

RELATÓRIO – *ELABORATION & CONSTRUCTION*

Construção

Conteúdos

Construção	1
1 Introdução	2
1.1 Sumário executivo	2
1.2 Controlo de versões	2
1.3 Referências e recursos suplementares	2
2 Arquitetura do sistema	3
2.1 Objetivos gerais	3
2.2 Requisitos com impacto na arquitetura	3
2.3 Decisões e justificação	4
2.4 Arquitetura do software	4
2.5 Arquitetura física de instalação	8
3 Incremento 1	9
3.1 Casos de utilização no Incremento 1	9
3.2 Histórias de utilização selecionadas	10
3.3 Estratégia e estado da implementação	11
4 Incremento 2	11
4.1 Casos de utilização no incremento 2	11
4.2 Histórias de utilização selecionadas	12
4.3 Aceitação e garantia de qualidade	13
4.4 Estado da implementação	15
Apêndice	16
5 Especificação dos casos de utilização	16
5.1 Pacote: compra online	16
5.1.1 CaU 1 Nome do caso aqui	16
5.1.2 CaU 7 Outro caso aqui	16
5.2 Pacote: gestão de parcerias	18
5.2.1 CaU 7 Outro caso aqui	18

1 Introdução

1.1 Sumário executivo

Este relatório apresenta os resultados da construção dos incrementos, adaptado os resultados esperados na etapa de *Elaboration* e *Construction*, do método OpenUP.

A caracterização dos cenários a suportado é detalhada nos casos de utilização apresentados em apêndice (secção 5)

O primeiro incremento, desenvolvido na Iteração 3, foca a validação da arquitetura proposta. Foram considerados sobretudo as funcionalidades relacionadas com o *core* do negócio, como por exemplo a pesquisa de um produto (Onde o cliente vai à shop e procura o produto que deseja ou comprar ou então pode ver o catalogo e adicionar produtos que chamem a atenção), e a compra de produto (Após o Cliente ter procurado os seus produto ele pode adiciona-los ao carrinho e assim que tiver acabado de adicionar todos os produtos que deseja apenas tem de carregar em checkout.

1.2 Controlo de versões

Quando?	Responsável	Alterações significativas
06/06/2022	Tiago Coelho	Escrita 2.4, 2.2
08/06/2022	Frederico Vieira	Escrita 2.1, 2.3, 3.3, 3.2, 3.1
09/06/2022	João Torrinhas	Escrita do ponto 5.1
09/06/2022	Tiago Coelho	Escrita ponto 1.3 e 2.5
10/06/2022	João Torrinhas	Finalização do ponto 5
10/06/2022	Diogo Torrinhas	Sumário executivo. Ponto 1.1
23/06/2022	João Torrinhas	Resolução do ponto 4.4 e início da resolução do ponto 4.2
23/06/2022	Tiago Coelho e Frederico Vieira	Resolução do ponto 4.3, criação do bot para testes e print de resultados
24/06/2022	João Torrinhas	Concluir a resolução do ponto 4.2
24/06/2022	Diogo Torrinhas	Escrita do ponto 4.1

1.3 Referências e recursos suplementares

Uma das motivações para a escolha de desenvolvimento de uma aplicação web foi a consulta de documentos que comprovem essa escolha, tais como o site (<https://www.comerciodigital.pt/media/2586/acepi-idc-estudo-da-economia-digital-em-portugal-2020.pdf>), que refere que “O telemóvel é o equipamento utilizado com mais frequência para aceder à internet. O computador pessoal mantém-se como um dos equipamentos principais e é o preferido quando se trata de efetuar compras online.”, assim escolheu-se aplicação web, por forma a fornecer aos principais dispositivos o acesso à aplicação.

O documento possui outras referências sobre o estudo da economia digital que suplementaram os conteúdos deste relatório.

2 Arquitetura do sistema

2.1 Objetivos gerais

Os clientes devem poder aceder à loja em qualquer lado, a partir de um browser, sem necessidade de instalar software específico. Para o login pode criar uma conta ou inicializar com o facebook ou google. A qualidade da experiência de utilização em ambientes desktop e mobile deve ser comparável.

O Cliente deve conseguir realizar a pesquisa dos seus produtos em tempo quase real e com suporte aos produtos mais vendidos no estabelecimento.

Os Fornecedores deveram ser notificados de uma encomenda assim que a mesma seja finalizada pelo cliente.

O Cliente deverá conseguir ver os estados das suas encomendas de algum modo acompanhar algum desse progresso. Por exemplo: “Encomenda está a ser preparada” ou “Encomenda está a caminho”.

Deverá ser possível para os Clientes ver todas as encomendas já realizadas, carteiras guardadas e moradas guardadas.

2.2 Requisitos com impacto na arquitetura

Requisitos	Descrição
Rint.5	O sistema deve suportar as integrações com sistemas externos, no caso Facebook e Google para login e também com a empresa de transportes.
RSeg.1 e RSeg.3	Todos os dados que o sistema usar para pagamentos, como os dados do pagamento e do cliente devem ser e permanecer privados, tendo assim a necessidade de o sistema possuir uma base de dados totalmente segura, guardando os dados de forma cifrada.
Rint.2	O sistema deve ter um design que ajude na recolha de informação, para tal usa-se o css e bootstrap.
Rint.6 (novo)	Disponibilidade do sistema para qualquer cliente que pretende aceder e que possam aceder em simultâneo sem problemas a qualquer momento.
Rint.7 (novo)	O portal da loja deve-se ajustar para ter uma apresentação adequada ao ecrã, designadamente para <i>smartphones</i> , <i>tablets</i> ou sistemas de secretária, tendo-se por esta razão escolhido a aplicação Web.

