



UNIVERSIDADE DO MINHO
Departamento de Informática

DESENVOLVIMENTO DE SISTEMAS DE SOFTWARE

Relatório Nº2:

Modelação conceptual



Figura 1: Equipa de Trabalho

Elaborado por:

António Santos Gil Afonso - A95180

Gonçalo Fernandes Lemos - A96484

Hugo Filipe Silva Abelheira - A95151

Simão Pedro Cunha Matos - A96115

Grupo 50

URL do repositório: https://github.com/ToGi22/DSS_G50

Ano Letivo 2022/23

22 de novembro de 2022

Índice

1	Introdução	2
2	Objetivos	2
3	Diagrama de Componentes	2
4	Diagramas de Classes	3
4.1	Utilizador	3
4.2	Carro	3
4.3	Corrida	4
4.4	Circuito	4
4.5	Campeonato	5
4.6	Racing Simulator	5
5	Código Legado	6
6	Diagramas de Sequência	6
7	Conclusão	10

1 Introdução

Este relatório foi elaborado no âmbito da unidade curricular de Desenvolvimento de Sistemas de Software, e corresponde à segunda fase do projeto proposto para o ano letivo de 2022/2023.

Esta segunda fase do projeto, consistiu no desenvolvimento da arquitetura conceptual que seja capaz de suportar os Use Cases definidos na fase anterior e dos modelos comportamentais necessários para descrever o comportamento pretendido para o sistema.

2 Objetivos

Numa fase inicial, tendo em conta o Modelo de Domínio e os Use Cases delineados na fase anterior, definimos as APIs do simulador de corridas, para isso dividimos o nosso sistema em subsistemas pelos quais estes métodos foram distribuídos. De seguida, passamos para a criação do diagrama de componentes no qual foram representados estes vários subsistemas que consideramos adequados para a realização desta aplicação. Após isto, criamos os diferentes diagramas de classes, definindo os atributos destas mesmas classes assim como as relações entre si. Foram definidos os diagrama de sequência, que representam a comunicação entre os vários objetos. Todo este processo referido, serviu para desenvolver um modelo conceptual coeso e concordante com a primeira fase desenvolvida pelo grupo. Todos os diagramas foram desenvolvidos usando a ferramenta Visual Paradigm.

3 Diagrama de Componentes

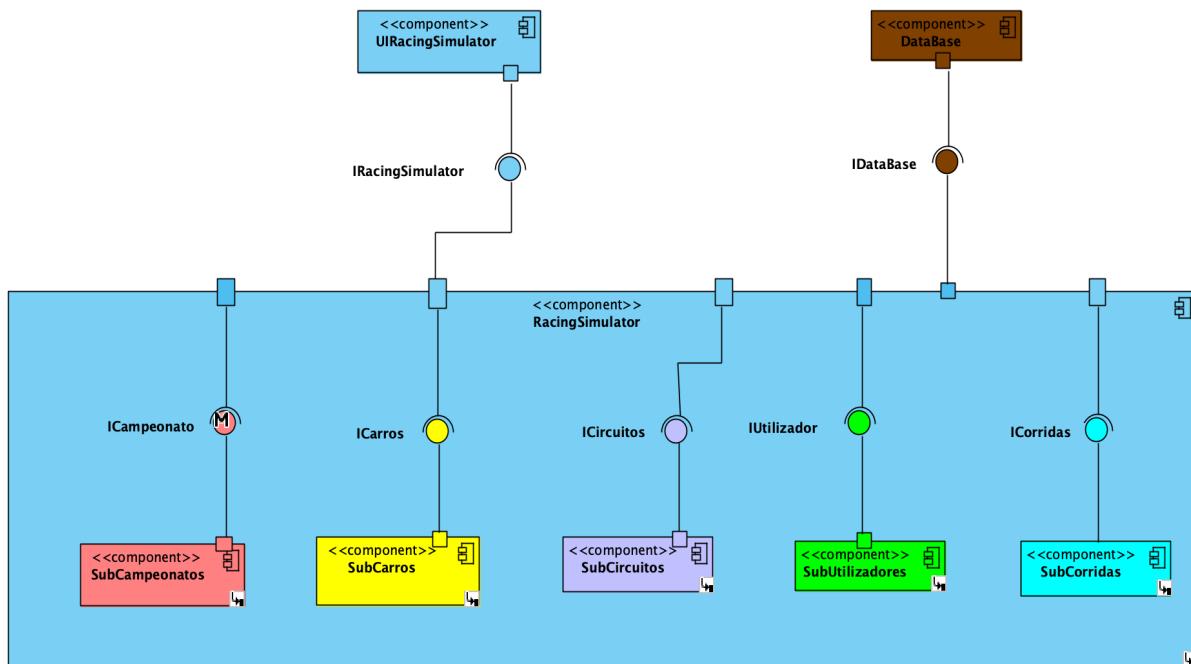


Figura 2: Diagrama de Componentes

A concepção de subsistemas permite uma melhor organização do projeto, uma vez que haverá o agrupamento dos métodos pelos subsistemas onde melhor se adequam. Esta prática é crucial no que toca a encapsulamento, uma vez que cada subsistema implementa uma interface com métodos que podem ser utilizados no "exterior", evitando assim que outros métodos sejam utilizados noutras subsistemas. Deste modo, ficaram definidos os seguintes subsistemas: subUtilizador, SubCarro, SubCorrida, SubCircuitos e Racing Simulator.

