

# Relatório Sistemas Operativos II



Realizado por:  
Pedro Vaz - 34265  
Luís Rosado - 34249

Este trabalho consiste na criação de uma WebApp de um serviço de reservas de espaços desportivos, com recurso a Java Servlet. Esta WebApp consiste num package deployable em tomcat e numa base de dados em PostgreSQL.

## Source Packages

- **clube\_desportivo**

### ClubeServlet

Classe com o Servlet que estende as funcionalidades do servidor. Processa os pedidos e respostas dos clientes gere os ficheiros JSP que são apresentados.

- **clube\_desportivo.beans**

### Reserve

Classe com o bean Reserva, objecto que representa a reserva. Operações set() e get() dos atributos de uma reserva.

### Space

Classe com o bean Space, objecto que representa o espaço. Operações set() e get() dos atributos de um espaço.

- **clube\_desportivo.database**

### ClubLogic

Interface com as operações da WebApp que utilizam a base de dados.

### DataManager

Para cada pedido de acesso à base de dados é instanciado um DataManager, assegura que cada utilizador tem um acesso único.

### GetPropertyValues

Faz a leitura do ficheiro com as propriedades do acesso à base de dados e retorna os dados pedidos.

## PostgresConnector

Implementa a conexão à base de dados e implementa as operações de consulta e escrita na base de dados.

## **Jsp's**

### **topmenu**

Contém a imagem que aparece no topo da webapp.

### **leftmenu**

Contém os botões que correspondem às ações de listar os espaços, verificar a disponibilidade de uma data, realizar uma reserva e por fim listar as reservas para um certo espaço.

### **listSpaces**

Esta página consulta a base de dados e cria uma tabela html com os respetivos espaços e seus preços.

### **listReserves**

Nesta página o utilizador escolhe qual o espaço que deseja consultar as respetivas reservas, em seguida é feito a consulta à base de dados e é criada uma tabela html com o código das reservas, o nome do responsável, a data do checkin e do checkout, caso não hajam reservas é apresentada uma mensagem a avisar o utilizador.

### **checkReserve**

Aqui é feita a verificação de uma data, é pedido ao utilizador que escolha o espaço que deseja, a data do checkin e do checkout, caso seja possível fazer reserva é apresentado ao utilizador o custo da reserva e 2 opções: fazer reserva ou então voltar à página inicial.

## **reserveSpace**

Esta página faz uma ligação direta para o **checkReserve**, o utilizador tem que verificar sempre primeiro se é possível efetuar uma reserva antes de a efetuar, caso seja possível e o utilizador querer fazer reserva é pedido o nome do responsável, o nº de telemóvel e o número de utilizadores. Depois de o utilizador fazer submit de todos os campos vai para a página **confirmReserve**.

## **confirmReserve**

Nesta página é apresentada o resultado da reserva, se foi efetuada com sucesso é apresentado o código único da reserva, se não foi bem sucedido é apresentada uma mensagem ao utilizador.

## **index**

Esta é a home page da webapp, contém um api do google maps, algumas imagens da cidade de Évora e dos espaços desportivos.

- Todas as páginas fazem include de duas páginas em particular, fazem include da página topmenu e da leftmenu, assim estas duas páginas estão sempre presentes aos olhos do utilizador.

## **Balanço**

Na nossa opinião o trabalho está bom, efetua todas as operações pedidas, tem um interface e um aspeto amigável ao utilizador. O único parâmetro não cumprido pelo nosso trabalho é a replicação, tivemos algumas dificuldades em compreender este aspeto do trabalho e decidimos não o incluir.