



**Politécnico
Castelo Branco**

Escola Superior
de Tecnologia

Escola Superior de Tecnologia

Licenciatura – **Engenharia Informática**

Licenciatura – **Informática e Multimédia**

Unidade Curricular: **Linguagens de Programação para a Internet**

Unidade Curricular: **Linguagens Web**

2º Ano/ 2º Semestre

Docentes: Carlos Alves

Ano Letivo 2024/2025

Trabalho Prático de Avaliação – Parte PHP

Objectivos

Pretende-se com este trabalho desenvolver e aperfeiçoar competências de:

- pesquisa, de análise e de síntese relativamente ao tema proposto;
- colaboração e cooperação de equipa;
- expressão verbal e corporal perante audiências;
- elaboração de uma aplicação informática com usabilidade e acessibilidade.

Conteúdo

Em grupos auto formados de um ou dois elementos do mesmo curso. Deverão ser elaborados:

- uma apresentação, subordinada ao tema abaixo proposto, sendo suportada por uma aplicação;
- elaboração de uma aplicação informática (com o tema descrito de seguida).

A avaliação final de cada um dos trabalhos por parte do docente terá em conta a abordagem ao tema proposto e a sua apresentação.

Pretende-se o desenvolvimento de uma aplicação que permita a gestão de um “*web site* para gestão de viagens de autocarro: FelixBus”, com uma implementação em PHP.

A aplicação deverá suportar as seguintes funcionalidades/requisitos:

- Site *web* básico com os dados da empresa (localização, contactos, horários de funcionamento, etc...);
- Consulta de rotas, de horários e de preços (acessível a todos os perfis de utilizador);
- No *site* deverá ser possível apresentar alertas/informações/promoções dinâmicas, definidas pelos administradores.
- Área de Clientes:
 - Registo de cliente;
 - *Login/Logout*;
 - Consultar e editar dados pessoais;

- Consultar rotas;
- Gestão da carteira (consultar, adicionar saldo, levantar saldo);
- Gestão de bilhetes;
- Carteira:
 - Cada cliente tem uma carteira com saldo para poder comprar as suas viagens. No contexto deste trabalho, as funcionalidades de adicionar e retirar saldo devem ser implementadas apenas com um formulário simples em que o cliente diz quanto quer adicionar ou retirar;
 - Existe uma carteira especial, da própria FelixBus. Sempre que um cliente fizer uma compra, o pagamento é feito mediante a transferência do valor da carteira do cliente para a carteira da FelixBus;
 - As operações que envolvam a carteira devem ser registadas para efeitos de auditoria. Deverá ser registada toda a informação relevante para saber a operação realizada, a data, o valor, as carteiras envolvidas, etc...
- Bilhetes
 - Cada bilhete para uma determinada viagem deverá ter um identificador único que permita a posterior validação pelo motorista;
 - Um bilhete apenas é válido para a viagem (rota, data, hora) para a qual foi comprado.
- Área de cliente (só acessível a clientes):
 - Gestão de compra de bilhetes;
 - Gestão da sua carteira;
 - Visualizar e editar os seus dados pessoais.
- Área de gestão (acessível só a funcionários e administradores)
 - Gestão de saldo da carteira de clientes;
 - Gestão de bilhetes (é possível comprar bilhetes para clientes);
 - Visualizar e editar os seus dados pessoais.
- Área de administração (só acessível a administradores):
 - Gestão de rotas (considera-se uma rota como um percurso com uma origem e um destino, podendo haver mais do que um horário por dia para a mesma rota; os autocarros têm uma capacidade máxima);
 - Gestão de utilizadores;
 - Gestão de alertas/informações/promoções;
 - Visualizar e editar os seus dados pessoais.

Existirão os seguintes perfis de utilizadores: visitante; cliente; funcionário; administrador.

A aplicação deverá estar preparada para permitir adicionar novas funcionalidades. Estas funcionalidades adicionais não fazem parte do projeto de avaliação, mas a aplicação desenvolvida deverá estar estruturada de forma que outros programadores possam prosseguir o trabalho.

Pretende-se o desenvolvimento de uma aplicação *on-line* para o tema proposto, assim como a gestão do *site*. Todos os dados referentes à aplicação são armazenados em MySQL. Para o desenvolvimento da aplicação será utilizada a linguagem PHP, juntamente com MySQL. **Quando se faz referência a gestão (p.e., de utilizadores, de reservas, etc) deve ser entendida como: inserir, editar, visualizar, apagar/inactivar.** Todos os dados referentes à aplicação são armazenados em MySQL.

A aplicação deverá permitir, pelo menos, as funcionalidades anteriormente referidas, utilizando no desenvolvimento da aplicação a linguagem PHP, juntamente com MySQL. O grupo terá total liberdade no desenvolvimento da aplicação (pretende-se que seja uma aplicação prática e

minimamente realista), desde que utilize PHP e que cumpra funcionalidades e obrigações indicadas.

O código deverá ser estruturado de forma lógica, indentado e comentado, para facilitar a sua compreensão. Cada módulo/função/bloco funcional deverá incluir um comentário com uma breve descrição da sua funcionalidade e lógica de funcionamento. O código não indentado ou não comentado será excluído da avaliação.

À semelhança do que vos vai acontecer nos vossos percursos profissionais futuros, o presente enunciado poderá ser omissivo ou vago em alguns aspetos. A identificação de dúvidas e respetivo esclarecimento por parte da equipa de desenvolvimento faz parte do projeto e será alvo de avaliação. As dúvidas poderão ser apresentadas aos docentes para esclarecimento durante as aulas práticas ou nos horários de atendimento. Todas as dúvidas deverão ser documentadas e apresentadas no relatório final juntamente com os respetivos esclarecimentos.

