

## Relatório de Projeto

### Arquitetura de Sistemas de Informação

(2020/2021)

Grupo Nº	Curso / Nome Projeto				
Beta	BOOZ				
Composição do Grupo					
Número / Nome	Esforço (Horas)				
	Pesqui. Web	Reuniões	Elabor. Diag.	Elabor. Relató.	Total
20200538 – Francisco Barros Cabreiro	2:00	19:10	8:00	3:30	32h40
20200758 – Gisela Marques	3:00	27:00	13:00	7:30	50h30
20200573 – Gonçalo Carvalho	3:50	25:30	15:30	6:45	51h35
20200154 – Madalena Barreiros Cardoso	2:45	32:30	19:00	20:30	75h45
20200908 – Pedro Domingues da Cunha	2:30	26:30	10:30	6:20	45h50
20200003 – Ricardo Fontes	3:05	25:00	12:30	7:30	48h05

# Índice

<b>SUMÁRIO EXECUTIVO</b>	<b>3</b>
<b>INTRODUÇÃO</b>	<b>3</b>
<b>DIAGRAMA DE CONTEXTO DO SISTEMA BOOZ</b>	<b>4</b>
<b>PROPOSTA DE PROJECTO - VISÃO DE NEGÓCIO</b>	<b>7</b>
<b>LEAN CANVAS</b>	<b>7</b>
<b>CLASSIFICAÇÃO DOS INDICADORES</b>	<b>10</b>
<b>APRESENTAÇÃO DA ARQUITETURA DO SISTEMA</b>	<b>12</b>
<b>DIAGRAMA DE CASOS DE USO DO SISTEMA BOOZ</b>	<b>15</b>
<b>CASOS DE USO – STANDARD</b>	<b>17</b>
<i>UC1 – Registar</i>	21
<i>UC2 – Introduzir dados pessoais e método de pagamento</i>	21
<i>UC3 – Autenticar no sistema</i>	21
UC4 – Criar carrinho	21
UC5 – Consultar catálogo	21
UC6 – Checkout carrinho	21
UC7 – Verificar stock	22
UC8 – Pagar encomenda	22
UC9 – Confirmar pagamento efetuado	22
UC10 – Indicar ponto de entrega	22

UC11 – Consultar encomenda	22
UC12 – Confirmar entrega da encomenda	22
UC13 – Avaliar entrega	22
UC14 – Consultar lista de entregas a fazer	22
UC15 – Alterar estado da entrega	23
UC16 – Autorizar GPS	23
UC17 – Consultar lista de pedidos	23
UC18 – Emissão de pedido de entrega	23
UC19 – Gestão de informação do sistema	23
UC20 – Adicionar, remover ou editar produtos	23
UC21 – Configurar as funcionalidades do sistema	19
<b>CASO DE USO COMPLEXO</b>	<b>20</b>
<i>UC4 – Registar</i>	20
<i>UC4 – Criar carrinho</i>	
<i>UC6 – Finalizar encomenda do carrinho</i>	20
<b>MODELO DE DOMÍNIO DO SISTEMA ENCOMENDA E ENTREGA DE BEBIDAS ALCÓLICAS(BOOZ)</b>	
<b>CONCLUSÕES E TRABALHO FUTURO</b>	<b>25</b>
<b>REFERÊNCIAS</b>	<b>27</b>
<b>BIOGRAFIA DOS AUTORES</b>	<b>27</b>
<b>ANEXO A: TABELA DE INDICADORES</b>	<b>29</b>

**ANEXO B: ESPECIFICAÇÃO DAS INTERFACES API****ANEXO C: LEVANTAMENTO DE REQUISITOS FUNCIONAIS****ANEXO D: FUNCIONAMENTO DA UNIDADE CURRICULAR**

34

## Sumário Executivo

Este relatório é uma exposição da nossa ideia enquanto grupo, Booz. Trata-se de uma aplicação para smartphone que poderá ser descarregada por qualquer pessoa residente em Portugal começando primeiramente por fazer entregas no centro da Grande Lisboa. O Booz oferece ao utilizador, uma forma rápida de obter as bebidas e snacks que pretende. Com a nossa aplicação, o cliente pode solicitar a sua encomenda a qualquer hora do dia. É algo inovador pois nas plataformas como Uber Eats ou a Garrafeira Nacional não existe a possibilidade de fazer encomendas com horário 24h.

**Palavras chave:** bebidas, encomenda, cliente, aplicação.

---

## Acrónimos

**API**     *Application Programming Interface*

**GPS**     *Global Positioning System*

**GUI**     *Graphical User Interface*

**Sol**     *System of Interest*

## 1 Introdução

Este projeto foi realizado no âmbito da unidade curricular Arquitetura de Sistemas de Informação, onde nos foi proposto a realização de um projeto. O nosso grupo (Beta) decidiu desenvolver uma aplicação de encomenda e entrega de bebidas alcoólicas e snacks durante 24h. Este projeto foca-se em resolver os problemas da Uber Eats e da Garrafeira Nacional ao permitir que o cliente usufrua um serviço durante 24h, podendo fazer o seu pedido a qualquer hora do dia.

O cliente interage com o nosso sistema, autenticando-se. De seguida insere os seus dados e consulta o catálogo de bebidas e snacks. Depois de selecionar os produtos pretendidos para o carrinho, dirige-se para finalizar a encomenda, onde indica o local de entrega e efetua o pagamento. É redirecionado para a entidade de pagamento (ator do nosso sistema) e após o pagamento ser confirmado, a encomenda é registada. O fornecedor consulta a lista de pedidos, prepara o pedido e solicita um estafeta, emitindo um pedido para a lista de entregas, a qual o estafeta tem acesso e permite a aceitação da entrega.

## 2 Diagrama de Contexto do Sistema BOOZ

O diagrama de contexto é um diagrama de fluxo de informação de alto nível que representa todo o sistema (black box). Ele identifica os fluxos de informações entre o system of Interest (Sol) e as entidades externas. Como intervenientes do nosso Diagrama de contexto, temos o **Cliente**, o **Estafeta**, o **Administrador de Sistemas**, o **Fornecedor** e a **Entidade Pagamento** que são os nossos stakeholders e todos eles realizam a sua interação como atores. Por último temos os nossos enabling systems que são a **Comunicação Wireless** e o **Dispositivo Móvel**.

