

TeSP - Tecnologias e Programação de Sistemas de Informação

Técnico Superior Profissional

Plano: Despacho n.º 12805/2021 - 29/12/2021

Ficha da Unidade Curricular: Tecnologias da Internet

ECTS: 6; Horas - Totais: 162.0, Contacto e Tipologia, TP:80.0;

Ano | Semestre: 1 | A

Tipo: Obrigatória; Interação: Presencial; Código: 60248

Área de educação e formação: Ciências informáticas

Docente Responsável

Tiago Manuel Henriques Carrão

Assistente Convidado

Docente(s)

Tiago Manuel Henriques Carrão

Assistente Convidado

Fábio Miguel Lopes Ferreira

Assistente Convidado

Objetivos de Aprendizagem

Aplicar tecnologias de programação, do lado do cliente, para produção de conteúdos, aplicações e interfaces Web dinâmicos, com respeito às mais recentes normas e recomendações do Consórcio W3C.

Objetivos de Aprendizagem (detalhado)

- A. Descrever a estrutura de um documento Web;
- B. Descrever as componentes e a estrutura do DOM (Document Object Model);
- C. Descrever as classes de objetos Web: propriedades e métodos;
- D. Desenvolver código em XHTML5, CSS3 e JavaScript;
- E. Desenvolver código através da biblioteca JQuery;
- F. Desenvolver aplicações gráficas;
- G. Implementar conteúdos, interfaces e aplicações Web;
- H. Desenvolver manipulação dinâmica de objetos Web;

- I. Implementar conteúdos, interfaces e aplicações Web interativas, reativas, gráficas e animadas.
- J. Conhecer IDEs disponíveis (Sublime Text, Visual Studio Code, etc) para desenvolvimento em HTML, HTML5, CSS, CSS3, Bootstrap.

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)

1. Normas do Consórcio W3C e evolução das tecnologias WEB; a Linguagem HTML5; Estrutura HTML; Estilos CSS; o "Document Object Model" para os objetos Web; a 5.^a Norma Restrita; representação de objectos Web;
2. Estrutura, organização, hierarquia e atributos dos objetos Web; sintaxe das "tags"; atributos de identificação e atributos de subclassificação;
3. Codificação de objetos de conteúdo, com cabeçalhos, parágrafos, listas e tabelas; codificação de estruturas de continência com divisórias; gestão de hierarquias de apresentação;
4. Linguagem CSS: sintaxe dos seletores; classes, subclasses, pseudo-classes e objectos; sintaxe, atributos e respetivos domínios de atribuição; hierarquias dos padrões de estilo;
5. Representação de dados complexos em javaScript, integrando átomos (estados booleanos, números e cadeias de caracteres) em objetos e em listas e em estruturas combinadas;
6. Linguagem javascript: sintaxe; propriedades e métodos das classes do DOM; classes de javaScript: Propriedades e métodos; gestão de formulários; técnicas de "motion"; o elemento Canvas; produção de dinâmica de conteúdos; Hierarquias funcionais; interfaces e aplicações WEB com características interativas, gráficas e dinâmicas;
7. Eventos, escutadores de eventos e desencadeamento de acções por eventos; comportamentos; Manipulação dinâmica de objetos Web via os seus conteúdos e atributos de estilo; comportamentos; interação avançada;
8. O objeto Canvas: contextos gráficos, métodos e propriedades gráficas; atributos geométricos e atributos gráficos; funções gráficas;
9. Tipologias de Animação: autónoma, interativa e guiada; iterações, esperas e paragens;
10. Desenvolvimento de projetos, com apresentação dinâmica de conteúdos, com exigências algorítmicas e com interação gráfica.

Metodologias de avaliação

A avaliação incide sobre a apresentação e defesa de 2 mini-projetos web do lado do cliente:

1. Implementação de um website (HTML e CSS) - 40%.
2. Implementação de um website interativo (HTML, CSS, JS) - 60%.

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

Visual Studio Code; Browser (Google Chrome, Opera, etc.)

Estágio

Não Aplicável

Bibliografia recomendada

- Data, R. (0). *w3schools.com - THE WORLD'S LARGEST WEB DEVELOPER SITE*.Acedido em10 de novembro de 2020 em <https://www.w3schools.com/>
- Geary, D. (2012). *Core HTML5 Canvas: Graphics, Animation, and Game Development*.. Prentice-Hall. Boston:
- Haverbeke, M. (2018). *Eloquent JavaScript - A Modern Introduction to Programming*.. No Starch Press. San Francisco
- Terlson , B. e International, E. (0). *ECMAScript 2018 – Language Specification (9.th Edition, June 2018)*.Acedido em10 de novembro de 2020 em <https://262.ecma-international.org/9.0/>

Coerência dos conteúdos programáticos com os objetivos

- Objetivo A: Conteúdos 1, 2, 3, 4.
- 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 5, 6, 7.
- Objetivo F: Conteúdos 8, 9 , 10.
- Objetivo G: Conteúdos 1, 2, 3, 4, 5, 6, 7, 8, 9, 10.
- Objetivo H: Conteúdos 9, 10.
- Objetivo I: 8, 9, 10.
- Objetivo J: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10.

Metodologias de ensino

1. Aulas de prática laboratorial por turmas em ambiente de desenvolvimento de software;
2. 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
