2019/2020



Trading Platform – 1ª fase

Arquiteruras de Software

Fábio senra A82108

César borges A81644

Índice

[Introdução 3](#_Toc22477674)

[Modelo de Domínio 4](#_Toc22477675)

[Principais entidades da plataforma: 4](#_Toc22477676)

[Descrição do modelo de domínio 4](#_Toc22477677)

[Funcionalidades do sistema 5](#_Toc22477678)

[Diagrama de use cases 5](#_Toc22477679)

[Use Case – Compra de CFD 6](#_Toc22477680)

[Use Case – Venda de CFD 7](#_Toc22477681)

[Definição de Stop Loss/Top Profit 7](#_Toc22477682)

[Mockups da plataforma 8](#_Toc22477683)

[Menu Inicial 8](#_Toc22477684)

[Menu de registo 8](#_Toc22477685)

[Menu de utilizador 8](#_Toc22477686)

[Menu “Comprar CFD” 8](#_Toc22477687)

[Menu “Ver portfolio” 8](#_Toc22477688)

[Menu “Ver CFDs em posse” 9](#_Toc22477689)

[Menu de definição de Top Profit/Stop Loss 9](#_Toc22477690)

[Menu de CFD possuído 9](#_Toc22477691)

[Diagrama de classes 10](#_Toc22477692)

[Diagrama de estados 11](#_Toc22477693)

[Figura 1 - Diagrama de Use Cases 5](#_Toc22477697)

[Figura 2 - Use Case - Compra de CFD 6](#_Toc22477698)

[Figura 3 - Use Case - Venda de CFD 7](#_Toc22477699)

[Figura 4 - Use Case - Definição de Stop Loss/Top Profit 7](#_Toc22477700)

[Figura 5 - Menu Inicial 8](#_Toc22477701)

[Figura 6 - Menu de registo 8](#_Toc22477702)

[Figura 7 - Menu de utilizador 8](#_Toc22477703)

[Figura 8 - Menu "Comprar CFD" 8](#_Toc22477704)

[Figura 9 - Menu "Ver portfolio" 8](#_Toc22477705)

[Figura 10 - Menu "Ver CFDs em posse" 9](#_Toc22477706)

[Figura 11 - Menu de definição de Top Profit/Stop Loss 9](#_Toc22477707)

[Figura 12 - Menu de CFD possuído 9](#_Toc22477708)

# Introdução

O projeto atual tem como principal a construção de uma arquitetura de software para uma plataforma de trocas. A plataforma deve permitir aos seus utilizadores várias formas de negociação de contratos de diferenças sobre vários tipos de ativos financeiros (ações, commodities, índices, moeda). Os valores destes contratos devem regular-se pelas subidas e descidas que ocorrem na bolsa de valores real.

É pretendido ainda que se realizem alguns diagramas em UML de maneira a ilustrar algumas das decisões tomadas ao longo do trabalho.

Começar-se-á pelo modelo de domínio, a definição das funcionalidades principais, quais os atributos de qualidade adotados, a elaboração de alguns use cases e os seus respetivos diagramas de sequência, terminando com um diagrama de estados que revela o fluxo que o utilizador poderá fazer na aplicação.

# Modelo de Domínio

Neste capítulo apresentar-se-á as principais entidades do sistema em questão e ainda as relações que essas entidades estabelecem entre si. Tudo isto será baseado em algumas plataformas já existentes no mercado e a forma como estas trabalham.

## Principais entidades da plataforma:

* Contrato de diferenças (CFD)
* Investidor
* Trader
* Saldo
* Portfolio
* Ativos financeiros (Ações, Commodities, Índices, Moedas)

## Descrição do modelo de domínio

A entidade principal do sistema será o contrato de diferenças, uma vez que, este englobará todas as outras entidades existentes. Por sua vez, os investidores e os Traders serão quem cria os contratos abrindo posições de compra ou venda, cada um destes tem ainda o seu saldo que lhes permite comprar contratos.

