Curso: Especializado Front-End Web Development

Duração: 183h
Área formativa: Cursos

Sobre o curso

Os programadores, ou developers, têm sido dos profissionais mais procurados no mercado de trabalho, com salários acima da média.

Um Front-End Web Developer é um programador que codifica o código de Front-end de um site e passa por converter ficheiros de design de sites em HTML, JavaScript (JS) e / ou código CSS.

O domínio de uma série de tecnologias associadas à função de Front-End Web Developer permite uma melhor visibilidade e uma melhor experiência de utilizador permitindo assim a construção de uma web mais acessível a partir da correta interpretação do briefing, do planeamento, elaboração básica de wireframes, da marcação semântica do código e da construção de elementos interativos.

Enquanto Front-End Web Developer serás responsável por garantir que o front-end visual de um site esteja livre de erros e se pareça exatamente com o que foi projetado e que tenha a mesma visibilidade em diferentes computadores e sistemas operativos.

Porquê apostar numa carreira em Programação Web?

#Empregabilidade O mercado está cheio de ofertas de emprego para developers de soluções web nos mais variados domínios e a possibilidade de emprego além-fronteiras é uma realidade.

#Versatilidade Na área de desenvolvimento web tens variadas opções de carreira, ditadas pela linguagem de programação em que te especializares ou meio para o qual programares.

#Valorização Ao complementares a tua formação de base com conhecimentos de programação estarás a valorizar o teu currículo e percurso formativo, com uma forte competência digital, que te irá distinguir enquanto profissional com visão tecnológica.

#Reconhecimento De acordo com o relatório Tech Careers 2020 da Landing.jobs, o salário médio do perfil de programador aumentou mais do que 15% nos últimos três anos.

Por onde deves começar?

Com o curso **Especializado Front-End Web Development** vais:

:: **Dominar todos os conceitos-base** necessários ao desenvolvimento da interface gráfica de um

website e solução de problemas de front-end;

:: **Conhecer as melhores práticas no mercado** da programação web, a nível global;

:: **Estudar as principais linguagens de programação, de estilo e interfaces/softwares usados no desenvolvimento web:** HTML e CSS, Javascript, frameworks como React e ferramentas de versionamento como o GitHub;

:: **Construir o teu portfólio** profissional através dos projetos desenvolvidos ao longo do curso, para que possas demonstrar o teu know-how, criatividade e capacidade de execução às entidades empregadoras.

Este curso pode ser totalmente adaptado em conteúdos, duração, datas e horários às tuas necessidades pessoais ou para um grupo de colaboradores da tua empresa. Queres saber mais sobre as nossas soluções de formação à medida?

Objectivos

Ao terminares este curso poderás trabalhar em qualquer empresa que desenvolva (para os seus clientes internos ou externos) produtos web, desempenhando funções de:

:: Front-end Developer, enquanto profissional responsável por garantir que o front-end visual de um site esteja livre de erros e se pareça exatamente com o que foi projetado, e que tenha a mesma visibilidade em diferentes computadores e sistemas operativos.

:: Web Developer, enquanto programador especializado no desenvolvimento de sistemas para a internet, responsável pelo perfeito funcionamento de um website ou aplicação web.



Ao concluíres com aproveitamento esta formação, receberás um Certificado de Formação (acreditado pela DGERT) e um badge digital (da Credly) para partilhares com a tua rede profissional online.
Sabe mais sobre os nossos badges digitais aqui.

Pré-requisitos

Para ingressares no curso **Especializado Front-End Web Development** deverás ter conhecimentos de informática na ótica do utilizador, mas também raciocínio lógico, domínio do inglês, orientação para a resolução de problemas e uma mente criativa e curiosa.

Destinatários

O **Especializado Front-End Web Development** destina-se a todos os interessados em iniciar ou consolidar uma atividade profissional como Front-End Web Developers.

