

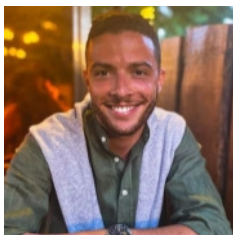
UMinho

Mestrado Engenharia Informática

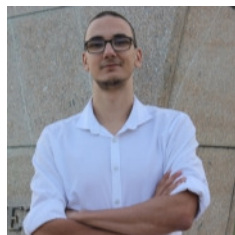
Requisitos e Arquiteturas de Software

Grupo ??, PL3 / Entrega 2

Francisco Alves Andrade PG47185
Luís Pedro Oliveira de Castro Vieira PG47430
Maria Quintas Barros PG47488
Tiago Carvalho Freitas PG47687



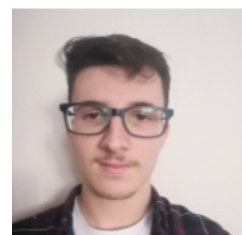
PG47185



PG47430



PG47488



PG47687

17 de dezembro de 2021
2021/22

Índice

1	Introdução e Objetivos	1
1.1	Requisitos Gerais	1
1.2	Público Alvo	1
2	Restrições de Arquitetura	2
2.1	Restrições Técnicas	2
2.2	Convenções	2
3	Contexto e Foco	3
3.1	Contexto do Negócio	3
3.2	Contexto Técnico	3
4	Estratégia de Solução	4
5	Building Block View	4
6	Runtime View	5
6.1	Login	5
6.2	Logout	6
6.3	Registrar Utilizador	6
6.4	Editar Perfil	7
6.5	Consultar Saldo	7
6.6	Consultar histórico	8
6.7	Cash-in	8
6.8	Cash-out	9
6.9	Consultar Jogos	9
6.10	Escolher aposta	10
6.11	Eliminar aposta	11
6.12	Confirmar Boletim	12
6.13	Eliminar Boletim	12
6.14	Consultar boletim	13
7	Deployment View	13
8	Conceitos Transversais	14
8.1	Modelo de Domínio	15
8.2	Experiência do Utilizador	15
8.2.1	Interface do Utilizador	15
8.2.2	Internacionalização	15
8.3	Conceitos <i>Safety</i> e <i>Security</i>	16
8.4	Conceitos “ <i>Under-the-hood</i> ”	16
8.4.1	Gestão de Transações	16
8.4.2	Gestão de Sessões	16
8.4.3	Tratamento de Erros e Excepções	16
8.4.4	Paralelização / Threading	17
8.5	Conceitos de Desenvolvimento	17
8.5.1	Configuração	17

8.5.2	Migração	17
8.6	Conceitos de Operacionalidade	17
8.6.1	Administração	17
8.6.2	Manutenção	17
9	Decisões Arquitetônicas	18
10	Requisitos de Qualidade	18
10.1	Facilidade de Utilização	18
10.2	Segurança	19
11	Riscos e Dívida Técnica	20
11.1	Riscos de Segurança	20
11.2	Riscos de Sobrecarregamento	20
11.3	Riscos de Obsolescimento	20
11.4	Riscos Legais	20
12	Glossário	21

1 Introdução e Objetivos

A plataforma proposta para desenvolvimento tem como ideia principal ser o suporte de apostas de uma casa de apostas desportivas. Como forma de perceber aquilo a que o projeto deveria ser capaz de responder e como deveria responder, a equipa decidiu investigar outros produtos do mesmo género como a Betclic e o Placard e a partir daí retirar informação suficiente para começar a moldar o projeto. Assim sendo, a plataforma desenvolvida terá de ser capaz de efetuar apostas, permitir aos utilizadores criarem conta com *plafond* inicial, permitir que retirem dinheiro, entre muitas outras ações.

1.1 Requisitos Gerais

O grande objetivo da *MRMPtech* é a gestão de apostas, isto é, gerir a aposta de um utilizador, bem como o dinheiro nela envolvido. Assim, lista-se a seguir os requisitos funcionais aos quais a plataforma terá de ser capaz de responder:

- Registrar contas de utilizadores com um saldo inicial;
- Efetuar login e logout de um utilizador
- Conceder uma opção para editar perfil
- Permitir que um utilizador consulte o seu saldo atual;
- Autorizar um utilizador a visualizar o seu histórico de apostas;
- Efetuar *cash-in* e *cash-out*
- Consultar Jogos (por desporto)
- Escolher aposta
- Gestão de boletim
- Consultar boletim

1.2 Público Alvo

A seguinte tabela apresenta todas as pessoas/empresas de interesse na plataforma a desenvolver.

Nome/ Alvo	Motivação
Comprador	Casa de apostas que pretende comprar a aplicação de maneira a evoluir a sua organização para o nível tecnológico.
Cliente	Investidor que propôs a criação da aplicação por forma a ser vendida posteriormente.
Utilizador	Utilizador com interesse desportivo e com gosto em efetuar apostas.

Table 1: Público alvo

2 Restrições de Arquitetura

2.1 Restrições Técnicas

Restrição	Descrição
Implementação em Python	A implementação da plataforma será feita em Python.
Execução por uma aplicação gráfica	A plataforma irá apresentar uma interface gráfica para o utilizador. Será uma aplicação web, isto é será executada através de um ficheiro html.
Imparcialidade	A aplicação terá que ser suportada nos diferentes sistemas operativos (MacOS, Linux, Windows).

Table 2: Lista de restrições técnicas

De referir ainda, ao contrário do que havia sido mencionado no relatório anterior, a framework a ser usada para o desenvolvimento da aplicação web não será Django mas sim Flask, por uma questão de aprendizagem mais rápida e acessível da mesma.

2.2 Convenções

Convenções	
Documentação da arquitetura	Arquitetura documentada seguindo o template arc42.
Linguagem	A documentação e a aplicação encontram-se em inglês.

