



Universidade do Minho
Escola de Engenharia

Desenvolvimento de Sistemas de Software

Trabalho Prático - Fase 1

Grupo 22

2022/2023



A93258 - Bernardo Garcia de Freitas Lima



A91671 - João Manuel Novais da Silva



A91660 - Pedro António Pires Correia
Leite Sequeira



A91697 - Luís Filipe Fernandes Vilas



A91677 - Vicente de Carvalho Castro

Índice

Conteúdo

1. Introdução	3
2. Modelo de Domínio.....	3
3. Modelo de use cases	6
3.1. Criação do campeonato.....	7
3.2. Criação de Circuito	8
3.3. Criação de Carro	9
3.4. Criação de pilotos	12
3.5. Configurar Campeonato	12
3.6. Configurar e simular corrida	13
3.7. Ver Ranking	14
4. Conclusão	15
5. Referências	16

1. Introdução

O principal motivo deste trabalho é encontrar uma solução para desenvolver o jogo Racing Manager. Foi solicitada a criação de um sistema que permitisse a criação de campeonatos, corridas, carros e pilotos, através de um utilizador Admin. Da mesma forma existe um utilizador Jogador que dá ênfase à utilização do jogo, ou seja, a simulação da corrida em si. Começou-se a primeira fase a criar um protótipo de funcionamento, construindo o diagrama de domínio, o diagrama de Use Case e os próprios Use Cases.

2. Modelo de Domínio (figura 1)

Depois de uma longa discussão sobre os diferentes cenários apresentados na sala de aula, decidimos formar o modelo de domínio com base no sistema. Este pode ser acedido por 3 tipos de utilizadores, 2 autenticados (**Administrador** e **Jogador**) e um não autenticado (**Convidado**). Destes apenas o **administrador** tem permissão para gerir os 4 elementos principais do sistema, estes são:

O **campeonato**, que poderá apenas ser configurado por jogadores e gerido pelo administrador.

Os **circuitos**, que são agrupados pelos diversos campeonatos já referidos e que incorporam diversos atributos, chicanes, curvas e restas, todas estas com um grau de dificuldade (GDU), situação meteorológica (chuva ou tempo seco) e um número de voltas variável.

Os **carros**, que podem pertencer a diferentes categorias (**C1,C2,SC,GT**), todas elas com limites de cilindrada distintos, para além de diferenças no número de motores (combustão ou híbrido), estes tem potências e a possibilidade de 3 modos distintos de operação (conservador, normal, agressivo) que influenciam a performance do carro tal como a probabilidade de este avariar com mais ou menos frequência. Para além destas diferenças, estas distintas categorias também delimitam a fiabilidade base do veículo, sendo que a categoria **GT**, por exemplo, depende unicamente da cilindrada e do número de voltas que o veículo já percorreu, ao contrário do **C2**, que já depende também da capacidade de quem o conduz.

Estes veículos também tem a opção de utilizar 3 tipos de pneus (de chuva, macios ou duros) que influenciam a condução em situações climáticas diferentes e também nas velocidades possíveis de alcançar, apresentando todos níveis de desgaste diferentes.

Relativamente ao chassi do carro, este tem um perfil aerodinâmico (PAC) e poderá ser otimizado anteriormente a uma corrida em termos de *downforce*.

E por último, os **pilotos**, que possuem duas características, o temperamento (segurança versus agressividade – SVA), que influencia nas decisões de ultrapassagem dentro de uma corrida, como também a preferência meteorológica de cada um (CTS) que poderá dar a vantagem a pilotos sobre as diferentes situações meteorológicas que possam enfrentar.

Já para o **jogador**, este apenas poderá interagir com os elementos geridos pelo administrador, tendo acesso a um ranking no sistema e neste serão colocadas as pontuações individuais de cada jogador.

É importante referir que todas as categorias de carros podem competir simultaneamente num campeonato, contudo, os jogadores serão classificados, no final do campeonato, em tabelas diferentes por categoria e com pontuações diferentes, especialmente se estes utilizarem veículos híbridos. Para além disso tem também permissão para configurar campeonatos.

Por fim, o **convidado** tem a possibilidade de interagir com os mesmos elementos que o jogador, mas limitado à entrada em campeonatos já configurados pelo jogador e sem direito a ranking do sistema.

3. Modelo de use cases

Neste capítulo iremos apresentar o modelo de use cases (figura 2) que achamos que melhor representa o funcionamento do sistema que gere a criação de corridas, campeonatos, etc. e as interações que o administrador e o utilizador podem ter com o sistema.