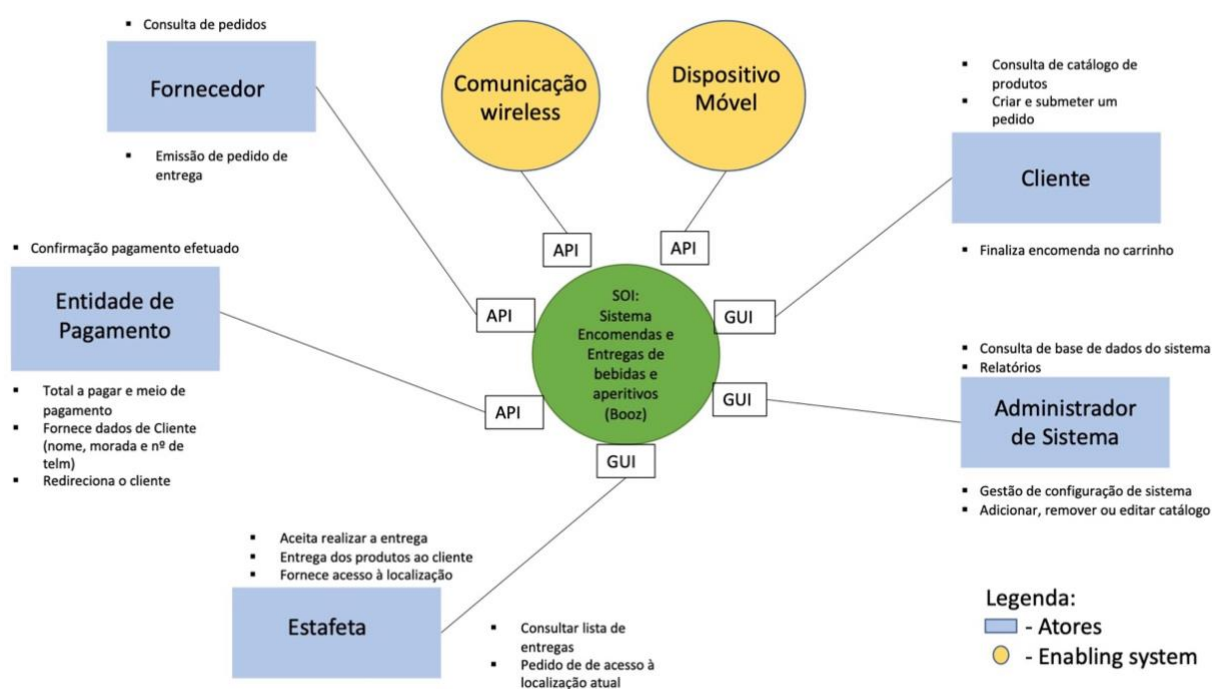


Figura 1. Diagrama de Contexto do sistema BOOZ

ATOR	DESCRIÇÃO
Cliente	O Cliente comunica com o nosso sistema através de um GUI (Graphical User Interface), é o nosso ator mais importante porque é para ele que todo o sistema funciona.
Administrador de Sistema	O Administrador de sistema comunica com o nosso sistema através de um GUI (Graphical User Interface) é responsável por instalar, suportar, manter servidores e outros sistemas.
Estafeta	O Estafeta comunica com o nosso sistema através de um GUI (Graphical User Interface) é responsável por recolher o pedido no fornecedor e entregar ao cliente.
Entidade de Pagamento	A Entidade de pagamento comunica com o nosso sistema através de uma API (Application Programing Interface), é responsável por assegurar a confirmação de pagamento, após efetuar o preenchimento de todas as informações necessárias para a finalização da encomenda, o cliente é direcionado para a entidade de pagamento para que a encomenda possa ser finalizada.
Fornecedor	O Fornecedor comunica com o nosso sistema através de uma API (Application Programing Interface) é responsável por consultar os pedidos e emitir os pedidos de entrega.
Comunicação Wireless	A comunicação wireless é um dos enabling systems, estando out of the scope, comunica com o nosso sistema através de uma API (Application Programing Interface), é necessária para que o sistema funcione pois dependemos de uma ligação à internet.
Dispositivo Móvel	O dispositivo móvel é um dos enabling systems, estando out of the scope comunica com o nosso sistema através de uma API (Application Programing Interface) é responsável por dar ao cliente todo o acesso à aplicação e ao serviço.

### 3 Proposta de Projeto - Visão de Negócio

Ser uma aplicação utilizada primeiramente na zona do centro da Grande Lisboa, que facilita a acessibilidade da população, maioritariamente os jovens adultos que procuram adquirir bebidas alcoólicas e snacks a qualquer hora do dia.

## a. Lean Canvas

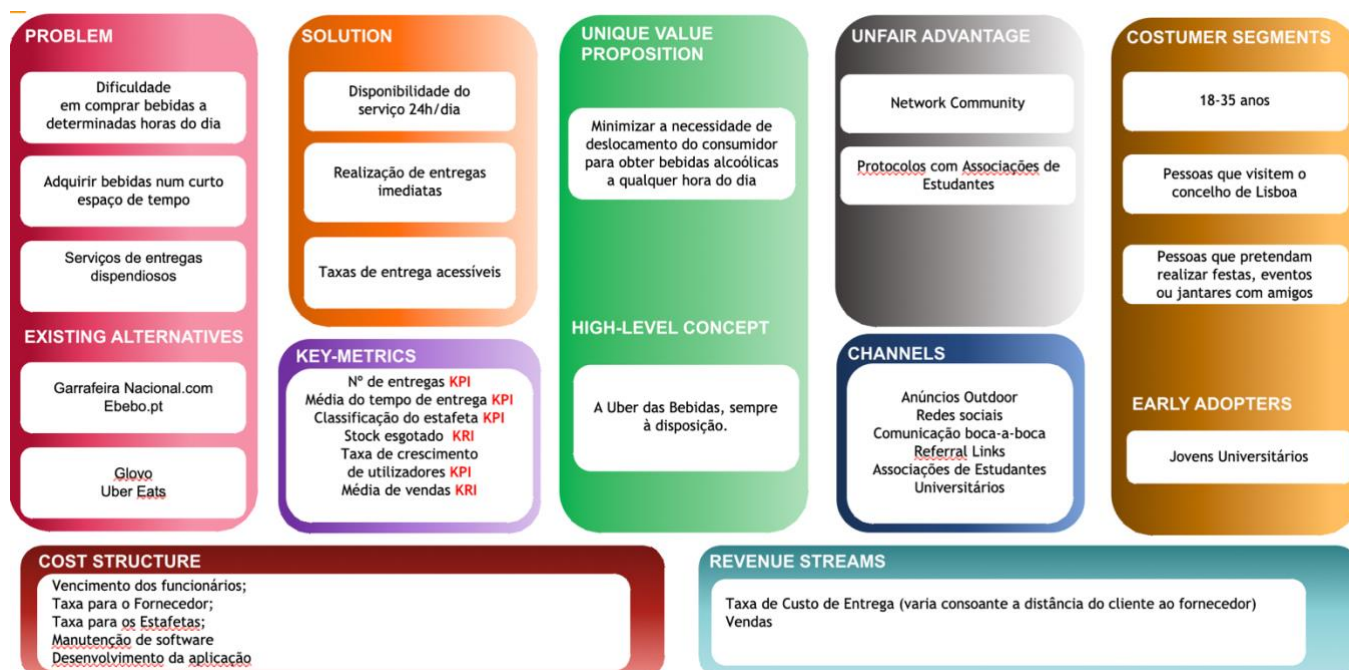


Figura 2. Caracterização da proposta BOOZ.

O Lean Canvas é um modelo de plano de negócios, adaptado do Business Model Canvas, que substitui planos de negócio elaborados por um modelo de negócio de uma só página direcionado em solucionar os problemas dos seus clientes. Ajuda as start-ups a desconstruírem as suas ideias de uma maneira mais eficaz. O Lean Canvas está dividido em 2 partes, Produto e Mercado, sendo estas divididas em 9 blocos ligeiramente modificados para atingirem as necessidades, objetivos e requisitos que as start-ups necessitam. Descrevendo o que cada bloco faz, temos:

- Problem/Existing Alternatives** – Os customer segments (clientes) que nos vamos focar têm um conjunto de problemas que têm de ser resolvidos perante a proposta feita. Têm de se encontrar até 3 problemas onde se consigam arranjar uma solução para cada problema, e mostrar empresas que já possam ter resolvido alguns desses problemas (existing alternatives).



- **Customer Segments/Early Adopters** – Os problemas e os customer segments estão interligados, pois sem customer segments não temos problemas e vice-versa. Eles têm que ser específicos e têm que mostrar os primeiros clientes (early adopters) que vão usar o produto e mostrar feedback do mesmo.
- **Unique Value Proposition/High-level Concept** – Uma value proposition é uma “promessa” do que a nossa proposta vai atingir e dar aos nossos clientes, ela mostra o que o nosso produto tem de único e as diferenças que tem das alternativas já existentes. O high-level Concept é uma frase curta e fácil de entender o nosso produto.
- **Solution** - São as soluções aos problemas que os clientes têm e explicam as experiências que vão ter através do nosso produto.
- **Channels** – São “canais de comunicação” que nos vão ajudar a atrair/chegar aos nossos clientes. Tentar encontrar channels que nos deem acesso aos clientes e compreensão dos mesmos.
- **Revenue Streams** – Como nós avaliamos a nossa empresa e as maneiras a partir das quais o nosso produto pode gerar receitas com cada customer segment.
- **Cost Structure** – Aqui são listados todos os custos que se fez para pôr o produto no mercado como por exemplo, salários, infraestruturas, manutenção, pesquisa de mercado, etc.
- **Key Metrics** – São usadas para monitorizar a performance e o progresso do projeto da start-up.
- **Unfair Advantage** – Uma coisa especial sobre o produto de que as outras empresas não possam ser capazes de copiar ou obter com facilidade.

Através destas descrições nós conseguimos fazer o nosso Lean Canvas de uma maneira mais rápida, eficaz e SMART. Como Top 3 problemas dos nossos clientes temos, dificuldade em comprar bebidas a determinadas horas do dia, adquirir bebidas num curto espaço de tempo e serviços de entrega dispendiosos, tendo arranjado 3 soluções para estes problemas, sendo elas, disponibilidade do serviço 24h/dia, realização de entregas imediatas e taxas de entrega acessíveis. Os nossos customer segments serão pessoas com idade legal para beber, 18-35 anos, pessoas que visitem o concelho de Lisboa e pessoas que pretendam realizar festas, eventos ou jantares com amigos, sendo os nossos early adopters jovens universitários. Como Unique Value Proposition comprometemo-nos a minimizar a necessidade de deslocamento do consumidor para obter bebidas alcoólicas a qualquer hora do dia e como High-level Concept acreditamos ser a Uber das bebidas, sempre à disposição. Como Unfair Advantage temos protocolos com Associações de Estudantes e Network Community. Os nossos canais são anúncios outdoor, redes sociais, comunicação boca-a-boca, Referral Links e associações de estudantes universitários. As nossas Key Metrics, a partir das quais vamos conseguir medir e monitorizar a nossa performance são, número de entregas, média do tempo de entrega, classificação do estafeta, stock esgotado, taxa de crescimento de utilizadores e média de vendas.

