**# Maratona Discover**

**## O que vamos ver?**

\* Hard Skills que você precisa aprender para estar pronto/a profissionalmente

\* Soft Skills que você precisa treinar para continuar a maratona até a linha de chegada

\* 2021 para os devs front-end

\* Construir uma experiência profissional, mesmo sem estar contratado/a

\* Estimativa de tempo para colher resultados

\* Fórmula para você mensurar seu tempo de estudo para chegar mais perto dos resultados

**## O que você ganhará com isso?**

\* Perspectiva de futuro

\* Os próximos passos a seguir

\* Mensurar melhor seus resultados

**## Vantagens de ser Dev Front-end**

\* Vagas

\* Salário [<https://www.salario.com.br/profissao/programador-front-end-cbo-317110/>]

\* Boa parte da aplicação reside no front-end

\* Um pouco mais tranquilo para iniciar do que o back-end (porém, é desafiador o domínio mais completo)

\* Pode ser divertido: design, código, cores, tipografia, experiência do usuário, serveless, ferramentas, frameworks ...

**## Desafios**

\* Muitas ferramentas para decidir

\* Muitos frameworks para confundir

\* Paradigmas e patterns pra estudar

\* Continuar estudando, pois mudanças acontecem muito rápido

**## Vamos alinhar a expectativa**

\* Você poderá \*\*não estar pronto\*\* em 2 meses

\* Se você está com a corda no pescoço, poderá se decepcionar

\* Se você acelerar demais, você vai capotar

\* Vai vai precisar ter foco pra chegar lá!

\* Você vai chegar lá, mas dependerá da fórmula TEMPO \* ESFORÇO

> Já prepare seu Linkedin

- Perfil profissional

- Foto

- Coloque o que você está estudando

- Habilidades

- Compartilhe coisas boas

- Fuja de discussões (não trazem nenhum benefício)

- Queira ajudar as pessoas e não criticar

- Apresente soluções e não problemas

**## Investimento de tempo (Hard Skills)**

### Inglês Instrumental

- Investimento de tempo constante

- Glossário

> É possível seguir sem o inglês?

- Sim, mas será bem mais desafiador.

> Devo fazer um curso de inglês?

- Se estiver no seu orçamento

### Entender os fundamentos da programação: 20h

- Visão geral da área

- A ciência por trás da computação

- Como funciona o computador

- Como funciona a WEB e a Internet

- Algoritmos

- Estrutura de dados

- Lógica de programação

- Cliente e Servidor

- HTTP

> Praticamente leitura e pesquisas

> Algoritmos, Estrutura de dados e Lógica é melhor assimilado com a prática

**### HTML e CSS: 100h**

\* Entenda os fundamentos (Discover)

\* 20% estudando 80% praticando

\* Como praticar?

\* Micro atividades

1. Escolha algum site que você gosta bastante

2. Tente replicar somente 1 compomente por vez

3. Crie 3 variações daquele componente

4. Repita...

- Exemplo de componentes

- Navbar, Buttons, Forms, Headers, Footers, Articles, Cards, Modals ...

\* Macro atividades

1. Crie de 5 a 10 mini sites (one page, multi-page)

2. Transformar layouts do Figma, Dribble, Behance

> Não pense, code!

- Informação é o alimento, você come o necessário para performar melhor

- Cuidado com a obesidade informacional.

- Aja mais! Code mais!

> A maior parte desse tempo está no CSS

> Construir alguma coisa só uma vez não é o suficiente

- Repetição é a chave que vai abrir as portas mentais do entendimento

> Ligue suas antenas para o Design!

\* Onde conseguir mais conteúdo?

- Documentação MDN

- CSS Tricks

- Codepen

- Google, Youtube

#### Adicionais ao HTML e CSS

- Acessibilidade

- SEO

- Responsividade

- Design

- Tipografia, Cores e Layouts

- Posicionamentos dos elementos

- Flex, Grid

- Animations e Transitions ...