Com este sistema interagem 2 atores o administrador e o utilizador. O administrador está responsável pela criação dos campeonatos, carros, pilotos e circuitos para depois o utilizador poder interagir com os mesmos. Estas ações só podem ser realizadas depois de fazer-se uma autenticação de que é realmente o administrador que está a realizá-las. Para além disso o administrador só pode criar campeonatos se já existirem circuitos criados. O utilizador pode configurar campeonatos, ou seja, jogar apenas se os carros, pilotos, circuitos e campeonatos já estiverem criados. Pode ainda ver o seu ranking, mas apenas se for um utilizador com conta criada no sistema, assim terá que haver uma autenticação de que foi feito o login. Depois de ser criado o campeonato o jogador pode ainda simular e configurar uma corrida.

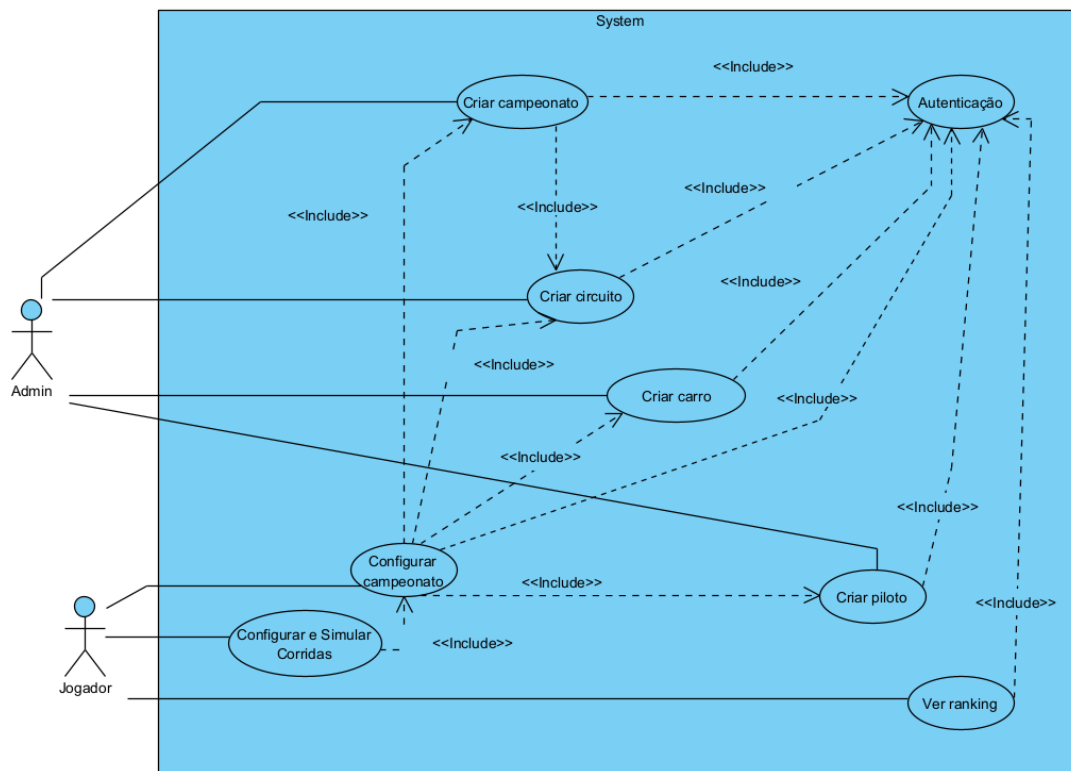


Figura 2 - Diagrama de Use Cases

3.1. Criação do campeonato

Ator: Administrador.

Descrição: Um administrador cria um novo campeonato.

Cenários: O João cria um novo campeonato com o nome "Campeonato do João".

Pré-condição: O utilizador autentica-se com conta de administrador já existem circuitos criados no sistema.

Pós-condição: Um campeonato com o nome desejado foi criado, com circuitos associados e o sistema foi atualizado.

Fluxo normal:

1. Administrador introduz nome do campeonato
2. Administrador indica os circuitos a adicionar
3. Sistema atualiza

Fluxo de exceção (1): [Já existe um campeonato com o mesmo nome] (passo 1)

- 1.1 Sistema informa administrador que já existe um campeonato com o mesmo nome
- 1.2 Regressa a 1

3.2. Criação de Circuito

Ator: Administrador.

Descrição: Um administrador cria um novo circuito.

