



TeSP - Tecnologia e Programação em Sistemas de Informação

Técnico Superior Profissional

Plano: Plano TeSP

Ficha da Unidade Curricular: Tecnologias da Internet

ECTS: 6; Horas - Totais: 162.0, Contacto e Tipologia, PL:75.0; OT:7.5;

Ano|Semestre: 1|S2; Ramo: Tronco comum;

Tipo: Obrigatória; Interação: Presencial; Código: 60248

Área de educação e formação: Ciências informáticas

Docente Responsável

José Manuel Palma Redes Ramos

Professor Coordenador

Docente e horas de contacto

José Manuel Palma Redes Ramos

Professor Coordenador, PL:75.0; OT:7.5;

Objetivos de Aprendizagem

1. Desenvolver código HTML, CSS e JavaScript;
2. Manipular atributos e conteúdos dos objetos do DOM do interface Web;
3. Desenvolver funcionalidades gráficas e cinemáticas interativas;
4. Desenvolver aplicações dinâmicas sobre interfaces Web;
5. Elaborar projetos Web do lado do cliente.

Conteúdos Programáticos

1. Codificação em HTML5;
2. Codificação em CSS, normas 1, 2 e 3;
3. O DOM (Document Object Model) Web;
4. Programação em JavaScript;
5. Aplicação de Bibliotecas jQuery e Bootstrap;
6. Geração de Gráficos Bitmap (Raster) através do elemento Canvas e de Gráficos Vetoriais através de SVG (Scalable Vector Graphics);
7. Interatividade e Animação;
8. Projeto de interfaces Web dinâmicos.

Conteúdos Programáticos (detalhado)

1. Codificação em HTML5 (HyperText Markup Language, norma 5): Estrutura geral das árvores de objetos; objetos vazios, objetos contedores e objetos arborescentes; Sintaxe das "tags" e respetivos atributos; Atributos especiais, de classificação ("class"), de identificação ("id") e de estilização ("style"); Atributos de comportamento para descrição de pares evento/ação; Categorização e representação dos diversos objetos Web.
2. Codificação em CSS1,2,3 (Cascading Style-Sheets, normas 1, 2 e 3): O objeto "style"; Descrição e sintaxe das representações CSS dos atributos do objeto "style"; O Modelo da Caixa; Representação de Superclasses, de Classes, de Pseudoclasses e de Objetos; Agrupamentos, especializações e exceções; Codificação decimal e hexadecimal das cores; Atributos avançados de estilo.

3. O DOM (Document Object Model) Web: Estrutura hierárquica do documento Web (objeto "document"), como uma árvore de objetos para representação e apresentação de conteúdos; Classes de objetos; Geração dinâmica; Gestão de atributos; Anexação, Clonagem e Extinção.
4. Programação em JavaScript: Sintaxe geral da linguagem; Âmbito das variáveis (públicas e privadas); Modelos de variáveis (lógico, numérico, string, array, objeto); Funções e respetivos argumentos; Métodos para cada classe de objetos do DOM; Funções de Expressão; Funções de Input, de Processo e de Output; Implementação de ações para pares evento/ação; Manipulação de atributos para cada classe de objetos do DOM; Processamento de formulários de input/output; Técnicas de "motion"; Mecanismos de interação.
5. Aplicação de Bibliotecas JQuery e Bootstrap: Funções das Bibliotecas; O seletor "\$" da jquery; Animação; Interfaces dinâmicos; Interação avançada; Aplicação dos recursos gráficos, de estilo e funções das bibliotecas Bootstrap na implementação de interfaces avançados.
6. Geração de Gráficos Bitmap (Raster) através do elemento Canvas e de Gráficos Vetoriais através de SVG (Scalable Vector Graphics): O contexto "2d" de um objeto "canvas"; Utilização dos métodos e propriedades gráficas desse contexto, através de programação javascript; Implementação de gráficos estáticos e dinâmicos e de animações gráficas de natureza "raster"; Representação XML de primitivas gráficas SVG; Geração de árvores XML de descritores SVG, através de programação javascript sobre o XML-DOM; Implementação de gráficos estáticos e dinâmicos e de animações gráficas de natureza vetorial;
7. Interatividade e Animação: comportamentos e pares evento-ação; Técnicas de 'framing'; Temporização e sincronização; Trajetórias.
8. Projeto de interfaces Web dinâmicos: Hierarquia dos objetos; Hierarquia dos processos; Modelos de interação; Animação e Storyboarding.

Metodologias de avaliação

Desenvolvimento de um projeto individual, consistindo em aplicações web do lado do cliente, a partir da enunciação de um modelo de requisitos.

Software utilizado em aula

Editor Notepad++; Browser Google Chrome;

Metodologias de ensino

1. Sessões práticas de programação em Laboratório de TIC;
2. Análise e desenvolvimento de miniprojetos;
3. Apoio didático por e-learning, através da plataforma Moodle.

Língua de ensino

Português

Docente Responsável



Diretor de Curso, Comissão de Curso



Conselho Técnico-Científico