Este curso é indicado para ti se:

:: Pretendes iniciar uma carreira na área de programação, e procuras **complementar o currículo com formação prática** que te ajude a **integrar** rapidamente no **mercado de trabalho**;

:: És profissional na área de programação – software ou applications developer – e pretendes **melhorar** o teu **portfólio**, alargar os teus horizontes e **enriquecer** as tuas **competências** com valências na área de **desenvolvimento web para front-end**, ou fazer uma **atualização** aos teus **conhecimentos**;

:: És um empreendedor e pretendes **dar forma aos teus projetos**;

:: Procuras oportunidades de **networking com profissionais ativos no mercado da programação web**.

Este curso pode ser totalmente adaptado em conteúdos, duração, datas e horários às tuas necessidades pessoais ou para um grupo de colaboradores da tua empresa.
Queres saber mais sobre as nossas soluções de formação à medida?

Programa

Introdução à Programação e Desenvolvimento Web

Fundamentos de programação

- O que é programação.
- Programação vs Linguagem
- Conceitos de “ordem”
- Usando o MIT Scratch como ferramenta de aprendizagem

Algoritmos

Algoritmos computacionais

Pseudocódigo

Arquitetura da Web

- O que é a Web: cliente e servidor, como a web funciona
- IP’s, Servidores, DNS
- Servidores e Cloud
- Alojamentos

Introdução ao Front-End e Back-End

- Introdução ao front-end e back-end: conceitos e responsabilidades
- Como funciona o HTML, CSS e JavaScript
- Como funciona o back-end: servidores e linguagens comuns
- Conceitos de base de dados

HTML e CSS

Introdução ao HTML

- O que é o HTML
- Evolução do HTML até ao HTML5
- As 3 linguagens base da Web
- Para que serve o HTML
- Ferramentas Úteis

Sintaxe do HTML

- Escrita do elemento base
- Atributos
- Regras de construção
- Estrutura base de uma página HTML

Elementos de texto e Imagem

- Aprender elementos como títulos
- Hierarquia dos headings
- Parágrafos
- Elementos de formatação de texto (bolds, itálicos etc)
- Criação de imagens

Hiperligações

- Links absolutos vs relativos
- Anchor vs Button
- Atributos do elemento <a>
- Acessibilidade no HTML
- Links com imagens

Listas

- Tipos de listas (ordenadas e não ordenadas)
- Regras de construção das listas
- Menus (listas com links)
- Atributos da lista

Tabelas

- Tag table
- Construção em linhas
- Elementos constituintes da tabela: thead, tbody e tfoot
- Células em múltiplas linhas ou múltiplas colunas

Elementos multimédia

- Formatos de ficheiros. Quais existem e para que servem
- Elemento <video> e <audio>
- Acessibilidade nas imagens, e nos elementos de vídeo e áudio dentro do html

Iframes

- Iframe
- Como funciona um iframe dentro do html
- Exemplos de incorporação de Youtube e Google Maps

Formulários

- Definição form
- Limitação de formulários dentro do html
- Tag <form>
- Tag <input> e os diversos tipos
- Atributo “name”
- Relação das labels com os inputs
- Elemento Textarea
- <select> e <option> dentro de um formulário
- Como submeter um formulário

Elementos Estruturais do HTML

- O elemento básico da , o que é e para que serve
- Elementos de HTML5 estruturais e com semântica (<section>, <nav> etc)
- A acessibilidade dentro de uma página
- A importância da sintaxe

Regras e boas práticas de acessibilidade

- O que é o ARIA
- A importância da acessibilidade
- Como melhorar a acessibilidade com atributos simples de html

Exercício prático de implementação

- Implementação de um layout básico, de forma a cimentar os conhecimentos de HTML

Introdução à CSS

- O que é a CSS
- Regra do “Cascading” na folha de estilos

Utilizar CSS em páginas web

- Formas de adicionar CSS
- Como o CSS afeta a web
- Várias folhas de estilo na mesma página

Seletores CSS

- Sintaxe de um selector e as suas propriedades
- Selectores de Elementos
- Selectores de Classes
- Selectores de ID
- Selector Universal
- Selectores de Atributos
- Selectores compostos/complexos

Especificidades dentro do CSS

- Regra de especificidades entre elementos, classes e IDs
- A Regra da cascata
- O CSS inline
- A regra do !important

Unidades de Medida

- Entender unidades absolutas e relativas
- Conhecer as várias unidades na CSS como px, rem, e vw
- Percentagens dentro da CSS