Cenários: O Joaquim cria um circuito diurno chamado "Quimdelícia".

Pré-condição: O utilizador autentica-se com conta de administrador

Pós-condição: Um circuito com o nome e especificações pretendidas é criado e o sistema é atualizado.

Fluxo normal:

1. Administrador introduz tamanho, número de curvas e número de chicanes do circuito
2. Administrador associa GDU a cada curva e reta do circuito
3. Administrador introduz número de voltas
4. Administrador insere o estado meteorológico pretendido
5. Sistema grava e atualiza

Fluxo de exceção(1): [Tamanho do circuito \leq 0 ou algum dos parâmetros nulos] (Passo 1)

- 1.1 Sistema avisa falha na introdução dos valores
- 1.2 Regressa a 1

Fluxo de exceção(2): [Número de voltas \leq 0 ou parâmetro nulo] (Passo 3)

- 3.1 Sistema avisa falha no número de voltas
- 3.2 Regressa a 3

3.3. Criação de Carro

Ator: Administrador.

Descrição: O Administrador cria um carro.

Cenários: O Luís cria um carro a combustão de classe C1 chamado "LuisGT 69" com 6000cm³ de cilindrada e 900cv e com um PAC de 0.1.

Pré-condição: O utilizador autentica-se com conta de administrador.

Pós-condição: O carro com os critérios pretendidos é criado e o sistema atualizado.

Fluxo normal:

1. Jogador escolhe categoria ,marca, modelo ,cilindrada e potência
2. Sistema verifica que é um carro C1, logo pode ser híbrido e necessita de uma fiabilidade
3. Jogador indica fiabilidade
4. Sistema verifica que fiabilidade é aproximadamente 95%
5. Jogador indica que carro não é híbrido
6. Jogador indica PAC
7. Sistema regista carro

Fluxo Alternativo(1): [Carro é C1 híbrido] (Passo 2)

- 2.1 Jogador indica que carro é híbrido
- 2.2 Jogador indica potência do motor elétrico
- 2.3 Regressa a 6

Fluxo Alternativo(2): [Carro é SC] (Passo 2)

- 2.1 Sistema verifica que é um carro SC, logo apenas pode ser a combustão
- 2.2 Sistema calcula os 25% de fiabilidade em função da cilindrada do carro (inserida no passo 1)

2.2 Regressa a 6

Fluxo Alternativo(3): [Carro é C2] (Passo 2)

2.1 Sistema verifica que é um carro C2, logo pode ser híbrido e necessita de uma fiabilidade

2.2 Jogador indica fiabilidade

2.3 Sistema verifica a cilindrada do carro e verifica que é de aproximadamente 80%

2.5 Jogador indica que carro não é híbrido

2.6 Regressa a 6

Fluxo Alternativo(4): [Carro é C2 híbrido] (Passo 2)

2.1 Sistema verifica que é um carro C2, logo pode ser híbrido e necessita de uma fiabilidade

2.2 Jogador indica fiabilidade

2.3 Sistema verifica a cilindrada do carro e verifica que a fiabilidade é de aproximadamente 80%

2.5 Jogador indica que carro é híbrido

2.6 Jogador indica a potência do motor elétrico

2.7 Regressa a 6

Fluxo Alternativo(5): [Carro é GT] (Passo 2)

2.1 Sistema verifica que é um carro GT, logo pode ser híbrido

2.2 Sistema calcula fiabilidade em função da cilindrada

2.3 Regressa a 5

Fluxo Alternativo(6): [Carro é GT híbrido] (Passo 2)

2.1 Sistema verifica que é um carro GT, logo pode ser híbrido

2.2 Sistema calcula fiabilidade em função da cilindrada

2.3 Jogador indica que carro é híbrido

2.4 Jogador indica a potência do motor elétrico

2.5 Regressa a 6

Fluxo de exceção (1): [Cilindrada limites da categoria] (passo 1)

1.1 Sistema avisa que o valor é inválido

1.2 Regressa a 1

Fluxo de exceção (2): [PAC ultrapassa limites da categoria] (passo 6)

6.1 Sistema avisa que o valor é inválido

6.2 Regressa a 2

Fluxo de exceção (3): [Potência ultrapassa limites] (passo 1)

1.1 Sistema avisa sobre o limite ultrapassado

1.2 Regressa a 1

Fluxo de exceção (4): [Fiabilidade não se enquadra na categoria do carro] (passo 3)

3.1 Sistema avisa sobre o valor inválido

3.2 Regressa a 3

Fluxo de exceção(5): [Potência híbrida combinada ultrapassa valores] (Fluxo Alternativo(1) passo 2.2; Fluxo Alternativo(4) passo 2.6; Fluxo Alternativo(6) passo 2.4)

1 Sistema avisa que valor combinado ultrapassa limites

2 Regressa a 2.2/2.4/2.6 (respetivamente o fluxo alternativo 1 ou 4 ou 6)

3.4. Criação de pilotos

Ator: Administrador.