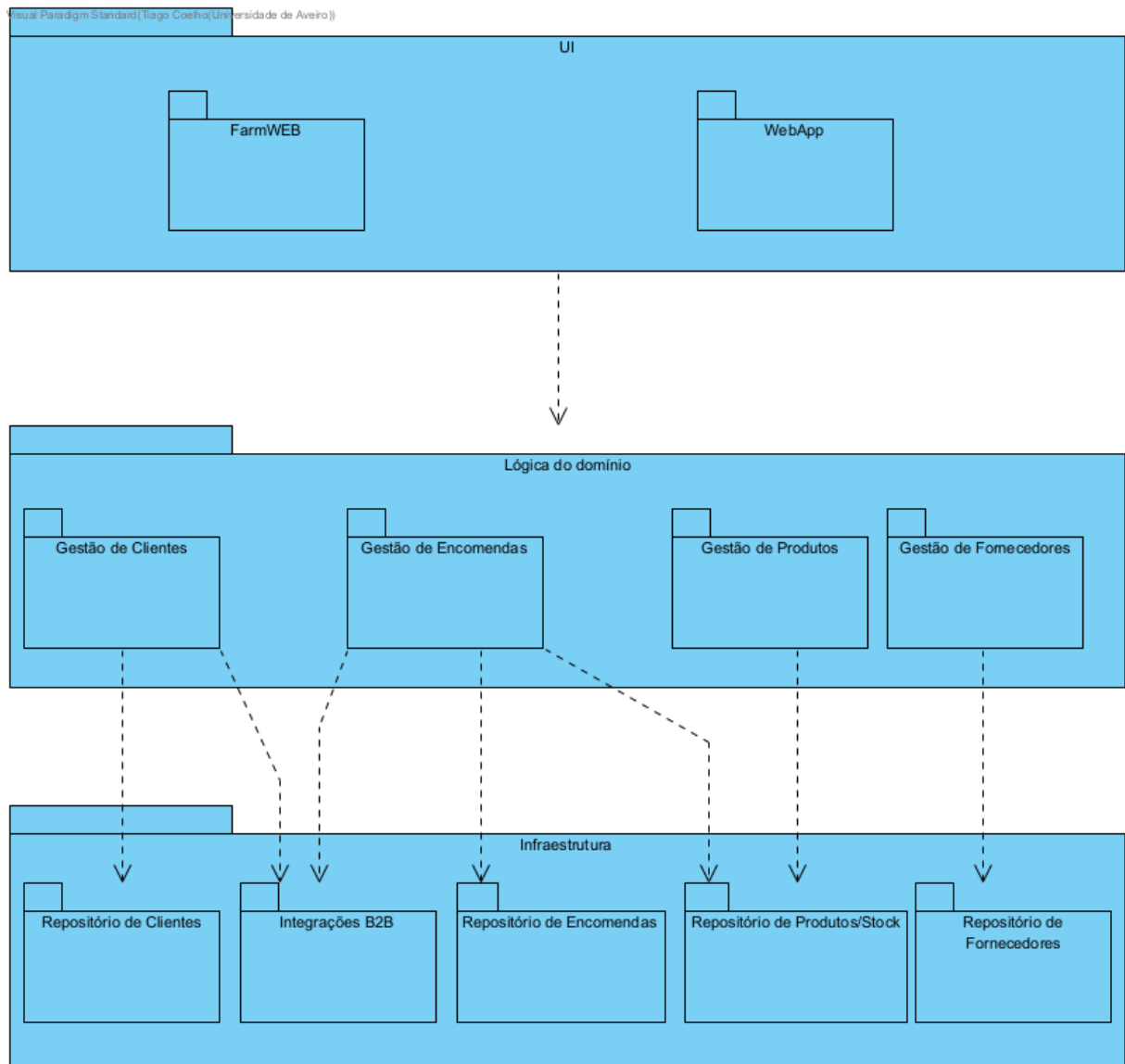
2.3 Decisões e justificação

Tendo em conta os objetivos para a arquitetura, e os requisitos levantados na Análise, foram tomadas as seguintes decisões:

- I. FrontEnd vai ser implementado usando HTML, CSS, javascript, BootStrap, pois temos conhecimento e facilidade com estas ferramentas.
- II. BackEnd deveria ser python flask para servidor e SQL para base de dados, pois para além de termos conhecimento é bastante simples a interação com python flask e SQL fornece uma base de dados solida e estável, no entanto no ambito da disiplina e como nos foi sugerido vai ser usado local storage.
- III. Para a construção será utilizado o Visual Studio Code.
- IV. Para a gestão de código será utilizado o Git Hub.

2.4 Arquitetura do software

Para demonstrar como o software vai estar organizado foi escolhido uma vista lógica, usando uma arquitetura lógica por camadas, visualizadas num diagrama de pacotes e recorrendo à relação de dependência, podendo assim explicar a divisão do sistema em partes. É possível identificar 3 camadas distintas, User Interface, Lógica do domínio e Infraestrutura.



