

```

10  // Let the user know about the status of
11  Toast.show({
12    text: 'You are offline now..',
13    position: 'bottom',
14    buttonText: 'Okay',
15    type: 'warning',
16  });
17
18  // async ratingPromptCheck() {
19  //   // Check if the rating prompt is enabled or not
20  //   let ratingPromptDisabled = await Cache.get('ratingPromptDisabled');
21  //
22  //   if(!ratingPromptDisabled) {
23  //     // Check if the timer is set or not
24  //
25  //     Cache.set('ratingPromptTimer', _now() / 1000);
26  //   } else
27  //     if(ratingPromptTimer && _now() / 1000 - ratingPromptTimer > 1000) {
28  //       this.displayRatingPrompt();
29  //     }
30  //
31  //   // async displayRatingPrompt() {
32  //     // alert()
33  //     'May we ask for a good rating?',
34  //     'Leave us a 5 star rating if you enjoy the app. Thank you for using our app.'
35  //     |
36  //     {text: 'Not now', onPress: () => {
37  //       // Reset the timer
38  //       Cache.set('ratingPromptTimer', _now() / 1000);
39  //     }}
40  //   }

```

# GUIA COMPLETO

# FRONT-END

SUJEITO PROGRAMADOR

# FRONT-END



## Índice

---

A base do front-end	2
O que estudar	3
Bibliotecas e frameworks	6
Próximo passo	7

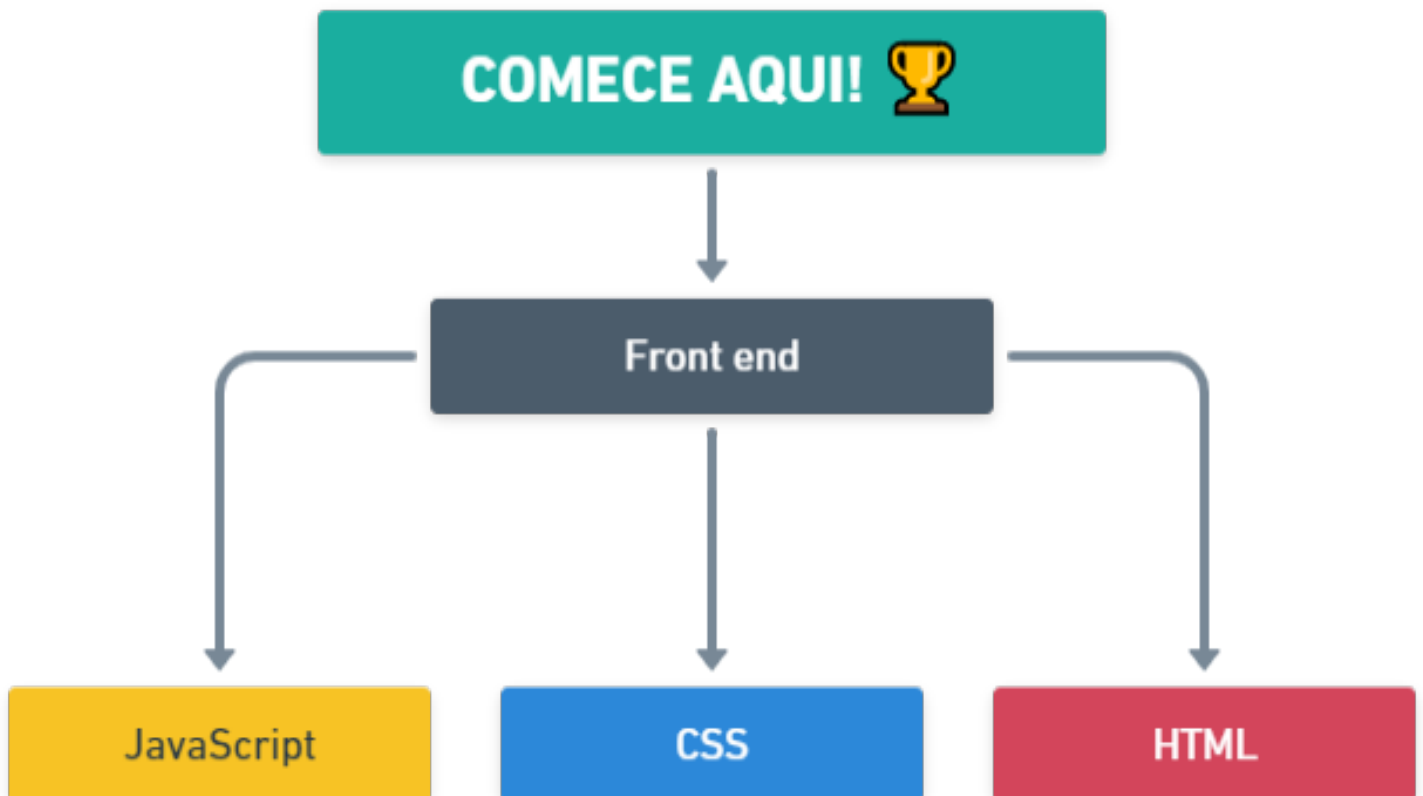
---

# FRONT-END

Vamos lá para nós começarmos, não tem como fugir da base, tudo que você vai desenvolver na web seja aplicativo, sistema, sites, sistemas desktops você precisa ter a base para conseguir evoluir.

Construir uma base sólida você não vai ser aquele desenvolvedor do CHUTE, ficar chutando tag html, chutando o que funciona ou o copiando código e colando para ver se funciona.