A nossa Cost Structure são o vencimento dos funcionários, taxa para o fornecedor, taxa para os estafetas, manutenção de software e o desenvolvimento da aplicação.

Nos Revenue Streams, temos a taxa de custo de entrega, que varia consoante a distância do cliente ao fornecedor e as vendas.

## b. Classificação dos Indicadores

KPI (Key Performance Indicator), que são o tipo de indicadores que estamos a utilizar, é uma métrica que empreendedores utilizam para avaliar a performance dos seus projetos. Fornece uma medida objetiva para ajudar as empresas a estabelecerem metas e objetivos para os seus projetos. Ajuda a medir o progresso e a distância a que se está de um determinado objetivo. Percebendo isto, os indicadores mostrados na tabela abaixo estão de acordo com a nossa necessidade de saber e poder monitorizar o desempenho da nossa aplicação e averiguar as modificações que precisam de ser implementadas.

#	Nome Indicador	Metadados de Caracterização
1	Nº de Entregas	<p><b>Descrição:</b> Esta KPI tem como objetivo averiguar quantas encomendas são entregues face ao total de encomendas realizadas. Esta informação tem por base os dados retirados da nossa aplicação.</p> <p><b>Categoria/Tipo:</b> Lagging/On-target</p> <p><b>Métrica:</b> Soma de todas as entregas feitas</p> <p><b>Threshold:</b> 98% e desvio aceitável de 1%.</p> <p><b>Frequência:</b> Semanalmente</p>
2	Média do tempo de entrega	<p><b>Descrição:</b> Esta KPI tem como objetivo calcular a média do tempo que o estafeta demora a efetuar a entrega, tanto a recolha no fornecedor como a entrega ao cliente.</p> <p><b>Categoria/Tipo:</b> Lagging/On-target</p> <p><b>Métrica:</b> Tempo total sobre a distância percorrida</p> <p><b>Threshold:</b> 1 minuto e 30 segundos /Km e um desvio aceitável de 1 minuto.</p> <p><b>Frequência:</b> Diariamente</p>
3	Classificação do Estafeta	<p><b>Descrição:</b> Esta KPI pretende mostrar o nível da classificação que o cliente atribuiu ao estafeta e ao serviço em geral. O cliente avalia numa escala de 1 a 5, sendo 1 muito mau e 5 excelente.</p> <p><b>Categoria/Tipo:</b> Lagging/On-target</p> <p><b>Métrica:</b> Classificações totais a dividir pelo número de classificações</p> <p><b>Threshold:</b> O objetivo seria obter uma classificação superior a 4,8. Sendo o desvio aceitável de 0,20.</p> <p><b>Frequência:</b> Mensalmente</p>

4	Stock Esgotado	<p><b>Descrição:</b> Esta KRI tem como objetivo monitorizar o número de produtos por dia armazenados.</p> <p><b>Categoria/Tipo:</b> Lagging / On-target</p> <p><b>Métrica:</b> Número de produtos no início do dia – número de produtos vendidos</p> <p><b>Threshold:</b> Quantidade de produto tem de ser superior a 10 unidades.</p> <p><b>Frequência:</b> Diariamente</p>
5	Taxa de crescimento de utilizadores	<p><b>Descrição:</b> Esta KPI tem como objetivo a contabilização do crescimento de utilizadores do serviço em percentagem.</p> <p><b>Categoria/Tipo:</b> Lagging / Benefício</p> <p><b>Métrica:</b> Soma de todos os novos utilizadores do mês / total de utilizadores * 100</p> <p><b>Threshold:</b> Cerca de 100 novos utilizadores por mês, não tem mínimo</p> <p><b>Frequência:</b> Mensalmente</p>
6	Média de vendas	<p><b>Descrição:</b> Esta KRI tem como objetivo elaborar o balanço geral de vendas executadas através da aplicação BOOZ e verificar se há um aumento ou diminuição significativa. Esta informação pode ser recolhida na base de dados da nossa aplicação.</p> <p><b>Categoria/Tipo:</b> Lagging / Benefício</p> <p><b>Métrica:</b> Vendas totais a dividir pelo número de encomendas</p> <p><b>Threshold:</b> O valor mínimo será de 1000 vendas e não tem valor máximo.</p> <p><b>Frequência:</b> Mensalmente</p>

## 4 Apresentação da Arquitetura do Sistema

Componentes de Software Grupo de Utilizadores	Autenticação (AAA)	Módulo Gestão Encomendas	Módulo Avaliação de Entrega	Módulo Configuração conta/user	Módulo de Gestão de Entregas	Módulo Consulta de dados	Módulo Gestão de Users	Módulo Configuração do Sistema	Módulo Localização Entrega
Cliente	CRU	CRD	U	RU	R	R			R
Estafeta	CRU			RU	RU	R			UR
Administrador de Sistemas	CRUD	R	R	RU		RU	CRUD	CRUD	
Fornecedor	CRU	RU			CRU				
Entidade de Pagamento		RU							

Figura 3. Matriz de CRUD da proposta BOOZ

A Matriz de CRUD é construída a partir do cruzamento das funcionalidades que são apresentadas no diagrama de blocos, com os atores que interagem com o nosso Sol (System of Interest). O alinhamento desse cruzamento permite-nos saber que autorizações são dadas a cada ator, utilizando o método CRUD (*Create; Read; Update; Delete*).

Quanto à Matriz de CRUD da equipa Beta, incluímos todas as entidades, tais como as respetivas funcionalidades(módulos) referentes ao nosso Sol, de modo a definir as suas autorizações perante o nosso sistema. O ator cliente vê-se capaz de fazer *Create, Read e Update* no que toca ao módulo de **autenticação AAA**, pois consegue criar uma conta através da nossa aplicação BOOZ, ler o que lhe for apresentado na interface e, consequentemente o sistema, atualiza o estado da sua conta de não registado para autenticado, tal como acontece com o estafeta, que passa a pertencer ao nosso grupo de estafetas. O administrador de sistemas também participa neste módulo com as quatro autorizações que compõem o CRUD, sendo elas o *Create, Read, Update e Delete*, sendo que é este que configura as funcionalidades deste módulo.

Quanto ao **módulo de gestão de encomendas**, ao cliente é-lhe dada a possibilidade de aceder a um catálogo BOOZ que, poderá efetuar as suas encomendas, sendo este um portador de *Create*,

*Read e Delete*. O administrador de sistemas apenas tem acesso à leitura (*Read*) dos dados. O fornecedor e a Entidade de pagamento conseguem ler e atualizar os dados (*Read e Update*).

No que toca ao módulo seguinte, o de **Avaliação de Entrega**, onde o cliente faz *Update*, classificando o serviço prestado pelo estafeta. O Administrador de sistemas tem acesso a cada classificação efetuada pelos clientes (*Read*).

O módulo de **Configuração da conta/ user**, envolve apenas três atores, o cliente, o estafeta e o administrador de sistemas. Estes podem apenas fazer *Read e Update*, dando a possibilidade de fazer as atualizações necessárias referentes à sua conta, como a mudança de *password*, do e-mail.

O módulo de **gestão de entregas** refere-se ao processo desde que o cliente finaliza o pedido até à confirmação da entrega por parte do mesmo. Aqui, é disponibilizada ao estafeta uma lista de entregas a realizar, onde o mesmo seleciona uma entrega a realizar. O estafeta faz o *Update*, iniciando o processo de entrega, faz *Read* de forma a visualizar as informações disponibilizadas pela lista de entregas, seguindo-se o *Update* do estado em que se encontra a encomenda por parte do estafeta. O fornecedor adiciona à lista de entregas os pedidos a realizar tendo acesso às funcionalidades de *Create, Read e Update*.

O módulo de **Consulta de dados**, permite tanto o cliente como ao estafeta visualizarem os dados (*Read*), sendo que não têm qualquer poder para alterar, criar ou apagar dados já registados. O administrador de sistema poderá consultar e editar os dados (*Read e Update*). Os módulos **Gestão de Users e Configuração do sistema**, são verificados pelo administrador de sistema sendo que em ambos os módulos realiza todas as operações CRUD, ou seja, faz a gestão das contas dos users, dando-lhe todas e quaisquer permissões sobre as contas para criar, alterar e apagar dados; tal como a configuração de sistema, onde o administrador de sistema monitoriza a aplicação e recebe relatórios de terceiros, de forma a melhorar algum aspeto ou tratar de algum problema relacionado com a aplicação. Por último, temos o módulo **Localização Entrega** no qual o estafeta fornece a sua localização ao cliente podendo o mesmo visualizá-la. Ao longo do percurso o módulo atualiza a localização do estafeta.

