

Ano letivo: 2022/2023

Curso: Lic. Engenharia De Redes E Sistemas De Computadores

Unidade Curricular	Programação Web
--------------------	-----------------

Lic.	Ano do curso	2º ano	2º semestre	ECTS	
------	--------------	--------	-------------	------	--

NOME do ALUNO: Henrique Leonardo Bento Afonso

Prova Escrita

Versão: B

Duração: 100 minutos

Leia atentamente toda a prova antes de iniciar.

A prova é individual, não sendo permitido consultar os seus colegas. No entanto, pode consultar os apontamentos das aulas e a Internet.

O resultado final deve ser enviado para o moodle incluindo o Word da prova e PDF da prova (gravar como PDF) e os ficheiros HTML e JS desenvolvidos. Deve ser anexado o link para Github no tópico Avaliação.

No documento de resposta deve ser incluída a versão da prova.

Durante a resolução deve ir gravando o trabalho para salvaguardar as alterações.

Parte I

(25 valores)

1. À luz do que aprendeu na UC, comente a seguinte imagem.

Cofinanciado por:



UNIÃO EUROPEIA
Fundo Social Europeu

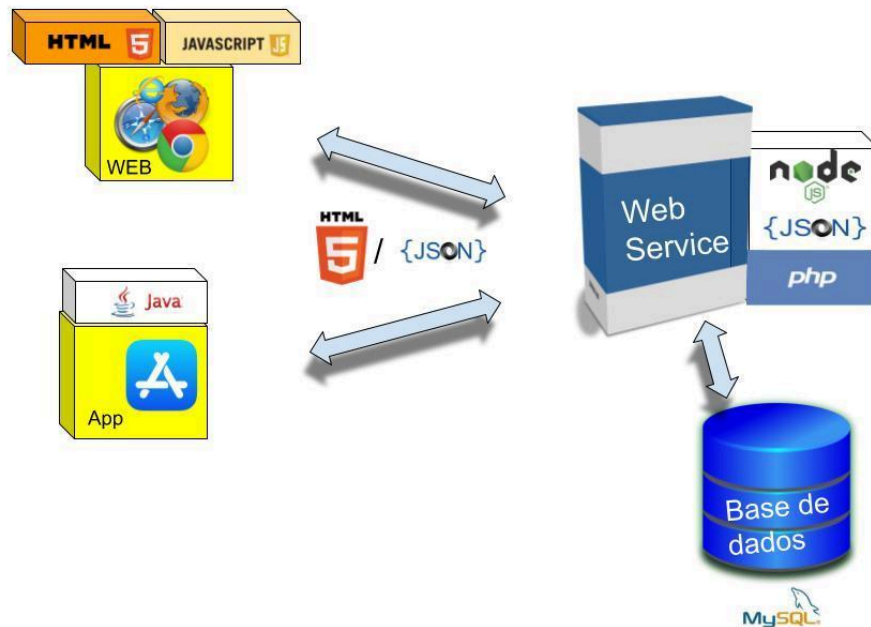


Figura 1 - Estrutura do documento

A imagem mostra a estrutura de uma aplicação web, indicando diferentes linguagens e tecnologias usadas em cada componente dessa estrutura. No entanto na área da web, falta CSS, e junto a JSON deveria estar XML invés de HTML.

Crie um protocolo para os alunos do IPVC para almoçar na cantina. Para que servem os protocolo e dê um exemplo

Parte II

(25 valores)

1. Considera os seguintes exemplos de objetos DOM.

- document.getElementById(id)
- document.getElementsByTagName(tagName)
- document.getElementsByClassName(className)

Cofinanciado por:

Porque no primeiro caso temos getElement e nos dois seguintes getElements? Dê um exemplo de utilização para cada exemplo

O primeiro caso refere-se a um elemento com o atributo id, este atributo tem de ser único, podendo o jQuery apenas encontrar um.

```
document.getElementById("exemplo")
```

```
document.getElementsByTagName("p")
```

```
document.getElementsByClassName("class")
```

2. Cria uma estrutura em JSON para registar Atores e Filmes. Faz um XML para a mesma estrutura. Comenta os resultados

O ficheiro JSON é muito mais simples, e mais fácil para um humano ler a informação

Parte III

(20 valores)

1. Qual a diferença entre <p> e <pre>

<p> é o elemento que cria parágrafos, sem preocupação com a formatação, <pre> mantém a formatação, mantendo espaços e parágrafos, mas também usa uma fonte com uma largura fixa

2. Para que serve

```
<meta charset="utf-8">
```

serve para que seja possível compreender caracteres que não os da língua inglesa, como no caso em português ç e ã

Parte IV

(30 valores)

1. Prepara uma página com uma tabela 2x2 com estilos CSS que permitam apresentar 4 marcas de produtos de rede. Usa cores de fundo e cores de escrita e o logotipo de cada marca.

Cofinanciado por:



UNIÃO EUROPEIA
Fundo Social Europeu

Parte V

(50 valores)

1. Usando o Bootstrap, construa uma página com cards que mostre 6 monumentos e atrações turísticas do seu local de residência.
2. Cada card tem de ter um botão “ver mais” para ver mais detalhes.

**

Parte VI

(50 valores)

Considere as imagens seguintes.

```
routes > JS products.js > ...
1  const productsRouter = require('express').Router();
2  const controller = require('../controllers/products');
3  const authMiddleware = require('../middlewares/auth/auth');
4
5
6
7
8
9
10
11
12  module.exports = productsRouter;
```

Figura 2 - Rotas

Cofinanciado por:



UNIÃO EUROPEIA
Fundo Social Europeu

```
controllers > JS products.js > ...
1  const apiResponse = require('../utils/response/apiResponse');
2  const Products = require('../data/entities/products');
3
4  > exports.getAll = async (req, res) => { ...
15 }
16
17 > exports.getById = async (req, res) => { ...
30 }
31
32 > exports.create = async (req, res) => { ...
49 }
50
51 > exports.update = async (req, res) => { ...
72 }
73
74 > exports.delete = async (req, res) => { ...
92 }
```

Figura 3 - Controller Produtos

1.1 - Complete o ficheiro de rotas dos produtos.

1.2 - Explique cada uma das linhas do ficheiro anterior

getAll, obtém todos os productos, getById, obtem um producto em especifico usando o id dado, create, cria um novo produto, update, atualiza as informações de um produto já criado, e delete, apaga um produto usando o id dado

1.3 - Desenvolva um ficheiro JSON que permita guardar a informação dos produtos e escreva o código para cada um dos métodos do controller products.

2. O Resultado final da prova escrita deve ser colocada no github sendo partilhado o link como resposta à prova

Cofinanciado por:





Bom trabalho!

António Lira Fernandes

Cofinanciado por:



UNIÃO EUROPEIA
Fundo Social Europeu