A articulação entre os módulos decorre da seguinte forma:

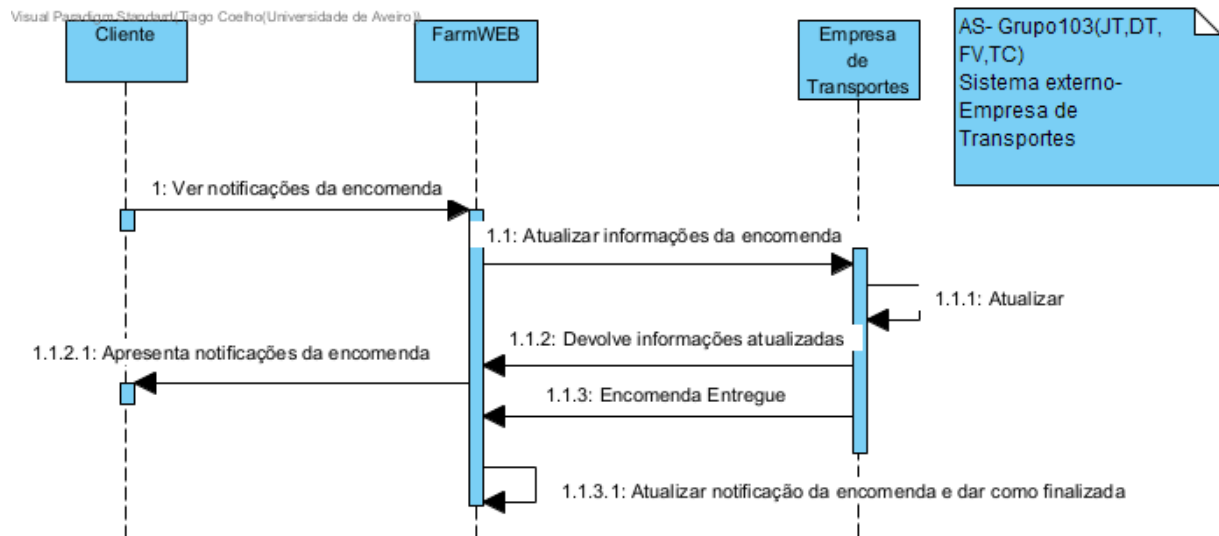
Primeiramente, temos a camada de User Interface, que é um módulo front-end que retrata a parte Web onde ocorrem as interações entre o cliente e o servidor. Este módulo possui a FarmWEB e a WebApp, que se articula com a Lógica do domínio.

A camada de lógica do domínio é o módulo do back-end que é responsável por gerir os serviços associados aos produtos e utilizadores da plataforma, que sendo assim possui diversos “gestores”, tais como Gestão de Clientes, Gestão de Encomendas, Gestão de Produtos e Gestão de Fomecedores.

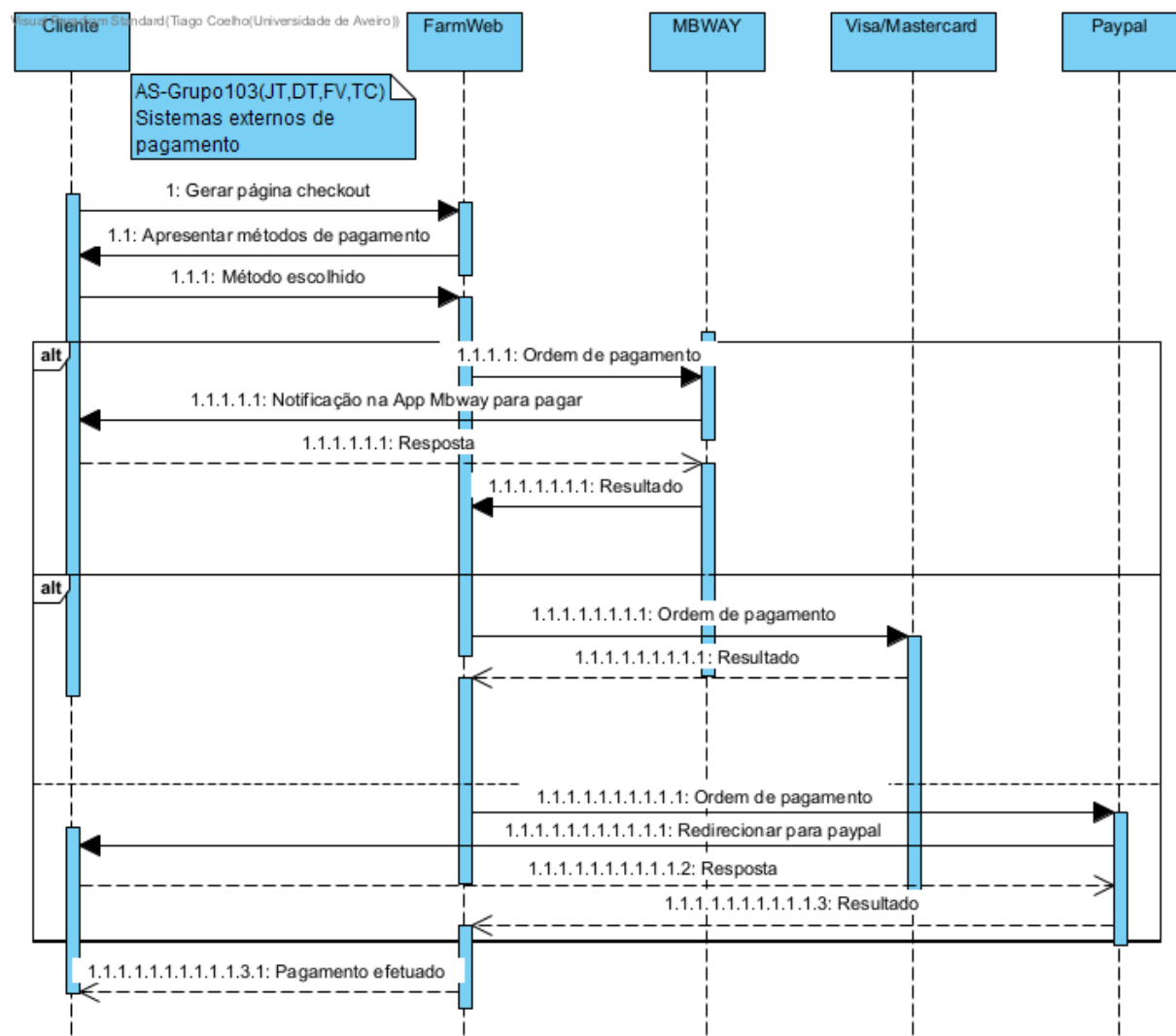
Esta camada consequentemente articula-se com a camada de Infraestrutura, em que estes “gestores” se articulam com os devidos repositórios, tais como Repositório de Clientes, Integrações B2B, Repositório de Encomendas, Repositório de Produtos/Stock e Repositório de Fomecedores. Estes “gestores” interagem com um ou vários repositórios como se pode ver no diagrama.

