer outra aplicação independente da linguagem ou plataforma em que a aplicação foi construída.

O *Web Service* expõe as suas operações por meio de um tipo de esquema XML chamado WSDL (*Web Service Description Language*). [6] Neste contexto o SoapUI facilita todo o processo de criação e depuração dos testes por meio de uma interface gráfica visual e intuitiva.

De entre as principais características do SoapUI podemos destacar as seguintes:

- ❖ Importação e geração automática das requisições descritas no WSDL;
- ❖ Capacidade de gerenciar um número ilimitado de requisições para cada operação;
- ❖ Gerenciamento de múltiplo *endpoint* para cada *Web Service*;
- ❖ Validação das requisições e respostas contra as suas definições no WSDL;
- ❖ Testes funcionais, de carga e de stress;
- ❖ Execução de diversos testes em paralelo;
- ❖ Suporta expressões XPath;
- ❖ Suporta criação de testes complexos utilizando scripts Groovy;

Serviços Web (Web Services)

Os serviços Web [6], fornecem os meios padrão para interoperabilidade entre diferentes aplicações de *software* a correr em diferentes tipos de plataformas e sistemas operativos.

Os serviços Web são muitas vezes considerados como simplesmente uma API exposta via Internet. A sua arquitetura básica define-se como um conjunto de serviços expostos por um servidor que por sua vez podem ser consumidos por uma aplicação cliente remota.

A comunicação dos serviços Web é feita essencialmente usando Soap como protocolo de comunicação sendo as mensagens codificadas em XML. Embora não seja obrigatório a sua inclusão, WSDL este costuma ser um requisito pelo menos para quem pretenda facilitar a criação automática de aplicações clientes que consumam os serviços descritos pelo WSDL. [7]

Os serviços Web facilitam em muito a gestão de sistemas distribuídos, podendo até ser responsáveis por tornarem alguns sistemas que antes se caracterizavam por serem de difícil distribuição. Neste sentido, podemos afirmar que s serviços Web promovem a utilização de sistemas distribuídos.

Uma outra especificação dos serviços Web é o UDDI (*Universal Description, Discovery and Integration*), protocolo para expor e descobrir meta data de serviços Web com a finalidade de possibilitar e facilitar a sua descoberta por outras aplicações.

WSDL

O WSDL (*Web Services Description Language*) [7] consiste numa linguagem baseada em XML, definindo um modelo para os serviços Web. O WSDL define a gramática XML para descrever um serviço Web como uma coleção de pontos (portos) capazes de trocar mensagens entre si (Figura 2).

O WSDL permite esquecer os problemas associados à comunicação entre serviços Web que utilizem diferentes protocolos. Para um cliente comunicar com um serviço Web primeiro tem de saber todos os detalhes desse mesmo serviço (protocolo, comandos permitidos, cabeçalhos e códigos de erro) e o WSDL fornece então a gramática XML necessária para descrever estes detalhes. O WSDL define:

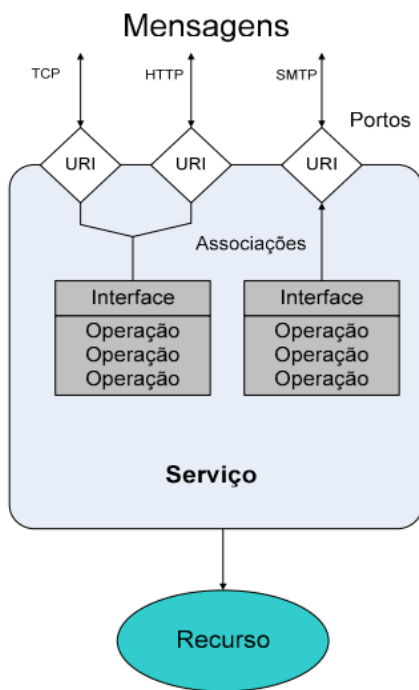


Figure 2 – Esquema troca de mensagens

Portos (*endpoints*) – Especifica um endereço para uma associação.

Associações (*bindings*) - Define um protocolo concreto e o formato dos dados para um *Port Type* específico.

Port Type – Um conjunto de operações abstratas suportadas por um ou mais portos.

Tipos (*Types*) – Um invólucro de definições de tipos de dados que são relevantes para as mensagens trocadas.

Mensagens (*Messages*) – Definição abstrata dos dados a serem comunicados.

Serviço (*Service*) – Um serviço agrupa um dado grupo de portos relacionados e dá-lhes uma identificação para se poder aceder ao serviço (URI).

De forma sucinta o WSDL define todas as características de um serviço Web e facilita a comunicação e simplifica o uso de serviços Web. Este descreve as mensagens a serem trocadas na comunicação, o tipo de informação trocada, define que protocolo está a ser usado pelo serviço e dá um identificador ao serviço Web (URI) para que este seja encontrado na rede por último o WSDL baseia-se em domínios (*namespaces*) que permitem identificar o serviço Web na rede.

Para além dos tutoriais explorados e da utilização de estas ferramentas foi também do meu interesse saber um pouco mais sobre a linguagem XML, uma vez que o SoapUI funciona com recurso a essa linguagem. Assim, descrevo de forma breve o XML.

XML

O XML, significa *Extensible Markup Language*, [9] é uma linguagem projetada para transmitir e armazenar dados em vez de exibí-los. É projetado para ser auto descritivo.

O XML [8] é apenas um complemento para o HTML. Na maioria dos sites da Web, o XML é usado para transportar a informação enquanto o HTML mostra a informação.

O XML é uma linguagem de marcação como HTML, com *tags* não definidas.

O XML e HTML são projetados para diferentes fins, que passamos a descrever:

- ❖ O XML foi concebido para o transporte e armazenamento de dados e o seu foco está no conteúdo dos dados.
- ❖ O HTML foi concebido para mostrar dados e concentrar-se na aparência dos mesmos.

Exemplo de XML [8]

```
<nota>
<para>Maria</para>
<de>Toze</de>
<cabeçalho>Lembre-se</cabeçalho>
<corpo>Não se esqueça de mim nesse fim de semana!</corpo>
</nota>
```

Testes SoapUi

Depois da leitura extensa das ferramentas Soap e SoapUi procedi à instalação do SoapUI para saber como este funcionava e as ferramentas que continha. Inicialmente deram-me um *Webservice* não funcional que dele apenas resultava o valor -1, este tipo de dados não se enquadrava com o objetivo. Assim e posteriormente, realizou-se uma reunião de dúvidas com o coordenador e Chefe de equipa onde me foi proposto um novo Webservice.

O Webservice em questão relaciona-se com futebol, com o item do “*GetPlayerByTeam*”, fig.3. Após colocar o item é solicitado o *TeamId*, em que neste caso dei o valor de Id 45. Após a colocação destes dados a aplicação devolve a informação relativa aos nomes dos jogadores, id's, nacionalidades entres outras informações dos jogadores, que pertencem aquele *TeamId=45*.

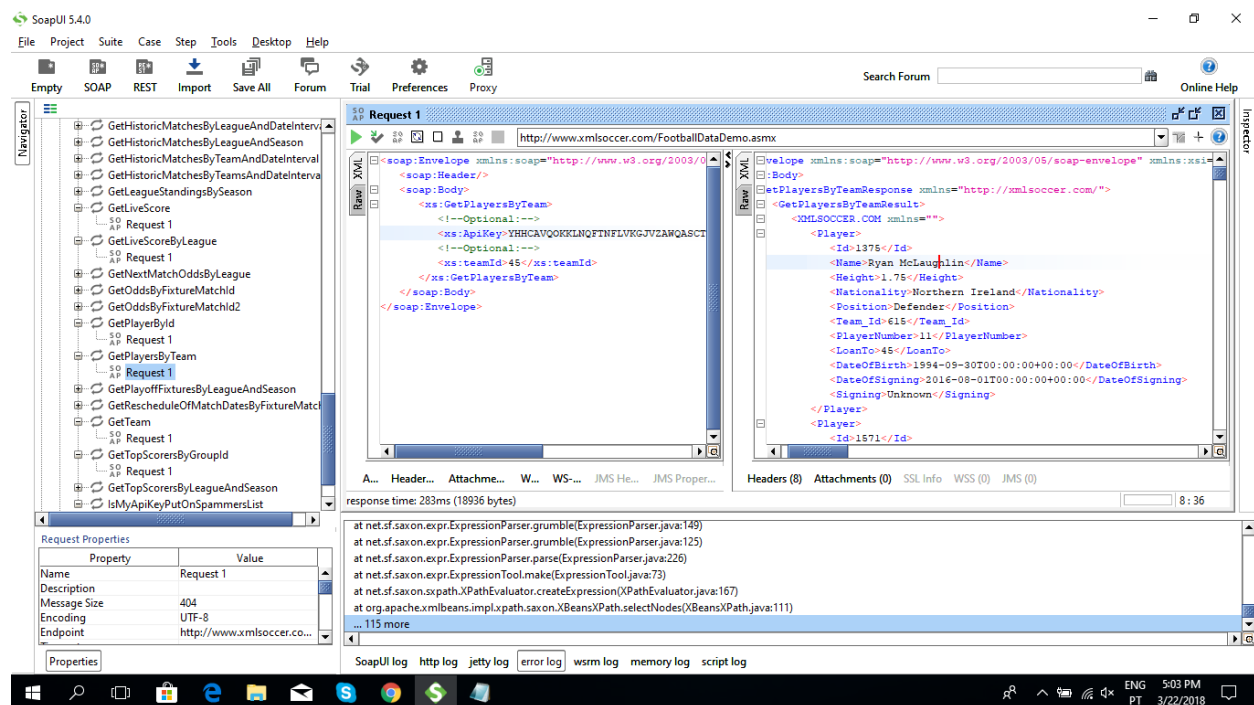


Figure 3 - Iniciação de testes no SoapUI

Assertions SoapUI

Após uma breve aprendizagem do modo de funcionamento do SoapUI e dos *WebService* comecei por tentar fazer uns testes simples da sua utilização. A ideia era a de entender e ser capaz de usar as ferramentas que o SoapUI oferece. Deste modo, comecei por criar alguns testes cases e testes steps e fui fazendo algumas *assertions* para ver também como estes funcionavam.