À medida que se vão realizando contratos de diferenças é importante guardar um Portfolio para que todas essas transações que foram efetuadas no passado sejam guardadas para que os utilizadores possam saber tudo o que já realizaram na plataforma. De referir ainda que, todos estes contratos são realizados sobre um tipo de ativo financeiros podendo este ser uma ação, um commodity (Ouro, Petróleo, Prata), um índice ou uma moeda.

POR IMAGEM DO MODELO DE DOMINIO

# Funcionalidades do sistema

Depois de algum estudo sobre as outras plataformas já existentes encontraram-se as seguintes funcionalidades:

* Autenticação: permite a entrada dos utilizadores nas suas contas.
* Compra e venda de CFD: os utilizadores podem abrir posições de compra/venda de CFD.
* Ver portfolio: os utilizadores podem ver todos os CFD que já estabeleceram.
* Definição de Top Profit/Stop Loss: os utilizadores devem conseguir estabelecer limites de perda/ganho em cada contrato.
* Depositar/levantar dinheiro: o saldo de cada utilizador pode ser alterado pelos mesmos.
* Criação de conta: para aceder à plataforma os utilizadores devem criar uma conta.
* Eliminação de contas: esta funcionalidade poderá ser apenas executada pelo administrador do sistema.

# Diagrama de use cases

Todas as funcionalidades supracitadas deram posteriormente origem aos use cases do sistema.

