# O que preciso saber para trabalhar no exterior?

## Algumas coisinhas que você precisa saber para poder conseguir um trabalho no exterior e/ou remoto.

## Introdução

¡Hola! Diretamente de Santiago de Compostela, estou aqui para escrever mais um post para vocês, espero que gostem. O post vai ficar separado na série [Trabalhar no exterior, remoto, viagens e etc.](_blank), tem outros dois posts já criados se você ainda não viu e agora tem mais um!

O post de hoje também foi extraído das milhares (sim, foram mais de 6500 perguntas! Várias repetidas, mas mesmo assim! O\_O) deste [formulário](https://willianjusten.typeform.com/to/MtDdfi).

Enquanto vou escrevendo esse post, vou ouvindo uma banda muito maneira que o [Wellington Mitrut](https://twitter.com/Wmitrut) me recomendou e que eu curiosamente já até tinha numa das minhas playlists favoritas. A banda em questão é a [Rival Sons](https://open.spotify.com/artist/356c8AN5YWKvz86B4Sb1yf), vai lá e ouve um pouco =)

A ideia aqui será escrever alguns pontos importantes que você precisa saber para conseguir um trabalho no exterior ou então trabalhar remoto para uma empresa do exterior. Não estou dizendo que se você tiver tudo isso você vai conseguir um emprego logo de cara, mas as chances de você conseguir algo serão bem maiores. E se você está achando que eu vou falar que você precisa saber aquele framework da moda e etc, está enganado. Eu vou falar mais sobre coisas que não necessariamente tem só a ver com seu conhecimento técnico. E vamos aos pontos!

## Perca o medo!

Isso mesmo! A primeira coisa que eu digo para você é, perca o medo! Se você está aqui nesse post é porque você provavelmente quer muito trabalhar fora, mas sempre fica com aquilo na cabeça "Ainda não estou no nível certo. Quando eu aprender X, aí sim eu vou arriscar...". Deixa isso para lá, se você nunca tentar, você nunca vai saber qual nível de fato que eles querem lá fora. E sabe o que vai acontecer se você não passar? Isso mesmo, nada!

Não pense que você está desperdiçando chances se tentar para uma empresa e não conseguir. Pelo contrário, você está se testando e analisando mercado. E muitas das empresas dão uma segunda chance depois de um tempo. Então quer algo melhor do que tentar uma vaga para algum lugar, não passar, mas pelo menos ter uma ideia do que se é pedido?

E quando eu digo perder o medo eu digo de tudo, isso conta aquele seu inglês que é bom e você nem sabe, porque tem medo de falar com qualquer pessoa. Eu vou falar mais disso num próximo ponto, continue lendo =D

## Tenha foco e não tente abraçar o mundo

Outra coisa que acontece demais com todo mundo (to me incluindo aqui) é querermos abraçar o mundo. O que digo com isso? É querer aprender tudo, querer saber Javascript (todos os frameworks lançados a cada segundo, enquanto você está lendo já lançaram 5!), Python, Ruby, Scala, Haskell e até aquela linguagem que você nem sabia que existia, mas te falaram que tem muita procura no mercado lá da Lituânia!

Não importa que todo dia nasce um Framework JS novo, não importa que te falaram que linguagem X dá muito dinheiro nos EUA. Tenha um foco em mente e se especialize naquilo. Existe mercado para todo mundo e ainda mais para aqueles que conhecem bem determinada coisa. Se você está se dedicando a aprender Angular (insira qualquer outra coisa aqui) e está entendendo e evoluindo, não pare só porque disseram que React é mais procurado. Quando você para e começa outra coisa muito frequentemente, você acaba não ficando realmente bom em nenhuma coisa e isso vai te prejudicar se no teste fizerem perguntas mais específicas. Entenda que não estou falando para não testar coisas novas, muito pelo contrário, isso é muito legal e super ajuda! Estou só dizendo para fazer uma coisa de cada vez =)

