

# Academia FLAGProfessional Full Stack Web Development

- **Formato do curso:** Presencial e Live Training
- **Localidade:** Porto
- **Data:** 06 Set. 2022 a 06 Jun. 2023
- **Preço:** 4250€
- **Horário:** Pós-laboral e Sábados - 3<sup>as</sup>, 5<sup>as</sup> das 18h30 às 21h30 e sábados das 10h00 às 17h00
- **Duração:** 366 horas

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

A elevada empregabilidade, associada aos desafios tecnológicos com que as organizações se deparam, tornaram a carreira de Full-stack Web Developer muito apelativa.

Enquanto full stack web developer irás apresentar aptidões multi-tasking no que diz respeito ao desenvolvimento de produtos digitais, pois estarás familiarizado com todas as fases do processo de criação e design dos produtos. Serás assim um elemento-chave na cooperação e otimização de equipas, ajudando na redução de custos e tempos de produção, e estarás mais capacitado a identificar e prevenir situações inesperadas.

Na **Academia FLAGProfessional Full Stack Web Development** irás aprender a trabalhar com as ferramentas necessárias para assumires uma atividade profissional ligada à programação de back-end e front-end, enquanto back-end, front-end, web ou full stack developer.

## Condições Financeiras

- Taxa de inscrição: 425€, dedutível no valor total.
- Possibilidade de pagamento faseado para particulares, **até 10 prestações, sem juros**.
- Estudantes não residentes no território nacional, terão de efetuar um pagamento de 50% do valor total da propina no momento da inscrição.
- Os valores apresentados não incluem IVA. Isenção do valor do IVA a particulares.
- Para informações completas sobre os requisitos, descontos e condições financeiras disponíveis, contacte-nos através do [formacao@rumos.pt](mailto:formacao@rumos.pt) ou do botão Saber Mais.

## 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 a **Academia FLAGProfessional Full Stack Web Development** vais:

- **Dominar todos os conceitos-base** necessários ao trabalho com servidores e configurações de host, integrações de dados (MySQL) e solução de problemas de back e front-end;
- **Conhecer as melhores práticas no mercado** da programação web, a nível nacional e internacional;
- **Estudar as principais linguagens de programação, de estilo e interfaces/software usados no desenvolvimento web:** HTML5, Javascript, PHP, CSS, GIT, Laravel, REST APIs, MongoDB e ExpressJS;
- **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.
- **Ter o apoio da FLAG na procura ativa de uma oportunidade profissional** através do nosso Career & Employability Hub. Esta equipa especializada gere uma rede com mais de uma centena de parceiros com necessidades de atração de novos talentos e visa a integração dos nossos alunos com avaliação igual ou superior a 16 valores no mercado de trabalho.

## Que profissões me esperam?

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

- **Full stack Web Developer**, enquanto especialista que domina aptidões multi-tasking quer do lado do cliente quer do lado do servidor, e está familiarizado com as várias tecnologias envolvidas na programação e design de websites.
- **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.
- **Back-end Developer**, enquanto profissional que cria ou contribui para o bom funcionamento de aplicações para a web e mantém um código limpo, leve e bem documentado.
- **Web Developer**, enquanto programador especializado no desenvolvimento de sistemas para a internet,

responsável pelo perfeito funcionamento de um website ou aplicação web.

Formação em parceria com



---

## Destinatários

**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**, 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**.

---

## Pré-requisitos

Para frequentares a **Academia FLAGProfessional Full Stack 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.

Considerando o elevado grau de exigência das Academias FLAGProfessional é fundamental que sejas organizado, focado e disciplinado, para que consigas garantir o correto acompanhamento das matérias lecionadas, bem como dos projetos a realizar.

---

## Programa

### **Sessão de onboarding**

#### **Introdução à Programação**

- Algoritmo
- Algoritmos computacionais
- Fluxogramas

- Pseudocódigo

## **Metodologias Ágeis para Gestão de Projetos: Scrum**

- Metodologias Processo de Design Digital
- Processos de desenvolvimento de produtos (Agile, Lean e Waterfall)
- Conceito de MVP (Minimum Viable Product)
- Processo de design (etapas)
- The Design Sprint (Google Product Design Sprint)
- Manifesto Ágil, Valores e Princípios Ágeis

## **Fundamentos de UX/UI para Developers**

- Introdução ao UX & UI
- Multi-screen & Responsive Challenges
- UX/UI and development
- Arquitetura de informação
- Design Patterns & Componentes
- Wireframing & Prototipagem
- Assets for development

## **HTML5**

- Estrutura de uma página HTML
- Elementos e atributos
- Elementos de texto
- Listas
- Hiperligações
- Imagens
- Tabelas
- Formulários
- Elementos multimédia
- Elementos estruturais do HTML5

## **CSS3**

- Evolução das CSS
- Utilizar CSS em páginas Web
- Selectores CSS
- Unidades
- Trabalhar com cores
- Vendor prefixes
- Formatação de texto
- Tipografia
- Backgrounds e gradientes
- Borders e sombras
- Box model