Assim, criei *assertions* do tipo *Contain* e *Not Contain* para verificar se existia a variável que eu introduzia, como é exemplo na figura 4, com a variável Scotland. Foram também geradas outras *assertions* com o objetivo de treinar e conhecer melhor o programa, como são exemplo a XPath, a XQuery, Scripts, códigos de estado HTTP. Para além do reconhecimento de uma variável também é importante analisar o tempo de resposta que define se a pesquisa passa do tempo estipulado. Caso este tempo seja ultrapassado o teste é considerado falha. Esta ação permite criar várias *assertions* para cada teste.

As próximas figuras mostram de forma breve a descrição de alguns testes, no caso da figura 4 gerou-se uma *assertions* do tipo *Contain* como acima referido, na figura 5 o teste encontra-se em execução e a figura 6 ilustra o momento em que o teste é validado.

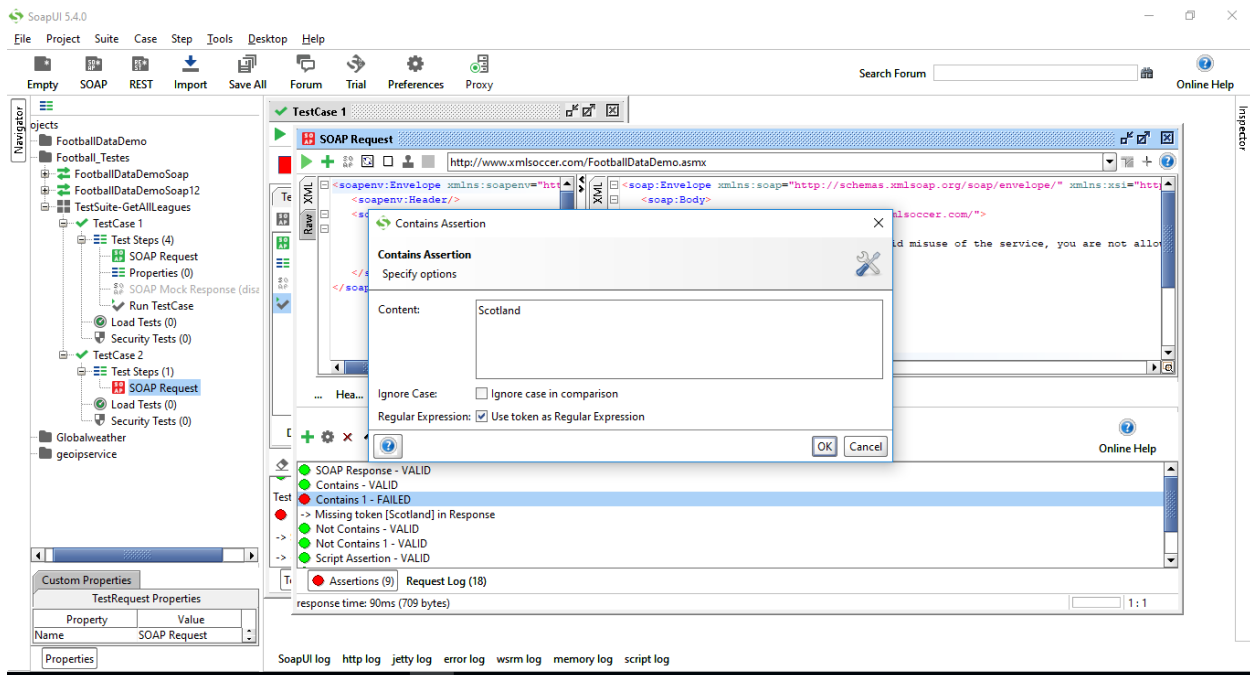


Figure 4 - Testes no SoapUI - Assertions

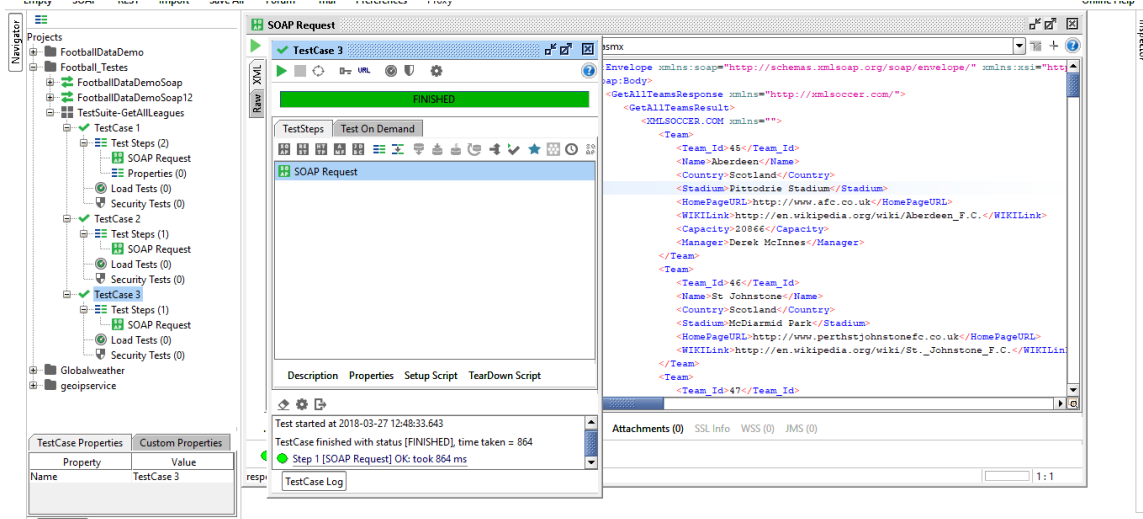


Figure 5 - Testes em execução

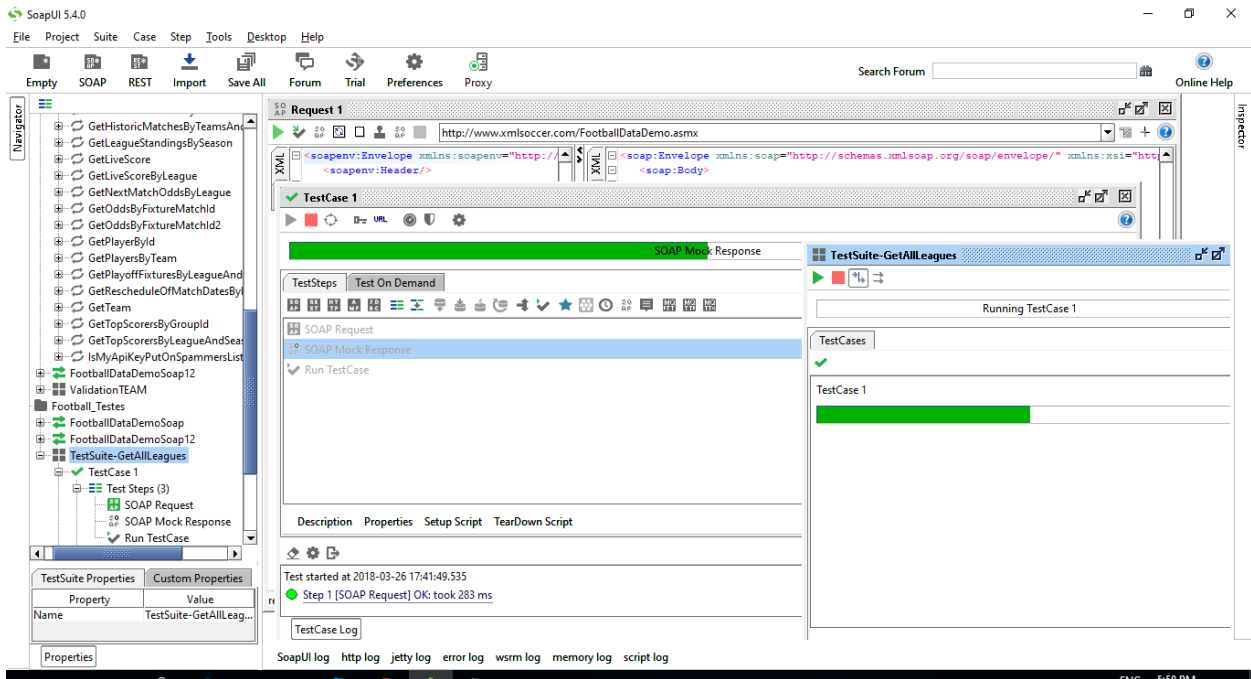


Figure 6 - Teste terminado - pass

Falhas dos testes

Após a execução de alguns testes surgiram falhas, figura 7, basicamente por que as *assertions* começaram todas a dar erro e o *webservice* deixou de responder. Mais tarde apercebi que na verdade a *API* tinha sido bloqueada e o *webservice* tinha deixado de responder

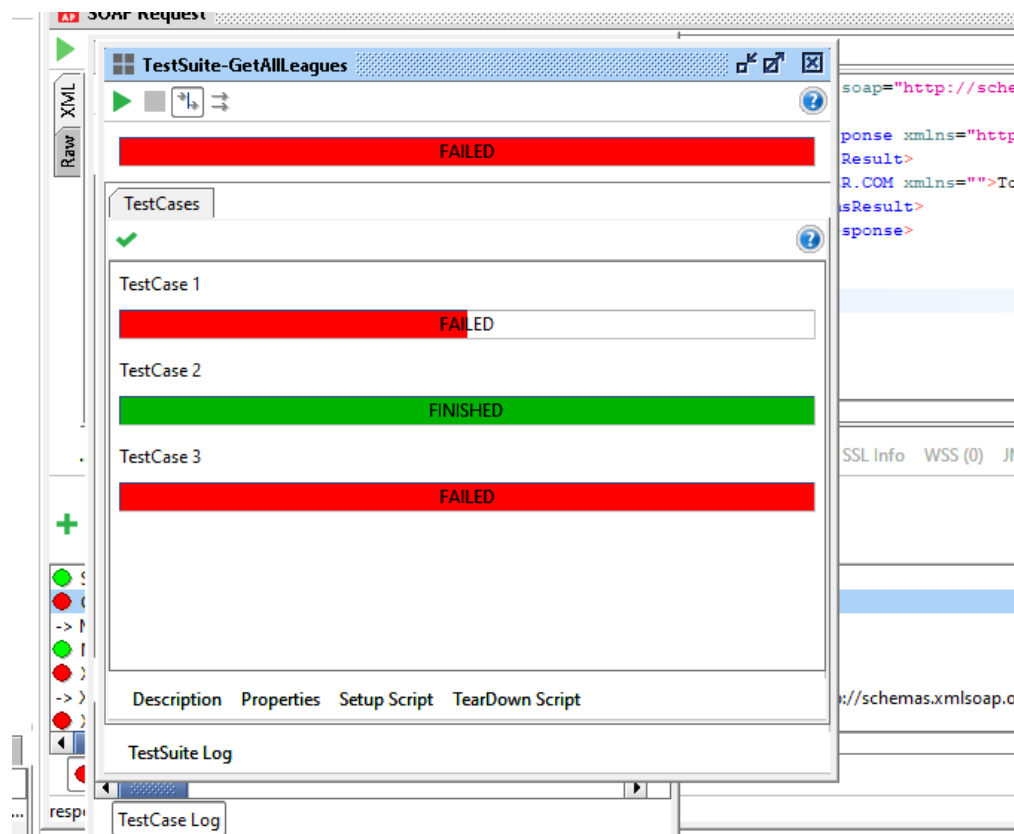


Figure 7 - Testes a falhar

A próxima figura ilustra o sucesso na realização dos testes, com o o *Test Suite*, tendo permitido o resultado de 4 *Test Cases* a passar. De modo a assegurar estabilidade dos testes decidi instalar o ReadyApi que é basicamente igual ao SoapUi, mas fornece mais ferramentas.