Assim, a camada de Infraestrutura é o módulo do back-end que é responsável por gerir os dados associados aos produtos e utilizadores da plataforma e pelas funções de acessos e armazenamento dos dados.

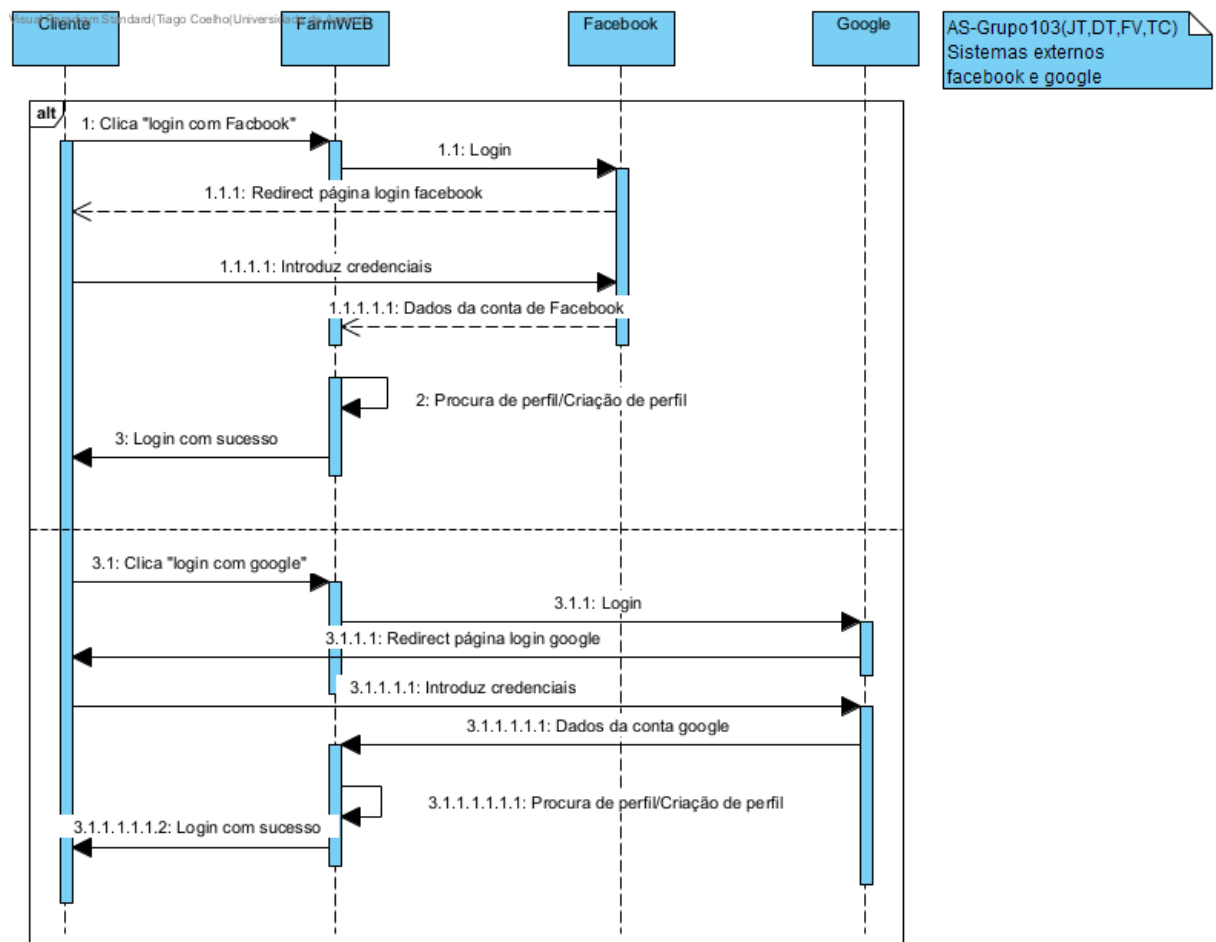
Integração com o Sistema externo da Empresa de transportes:



Integração com os Sistemas externos de pagamento:

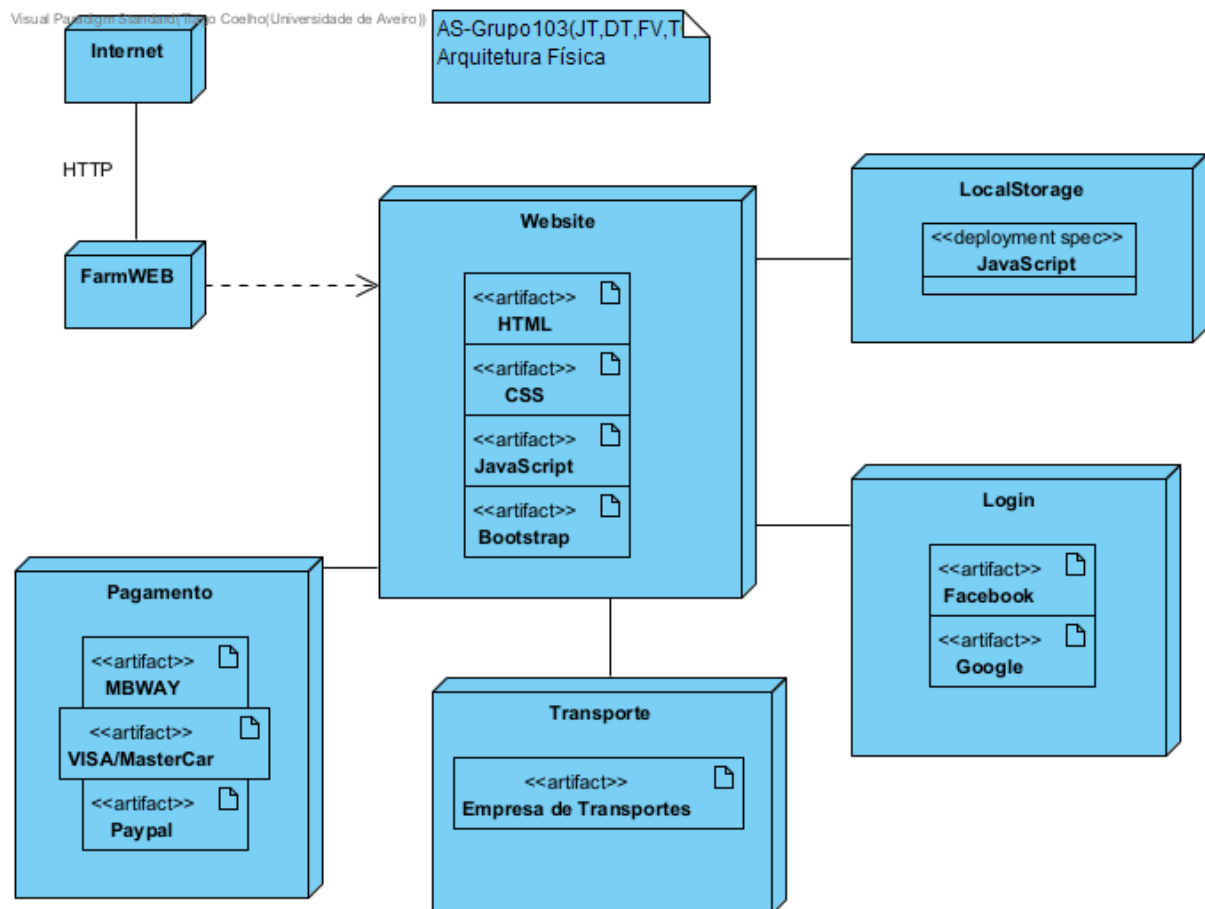


Integração com os Sistemas externos Facebook e Google:



Nota: A integração com estes sistemas externos é meramente ilustrativa, pois não temos conhecimento de como estas integrações funcionam realmente.

2.5 Arquitetura física de instalação



O cliente acede à aplicação web através da internet, a aplicação pode ser acedida por qualquer dispositivo que possua ligação à internet e um browser. O conteúdo da app é apresentado via html, css, bootstrap e JavaScript. Para guardar dados e gerir os mesmos é usado LocalStorage através do JavaScript. No caso dos sistemas externos, temos o Login que tem as Integrações com Facebook e Google, temos o Transporte que faz integração com a Empresa de Transportes e temos o Pagamento que faz integração com a MBWAY, VISA/MasterCard e Paypal.

3 Incremento 1

3.1 Casos de utilização no Incremento 1

No primeiro incremento implementado, o foco esteve na validação da arquitetura proposta, através da implementação de funcionalidade representativa do *core* do negócio. Para isso, selecionámos

O caso de utilização 3 – Pesquisa de um produto (Onde o cliente vai à shop e procura o produto que deseja ou comprar ou então pode ver o catalogo e adicionar produtos que chamem a atenção), o caso de utilização 4- Compra de Produto (Após o Cliente ter procurado os seus produto ele pode adicioná-los ao carrinho e assim que tiver acabado de adicionar todos os produtos que deseja apenas tem de carregar em checkout, vai ser redirecionado para outra página onde tem de preencher o formulário de seguida basta efetuar o pagamento), caso de utilização 7 (novo) - Fornecedor Adicionar ou remover produtos do catálogo (Após ter realizado o login como Fornecedor é possível adicionar produtos basta ir à pagina adicionar produto. É possível o fornecedor adicionar produtos da base de dados ou então adicioná-los manual preenchendo um formulário sobre o produto. Também pode remover produtos basta ir à página de remover produtos e pode remover o/os produto/s introduzidos).

A especificação detalhada dos casos de utilização encontra-se em anexo (secção 5). A partir dessa análise, definiram-se as histórias de utilização a implementar.

3.2 Histórias de utilização selecionadas

As histórias (*user stories*) incluídas nesta interação fazem parte do *backlog* do projeto, acessíveis em <https://www.pivotaltracker.com/n/projects/2575609>, ou no relatório E1 através de prints.

