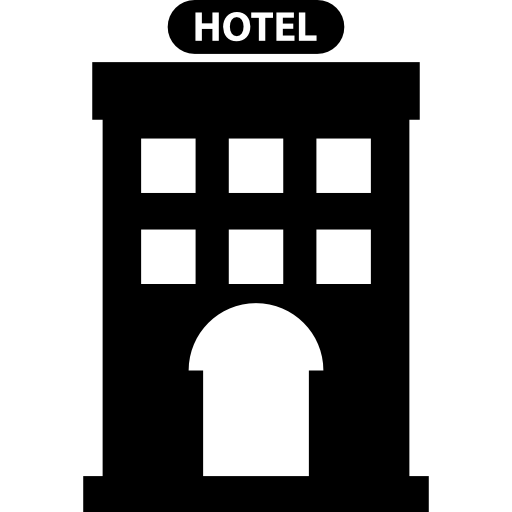
**Escola Superior de Tecnologia**

Engenharia de Sistemas Informáticos

Gestão de Alojamento Turístico

Alexandre Mykyta, nº27998

Índice

[Introdução 3](#_Toc182314520)

[Fase 1 4](#_Toc182314521)

[Classes Identificadas 4](#_Toc182314522)

[Classe Users 4](#_Toc182314523)

[Classe Accommodation 4](#_Toc182314524)

[Classe Booking 4](#_Toc182314525)

[Classe Client 4](#_Toc182314526)

[Classe CheckIn 5](#_Toc182314527)

[Classe Reservation 5](#_Toc182314528)

[Classe Registry 5](#_Toc182314529)

[Estruturas de dados a utilizar 6](#_Toc182314530)

[Listas 6](#_Toc182314531)

# Introdução

Neste trabalho prático, realizado no âmbito da unidade curricular de Programação Orientada a Objetos, pretendo consolidar conceitos fundamentais do Paradigma Orientado a Objetos, analisar problemas reais, desenvolver capacidades de programação em C#, potenciar a experiência no desenvolvimento de software e assimilar o conteúdo da Unidade Curricular.

O tema selecionado envolve o desenvolvimento de uma aplicação relacionada com a Gestão de Alojamentos Turísticos. Esta aplicação será projetada para oferecer funcionalidades essenciais, incluindo a gestão de reservas, a consulta e o registo de alojamentos disponíveis, o processo de check-in e check-out de clientes, e a administração das informações dos hóspedes. Além disso, a aplicação permitirá ao administrador aprovar ou rejeitar check-ins, fornecendo uma plataforma eficiente e intuitiva para a gestão de alojamentos turísticos.

# Fase 1

## Classes Identificadas

### Classe Users

A classe Users representa os utilizadores do sistema, que podem ser administradores ou clientes. Esta classe armazena informações básicas, incluindo o nome de utilizador, a palavra-passe e o tipo de utilizador. O sistema utiliza esta classe para direcionar o utilizador para o painel adequado (administração ou cliente) após o login. O administrador pode ver e aprovar reservas, enquanto o cliente pode fazer novas reservas e visualizar informações sobre os alojamentos.

### Classe Accommodation

Esta classe representa os alojamentos disponíveis no sistema, contendo informações como nome, tipo (por exemplo, quarto duplo ou apartamento), capacidade e preço. A classe Accommodation facilita a listagem e visualização dos alojamentos disponíveis, permitindo que o cliente selecione e visualize detalhes ao fazer uma reserva.

Classe Booking  
 A classe Booking representa uma reserva, ligando um cliente a um alojamento específico para um intervalo de datas. Ela contém informações sobre o cliente, o alojamento reservado e as datas de início e fim da reserva. Esta classe permite que o sistema registe e consulte as reservas, servindo também como base para a confirmação de check-ins pelo administrador.

Classe Client  
 A classe Client representa os clientes que utilizam o sistema para fazer reservas. Ela armazena os dados de contacto e preferências do cliente, facilitando o processo de reserva. Cada cliente pode ter uma ou mais reservas associadas.

Classe CheckIn  
 A classe CheckIn representa o processo de entrada do cliente no alojamento reservado. Esta classe armazena o estado do check-in (pendente, aprovado ou recusado) e permite ao administrador aprovar ou rejeitar o check-in com base nas reservas realizadas. Esta funcionalidade é importante para controlar o acesso dos clientes aos alojamentos.

Classe Reservation  
 A classe Reservation gere o processo de registo de uma reserva e organiza todas as operações relacionadas com o ato de reservar, incluindo a verificação da disponibilidade do alojamento e a criação da reserva. Esta classe centraliza a lógica necessária para registar e consultar reservas, facilitando o processo para os utilizadores.

Classe Registry  
 A classe Registry serve para manter um histórico de ações e eventos relevantes no sistema, como registos de reservas e check-ins. Ela armazena todas as atividades e alterações efetuadas no sistema, permitindo a consulta posterior, caso seja necessário. Esta funcionalidade é útil para fins de auditoria e para fornecer um histórico ao administrador

# Estruturas de dados a utilizar

## Listas

Foram extensivamente utilizadas para manter coleções dinâmicas de objetos. Cada classe possui uma lista correspondente para armazenar os seus elementos associados. Estas listas são específicas para cada instância.

Foram também usadas listas estáticas, para manter informações compartilhadas entre todas as instâncias da classe tornando-se assim uma lista geral da classe