

Eduardo Oliveira (98445), Maria Tavares (98447), Lara Matos (95228), Nicole Rakov (96661)

Versão deste relatório: **2022-06-11**, v1.0

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	5
3 Incremento 1	5
3.1 Casos de utilização no Incremento 1	5
3.2 Histórias de utilização selecionadas	6
3.3 Estratégia e estado da implementação	8

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.

O primeiro incremento, desenvolvido na Iteração 3, foca a validação da arquitetura proposta. Foram considerados sobretudo as funcionalidades relacionadas com a compra de combustível através da aplicação e a consequente criação de códigos QR para serem utilizadas nas bombas de abastecimento.

O segundo incremento, considerado na Iteração 4, evolui o trabalho anterior e foca em especial a troca de combustível que é possível ser feita na aplicação, substituindo a quantidade, em litros, de tipo de combustível, por outra quantidade ao preço do novo combustível.

1.2 Controlo de versões

Quando?	Responsável	Alterações significativas
01/06/2022	Eduardo Oliveira	Backlog do projeto
01/06/2022	Eduarda Tavares	2.2 - Requisitos com impacto na arquitetura
01/06/2022	Lara Matos	1.1 – Sumário executivo
01/06/2022	Nicole Rakov	2.3 - Decisões e justificação
02/06/2022	Eduardo Oliveira	2.1 - Objetivos Gerais
06/06/2022	Eduardo Oliveira	3.2 - Histórias de utilização usadas
06/06/2022	Eduarda Tavares	3.1 - Casos de utilização no incremento 1
07/06/2022	Lara Matos	Criação da estrutura do PowerPoint da apresentação
07/06/2022	Nicole Rakov	2.4 - Arquitetura do software
09/06/2022	Eduarda Tavares	2.5 - Arquitetura física de instalação
09/06/2022	Eduardo Oliveira	3.3 - Estratégia e estado de implementação
11/06/2022	Lara Matos	Revisão geral do relatório

1.3 Referências e recursos suplementares

- <https://www.idalko.com/jira-agile/> JIRA
- Firebase para a base de dados da aplicação.
- FlutterFlow para a criação da aplicação:
 - <https://app.flutterflow.io/share/a-s-fast-and-fuel-2-w6ofx5>

2 Arquitetura do sistema

2.1 Objetivos gerais

- A aplicação deverá integrar uma plataforma de pagamentos eletrónicos de forma a suportar as transações efetuadas pelos clientes para a Fast&Fuel. As transações deverão ser efetuadas em menos de 1 minuto.
- A aplicação deverá integrar a tecnologia de criação de Qr codes para uma futura leitura no posto de abastecimento. Para isso o sistema de leitura também terá de conseguir reconhecer e atualizar a base de dados do sistema.
- A aplicação deverá raramente falhar em termos de performance/bugs, uma vez que uma pequena falha pode custar vários milhares de euros, neste tipo negócio. Com isto, também se está a incluir as devidas precauções aos picos de utilização aos fins-de-semana, promoções ou anúncios de subida de preço futura.

