

**Relatório 1º Projeto**

**Algoritmos e Estruturas de Dados**

**Biblioteca de Jogos**

Turma 4- Grupo 9

João Paulo Gomes Torres Abelha- [up201706412@fe.up.pt](mailto:up201706412@fe.up.pt)

João Rafael Gomes Varela -[up201706072@fe.up.pt](mailto:up201706072@fe.up.pt)

Vítor Hugo Pereira Barbosa -[up201703591@fe.up.pt](mailto:up201703591@fe.up.pt)

25-11-2018

# Índice

Descrição do tema do trabalho	2
Solução implementada	3
Classes implementadas	4
Diagramas UML	6
Lista de Casos de Utilização	7
Dificuldades encontradas no decorrer do trabalho	9
Contribuição dos diferentes membros na evolução do trabalho	10

## Tema do Trabalho

O objetivo do trabalho desenvolvido “Biblioteca de Jogos”, ao longo da primeira metade do semestre, foi fazer a gestão eficiente de várias bibliotecas de jogos, analisando, da melhor forma, várias estatísticas daqui provenientes.

Cada utilizador possui uma biblioteca à qual pode adicionar títulos, *home* ou *online*, caso tenha um cartão de crédito válido (com data válida e com dinheiro suficiente). Um título pode existir em enumeras plataformas, e o seu preço pode variar. Apenas os títulos *online* são usados na formação de rankings, que nos dão, não só, informações acerca da preferência dos utilizadores, mas também da sua rentabilidade. Para além disse são construídas estatísticas gerais acerca da biblioteca, como por exemplo o cálculo do seu preço médio de construção.

## Solução Implementada

Com o intuito de criar uma situação o mais realista possível foi desenvolvido um algoritmo que permite fazer a gestão de vários utilizadores e as suas bibliotecas, guardando todos os utilizadores num ficheiro e extraíndo sempre estes, havendo a possibilidade de acrescentar mais ao sistema (criação de conta). Para além disso, permite acrescentar títulos ao sistema que poderão ser comprados posteriormente pelos utilizadores (informação guardada num ficheiro e processada automaticamente quando o programa corre ).

Deste modo, sempre que é criado um novo utilizador, este vai começar a fazer parte do sistema, podendo adquirir títulos, juntando à sua biblioteca, inicialmente vazia, guardar os seus cartões de crédito e toda a sua informação no sistema, imperativo para a formação de rankings e estudos estatísticos. É ainda possível fazer diferentes listagens conforme o critério de ordenação pretendido.

Quanto à biblioteca de cada usuário, desde que ele tenha um cartão de crédito e este esteja dentro da data de validade e tenha saldo suficiente, podemos adicionar a ela um título para uma determinada plataforma, sendo que é possível atualizar a sua data de validade do cartão de crédito. Os preços de cada título podem ser modificados, sendo que o preço mais elevado corresponde ao primeiro, preço de lançamento. Cada título poderá ser do tipo *home* ou *online* que mediante a sua natureza vai ser gerido de diferente forma. Para a sua implementação são usados os conceitos de puro virtual e de polimorfismo.

Cada título tem a si associadas informações que serão, então, usadas no sistema para que ao ter acesso a todos os utilizadores permita recolher as informações para que depois de processadas e organizadas possam ser dispostas ao usuário diferentes estatísticas, tendências e preferências dos utilizadores.

## Classes Implementadas

### Sistema

A classe Sistema é responsável pelo armazenamento das informações do sistema que armazena todos os títulos disponíveis para compra, que são guardados num ficheiro, e a informação relativa a todos os utilizadores, isto é, as bibliotecas correspondentes. Cabe à classe processar a informação de diferentes formas, mediante aquilo que o utilizador pede. Para além disso, é responsável por ler a informação de ficheiros que contêm toda a informação do sistema, bem como voltar a guardá-la, funcionando ainda como uma charneira entre as outras classes.

### Biblioteca

A classe Biblioteca é responsável por armazenar todos os títulos de um utilizador, adicionando-os, caso seja possível, à sua biblioteca.

### Banco

A classe Banco é usada com o intuito de armazenar os diferentes cartões de crédito dos utilizadores, garantindo que são válidos, isto é, que estão dentro da validade, sendo a data atual calculada, e garantindo que não há nenhum cartão de crédito repetido (com o mesmo identificador).

### Data

A classe Data corresponde a uma data, no formato DD/MM/AAAA, que serve para atribuir uma data de validade aos cartões, bem como permitir a atualização destes. Serve ainda para atribuir uma data de lançamento a um título, e ainda datas de atualizações para os títulos *home* e guardar um histórico de datas em que um utilizador jogou um jogo *online*.

### Erro

A classe Erro corresponde a um tipo de exceção lançada no decorrer do programa, caracterizando o tipo de erro que foi lançado.