Table 3: Convenções

3 Contexto e Foco

3.1 Contexto do Negócio

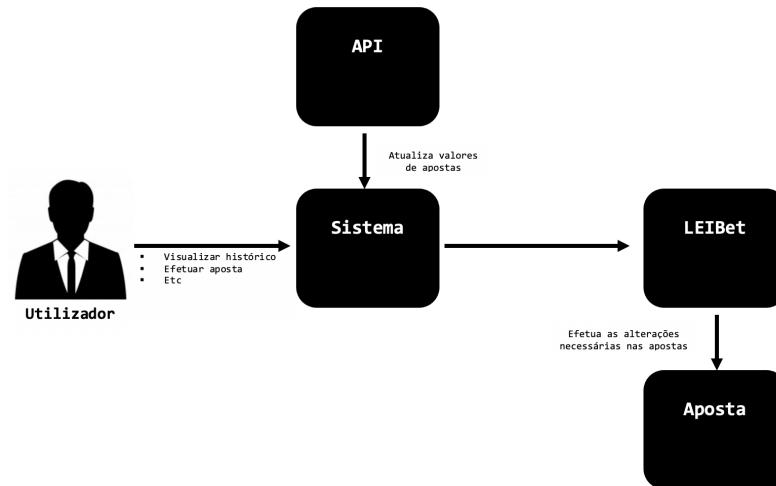


Figure 1: Diagrama de contexto do sistema

3.2 Contexto Técnico

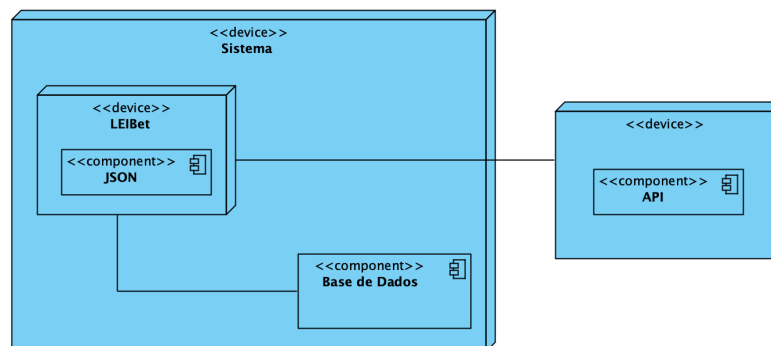


Figure 2: Diagrama de Instalação

Dentro do dispositivo que ocorre a aplicação, ao executar a mesma irá ser estabelecida uma conexão com a base de dados da LEIBet e também com a API fornecida, onde se vai buscar os valores atuais ativos. A API fornecida não se encontra no mesmo dispositivo onde a plataforma se encontrará a correr.

4 Estratégia de Solução

Para este projeto, decidiu-se que seria benéfico a utilização da arquitetura **Cliente-Servidor**, uma vez que se trata do desenvolvimento de uma aplicação Web, na qual o servidor consegue receber e processar pedidos de diversos clientes, visto que, um apostador ao utilizar a plataforma vai realizando operações (apostar, *cash-in*, *cash-out*, ...) e o sistema vai respondendo e apresentando o que é suposto e requerido. Além disso, a este padrão de arquitetura junta-se também uma camada de dados que contém toda a informação necessária que o servidor precisa no processamento dos pedidos. Sendo assim, acaba por se ter três camadas: uma de *frontend*, uma camada intermediária e, por último, uma de *backend*.

Após realizada uma análise mais detalhada e de acordo com o tema do projeto, as ambições e objetivos definidos pela equipa verificou-se que o sistema possui duas entidades principais, Utilizador e Aposta, à volta das quais ocorre o funcionamento do programa.

A entidade Utilizador contém todas as informações do mesmo, a partir do momento do registo, que estarão armazenadas numa base de dados. Estas são alteradas em consequência das operações que o apostador faça e dos resultados das partidas associadas a este nos seus boletins.

A Aposta está diretamente associada a um jogo/partida, recolhido da API que será fornecida, e indica a opção pretendida pelo apostador (e respetiva *odd*). Um boletim contém uma ou mais Apostas e a sua *odd* final é o resultado da multiplicação de todas as outras associadas a cada Aposta pertencente. Multiplicando este valor pela quantia apostada é obtido o montante que o apostador pode ganhar.

5 Building Block View

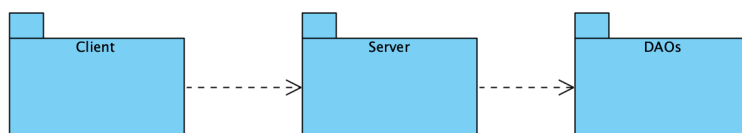


Figure 3:

Como se pode perceber, a figura caracteriza completamente o padrão arquitetural que foi apresentado anteriormente que, como se sabe, está dividido em três: camadas lógicas.

Client - É a camada que contém a parte da interface apresentada ao utilizador. Trata de receber os pedidos dos mesmos e, posteriormente, enviá-los ao server aguardando resposta do mesmo para apresentar ao utilizador.

Server - Esta camada é responsável por todo o funcionamento da aplicação. É nesta camada que o sistema interpreta e processa novas apostas, entre muitas outras ações. No final do processo envia a resposta ao cliente. Sempre que necessário, esta camada irá aceder à camada inferior para

recolher e obter dados.

DAOs - É nesta camada que são guardados todos os dados do programa, desde as informações dos utilizadores como informações de apostas, boletins, etc. É também nesta camada que é feita a ligação à base de dados *rasdb* que torna possível o armazenamento dos dados.

6 Runtime View

6.1 Login

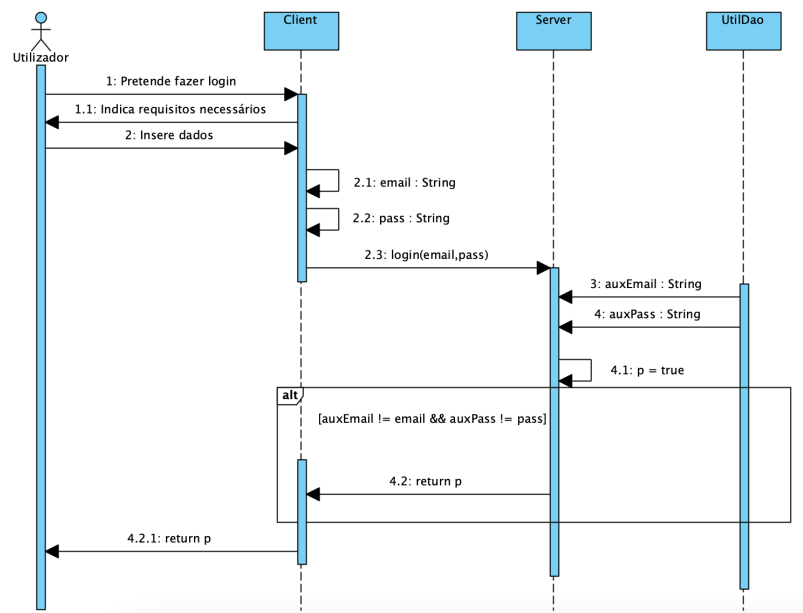


Figure 4: Diagrama de Sequência - Login

Quando um utilizador pretende usufruir da aplicação na sua totalidade e retirar o máximo de proveito das funcionalidades que ela dispõe, deve efetuar login introduzindo para tal o seu email e password com os quais efetuou o registo. Caso estes se encontrem na base de dados e estejam corretos, o utilizador fica corretamente autenticado e passa a ter acesso total. Caso contrário, é informado de que um dos seguintes aconteceu: ou não existem tais dados ou um deles está incorreto; neste caso é informado pelo sistema de que um erro ocorreu e que deve introduzir novamente as suas credenciais.