Descrição: O Administrador cria um piloto.

Cenários: O André cria um piloto para os amigos.

Pré-condição: O utilizador autentica-se com conta de administrador.

Pós-condição: O piloto é criado e o sistema atualizado.

Fluxo normal:

- 1 Administrador insere o nome, CTS e SVA do Piloto
- 2 Sistema guarda o piloto e atualiza

Fluxo de exceção (1): [Valores CTS e/ou SVA inválidos] (passo)

- 1.1 Sistema avisa sobre os valores inválidos
- 1.2 Regressa a 1

3.5. Configurar Campeonato

Ator: Utilizador.

Descrição: O jogador prepara um campeonato para jogar.

Cenários: O João configura o campeonato "Quimdelícia" para poder jogar com os amigos

Pré-Condição: Pelo menos um dos jogadores está autenticado

Pós-Condição: Campeonato configurado

Fluxo normal:

1. Um jogador autenticado seleciona o campeonato pretendido e inscreve-se, selecionando o seu carro e piloto
2. O campeonato fica disponível no sistema para outros jogadores se inscreverem
2. Todos os jogadores que pretendam jogar nesse campeonato inscrevem-se e selecionam o carro e piloto

3. Sistema guarda e atualiza os campeonatos disponíveis

3.6. Configurar e simular corrida

Ator: Utilizador.

Descrição: Inicia-se a configuração das corridas do campeonato e posteriormente a simulação das mesmas

Cenários: Como todos os amigos estão registados, o António decide começar a configuração para poder iniciar a simulação.

Pré-condição: Pelo menos um campeonato já configurado

Pós-condição: Corrida simulada e pontuação atualizada para os jogadores autenticados

Fluxo normal:

1. O jogador que configurou o campeonato inicia a configuração/simulação
2. As condições da corrida, o circuito inicial e a situação meteorológica são apresentadas aos utilizadores
3. Jogadores selecionam, se pretenderem, efetuar afinações, mudanças de pneu e/ou modo do motor
4. Sistema inicia a simulação, baseando-se nas posições relativas dos carros
5. Sistema apresenta, durante a corrida, eventuais despistes, avarias e/ou ultrapassagens ao longo do circuito
6. Sistema apresenta, no final de cada volta, as posições dos jogadores
7. No final da corrida, o sistema apresenta as classificações de cada jogador
8. Caso não seja a última corrida, regressa a 1
9. No final do campeonato são apresentadas as pontuações finais de cada jogador

Fluxo Alternativo(1): [Todos os jogadores sofrem acidentes ou avarias] (passo 5)

5.1 O sistema avisa que não há jogadores em condição de continuar (sem atribuição de pontos para esta corrida)

5.2 O sistema termina a corrida atual

5.3 Regressa a 8

Fluxo de exceção(1): [Número de limite de afinações ultrapassado] (passo 3)

3.1 Sistema avisa o limite ultrapassado de afinações (2/3 do número de corridas do campeonato)

3.2 Regressa a 6

3.7. Ver Ranking

Ator: Utilizador.

Descrição: O Inácio vê a sua classificação global

Pré-Condição: O jogador está autenticado

Pós-Condição: O jogador visualiza as pontuações do sistema

Fluxo normal:

1. O jogador seleciona a opção de "Ver Ranking"
2. O sistema mostra os rankings de todos os jogadores (registados), realçando o jogador que visualiza.

4. Conclusão

Nesta primeira fase foi feita uma análise de todos os requisitos presentes no enunciado entregue pela equipa docente de forma a podermos realizar o melhor possível os diagramas de domínio e de uses cases que melhor representam o funcionamento do sistema, as várias entidades que o sistema possui e a forma como estas entidades interagem com o sistema.

Nesta fase foram feitas várias interpretações daquilo que seria necessário representar nos diagramas acabando no final por escolhermos aquela que, na nossa opinião, melhor representa os requisitos do enunciado.

5. Referências

GitHub do Grupo: <https://github.com/LuisFilipe6/DSS2223>