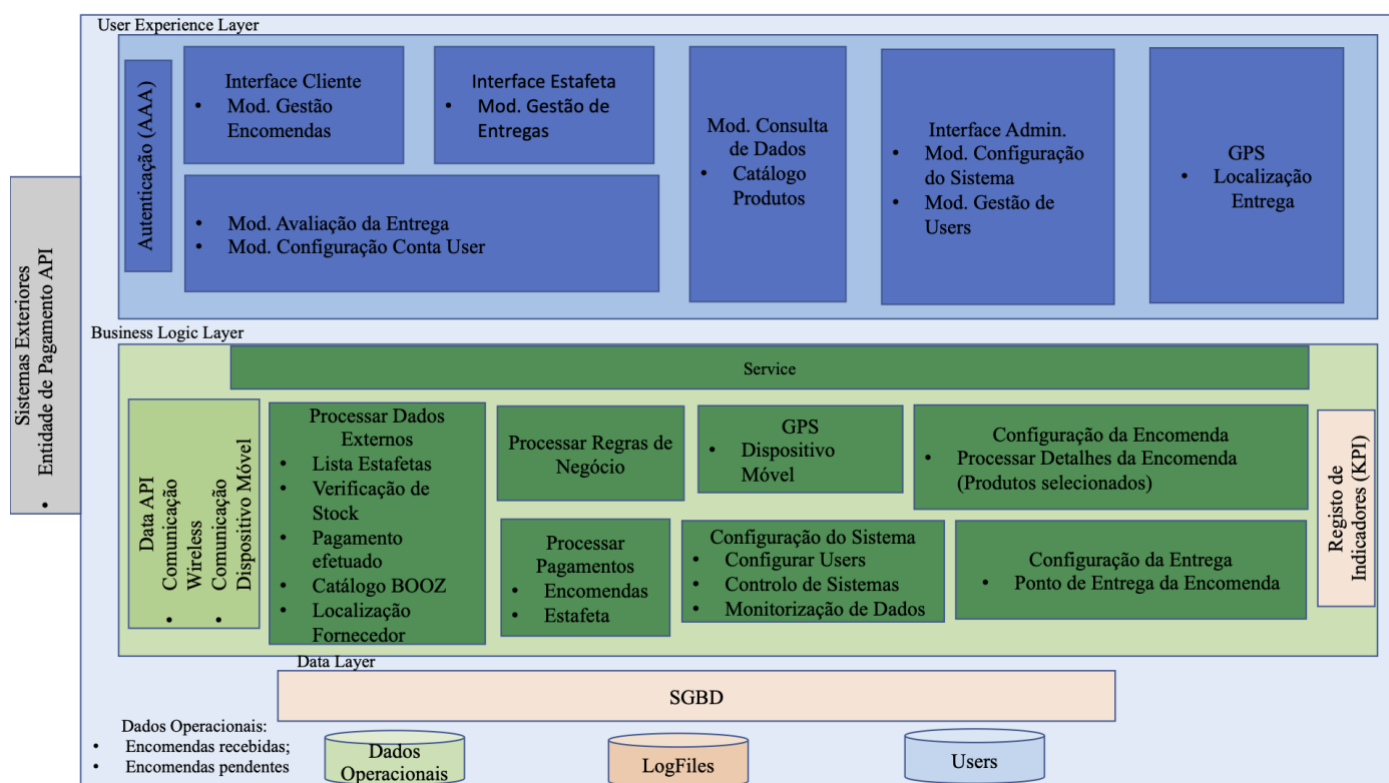


Figura 4. Diagrama de blocos da proposta BOOZ.

O Diagrama de Blocos é uma representação informal do diagrama de contexto. É o diagrama principal para mostrar a composição da arquitetura do Sol (System of Interest), no qual partes principais do sistema (módulos de software) são representados por blocos e podem estar conectados por linhas que demonstram a relação entre os mesmos.

O nosso Diagrama de Blocos demonstra a arquitetura distribuída entre o cliente, servidor e o ambiente envolvido. Temos a camada de apresentação, onde o processamento é feito a partir da interface dos utilizadores (figura 4).

O diagrama é constituído por layers, uma layer é um componente lógico dentro de um software que executa uma dada funcionalidade. Neste caso temos 3: a User Experience Layer, a Business Logic Layer e a Data Layer.

Na primeira Layer, temos os diversos portais para cada um dos users, que pode processar informação (o Cliente, o Estafeta e o Administrador de Sistema), estando sujeitos ao protocolo de autenticação (AAA), para além disto temos também outros módulos de software.

A interface Cliente, é constituída por um módulo de gestão de encomendas, onde o cliente poderá consultar o Catálogo BOOZ e o histórico de encomendas.

A interface Estafeta, é formada por um módulo de Gestão de Entregas, onde ele poderá consultar a Lista de Entregas. O Portal do Administrador de Sistema, contém 2 módulos, onde ele pode configurar Users, controlar sistemas e monitorizar dados.

Os módulos de software existentes nesta camada são: GPS, que permite o acesso à localização da Encomenda, de forma ao cliente poder localizar a sua encomenda; o Módulo de Consulta de Dados, onde o cliente terá acesso e poderá consultar o catálogo BOOZ; o Módulo de Avaliação do Serviço que permite realizar uma avaliação do serviço de entrega e, por último, o Módulo de Configuração Conta User no qual o User pode editar os seus dados pessoais.

Dentro da Business Logic Layer, onde os dados são transformados em informação e direcionados pelo service orchestrator para os respetivos lugares, podemos encontrar os módulos responsáveis pelo processo da Encomenda, Configuração da Encomenda, onde são retirados os detalhes da Encomenda, como por exemplo, produtos selecionados, e Configuração da Entrega, em que são processados os dados de entrega como o ponto de entrega da encomenda.

Para executar o pagamento das encomendas usamos um sistema exterior denominado Entidade de Pagamento que nos vai confirmar se o pagamento foi efetuado com sucesso. No entanto terá de ser validado pelos módulos Processar Regras de Negócio e Processar Pagamentos, este último módulo também faz o processo de pagamentos de encomendas e estafeta.



Temos o processo da localização da encomenda, que nos vai dar o tracking da Encomenda, fornecido pela comunicação Dispositivo Móvel e permite assim ao Cliente receber updates da localização da sua encomenda.

O Módulo Processar Dados Exteriores, transforma os dados transmitidos por uma Data Gateway API, provenientes de sistemas externos como por exemplo a lista de Estafetas, a verificação de Stock, o Pagamento Efetuado, a Localização do Fornecedor, o Catálogo BOOZ.

O módulo de Configuração do Sistema, onde é possível Configurar Users, Controlar Sistemas e Monitorizar Dados.

Por último temos o Registo de Indicadores (KPI) que permite visualizar os diferentes indicadores implementados e realizar uma análise detalhada dos mesmos. As KPI's (Key Performance Indicators), são métricas impostas na empresa para analisar o progresso da empresa e observar se os targets definidos pela mesma estão a ser alcançados.

Na Data Layer, temos as 3 bases de dados controladas pelo SGBD (Sistema de Gestão de Base de Dados). As bases de dados contêm informações processadas pelos Users e neste caso, é efetuado o registo dos Users, o LogFiles e dos dados operacionais (encomendas recebidas, encomendas pendentes).