Cores

- Sistemas de cor na web
- Propriedades que podem receber cor
- Como trabalhar unidades de cor

Trabalhar o texto

- Propriedades afetas ao texto
- A diferença entre trabalhar o aspeto do texto e o significado do elemento de HTML

Tipografia

- Alterar a font do nosso site ou elemento
- Adicionar fonts google no documento
- Adicionar fonts “custom” com o @fontface

Backgrounds

- Trabalhar o background-color

- Trabalhar as background images e todas as propriedades associadas
- Background-attachment
- Como criar gradients
- Shorthand de background

Borders e shadows

- Como funciona as borders no CSS, e que espaço ocupam
- Shorthand para borders
- Como funcionam as shadows
- Trabalhar as shadows de forma eficaz
- Trabalhar cantos arredondados

Box Model

- O conceito de box-model
- A propriedade box-sizing
- Como funciona o padding
- O que é a Margin

Positions em CSS

- Tipos de posicionamento (Absoluto, Relativo, Fixo etc)
- Como as propriedades Top, Bottom, Right e Left são afetadas pelo tipo de posicionamento

Tipos de Displays

- Os vários valores de display no CSS (block, inline, flex, grid etc.)
- Qual a diferença entre Flex e Grid em relação aos outros

Métodos de layout em CSS

- Entender o display flex
- O que são flex-childs
- A Gap
- Propriedades do Flex container
- Propreidades do Flex Child
- A CSS Grid, o que é e porque é diferente do flex
- Como criar uma grid com colunas
- Como posicionar elementos dentro das colunas
- Como ocupar mais de um espaço dentro da minha grid

Pseudo-classes e pseudo-elementos

- O que são Pseudo-classes como o hover
- Os pseudo-elementos ::before e ::after

Overflow e Z-indez

- Entender o que é a propriedade overflow e como pode ser trabalhada na construção de layouts
- O que é a propriedade z-indez, e como é afetada pelo tipo de posicionamento dos elementos

Transforms, Transitions e Animations

- Propriedade de Transform, e os seus diferentes valores
- Usar as transitions para alterações de estado
- As @keyframe animations e como se usam

Workshop: Versões de Código com Github

O que é um Controlador de versões

- Objetivo
- As mais conhecidas filosofias de versionamento de código
- Filosofia do controlo de versões recorrendo a Git
- Vantagens da abordagem GIT

Instalação e uso do software Git

- Instalação do GIT
- Configurações e autenticação
- Controlo de versões de código
- Partilha de informação entre diferentes elementos da equipe
- Repositórios
- Conceito de Branch e casos de utilização

Utilização do Git para controlo de versões de código

- Boas práticas para criação de repositórios
- Instrução add
- Instrução push
- Instrução commit
- Instrução merge
- Resolução de problemas
- Criação e gestão de um novo branch

Conhecer o GitHub

- Criação e configuração de conta no Github
- Criação e Gestão de projetos
- Criação e Gestão de utilizadores e equipas
- Visibilidade de repositórios
- Análise de histórico de projeto
- Controlo de qualidade recorrendo a ferramenta Github

Javascript para Frontend

Introdução, história e evolução dos vários standards da linguagem

Conceito de variável e Tipos de Dados numa linguagem de programação (Variáveis, constantes, …)

Conceitos / Instruções Básicas em javascript

Expressões

Comentários

Variáveis e constantes

Operadores

Tipos de dados primitivos em Javascript

Numbers, Strings, Boolean, null, undefined

Estruturas de decisão e repetição em javascript

Estruturas de decisão no controle do fluxo de um programa: (If, if/else, switch)

Estruturas de repetição: (For, While, Do/While)

Tipos de dados Complexos, Arrays, Objectos, Funções e Built-In Objects

Utilização de Arrays

Criação de Objectos, propriedades e métodos

Conceito de Função, funções com parâmetros

Funções anónimas e IIFE (Imediate Invoked Function Expression)

Funções “construtoras” de Objectos

Conceito de “Scope” de uma variável, varáveis locais e globais, diferenças em ES6 com let e const

