

**Ciências
ULisboa**

Relatório Fase 2

CSS 007

**Miguel Lemos 62248
Gonçalo Moutinho 62256
Francisco Parasca 62233**

Descrição da Arquitetura em camadas

- Camada de Negócio
- Camada de Dados
- Camada de Apresentação

Na camada de negócio encaixam-se os ficheiros do padrão *Service* que foi utilizado por causa dos vários recursos diferentes que as transações utilizam.

Na camada de dados encaixam-se os ficheiros do padrão *Repository* que foi utilizado para criar um encapsulamento dos dados.

Na camada de apresentação encaixam-se os ficheiros do padrão *Controller* que foi utilizado para receber pedidos e responder em *HTTP*.

Os ficheiros *DTO* são usados para estabelecer a ligação entre camadas.

Os ficheiros *Entity* são uma representação dos objetos do sistema.

Modelo de domínio

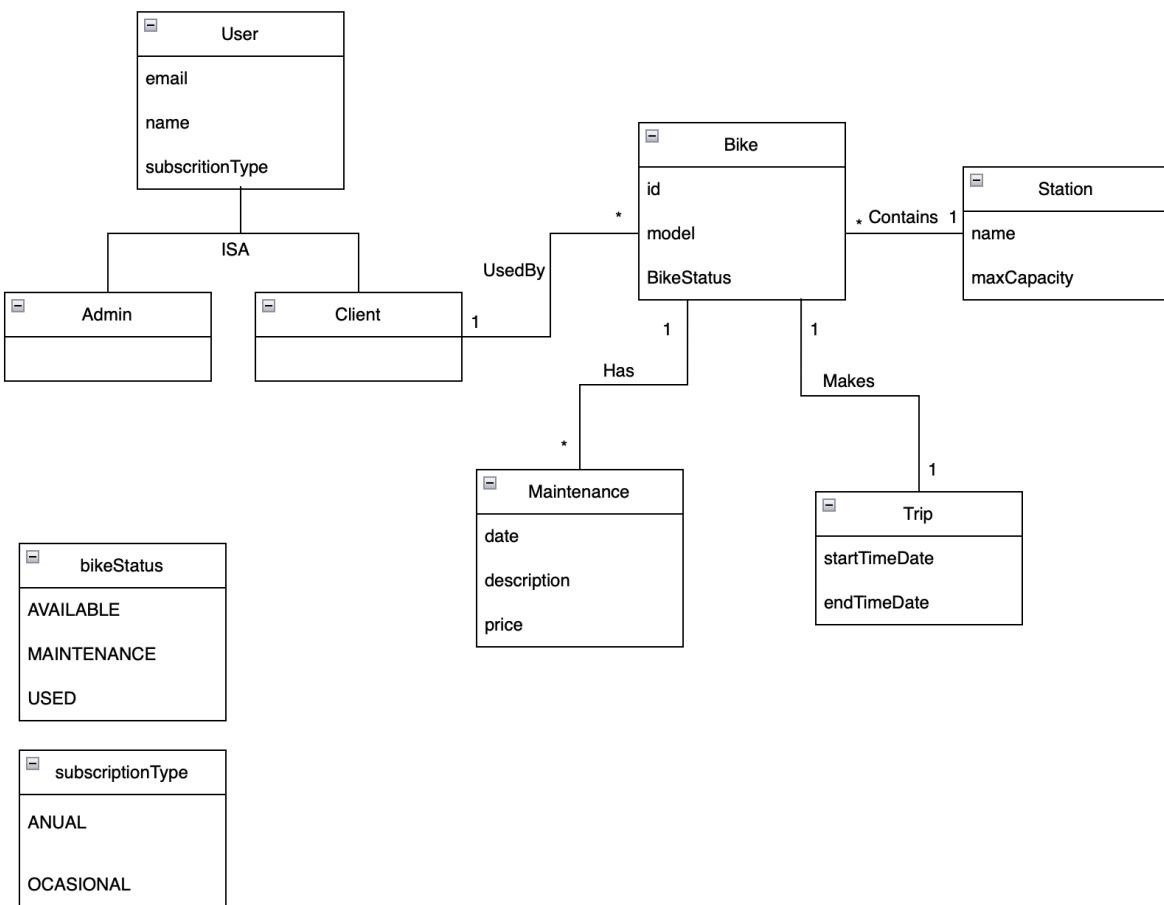
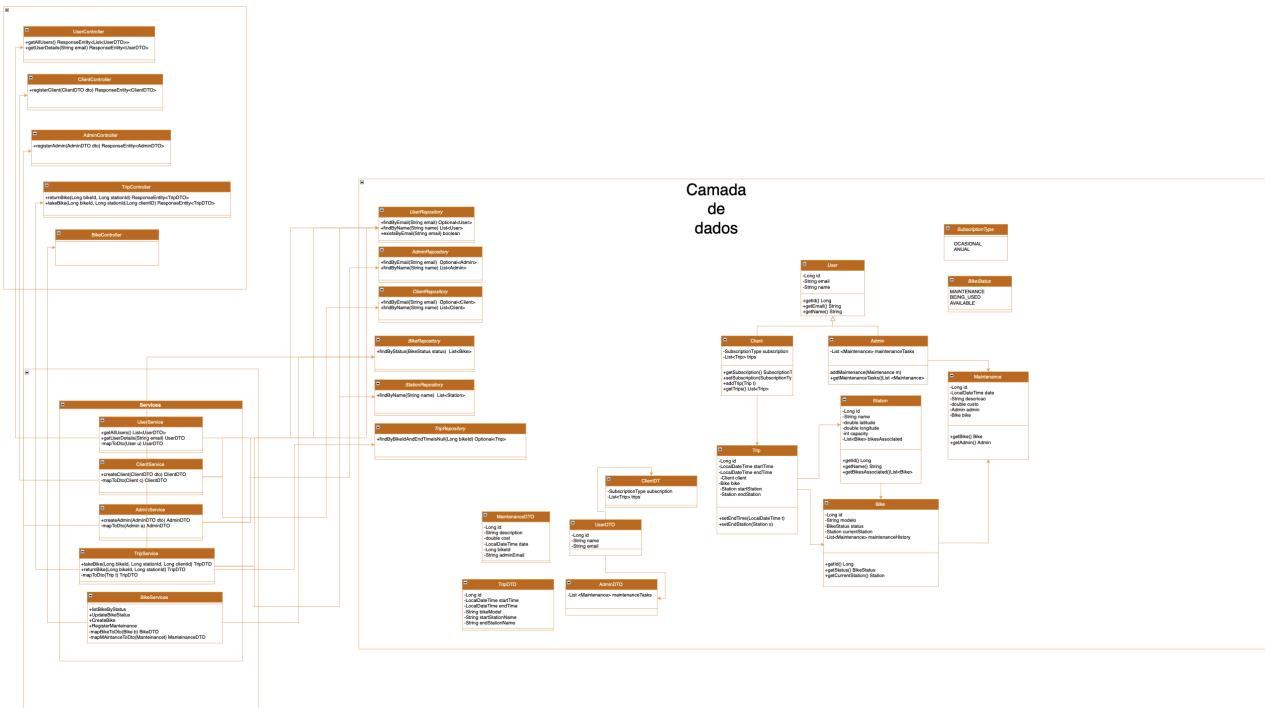
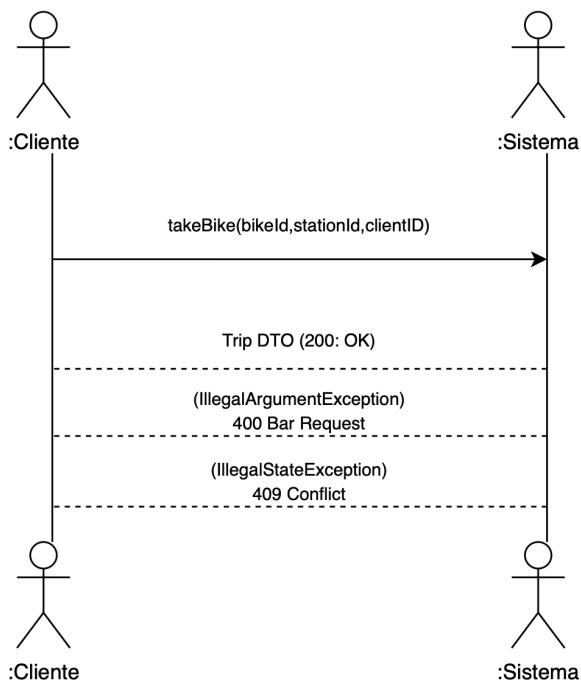