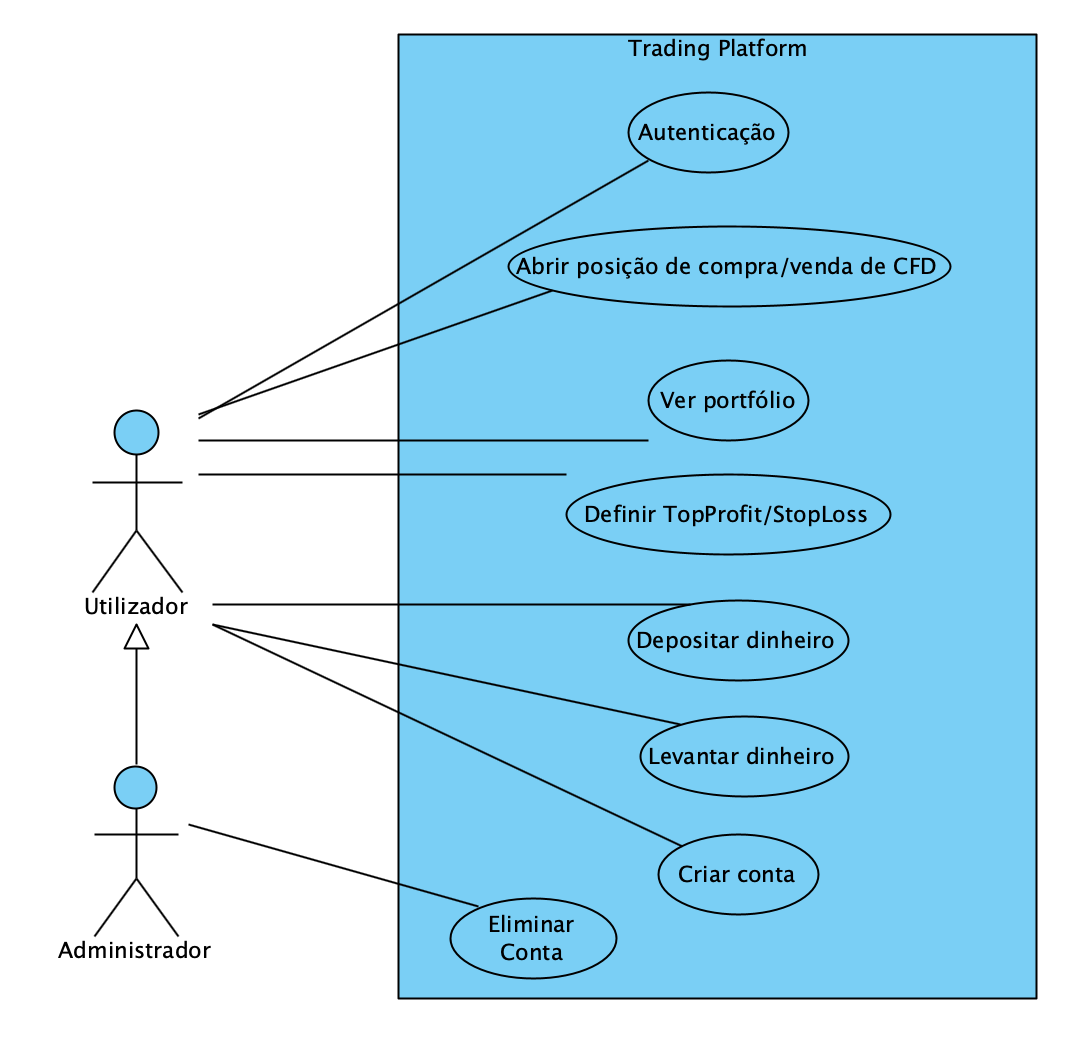


Figura - Diagrama de Use Cases

Na especificação dos use cases foi decidido que apenas seriam especificados os use cases mais importantes do sistema.