6.2 Logout

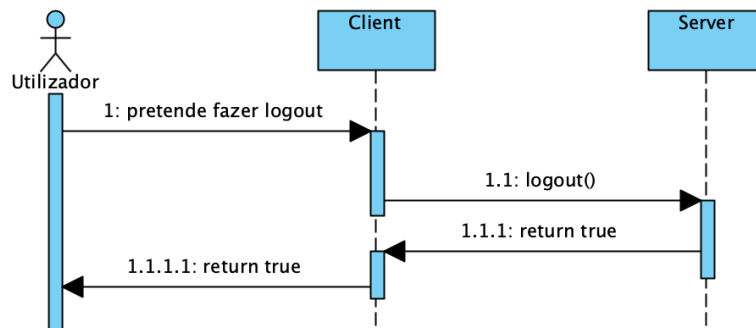


Figure 5: Diagrama de Sequência - Logout

Quando um utilizador pretende sair da aplicação, efetuar logout, informa o sistema de tal e este processa o pedido.

6.3 Registar Utilizador

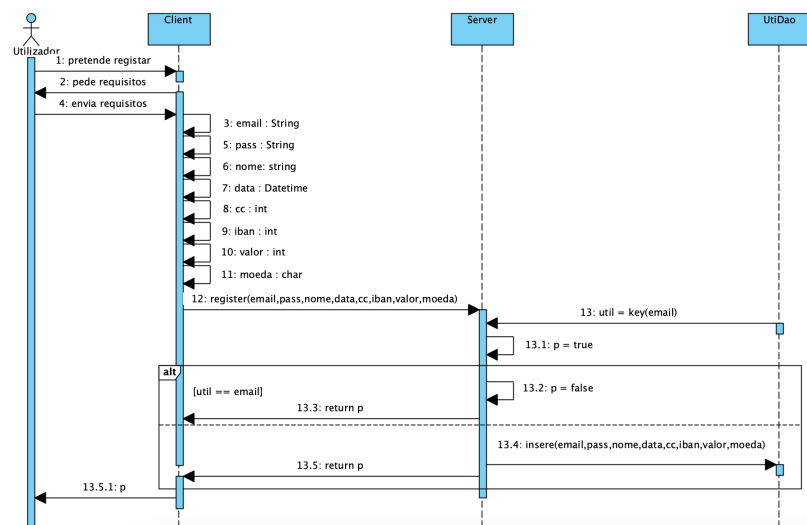


Figure 6: Diagrama de Sequência - Registar utilizador

Qualquer utilizador que pretende utilizar a aplicação na sua totalidade e aceder efetivamente às ferramentas que ela disponibiliza, caso ainda não o tenha feito, deve registar-se na mesma. Para tal introduz, aquando do registo, o seu email, a password que pretende utilizar para posteriormente efetuar login, o seu nome, a sua data de nascimento, dadas as restrições legais existentes para o tipo de aplicação a ser desenvolvido, o número de cartão de cidadão, para verificação da identidade, o

IBAN, para efetuar transações de dinheiro, o valor inicial que pretende creditar na conta e, por fim, a moeda que utiliza (euro,dólar,libra,real,...). Existem alguns parâmetros que devem ser cumpridos para certos atributos e, em caso de falha de tal, o utilizador é informado e deve efetuar novamente o registo completo.

6.4 Editar Perfil

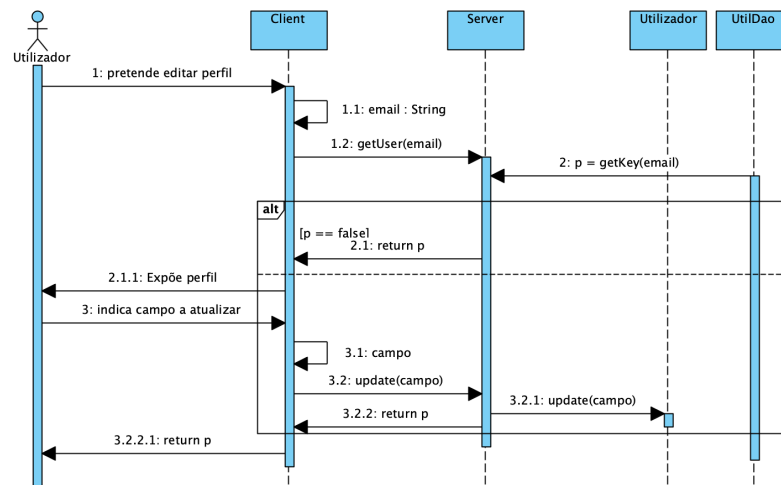


Figure 7: Diagrama de Sequência - Editar perfil

Quando um utilizador pretende alterar dadas informações presentes no seu perfil, deve indicar tal intuito ao sistema e posteriormente indicar que atributo pretende alterar. Dado o atributo a alterar, o utilizador insere o novo valor e, caso cumpra com os parâmetros existentes para esse atributo, o sistema atualiza o mesmo.

6.5 Consultar Saldo

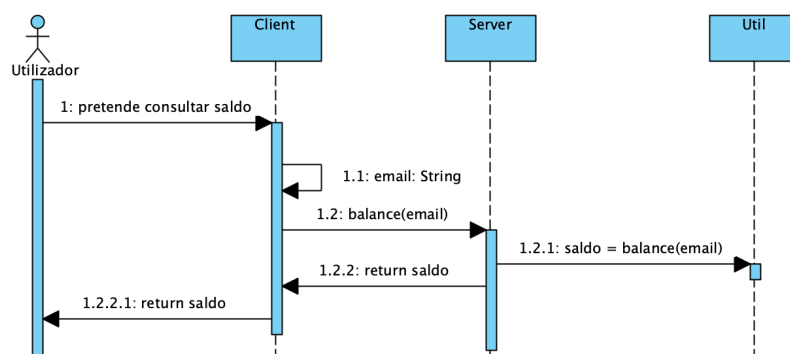


Figure 8: Diagrama de Sequência - Consultar saldo

Quando um utilizador pretende consultar o montante que possui ao seu dispor para efetuar apostas, indica-o ao sistema que após processamento do pedido apresenta ao utilizador o valor requisitado.