- Layout de páginas
- Novos métodos de layout em CSS3
- Propriedades para tabelas e formulários
- Criação de elementos de navegação
- CSS sprites e elementos interactivos
- Transforms, transitions e animations
- Frameworks front-end - Bootstrap

### **Workshop: SASS**

- Preprocessing
- Variáveis
- Nesting
- Partials
- Modules
- Mixins
- Extend & Inheritance
- Operadores

### **Workshop: Responsive Web Design**

- Princípios básicos do responsive
- Responsive websites vs. Mobile website vs. Mobile app
- Introdução ao processo de desenvolvimento
- HTML5 + CSS3 princípios básicos
- Media queries
- Fix and fluid Layouts
- Adaptive and responsive web design
- Responsive design
- Frameworks

### **Fundamentos de Programação com Javascript**

- Paradigmas de Programação (Object Oriented, Imperative e Functional Programming,...)
- Processo de desenvolvimento de um algoritmo (programa) (Utilização de pseudo-código e fluxogramas)
- Tipos de Dados numa linguagem de programação (Variáveis, constantes, ...)
- Operadores e cálculos numa linguagem de programação
- Controle da execução de um programa com recurso a estruturas de decisão e repetição

### **Javascript**

- Introdução, história e evolução dos vários standards da linguagem
- Breve introdução à historia de desenvolvimento da linguagem, e a sua evolução até aos dias de hoje.
- 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 (Immediate Invoked Function Expression)
  - Funções “construtoras” de Objectos
  - Conceito de “Scope” de uma variável, variáveis locais e globais, diferenças em ES6 com let e const
  - Built-In Objects em JS
- 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 seleccionar 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, ...)

## **Front-End Frameworks: React**

- Build Workflow
- Componentes
- JSX
- Restrições JSX
- State
- Handling Events
- Two Way Binding
- Lists, keys & mapping
- Components Lifecycle
- React Hooks
- Ajax in React
- Forms and Form Validation
- Redux
- Redux Advanced
- Deploying React project

## **Workshop: GIT**

- Instalação e configuração de Servidor GIT
- Instalação de Cliente GIT
- Criar uma equipa
- Criar um projecto
- Configurar repositório
- Criar um branch a partir de master
- Commit
- Revert
- Merge
- Deploy

## **Pitstop**

## **Projeto: Front-End**

## **Fundamentos de Back-End Development**

- Modelo Cliente – Servidor
- Protocolos de transferência de informação

- Configuração da máquina de desenvolvimento

## **Introdução à estruturação de dados**

- Modelo Relacional
- Entidades ; atributos; relacionamentos
- Tipos de relação
- Formas normais

## **Base de dados MySQL**

- Modelo físico
- Construção de queries MySQL: INSERT, DELETE , CREATE, SELECT, UPDATE, DROP, JOIN,...
- SP e Cursors

## **PHP**

- Introdução ao PHP
  - Anatomia de um script
  - Tipos de dados
  - Variáveis e operadores
  - Estruturas de controlo e decisão
  - Gestão de erros
- Funções
  - Sintaxe base
  - Retornar valores
  - Domínio de variáveis (scope)
- Arrays
  - Introdução às arrays
  - Operações (comparação, contagem, pesquisa e gestão de elementos)
  - Iterações (ponteiros)
  - Ordenação
- Strings
  - Introdução às strings
  - Comparação, pesquisa e substituição de caracteres
  - Formatação
- OOP
  - Declarar uma classe e criar instâncias
  - Métodos e propriedades
  - Interfaces
- Programação Web
  - Formulários e URLs (GET, POST, Uploads)
  - HTTP Headers (redirecionamento, cookies, sessões)
  - Envio de emails (função mail)
- Integração com MySQL
  - Abrir e fechar uma ligação ao MySQL



- Executar comandos SQL
- Processar dados da base de dados
- Streams
  - Aceder a ficheiros
- Trabalhar com diretorias

## **PHP Frameworks: Laravel**

- O que é uma framework e uma micro-framework
- Vantagens de usar uma Framework
- Diferenças entre frameworks
- Utilização do composer
- Padrão de desenho MVC
- O que é o controlador
- O que é o modelo
- O que é uma vista
- Criação de uma aplicação PHP usando uma Framework

## **Criação de REST APIs**

- O que é uma API REST
- Protocolo HTTP
- Tipos de pedidos e códigos de resposta
- Respostas JSON
- Respostas XML
- Autenticação com base em tokens
- Criação da documentação para a API

## **Workshop MongoDB + NodeJS + ExpressJS**

- MongoDB
  - Instalação
  - Criação de modelo de dados
  - Querys
- NodeJS
  - O que é o NodeJS
  - Utilização do NPM – Node Package Manager
  - Criação de uma aplicação NodeJS
  - Funcionamento do express para resposta a pedidos HTTP
  - Bases de dados não relacionais com MongoDB
  - Utilização de WebSockets usando SocketIO

## **Pitstop**

## **Projeto: Back-End**