2.2 Requisitos com impacto na arquitetura

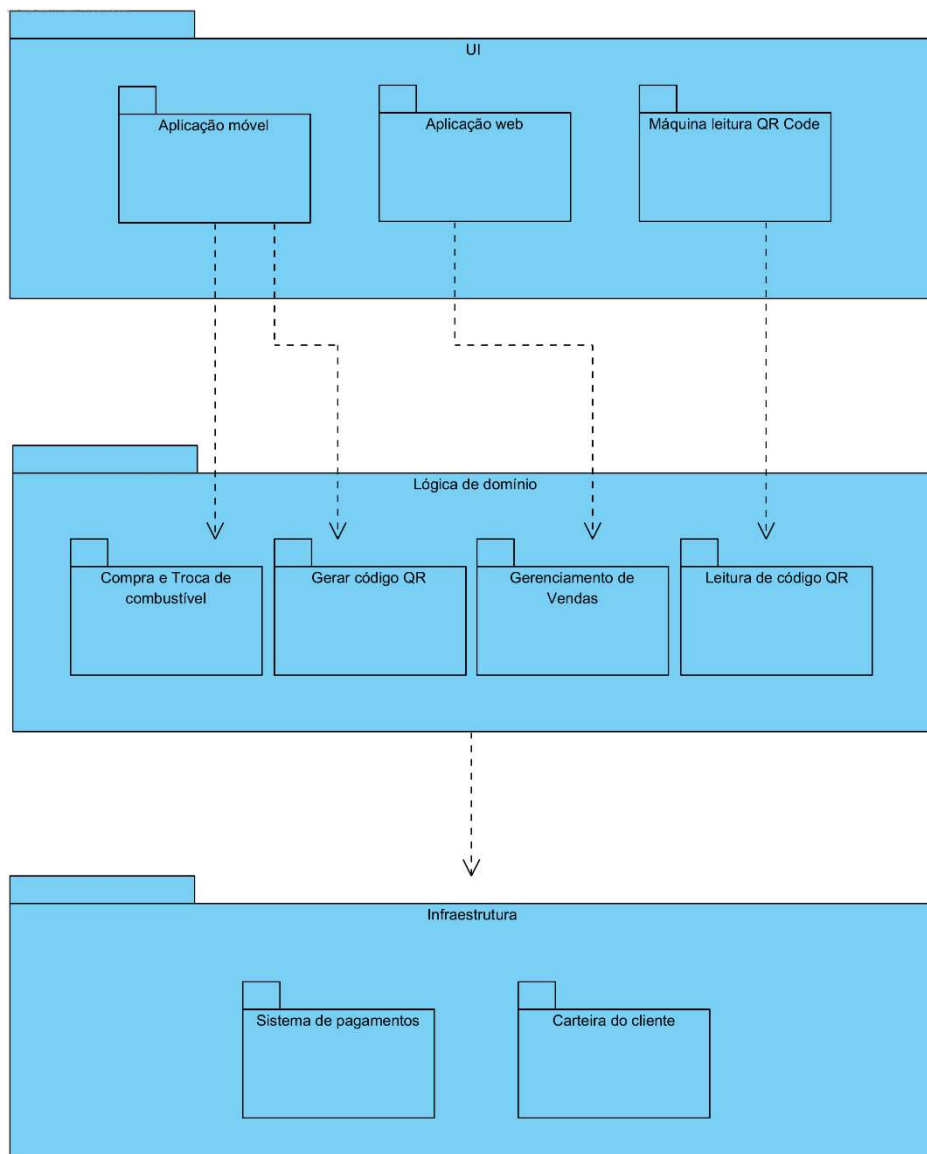
Requisitos	Descrição
RU-1	Garantir a legibilidade do texto, através da fonte e cores legíveis e agradáveis - permitindo a leitura do texto a 1m do ecrã.
RU-2	A interface da aplicação deve ser <i>user-friendly</i> , sendo assim acessível para qualquer usuário - desde jovens adultos a idosos.
RD-1	Garantir que todos os pagamentos são processados num curto espaço de tempo: em menos de 1 minuto.
RS-1	A informação pessoal dos clientes deverá ser guardada de forma segura, cifrada.
RS-2	Garantir que são utilizadas palavras-passes seguras.
RI-1	Integrar sistemas de pagamento seguros associados a serviços externos.
RI-2,3	Integrar uma base de dados atualizada e segura.
RI-4	Integrar a tecnologia responsável pela leitura do QR code.

