



Universidade do Minho
Escola de Engenharia

Relatório N°1



<https://github.com/joaoabreu5/DSS-Grupo40>

Desenvolvimento de Sistemas de Software

João Abreu

João Faria

Ricardo Sousa

Rui Silva

Tiago Ferreira



A91755

A97652

A96141

A97133

A97141

Grupo 40

2022/23

Índice

Introdução	2
Objetivos	2
Modelo de Domínio.....	3
Vista do Piloto.....	3
Vista do Campeonato	3
Vista do Carro	3
Vista do Jogador	4
Use Cases	4
Anexos.....	10
Modelo de Domínio.....	10
Diagrama de Use Cases	11
Conclusão e Análise Crítica.....	11

Introdução

Na Unidade Curricular de Desenvolvimento de Sistemas de Software foi-nos proposta a conceção de um simulador de campeonatos de automobilismo. A génese por detrás do sistema pedido é similar ao já conhecido jogo *F1 Manager*.

É esperado que a aplicação seja capaz de simular um campeonato automobilístico, onde um administrador tem o poder de criar campeonatos, que irão ser constituídos por um número variável de circuitos, também posteriormente criados e associados ao campeonato pelo próprio administrador. O administrador do campeonato deverá, ainda, ter a possibilidade de criar novos carros e pilotos, que após a adição ao jogo ficarão disponíveis para a sua seleção por parte dos respetivos jogadores.

Os jogadores deverão ser capazes de configurar cada corrida do campeonato que anteriormente selecionaram, e simular as mesmas. No final de cada corrida, os jogadores que estiverem autenticados no sistema receberão pontos consoante as respetivas classificações ao longo do campeonato, sendo que todos estes pontos registados serão reunidos num ranking do *Racing Manager*.

Nesta fase do trabalho prático foi pedido o desenvolvimento de um modelo de domínio do caso de estudo e, ainda, o desenvolvimento de um diagrama de *Use Cases* (diagrama e respetivas especificações) que corresponderá às funcionalidades da aplicação, bem como as nossas decisões na elaboração deste modelo e diagrama, respetivamente. De referir ainda, que a conceção e desenvolvimento desta aplicação segue uma abordagem baseada em modelos (suportada por UML).

Objetivos

Durante esta primeira fase de desenvolvimento deste trabalho prático estabelecemos como objetivos a atingir os seguintes pontos:

- Aprimorar o nosso conhecimento no que diz respeito ao desenvolvimento de um modelo de domínio;
- Apresentar um conjunto completo de use cases para a aplicação, sem nunca quebrar o que são os limites delineados para a mesma.
- Consequentemente, terminar esta fase com um modelo que dê um bom suporte às fases de desenvolvimento seguintes e que garanta o funcionamento pretendido para a nossa aplicação;

Modelo de Domínio

O nosso Modelo de Domínio inclui todas as entidades que acreditamos serem necessárias para o nosso projeto, nomeadamente: Carro, Jogador, Circuito, Campeonato, etc. As nossas entidades encontram-se divididas de forma a conseguirmos planear melhor a maneira como pretendemos que o nosso programa funcione e, também, tendo em conta os passos futuros do desenvolvimento deste projeto.

Para a realização deste modelo, decidimos dividi-lo em quatro vistas: vista do piloto, vista do carro, vista do jogador e vista do campeonato.

Vista do Piloto

As entidades presentes nesta vista são: Piloto, Nome Piloto, Nível Perícia, CTS e SVA.

Muito à semelhança do que nos é descrito no enunciado, o Piloto tem um nome de piloto e um nível de perícia que, por sua vez, é definido pelos valores de SVA e CTS do mesmo.

Vista do Campeonato

As entidades presentes nesta vista são: Campeonato, Nome Campeonato, Circuito, Nome Circuito, Comprimento, Zona, Chicane, Curva, Reta, GDU, Difícil, Possível, Impossível, Corrida, Volta, Condição Meteorológica, Sol, Chuva.

Deste modo, o campeonato tem um nome e é constituído por vários circuitos e várias corridas. O circuito, por sua vez, tem também um nome e um comprimento, pode estar presente em vários campeonatos, é constituído por várias zonas e permite que se realizem nele várias corridas. A zona pertence apenas a um circuito, pode ser uma chicane, uma curva ou uma reta e possui um grau de dificuldade de ultrapassagem (GDU) que pode ser classificado como possível, difícil ou impossível. Por fim, a corrida pertence a apenas um campeonato e a um circuito, tem várias voltas e é caracterizada por uma condição meteorológica, que pode ser de sol ou chuva.