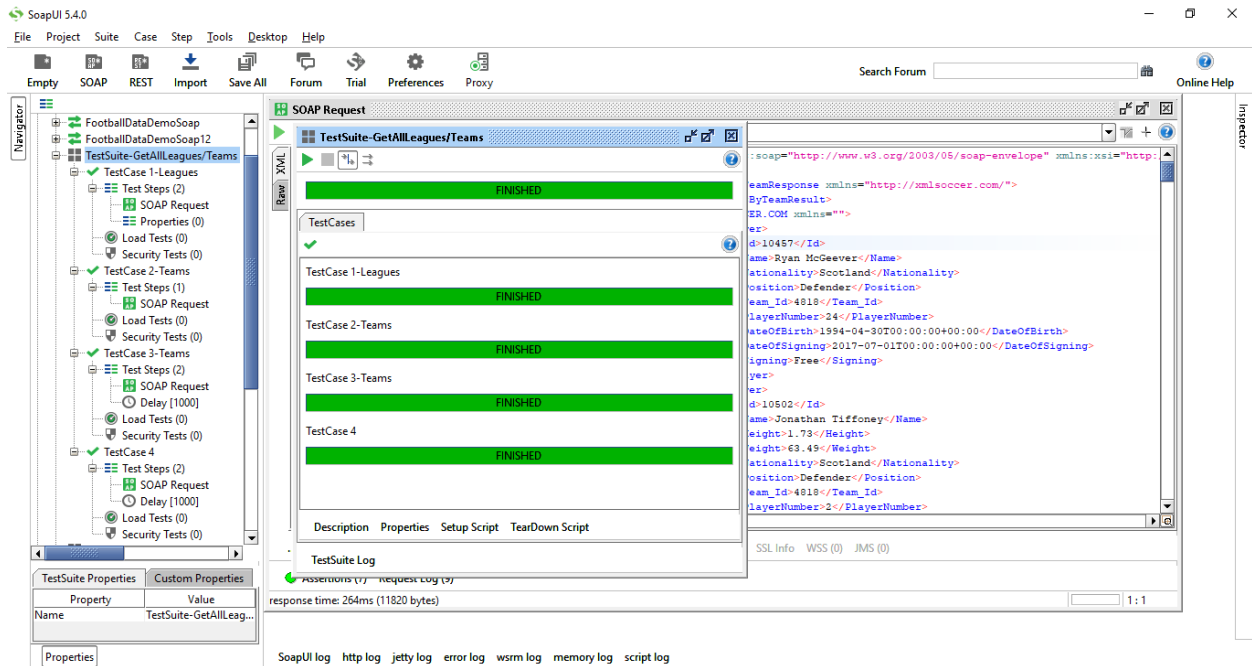


Figure 8 - Testes concluídos e todos a passar

De modo a otimizar os testes optou-se também por encontrar um novo *Webservice*, figura 9, o que permitiu terminar este conjunto de testes, com 2 *Test Suite* criados, 5 *Test Cases*, 9 *Test Steps* e com um total de 36 *assertions* divididas por cada *Test Step*.

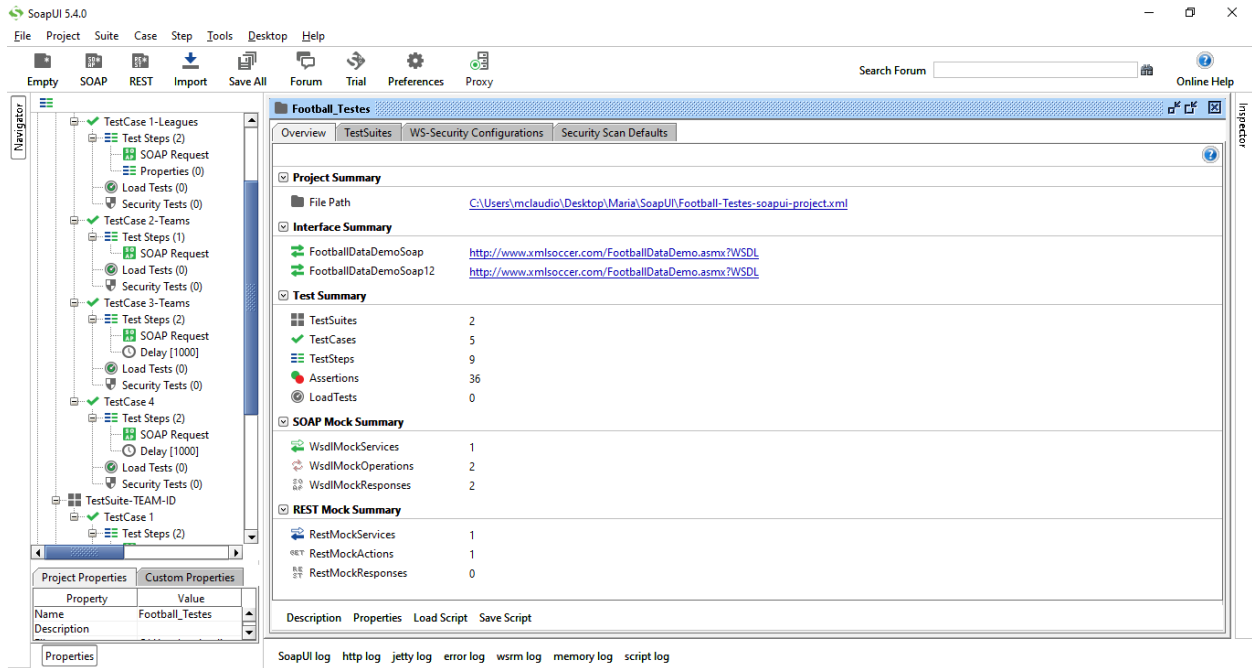


Figure 9 - Sumário de testes

Testes ReadyApi

A instalação do ReadyApi teve como objetivo o de permitir realizar testes estáveis uma vez que isso não estava a acontecer como referido anteriormente. O ReadyApi equivale ao SoapUI Pro, mas uma vez que não tinha licença só conseguiria trabalhar nele durante o prazo de 14 dias grátis. Nestes 14 dias deu para perceber as ferramentas que este tinha e que o SoapUI Free não tinha possibilitando entender as diferenças entres estas aplicações.

O ReadyApi permite ir buscar testes já realizados e criar novos testes, mais complexos, uma vez que existem mais funcionalidades.

As seguintes figuras mostram um extrato do programa, bem como as operações realizadas. A figura 10 e 12 ilustram a execução de um teste normal onde é recebida uma resposta do *Webservice* dependendo da variável que lhe é atribuída. A figura 11 ilustra os detalhes dos *Test Suite* de modo a especificar o que foi feito.

A figura 13 apresenta os resultados do *Load* teste graficamente, isto é, ilustra o desempenho do teste apresentando o tempo de resposta, as falhas e a duração do teste em si.

The screenshot displays the ReadyAPI 2.3.0 - SoapUI interface. The top menu bar includes File, Edit, View, Project, Suite, Case, Step, Request, Test Runs, and Help. The toolbar contains icons for Run, New, Save, and other actions. The Navigator pane on the left shows a project structure with folders like 'Football_Tests' and 'TestSuite-GetAllLeagues/Teams'. The main workspace is divided into 'Request' and 'Response' sections. The 'Request' section shows a SOAP request for 'GetAllLeagues' with an API key. The 'Response' section shows the resulting XML response for the English Premier League.