Histórias incluídas nesta interação:

História/ <i>use case slice</i>	Critérios de aceitação
O Fernando Almeida Adiciona Produtos ao Catálogo Sendo o Fernando, um Fornecedor do site FarmWeb, Quero Adicionar ao stock do site 100 melancias e 100 Broccolo.	Cenário 1: Adicionar Produtos Manualmente Estou na página inicial clico em dar login como fornecedor. Dou login como fornecedor de seguida volto para a pagina inicial ja logado como fornecedor Vou ao add products e meto adicionar produto De Seguida preencho o formulário apresentado sobre o produto e adiciono Após adicionar verifico se o produto foi adicionado ao catalogo com sucesso E posso verificar que sim o produto foi adicionado com sucesso. Cenário 2: Adicionar Produtos Com Database Estou na página inicial clico em dar login como fornecedor. Dou login como fornecedor de seguida volto para a pagina inicial ja logado como fornecedor Vou ao add products e meto adicionar produto da Database Após adicionar verifico se o produto foi adicionado ao catalogo com sucesso E posso verificar que sim o produto foi adicionado com sucesso. Cenário 3: Remover Produtos do Catálogo Estou na página inicial clico em dar login como fornecedor. Dou login como fornecedor de seguida volto para a pagina inicial ja logado como fornecedor Vou ao Remove products e remove um produto que já não esteja em stock Após remover verifico se o produto foi removido do catalogo com sucesso E posso verificar que sim o produto foi removido com sucesso.
O Paulo Carvalho Pretende Comprar Produtos Sendo o Paulo um dono de um restaurante. Necessito de comprar 30 melancias e 15 Broccolo para o meu restaurante	Cenário 1: Pesquisar os Produtos No catalogo e Compra-los Estou na página inicial clico em dar login como Cliente. Dou login como Cliente de seguida volto para a pagina inicial ja logado como Cliente Vou a shop para aceder ao catalogo. De Seguida procuro os produtos que procuro e vou adiciona-los ao carrinho Após adicionar ao carrinho verifico que aparece no ecrã o meu carrinho Como já terminei as minhas comprar clico em checkout e sou direcionado para outra pagina Após preencher o formulário com os meus dados e clicar para efetuar o pagamento, finalizo a minha encomenda. Cenário 2: Ver Encomendas

	Após ser realizado um pagamento ou noutra ocasião é possível, caso já tenha realizado alguma encomenda, ver se for a encomendas verificar as encomendas que já efetuou.
--	---

3.3 Estratégia e estado da implementação

De momento foi implementado grande parte do design da plataforma e também foi implementado as pesquisas por produtos e a possibilidade de adicionar ou remover produtos do carrinho também é possível, sempre que é efetuado o pagamento de uma encomenda, ver as encomendas efetuadas. Na parte dos fornecedores também já foi implementado a parte de ser possível cada fornecedor adicionar produtos à loja ou removê-los. Neste caso o fornecedor consegue gerir os produtos na aplicação podem ser produtos já introduzidos na base de dados os produtos manualmente.

Foi utilizado o HTML e CSS e Bootstrap para o design de cada uma das páginas. JavaScript foi utilizado para realizar as pesquisas de produtos e adicionar os mesmo ao carrinho, a possibilidade de adicionar ou remover produtos como fornecedor assim como a criação da lista de encomendas após cada compra. Tudo isto é guardado localmente utilizando local storage tendo sido sugerido no âmbito desta disciplina.

4 Incremento 2

4.1 Casos de utilização no incremento 2

A especificação detalhada dos casos de utilização encontra-se em anexo (secção 5).

No segundo incremento implementado, o foco esteve no desenvolvimento da visualização e *tracking* das encomendas realizadas, nomeadamente o Cau6(Acompanhar o estado da encomenda): o utilizador depois de ter efetuada a compra da sua encomenda, consegue ver o progresso da mesma; e o Cau8(Utilizador visualiza o histórico de encomendas): o utilizador consegue ver o histórico de encomendas realizadas anteriormente. Assim o utilizador consegue verificar se as encomendas que realizou ficaram guardadas e também consegue fazer o acompanhamento das mesmas. Posto isto, o incremento desenvolvido consistiu na criação destas duas páginas de forma que o cliente possa acompanhar todo o processo da sua encomenda, depois de ter completado a sua compra.

4.2 Histórias de utilização selecionadas

Histórias desenvolvidas nesta interação:

História/ <i>use case slice</i>	Critérios de aceitação
O Filipe pretende visualizar as encomendas realizadas Sendo o Filipe, um visitante do site FarmWeb, Quero ver as encomendas que eu realizei	Cenário 1: Realizou encomendas Entro na aplicação Web e pressiono o botão “login cliente”. Depois de a minha autenticação ser validada, com sucesso, pressiono o botão “minha conta”. Vou ver os meus pedidos e vou verificar se a compra das 5 melancias e 5 maçãs ficou guardada. Quando abro os meus pedidos verifico que a minha encomenda ficou guardada e agora só preciso de esperar no meu domicílio. Cenário 2: Não fez nenhuma encomenda Entro na aplicação Web e pressiono o botão “login cliente”. Depois de a minha autenticação ser validada, com sucesso, pressiono o botão “minha conta”. Quando entro nos meus pedidos vejo que não tenho nenhuma encomenda, uma vez que, todas as encomendas que tinha ativas já foram entregues no meu domicílio.
O Filipe pretende ver o estado das suas encomendas Sendo o Filipe, um visitante do site FarmWeb, Quero ver o estado das encomendas que eu realizei	Cenário 1: Vê as encomendas e o estado delas Entro na aplicação Web e pressiono o botão “login cliente”. Depois de a minha autenticação ser validada, com sucesso, pressiono o botão “minha conta”. Eu tinha encomendado 5 maçãs e 5 melancias, por isso, quero ver o estado em que se encontra a minha encomenda. Abro a página com os meus pedidos e encontro a minha encomenda. Carrego para ver o estado dela e vejo que está pronta para ser entregue. Portanto, agora só preciso de esperar até chegar ao meu domicílio.