## Não tenha medo de teoria

Muitos desenvolvedores acabam aprendendo boa parte das coisas na prática e isso é fantástico! Só que dentro disso, está uma pequena armadilha também. Muita gente aprende na prática que se usar X funciona, mas se usar Y não funciona. Só que o mais importante "Por que isso funciona assim?", as vezes é deixado de lado. Se você conseguir aliar a teoria à prática, de forma que você realmente entenda os processos, isso além de te salvar horas debugando, vai te permitir ter discussões melhores e ideias melhores sobre determinado assunto. É por isso que muitos testes para fora são mais focados em algoritmos e estrutura de dados, pois eles entendem que se você sabe sobre os processos, aprender o ecossistema da empresa será muito mais rápido e fácil. Então toma um tempinho para dar uma olhada em Algoritmos e Estrutura de Dados, tente fazer alguns desafios do [HackerRank](https://www.hackerrank.com/) ou outros similares, garanto que vai te abrir o mundo.

O [Jean Carlo Emer](https://twitter.com/jcemer) fez uma palestra bem foda falando um pouco disso, [olha os slides aqui](https://www.slideshare.net/jeancarloemer/what-i-learned-interviewing-for-frontend-developer-roles) e o [vídeo aqui](https://www.youtube.com/watch?v=nfk3G3VB0OY).

## Pesquise, pergunte, não tenha dúvidas

Se você tem vontade de trabalhar fora, primeiro você precisa fazer várias perguntinhas e colocar elas numa balanca.

* Quais empresas eu gostaria de trabalhar?
* Quero trabalhar e viajar ao mesmo tempo?
* Quero só me mudar do Brasil?
* Quais países eu gostaria de morar? Esse país é quente, é frio? Será que vou me adaptar?
* Quais os requisitos que esse país pede?

Não adianta muito você simplesmente dizer que quer trabalhar fora e não se orientar sobre nada, porque bom... Existem centenas de países por aí e se você sequer sabe uma empresa que quer ou um país, como que você vai começar?

Tendo algumas ideias, saia perguntando para quem você conhece que já trabalhou no lugar ou que mora próximo, essas coisas. Crie uma rede! As vezes essa pessoa pode te ajudar mais do que você pensa =)

## Inglês, ahh o temido Inglês...

Não poderia deixar de aparecer esse assunto quando falamos em exterior né? O Inglês é uma língua global e fundamental para quem está na área de desenvolvimento, mesmo que você sequer queira trabalhar fora ou se mudar de país. Um tempo atrás eu postei o seguinte no Twitter:

Muita gente concordou, mas tiveram alguns poucos que não concordaram com a afirmação. Lembrando que eu não estava generalizando e dizendo que se você não sabe inglês, você não é um bom programador. Eu conheço programadores fantásticos, mas que não são tão bons no inglês, mas já estão correndo atrás pelo menos =p

Eu falei no [post passado](https://willianjusten.com.br/devs-brasileiros-gringos-trabalhar-exterior/) que temos devs brasileiros tão bons ou até melhores que os programadores lá de fora. Mas o que acontece é que o número de programadores que fala/escreve em inglês é muito superior aos que o fazem em português. Com isso, o material em inglês acaba sendo mais rico e mais vasto. E se você sabe inglês, olha quanto conteúdo maneiro se abre para você? Isso te permite evoluir e aprender muito mais rápido, o que, consequentemente, vai te permitir melhores oportunidades tanto dentro como fora do Brasil.

Ok, você já entendeu que o inglês é de extrema importância para nossa área. Mas e aí fica a dúvida, qual nível que eles pedem lá fora? Pessoal, relaxem, ninguém vai te pedir para escrever um livro utilizando a melhor gramática de Oxford ou te fazer dar um discurso para presidente dos EUA.

O mais importante numa comunicação é entender e se fazer entendido. Nada mais.

Claro que não rola fazer entrevistas se você só sabe o verbo to be e the book is on the table. Mas se você é capaz de se comunicar (mesmo que lhe falte um pouco de vocabulário e alguns errinhos de concordância) e principalmente de entender o que falam contigo, isso é mais do que meio caminho andado. Você vai ver que depois de um tempo imerso na língua, as dificuldades vão começar a desaparecer.

