

## Roadmap Dev

### 1- Entendendo HTML e CSS (se necessário)

A primeira parte desse roadmap serve para introduzir você ao universo do desenvolvimento de aplicações web. Antes de prosseguirmos para o desenvolvimento de sistemas mais complexos, precisamos entender como funciona o esqueleto de uma página web. Você NÃO vai desenvolver um projeto somente com o conteúdo apresentado nos seguintes vídeos, porém eles são fundamentais para que você consiga compreender conteúdos mais avançados no futuro deste roadmap.

[pt-br, 43min] [Curso de HTML Para Completos Iniciantes](#)

[pt-br, 1h13] [Curso de CSS Para Completos Iniciantes](#)

[pt-br, 37min] [Desvendando o CSS Grid na prática | Mayk Brito](#)

[pt-br, 43min] [CSS Grid Layout e Flexbox - Quando Utilizar \(opcional\)](#)

[PRÁTICO] [pt-br, 36min] [Site Responsivo HTML5 e CSS3](#)

[PRÁTICO] [inglês, 5h] [FreeCodeCamp: Criando uma landing Page com HTML, CSS e JavaScript\\*](#)

\*Esse curso é MUITO BOM, com o conteúdo visto até aqui você já consegue entender e acompanhar a instrutora. Nesse vídeo você terá a oportunidade de conhecer JavaScript, porém por questões de ter membros disponíveis para projetos, recomendo que ele seja feito só depois do curso de Wordpress. Se optar por fazer esse curso agora, pule o curso de wordpress.

### 2- Versionamento de código / Git

Nesse momento você já deve ter algumas pastas no seu computador com os arquivos dos cursos anteriores. Esses arquivos podem ser upados para um repositório na internet, possibilitando que você armazene e compartilhe seus projetos. Nessa sessão do RoadMap vamos aprender sobre GIT e versionamento de código.

[pt-br, 7min] [O que é Git e GitHub?](#)

[pt-br, 10min] [Entenda o que é GIT em 10 minutos](#)

[ARTIGO] [pt-br] [Guia Prático](#)

[pt-br, 10min] [Como instalar e configurar o GIT no windows](#)

[pt-br 16min] [Git prático](#)

[pt-br, 22min] [Como criar um READ.ME](#)

### 3- Wordpress, o quebra galho.

Agora que já entendemos como funciona uma página HTML e como ela é estilizada, iremos conhecer uma plataforma para criação de sites muito utilizada no mercado, que não exige conhecimentos avançados de programação ou desenvolvimento web. O Wordpress possui uma série de recursos que eliminam todo o processo de estruturação da página, um destes recursos é a vasta biblioteca de temas prontos, ficando a cargo do desenvolvedor o trabalho de “arrastar bloquinhos” para criar o conteúdo da página, enquanto menus, estilos e footers são entregues de bandeja na nossa mão. Cerca de 60% dos sites que você acessa na internet é feito em wordpress, você pode verificar se um site é ou não wordpress através do serviço [Isitwp](#). Contudo, essa “facilidade” é muito interessante quando queremos construir

portais com Blogs integrados, ou lojas virtuais, uma vez que esses recursos também já estão disponíveis para o *dev*.

[pt-br, 10min] [O que é Wordpress?](#)

[pt-br, 13min] [Wamp: instalando e Configurando o ambiente](#)

[PRÁTICO] [pt-br, 1h] [Como Criar Um Site No Wordpress | 2020 | Para Iniciantes\\*](#)

\*Neste vídeo, ele não ensina a instalar nem configurar o ambiente pois ele ensina como utilizar direto do servidor. Nosso servidor local é aquele configurado pelo WAMP. Logo, pule direto para a o minuto 18h17 e siga as aulas normalmente. O domínio que você irá digitar no navegador é simplesmente (**localhost/NomedoSeuSiteNaPastaHtdocs**) quando o WAMP estiver ligado.

[PRÁTICO] [pt-br, 1h58] [Curso Completo de Elementor Passo a Passo 2020 para Iniciantes](#) (opcional)

[PRÁTICO] [pt-br, 2h17] [Criando loja virtual com woocommerce](#) (opcional)

#### 4- O que é Backend, Frontend, API?

Nesse momento você está apto para desenvolver landing pages e lojas virtuais, seja “na unha” ou utilizando CMS. Com esse conhecimento você já é capaz de pegar projetos reais e não terá problemas com seu desenvolvimento.