4.3 Aceitação e garantia de qualidade

As histórias incluídas nesta secção foram testadas com um bot criado com Selenium em Python. O bot dá login como fornecedor, adiciona produtos da database e dá logout. De seguida, dá login como cliente “test”, dirige-se à loja e compra 3 bananas. Dá checkout e vai até à página de “My account” e “Your Orders”. Aqui visualiza as suas encomendas como na primeira história. De seguida clica para ver o tracking e visualiza o estado da encomenda completando a segunda história implementada. O bot em cada passo escreve num ficheiro todos os passos e guarda.

Aqui em baixo poderá ver os resultados que o bot printa como o código do bot.

Nota: O código e resultados pode ser também encontrado na pasta “Bot Selenium” no repositório do GitHub.

```
Dei login como fornecedor!  
Adicionei Produtos da database!  
Sai da conta do fornecedor!  
Dei login como cliente!  
Entrei na loja!  
Adicionei uma banana ao carrinho!  
Adicionei duas bananas ao carrinho!  
Adicionei tres bananas ao carrinho!  
Dei checkout!  
Completei o pedido!  
Cheguei a pagina do historico de encomendas!  
Cheguei a pagina de tracking de encomendas!  
Completei o teste!
```

```

import selenium
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC
from selenium.webdriver.chrome.options import Options
import time
import os

driver = webdriver.Chrome(
    executable_path="/Users/TiagoC/Desktop/chromedriver"
)

driver.get("https://trcoelhoo.github.io/AS-FARMWEB/")
f = open("results.txt", "a")
driver.maximize_window()
time.sleep(1)
driver.find_element_by_xpath('/html/body/header/nav/div[1]/div[2]/ul/li[5]/a').click()
time.sleep(1)
#driver.switch_to.frame("textarea_iframe")
driver.find_element_by_id("email").send_keys("test")
time.sleep(1)
driver.find_element_by_id("password").send_keys("testpass")
time.sleep(1)
driver.find_element_by_class_name("btnadmin1").click()
time.sleep(1)
f.write("Dei login como fornecedor!\n")
driver.find_element_by_xpath('/html/body/header/nav/div[1]/div[2]/ul/li[5]/a').click()
time.sleep(1)
driver.find_element_by_class_name("btnadmin3").click()
time.sleep(1)
f.write("Adicionei Produtos da database!\n")
driver.find_element_by_class_name("logo").click()
time.sleep(1)
driver.find_element_by_xpath('/html/body/header/nav/div[1]/div[2]/ul/li[6]/a').click()
time.sleep(1)
f.write("Sai da conta do fornecedor!\n")
driver.find_element_by_xpath('/html/body/header/nav/div[1]/div[2]/ul/li[4]/a').click()
time.sleep(1)
#driver.switch_to.frame("textarea_iframe")
driver.find_element_by_id("email").send_keys("test")
driver.find_element_by_id("email").send_keys("test")
time.sleep(1)
driver.find_element_by_id("password").send_keys("testpass")
time.sleep(1)
driver.find_element_by_class_name("btnadmin1").click()
time.sleep(1)
f.write("Dei login como cliente!\n")
driver.find_element_by_xpath('/html/body/header/nav/div[1]/div[2]/ul/li[2]/a').click()
time.sleep(1)
f.write("Entrei na loja!\n")
driver.execute_script("window.scrollTo(0, document.body.scrollHeight);")
time.sleep(1)
driver.find_element_by_class_name("add-to-cart").click()
time.sleep(1)
f.write("Adicionei uma banana ao carrinho!\n")
driver.find_element_by_class_name("add-to-cart").click()
time.sleep(1)
f.write("Adicionei duas bananas ao carrinho!\n")
driver.find_element_by_class_name("add-to-cart").click()
time.sleep(1)
f.write("Adicionei tres bananas ao carrinho!\n")
driver.find_element_by_class_name("checkout").click()
time.sleep(1)
f.write("Dei checkout!\n")
driver.find_element_by_link_text("Place Order").click()
time.sleep(1)
f.write("Completei o pedido!\n")
driver.find_element_by_xpath('/html/body/header/nav/div[1]/div[2]/ul/li[3]/a').click()
time.sleep(1)
driver.find_element_by_class_name("fa-gift").click()
time.sleep(1)
f.write("Cheguei a pagina do historico de encomendas!\n")
driver.find_element_by_class_name("track").click()
f.write("Cheguei a pagina de tracking de encomendas!\n")
time.sleep(1)
f.write("Completei o teste!\n")
print (os.path.abspath("results.txt"))
f.close()
driver.quit();

```

4.4 Estado da implementação

Para esta iteração, foi desenvolvido, em mais detalhe, a parte das encomendas, ou seja, o utilizador podia ver todas as encomendas que realizou e o estado das mesmas (Em preparação, aceite, pronta a ser entregue, etc...). Conseguimos desenvolver tudo o que era esperado para esta iteração e não ficou nada em falta.

Link para o produto final: <https://trcoelhoo.github.io/AS-FARMWEB/>

Apêndice

5 Especificação dos casos de utilização

5.1 Pacote: compra online

5.1.1 CaU 1 Aderir ao sistema

Atores	Cliente
Breve descrição	O cliente cria uma conta ao qual tem de associar o seu email e definir uma password para poder aceder aos serviços da aplicação Web
Trigger	Pressionar o botão para registar
Pré-Condição	PRE-1. Ter endereço de email PRE-2. Não pode existir outra conta criada com esse email
Pós-Condição	POS-1. Nova conta criada
Fluxo Base	<ol style="list-style-type: none">1. Aceder à Web App O cliente entra na aplicação2. Clicar no botão de registar3. Preencher os dados necessários O cliente tem de inserir um email e uma password para poder criar conta
Fluxo Alternativo	
Exceções	Passo3: Email associado a outra conta Não é possível ter duas contas associadas ao mesmo email. O cliente tem de usar outra email e voltar a fazer o passo3 Passo3: Password fraca Para garantir a segurança o cliente deverá usar letras maiúsculas, minúsculas, caracteres especiais e números

5.1.2 CaU 2 Login

Atores	Cliente
Breve descrição	O cliente abre a aplicação Web e efetua o login para entrar no sistema
Trigger	Pressionar o botão Login
Pré-Condição	PRE-1. Possuir conta no Sistema
Pós-Condição	POS-1. Aceder às funcionalidades do sistema
Fluxo Base	<ol style="list-style-type: none">1. Aceder à App O cliente entra no Sistema2. Efetuar o Login O cliente insere o email e a password
Fluxo Alternativo	Passo2: Criar Conta Clicar no botão de registar. Preencher os dados da conta.