## Diagrama de Casos de Uso do sistema BOOZ

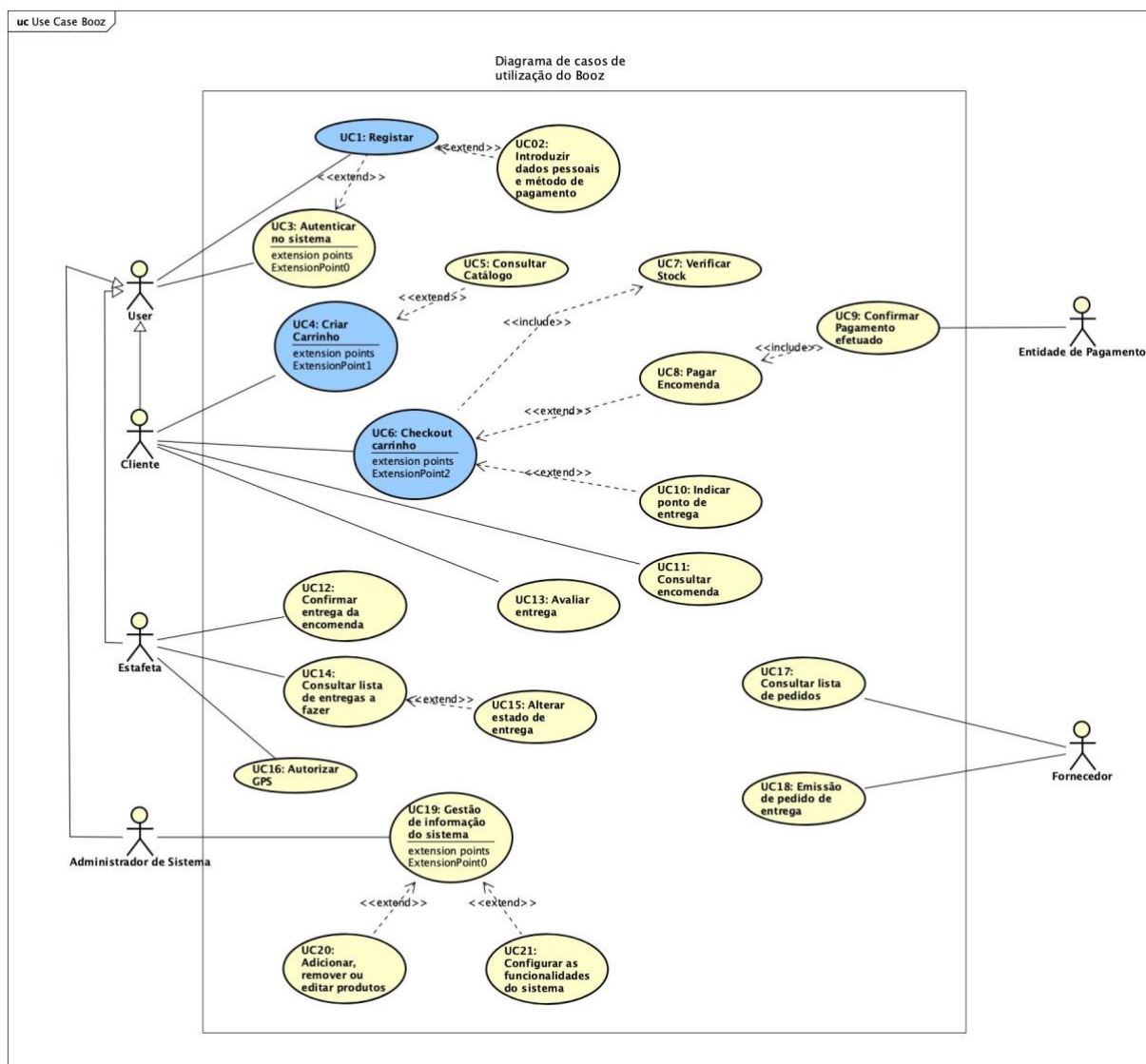


Figura 5. Diagrama de Casos de Uso da proposta BOOZ.

O nosso Diagrama de Casos de Utilização apresenta o nosso sistema de um ponto de vista comportamental. Antes de realizar uma análise mais minuciosa do diagrama em si e do que representam os seus componentes, é importante mencionar alguns conceitos que o compõem, de modo a que, a sua explicação esteja depois mais bem enquadrada e consequentemente mais explícita.

- **Caso de Uso:** Um Caso de Uso descreve uma ação ou sequência de ações que o utilizador pode executar através do sistema. Os Casos de Uso são geralmente representados através de uma elipse horizontal.
- **Ator:** Um Ator, é uma entidade exterior ao sistema, que interage com o mesmo através dos casos de uso, os quais estão representados no Diagrama de Casos de Utilização. Um ator pode ser por exemplo uma pessoa ou uma organização, desde que este interaja com o sistema de uma ou mais maneiras.
- **Relação de Inclusão:** A Relação de Inclusão acontece sempre entre dois casos de uso, na qual, a função executada já está presente num outro caso de uso.
- **Relação de Extensão:** A Relação de Extensão acontece sempre entre dois casos de uso, nesta, um dos casos de uso é estendido pelo outro. Aqui, ambos os casos de uso são independentes, sendo que esta relação é expressa por vezes para demonstrar casos de uso que são apenas de execução opcional.
- **Associação:** Uma Associação existe sempre que um ator está envolvido numa interação descrita por um Caso de Uso. A seta, que se encontra na ponta de uma Associação serve para indicar que parte invoca a outra.
- **Generalização:** A Generalização de um ator é quando este herda o papel de outro, herdando também os seus Casos de Uso. Um ator descendente tem sempre pelo menos um Caso de Uso específico a este.

Esclarecidos os conceitos, pretendemos realizar uma análise mais detalhada do nosso Diagrama de Casos de Utilização. Apenas após a clarificação destes conceitos, é possível começar a pensar nos atores e casos de uso que iremos representar neste diagrama. Na definição de atores procurámos ser coerentes com os restantes diagramas para assegurar a recursividade entre eles.

Podemos primeiramente observar o User, o qual após a autenticação no sistema, pode passar a ser um Cliente, um Estafeta ou um Administrador de Sistema, as três generalizações do User.

O Cliente pode executar onze casos de uso, seja diretamente ou através de extensões. Entre eles, por exemplo, o Caso de Uso 4: Criar carrinho, Caso de Uso 6: Checkout carrinho ou o Caso de Uso 13: Avaliar a entrega.

O estafeta consulta uma lista de entregas (uc:14), confirma a entrega da encomenda (uc:12) e autoriza o seguimento da encomenda por GPS numa primeira instância (uc:16).

O administrador de sistemas gere a informação do sistema (uc:19), adiciona, remove ou edita os produtos da app (uc:20) e configura as funcionalidades do sistema (uc:21).

A entidade de pagamento faz a confirmação dos mesmos (uc:9).

O fornecedor, consulta a lista de pedidos (uc:17) e emite os pedidos de entrega (uc:18).

## a. Casos de Uso – Standard

### UC1 – Registrar

Descrição	O Cliente efetua o seu registo introduzindo os seus dados pessoais para ter uma conta e se poder autenticar.
-----------	--

### UC2 – Introduzir dados pessoais e método de pagamento

Descrição	Use case responsável pela introdução dos dados pessoais para associar à conta.
-----------	--

### UC3 – Autenticar no Sistema

Descrição	Os Users validam a sua entrada no sistema, sendo reconhecidos pelo mesmo como cliente, estafeta ou administrador de sistema.
-----------	--

### UC4 – Criar Carrinho

Descrição	O cliente tem acesso ao catálogo dos produtos. Este use case representa a adição ou remoção de produtos que será a encomenda inserida no sistema.
-----------	---

### UC5 – Consultar Catálogo

Descrição	O sistema ativa a interface que apresenta o catálogo de produtos disponíveis para venda como uma fotografia e preço.
-----------	--

### UC6 – Checkout Carrinho

Descrição	Cliente finaliza o pedido de entrega, cedendo a algumas informações finais e após verificar stock.
-----------	--

**UC7 – Verificar Stock**

<b>Descrição</b>	Sistema efetua a verificação de existência de stock.
------------------	--

**UC8 – Pagar Encomenda**

<b>Descrição</b>	Cliente efetua o pagamento da encomenda.
------------------	--

**UC9 – Confirmar Pagamento Efetuado**

<b>Descrição</b>	A entidade de pagamento confirma ao sistema o pagamento da encomenda por parte do cliente.
------------------	--

**UC10 – Indicar Ponto de Entrega**

<b>Descrição</b>	O Cliente indica o local onde pretende que a sua encomenda seja entregue.
------------------	---

**UC11 – Consultar Encomenda**

<b>Descrição</b>	Neste use case o cliente consegue aceder ao GPS da encomenda.
------------------	---

**UC12 – Confirmar Entrega da Encomenda**

<b>Descrição</b>	O cliente e o estafeta confirmam ambos que a entrega foi realizada com sucesso.
------------------	---

**UC13 – Avaliar Entrega**

<b>Descrição</b>	O Cliente após receber a encomenda faz a avaliação da mesma.
------------------	--

**UC14 – Consultar Lista de Entregas a Fazer**

<b>Descrição</b>	Estafeta acede à lista de pedidos de entregas emitidas pelo fornecedor.
------------------	---

**UC15 – Alterar Estado de Entrega**

<b>Descrição</b>	Neste use case o estafeta consegue alterar o estado da entrega, encomenda aceite, encomenda entregue.
------------------	---

**UC16 – Autorizar GPS**

<b>Descrição</b>	Neste use case o estafeta autoriza o seguimento por GPS por parte do cliente.
------------------	---

**UC17 – Consultar Lista de Pedidos**

<b>Descrição</b>	Fornecedor tem acesso à lista de pedidos de encomenda emitidas pelo sistema.
------------------	--

**UC18 – Emissão de Pedido de Entrega**

<b>Descrição</b>	Fornecedor emite o pedido de entrega ao estafeta.
------------------	---