Vista do Carro

A vista do Carro está associada às entidades: Pneu, Pneu Macio, Pneu Duro, Pneu de Chuva, Categoria, C1, C2, GT, SC, Fiabilidade, Cilindrada, Afinação, PAC, Motor Combustão, Híbrido, Motor Elétrico, Potência, Modo Motor, Agressivo, Normal, Conservador, Marca e Modelo.

Um carro possui uma marca, um modelo, um motor de combustão com uma certa potência, e pneus, que podem ser duros, macios ou pneus de chuva.

Estão divididos em 4 grandes categorias, e temos abertura a que mais possam vir a ser adicionadas no futuro. Todas as categorias impõem limites de cilindrada ao motor e cada uma apresenta também uma fiabilidade distinta. Para além disso, existem duas categorias cujos seus carros podem sofrer alterações na sua afinação (C1 e C2). Ainda, os carros pertencentes às categorias C1, C2 e GT podem ser híbridos, ou seja, podem dispor também de motor elétrico, responsável por um aumento da potência dos carros em questão.

Vista do Jogador

Esta vista possui as entidades Administrador, Pontuação Global, Campeonato

Um jogador, podendo ou não ser administrador, possui uma pontuação global.

Modelo final (junção das várias vistas)

As várias vistas do modelo unem-se devido a duas situações: quando um administrador adiciona carros, pilotos, campeonatos ou circuitos ao sistema ou quando um jogador se regista num campeonato.

No que diz respeito à primeira situação, cada administrador pode criar vários carros, pilotos, campeonatos ou circuitos, sendo cada um destes objetos criado apenas por um administrador.

Quanto à segunda situação, um ou mais jogadores podem registar-se num campeonato, sendo esta ação caracterizada por uma classe de associação, o Registo. O Registo, por sua vez, possui um piloto e um carro associados, sendo que estes podem estar presentes em vários registos.

Use Cases

Use Case: Autenticar utilizador

Descrição: Utilizador autentica-se no sistema

Pré-condição: *True*

Pós-condição: Utilizador fica autenticado

Fluxo Normal:

1. Utilizador apresenta *username*
2. Utilizador apresenta palavra-passe
3. Sistema permite o acesso e autentica o utilizador

Fluxo de Exceção (1): [Palavra-passe incorreta e número de tentativas erradas menor que 3]
(passo 3)

- 3.1. Sistema avisa que a palavra-passe introduzida não é correta

3.2. Regressa a 2.

Fluxo de Exceção (2): [Palavra-passa incorreta e número de tentativas erradas igual a 3] (passo 3)

3.1. Sistema avisa que a palavra-passe introduzida não é correta

3.2. Sistema termina o processo

Use Case: Registar campeonato

Descrição: Administrador regista um novo campeonato

Cenário: Cenário 1 - Campeonatos

Pré-condição: Administrador está autenticado

Pós-condição: Campeonato fica disponível para jogar

Fluxo Normal:

1. Ator/Administrador indica o nome do campeonato
2. Sistema verifica que o nome do campeonato está disponível
3. Administrador aceita o nome do campeonato
4. Administrador escolhe os circuitos que compõem o campeonato
5. Administrador adiciona o campeonato à lista de campeonatos disponíveis
6. Sistema regista o campeonato

Fluxo de Exceção (1): [Nome do campeonato já se encontra utilizado] (passo 2)

2.1. Sistema indica que o nome do campeonato introduzido já se encontra utilizado

2.2. Regressa a 1.

Use Case: Registar circuito

Descrição: Administrador regista um circuito

Cenário: Cenário 2 - Circuitos

Pré-condição: Administrador está autenticado

Pós-condição: Circuito fica disponível para jogar

Fluxo Normal:

1. Ator/Administrador indica o nome do circuito
2. Sistema verifica que o nome do circuito é único
3. Sistema aceita o nome do circuito
4. Administrador indica a distância do circuito, número de curvas e chicanes
5. Sistema apresenta a lista de curvas e retas
6. Administrador indica o GDU de cada curva e de cada reta
7. Administrador indica o número de voltas

8. Sistema verifica que o número de voltas introduzido é válido
9. Sistema regista o circuito

Fluxo de Exceção (1): [Nome do circuito já se encontra utilizado] (passo 2)

- 2.1. Sistema indica que o nome do circuito já está a ser utilizado
- 2.2. Regressa a 1.

Fluxo de Exceção (2): [Número de voltas não é válido] (passo 8)

- 8.1. Sistema indica que o número de voltas introduzidas é inválido
- 8.2. Regressa a 7.