4 Diagramas de Classes

Com base no Diagrama de Domínio e acrescentando novas classes relativas à lógica de negócios construímos o nosso diagrama de classes. Decidimos dividi-lo pelos diferentes subsistemas, onde cada um destes possui uma interface e uma classe Facade, esta com o propósito de ser o intermediário entre subsistemas.

O grupo optou por omitir métodos básicos como getters, setters e clones dos diferentes atributos das classes.

4.1 Utilizador

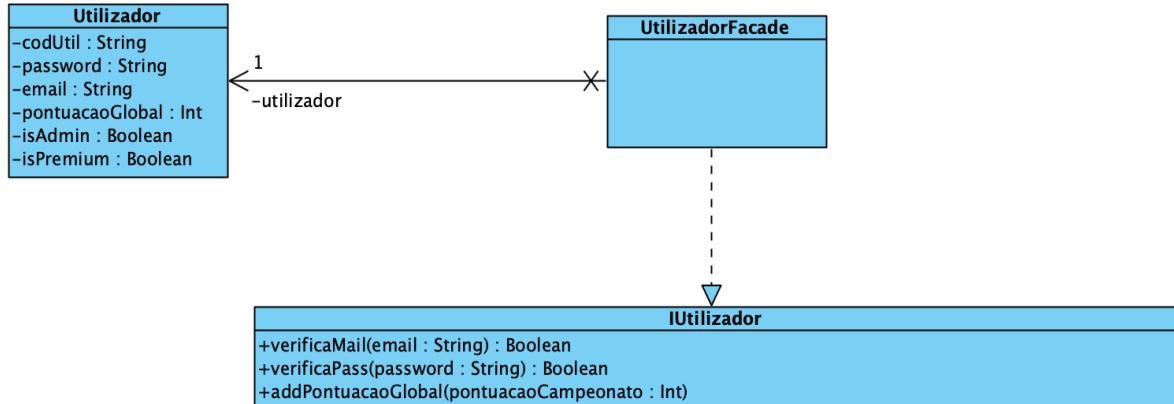


Figura 3: Diagrama de Classes- SubUtilizador

4.2 Carro

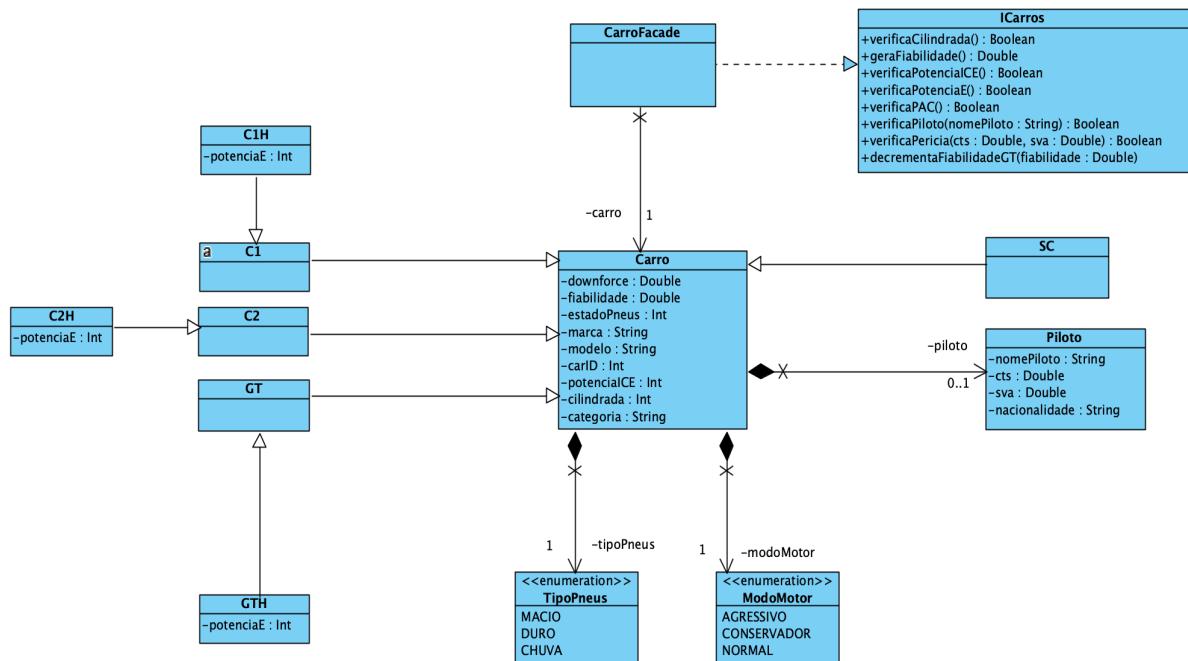


Figura 4: Diagrama de Classes- SubCarro

4.3 Corrida

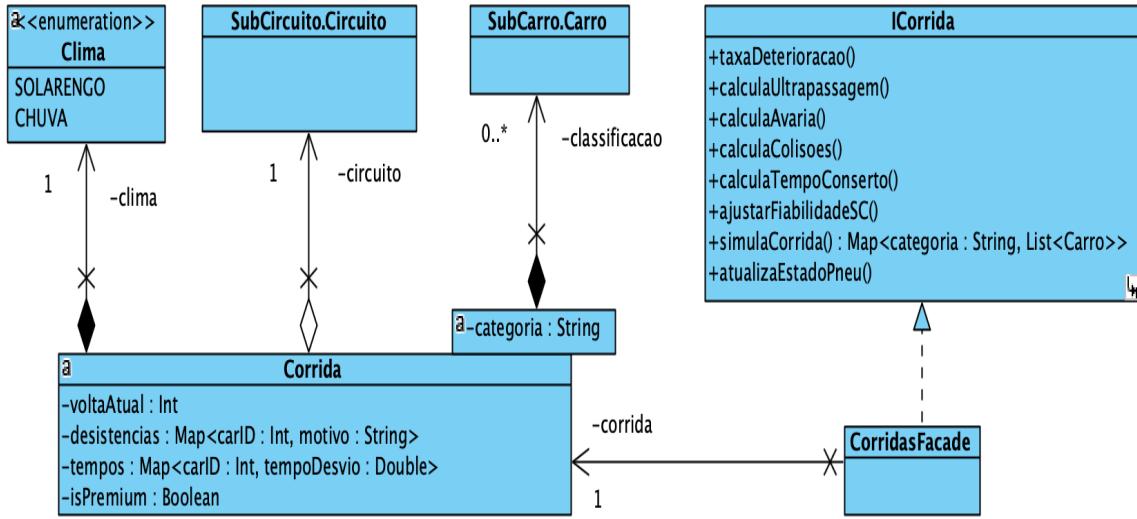


Figura 5: Diagrama de Classes- SubCorrida

4.4 Circuito

