

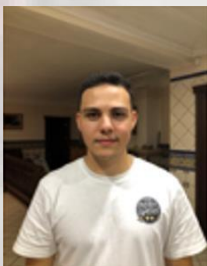
Desenvolvimento de Sistemas de Software

- Trabalho Prático -

- Jogo Formula 1 Manager (Fase 2) -

Universidade do Minho
Licenciatura em Engenharia Informática

Trabalho realizado pelo grupo 14 composto por:



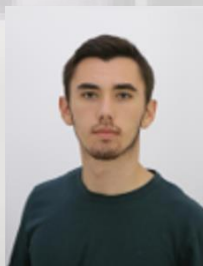
- A95442 -

André Oliveira Gonçalves



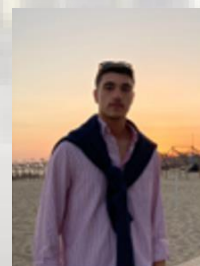
- A95460 -

Diogo Luís Almeida Costa



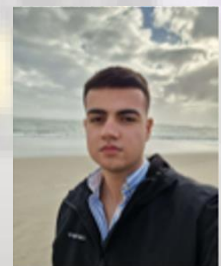
- A95323 -

Henrique Ribeiro Fernandes



- A96726 -

Ivo Miguel Alves Ribeiro



- A95641 -

João Pedro Moreira Brito



Índice

Objetivos desta fase	3
Modificações efetuadas em relação à segunda fase	3
Descrição do trabalho	4
1.Utilizadores	4
2.Carros	4
3.Campeonatos	5
4.Data	5
5.UI	5
Conclusão	6

Repositório do trabalho: <https://github.com/jpmbrito123/DSS>



Objetivos desta fase

Para esta terceira e última fase do projeto foi-nos proposto a entrega dos modelos necessários à descrição da implementação do sistema e a implementação do mesmo.

Modificações efetuadas em relação à segunda fase

Relativamente à segunda fase, alterou-se todos os diagramas de sequência que tínhamos apresentado, pois estes tinham todos representados a interface com o utilizador e numa aula de comentários à fase 2, o docente referiu que isto era visto como um aspeto negativo, assim, refizemos os diagramas tendo em conta a Lógica de negócio.

Também se atualizou o diagrama de classes feito na segunda fase, para que este consiga refletir a implementação do sistema com os DAOs e esteja de acordo com a especificação feita nas fases anteriores.



Descrição do trabalho

De forma a tornar mais explícita a nossa apresentação sobre a implementação do sistema iremos dividir a descrição do trabalho por partes, cada parte representa uma porção do código essencial para o funcionamento do sistema.

1.Utilizadores

Na parte dos utilizadores possuímos a classe Utilizador, Admin, UtilizadorFacede e IUtilizadorFacede.

- Utilizador: classe que possui as variáveis referentes ao utilizador e métodos getters.
- Admin: classe que estende a classe Utilizador e possui a variável de identificação dos admins.
- IUtilizadorFacede: classe que descreve a interface que é implementada no UtilizadorFacede, possui métodos relacionados com os pilotos, carros, campeonatos e circuitos.
- UtilizadorFacede: classe facade que implementa os métodos definidos na interface IUtilizadorFacede.

2.Carros

A parte dos carros é a parte com mais classes pois nós possuímos uma classe para cada tipo de carro.

- Carro: classe abstrata que tem as variáveis necessárias para todos os carros, como o identificador, o modelo, a marca, a cilindrada, a potência, pac e fiabilidade.
- C1, C1H, GT, SC, GTH, C2, C2H: classes que indicam as categorias existentes no sistema, cada classe possui os métodos que definem os atributos para cada categoria e o método que calcula a probabilidade de um carro se despistar. A diferença das classes com “H”, relativamente às sem é que estas referem-se a um carro híbrido e possuem um método para calcular a potência do motor elétrico.



3. Campeonatos

Esta parte engloba as corridas, os circuitos e a configuração dos jogadores.

- **Campeonato:** classe que possui as variáveis necessárias para um campeonato, que são o nome, uma HashMap com as classificações, um contador das corridas realizadas, uma lista de configurações de jogadores e uma lista de corridas. Esta classe é responsável pelos métodos para atualizar e ordenar as classificações.
- **Corrida:** classe que tem as variáveis necessárias para uma corrida, como um identificador, o clima, a lista de jogadores etc. Esta classe possui um dos métodos mais importantes do sistema que é o `simuladorCorrida`, responsável por simular uma corrida tendo em consideração todos os fatores referidos no enunciado.
- **Circuito:** classe que possui as variáveis precisas para um circuito, como nome, número de chicanes, retas e curvas, a distância do circuito, o número de voltas, uma lista com as dificuldades das retas, outra lista com as dificuldades das curvas etc.
- **PlayerSet:** classe essencial para as configurações do jogador, possui as variáveis como identificador, `agressividadeMotriz`, piloto, carro, pneu e um contador para as alterações do jogador. Esta classe possui os métodos que calculam o desempenho e se um corredor se encontra despidado.

4. Data

Esta parte engloba as classes DAO utilizadas para interligar o nosso programa com a base de dados. Estas classes são: `CarrosDAO`, `CircuitoDAO`, `DAOconfig`, `PilotosDAO`, `PlayerSetDAO`, `UtilizadorDAO`.

5. UI

Esta parte possui todas as classes necessárias para definir a interface gráfica do sistema. Estas classes são: `CampeonatoUI`, `Menu`, `PlayerUI`, `TextUI`.



Conclusão

Em suma, este trabalho permitiu-nos consolidar os conceitos dados no âmbito da cadeira de Desenvolvimento de Sistemas de Software, e ganhar uma maior noção sobre a necessidade de possuir um bom planeamento e modelação para a realização de uma aplicação.

Com esta fase, o grupo reparou que certos parâmetros e certos diagramas realizados e definidos nas fases anteriores poderiam ser melhorados, ou até feitos de outras formas mais simples e claras, contudo seria contraproducente alterar esses diagramas agora, pois estaríamos a desenvolver o projeto em forma “cascata”, e não seria o suposto para este projeto.