6.6 Consultar histórico

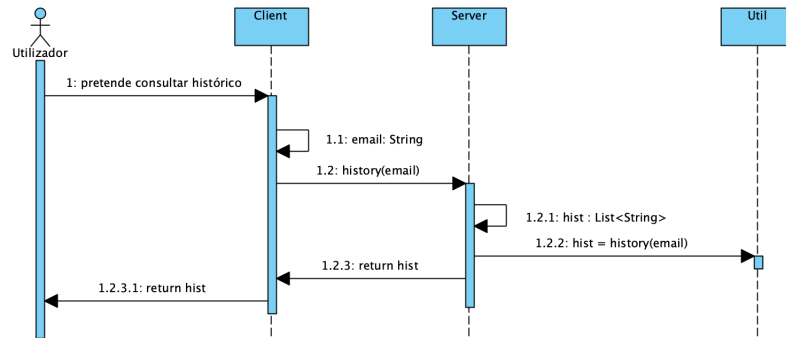


Figure 9: Diagrama de Sequência - Consultar Histórico

Quando um utilizador pretende consultar as apostas que já efetuou na aplicação, o sistema processa o pedido e apresenta a listagem de apostas realizadas pelo utilizador ao mesmo.

6.7 Cash-in

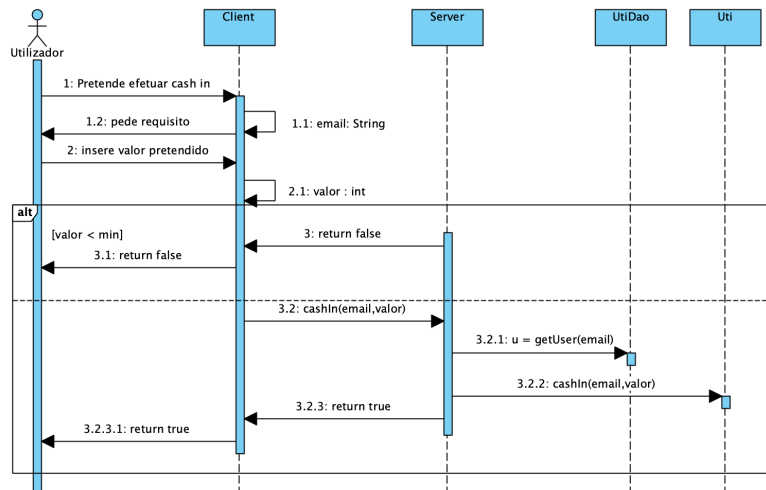


Figure 10: Diagrama de Sequência - Cash-in

Quando um utilizador pretende acrescentar um dado valor monetário à sua conta informa o sistema, que de seguida requisita ao utilizador o valor a acrescentar à sua conta. Caso este seja

inferior ao valor mínimo aceite pela aplicação o utilizador é informado de tal e é lhe pedido novamente para inserir um valor que pretenda adicionar à sua conta.

6.8 Cash-out

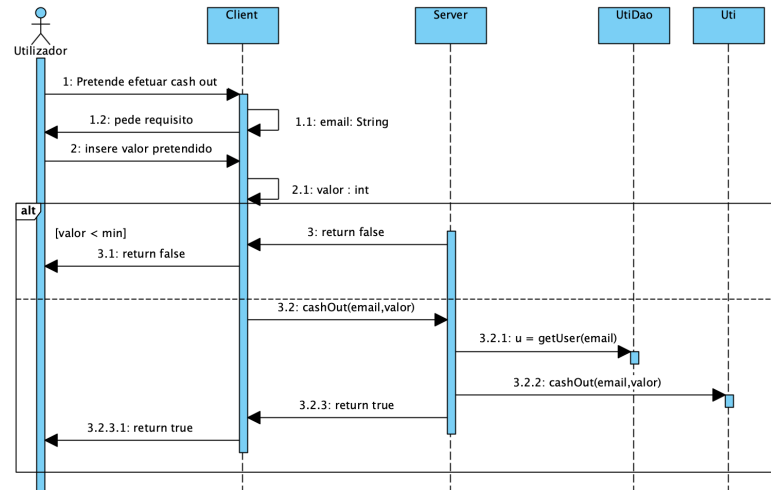


Figure 11: Diagrama de Sequência - Cash-out

Quando um utilizador pretende retirar os seus ganhos da sua conta na aplicação para a sua conta bancária, informa o sistema de tal e este pede ao utilizador o valor que pretende transferir. No caso do valor introduzido ser inferior ao valor mínimo permitido para retirar, o utilizador é informado pelo sistema de tal e deve introduzir um novo valor.

6.9 Consultar Jogos

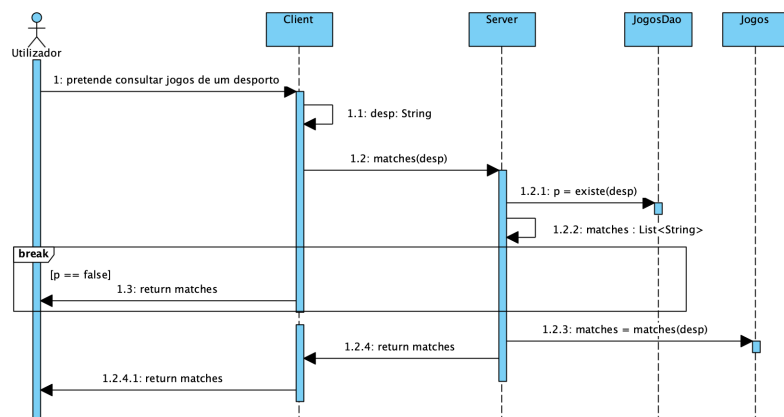


Figure 12: Diagrama de Sequência - Consultar Jogos

Quando um utilizador pretende consultar os jogos nos quais é possível apostar, o sistema processa o pedido e apresenta a listagem ao utilizador.

