

DESENVOLVEDOR FULL STACK JAVASCRIPT REDUX

Carga Horária: 112h

Pré-requisitos: Lógica de programação

Objetivo: O JavaScript está em todos os lugares hoje em dia. Por isso, surgiu a ideia de criar sistemas baseados apenas em JavaScript. Assim, apenas com o conhecimento em uma linguagem de programação, o programador pode trabalhar no Front End, Back End e Banco de Dados. MEAN é uma palavra formada pelas iniciais de cada uma das ferramentas usadas para se criar esses sistemas inteiramente em JavaScript- MongoDB (banco de dados), Express (back end), Angular (front end) e Node.js. Neste curso teremos uma breve introdução a cada uma das ferramentas usadas, iremos aprender a integrá-las e, finalmente, subir um projeto para um servidor.

Curso	Carga Horária	Pré-requisitos Obrigatórios	Pré-requisitos Recomendados
JavaScript	20	Lógica	
TypeScript	12	JavaScript	
NodeJS	28	JavaScript	
MongoDB	24	JavaScript	
React, com Redux	28	JavaScript	
	Total: 112 hrs		

JavaScript

- Sintaxe da linguagem
- Tipos
- Arrays
- Números
- Strings
- Datas
- Objetos como Maps
- Funções e programação funcional
- Closures
- Callbacks
- Promises
- async e await
- Uso de this
- Melhores práticas
- ECMAScript
- JSON

TypeScript

- Porque usar TypeScript?
- Classes
- Interfaces
- Tipos
- Asserção de tipos
- Inferência de tipo
- Escopo de variáveis
- Módulos
- Construtores
- Atributos e Métodos
- Arrays
- Unions
- Namespaces
- Módulos
- Transpilação para JavaScript

NodeJS

- Configuração do ambiente
- REPL
- NPM
- package.json
- Callbacks
- Eventos
- Sistema de arquivos
- require, exports e module.exports
- Objetos globais
- Módulos utilitários
- Escrevendo seus próprios módulos
- Web e Express
- Servindo conteúdo estático
- Web Services Restful com banco MongoDB

MongoDB

- Introdução a bancos NoSQL
- JSON e BSON
- Vantagens
- Qual o problema com os bancos relacionais?
- Instalação e configuração do ambiente
- Coleções e Documentos
- Tipos de dados
- Modelagem de dados
- Inserção, Edição, Consulta e Exclusão de documentos em coleções
- Índices
- Replicações
- Sharding
- Relacionamentos (referenciados e embutidos)
- Agregação
- Projeções
- Escrevendo Scripts
- Administração do servidor

React com Redux

1. Fundamentos

- Criando o primeiro projeto
- Interagindo com o DOM
- JSX
- Criando Componentes
- Carregando CSS em um componente
- Entendo o PROPS. (Propriedades de um componente)
- React Fragment
- Enviando dados entre componentes
 - Comunicação direta
 - Comunicação indireta

2. Desenvolvimento de aplicação real

- Componente APP, para renderização de todos os outros componentes da aplicação
- Rotas
- Gerência de Estado
- Usando template AdminLTE para geração de interface da aplicação
- HTTP Client AXIOS
- CRUD com gestão de estado por componente

3. REDUX

- O que é o REDUX
- Porque usar REDUX
- REDUX x Gestão independente de estado
- Actions
- Reducers
- REDUX FORM
- REDUX FORM TOASTR
- CRUD com REDUX e REDUX FORM

4. Segurança

- Gestão de autenticação do CLIENT e BackEnd
- Armazenamento e segurança da TOKEN
- Gestão de autorização na aplicação por TOKEN de autenticação

5. Gestão de versionamento de arquivos

- GIT
- Enviando código fonte a um repositório GIT