

**Engenharia Informática**

Licenciatura, 1º Ciclo

Plano: Despacho n.º16228/2009 - 15/07/2009

**Ficha da Unidade Curricular: Tecnologias da Internet I**

ECTS: 6; Horas - Totais: 165.0, Contacto e Tipologia, PL:70.0; OT:5.0; O:5.0;

Ano | Semestre: 1 | S2

Tipo: Obrigatória; Interação: Presencial; Código: 911910

Área Científica: Sistemas de Informação

**Docente Responsável**

José Manuel Palma Redes Ramos

Professor Coordenador

**Docente(s)**

José Manuel Palma Redes Ramos

Professor Coordenador

**Objetivos de Aprendizagem**

- A. Desenvolver código HTML, CSS e javaScript;
- B. Codificar conteúdos Web estáticos;
- C. Codificar conectividade Web;
- D. Codificar gráficos Web;
- E. Implementar modelos de interação Web;
- F. Implementar interfaces Web dinâmicos;
- G. Implementar projetos Web do lado do cliente.

**Objetivos de Aprendizagem (detalhado)**

1. Desenvolver código HTML, CSS e javaScript;
2. Codificar conteúdos Web estáticos e formatados: textos, imagens, templates e layouts;
3. Codificar conectividade Web (interna, externa e mista);
4. Codificar interfaces Web interativos: comportamentos, pares evento/ação, modelos de input/output;
5. Codificar gráficos Web: programação do objeto Canvas;

6. Aplicar estruturas de dados em JavaScript: arrays, objetos e estruturas mistas;
7. Aplicar metodologias de HTML/CSS Dinâmicos e de Document Object Model (geração de objetos de interface através do objeto Document);
8. Aplicar interação gráfica e cinemática: processos de interação e de animação de componentes do interface Web;
9. Implementar interfaces Web, gráficos e interativos;
10. Implementar interfaces Web a partir de Front-end Frameworks;
11. Implementar processos através de bibliotecas e frameworks de JavaScript;
12. Implementar projetos de aplicações Web, do lado do cliente, com funcionalidades gráficas, dinâmicas e interativas, a partir de fontes diversificadas de dados e de recursos.

### **Conteúdos Programáticos**

1. Programação com HTML, CSS e JavaScript;
2. Superclasses, subclasses e instâncias em HTML;
3. Codificação de conteúdos e interfaces em HTML5;
4. Codificação de estilos em CSS;
5. Estruturas de dados em JavaScript;
6. Classes de JavaScript;
7. Dinâmica, Interação e Comportamento;
8. Gráficos Web bitmap (o objeto Canvas);
9. Técnicas de animação;
10. Metodologias de Projeto Web;

### **Conteúdos Programáticos (detalhado)**

Não Aplicável

### **Metodologias de avaliação**

A avaliação incide sobre a apresentação e defesa de três mini-projetos Web do lado do cliente:

- a) Implementação de uma apresentação interativa de conteúdos a partir de uma estrutura de dados estabelecida - 35% da avaliação;
- b) Implementação de uma aplicação web interativa com características gráficas ou de animação - 35% da avaliação.
- c) Implementação de uma aplicação web interativa com exigências algorítmicas - 30% da avaliação.

O processo de avaliação é realizado em sessão de apresentação dos mini-projetos e a aprovação na Unidade Curricular está sujeita à classificação mínima de 7/20 valores em qualquer dos mini-projetos.

### **Software utilizado em aula**

Editor Notepad++; Browser Google Chrome;

### **Estágio**

### **Bibliografia recomendada**

- Haverbeke, M. (2018). *Eloquent JavaScript - A Modern Introduction to Programming*. No Starch Press. San Francisco
- Geary, D. (2012). *Core HTML5 Canvas: Graphics, Animation, and Game Development*. Prentice-Hall. Boston:
- Terlson, B. e International, E. (0). *ECMAScript 2018 – Language Specification (9.th Edition, June 2018)* Acedido em 10 de novembro de 2020 em <https://262.ecma-international.org/9.0/>
- Data, R. (0). *w3schools.com - THE WORLD'S LARGEST WEB DEVELOPER SITE* Acedido em 10 de novembro de 2020 em <https://www.w3schools.com/>

### **Coerência dos conteúdos programáticos com os objetivos**

Objetivo A: Conteúdos 1, 2, 3, 4, 5, 6.

Objetivo B: Conteúdos 1, 2, 3.

Objetivo C: Conteúdos 1, 2, 3.

Objetivo D: Conteúdos 1, 2, 3, 4, 5, 6, 7, 8.

Objetivo E: Conteúdos 1, 2, 3, 4, 5, 6, 7, 8.

Objetivo F: Conteúdos 1, 2, 3, 4, 5, 6, 7, 8, 9.

Objetivo G: Conteúdos 1, 2, 3, 4, 5, 6, 7, 8, 9, 10.

### **Metodologias de ensino**

1. Aulas teórico-práticas plenárias com exposição e debate - 28 horas;
2. Aulas de prática laboratorial por turmas em ambiente de desenvolvimento de software - 42 horas;
3. Apoio didático por e-learning, através da plataforma Moodle.

### **Coerência das metodologias de ensino com os objetivos**

A aplicação de uma dimensão teórico-prática do processo de aprendizagem segundo um método expositivo, ilustrativo e demonstrativo, com utilização de meios audiovisuais avançados, com a permanente participação dos alunos, é coerente com a complexidade dos conteúdos de suporte para a consecução dos objetivos, como se demonstra a seguir.

Trata-se de consolidar a aquisição do conhecimento teórico complexo, através de observação participativa das respetivas aplicações, tanto à priori como à posteriori da sua aplicação individual por cada aluno. O professor apresenta as fontes de consulta e demonstra os processos de desenvolvimento e implementação de pequenos projetos que os alunos deverão treinar nas suas atividades autónomas. Os alunos apresentarão as dúvidas e dificuldades. Professor e alunos colaboram na interpretação dos enunciados dos pequenos projetos e na análise e decomposição das suas fases de implementação.

A aplicação de uma dimensão de prática laboratorial do processo de aprendizagem segundo um ambiente de implementação de pequenos projetos evolutivos de programação Web, é coerente com a prossecução de todos os objetivos, como se demonstra a seguir.

Trata-se de formar e consolidar competências de implementação efetiva de pequenos projetos

Web, que incluem a interpretação dos enunciados de pequenos projetos, a análise e decomposição das suas fases de implementação e a sua concretização, com recurso a repositórios de recursos mediáticos (textos, imagens, sons, vídeos, websites) e a linguagens de marcação (HTML5 e XML), de estilização (CSS) e de programação (JavaScript), com especial incidência para esta última. Remetem essencialmente para a formação de competências duras (hard skills) de desenvolvimento e implementação de pequenos projetos Web. É essencial a vivência em ambiente laboratorial com uso de recursos computacionais, incluindo editores de código e web-browsers.

Conclui-se assim pela coerência das metodologias de ensino com o Objetivo Geral de “Implementar projetos Web do lado do cliente”.

### **Língua de ensino**

Português

### **Pré-requisitos**

Não Aplicável

### **Programas Opcionais recomendados**

Não Aplicável

### **Observações**

Não Aplicável

Objetivos de Desenvolvimento Sustentável:

- 4 - Garantir o acesso à educação inclusiva, de qualidade e equitativa, e promover oportunidades de aprendizagem ao longo da vida para todos;
- 9 - Construir infraestruturas resilientes, promover a industrialização inclusiva e sustentável e fomentar a inovação;

---

### **Docente responsável**

---