6.10 Escolher aposta

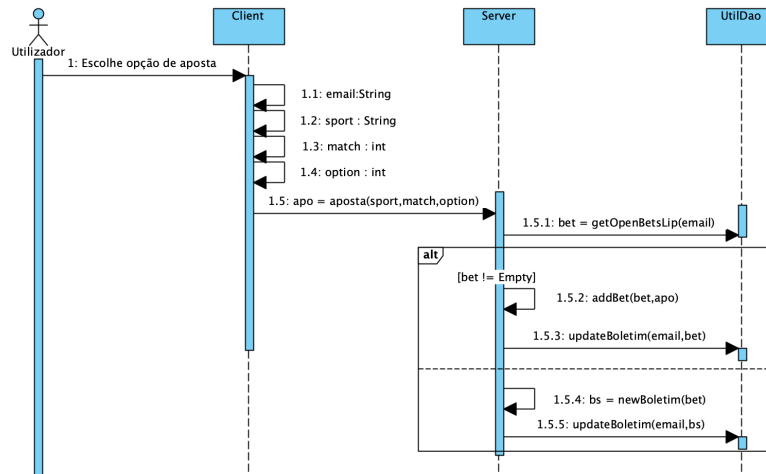


Figure 13: Diagrama de Sequência - Escolher aposta

Quando o utilizador pretende escolher um jogo em que apostar deve primeiramente indicar ao sistema o desporto no qual pretende apostar, o jogo no qual pretende apostar e a opção na qual pretende apostar. Assim que tiver indicado estes requisitos ao sistema, o mesmo regista num boletim, novo caso ainda não exista nenhum criado, a aposta.

6.11 Eliminar aposta

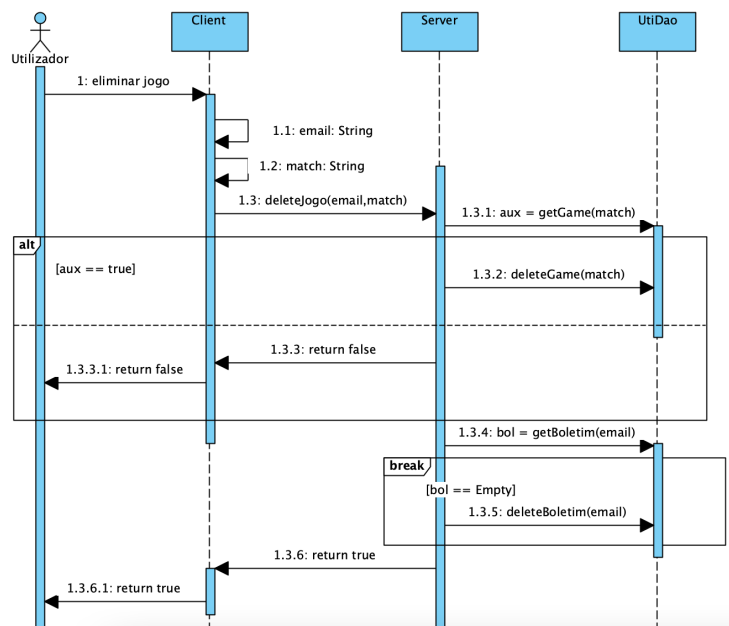


Figure 14: Diagrama de Sequência - Eliminar Aposta

Quando o utilizador pretende remover uma aposta do boletim onde se encontra a inserir as apostas, indica ao sistema o jogo que pretende remover e caso este se encontre no boletim, remove o jogo especificado deste. Caso, ao remover o jogo do boletim, o boletim passe a estar vazio, este é também eliminado pelo sistema.

6.12 Confirmar Boletim

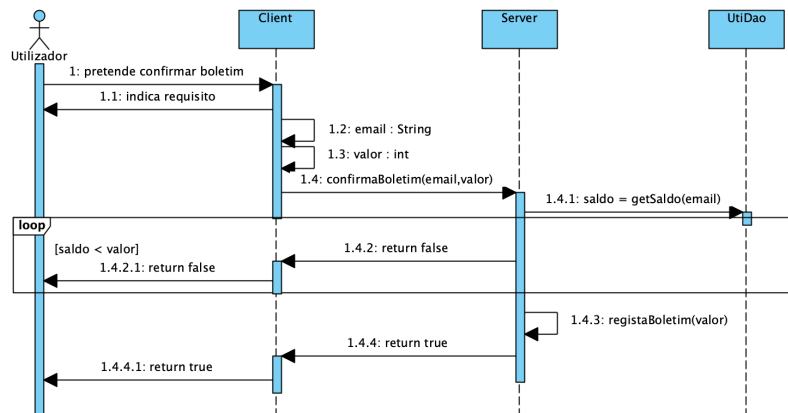


Figure 15: Diagrama de Sequência - Confirmar boletim

Quando um utilizador pretende confirmar um boletim que contém as apostas a realizar, informa o sistema do boletim que pretende confirmar. O sistema verifica se o utilizador contém na sua conta saldo suficiente para realizar tal confirmação e, em caso negativo, é informado do mesmo e a confirmação não é possível, em caso positivo, o boletim é confirmado e adicionado ao histórico de apostas do utilizador.

6.13 Eliminar Boletim

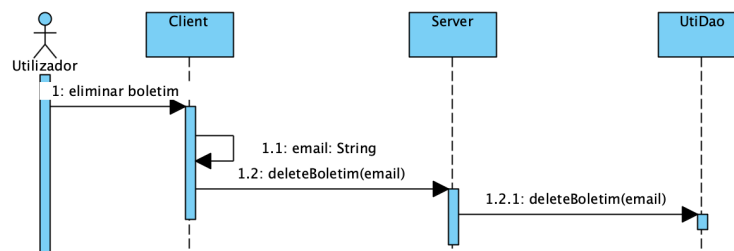


Figure 16: Diagrama de Sequência - Eliminar boletim

Quando um utilizador pretende eliminar um boletim informa o sistema de tal que o remove por completo.

6.14 Consultar boletim

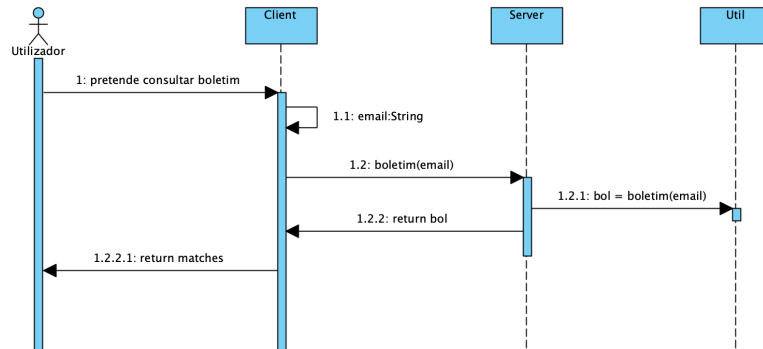


Figure 17: Diagrama de Sequência - Consultar boletim

Quando um utilizador pretende consultar um boletim que contém as apostas que irá realizar informa o sistema de tal que, após processar o pedido, apresenta ao utilizador o mesmo.