Relatório

Deverá ser elaborado um relatório com as seguintes informações:

- Capa com a identificação completa dos alunos;
- Índice;
- Descrição sucinta das funcionalidades implementadas e opções tomadas;
- Matriz com a identificação de todos os requisitos da aplicação, juntamente com a indicação de se foram ou não implementados na aplicação entregue;
- Modelo de dados com a identificação de todas as tabelas e colunas e descrição sucinta do seu significado/finalidade;
- Resumo sucinto de dúvidas e esclarecimentos;
- *Print screens* da aplicação e respetiva legenda;
- Conclusão/justificação das decisões tomadas.

O trabalho tem que respeitar obrigatoriamente as seguintes indicações:

- é obrigatório apresentar um relatório com, pelo menos, “User Cases”, modelo relacional da base de dados, apresentação dos *screens* da aplicação, bem como conclusão/justificação das decisões tomadas;
- ter, **exactamente**, a seguinte estrutura de directorias: nome e apelido de cada aluno do grupo (p.e., “AntonioSilva_MariaFaustino”); “basedados” (dentro da directoria “AntonioSilva_MariaFaustino”); “paginas” (dentro da directoria “AntonioSilva_MariaFaustino”); “relatorio” (dentro da directoria “AntonioSilva_MariaFaustino”);
- a directoria “basedados” tem que ter os ficheiros “basedados.h”, “criar_bd.sql” e “apagar_bd.sql” (exactamente estes nomes);
- a directoria “paginas” tem que ter todo o código da aplicação (não pode ter sub-directorias);
- a directoria “relatorio” tem que ter o relatório **em formato pdf**;
- um *script* com o nome “criar_bd.sql” para criar a base de dados, criar as tabelas, inserir registos nas tabelas. Este *script* estará na directoria “basedados”;
- um *script* com o nome “apagar_bd.sql” para eliminar da base de dados todos os objetos relacionados com o projeto. Este *script* estará na directoria “basedados”;
- um módulo com o nome “basedados.h” para ligação à base de dados. O *connect* ao MySQL é feito **exclusivamente** usando as funções deste módulo (isto é, não pode aparecer em qualquer outra parte do código);
- existir, **exactamente**, os seguintes utilizadores e *passwords*, já existentes na base de dados e possíveis de efectuar *login* na aplicação: cliente/cliente; funcionario/funcionario; admin/admin;

- *login* de utilizadores com uso de sessões e respectiva **protecção de todas as páginas/scripts** de forma que apenas o utilizador com o respectivo perfil aceda a essa página/script;
- utilização de MySQL para armazenamento de dados;
- o código deverá estar devidamente indentado e comentado;
- o trabalho será enviado para os endereços de *email* identificados abaixo, até à data limite, sendo que em CC tem de constar todos os elementos do grupo. No email estará em anexo (em formato zip ou rar) o trabalho (não serão aceites os trabalhos cujo conteúdo não esteja em anexo, ou seja, o conteúdo não pode ser enviado através de WeTransfer, OneDrive ou similar);
- não é permitida qualquer consulta durante a avaliação;
- após a entrega do trabalho prático não será permitida qualquer alteração ao mesmo.

A defesa e apresentação do trabalho será, **exclusivamente**, num computador da sala de aulas práticas ou equivalente (p.e., no gabinete do docente, na impossibilidade de utilização da sala de aula prática), com uma duração de cerca de 20 minutos, durante esse tempo o grupo instala a aplicação, demonstra as funcionalidades e responde às perguntas colocadas pelo docente (a resposta é obrigatória, a não resposta ou resposta incorreta implica automaticamente a nota ser 0 valores).

Recomendações

Recomenda-se a criação de uma *checklist* com todos os requisitos identificados neste enunciado (mais os que eventualmente possam vir a surgir na sequência do esclarecimento de dúvidas).

Data da entrega

Consultar <https://www.ipcb.pt/sa/calendarios>.

- época de frequência: até ao final do dia indicado no calendário de avaliação por *email* cmoa@ipcb.pt
- época de exame/recurso: até ao final do dia do exame/recurso indicado no calendário de avaliação por *email* para cmoa@ipcb.pt

Data da apresentação

Consultar <https://www.ipcb.pt/sa/calendarios>.

- época de frequência no dia indicado no calendário de avaliação de acordo com a respectiva turma prática e na respectiva aula prática, sempre que possível;
- época de exame/recurso: a agendar após a entrega/recepção dos trabalhos e divulgado por *email/moodle* (em princípio, após 2 dias a data do exame/recurso).

Avaliação

Os critérios de avaliação estão detalhados num ficheiro *excel* que está disponível no portal Moodle juntamente com o presente enunciado.

Notas:

- A soma de cotações das funcionalidades solicitadas totaliza mais de 20 valores. Isto permite aos grupos optar por não implementar todas as funcionalidades solicitadas e dessa forma gerir o esforço a dedicar ao projeto, ou compensar alguma penalização resultante de erros identificados durante a avaliação. A soma de cotações satura nos 20 valores (ou seja, qualquer resultado acima de 20 será assumido como 20).
- Durante o processo de avaliação o avaliador poderá penalizar a cotação obtida pelas funcionalidades da aplicação, com base em vários critérios. A nota final será a nota resultante das funcionalidades implementadas (num máximo de 20 valores) descontada das penalizações que forem aplicadas.
- A avaliação é individualizada para cada elemento do grupo.