2.3 Decisões e justificação

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

- Frontend implementado em Dart com a framework Flutter. Gerado pela aplicação web “FlutterFlow” e aperfeiçoado manualmente. O uso desta plataforma foi escolhido uma vez que, através da adesão ao pacote “Pro” do FlutterFlow (disponível durante os primeiros 14 dias), é possível fazer o download do código para o utilizar fora da plataforma, criar a Web API que é necessária e construir a APK.
- Integração da API do Google Maps para mapas, uma vez que é fácil utilização ao usuário.
- Integração com MB Way e PayPal para pagamentos. Uma vez que garante diversidade em formas de pagamento e segurança ao cliente.
- Uso da plataforma Jira para implementação Agile. A fim de garantir organização e rapidez em todas as etapas do desenvolvimento.
- Gestão de base de dados com FireBase. Plataforma segura e com integração direta com FlutterFlow.

2.4 Arquitetura do software



A user interface (UI) do sistema baseia-se em duas aplicações - a aplicação móvel e a web. Ambas com finalidades distintas. Além disso, há um dispositivo IOT, leitor de códigos QR.

Em relação a lógica de domínio, a aplicação móvel possibilita a troca e venda de combustível e gera um código QR com a finalização de uma transação. Uma vez gerado, o dispositivo IoT no posto de combustível realiza a leitura do mesmo. A aplicação web, permitiria o gerenciamento de vendas da aplicação por parte das gasolinhas.

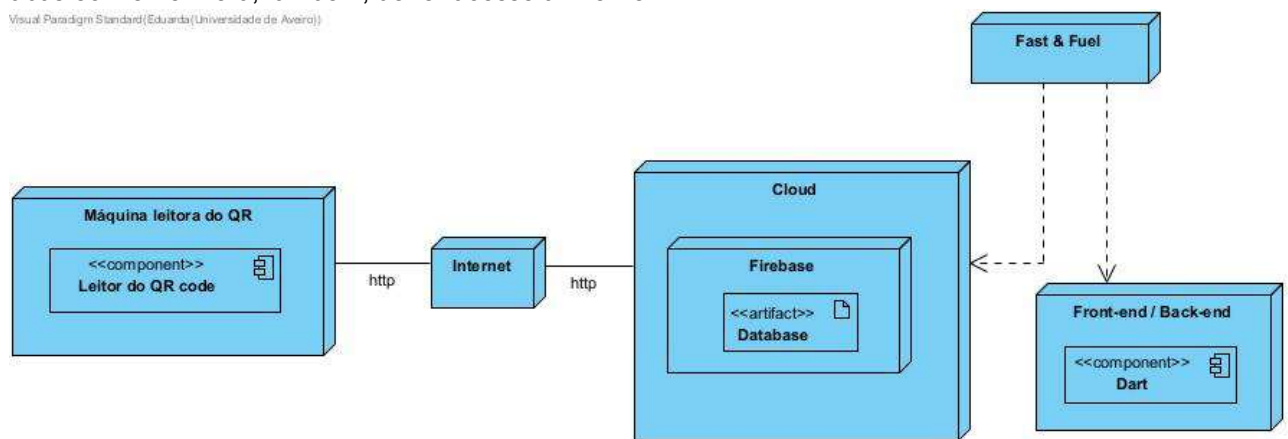
A infraestrutura, diz-se respeito a carteira do cliente, a qual dispõe do histórico de compras, trocas e saldo de combustível a ser resgatado ou comprado.

2.5 Arquitetura física de instalação

Quanto à arquitetura física de instalação, a aplicação móvel dependerá de uma Cloud onde estará inserida a base de dados, integrada com Firebase. Quanto ao front-end e back-end do aplicativo, serão ambos desenvolvidos com código Dart.

A máquina com tecnologia capaz de ler o QR code – que estará disponível nos vários postos de abastecimento - terá, também, de ter acesso à internet.

Visual Paradigm Standard (Eduardo (Universidade de Aveiro))



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 os seguintes CaU : caso de uso relativo ao processo da compra de combustível através da aplicação; caso de uso relativo ao levantamento de combustível e, finalmente, o caso de uso relativo à - CaU1 e CaU2 (como consta do documento referente à iteração 2), respetivamente. Para além destes, acrescentámos outros dois casos de uso: a leitura do relatório e a visualização da palestra.