7 Deployment View

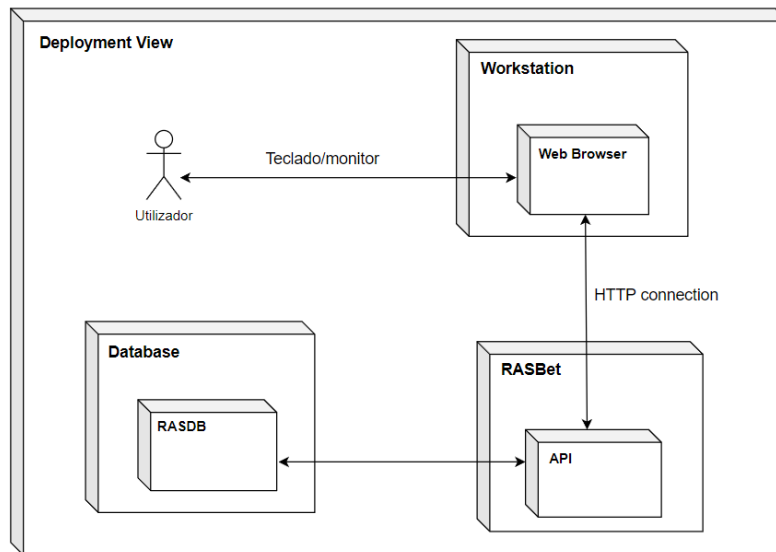


Figure 18: Deployment View

Como podemos observar pela figura acima, o utilizador quando pretende usufruir da aplicação faz uso dos seus periféricos para interagir com a mesma através de uma página web. Todos os pedidos que faça na mesma são enviados para a API da RASBet que tem uma base de dados própria.

8 Conceitos Transversais

No decorrer da construção do projeto serão necessários ter em conta diversos conceitos que estarão de uma forma ou de outra relacionados. É necessário então debruçarmo-nos sobre as soluções encontradas para os mesmos.