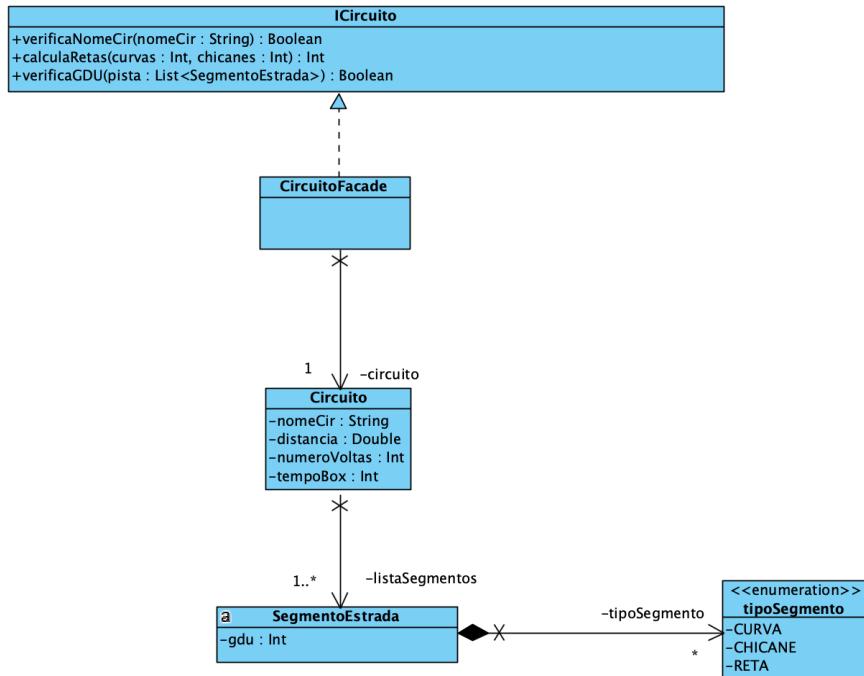


Figura 6: Diagrama de Classes- SubCircuito

4.5 Campeonato

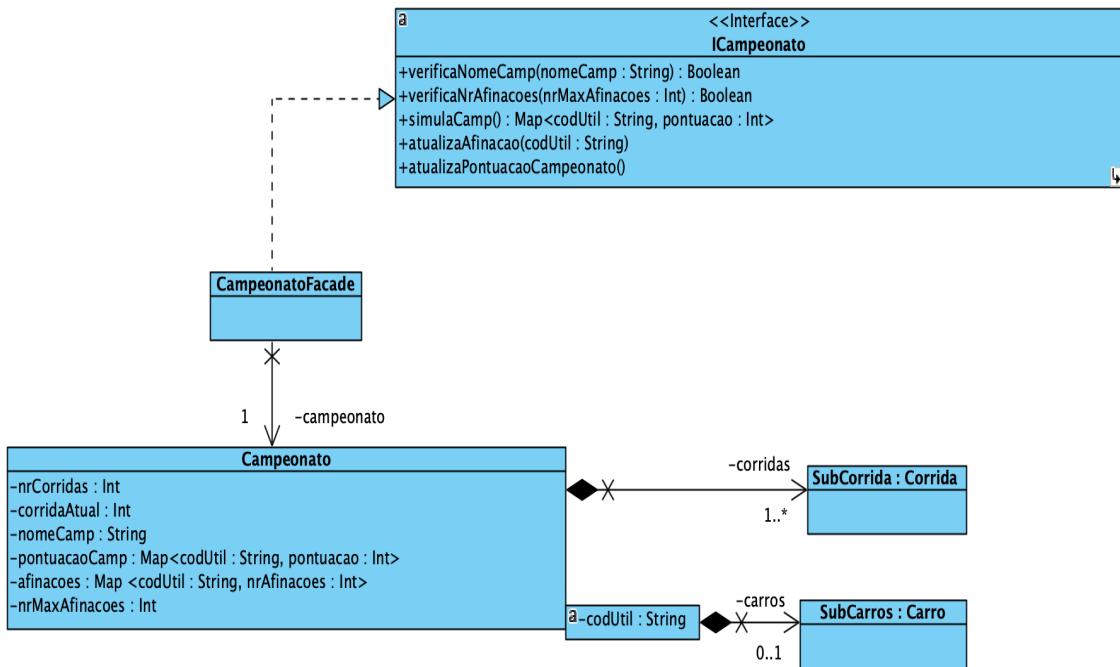


Figura 7: Diagrama de Classes- SubCampeonato

4.6 Racing Simulator

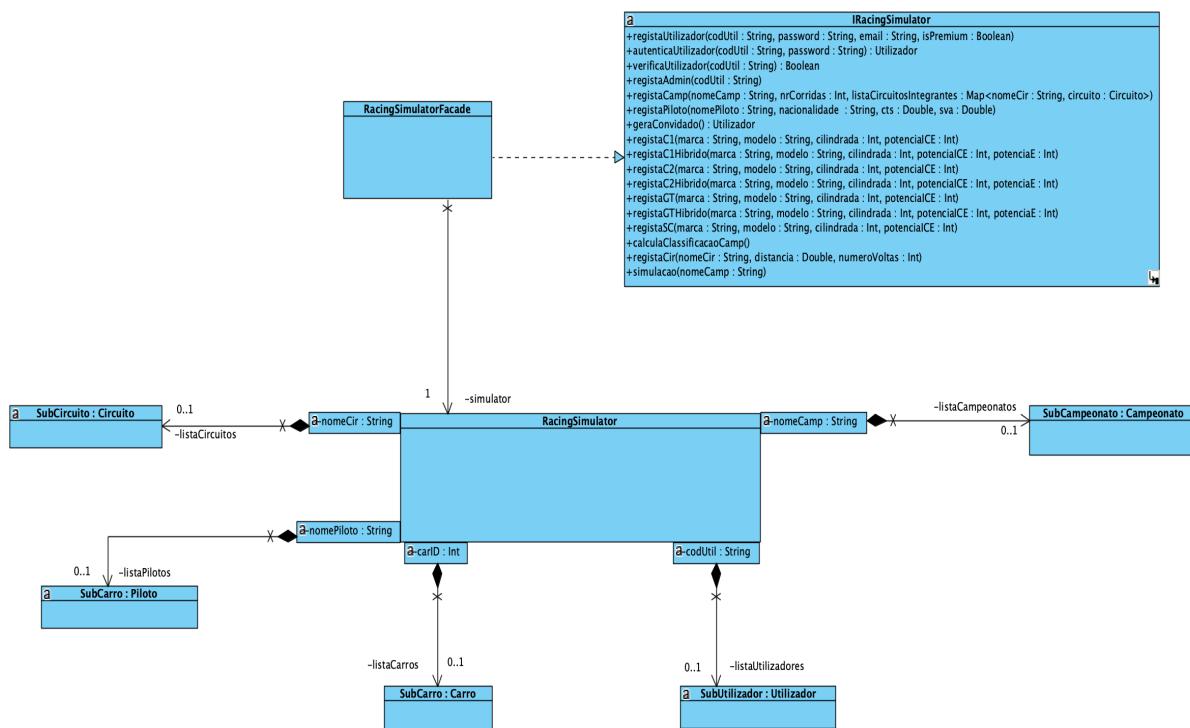


Figura 8: Diagrama de Classes- Racing Simulator

5 Código Legado

Foi realizada uma análise do código disponibilizado pelos docentes e o nosso grupo decidiu estruturar o sub-sistema relativo aos carros com base no código legado, ou seja, as classes dos diferentes carros herdam de uma classe "Carro" sendo esta a superclasse onde tem os atributos comuns aos diferentes tipos de carros e por sua vez as classes de carros híbridos herdam da classe do seu tipo de carro, por exemplo, a classe "C1 híbrido" herda da classe "C1" e esta por sua vez herda da classe "Carro". Também optamos por juntar a classe piloto ao Sub-sistema Carro , não existindo pilotos fora do contexto de existência de carros.

O nosso grupo decidiu também não utilizar a parte referente ás equipas competindo todos os jogadores de forma individual, visto no enunciado do projeto não estar referenciada qualquer necessidade de estes competirem em equipas.

6 Diagramas de Sequência

Foram realizados Diagramas de Sequência para os métodos que o grupo achou mais relevantes, como por exemplo os referentes ao Cenário 5. Estes são diagramas de especial importância, visto que ilustram, de uma maneira facilmente legível, o funcionamento da função, sendo esse funcionamento facilmente convertido para código.