Pode existir MILHÕES de tecnologias, mas tudo começa com essa base que separei aqui:



# FRONT-END



Certo entendido que essa é a base para começar no **front-end** e ai vem aquela famosa pergunta: **Tem alguma ordem para começar estudar essas tecnologias?**

Não tem nenhuma LEI ou algo dependente aqui que precise ser uma tecnologia primeiro e não a outra, você pode começar estudando HTML, depois CSS e depois evoluindo para o Javascript ou se quiser estudar primeiro a linguagem de programação então pode começar estudando na ordem Javascript, Html e CSS.

Estudando a base, os conhecimentos básicos você já esta dando inicio no frontend.

Bora ver o que você deve focar para começar com tudo e de forma certa!

## HTML

O HTML é uma linguagem de marcação responsável pela estrutura de páginas web, dando significado para cada parte que a compõe.

### Sintaxe do HTML

A sintaxe do HTML foi criada para ser de fácil entendimento tanto para nós quanto para a máquina, além disso você não precisa de nada mais do que um editor simples. Como vimos, HTML é uma linguagem de marcação.

Essas marcações são chamadas de tags, que são abertas e depois fechadas. É com as tags que dizemos para o navegador o que é um título, parágrafo, botão, formulário, dentre outros.

# HTML

Comece estudando:

- Sintaxe.
- Entenda a estrutura básica de um arquivo HTML.
- HTML Semântico.
- Tags básicas.
- Tags Meta, melhorando (SEO).

# CSS

Comece estudando:

- Sintaxe.
- Entenda a estrutura básica de um CSS.
- Estude como aplicar class.
- Estude como aplicar ID.
- Entenda diferenças de quando usar class e ID.
- Flexbox e Grid são importantíssimos para desenvolver layouts dos projetos.
- Medidas px, rem, em.
- Media Queries (Para deixar seus projetos responsivos em qualquer dispositivo seja Desktop, tablet ou celular).

# JAVASCRIPT

Comece estudando:

- Entenda a Sintaxe da linguagem.
- Variáveis, tipos de variáveis, manipular as variáveis.
- Comandos de entrada e saída, estude sobre o document.
- Funções.
- Como modificar o html, tornar ele dinâmico através do Javascript.
- Aprenda trabalhar com loops.
- Eventos de mouse, teclado, eventos da página.
- Manipular datas.
- Trabalhar com formulários.
- Usar temporizadores.
- E parar finalizar estude sobre ES6+ ou seja diferenças entre var, let e const, desconstruir objetos, spread operator, rest operator, imutabilidade, funções anônimas.

## HTML

## CSS

## JAVASCRIPT

Sabendo esses pontos falados acima você está mais que pronto para criar sistemas, sites mas se você quer evoluir MAIS ainda e aplicar tudo isso em uma biblioteca ou framework agora você é mais que bem vindo, bora falar sobre isso e qual próximo passo.

## Bibliotecas / Frameworks

Agora sim você já tem a base sólida para evoluir e aproveitar o máximo da produtividade e do poder que uma Biblioteca ou Framework vai trazer no seus projetos.

Vou colocar abaixo algumas das opções mais buscadas e usadas pelo mercado de trabalho:

**React JS**

**Angular**

**Vue JS**

E para finalizar e completar seu leque separei as ferramentas essenciais para entrar no mercado de trabalho ou até mesmo trabalhar em equipe.

## Ferramentas

Bom entender sobre o nodejs, npm, yarn, agora o mais fundamental de todos é dominar o GIT e Github isso é mais que essencial seja se você quer criar sistemas web, aplicativos, sistemas desktop.

Git e Github é fundamental para versionar seu código, manter projetos maiores, trabalhar em equipe ou até para entrar em qualquer vaga como desenvolvedor, pode ter certeza que esse requisito vai ser exigido.

# FRONT-END

O Mercado para **React JS** e **React Native** está super aquecido no brasil e muito bem pago além dos gigantes da tecnologia que aplicam em seus produtos/serviços.

Então se você quer ser um desenvolvedor web reconhecido, onde tem vagas no mercado que pagam bem a **Stack React** é o caminho, você vai encontrar facilmente vagas que pagam desde **R\$ 3mil** até **R\$ 12mil** pra mais.

Mas fica tranquilo que **não acabou** preparamos algo **exclusivo pra você** que chegou até aqui.

Para você se tornar um desenvolvedor profissional criar aplicações completas do zero, desenvolver aplicativos, sistemas, tudo isso usando React e mudar sua carreira de uma vez por todas.

**SIM! QUERO SER UM(A) DESENVOLVEDOR(A) PROFISSIONAL!**





Chegamos ao fim dessa jornada incrível de muito aprendizado.

## Acompanhe nossas redes sociais:

 **SujeitoProgramador**

 **SujeitoProgramador**

 **SujeitoProgramador**

# FRONT-END

```
let ratingPromptTimer = null;
let ratingPrompt = null;

if (ratingPromptTimer) {
  Cache.set('ratingPromptTimer', 'now' / 1000);
} else {
  if (ratingPromptTimer && 'now' / 1000 - ratingPromptTimer < 60000) {
    this.displayRatingPrompt();
  }
}

async displayRatingPrompt() {
  alert(
    'May we ask for a good rating?',
    'Leave us a 5 star rating if you enjoy the app, thank you for using our app!'
  );
  (text: 'Not now', onPress: () => {
    // Reset the timer
    Cache.set('ratingPromptTimer', 'now' / 1000);
  })
}
```