Lembre-se também que, com exceção de EUA, Canadá, UK, a maioria dos outros países também não tem inglês como língua nativa. Isso significa que a mesma dificuldade que você tem as vezes, as pessoas também terão! Eu até hoje não conheci um Francês/Belga que tivesse uma pronúncia 100% correta, é difícil para caramba para eles! O mesmo eu diria dos espanhóis, eu estou passando uma temporada na Espanha e já trabalhei em alguns coworkings aqui onde tinha bastante gente que sabia beeeem pouquinho de Inglês, mas ficavam super felizes de conversar e treinar comigo.

Então minha dica para vocês é, se dedique em inglês tanto quanto a linguagem de programação, pois o inglês vai te abrir portas muito mais facilmente que só a linguagem.

Aí vem aquela outra pergunta bem comum "Tem como aprender rápido e fácil? Preciso para ontem!". A resposta para essa pergunta é "depende", eu diria que rápido e fácil não tem como. Não espere sair do Nível Joel Santana para um palestrante do TEDX da noite para o dia. O inglês como qualquer outra língua requer tempo para que você se acostume e se adapte a ela. Mas existem algumas maneiras que podem ajudar no processo.

Eu diria que as melhores formas que eu conheço seriam: Intercâmbio em algum país que tenha inglês como língua nativa ou ter aulas com algum professor de língua nativa. E claro, o mais importante de tudo, é praticar e praticar e praticar. Mudar todas os eletrônicos e contas para o inglês. Assistir séries em inglês, ouvir as músicas e ler as letras. Enfim, tentar ficar o mais imerso possível!

Sobre ter aulas com professor de língua nativa, eu conheço o [iTalki](https://www.italki.com/), alguns amigos fazem ou já fizeram e dizem que ajuda bastante. Lá você pode marcar aulas com um professor nativo e ele vai te guiar da melhor forma possível. É muito mais efetivo que uma turma de 20 cabeças no Brasil, por exemplo...

E sobre intercâmbio, eu fiz o meu na Irlanda e você pode ler [nesse post aqui](https://willianjusten.com.br/vindo-morar-e-estudar-na-irlanda/). Eu era uma das pessoas que já tinha um inglês razoável, mas achava que era um bosta completo e tinha muito muito nervoso de falar em inglês até para o espelho! E bom, quando eu cheguei na Irlanda, eu tive que me virar. Claro que na Irlanda tem muitos brasileiros, mas eu sempre me forçava a falar em inglês e também fazia muita coisa sozinho, só para ter que falar.

Lembro até hoje que no meu segundo dia em Dublin, eu parei para ver um artista de rua fazendo uma pintura e ele começou a conversar comigo, eu suava da cabeça aos pés, mas fui falando e tentando entender na medida do possível. No final da conversa eu pedi desculpas pelo meu inglês ruim e o cara simplesmente falou "Eu notei que você não é daqui, mas eu entendi tudo que você falou e você me entendeu também, não é isso que importa?", me deu um tapinha nas costas e sorriu. Eu saí tão feliz dali que depois disso eu tentava falar sobre qualquer coisa com qualquer um.

Tem um outro relato sobre intercâmbio bem legal de um programador também, que você pode [ler aqui](http://www.igorapa.com/fui-aprender-ingles.html). Ele foi para os EUA e também se virou para aprender.

E como sempre, o fórum do FrontendBR sempre tem uma issue bacana para ajudar em qualquer coisa, [você pode ler essa daqui](https://github.com/frontendbr/forum/issues/109).