Use Case: Registar Carro

Descrição: Administrador regista um novo carro

Cenário: Cenário 3 - Carros

Pré-condição: O Administrador está autenticado

Pós-condição: Carro fica disponível para jogar

Fluxo Normal:

1. Sistema apresenta as categorias disponíveis
2. Ator/Administrador indica categoria, marca, modelo, cilindrada e potência
3. Sistema verifica que a cilindrada indicada respeita os limites de cilindrada da categoria
4. Sistema verifica que a categoria é C1, logo pode ser híbrido e necessita de uma fiabilidade
5. Ator indica fiabilidade
6. Sistema verifica que a fiabilidade é de aproximadamente 95%
7. Ator indica que o carro não é híbrido
8. Ator indica PAC
9. Sistema regista o carro

Fluxo Alternativo (1) [Carro é da categoria C2] (passo 4)

- 4.1. Sistema verifica que o carro é da categoria C2, logo o veículo pode ser híbrido e não necessita de uma fiabilidade
- 4.2. Regressa a 7.

Fluxo Alternativo (2) [Carro é da categoria SC] (passo 4)

- 4.1. Sistema verifica que o carro é da categoria SC, logo o veículo não pode ser híbrido e não necessita de uma fiabilidade
- 4.2. Regressa a 9.

Fluxo Alternativo (3) [Carro é da categoria GT] (passo 4)

4.1. Sistema verifica que o carro é categoria GT, logo o veículo pode ser híbrido e não necessita de fiabilidade

4.2. Ator indica que o carro não é híbrido

4.3. Regressa a 9.

Fluxo Alternativo (4) [Carro da categoria GT é híbrido] (passo 4.2)

4.2.1. O Ator indica que o carro é híbrido

4.2.2. O Ator indica a potência do motor elétrico

4.2.3. Regressa 9.

Fluxo Alternativo (5) [Carro é híbrido] (passo 7)

7.1. O Ator indica que o carro é híbrido

7.2. O Ator indica a potência do motor elétrico

7.3. Regressa a 8.

Fluxo de Exceção (1) [Cilindrada fornecida não se respeita os limites da categoria] (passo 2)

2.1. O sistema informa que a cilindrada fornecida não respeita os limites da categoria

2.2. Regressa a 2.

Fluxo de Exceção (2) [Fiabilidade não é aproximadamente 95%] (passo 6)

6.1. O sistema verifica que a fiabilidade não é de aproximadamente 95%

6.2. Regressa a 5.

Fluxo de Exceção (3) [Valor de PAC não se encontra entre 0 e 1] (passo 8)

8.1. O sistema informa que o valor de PAC não é valido

8.2. Regressa a 8.

Use Case: Registar piloto

Descrição: Administrador adiciona um piloto

Cenário: Cenário 4 - Pilotos

Pré-condição: Administrador está autenticado

Pós-condição: Piloto fica disponível para jogar

Fluxo Normal:

1. Ator/Administrador indica o nome do piloto
2. Sistema aceita o nome do piloto
3. Administrador indica o valor do SVA
4. Sistema aceita o valor de SVA
5. Administrador indica o valor de CTS
6. Sistema aceita o valor de CTS
7. Sistema calcula os níveis de perícia

8. Sistema regista o piloto**Fluxo de Exceção (1):** [Nome do piloto não é válido] (passo 2)**2.1.** Sistema indica que o nome do piloto já está a ser utilizado**2.2.** Regressa a 1.**Fluxo de Exceção (2):** [Valor de SVA não se encontra entre 0 e 1] (passo 4)**4.1.** Sistema indica que o valor do SVA não se encontra entre 0 e 1**4.2.** Regressa a 3.**Fluxo de Exceção (3):** [Valor de CTS não se encontra entre 0 e 1] (passo 5)**5.1.** Sistema indica que o valor de CTS não se encontra entre**5.2.** Regressa a 5.**Use Case:** Configurar campeonato**Descrição:** O Francisco e três amigos resolvem jogar um campeonato**Cenário:** Cenário 5 - Configurar Campeonato**Pré-condição:** *True***Pós-condição:** Campeonato fica configurado**Fluxo Normal:**

1. Jogador escolhe um campeonato da lista de campeonatos
2. Sistema apresenta os circuitos que pertencem a esse campeonato
3. Jogador escolhe um carro da lista de carros
4. Jogador escolhe um piloto da lista de pilotos
5. O Sistema configura o campeonato

Use Case: Configurar Corrida**Descrição:** Jogador prepara o carro para a corrida**Cenário:** Cenário 5 - Configurar Corrida**Pré-condição:** Campeonato configurado**Pós-condição:** Corrida fica pronta a começar**Fluxo Normal:**

1. Sistema apresenta o circuito e a situação meteorológica
2. Sistema verifica que é possível alterar PAC e escolher modo de motor por se tratar de um carro da categoria C1 ou C2
3. Jogador indica um novo valor do PAC para o carro
4. Sistema aceita o novo valor do PAC para o carro
5. Jogador escolhe um novo modo do motor

6. Jogador escolhe os pneus a utilizar
7. Sistema termina o processo de configuração de corrida

Fluxo Alternativo (1): [Não é possível alterar PAC nem escolher modo de motor por não se tratar de um carro da categoria C1 ou C2] (passo 2)

- 2.1. Sistema verifica que o PAC do carro não pode ser alterado e que não é possível escolher o modo de motor
- 2.2. Regressa a 6.