XML Node	Value
soap:Envelope	(Envelope)
soap:Body	(Body)
GetAllLeaguesResponse	
GetAllLeaguesResult	
XMLSOCCER.COM	
League	
Id	1
Name	English Premier League
Country	England
Historical_Data	Yes
Fixtures	Yes
Livescore	Yes
NumberOfMatches	6774
LatestMatch	2018-04-01T17:00:00+00:00
IsCup	false

Figure 10 - Inicio de testes no ReadyApi

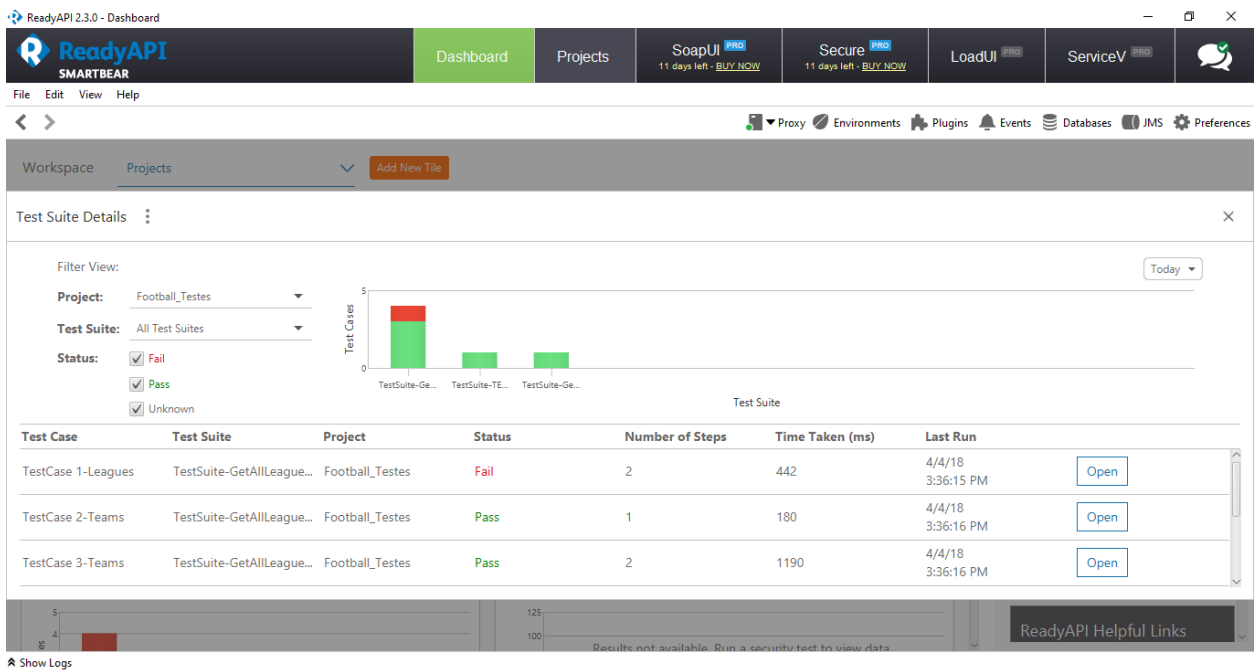


Figure 11 - Detalhes do Teste Suite

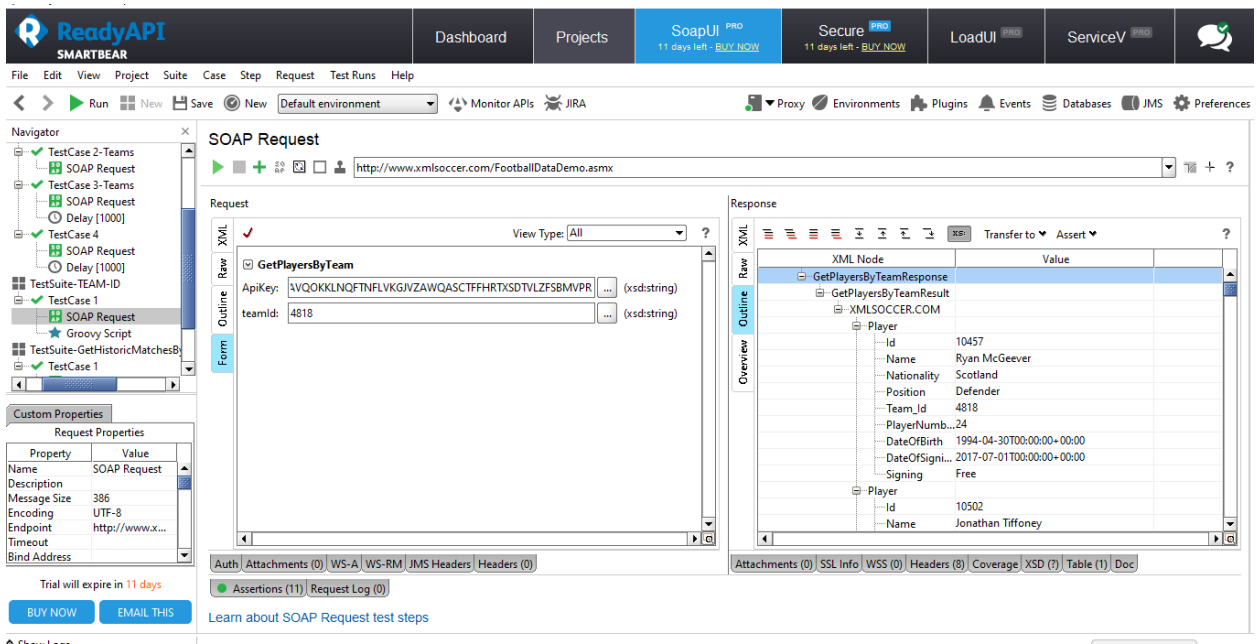


Figure 12 - Testes no ReadyApi

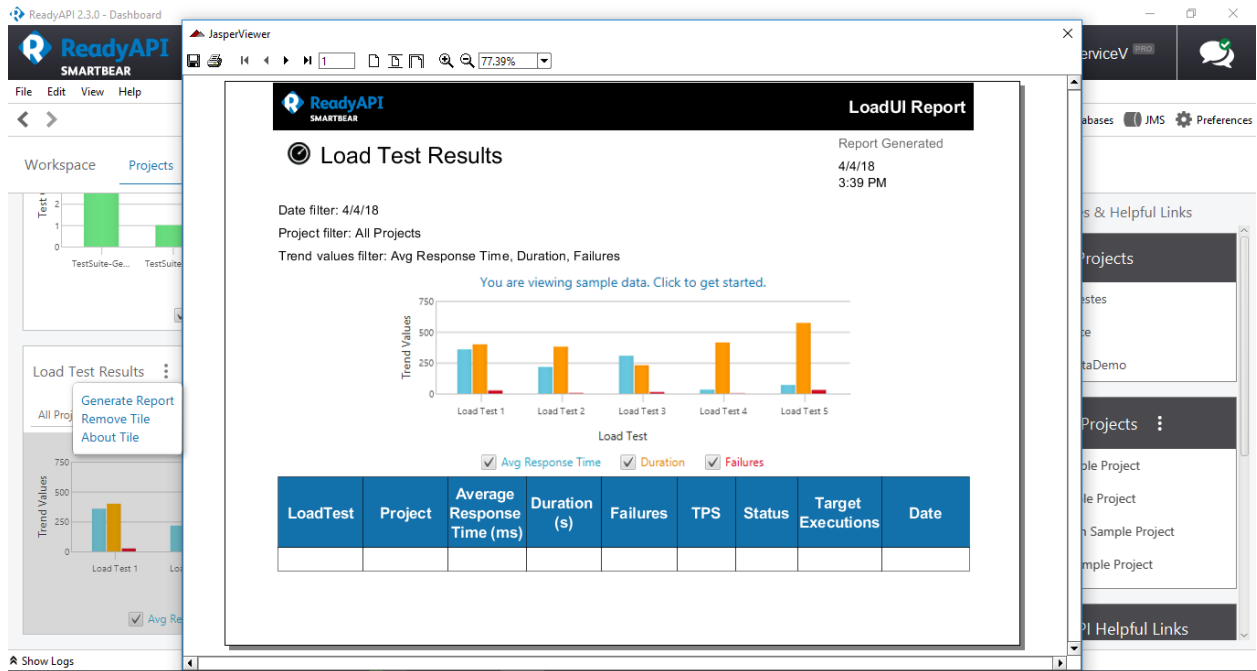


Figure 13 – Resultados do Load teste no ReadyApi

Assertions no ReadyApi

Uma vez que o RedayApi permite criar *assertions* foram também geradas algumas para melhorar a aprendizagem sobre as XQuery e sobre as XPath. A figura 14 representa a XQuery pedida onde foi dada o nome da Liga e onde o resultado esperado foi o mesmo do resultado obtido. A figura 15, retrata a XPath que é usado para calcular valores.