	<p>A aplicação retorna à página de login para ser realizado o passo2</p> <p>Passo2: Aceder com a conta do Facebook</p> <p>Caso pretenda, o utilizador pode utilizar a conta do Facebook para entrar na aplicação</p> <p>Passo2: Aceder com a conta da google</p> <p>Caso pretenda, o utilizador pode usar a conta da google para entrar na aplicação</p>
Exceções	<p>Passo2: Email/Password erradas</p> <p>Aparece um pop-up ao utilizador a informar que as credenciais introduzidas estão erradas. O utilizador deve voltar a repetir o passo.</p>

5.1.3 CaU 4 Compra um produto

Atores	Cliente
Breve descrição	O cliente adiciona ao carrinho os produtos que pretende comprar, podendo também fazer a gestão do carrinho, i.e., retirar e adicionar produtos. Posteriormente, efetua a compra dos mesmos
Trigger	Adicionar produtos ao carrinho
Pré-Condição	PRE-1. Estar logado PRE-2. Ter produtos no carrinho virtual
Pós-Condição	POS-1. Não ter produtos no carrinho virtual POS-2. Estar mais uma encomenda na página associada ao acompanhamento das encomendas
Fluxo Base	<ol style="list-style-type: none"> 1. Aceder à App O cliente entra na aplicação Web 2. Procurar os produtos O cliente procura pelo produto que quer comprar 3. Adicionar ao carrinho O cliente seleciona os produtos que pretende comprar 4. Efetuar o pagamento O cliente pressiona o botão para comprar e é redirecionado para uma página onde vai escolher o método de pagamento e posteriormente efetuar a compra.

5.2 Pacote: Adicionar/Remover produtos da loja

5.2.1 CaU 1 Aderir ao Sistema

Atores	Fornecedor
Breve descrição	O Fornecedor cria uma conta ao qual tem de associar o seu email e definir uma password para poder aceder aos serviços da aplicação Web
Trigger	Pressionar o botão para registar
Pré-Condição	PRE-1. Ter endereço de email PRE-2. Não pode existir outra conta criada com esse email
Pós-Condição	POS-1. Nova conta criada
Fluxo Base	<ol style="list-style-type: none">1. Aceder à Web App O Fornecedor entra na aplicação2. Clicar no botão de registar3. Preencher os dados necessários O Fornecedor tem de inserir um email e uma password para poder criar conta
Fluxo Alternativo	
Exceções	<p>Passo3: Email associado a outra conta Não é possível ter duas contas associadas ao mesmo email. O Fornecedor tem de usar outra email e voltar a fazer o passo3</p> <p>Passo3: Password fraca Para garantir a segurança o Fornecedor deverá usar letras maiúsculas, minúsculas, caracteres especiais e números</p>

5.2.2 CaU 2 Login

Atores	Fornecedor
Breve descrição	O Fornecedor abre a aplicação Web e efetua o login para entrar no sistema
Trigger	Pressionar o botão Login
Pré-Condição	PRE-1. Possuir conta no Sistema
Pós-Condição	POS-1. Aceder às funcionalidades do sistema
Fluxo Base	<ol style="list-style-type: none"> 1. Aceder à App O Fornecedor entra no Sistema 2. Efetuar o Login O Fornecedor insere o email e a password
Fluxo Alternativo	<p>Passo2: Criar Conta Clicar no botão de registar. Preencher os dados da conta. A aplicação retorna à página de login para ser realizado o passo2</p> <p>Passo2: Aceder com a conta do Facebook Caso pretenda, o Fornecedor pode utilizar a conta do Facebook para entrar na aplicação</p> <p>Passo2: Aceder com a conta da google Caso pretenda, o Fornecedor pode usar a conta da google para entrar na aplicação</p>
Exceções	<p>Passo2: Email/Password erradas Aparece um pop-up ao utilizador a informar que as credenciais introduzidas estão erradas. O Fornecedor deve voltar a repetir o passo.</p>

5.2.3 CaU 8 Adicionar/Remover produtos

Atores	Fornecedor
Breve descrição	O Fornecedor abre a aplicação Web e escolhe o produto para adicionar/remover
Trigger	Pressionar o botão de adicionar/remover
Pré-Condição	PRE-1. O produto existir se quiser eliminar PRE-1. O produto não existir se quiser adicionar
Pós-Condição	POS-1. Um produto a mais na loja (Adicionar) POS-2. Um produto a menos na loja (Remover)
Fluxo Base	<ol style="list-style-type: none"> 1. Aceder à App O Fornecedor entra no Sistema 2. Efetuar o Login O Fornecedor insere o email e a password 3. Procurar o(s) produto(s) O Fornecedor procura pelo produto que quer remover, ou escolhe o produto que quer adicionar 4. Adicionar/Remover produto(s) O Fornecedor após escolher o produto que quer remover/adicionar introduz as informações do(s) mesmo(s) e executa a ação (remover/adicionar)