#### Vamos alinhar as expectativas

\* Você saberá que está no ponto assim que conseguir criar componentes e páginas em poucos minutos / horas

\* Diferença entre não saber o que pesquisar e saber o que pesquisar

\* Lembre do momento platô

- grandes resultados no início

- sensação de poucos resultados por um tempo (platô)

\* PROVAVELMENTE, você não terá dominado tudo de CSS e HTML após esse tempo...

**### Git, Github, Linha de comando: 10h**

Fundamentos necessários para controlar o histórico do seu código

Buscar familiaridade com o terminal

\* Em breve + conteúdos no Discover

> Já prepare seu Github estudante/profissional

- Ele será seu portfólio

- Ele mostrará seu progresso e sua constância

**### JavaScript: 400h**

> Você não precisa estudar outra linguagem

\* O que aprender?

- Fundamentos

- Estrutura de dados

- Variáveis

- Objetos

- Funções

- Tipos de Dados

- Interação com HMTL (DOM)

- Formulários

- Validações

- Eventos

- Animações

- APIs

- Lógica

- Games

\* O que construir/fazer?

\* Micro atividades

- Components

- Sliders

- Galeria de imagens

- Player de audio/video

- Validações de formulários

- CRUDs

- Adicionar interação aos seus componentes/sites/apps

- Pratique algoritmos e lógica em sites de desafios (será desafiador)

- JSChallenger

- Codewars

- freecodecamp

- front-end mentor

- <https://www.uplabs.com/>

- <https://www.hackerrank.com/>

- <https://www.devchallenge.com.br/>

\* Macro atividades

- Jogos (replique jogos básico como jogo da velha por exemplo)

- Aplicações (por temas)

- Financeiras

- Relacionamentos

- Culinária

- Vendas

- Música

\* Onde conseguir conteúdo?

\* Discover

- Aprenda algo novo e aplique

- Pense em diferentes maneiras para modificar/aplicar

\* Livros?

- Eloquent Javascript

- You don't know Javascript

\* Documentação (MDN)

\* Blogs, Youtube ...

\* Google, Stackoverflow

**#### Avançando no JS**

- AJAX

- Promises

- Async/Await

- Programação funcional

- Programação orientada a objetos

#### Vamos alinhar as expectativas

\* Continue a nadar!

- Pesquisar poderá trazer um desanimo

- A maioria das soluções estão lá a questão é: quando você vai encontrar?

- Quando o seu cérebro estiver doendo saiba que você está aprendendo.

\* Seu maior desafio será buscar soluções!

- Melhore suas perguntas para melhorar seus resultados

- Aumente seu vocabulário na linguagem, pra saber o que pesquisar

\* O problema poderá parecer impossível, mas se alguém resolveu, você também pode!

\* Dominar JS vai levar mais tempo que isso. Quer saber se dominou JS? Escreva a biblioteca jQuery do Zero ou o framework React

\* Você já vai ter entendido que programar é praticar e praticar sempre!

#### Aumente a dificuldade

- Prototype

- Arquitetura de projetos (Alguns aplicados na prática)

- Tente fazer clones de aplicações complexas, aplicando as funcionalidades

- Ecommerce (Mercado Livre, Amazon)

- Forum (Reddit)

- Rede social (FB, Tiktok, Twitter)

- Aplicativos diversos

### Ferramentas mais avançadas: 20h

Comece a introduzir algumas ferramentas mais avançadas

- SASS

- Babel

- Webpack

### Fundamentos de comunicação com Back end: 50h

- APIs

- REST

- Fundamentos de Back end

- NodeJS, Express

- Construir e consumir APIs

- Databases: MongoDB e SQL

- Casos de uso de NoSQL e SQL

**### React: 100h**

\* A escolha

- Existem muitas opções e isso gera dúvidas

- Não existe o melhor framework mas existe aquele que dá mais vagas

- Escolha o seu e aprenda, mas aprenda bem!