**UC19 – Gestão de Informação do Sistema**

<b>Descrição</b>	Neste use case o administrador gere toda a informação do sistema.
------------------	---

**UC20 – Adicionar, Remover ou Editar Produtos**

<b>Descrição</b>	Neste use case o administrador efetua a manutenção do catálogo.
------------------	---

**UC21 – Configurar as Funcionalidades do Sistema**

<b>Descrição</b>	Neste use case o administrador configura todas as funcionalidades do sistema.
------------------	---

## b. Caso de Uso Complexo

### UC1 – Registar

ITEM	VALUE
UseCase	UC1: Registar
Actor	User
Description	O Cliente efetua o seu registo introduzindo os seus dados pessoais para ter uma conta e se poder autenticar.
Precondition	Ter a aplicação instalada e ter comunicação wireless via internet.
Main Scenario	1. O utilizador introduz o nome, apelido, morada, data de nascimento, código postal 2. Insere um email 3. Confirmação do email 4. Define uma password 5. Confirmação da password 6. Confirmação do registo 7. O utilizador introduz o nome 8. Apelido 9. Morada 10. Data de nascimento 11. Código postal 12. Método de pagamento 13. Autenticação no sistema
Alternative Scenario	1. O utilizador introduz o nome, apelido, morada, data de nascimento, código postal 2. Insere um email 3. Confirmação do email 4. Define uma password 5. Confirmação da password 6. Confirmação do registo 7. Autenticação no sistema
Exception Scenario	2.1. Caracteres inválidos 2.2. Email já registado 3.1. Emails não são iguais 4.1. Password não cumpre os requisitos 5.1. Passwords não correspondem 7.1. Caracteres inválidos 8.1. Caracteres inválidos 9.1. Morada inválida 10.1. Caracteres inválidos 11.1. Código postal inválido 12.1. Método de pagamento inválido 13.1. Autenticação inválida
Postcondition	Registo e autenticação efetuados com sucesso



## UC4 – Criar Carrinho

ITEM	VALUE
UseCase	UC4: Criar Carrinho
Actor	Cliente
Description	O cliente tem acesso ao catálogo dos produtos. Este use case representa a adição ou remoção de produtos que será a encomenda inserida no sistema.
Precondition	O Cliente autenticado e validado com sucesso.
Main Scenario	1. O Cliente consulta o catálogo de produtos 2. Seleciona os produtos que pretende
Alternative Scenario	1. O Cliente consulta o catálogo de produtos
Exception Scenario	2.1. O cliente cancela o pedido de encomenda do carrinho
Postcondition	O pedido de encomenda do carrinho é processado e é direcionado para finalizar a encomenda.

## UC6 – Checkout Carrinho

ITEM	VALUE
UseCase	UC6: Checkout carrinho
Actor	Cliente
Description	Cliente finaliza o pedido de entrega, cedendo a algumas informações finais e após verificar stock.
Precondition	Verificar o stock
Main Scenario	1. Sistema informa o cliente do total a pagar 2. Cliente efetua o pagamento 3. Cliente indica o ponto de entrega
Alternative Scenario	1. Sistema informa o cliente do total a pagar 2. Cliente efetua o pagamento
Exception Scenario	3.1. Ponto de entrega inválido
Postcondition	Confirmação de pagamento efetuado

## 5 Modelo de Domínio do Sistema Entrega e Encomenda de bebidas alcoólicas e snacks (BOOZ)

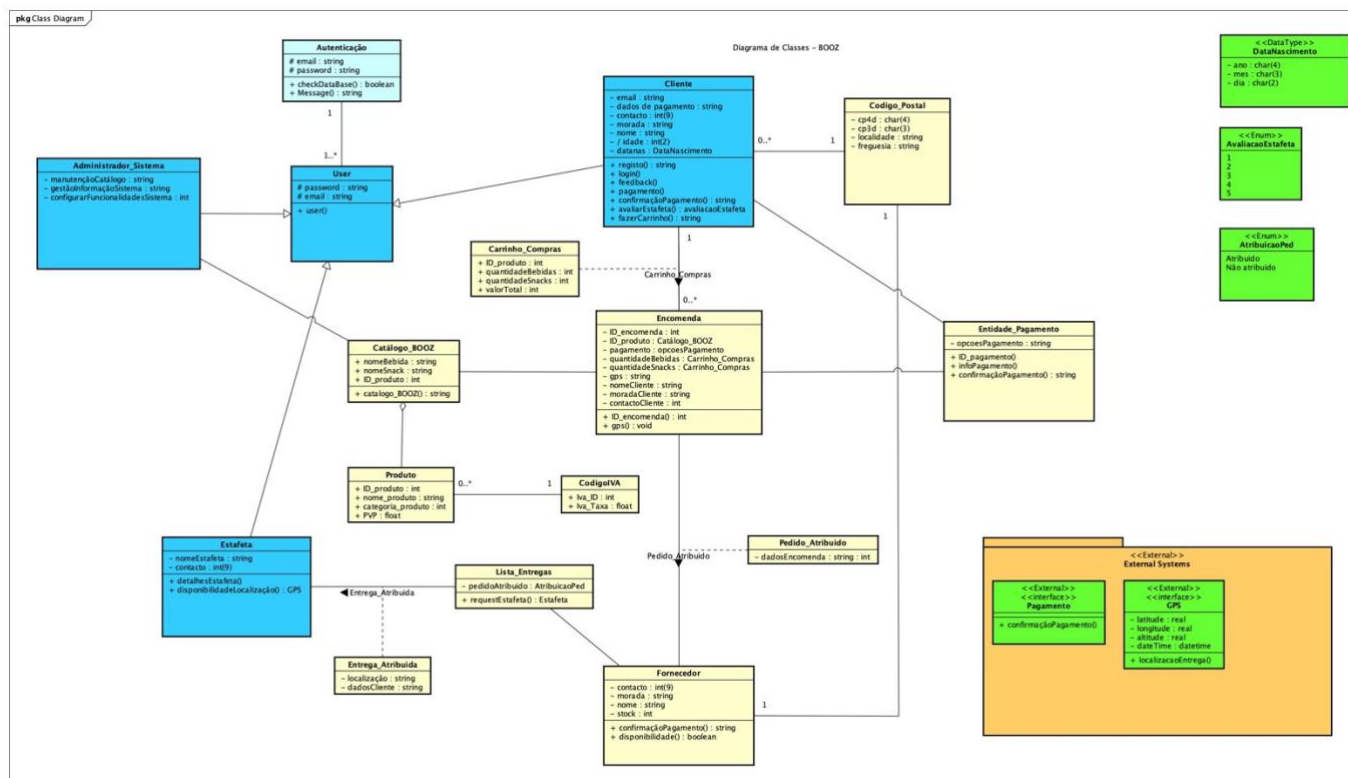


Figura 6. Modelo de Domínio da proposta BOOZ.

O nosso Modelo de Domínio representa como o nosso sistema funciona de forma estática tendo em conta a coerência e funcionalidades definidas de acordo com o project Scope que apresentamos. Antes de explicarmos de forma desconstrutiva o nosso modelo de domínio é importante referir alguns conceitos usados neste diagrama:

- **Classe:** Uma classe é uma abstração de um objeto na vida real, exemplificando com o nosso diagrama, existem vários tipos de abstração como o caso de identificarmos pessoas como objetos.

- **Atributo:** Atributos são o que definem os objetos, podemos dizer que é quem molda o objeto para que a classe se torne mais idêntica ao objeto planeado na vida real.
- **Operações:** Operações são algo que o objeto realiza no nosso sistema. Podemos dizer que são as funcionalidades que depois de o objeto ser transformado em classe pode executar no nosso sistema.

A partir daqui conseguimos começar a formar vários tipos de objetos com a sua identidade (Atributos) e funcionalidades (operações) para representarmos como a nosso sistema funciona e mais importante ser coerente com a nossa lógica de negócio. Do ponto de vista de um Analista de Sistemas os clientes são os programadores que vão desenvolver o nosso sistema.

Tendo em conta que já explicamos o que é uma classe num sistema necessitamos de outros conceitos que envolvam o relacionamento entre classes. Neste campo temos vários tipos de associações entre classe. Utilizamos associações de dependência, de partilha de objetos, e associações onde tenha que existir uma classe que faça algo que só acontece quando as classes estão interligadas (Classe intermédia). As relações devem também utilizar Cardinalidade

Stereotypes servem para definir roles de classes.