Diagrama de Classes e SSD

Diagrama de Classes



SSD - Caso de Uso K



Decisões de mapeamento

@Entity - É usado para indicar que é uma tabela e usamos em cada uma das classes que fazem parte das Entities.

@Id - Identifica a chave principal de cada uma das tabelas e pode ser encontrado em todas as tabelas.

@GeneratedValue(strategy = GenerationType.IDENTITY) - Segue o @Id e faz com que o valor da chave principal seja automaticamente gerada e que nunca seja igual a outra que tenha sido anteriormente gerada.

@Column - Associa uma coluna da tabela ao valor de uma variável e pode ser encontrado múltiplas vezes em cada tabela (Entity).

@OneToMany - Faz a associação de um objeto a múltiplos outros. Um exemplo desses pode ser a associar uma única estação a múltiplas bicicletas.

@ManyToOne - Faz a associação de múltiplos objetos a apenas 1 outro objeto. Um exemplo pode ser o inverso do exemplo do @OneToMany, ou seja, múltiplas bicicletas para 1 estação.

@JoinColumn - Acompanha a @ManyToOne e serve para identificar a respetiva coluna como uma chave estrangeira.

Decisões de negócio

Concrete Class vs Join Class - Escolhemos a *Join Class* porque o projeto não é de grande dimensão e assim este tipo de *Class* fica mais intuitivo e ajuda a manter os dados mais seguros.