## Use Case – Compra de CFD

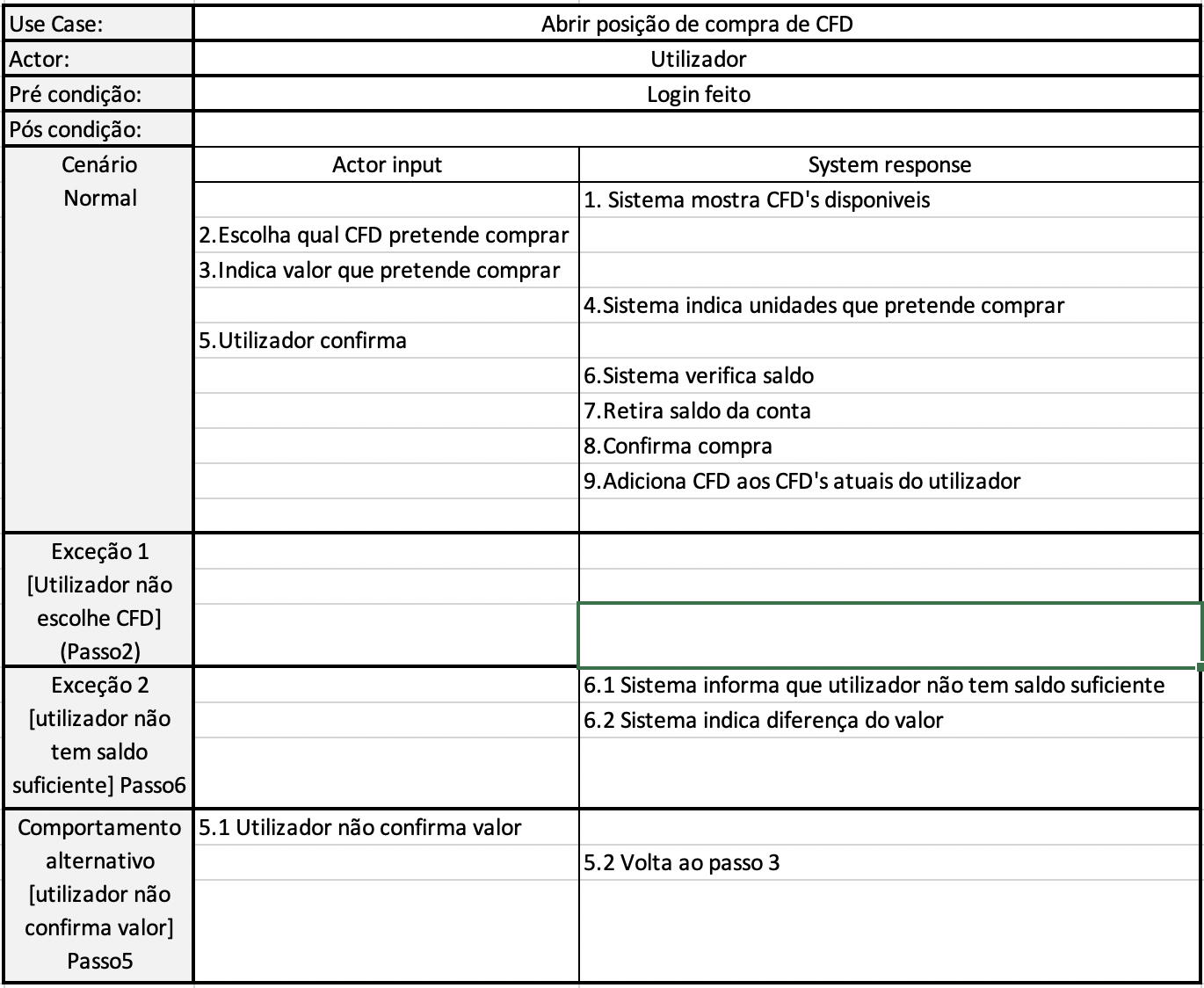


Figura - Use Case - Compra de CFD

## Use Case – Venda de CFD

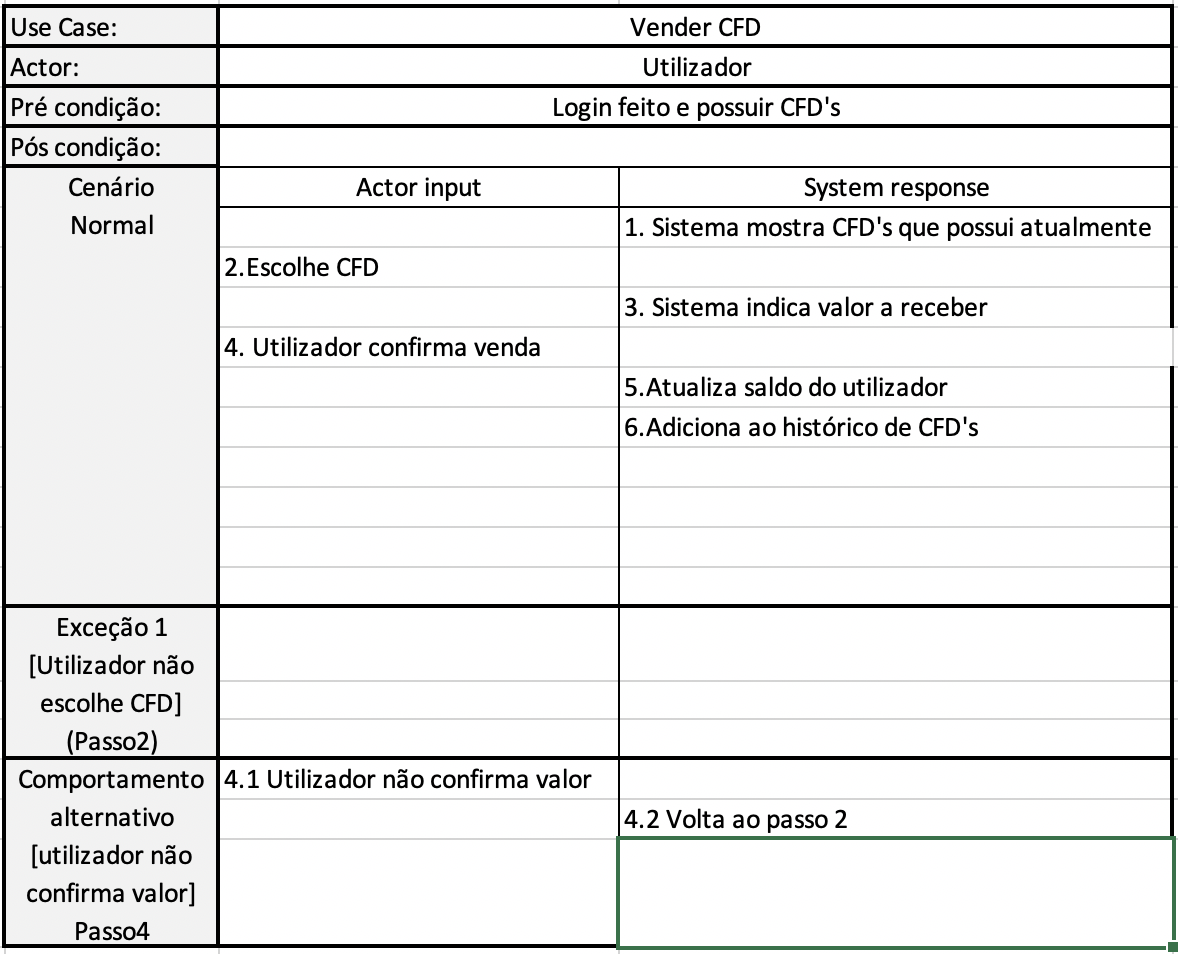


Figura - Use Case - Venda de CFD

## Definição de Stop Loss/Top Profit

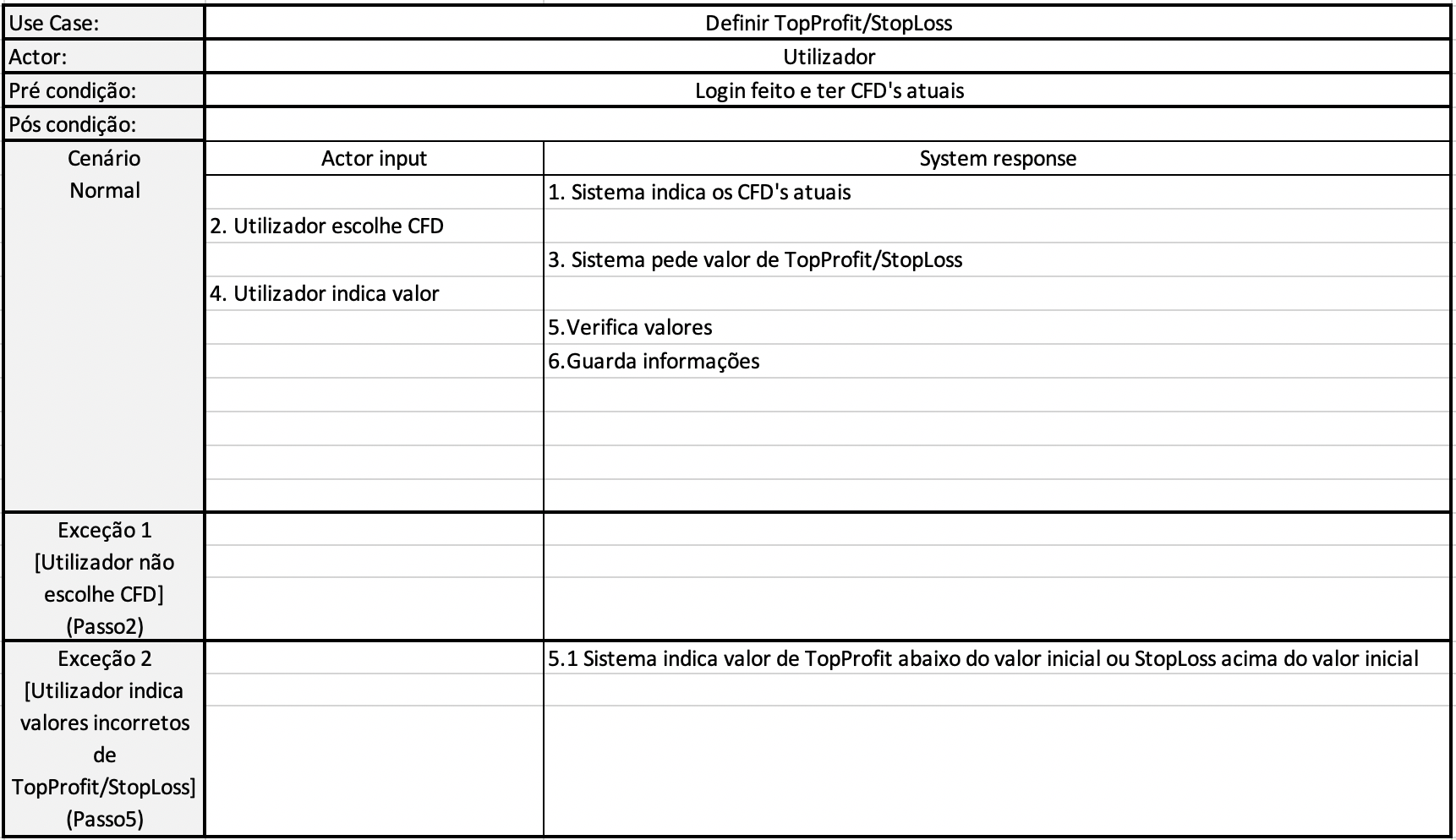


Figura 4 - Use Case - Definição de Stop Loss/Top Profit

# Mockups da plataforma

Nesta secção mostrar-se-ão as interfaces que os utilizadores vão ver no uso das várias funcionalidades do sistema.

## Menu Inicial

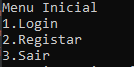


Figura - Menu Inicial

## Menu de registo



Figura - Menu de registo

## Menu de utilizador

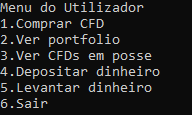


Figura - Menu de utilizador

## Menu “Comprar CFD”

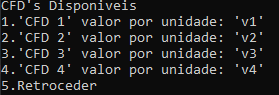


Figura - Menu "Comprar CFD"

## Menu “Ver portfolio”

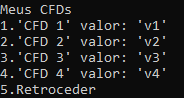