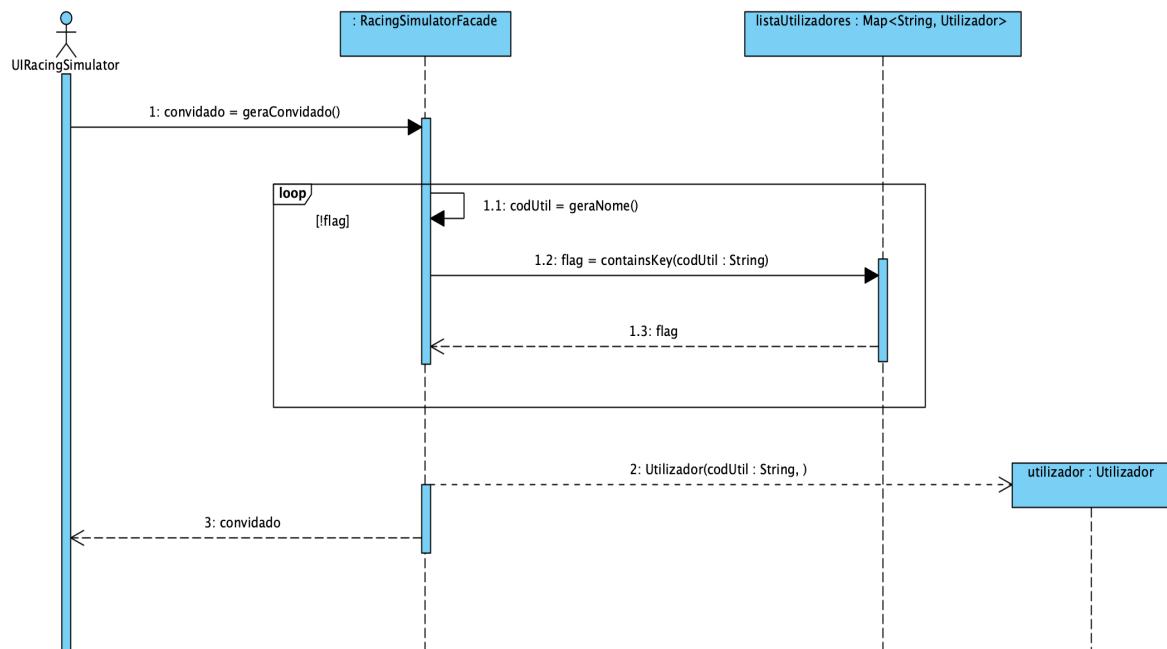


Figura 9: Diagrama de Sequência- Gerar Convidado

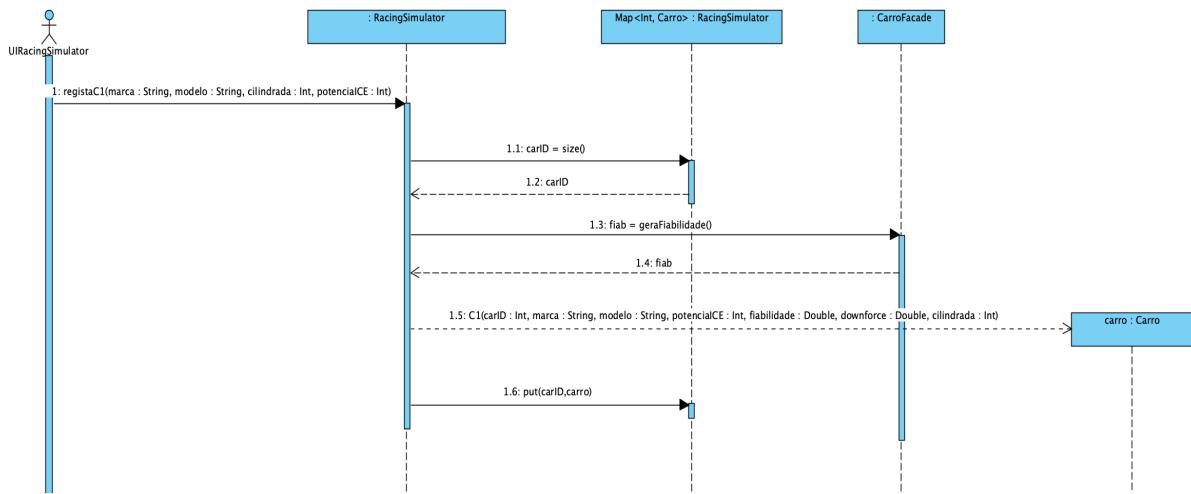


Figura 10: Diagrama de Sequência- Registrar Carro C1

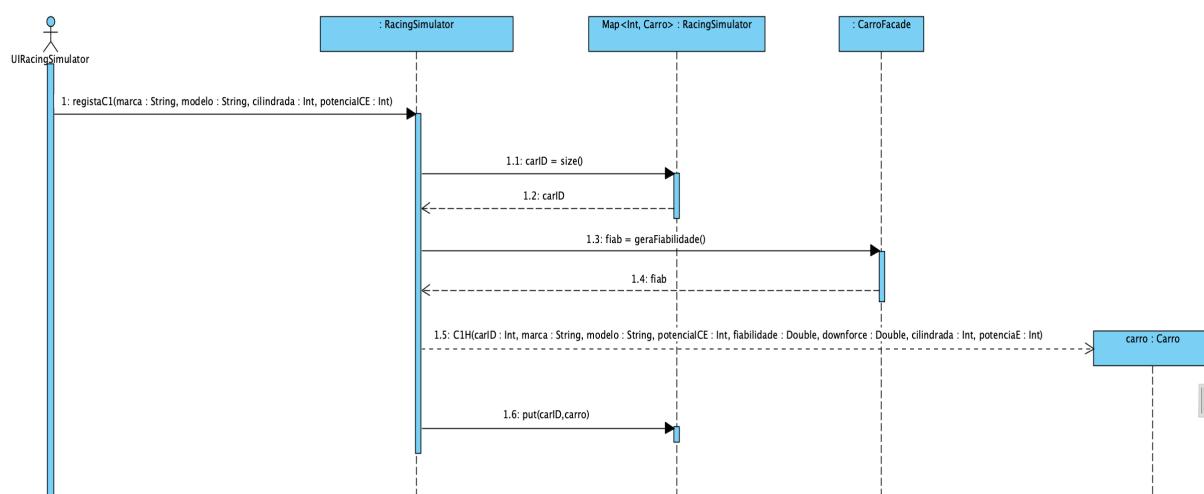


Figura 11: Diagrama de Sequência- Registrar Carro C1 Híbrido

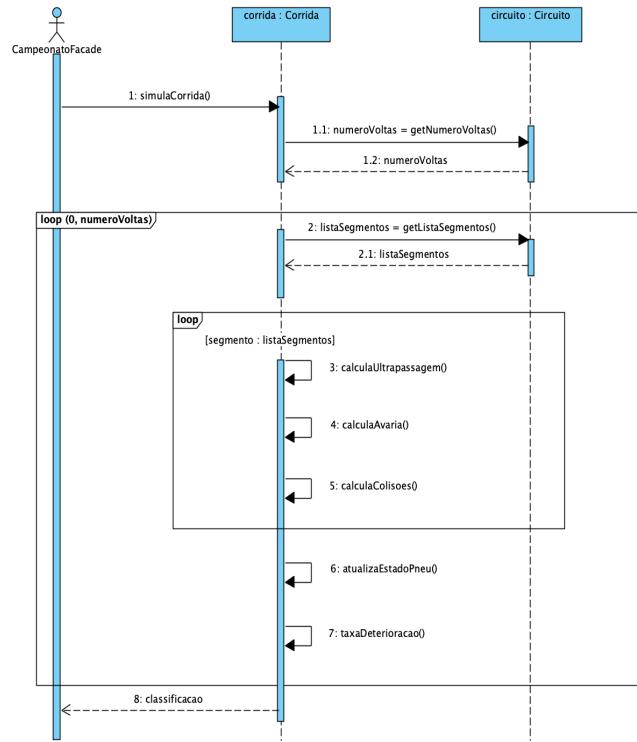


Figura 12: Diagrama de Sequência- Simular Corrida