Built-In Objects em JS

Object Oriented, Imperative e Functional Programming e como usá-las em Javascript

Browser Object Model (BOM)

Document Object Model (DOM)

Global Objects (Number, String, Boolean, Undefined, Null, Object, Math, Date, …)

Conceito e manipulação do DOM (Document Object Model) numa aplicação web

Como o Browser interpreta o HTML e cria um Objecto (DOM), representativo dos elementos da página

Tipos de elementos (Nodes) do DOM (Document, Element, Attributes e TextNodes)

Processo e Conceito de manipulação do DOM

Métodos para selecionar um elemento(node) ou uma coleção de elementos(nodelist) do DOM (Live e static methods)

Propriedades e métodos para percorrer o DOM

Alterar conteúdos de elementos do DOM

Inserir, mover ou eliminar elementos do DOM

Manipulação de atributos de elementos do DOM

Trabalhar com Eventos em Javascript

Conceito de interactividade com Eventos em JS

Categorias (tipos) de Eventos em JS

Definição de Eventos em JS (HTML event handler's, Event Handler's, DOM EventListeners)

Conceito de Event Flow e suas fases em JS (Capturing, targeting e bubbling)

Conceito de Event Object em JS

Delegação de Eventos em JS

Introdução à especificação ES6

Evolução do standard de javascript e novas especificações ES6, ES7 e ES8

Utilização de "Transpilers" como o BabelJS para a compilação de ES6 em "Vanilla Javascript" standard (ES5)

Variáveis e parâmetros em ES6 (let e const, Destructuring, Default Parameters, Classes, Rest e Spread operators, multiline e template strings,…)

Arrow functions, Iterators, e outros conceitos de Functional programming

Novos métodos de manipulação de Arrays e Objectos (map, reduce, Object.Assign(), …)

AJAX e JSON com Javascript (ES6)

Conceito de Ajax com JS (requests e responses)

Tipos de dados numa comunicação assíncrona com o servidor

Conceito de Promises (ES6)

Utilização da "Fetch API" para chamadas assíncronas.

Gestão de erros em Javascript

Filtrar, Ordenar e pesquisar numa estrutura de dados em JS

Introdução e utilização de algumas API's de HTML5 (LocalStorage, SessionStorage, Geolocation, …)

React Foundations

A Biblioteca React.js

- A organização do código tendo em vista os componentes
- Introdução a biblioteca de React,js
- Princípios de arquitetura da biblioteca React.js
- Introdução a organização do código tendo em conta a arquitetura do React.js
- Criação de um projeto React.js

- Organização de um projeto React.js
- Criação de um componente base
- Introdução ao conceito de SASS
- Debug de aplicação React.js

JSX

- O que é o JSX
- Vantagens do uso do JSX
- Estruturar programa para desenvolvimento em JSX

Componentes React.js

- Components and properties
- Componente lifecycle
- Tipos de components
- Props e Classes
- Nested Components
- Handling Data
- Use Effect e React.js ho

React.js

- Uso do contexto em React.js
- Nested Components
- Databind em aplicação React.js
- Passagem de parâmetros entre componentes
- Partilha de contexto em react.js
- Gestão de estado em react.js
- Pedidos assíncronos em React.js
- Forms validation
- React.js Hooks

Organização de código em Single page applications baseadas em React.js

- Introdução as boas práticas de arquitetura e implementação de SPA’s
- Identificação de erros comuns
- Use cases de implementação e arquitetura baseados em cenários reais

Debuging em React.js

- Devtools em React.js
- Debuging de hierarquia de componentes
- Debuging de propriedades
- Debuging de funções em aplicações React.js

React Avançado

Gestão de estado entre componetes react

- Soluções out of the box da biblioteca react
- Soluções baseadas em arquitetura e hierarquia de componentes

Redux

- Introdução ao Redux
- O Redux como meio de comunicação entre componentes
- Implementaçães baseadas em Redux
- Componentes Redux (Store, Action, Reduer e Dispatcher)

Debugging em redux

- Devtools para redux
- Técnicas de debug em soluções baseadas em redux

Pitstop

Projeto: Aplicação interactiva com React