## Crie um portfólio

Não, eu não estou falando que você precisa criar aqueles sites mirabolantes com os seus 189723871 trabalhos que você na real nem tem e vai ficar desesperado lendo isso. Um portfólio para um dev pode ser uma simples conta no Github com alguns códigos, que podem ser até de estudos mesmo! Você pode ajudar em issues de tradução, criação de documentação, pesquisar por tags como [starter](https://github.com/search?l=&q=label%3Astarter&ref=advsearch&state=open&type=Issues&utf8=✓),[beginner](https://github.com/search?l=&q=label%3Abeginner&ref=advsearch&state=open&type=Issues&utf8=✓) e começar a ajudar as pessoas com as coisas fáceis. Eu escrevi um [post](https://willianjusten.com.br/a-importancia-do-github-para-desenvolvedores/) falando exatamente da importância do Github para o Desenvolvedor.

Você também pode ajudar mais na comunidade, escrever num blog, participar de eventos, palestrar, etc, isso também ajuda bastante. Mas lembre-se, o objetivo não é começar a fazer essas coisas só porque você quer ganhar visibilidade. Essas coisas vem naturalmente, faça um trabalho bem feito e as pessoas vão começar a te seguir e os convites vão rolar =)

Outra coisa bastante importante para o RH lá fora é ter um [Linkedin](https://www.linkedin.com/in/willianjusten) bem completo, pois muitos headhunters procuram por ali e se você tiver algumas coisas por lá, fica mais fácil de entrarem em contato contigo.

## Mire nas empresas e vai na fé!

Se você acha que já tem tudo acima e só tava com medo [perde esse medo](https://willianjusten.com.br/o-que-preciso-saber-para-trabalhar-no-exterior/" \l "perde-esse-medo)! Aplique para as empresas e vê no que vai dar. Talvez você seja reprovado na primeira, na segunda, na décima! Mas o importante é não abaixar a cabeça, não desistir e continuar tentando que uma hora vai!

Sobre onde procurar vagas, vou fazer aqui um jabá, a [Toptal tá sempre contratando](https://www.toptal.com/careers), mas tem também o [board do Stackoverflow](https://stackoverflow.com/jobs), o [Linkedin Jobs](https://www.linkedin.com/jobs/) e se algum de vocês conhece outros maneiros, manda que eu coloco aqui!

## Conclusão

Bom gente, espero que com esse post vocês entendam que não é um bicho de sete cabeças trabalhar para uma empresa do exterior e que o mais importante está nas Soft Skills. O negócio é não ter medo de tentar, como já diziam "Quem não arrisca, não petisca!". E lembrando, isso é uma série de posts, se ficou faltando algo, provavelmente virá em um outro post, então fica ligado nas redes para não perder o próximo post!

https://willianjusten.com.br/o-que-preciso-saber-para-trabalhar-no-exterior/

# Devs Brasileiros, Gringos e trabalhar no exterior

## Será que a diferença é tão grande assim? Vamos conversar um pouco sobre isso.

## Introdução

Faaaala pessoal, o post de hoje vai ser retirado da minha primeira iteração com as respostas do [formulário](https://willianjusten.typeform.com/to/MtDdfi) que passei pedindo para as pessoas me perguntarem coisas sobre Trabalho Remoto, viagens e etc. Todos esses posts vão ficar separados na série chamada [Trabalhar no exterior, remoto, viagens e etc.](https://willianjusten.com.br/series/" \l "trabalhar-no-exterior-remoto-viagens-etc).

Tiveram muiiitas perguntas que acho que dá para criar uns 50 posts só disso. Mas uma pergunta que teve bastante repetição e por isso resolvi falar especificamente sobre foi:

Como é trabalhar com pessoal de fora? O que preciso aprender para ficar no nível deles?

Como esse é um assunto bem interessante e pertinente, resolvi começar por ele. Relaxa que vai ser um post bem levinho e pequeno, sem nenhuma treta. São só algumas coisas que fui percebendo sobre meu tempo morando na Europa e trabalhando com gringos. Eu vou falar um pouquinho também sobre a europa em si, pois isso influi diretamente no comportamento do dev.