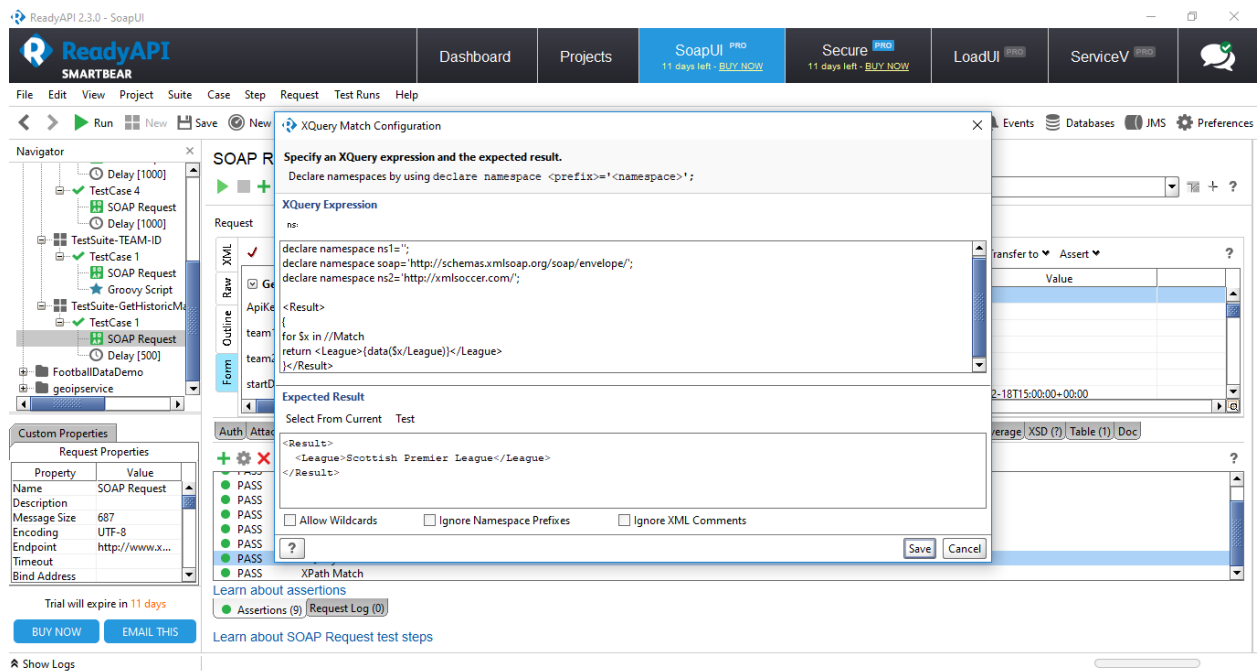


Figure 14 - XQuery no ReadyApi

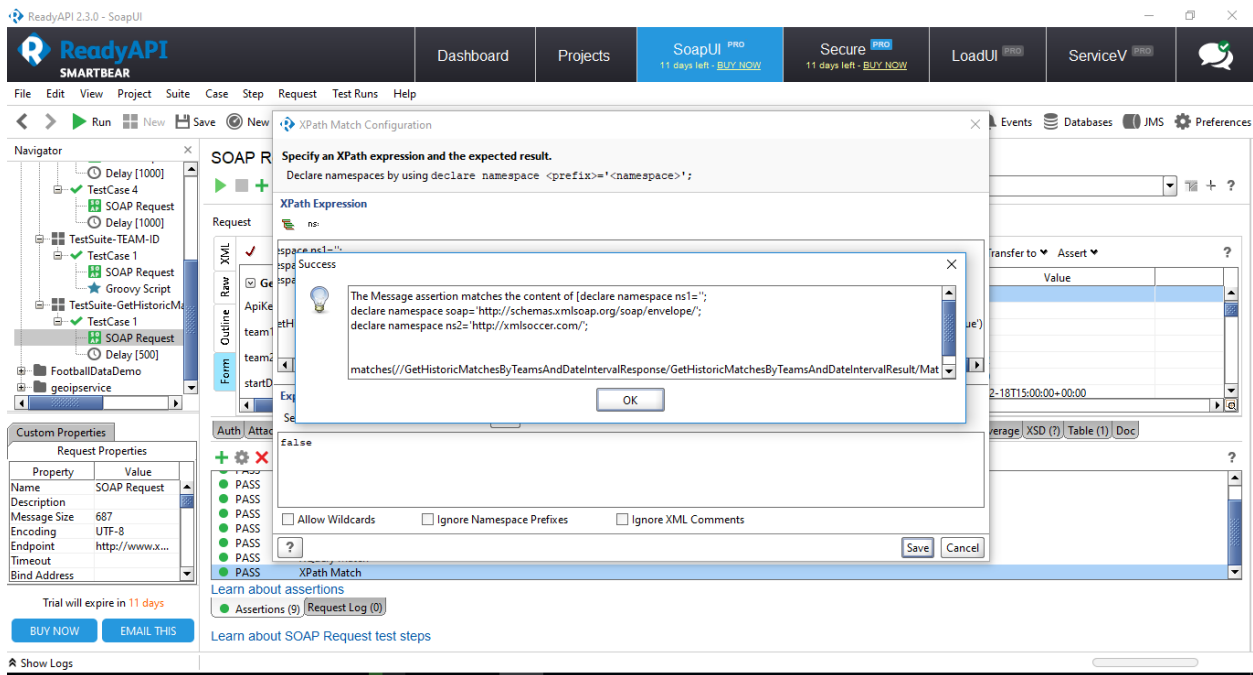


Figure 15 - XPath no ReadyApi

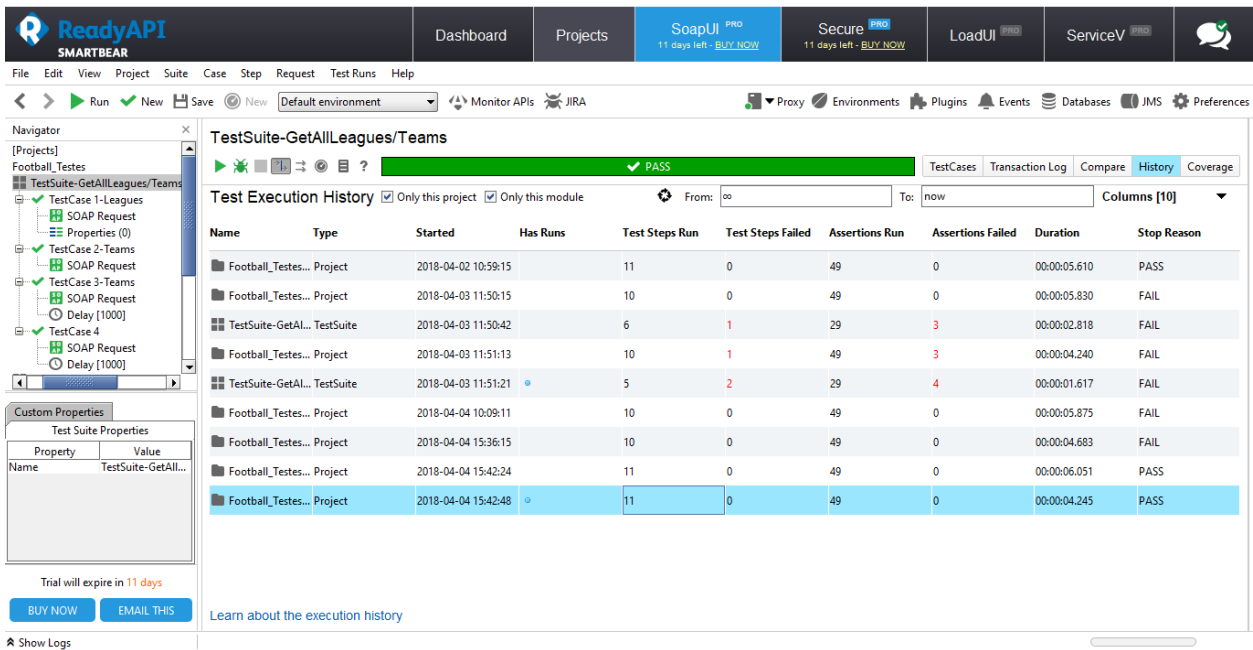


Figure 16 - Histórico da execução de testes

A figura descrita acima (fig. 16), mostra o histórico da execução de testes onde apresenta de num modo geral os testes *steps* que falham e que passam. Através da imagem é também possível obter informação do número de assertions que passaram ou falharam.

A figura 17 mostra também os resultados de testes, e como podemos visualizar esta fornece detalhes dos 11 steps que passaram, como também o dia que o teste foi realizado e o tempo de duração do teste.

The screenshot displays the SoapUI interface for a test project named 'Football_Testes'. The main window shows a summary of test results: 'Results: 11 Steps PASS 0 Steps FAIL'. Below this, a table lists the individual test steps, including their names, test cases, test suites, start and finish times, time taken, and status. All listed steps are marked as 'PASS'. The interface also includes a sidebar with a project tree, a 'Custom Properties' panel, and a top navigation bar with various tool options.

Name	Test Case	Test Suite	Start Time	Finish Time	Time Taken	Status
Football_Testes			2018-04-04 15:42:48.253	2018-04-04 15:42:52.498	4245ms	PASS
TestSuite-GetAllLeag...			2018-04-04 15:42:48.282	2018-04-04 15:42:51.362	3080ms	PASS
TestCase 1-Leagues		TestSuite-GetAllLeag...	2018-04-04 15:42:48.286	2018-04-04 15:42:48.566	280ms	PASS
SOAP Request	TestCase 1-Leagues	TestSuite-GetAllLeag...	2018-04-04 15:42:48.298	2018-04-04 15:42:48.578	280ms	PASS
Properties	TestCase 1-Leagues	TestSuite-GetAllLeag...	2018-04-04 15:42:48.665	2018-04-04 15:42:48.665	0ms	PASS

Figure 17 - Resultados dos testes com 11 steps que passam

Load Test

Realizei também testes de carga através de um teste *Load Test*, para verificar os limites dos dados processados pelo *Software*. A figura 18 mostra exatamente o teste no momento em que este é executado. Este novo teste foi também realizado com um novo *Webservice* uma vez que o outro falhava constantemente. O novo *Webservice* criado é também sobre *football* muito parecido com o anterior. A diferença entre ambos é que neste já não necessitava de Api's o que facilitava bastante a sua realização.

The screenshot displays the SoapUI 5.4.0 interface during a LoadTest execution. The main window shows the 'LoadTest 1' configuration with the following parameters:

- Threads: 5
- Strategy: Simple
- Test Delay: 1000
- Random: 0.5
- Limit: 60 (Seconds)
- 100% (Progress)

The execution results are shown in a table with the following columns: Test Step, min, max, avg, last, cnt, tps, bytes, bps, err, and rat.

Test Step	min	max	avg	last	cnt	tps	bytes	bps	err	rat
Properties (2)	1	6	0.14	0	48	0.77	0	0	0	0
SOAP AllCards	1231	6553	3,236.64	4348	48	0.77	9022504	145726	8	16
Delay [1000]	998	1001	999.47	999	40	0.64	0	0	0	0
SOAP AllDefenders	260	2635	467.82	364	40	0.64	26765	432	1	2
Delay 1 [1000]	998	1001	999.46	1001	39	0.62	0	0	0	0
TestCase:	3488	11196	5,271.6	6712	48	0.77	9049269	146158	9	18

Below the table, the 'Show Types' and 'Show Steps' are set to '- All -'. The log window shows the following entries:

time	type	step	message
2018-04-11 10:37:50.993	Message		LoadTest started at Wed Apr 11 10:37:50 BST 2018
2018-04-11 10:37:51.578	Step Status	SOAP AllCards	TestStep [SOAP AllCards] result status is FAILED; [[Script Assertio...
2018-04-11 10:37:51.581	Step Status	SOAP AllCards	TestStep [SOAP AllCards] result status is FAILED; [[Script Assertio...
2018-04-11 10:37:51.797	Step Status	SOAP AllCards	TestStep [SOAP AllCards] result status is FAILED; [[Contains-Franc...

The interface also shows a 'LoadTest Properties' table with the following data:

Property	Value
Name	LoadTest 1
Description	

At the bottom, there are links to various logs: SoapUI log, http log, jetty log, error log, wsrm log, memory log, and script log.

Figure 18 - LoadTest no SoapUI

A figura 19, ilustra também o teste de carga mas com um detalhe diferente onde se pode observar os dados estatísticos relativos aos teste mas de um modo gráfico.