\* Por que usar um framework?

- Desenvolver mais rápido e com padrão pré estabelecidos

- Maior escalabilidade e manutenção do seu projetos

- Melhor trabalho com o estado da aplicação e dos componentes

- Se desafie a entender mais motivos para usar no seu projeto

> Mas dá pra fazer tudo sem framework?

- Dá, não significa que seja o melhor caminho

> Acho React muito complexo

- Pode ser que você precise um pouco mais de fundamentos

\* Como praticar?

- Refazer tudo o que você construiu até aqui com com JS puro

- Crie mais projetos e aprenda como melhorar seu código

- Uns 10+ Projetos (micros)

- Faça um projeto gigante e use ele como teste de constância

#### Alinhando as expectativas

\* Trabalhe duro, mas trabalhe com foco!

\* Direção é tão importante quando o esforço aplicado

\* Se estiver complexo, volte alguns passos, pois o aprendizado não é gráfico linear

## Quanto tempo você investiu até aqui?

~700h

\*Formula\*

\_Hora por dia x Dias da semana x Semanas do mês x Meses\_

Exemplo:

Vou dedicar 1h e meia por dia,

de segunda a sexta,

todas as semanas do mês durante 24 meses (2 anos),

quantas horas terei investido?

1h30(h) \* 5 (dias) \* 4 (semanas no mês) \* 24 (meses) = 720hrs de estudo

\* Qual a formula para você transformar 2 anos em 4 meses?

- 2 anos de falta de foco

- 2 anos de procrastinação

- 2 anos de "deixa pra mais tarde"

- 4 meses de foco absoluto

- 4 meses de prioridade

7h30 \* 6 dias \* 4 semanas \* 4 meses = 720hrs de estudo

> Dá pra alcançar esses resultados no menor tempo possível!

Você só precisa o veículo ideal para essa jornada!

> Quer ganhar mais tempo?

\* Use de 0 a 15 minutos de redes sociais por dia

- A não ser que você esteja fazendo dinheiro com isso.

\* Desabilite as notificações do whatsapp e abra no máximo de 2 em 2h, somente por 5 minutos para respostas mais urgentes.

\* Use alguma técnica para gerenciar o vazamento de tempo

- Escreva num papel suas interrupções

- Descubra como melhorar isso

- Use a técnica do pomodoro para ter mais foco na atividade

- Tenha um ambiente adequado para seu trabalho/estudo

- Sem interrupções

- Isso fará uma ENORME diferença na sua assimilação de conteúdo.

- Você facilitará seus desafios

> Você não precisa de mais tempo, você só precisa entender suas prioridades.

- Família

- Trabalho

- Estudo

- As outras coisas

- Festas

- Viagens

- ...

## Soft Skills

- Não deixe pra depois! (comprometimento)

- Code todos os dias! (constância)

- Seja sua melhor versão (aperfeiçoamento)

- Não faça nada com pressa (paciência)

- Tenha uma comunidade alinhada com seus propósitos

(networking)

- Não faça pelo dinheiro (propósito)

- Entenda seu momento, seu contexto e seus objetivos (visão clara)

- Saúde

- Durma bem

- Se alimente bem

- Faça exercícios físicos

```js

Curta sua jornada!

Ame o aprendizado!

Pratique a gratidão!

**NEVER STOP LEARNING**

* Foco!
* Comunidade
  + Fazer tudo isso sem ajuda é muito mais desafiador.
  + Compartilhe suas vitórias, mas também seus desafios
* Seja uma pessoa blindada!
  + Não dê ouvidos a qualquer opinião
  + Não entre em discussões que não levam a nada
  + Permita ficar ao seu redor, pessoas que te coloquem no próximo nível.
* Faça mais! Pratique mais! Foque no seu objetivo e no seu resultado!