Enquanto vou escrevendo, vou ouvindo o primeiro album da banda [Greta Van Fleet](https://open.spotify.com/album/6CGAeo2AyBkblQpnD9Gj22), que para mim é o Led Zeppelin da atualidade. Já tinha postado sobre eles antes e agora eles merecem destaque de novo, som de altíssima qualidade.

## Resumo para os preguiçosos

Trabalhar com o pessoal de fora é praticamente a mesma coisa que trabalhar com outros brasileiros. O nível deles não é mais alto que o nosso, pelo contrário, julgo a dizer que, em boa parte, eles estão abaixo do nível dos devs brasileiros. O que diferencia muito o dev brasileiro para o gringo é o estilo de vida e vou explicar um pouco sobre isso se você quiser continuar lendo =)

## Nível dos Devs Gringos

Bom, a primeira coisa que eu vou falar é nesse aspecto. Será realmente que o nível dos devs gringos é maior que os devs brasileiros? Eu pude participar de muiiitos meetups e pequenos eventos enquanto estive na Irlanda. Eu fui em Meetups como o NodeJS Dublin, JS Dublin, React Dublin, Google Developers e vários outros. E a primeira coisa que percebi é que muitas das coisas que o pessoal discutia e apresentava eram, em sua maioria, coisas antigas e ou muito básicas. Sim, isso mesmo que você leu, eu fui em Meetup que tinha "Introdução a AngularJS" e não era das versões novas não, era do primeirão mesmo...

Também ao conversar com o povo, pude perceber o como a galera ainda era receosa de atualizar as coisas e estudar coisas novas. Lembro de ter ido num Meetup da Zendesk (que tem um escritório lindo e enorme <3) e o cara estava falando de ES6 para o povo e quando ele perguntou quem já tinha ao menos testado algo, quase ninguém levantou a mão! Isso foi em Novembro do ano passado gente!

É claro que tem devs muito bons e técnicos, eu pude conhecer vários. Mas não vá pensando que todo dev gringo é um [Dan Abramov](https://twitter.com/dan_abramov) da vida e que é melhor que você.

## Estilo de Vida e como trabalham

É aqui onde as coisas mudam muito entre o brasileiro e o gringo. Novamente, isso se aplicando a minha realidade e ao que vi (isso pode variar em outras regiões). Dentro do trabalho eles são bastante sérios, diretos e focados.

É super normal você fazer um código e submeter para code review e os caras serem secos e falarem na lata que tá ruim, para fazer X ou Y. No Brasil eu acho que as vezes a gente tem aquela coisa de "tentar amenizar" a parada e falar "com jeitinho". O europeu não vai ter disso não, se estiver errado/ruim, ele vai falar "seco". Para quem não está acostumado com esse estilo, pode achar que os caras são grossos, arrogantes e outras coisas mais. Só que na realidade é só a forma deles falarem mesmo, sempre tentando ser o mais concisos possível. E no final, sendo muito sincero, isso é ótimo! Eu prefiro ter um feedback importante sobre o meu código para que eu possa evoluir, do que ficar sempre recebendo tapinha nas costas e não crescer.

E outra coisa extremamente importante é que o Europeu preza ainda mais sua vida fora do trabalho. Deu 17/18h o pessoal está saindo e é isso aí. O mundo está caindo, eles vão desligar o computador e ir embora. Não tem essa cultura de que "aquele que trabalha até mais tarde que tá mandando bem", raramente você vai ouvir relatos de pessoas que ficaram até tarde nos escritórios.

Outra coisa é a pontualidade, brasileiro é famoso em tudo que é canto pelos atrasos. E bom, isso é verdade, acho que não teve uma empresa brasileira que eu tenha trabalhado que não tenha tido atrasos constantes (até eu me incluo na parte de se atrasar). Como os gringos querem entrar e sair corretamente do trabalho, eles também prezam as reuniões e os tempos determinados. Se a reunião é para começar as 15h, ela vai começar as 15h tendo todo mundo ou não. E se é para demorar 30min, ela vai durar isso. Se for notado que o papo vai estender, já olham a agenda e separam um outro momento para isso.