Utilizamos vários tipos de **Stereotypes** de classe:

- **Enumerações:** enumerações funcionam como listas, mas com valores predefinidos, no nosso caso utilizamos uma enumeração considerado se a entrega foi ou não atribuída a um estafeta e a classificação da mesma.
- **DataType:** Como o nome indica este *Stereotype* tem como a importância referir datas de acordo como nós desejamos. Exemplificando no nosso sistema existe um DataType. É referido como um genérico para a data de nascimento apenas contendo o formato AAAA/MM/DD.

- External: utilizamos este Stereotype para identificar que estas classes fazem parte de outro package que comunica com o exterior. São basicamente onde as nossas classes vão buscar informação real, para o que necessitam.
- Interface: Interfaces são o tipo de stereotype onde podemos declarar, mas não instanciar vários tipos de operações.

O nosso modelo de domínio representa a nossa lógica de negócio onde:

- 1- O cliente que faz a encomenda;
- 2- A encomenda passa para a Lista de pedido atribuído;
- 3- A encomenda passa para o fornecedor que emite um pedido de entrega para o estafeta;
- 4- O estafeta confirma que a encomenda foi entregue;
- 5- O cliente avalia a prestação do estafeta numa escala de 1 a 5 valores;

Começando por explicar o cliente que representa uma generalização do User, ou seja, o User partilha todos os atributos protected para o cliente, e no cliente temos atributos privados onde são restritos apenas ao cliente. É perceptível através da cardinalidade que um cliente pode ter várias encomendas, mas que cada encomenda apenas tem atribuído um cliente.

A classe encomenda tem vários atributos com tipos de outra classe porque é a classe mais importante no diagrama, toda a nossa lógica de negócio depende da encomenda, ou seja, esta contém tudo o que o cliente necessita para fazer a encomenda.

Após a encomenda ser efetuada com sucesso, a mesma passa para a ser processada na lista de pedidos do fornecedor, que posteriormente emite uma Lista de Entregas que é preenchida pelo estafeta com o estado de atribuída ou não atribuída.

As classes a azul representam as classes que derivam da autenticação. A autenticação contém as informações necessárias para o cliente entrar no sistema e ser redirigido para a página consoante o tipo de user. Existem três tipos de user no nosso sistema que são direcionados para diferentes páginas. O Cliente tem acesso à página onde pode fazer encomenda ou editar os seus dados. O estafeta tem acesso à lista de entregas a fazer, e o administrador de sistema tem acesso ao seu dashboard para gerir funcionalidades no sistema.

## **1 Conclusões e Trabalho Futuro**

A elaboração deste projeto teve a duração de cerca de 4 meses na qual consistia na conceção de um modelo de negócio. O desenvolvimento deste projeto tem como objetivo alcançar o mercado de bebidas alcoólicas, pois após a realização de uma análise de mercado feita pelo grupo de trabalho verificou-se uma grande dificuldade na logística de distribuição destes produtos. Foram retiradas ideias de diversas empresas como a Uber Eats que já têm uma posição estável no mercado atual e também em empresas reconhecidas a nível nacional como a Garrafeira Nacional. Este mercado é um mercado em ascensão e por este motivo decidimos desenvolver um projeto de start-up que resolve as lacunas dos mesmos acima referidos.

O nosso sistema de informação pretende fornecer inicialmente à zona da Grande Lisboa a disponibilidade que estas empresas já reconhecidas no mercado não conseguem corresponder a estas necessidades.

No desenvolvimento deste projeto verificámos que um dos problemas mais graves e que poderá causar um impacto negativo na implementação deste novo projeto indicia ser a pandemia atual que vivemos, pois transmite incertezas à sociedade e à economia que se encontra em declínio constante.

Tendo em consideração as mudanças tecnológicas utilizadas nestes últimos anos recorrentes do avanço da indústria 4.0, que visa a desmaterialização de processos bem como a mão de obra, considerámos a utilização de plataformas digitais tais como websites e aplicações móveis como um meio viável e que vai ao encontro da realidade dos dias de hoje. A acessibilidade e a simplificação são um dos focos da nossa aplicação.

De modo a que não sejam comprometidos os dados pessoais dos nossos utilizadores, optámos pela criação de um sistema de gestão de base de dados no qual assegura a total confidencialidade de cada utilizador registado no mesmo. Foram implementados módulos focados num cumprimento de restrições de integridade e qualidade dos dados, de modo a garantir a exatidão e a consistência dos mesmos.

## **Referências**

- Conteúdos programáticos lecionados na unidade curricular de Arquitetura de Sistema de Informação e Ciências Fontes e Métodos.

## Biografia dos Autores



### **(20200538) Francisco Barros Cabreiro**

Venho do curso de ciências socioeconómicas, de momento frequento o primeiro ano de informática de gestão. Estou a gostar muito do curso em si pois está a corresponder as minhas expectativas.

Em relação a esta cadeira acho que aborda conteúdos bastante interessantes que nos ajudaram ao longo do curso.



### **(20200758) Gisela Joice Cuber Marques**

Estes primeiros meses do primeiro ano da licenciatura em informática de gestão têm sido enriquecedores e interessantes tenho gostado até agora de tudo que foi feito, bem como a execução deste projeto que é muito útil para o que possamos fazer futuramente.



### **(20200573) Gonçalo Nuno Borges de Carvalho**

Estudante no IADE matriculado no curso de Informática de Gestão do primeiro ano, subdelegado da turma e representante dos alunos do mesmo curso.

Concluí o curso de Ciências e Tecnologias na Escola Secundária de Camões, frequentei durante 6 anos a Escola de Música do Conservatório Nacional e trabalhei no passado numa papelaria, num call center e num restaurante. Nasci e vivo em

Lisboa.



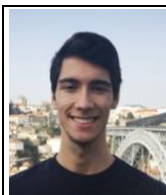
### **(20200154) Madalena Almeida Mendes Barreiros Cardoso**

Sou aluna do primeiro ano de Informática de Gestão e até agora estou a gostar do curso em geral e das cadeiras. Esta cadeira em específico, do meu ponto de vista, trata conteúdos que nos iram ajudar bastante no futuro quando estivermos no mercado de trabalho.



### **(20200908) Pedro Domingues da Cunha**

Sou aluno do primeiro ano de informática de gestão no IADE e escolhi este curso porque é uma área de interesse pessoal e além disso, do meu ponto de vista, pareceu um curso bastante promissor para o futuro e até agora está a corresponder com as minhas expectativas.



### **(20200003) Ricardo Fonseca Fontes**

O gosto pela área começou cedo, despertou um interesse enorme que me fez chegar onde estou hoje. Tirei tanto o curso profissional como um curso técnico superior na área de informática de gestão. As oportunidades que as novas tecnologias trazem são infinitas e tenho a certeza que é esse o caminho que pretendo trilhar.

## Anexo A: Tabela de Indicadores

#	Nome Indicador	Metadados de Caracterização
1	Nº de Entregas	<p><b>Descrição:</b> Esta KPI tem como objetivo averiguar quantas encomendas são entregues face ao total de encomendas realizadas. Esta informação tem por base os dados retirados da nossa aplicação.</p> <p><b>Categoria/Tipo:</b> Lagging/On-target</p> <p><b>Métrica:</b> Soma de todas as entregas feitas</p> <p><b>Threshold:</b> 98% e desvio aceitável de 1%.</p> <p><b>Frequência:</b> Semanalmente</p>
2	Média do tempo de entrega	<p><b>Descrição:</b> Esta KPI tem como objetivo calcular a média do tempo que o estafeta demora a efetuar a entrega, tanto a recolha no fornecedor como a entrega ao cliente.</p> <p><b>Categoria/Tipo:</b> Lagging/On-target</p> <p><b>Métrica:</b> Tempo total sobre a distância percorrida</p> <p><b>Threshold:</b> 1 minuto e 30 segundos /Km e um desvio aceitável de 1 minuto.</p> <p><b>Frequência:</b> Diariamente</p>
3	Classificação do Estafeta	<p><b>Descrição:</b> Esta KPI pretende mostrar o nível da classificação que o cliente atribuiu ao estafeta e ao serviço em geral. O cliente avalia numa escala de 1 a 5, sendo 1 muito mau e 5 excelente.</p> <p><b>Categoria/Tipo:</b> Lagging/On-target</p> <p><b>Métrica:</b> Classificações totais a dividir pelo número de classificações</p> <p><b>Threshold:</b> O objetivo seria obter uma classificação superior a 4,8. Sendo o desvio aceitável de 0,20.</p> <p><b>Frequência:</b> Mensalmente</p>
4	Stock Esgotado	<p><b>Descrição:</b> Esta KRI tem como objetivo monitorizar o número de produtos por dia armazenados.</p> <p><b>Categoria/Tipo:</b> Lagging / On-target</p> <p><b>Métrica:</b> Número de produtos no início do dia – número de produtos vendidos</p> <p><b>Threshold:</b> Quantidade de produto tem de ser superior a 10 unidades.</p> <p><b>Frequência:</b> Diariamente</p>