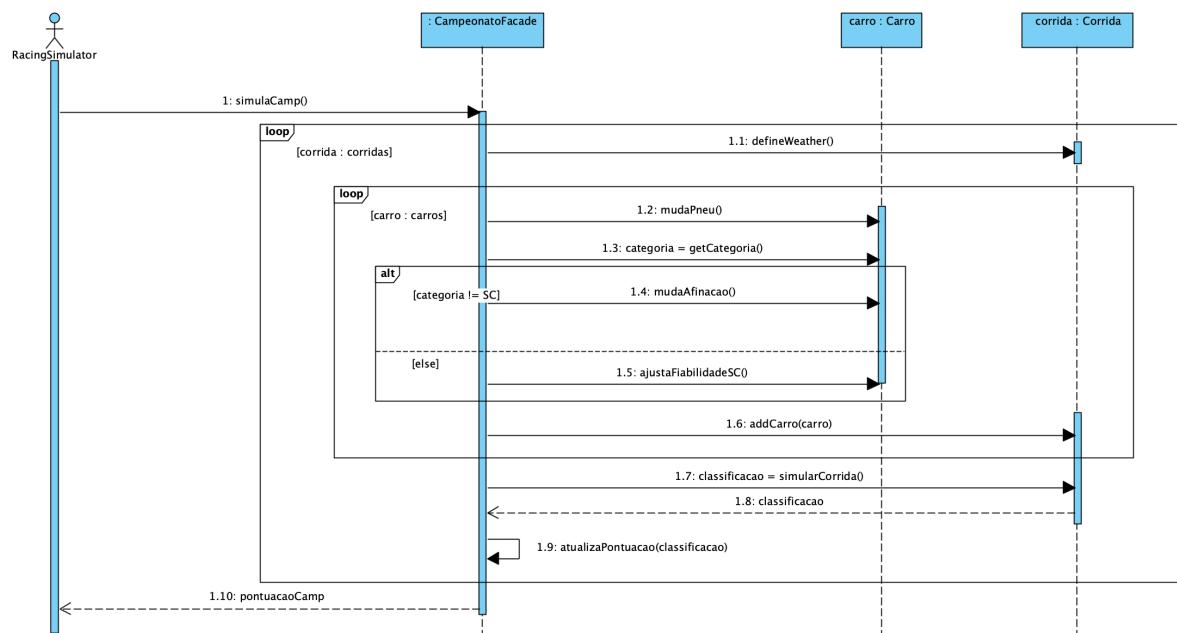


Figura 13: Diagrama de Sequência- Simular Campeonato

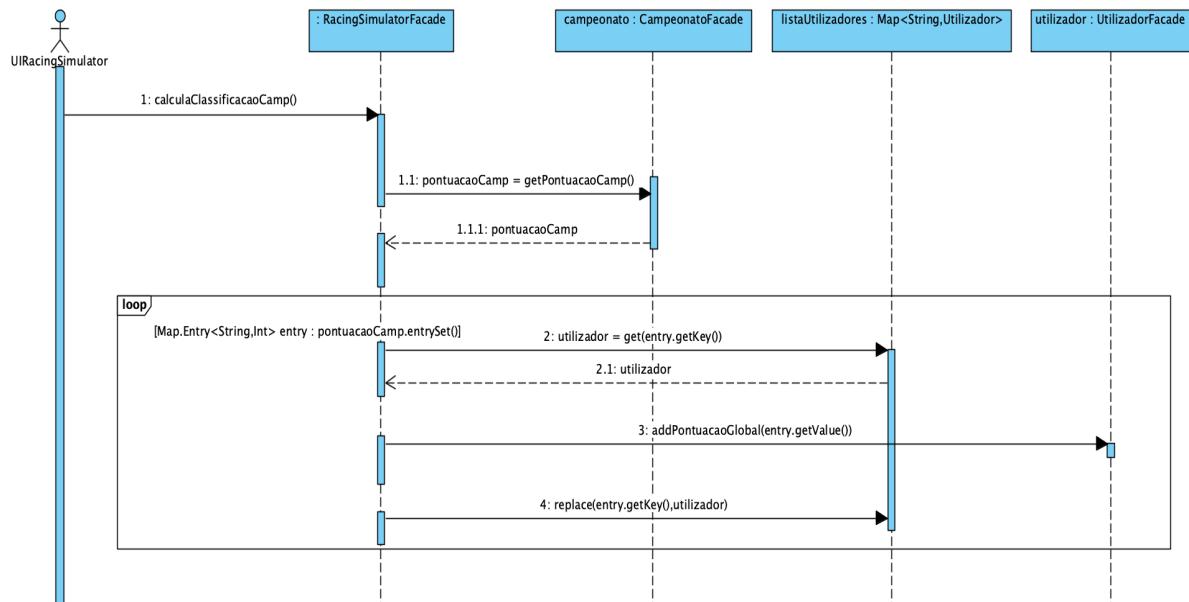


Figura 14: Diagrama de Sequência- Calcular Classificação

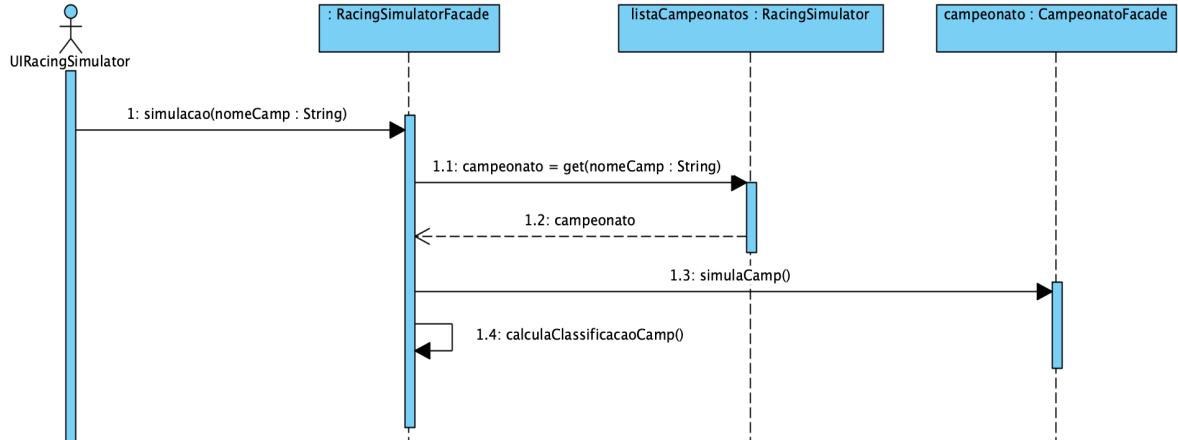


Figura 15: Diagrama de Sequência- Simulação

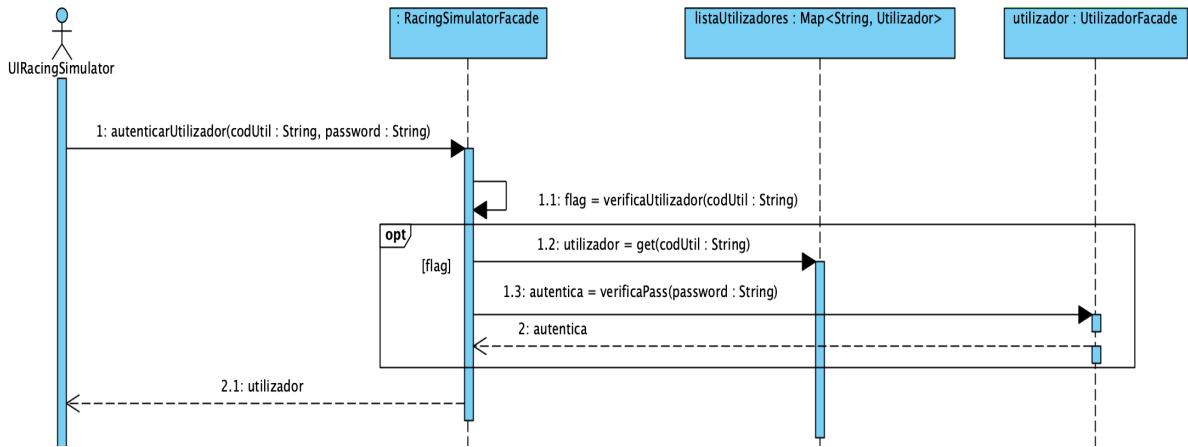


Figura 16: Diagrama de Sequência- Autenticar Utilizador

7 Conclusão

Esta fase do projeto envolveu a análise dos use cases, identificação das principais classes do projeto e a sua implementação. Isto permitiu-nos consolidar os conhecimentos previamente adquiridos durante este semestre sobre arquitetura de sistemas e modelação comportamental. É importante referir que ao longo desta 2^a Parte do trabalho foram realizadas várias interpretações e abordagens, sendo que no final optamos pela que, a nosso ver, é a mais completa e mais respeita o trabalho realizado até à data e o enunciado do TP.

Em suma, consideramos que cumprimos os objetivos apontados a esta fase do projeto de forma completa e adequada, estando o grupo satisfeito com o seu resultado final.