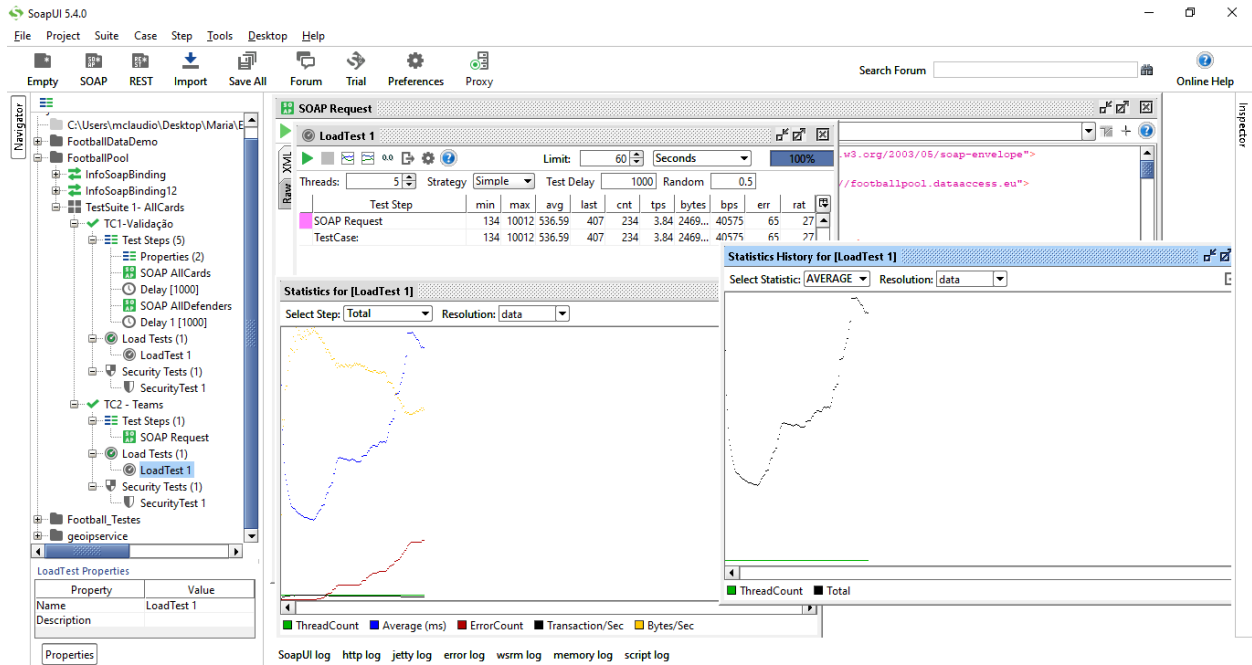


Figure 19 - Dados estatísticos no gráfico - SoapUI

Loop

A figura 20 ilustra a utilização do *Loop*, utilizado duas vezes com o objetivo que quando chegasse ao lugar do *loop* o teste volte a ser executado no início de forma a garantir que não falha em lado nenhum ou seja o *loop* faz uma repetição do que fez e garante que não existem falhas.

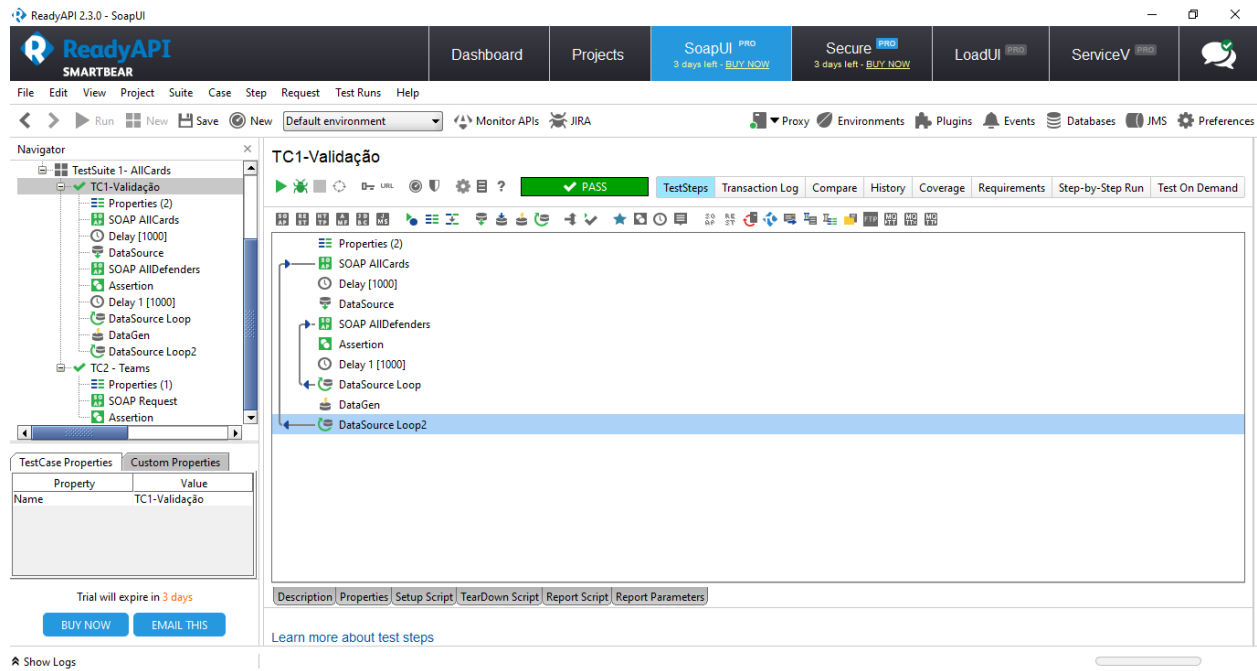


Figure 20 - Teste no ReadyApi com utilização do Loop

As figuras seguintes apresentam novos testes, criados no ReadyApi, como são exemplo o *Load Test* e o *Groovy Script*. O *Load test* (figura 21) já foi mencionado acima, sendo usado para verificar os limites dos dados processados pelo *Software*, a única diferença apresentada é que antes este tinha sido executado no SoapUi e neste caso o mesmo é executado no ReadyApi.

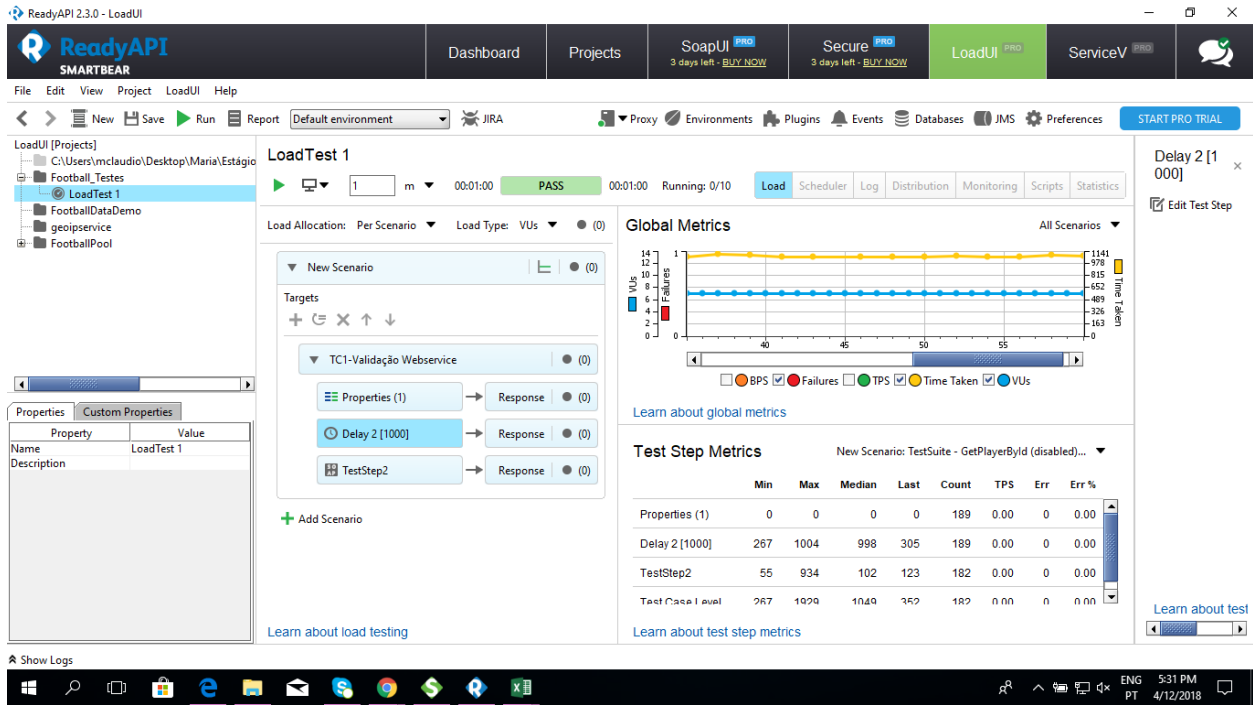


Figure 21 - Load Test no ReadyApi

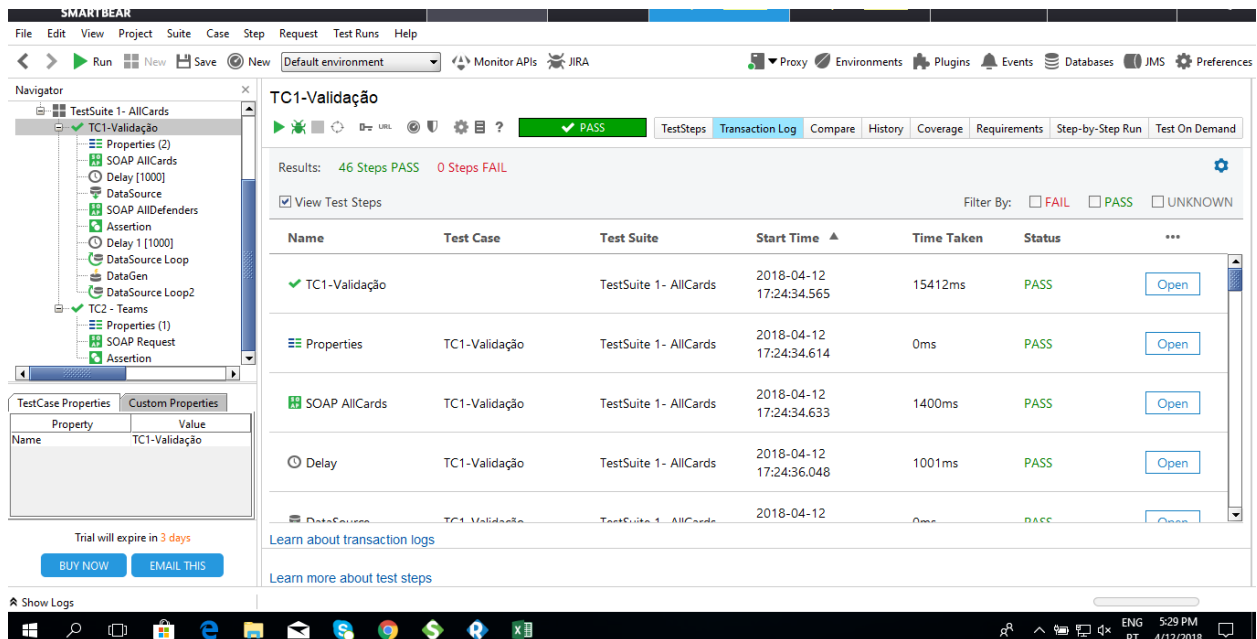


Figure 22 - Resultados dos testes com 46 steps que passam

Conditional Goto e Groovy Script

O goto condicional (figura 23), é simplesmente uma condição XPath usada para calcular valores. Na figura 24, está referenciado o código do *Groovy Script*. O código é uma linguagem de programação orientada a objetos desenvolvida para a plataforma Java como alternativa à linguagem de programação Java. Nas próximas duas figuras (figura 25 e 26), demonstram os testes a *Pass* após finalizados.