Fluxo Alternativo (2): [Jogador não pretende alterar o PAC do carro] (passo 3)

- 3.1. Jogador não pretende alterar o valor do PAC do carro
- 3.2. Regressa a 5.

Fluxo de Exceção (1): [Valor de PAC não se encontra entre 0 e 1] (passo 4)

- 4.1. Sistema não aceita valor do PAC do carro, visto que não se encontra entre 0 e 1
- 4.2. Regressa a 3.

Use Case: Simular corrida

Descrição: *Racing Manager* simula a corrida

Cenário: Cenário 5 – Jogar (Simular Corrida)

Pré-condição: Corrida configurada

Pós condição: Simulação da corrida terminada

Fluxo Normal:

1. Sistema distribui os carros, ao longo da grelha de partida, agrupados pelas respetivas categorias
2. Sistema inicia a corrida
3. Sistema indica para cada curva, reta e chicane de cada volta, eventuais ultrapassagens, despistes e avarias.
4. Sistema indica as posições dos carros/pilotos/jogadores no final de cada volta
5. Sistema termina a corrida
6. Sistema apresenta os resultados da corrida
7. Sistema apresenta os pontos de cada jogador
8. Sistema regista e apresenta os pontos de cada jogador no respetivo ranking do campeonato, consoante categoria automobilística escolhida.

Use Case: Registar Pontos Globais

Descrição: Os pontos são registados no ranking global do *Racing Manager*

Cenário: Cenário 5 – Jogar (Resultado Final)

Diagrama de Use Cases

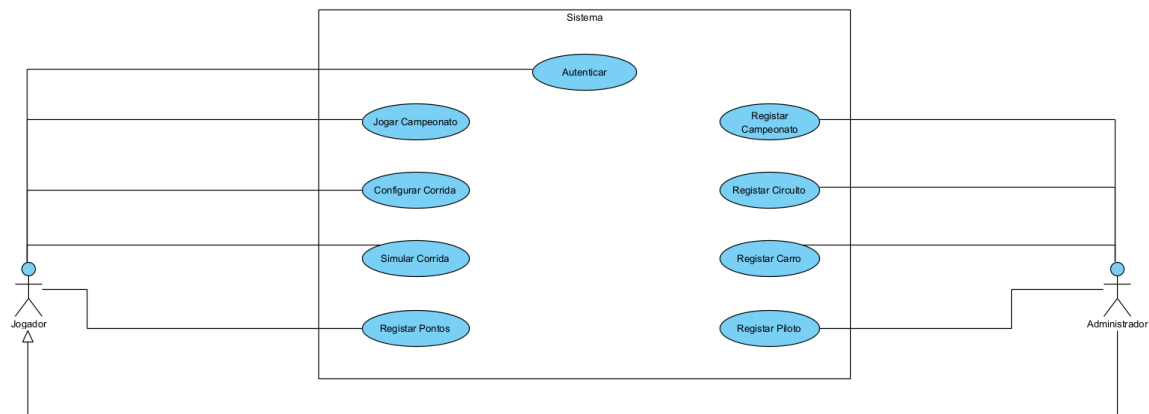


Figura 2 - Diagrama de Use Cases

Conclusão e Análise Crítica

Todo o desenvolvimento desta fase do trabalho prático foi encarado com serenidade, procurando dar uma boa resposta a todos os pontos elaborados neste relatório de forma a atingir um produto final do nosso agrado. Claramente exigiu muito tempo e tornou-se também árduo concluir o processo de aprimoramento das definições que nos eram pedidas nesta etapa do projeto, resultado de muitos momentos de discussão de parâmetros e dinâmicas do nosso Modelo de Domínio e *Use Cases*. Consideramos que a nosso desenho atual de análise de requisitos satisfaz as condições propostas e possui o que precisa para uma possível boa continuação da evolução do projeto.

Por fim, para trabalho futuro, sabemos que se segue então a fase de “Conceção da Solução”, na qual iremos elaborar uma arquitetura conceptual do sistema, capaz de suportar os requisitos identificados, bem como apresentar os modelos comportamentais necessários para descrever o comportamento pretendido para o sistema. Seguimos confiantes no nosso trabalho desenvolvido até então, e otimistas com o progresso que temos pela frente.