A partir de agora vamos entrar em campos um pouco mais técnicos e intermediários. Nosso intuito como EJEC é fazer cada vez menos “sites” e cada vez mais “sistemas”. Sistemas possuem regras de negócio muito específicas, o que não nos possibilita usar Wordpress. Sistemas exigem criação e manipulação de dados com o intuito de cumprir um objetivo.

Contudo, não abandonaremos o conteúdo que vimos nas aulas anteriores. Os vídeos a seguir são conceituais para que você entenda o que é Frontend e Backend e como essas partes se comunicam.

[pt-br, 10min] [Backend e Frontend](#)

[pt-br, 34min] - [RocketSeat - API, REST e RESTful](#)

[pt-br, 7min] [O que é JSON?](#)

[PRÁTICO] [pt-br, 43min] [Criando uma API do Zero - NodeJS\\*](#)(opcional)

\*Na próxima sessão você conhecerá o NodeJS e estará pronto para realizar o curso de cima, porém vale dar uma assistida só para ver como é a criação de uma api na prática.

#### 5- A programação vai começar, iniciando com fundamentos de JavaScript

Já que vamos começar a desenvolver sistemas complexos, precisamos escolher a linguagem que irá rodar no backend dos nossos projetos. Esse roadmap irá focar no universo do JavaScript, pois é uma das linguagens mais consolidadas no mercado, possui muitas bibliotecas e inúmeras aplicações. A partir desse momento, os conteúdos mais relevantes serão apresentados em inglês. Porém alternativas em português poderão ser encontradas.

[pt-br, 11min] [O que é JavaScript?](#)

[espanhol, 6min] [breve História do JavaScript](#)

Escolha **UMA** das seguintes opções:

[PRÁTICO] [pt-br, 4h36] [Curso Javascript Completo 2020 \[Iniciantes\] + 14 Mini-Projetos](#)

[PRÁTICO] [inglês, 3h27] [Learn JavaScript - Full Course for Beginners](#)

[PRÁTICO] [inglês, 1h50] [Learn JavaScript by Building 7 Games - Full Course](#) (opcional)

[pt-br, 11min] [Node.js](#)

[inglês, 1h31] [Node.js Crash Course](#)

Nesse momento, você pode pesquisar por “Simple JavaScript projects” no youtube que encontrará muitos tutoriais práticos.

## 5- Banco de dados: SQL, noSQL

Todo sistema é alimentado por dados, eles precisam ser armazenados para que possam ser processados quando solicitados. Existem algumas formas de armazenar essas informações, as mais comuns são os bancos de dados e podem ser classificados entre SQL ou noSQL. Nessa sessão você entende como funcionam e quais as suas diferenças.

[pt-br] [O que é SQL e NoSQL?](#)

[inglês, 1h12] [MySQL Crash Course | Learn SQL](#) (SQL)

[inglês, 37min] [MongoDB Crash Course](#) (noSQL)

## 6- Escolhendo sua STACK

Você nesse momento está pronto para escolher uma STACK para começar a desenvolver sistemas completos. Os conteúdos de front e back começam a ficar fortemente ligados, então fica mais fácil aprender uma framework de frontend já aplicada em uma stack. Existem diversas frameworks de frontend JavaScript.

[ARTIGO] [O que é uma STACK?](#)

[pt-br, 13min] [O que é Biblioteca, Framework e API?](#)

[ARTIGO] [inglês] [Lista de Frameworks JS](#)

[inglês, 14min] [Choosing Your JavaScript Framework](#)

Algumas opções de stack:

**MEAN:** MongoDB - Angular- Express.js - Node.js

**MERN:** MongoDB - React - Express.js - Node.js

**PERN:** Postgres - React - Express.js - Node.js

**MEVN:** MongoDB - Vue - Express.js - Node.js

Sugestões de cursos:

[inglês, 1h23] [PERN Stack Course - Postgres, Express, React, and Node](#)

[inglês, 1h47] [Learn the MERN Stack - Full Tutorial \(MongoDB, Express, React, Node.js\)](#)

[pt-br, 4h42] [Construindo um app de mensagem com MEVN Stack](#)

[ing, 12h] [Construindo um app com MEAN Stack](#)

[pt-br, xxh] [Todos os cursos da RocketSeat no drive da EJEC](#)