DEPARTAMENTO DE ENGENHARIA INFORMÁTICA

POLITÉCNICO DE LEIRIA

HTTP://WWW.DEI.ESTG.IPLEIRIA.PT/

TeSP – Programação de Sistemas de Informação

PROGRAMAÇÃO PARA A WEB - SERVIDOR

---

1. Introdução

O objetivo deste projeto é consolidar conhecimentos na área de desenvolvimento web servidor, aplicando os conceitos adquiridos na unidade de curricular de Programação para a Web - Servidor (PW-S). Como tal este projeto terá de ser desenvolvido integralmente em PHP recorrendo à *framework* WeblogicMVC, utilizando uma pequena base de dados de apoio para o seu correto funcionamento.

Antes de iniciar o desenvolvimento devem criar um repositório Github de modo a articular a gestão das tarefas de desenvolvimento na unidade curricular de Metodologias de Desenvolvimento de Software (MDS).

2. Cenário

Pretende-se implementar uma plataforma de gestão de um aeroporto, cuja denominação é *FlightTravelAir*. Possui como principal objetivo permitir a um passageiro efetuar a compra de uma passagem aérea de forma simples, rápida e direta. Através da plataforma, o passageiro pode visualizar a informação relativamente aos voos e respetivas escalas e horários.

A plataforma *FlightTravelAir* permite também a administração e gestão de voos e toda a informação que está associada. Esta funcionalidade encontra-se disponível apenas para o administrador.

3. Desenvolvimento

Pretende-se que cada grupo de trabalho tenha em conta os seguintes aspetos no desenvolvimento do projeto:

3.1 Especificações (dos requisitos)

A plataforma possui quatro zonas reservadas, o que corresponde a quatro perfis de utilizador diferente:

1. Passageiro
2. Operador de *checkin*
3. Gestor
4. Administrador

3.1.1 Zona reservada Passageiro:

- Um passageiro/cliente final tem de realizar o registo e respetiva autenticação para aceder à sua zona reservada;
- O registo é composto por: nome completo, data de nascimento, email, telefone, *username* e *password*;
- Um passageiro/cliente final pode atualizar os seus dados na zona reservada (exceto *username*);
- Realiza a compra da passagem *online*;
- Consulta os voos pela origem e o destino;
- Consulta os voos entre 2 datas e com origem e destino (para efetuar compra);
- Pode adquirir passagem de ida ou de ida/volta;
- Simula o pagamento de uma passagem;
- Descarrega/Visualiza o bilhete de avião detalhado (após o pagamento);
- Consulta o histórico de passagens já adquiridas.

3.1.2 Zona reservada Operador de *Checkin* (*back-office*):

- Realiza autenticação;
- Realiza o *checkin* de um passageiro;
- Visualiza os detalhes do voo (clientes com *checkin* já realizado ou não).

3.1.3 Zona reservada Gestor de Voo (*back-office*):

- Realiza autenticação;
- Introduz os voos (CRUD);
- Introduz as escalas (CRUD);
- Introduz os aviões (CRUD).

3.1.4 Zona reservada Administrador (*back-office*):

- Realiza autenticação;
- Administra as contas dos operadores de *ckeckin*;
- Administra as contas dos gestores de voo;
- Introduz aeroportos (CRUD).

3.2 Dados de teste

Para testar a aplicação deverão introduzir pelo gestor de voo:

- 3 voos de uma escala (ida e volta), com no máximo 4 aeroportos;
- 3 voos com 2 escalas (ida e volta);
- 3 voos com 3 escalas (ida e volta).

Para cada voo deverão definir:

- Preço do voo;
- Todas as escalas devidamente ordenadas e com a informação relativamente às mesmas;
    - Os pontos de origem e de destino nas escalas serão exclusivamente aeroportos;
    - Deve ser atribuído um avião a cada escala, com a sua designação e lotação do mesmo.

3.3 Pressupostos

- A entidade Aeroporto armazena os dados de cada aeroporto e será utilizada como origem e destino nas escalas de voos;
- A entidade Voo pode ter uma ou mais escalas e deve ser associado o preço de venda do voo;
- A entidade Escala é caracterizada por:
    - Aeroporto de origem e destino;
    - Data e hora da origem e do destino;
    - Distância.
- A entidade PassagemVenda é constituída por um ou 2 voos (ida ou ida/volta);

- A passagem deve ter associado o preço pago pelo passageiro. Calculado com base no preço de venda do voo. Deve ser ainda registado a data e hora da compra da passagem, bem como, o registo do *checkin*;
- A entidade Avião é constituída pela referência (por exemplo: A11004), a lotação e o tipo de avião (por exemplo: airbus A330);
- A entidade Utilizador é constituída por nome, morada, email, nif, telefone, *username*, *password* e perfil;
- A entidade Utilizador possui uma relação 1-N com a entidade PassagemVenda, por forma a manter o histórico de passagens adquiridas;
- O número de passagens vendidas é armazenado numa entidade que liga a entidade Voo à entidade Avião.

3.4 Outros

- O trabalho deverá ser implementado recorrendo ao padrão estrutural MVC.
- As sessões devem ser usadas exclusivamente para efeitos de autenticação e autorização. Os restantes dados necessários ao correto funcionamento da aplicação devem ser persistidos em Base de dados.

5. Entrega

A entrega realizar-se-á na data indicada no calendário de avaliações e consiste na submissão (na página do moodle) de um ficheiro Zip com todo o código implementado (pasta da framework), respetivo SQL do projeto e um ficheiro Readme com as credenciais necessárias para acesso ao projeto e o link do repositório git.

O nome do ficheiro a entregar deve ter o seguinte formato:

PSI_PWS_PL<1/2>-<LetraGrupo>_<1ºNome.Apelido>_<1ºNome.Apelido>_<1ºNome.Apelido>.zip

Exemplo: PSI_PWS_PL1-A_Ana.Silva_Joao.Sousa_Lucio.Machado.zip

Após a entrega do projeto será realizada uma apresentação oral em grupo, bem como, uma defesa prática individual.

6. Critérios de avaliação

Nº	Peso	Componente
1	**30%**	Zona Reservada Passageiro
2	**10%**	Zona Reservada Operador de *Checkin*
3	**15%**	Zona Reservada Gestor de Voo

4	**15%**	Zona Reservada Administrador
5	**15%**	Correta implementação da infraestrutura em MVC
6	**15%**	Otimização, inovação e qualidade de implementação

Nos 4 primeiros critérios é analisado o produto resultante, ou seja, a aplicação web do ponto de vista da perceção do utilizador (e consequentemente a correta implementação no servidor). Neste caso, é necessário ter em conta aspetos funcionais, como o cumprimento da lógica de negócio e restantes requisitos da aplicação.

O critério 5 foca o elemento estrutural do projeto, isto é, a correta utilização do padrão estrutural MVC (Model-View-Controller).

No critério otimização, inovação e qualidade de implementação, serão tidos em conta aspetos relacionados com a implementação interna do sistema, e cumulativamente, inovações introduzidas pelos estudantes que melhorem a plataforma, a sua fluidez e/ou a qualidade da implementação interna da aplicação. Consideram-se inovações: a implementação de regras e funcionalidades não especificadas no enunciado e que tenham impacto na qualidade do projeto.

Para a qualidade da implementação interna contribuem entre outros aspetos: a arquitetura e estrutura do site, as classes/funções; a correta aplicação dos conceitos e técnicas da linguagem usada; a qualidade e fiabilidade do código produzido; a aplicação de técnicas para otimizar o desempenho da aplicação; a qualidade de reutilização do código produzido, entre outros.