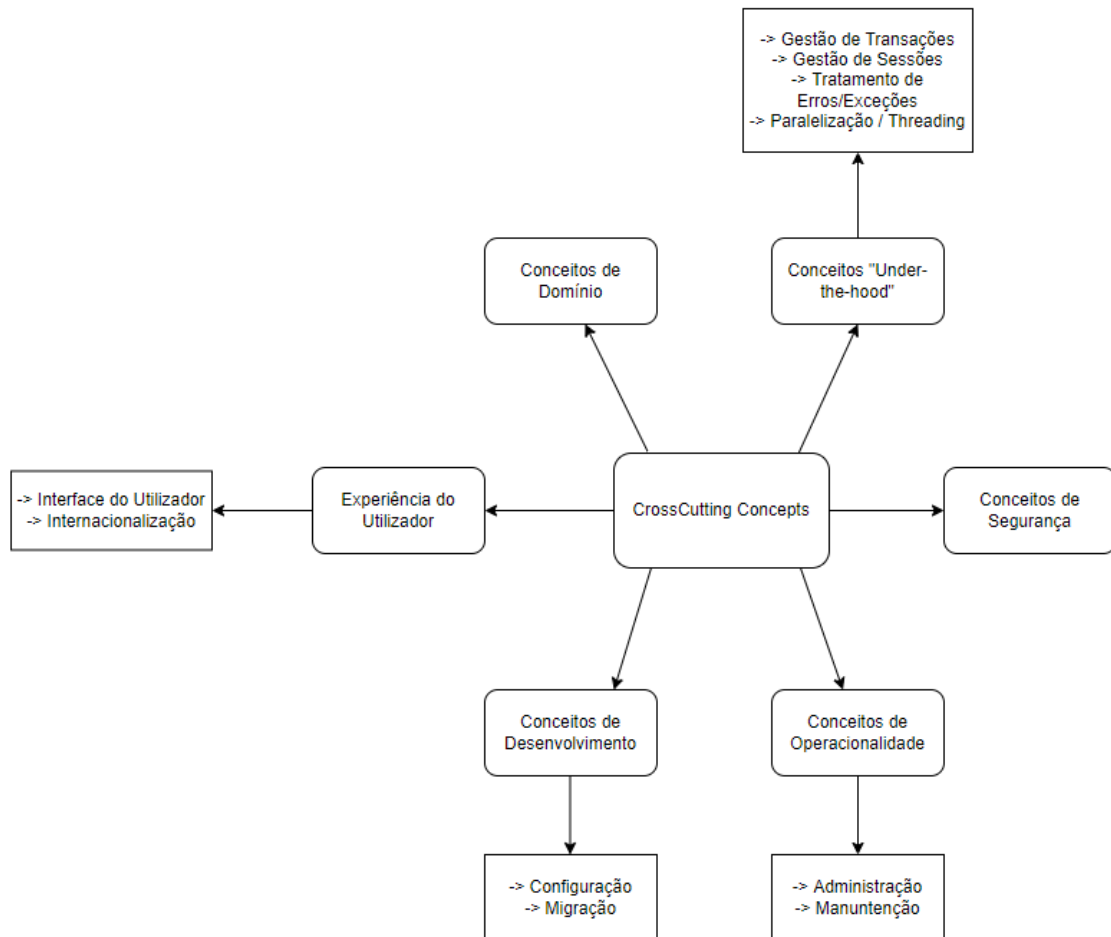


Figure 19: Modelo de CrossCutting Concepts

8.1 Modelo de Domínio

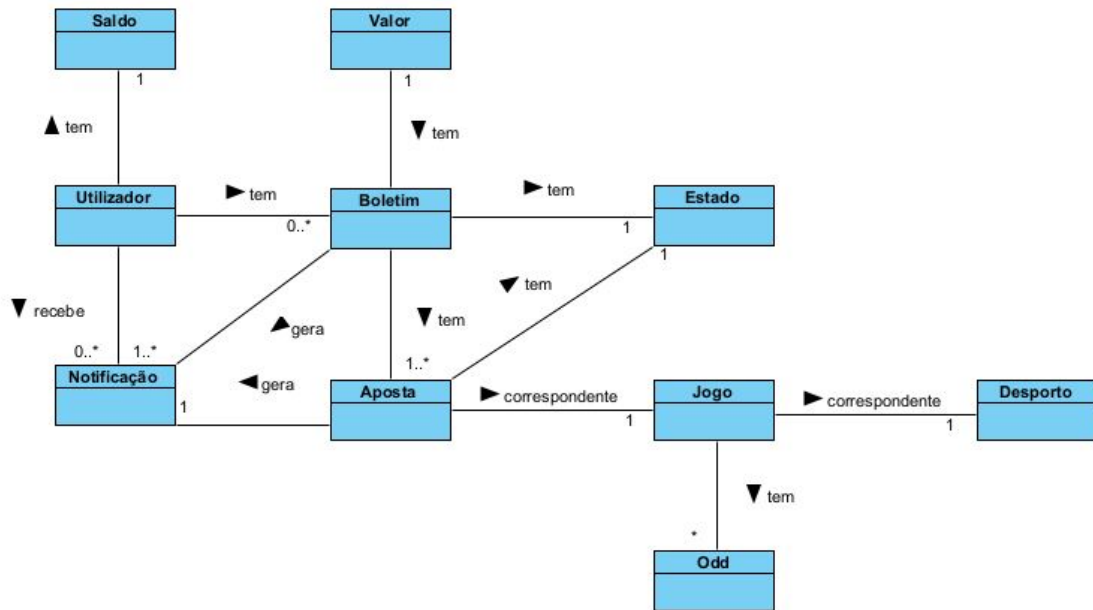


Figure 20: Modelo Domínio

8.2 Experiência do Utilizador

A experiência da utilização da aplicação por parte do utilizador tem fatores deveras influentes e que são importantes ter em conta. De seguida avaliaremos alguns tópicos que consideramos ser os primeiros e maiores impactos que o utilizador tem quando utiliza a aplicação desenvolvida pela *MRMPtech*.

8.2.1 Interface do Utilizador

Com o intuito de manter a simplicidade da aplicação e ao mesmo tempo ser algo apelativo a todo o tipo de utilizador, o projeto a ser desenvolvido deverá apresentar uma interface que proporcione tais condições.

Numa fase inicial a mesma é somente um terminal, algo simples mas não propriamente apelativo aos olhos de um novo utilizador. Assim, o conceito é desenvolver algo já com o intuito de introduzir a aplicação em web/mobile que seja compatível com as ambições da equipa.

8.2.2 Internacionalização

De modo a alcançar o maior número de utilizadores possíveis, podendo assim expandir o negócio a outras casas de apostas, toda a aplicação, como mencionado anteriormente, encontra-se em Inglês, bem como a sua documentação, uma vez que está a língua mais falada a nível mundial.

8.3 Conceitos *Safety* e *Security*

Apesar de parecem os mesmos conceitos, divergem no sentido em que são aplicados. Quando se fala da "safety" da aplicação a ser desenvolvida referimo-nos à fiabilidade de uso da mesma. Já quando falamos da "security", referimo-nos à prevenção de ataques indesejados ou ilegais à aplicação. São conceitos ligeiramente distintos mas intimamente relacionados, uma vez que um leva a outro.

Neste sentido a *MRMPtech* pretende desenvolver uma aplicação consistente com os dois conceitos tendo em conta a sensibilidade do projeto que está a desenvolver, o qual irá trabalhar com informações sensíveis dos utilizadores, bem como valores monetários, independentemente de quais, dos mesmos.

8.4 Conceitos "*Under-the-hood*"

Aliados ao desenvolvimento do projeto em vigor existem certos conceitos que não nos apercebemos do seu método de funcionamento sem olharmos para "debaixo do capô", isto é, se não consultarmos o código por de trás. Assim, passaremos a explicar alguns dos conceitos que achamos relevantes mencionar no contexto da aplicação em desenvolvimento.

8.4.1 Gestão de Transações

Como se trata de um projeto a ser desenvolvido a nível local as transações são feitas diretamente na base de dados, tendo apenas noção do montante que qualquer utilizador possui na sua conta. Algo a implementar num projeto de maior escala seria um protocolo ou a utilização de APIs existentes que permitam transações diretas com os serviços prestados pelos bancos, de forma a permitir ao utilizador fazer um cash-out real e não apenas uma simulação.

8.4.2 Gestão de Sessões

De forma a garantir que a utilização da aplicação é feita de forma correta, é exigido aos utilizadores que se registem e posteriormente façam login na mesma para terem acesso à totalidade da aplicação. Isto permite, futuramente, que quem se encontra a administrar a aplicação tenha noção de quem usufrui da mesma e caso seja necessário averiguar situações mais complicadas como roubos de contas.

8.4.3 Tratamento de Erros e Exceções

Em todas as aplicações existem erros e exceções que devem ser tratados para que a execução e utilização das mesmas sejam fluídas de tal forma que o utilizador não tenha noção de que algo de errado aconteceu ou, noutros casos em que é suposto o utilizador ser alertado, seja evidente tal tratamento.

De forma mais concreta, na aplicação em desenvolvimento não é possível, por exemplo, efetuar um login com informações erradas ou não existentes, sendo o utilizador informado em caso de ocorrência desses erros.

Já após ter efetuado o login, que é o que permite a utilização na totalidade da aplicação, o utilizador não poderá realizar apostas se não tiver escolhido um desporto em que apostar, por exemplo, não poderá apostar um dado montante caso não possua esse mesmo valor creditado na sua conta, entre outros.

8.4.4 Paralelização / Threading

Com o intuito de permitir que múltiplos utilizadores façam uso da aplicação em simultâneo, é necessária uma forma de tratar das sessões de cada um de forma individual. Assim recorrendo ao uso de threads e designando uma para cada cliente, conseguimos garantir tal feito.

8.5 Conceitos de Desenvolvimento

Ao longo do decorrer do desenvolvimento da aplicação a *MRMPtech* teve em conta alguns conceitos que podem influenciar a escolha da sua aplicação face a outras empresas que se encontrem a trabalhar em projetos idênticos.

8.5.1 Configuração

Como todo o projeto se encontra, neste momento, a ser desenvolvido e lançado a nível local a única necessidade de alteração das configurações será a nível de acesso à base de dados. Neste sentido, será apenas necessário alterar as informações base transmitidas à biblioteca de Python, nomeadamente : host, username, password e o nome da base de dados.

8.5.2 Migração

Visto nos encontrarmos a trabalhar com uma base de dados produzida em MySQL, a migração de dados, caso seja necessário, é algo facilmente exequível e de duração relativamente baixa dadas as ferramentas que são já existentes para este efeito.

8.6 Conceitos de Operacionalidade

Ambicionando sempre manter a total e funcional operacionalidade da aplicação, a *MRMPtech* investiu algum do seu tempo em conceitos que facilitem tal processo.

8.6.1 Administração

Conciliando também com a gestão de sessões da aplicação, a administração da mesma permite a quem a controla manter a sua operacionalidade e potencial correção de imprevistos.