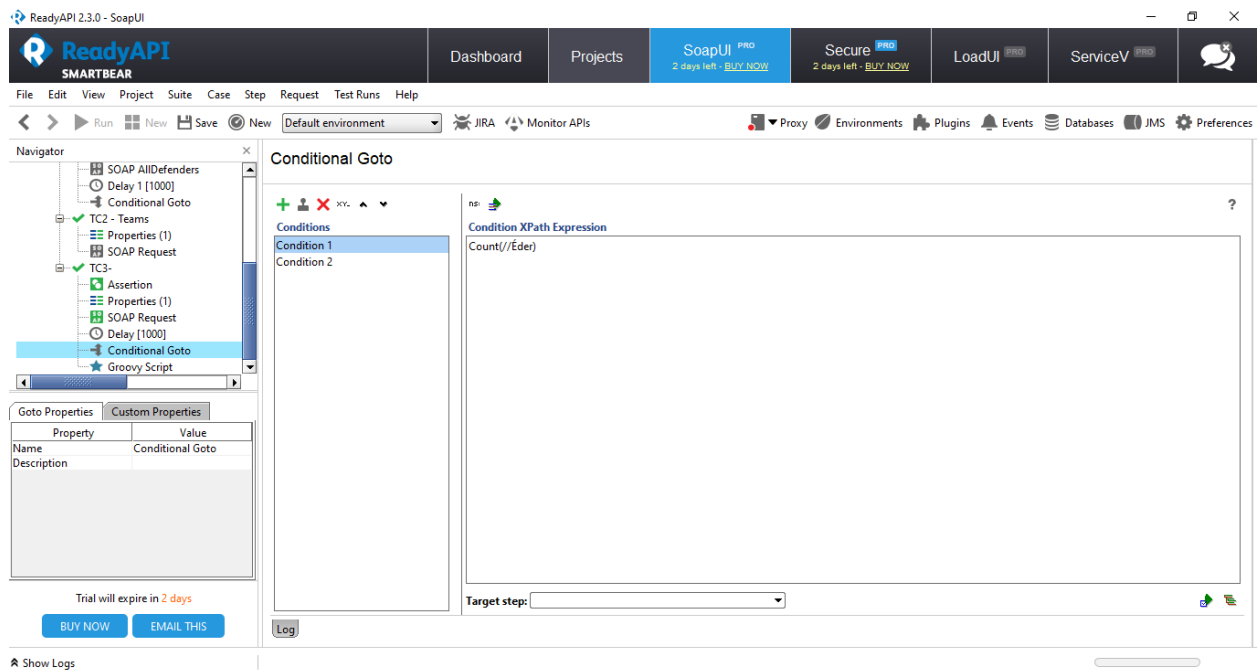


Figure 23 - Condição XPath - Goto Condicional

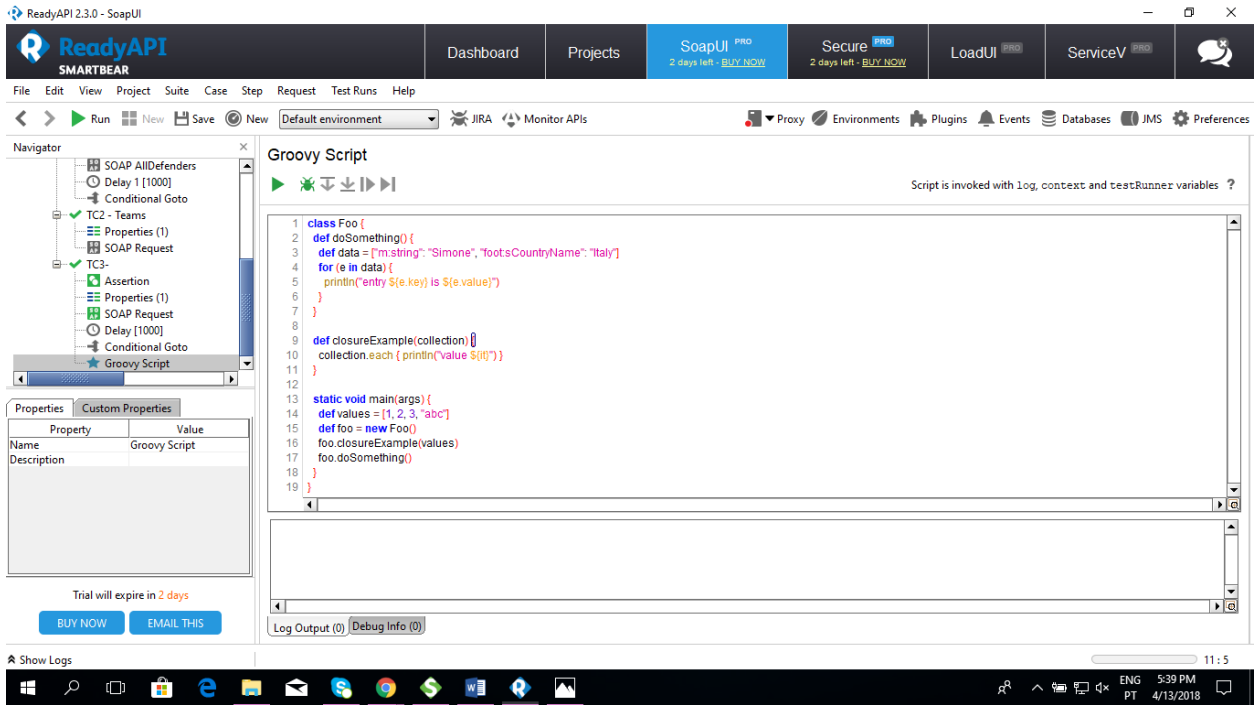


Figure 24 - Groovy Script

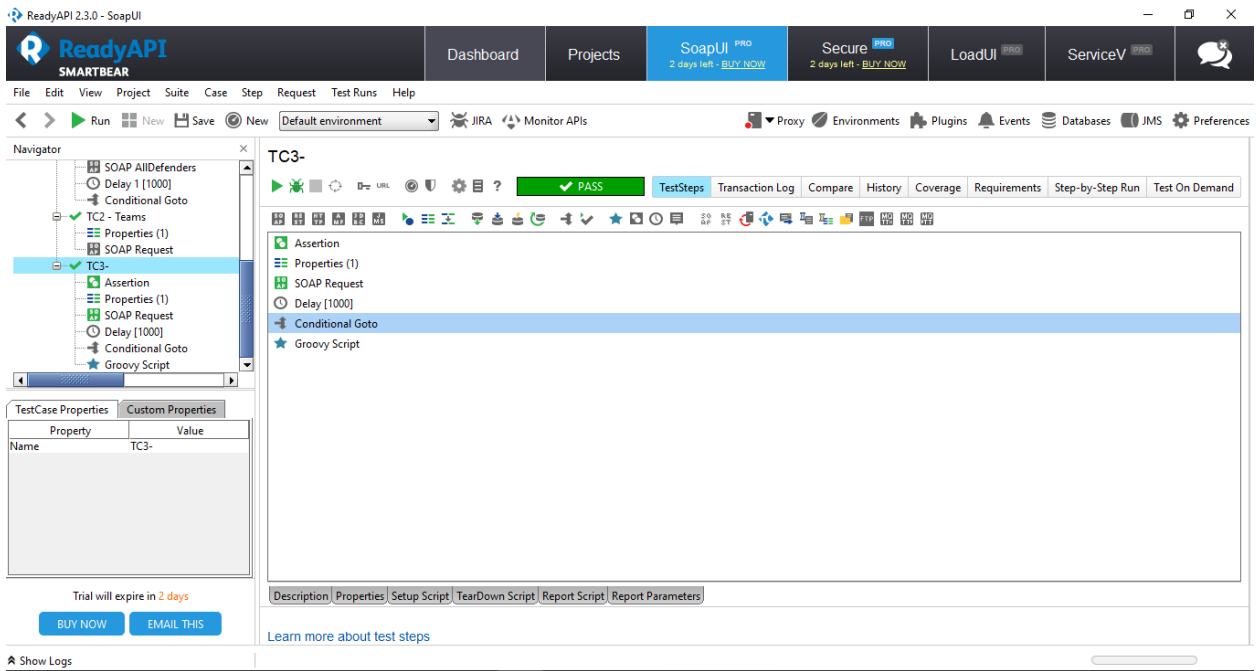


Figure 25 - Teste Completo

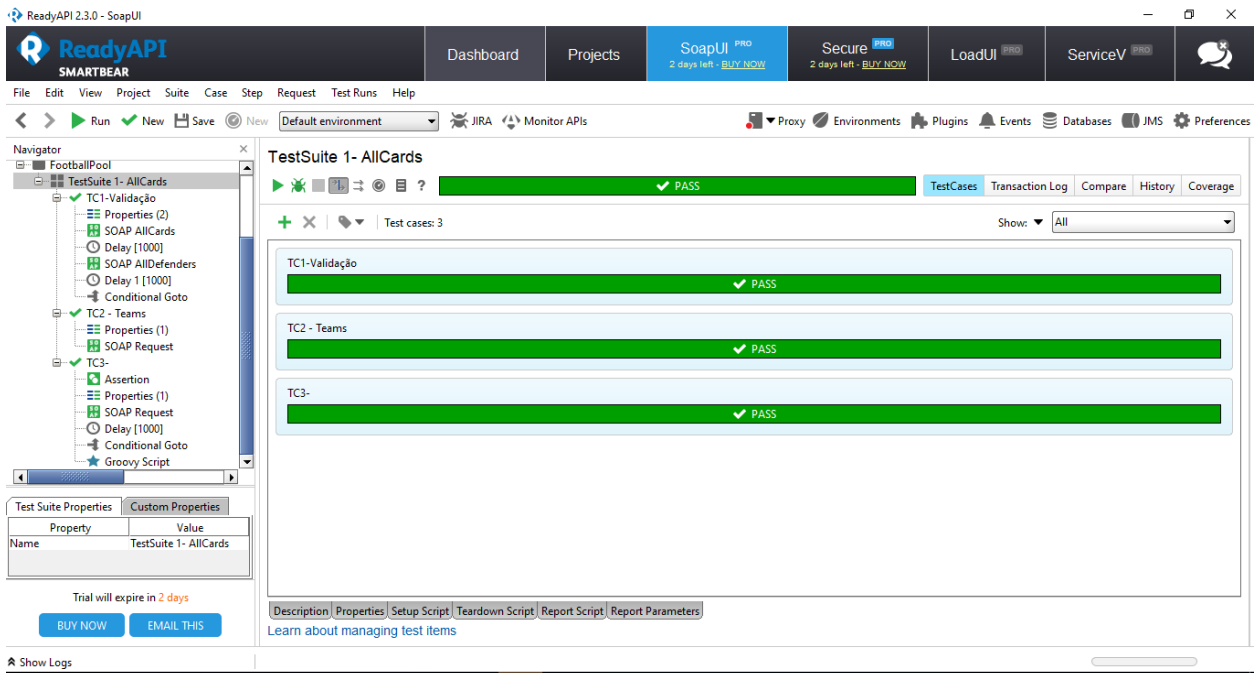


Figure 26 - Execução de testes terminada

Testes SoapUi

Após terminar o tempo de utilização do ReadyApi, foi necessário continuar a trabalhar no SoapUi (fig.27). O próximo passo foi o de continuar a realizar novos testes e fazer também o novo caderno de testes no Excel, os testes são basicamente iguais aos antigos, muda apenas o contexto que se enquadra. A figura seguinte (fig.28), mostra-nos o caderno de testes no excel, onde foram explícitos todos os casos de teste possíveis para cada *Test Case* e *Test Step*.

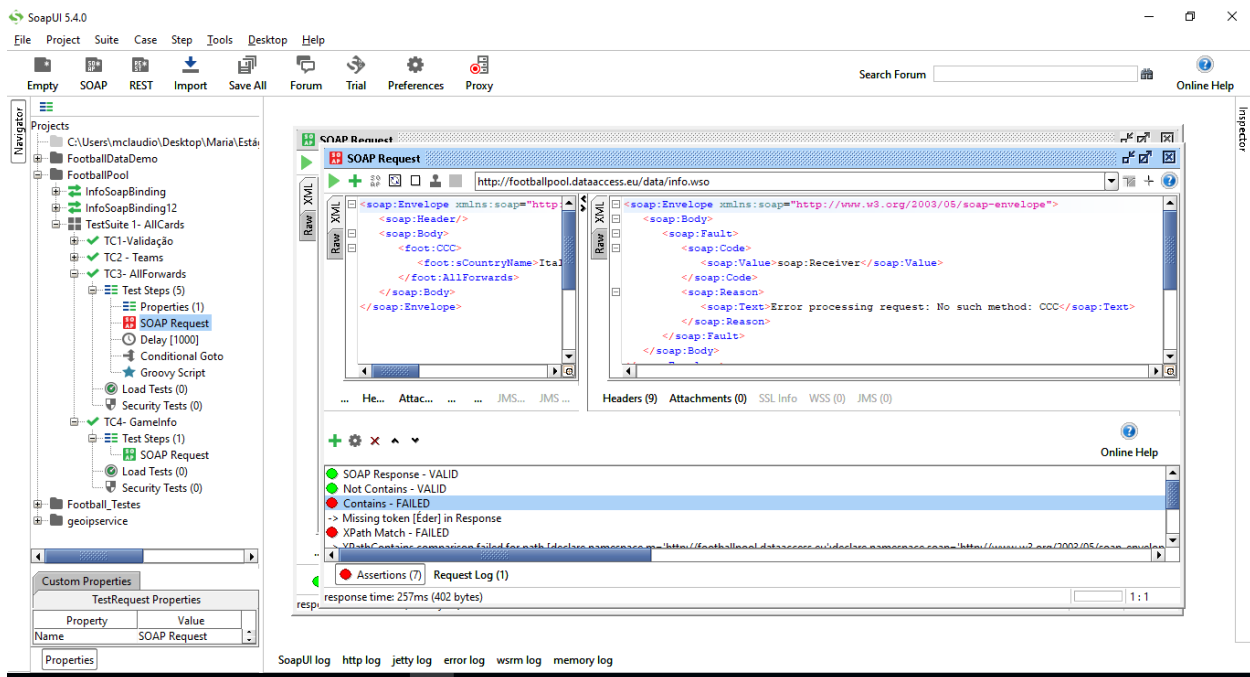


Figure 27 - Novos testes SoapUI com novo Webservice

	Precondition	Assigned To (nome do Tester)	Priority (Critical, High, Medium, Low)	Type (Manual, Automated)	Test Case ID	Test Case Steps ID	Test Case - Description	Test Step - Description	Test Data	Expected Result	Actual Result
1											
2	Aplicação SoapUI inicializada	MC	H	M	TC 1	1	Validar se o webservice responde	Validar AllCards	Submit Request	Retorna Id, names, ...	Erro
3	Aplicação SoapUI inicializada	MC	H	M	TC 1	2	Validar se o webservice responde	Validar AllCards	Criar assertions Contains com valor de " France "	Valid	valid
4	Aplicação SoapUI inicializada	MC	H	M	TC 1	3	Validar se o webservice responde	Validar AllCards	Criar assertions Contains com valor de " Razvan Rat "	O teste passa	O teste passa
5	Aplicação SoapUI inicializada	MC	H	M	TC 1	4	Validar se o webservice responde	Validar AllCards	Criar assertions Contains com valor " m:sPlayerName "	O teste passa	O teste passa
6	Aplicação SoapUI inicializada	MC	H	M	TC 1	5	Validar se o webservice responde	Validar AllCards	Criar assertions NotContains com valor de " Maria "	O teste passa porque não contem o valor Maria	O teste passa
7	Aplicação SoapUI inicializada	MC	H	M	TC 1	6	Validar se o webservice responde	Validar AllCards	Criar assertions XPath onde confirma se contem " count(/GameTeam2=Romania) "	Retorna se é verdade	Retornou o valor 1
8	Aplicação SoapUI inicializada	MC	H	M	TC 1	7	Validar se o webservice responde	Validar AllCards	Criar assertions XPath onde confirma se contem " matches(/AllCardsResponse/AllCardsResult/CardInfo[1]/PlayerName,'Vlad Chiriches') "	Retorna o valor	Retornou o valor 1
9									Criar assertions Xquery onde é pedido " <Result>		Retornou o

Figure 28 - Cadernos de casos de teste - Excel