Figura - Menu "Ver portfolio"

## Menu “Ver CFDs em posse”



Figura - Menu "Ver CFDs em posse"

## Menu de definição de Top Profit/Stop Loss

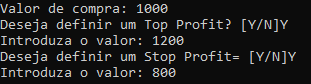


Figura - Menu de definição de Top Profit/Stop Loss

## Menu de CFD possuído



Figura - Menu de CFD possuído

# Atributos de qualidade e cenários

Como principais atributos de qualidade foram escolhidos a **usabilidade** e **manutenibilidade**.

Sendo o nosso principal foco o utilizador, ter um sistema com o qual o utilizador se sinta confortável a utilizar é uma prioridade. Por isso, é necessário que o sistema tenha uma interface amigável, não apresentar um grande nível de complexidade, isto é, as suas funcionalidades sejam facilmente aprendidas pelo utilizador, ser um sistema que evolua de acordo com as necessidades dos mesmos.

Por outro lado, a evolução e o acréscimo de funcionalidades ao sistema é também algo importante para que o sistema continue a corresponder as necessidades dos seus utilizadores. Deste modo, a manutenibilidade é um aspeto essencial para que o sistema responda de forma correta à adição de novas funcionalidades, à modificação de funcionalidades já existentes e até à remoção de funcionalidades que se vão tornando obsoletas.

# Diagrama de classes

Depois de uma análise de todos os diagramas construídos, chegou-se ao seguinte diagrama de classes, ainda sendo uma previsão do que se pretende implementar podendo ainda sofrer alterações no futuro.

POR IMAGEM DO DIAGRAMA DE CLASSES

# Diagrama de estados

Nesta secção pretende-se mostrar o fluxo que cada utilizador pode fazer desde o momento em que entra na plataforma até à saída do sistema, passando por todas as funcionalidades que já foram descritas em cima.