Para dar início ao primeiro caso de uso (CaU1), o utilizador deverá primeiramente selecionar a opção da compra de combustível - que pode encontrar tanto no menu principal como na taskbar da aplicação. De seguida, o cliente irá escolher o tipo de combustível que deseja comprar e indicar a quantidade desejada.

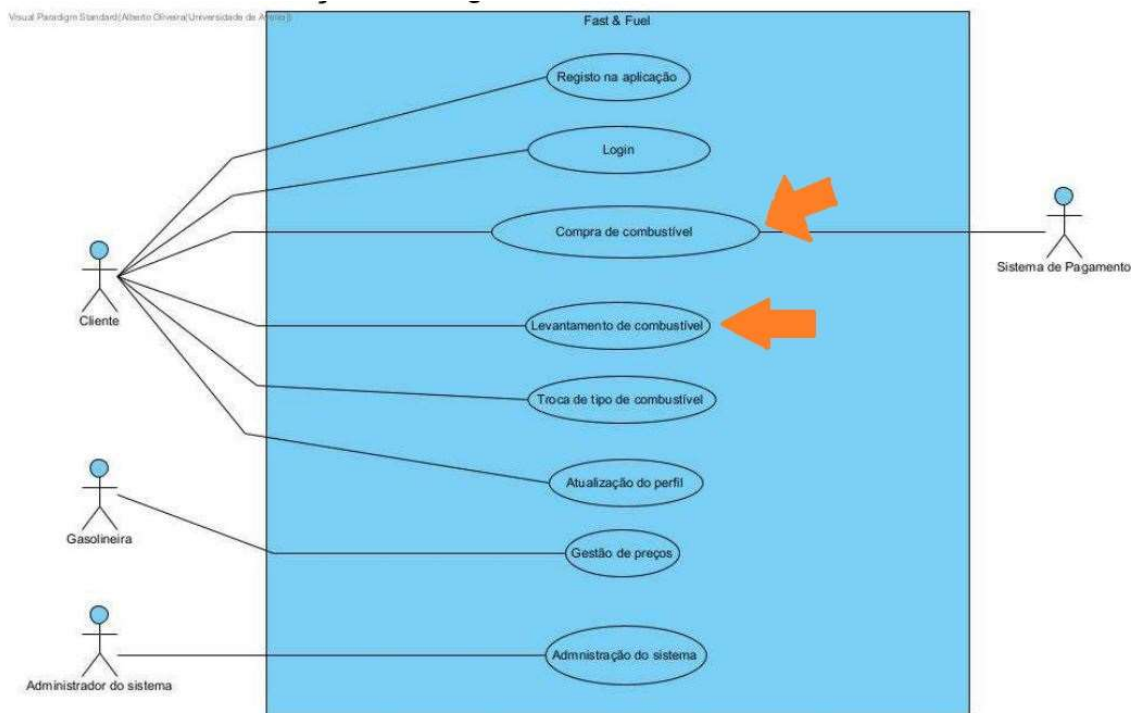
Após ser processado o pagamento, o usuário deverá encontrar os litros de combustível que comprou já disponíveis na sua carteira.

No que toca ao segundo caso de uso (CaU2), o utilizador deverá primeiramente aceder à área de emissão de códigos QR. Após lá estar e ter escolhido o tipo de combustível que pretende levantar,

assim como a quantidade, o sistema irá gerar um código QR para que o cliente possa utilizar no abastecimento de combustível, num posto à sua escolha.

Quanto ao terceiro CaU – a leitura do relatório - consta de um caso de uso importante visto que se pode tornar num fator decisivo no que toca à decisão de investir, ou não, na aplicação.