Após a utilização e compreensão destas ferramentas foi-me finalmente possível integrar o projeto Bouygues e sua equipa. De forma breve passo a descrever as tarefas aí executadas.

Projeto Bouygues

Iniciei as minhas tarefas no projeto lendo inicialmente documentação sobre o mesmo. Após a leitura da documentação forma me atribuídas as seguintes etapas que consistiram em:

- Criar fichas de clientes e realizar casos de teste;
- Lançar scripts em linhas de comandos;
- Verificar linha de cliente em base de dados;
- Lançar e avançar *Webservices* através do SoapUi.

Devido a ter integrado um projeto confidencial, com termos de confidencialidade, não me é permitido expor imagens, nem falar mais, de forma detalhada, do projeto na qual fui integrada.

4. Conclusão

Julgo que cumpri o plano de estágio proposto para este estágio curricular, porque para além de ter feito tudo o que me foi inicialmente proposto, tentei sempre fazer mais um pouco e saber ainda mais.

Tentei ser sempre prestável e tentei superar todas as dúvidas que surgiram, pesquisando de forma autónoma ou por vezes questionando o meu coordenador ou alguém da equipa. Posso referir que estes sempre se mostraram disponíveis em ajudar.

Este estágio foi uma mais-valia, enquanto estudante do curso Testes de Software, pois foi possível aplicar alguns conhecimentos e conhecer e explorar novos programas que possibilitam a criação de testes.

Este trabalho reflete, assim, os conhecimentos adquiridos durante 2 anos letivos.

5. Bibliografia

- [1] Altran, “Overview”. Acedido a 12 de Abril de 2018 em:
<https://www.altran.com/us/en/about-us/overview/>
- [2] Altran, “Quem somos – Altran Portugal”. Acedido em 12 de Abril de 2018 em:
https://www.altran.com/pt/pt-pt/quem-somos/#.V2lk0_krIdV
- [3] Altran, “Nearshore – Altran Portugal”. Acedido a 13 de Abril em:
<http://www.altran.pt/os-nossos-servicos/solucoes/information-systems/nearshore.html#.WzoBHvZFzIV>
- [4] Soap, “Soap Quick Guide”. Acedido a 19 de Março de 2018 em:
https://www.tutorialspoint.com/soap/soap_quick_guide.htm
- [5] SoapUi, “SoapUi”. Acedido a 21 de Março de 2018 em:
<https://en.wikipedia.org/wiki/SoapUI>
- [6] SoapUi, “Soap and WSDL”. Acedido a 19 de Março de 2018 em:
<https://www.soapui.org/soap-and-wsdl/working-with-wsdl.html>
- [7] Webservices, “Webservices”. Acedido a 20 de Março de 2018 em:
<https://www.tutorialspoint.com/webservices/index.htm>
- [8] XML, “Xml”. Acedido a 25 de Março de 2018 em:
https://www.w3schools.com/xml/xml_what.asp
- [9] XML, “Xml”. Acedido a 25 de Março de 2018 em:
<https://pt.wikipedia.org/wiki/XML>