8.6.2 Manutenção

Contando que seja necessário realizar manutenções regulares, todo o desenvolvimento da aplicação é feito de modo a que aquilo que é preciso alterar ou corrigir seja facilmente identificável e de fácil e rápido acesso, desde funcionalidades da mesma à informação armazenada na base de dados.

9 Decisões Arquitetónicas

Foi decidido implementar a base de dados do projeto em MySQL, devido ao seu elevado desempenho e segurança, pela sua compatibilidade com os diversos sistemas operacionais e também por todos os elementos da equipa estarem bastante familiarizados a trabalhar com esta tecnologia e com a respetiva linguagem.

Como já foi referido anteriormente, o nosso sistema estará assente numa arquitetura Cliente-Servidor, podendo diversos utilizadores estarem a usufruir da aplicação de forma simultânea, o que leva a inúmeros acessos à memória e, para isso, contar-se-á com o auxílio de DAO's, como ligação à base de dados, ou seja na execução de comandos SQL para realizar as devidas operações. O uso deste tipo de "objetos" garante a persistência e a consistência dos dados pois faz com que a camada de dados e a lógica de negócios estejam claramente separados. Resumindo, estes servirão de ponte entre a base de dados e o sistema em si.

10 Requisitos de Qualidade

Depois de uma análise detalhada dos requisitos gerais, chegou-se à conclusão que o sistema teria que possuir os seguintes atributos de qualidade.

10.1 Facilidade de Utilização

O sistema deve ser simples de forma a que a sua utilização seja intuitiva e rápida.

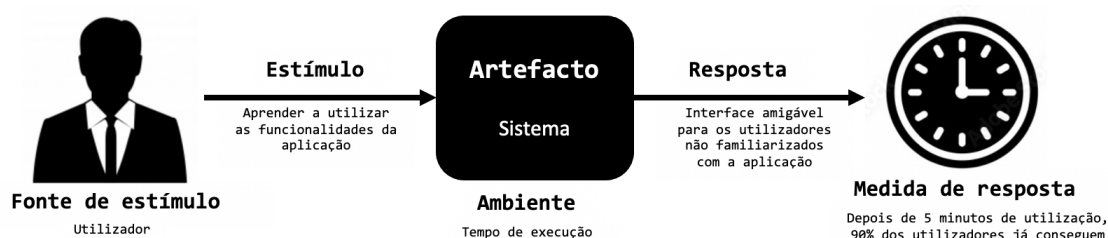


Figure 21: Cenário concreto de Utilização

A utilização do sistema deve ser fácil e direta, sendo que o único requisito básico necessário para os utilizadores é conter os conhecimentos básicos de manuseamento de computadores/ internet.

10.2 Segurança

O sistema deve proporcionar segurança dos dados pessoais de cada utilizador.

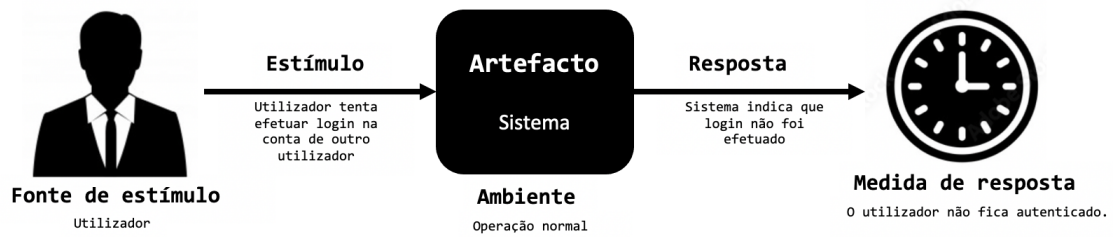


Figure 22: Cenário concreto de Segurança

O sistema através da autenticação, protege dados pessoais. Assim, só um utilizador autenticado é capaz de usufruir das funcionalidades dos sistema.

11 Riscos e Dívida Técnica

Em todos os projetos existem riscos e probabilidades de falhas nos mais diversos aspetos do mesmo e é portanto necessário apontá-los de forma a conseguir prevenir os mesmos num futuro próximo.

11.1 Riscos de Segurança

A segurança é um padrão indispensável em variados parâmetros como na prevenção da exposição de dados (da aplicação e dos seus utilizadores), no controlo da aplicação e na sua manipulação. No ponto 10.2 demonstramos parte do nosso esquema de segurança.

11.2 Riscos de Sobrecarregamento

Um dos pontos de foco é também a falta de capacidade do sistema lidar com o "*stress*" causado pela quantidade de dados a movimentar e armazenar, ou até mesmo com a quantidade de utilizadores online na aplicação em simultâneo. Assim, temos de conseguir reunir as condições necessárias para fazer essa gestão da melhor forma possível, nunca comprometendo nenhum aspeto da aplicação, especialmente uma que envolva a segurança ou em qualquer aspeto da usabilidade do utilizador.

11.3 Riscos de Obsolescência

Neste sentido, temos de ser capazes de tornar o nosso sistema o mais atrativo e dinâmico possível, de maneira a que não se torne apenas mais uma aplicação de apostas. A inovação constante e um bom *customer service* são duas características muito importantes que fazem com que os utilizadores queiram e gostem de recorrer à nossa aplicação.

11.4 Riscos Legais

Uma aplicação da área das apostas envolve, entre outros, o movimento de grandes quantias de dinheiro e a gestão e proteção de dados dos utilizadores, o que requer um cuidado enorme em termos de segurança, como referido anteriormente. Assim sendo, qualquer passo em falso no sentido legal pode levar a graves problemas não só para a empresa responsável como para os utilizadores. Isto leva a que seja necessário uma preocupação bastante grande com este departamento, recorrendo, se necessário, a ajuda especializada.

12 Glossário

Termo	Definição
Simple Bet	Aposta num único jogo com um valor de retorno resultante da multiplicação do valor apostado com a odd do evento.
Multiple Bet	Aposta em dois ou mais jogos com um valor de retorno resultante da multiplicação do valor apostado com a multiplicação das odds de todos os eventos no qual o utilizador decidiu apostar. Neste tipo de aposta o utilizador só é considerado vencedor e recebe o valor calculado caso acerte em todos os resultados.
Odd	Cotações dadas a um determinado evento. Probabilidades de algum evento ocorrer.
Flask	Framework de desenvolvimento de aplicações Web em Python.
API	Conjunto de definições e protocolos usados no desenvolvimento de uma aplicação.
Local Database	Base de dados inserida e mantida no sistema local.
MySQL	Serviço de gerenciamento da base de dados que faz uso da linguagem SQL como interface.
RASBet	Sistema de suporte a apostas desportivas.