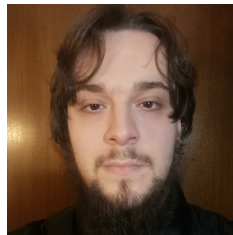
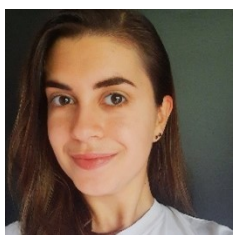


UMinho
Mestrado em Engenharia Informática
Requisitos e Arquiteturas de Software

Entrega 3

PG46983	Ana Luísa Lira Tomé Carneiro
PG46988	Ana Rita Abreu Peixoto
PG47194	Francisco José Vilão Matos Queiróz Peixoto
A89582	Henrique Manuel Ferreira da Silva Guimarães Ribeiro
PG47428	Luís Miguel Lopes Pinto



Janeiro de 2022

2021/2022

Novo Requisito

Implementação do novo requisito

O novo requisito permite que os apostadores possam ter uma carteira com diferentes saldos para cada moeda, tais como Euros, Dólares americanos, criptomoeda Cardano, entre outras. Podem ainda ser adicionadas ou removidas moedas pelo gestor possibilitando que os diferentes apostadores tenham acesso às diversas moedas. Adicionalmente, é possível associar um imposto ao câmbio entre as moedas consideradas, permitindo ao serviço RASBet ter lucro com os câmbios e permitindo aos apostadores conseguir depositar ou levantar saldo entre as moedas. Deste modo, cada utilizador possui diferentes carteiras associadas a uma dada moeda e cada uma delas possui um saldo específico. O programa permite que haja transferência monetária de carteira para carteira, cobrando uma taxa para o serviço RasBet.

Incorporação na Solução

Alteração no Diagrama Classes

Para a implementação do requisito demonstrado acima houve necessidade de alterar o diagrama de classes de forma a acrescentar duas novas classes: **Moeda** e **Saldo**. A classe Moeda representa as diversas moedas disponíveis pelo sistema e é composta pelo nome da moeda (exemplo Euro), *token* que representa o símbolo da moeda (€), *ratio* que indica a diferença dessa moeda para o euro e o imposto que indica o valor que o apostador paga para fazer o cambio. Desta forma a taxa de câmbio de uma moeda é dada por $ratio * (1 - imposto) * ratio$. O saldo vai representar o conjunto de saldos nas várias moedas a que o utilizador tem acesso e é composta pelo uma string que identifica o apostador associada a esses saldos e um *map* de moeda para um valor *float*, associando desta forma os saldos e as respectivas moedas. Na figura abaixo podemos ver o diagrama de classes onde a laranja encontram se as novas classes assim como as relações entre elas e as entidades já existentes.

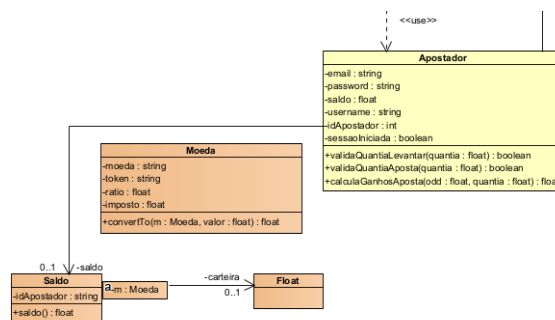


Figura 1: Adições ao diagrama de classes

Alteração no Modelo Lógico

Para a implementação da nova funcionalidade de suporte a diferentes carteiras, foi necessário adicionar algumas tabelas ao modelo lógico: Carteira e Moeda. Desta forma o apostador pode possuir várias carteiras, desde que sejam suportadas pela aplicação RASBet, cada uma com uma moeda associada. As novas tabelas criadas permitem armazenar de forma persistente os dados que suportam o novo requisito. Na figura seguinte podemos observar parte do modelo lógico onde estão presentes as novas tabelas e as suas relações com as outras entidades do programa:

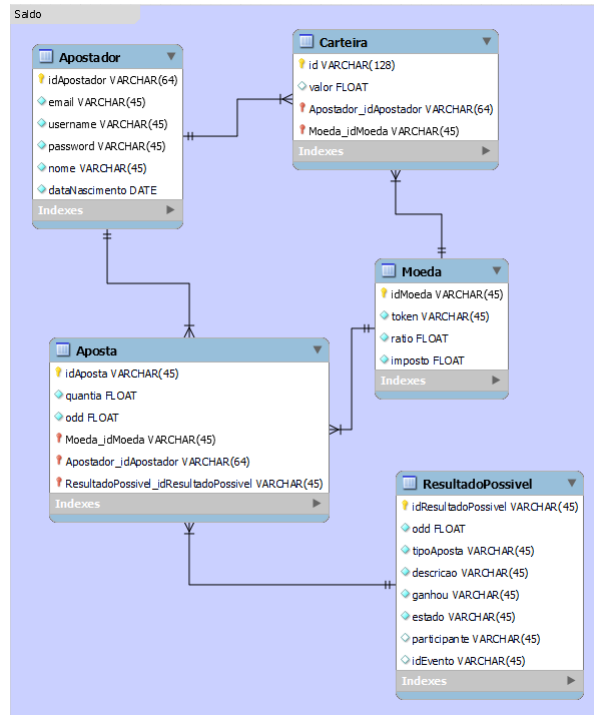


Figura 2: Adições ao modelo lógico

Conclusões finais

Chegando ao final da terceira fase do trabalho, consideramos conveniente efetuar uma reflexão crítica sobre o trabalho efetuado e em concreto sobre a adição do novo requisito. O processo de planeamento do trabalho (fases 1 e 2) foi efetuado com vista a não restringir o programa quanto à implementação de novas funcionalidades e também com o objetivo de encaminhar o desenvolvimento da aplicação. Deste modo, nesta terceira fase do trabalho, a implementação do programa recaiu em grande parte nos diagramas efetuados e nos requisitos levantados. Além disso, dado que no planeamento efetuado não existiam restrições à adição de novas funcionalidades, então houve abertura para a adição de uma carteira multi-moeda sem que isso se refletisse em mudanças drásticas no sistema. Adicionalmente, a funcionalidade descrita já haveria sido tida em consideração na 1ª fase do trabalho, no levantamento de requisitos. Tal como apresentado acima, as mudanças para o suporte deste novo requisito focaram-se essencialmente na criação de novas classes que levaram à criação de novas tabelas na base de dados.