Por fim, o último CaU consiste na visualização de uma palestra associada à Fast&Fuel – com o objetivo de dar a entender no que se baseia este sistema: quais são as suas vantagens, para que serve e como é utilizado.



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

3.2 Histórias de utilização seleccionadas

As histórias (*user stories*) incluídas nesta interação fazem parte do *backlog* do projeto, acessíveis em <https://fastfuel.atlassian.net/jira/software/projects/FAF/boards/1/backlog>

Histórias incluídas nesta interação:

História/use case slice	Critérios de aceitação
O Carlos compra gasolina Sendo o Carlos um utilizador da Fast&Fuel, Quero comprar 50 Litros de gasolina 95 De modo a ficarem guardados para quando quiser ir abastecer.	Cenário 1: Compra com sucesso Dado que estou no menu principal Quando clico na opção de comprar gasolina E seleciono o tipo de gasolina E seleciono a gasolineeira E seleciono o tipo de pagamento E concluo o pagamento Então uma janela com o QR code a informar que compra foi feita com sucesso aparece

	<p>Cenário 2: Compra falhada Dado que estou no menu principal Quando clico na opção de comprar gasolina E seleciono o tipo de gasolina E seleciono a gasolinera E seleciono o tipo de pagamento E rejeito o pagamento Então uma janela que compra foi falhada aparece</p>
<p>A Joana levanta combustível Sendo a Joana uma utilizadora com gasolina na Fast&Fuel Quero abastecer 20 Litros de gasolina De modo a conseguir ter combustível no carro para se deslocar</p>	<p>Cenário 1: Abastecimento efetuado com sucesso Dado que estou no menu principal Quando clico na opção de emitir o QR code E seleciono a quantidade desejada E tipo de combustível desejado Então o QR code gerado aparecerá E dirigi-me para o posto de abastecimento Onde faço scan do QR code que é aceite e abasteco</p> <p>Cenário 2: Abastecimento sem sucesso <i>(a ser implementado apenas no incremento 2)</i> Dado que tenho a fotografia de um QR code usado E me desloco ao posto de abastecimento E faço o scan do QR code Então uma mensagem a dizer que o QR code já foi usado aparecerá</p>
<p>O Paulo lê um relatório sobre a análise e construção do projeto Fast&Fuel Sendo o Paulo um investidor, Quero ler sobre a arquitetura do projeto da Fast&Fuel De modo a saber se este sistema é um bom investimento analisando as limitações e as suas potencialidades</p>	<p>Cénario 1: Leitura com sucesso Dado que tenho acesso ao relatório E decido ler Então consigo saber se devo investir ou não na Fast&Fuel analisando a arquitetura</p>
<p>A Maria assistiu a uma palestra Sendo a Maria uma acionista de uma gasolinera, Quero assistir a uma palestra De modo a saber se a Fast&Fuel é adequada para a gasolinera onde tenho as ações e um grande poder de influência</p>	<p>Cenário 1: Palestra atrativa Dado que fui ouvir a palestra E estive atenta Então pode aconselhar toda a gente a investir na Fast&Fuel devido à boa arquitetura e implementação que fizeram</p>

3.3 Estratégia e estado da implementação

Nesta iteração em termos de implementação do frontend usámos o FlutterFlow, uma vez que já tínhamos usado no protótipo fez com que o trabalho fosse facilitado. Para este usamos a linguagem Dart.

Quanto ao backend usamos o FireBase para termos uma Base de Dados, no entanto não conseguimos concluir, de forma eficaz, a realização desta etapa pelo que continua ausente uma ligação entre as páginas e os dados guardados na nossa base.

Uma outra etapa que estávamos à espera de realizar era a criação do código QR individual, o que foi dificultado pelo problema referido anteriormente.

De forma a reunir toda a informação para a realização deste incremento foi usado o Jira, uma plataforma que nos permitiu uma gestão ágil de todo o incremento e uma organização de tarefas mais eficaz.