### Título

A classe Título é uma classe virtual pura que herda duas subclasses *home* e *online*, conforme o tipo de título adquirido. Esta classe armazena toda a informação referente a título, essencial para que se consigam construir estatísticas e rankings relativamente a estes.

## Home

A classe Home, derivada de Título, armazena as informações de um título *home*, que se destaca por possuir datas de atualizações, e um preço fixo de atualização de um euro.

## Online

A classe Online, derivada da classe Título, destaca-se por possuir um preço de subscrição, que pode ser fixa ou variável, um conjunto de datas em que o jogador jogou esse título e as horas que jogou esse título.

## Utilizador

Esta classe tem como objetivo guardar toda a informação relativa a um utilizador, sendo de destacar a biblioteca de jogos que possui. Para além de um nome, idade, morada e email, também estará a ele associado um ou mais cartões de crédito que poderá usar para a compra de títulos numa determinada plataforma.

## **Diagrama UML**

Por uma questão de legibilidade, foi optado por mandar o diagrama de UML, em anexo.

## Casos de Utilização

O usuário do programa, após ter sido recolhida a informação relativa ao sistema que guarda as informações dos utilizadores já existentes, depara-se com um menu principais que lhe dá oito opções diferentes. As primeiras três opções são referentes aos utilizadores, as seguintes três relativas a um título, jogo, e ainda a última que permite sair do programa, guardando a informação de todos os utilizadores.

Na primeira opção, será possível visualizar os jogos que pertencem à biblioteca. Após o utilizador escolher esta opção, será aberto um menu secundário que permite mostrar no ecrã os títulos *online* e *home*. Será ainda possível visualizá-los segundo um critério crescente ou decrescente de data de lançamento, de idade mínima para poder jogar o jogo e ainda uma ordenação alfabética crescente e decrescente da empresa fundadora do jogo.

Na segunda opção, o usuário pode adicionar um utilizador ao sistema, fornecendo informação relativa a ele que será guardada, caso esta seja válida.

Na terceira opção é possível pesquisar utilizadores. Para esta opção é aberto um outro menu que permite ver os utilizadores do sistema. O usuário poderá optar pela forma que quer ver a listagens de utilizadores, se pela sua idade crescente ou decrescente, se por ordem crescente ou decrescente alfabética e ainda de forma crescente ou decrescente do número de títulos por biblioteca.

Será possível manipular informação relativa a ele, sendo aberto de novo um menu, que permite adicionar um título, um cartão de crédito, verificar o seu saldo, ordenar os seus títulos segundo determinadas opções semelhantes à opção dois. Permite ainda jogar um título que tenha, facto que será relevante para a construção de estatísticas.

Na quarta opção, permite adicionar um jogo à biblioteca, que irá corresponder a um lançamento de um jogo que os utilizadores poderão posteriormente comprar. Será nesta estabelecida o preço de lançamento e a plataforma em que é lançada.

Na quinta opção, é possível procurar um utilizador, podendo adicionar um título, um cartão de crédito, consultar o saldo do utilizador, jogar um determinado título online durante durante um determinado intervalo de tempo e verificar o número de horas que esse utilizador jogou títulos *online*.

Na sexta opção é possível pesquisar um determinado jogo, fornecendo o nome e plataforma. Posteriormente, será possível estabelecer um novo preço, inferior ao de lançamento, adicionar uma atualização caso seja um título *home*. Caso seja *online* é fornecida uma lista de datas em que foi jogado e o número total de horas jogadas.

Na sétima opção, são construídas e mostradas as estatísticas referentes aos utilizadores, nomeadamente o número médio de títulos por biblioteca, o preço médio de construção numa biblioteca, a plataforma preferida e género preferido pelos utilizadores (ranking de géneros). É ainda feito um ranking de rentabilidade que mostra os títulos mais rentáveis do sistema. Será ainda possível criar grupos etários que contabilizam o número de utilizadores para um determinado intervalo de idades.

Cada menu, apresenta ainda uma opção “sair” que serve para retroceder para o menu anterior.

A oitava e última opção permite sair do programa.



Com estas operações, a gestão de várias bibliotecas de jogos é possível ser feita de maneira bastante fácil, intuitiva e eficaz.

## **Dificuldades encontrar no desenvolvimento do projeto**

No decorrer do projeto, não houve grande dificuldade na aplicação dos diferentes conceitos aprendidos nas aulas.

No entanto, as principais dificuldades encontradas foram: na gestão de tempo, organização de todas as classes que permitisse um uso eficiente e correto destas e ainda houve alguma dificuldade na partição do trabalho em partes igualitárias.

## **Contribuição dos diferentes membros na evolução do trabalho**

Todos os membros do grupo contribuíram e participaram na construção de todas as classes de forma equilibrada e igualitária. O relatório foi realizado maioritariamente pelo elemento João Abelha, o doxygen maioritariamente por Vítor Barbosa e o UML, maioritariamente, por Rafael Varela, contudo todos participaram na sua elaboração e revisão.