5	Taxa de crescimento de utilizadores	<p><b>Descrição:</b> Esta KPI tem como objetivo a contabilização do crescimento de utilizadores do serviço em percentagem.</p> <p><b>Categoria/Tipo:</b> Lagging / Benefício</p> <p><b>Métrica:</b> Soma de todos os novos utilizadores do mês / total de utilizadores * 100</p> <p><b>Threshold:</b> Cerca de 100 novos utilizadores por mês, não tem mínimo</p> <p><b>Frequência:</b> Mensalmente</p>
6	Média de vendas	<p><b>Descrição:</b> Esta KRI tem como objetivo elaborar o balanço geral de vendas executadas através da aplicação BOOZ e verificar se há um aumento ou diminuição significativa. Esta informação pode ser recolhida na base de dados da nossa aplicação.</p> <p><b>Categoria/Tipo:</b> Lagging / Benefício</p> <p><b>Métrica:</b> Vendas totais a dividir pelo número de encomendas</p> <p><b>Threshold:</b> O valor mínimo será de 1000 vendas e não tem valor máximo.</p> <p><b>Frequência:</b> Mensalmente</p>

## Anexo B: Especificação das Interfaces API

Descrição: Application Programing Interface				Versão: 4.0
1	Dispositivo Móvel		Descrição: O dispositivo móvel é um dos enabling systems, estando out of the scope comunica com o nosso sistema através de uma API (Application Programing Interface) é responsável por dar ao cliente todo o acesso à aplicação e ao serviço.	
	Parâmetros de Input			
	Atributo	Tipo	Obr.	Descrição do Atributo
	N/A		S/N	N/A
	Parâmetros de Output			
	Atributo	Tipo	Obr.	Descrição do Atributo
	N/A			N/A
2	Comunicação Wireless		Descrição: A comunicação wireless é um dos enabling systems, estando out of the scope, comunica com o nosso sistema através de uma API (Application Programing Interface), é necessária para que o sistema funcione pois dependemos de uma ligação à internet.	
	Parâmetros de Input			
	Atributo	Tipo	Obr.	Descrição do Atributo
	N/A			N/A
	Parâmetros de Output			
	Atributo	Tipo	Obr.	Descrição do Atributo
	N/A			N/A
3	Fornecedor		Descrição: O Fornecedor comunica com o nosso sistema através de uma API (Application Programing Interface) é responsável por consultar os pedidos e emitir os pedidos de entrega.	
	Parâmetros de Input			
	Atributo	Tipo	Obr.	Descrição do Atributo
	Consulta de pedidos			Consulta a lista de pedidos pendentes
	Parâmetros de Output			
	Atributo	Tipo	Obr.	Descrição do Atributo
	Emissão de pedido de entrega			Emite um pedido de entrega aos estafetas
4	Entidade de pagamento		Descrição: A Entidade de pagamento comunica com o nosso sistema através de uma API (Application Programing Interface), é responsável por assegurar a confirmação de pagamento, após efetuar o preenchimento de todas as informações necessárias para a finalização da encomenda, o cliente é direcionado para a entidade de pagamento para que a encomenda possa ser finalizada.	
	Parâmetros de Input			
	Atributo	Tipo	Obr.	Descrição do Atributo
	Total a pagar e meio de pagamento			Recebe o total a pagar e o meio de pagamento utilizado
	Fornecer dados de cliente			Recebe os dados de pagamento
	Parâmetros de Output			
	Atributo	Tipo	Obr.	Descrição do Atributo
	Confirmação de pagamento efetuado			Emite a confirmação de pagamento
	Redireciona o cliente			Redireciona o cliente

## Anexo C: Levantamento de Requisitos Funcionais

#	Nome Requisito	Descrição Sucinta
R01	Registar	O Cliente efetua o seu registo introduzindo os seus dados pessoais para ter uma conta e se poder autenticar.
R02	Introduzir dados pessoais e método de pagamento	O cliente efetua a introdução dos dados pessoais para associar à conta.
R03	Autenticar no Sistema	Os Users validam a sua entrada no sistema, sendo reconhecidos pelo mesmo como cliente, estafeta ou administrador de sistema.
R04	Criar Carrinho	O cliente tem acesso ao catálogo dos produtos. Este use case representa a adição ou remoção de produtos que será a encomenda inserida no sistema.
R05	Consultar Catálogo	O sistema ativa a interface que apresenta o catálogo de produtos disponíveis para venda como uma fotografia e preço.
R06	Checkout Carrinho	Cliente finaliza o pedido de entrega, cedendo a algumas informações finais e após verificar stock.
R07	Verificar Stock	Sistema efetua a verificação de existência de stock.
R08	Pagar Encomenda	Cliente efetua o pagamento da encomenda.
R09	Confirmar Pagamento Efetuado	A entidade de pagamento confirma ao sistema o pagamento da encomenda por parte do cliente.
R10	Indicar Ponto de Entrega	O Cliente indica o local onde pretende que a sua encomenda seja entregue.
R11	Consultar Encomenda	O cliente consegue aceder ao GPS da encomenda.
R12	Confirmar Entrega da Encomenda	O cliente e o estafeta confirmam ambos que a entrega foi realizada com sucesso.
R13	Avaliar Entrega	O Cliente após receber a encomenda faz a avaliação da mesma.
R14	Consultar Lista de Entregas a Fazer	Estafeta acede à lista de pedidos de entregas emitidas pelo fornecedor.
R15	Alterar Estado de Entrega	O estafeta consegue alterar o estado da entrega, encomenda aceite, encomenda entregue.
R16	Autorizar GPS	O estafeta autoriza o seguimento por GPS por parte do cliente.
R17	Consultar Lista de Pedidos	Fornecedor tem acesso à lista de pedidos de encomenda emitidas pelo sistema.
R18	Emissão de Pedido de Entrega	Fornecedor emite o pedido de entrega ao estafeta.
R19	Gestão de Informação do Sistema	O administrador gere toda a informação do sistema.
R20	Adicionar, Remover ou Editar Produtos	O administrador efetua a manutenção do catálogo.
R21	Configurar as Funcionalidades do Sistema	O administrador configura todas as funcionalidades do sistema.

## **Anexo D: Funcionamento da Unidade Curricular**

O funcionamento da Unidade Curricular de Arquitetura de Sistemas de Informação foi a cadeira introdutória ao desenvolvimento, análise e implementação de sistemas de informação.

Primeiramente foi feita a introdução à linguagem UML, de modo a permitir uma compreensão metódica do sistema em si. Foram introduzidos vários conceitos fundamentais relacionados com a conceção de sistemas de informação e respetivos diagramas que auxiliam o seu desenvolvimento.

O primeiro diagrama a ser introduzido é o diagrama de contexto, um diagrama informal, que explica a interação dos atores com o nosso sistema assim como os seus enabling systems. Foi-nos dado formação de UML através de diagramas formais como o diagrama de classes. O diagrama comportamental que nos foi introduzido foi o diagrama de Casos de Utilização, onde conseguimos visualizar as funcionalidades e comportamentos do sistema. A unidade curricular não se baseia só em modelar o nosso sistema, mas também em adquirir certos conceitos relativos à lógica de negócio. Para tal aprendemos a modelar o Lean Canvas, um modelo informal acerca da lógica de negócio, onde nos permite ter a noção de quem queremos atingir com a criação do nosso sistema e de como podemos atenuar os seus problemas. Este modelo também permite a análise de despesas e lucros relativos ao nosso sistema.

O diagrama de blocos organiza todos os módulos do sistema em três camadas distintas, camada de apresentação, camada de aplicação e camada de dados. A camada de apresentação foca-se na identificação dos módulos de software associados aos GUI (Graphical User Interface) onde devem ser validados requisitos de interface gráfica, com facilidade de acesso e leitura do end user. A camada de aplicação foca-se na identificação do módulo de software responsáveis pelo processamento dos dados e validação de regras de negócio, inclui toda a componente de API (Application Programming Interface) para assegurar a comunicação com sistemas externos. A camada de dados tem como foco a gestão da persistência dos dados, assegurar o cumprimento de Restrições de Integridade (RI) e a qualidade dos dados nas Bases de Dados, sobretudo as RI

associadas a regras de negócio (RIN), validação de acessos, incluindo o registo de dados decorrente do AAA (representado esquematicamente pelo LogFile).

A informação que nos ficou mais retida foi que todos os diagramas devem seguir uma certa rastreabilidade para que a análise fique coerente e assertiva para podermos minimizar o risco.

Sugestões de melhorias: mais fornecimento de teoria em determinados tópicos, equilíbrio do conteúdos, melhor introdução e explicação do Project description.