## Situação Financeira, estudos e pressão

Isso é uma coisa interessante que estava até conversando com uns amigos hoje. Todos sabemos que a situação financeira acaba influindo no comportamento das pessoas e como elas vivem. Por mais que você diga "Dinheiro não é importante, o importante é trabalhar no que gosta.", o mundo é capitalista e não funciona 100% assim.

Os europeus, em sua maioria, tem uma situação financeira melhor que nós brasileiros. E isso acaba lhes dando um pouco mais de comodidade, que pode vir a ser uma coisa boa ou ruim. Por terem essa facilidade, eles acabam querendo aproveitar mais da vida e muitos não estudam tanto ou evoluem fora do trabalho. Afinal de contas, eles já vivem bem e tem sua estabilidade.

Por outro lado, o brasileiro vive na incerteza, as vezes você não sabe se vai estar empregado até amanhã, além da taxa de desemprego e pessoas fora da área de trabalho ser muito grande. E por causa dessa pressão e necessidade de se mostrar eficiente e bom, o brasileiro estuda no trabalho, em casa, no feriado, final de semana, dia santo e as vezes tá dormindo e estudando. E sabe o que acontece?

Ele acaba virando um desenvolvedor quase sempre atualizado e com bons conhecimentos, mas ao mesmo tempo lhe bate a síndrome do vira lata, porque ele está estudando e com isso está vendo grandes desenvolvedores de fora e pensando que todos são como aqueles caras.

Tenta pensar naquele seu parceiro do trabalho que manda absurdamente bem, mas tá sempre achando que é ruim e que a galera de fora é muito melhor. Te garanto que você vai achar vários exemplos. Manda esse post para ele, vai que ele acorda um pouquinho =D

## Eventos e Meetups

Uma coisa que na Irlanda eu e uns amigos brincávamos era como tinha Meetup. Você podia comer e beber de graça todos os dias a noite, afinal de contas tinha Meetup todo dia! Isso é uma coisa bem bacana lá de fora e que ajuda o povo a ter um networking legal e com isso muitos ganham mais visibilidade e evoluem mais rápido.

No Brasil temos alguns eventos muitos maneiros e em São Paulo tiveram já dois anos tendo a semana Frontend. Isso foi fantástico! Mas ainda é muito pouco se comparar com grandes polos daqui de fora. E não pode ser só em São Paulo, eu sou do Rio (na real da Serra do Rio) e eu ouvi vários amigos comentando como que tem estado morta a comunidade...

Eu sei como é difícil organizar essas coisas, tenho amigos que tão aí fazendo evento todo ano e eles contam como é pedreira. Então, se você não sabe como ajudar na área, já pensou em se oferecer para ajudar nos eventos? E lembre-se, o evento não é só no dia. Na realidade o antes e o depois são muiiiito mais trabalhosos e os caras tão sempre precisando de ajuda.

## Estrutura, empresas, transporte...

Muito pela Europa ser mais evoluída, a estrutura das empresas acaba sendo muito melhor. E não, eu não estou falando sobre a empresa dar aquela cervejinha e lanchinho ~~e salário baixo~~. Eu digo sobre o equipamento para trabalhar, mesas, cadeiras, prédios com boa acústica para não ficar um ruído horrendo o tempo todo. Além é claro, os salários são muito bons. Os salários de desenvolvedores são sempre bem acima do salário mínimo, o que já te permite viver de forma mais confortável. Quando você não precisa se desesperar se vai ter dinheiro para pagar o aluguel, você acaba tendo a cabeça mais limpa para poder trabalhar.

Outra coisa importantíssima da Europa é que o transporte aqui funciona maravilhosamente bem em quase todas as grandes cidades. Existem várias opções e com isso, o tempo de deslocamento para o trabalho passa a ser bem menor ou menos sofrido.

Pensa no brasileiro comum, a média de deslocamento que ele leva para o trabalho é coisa de 1h. Claro que tendo pessoas com menos tempo e pessoas que chegam a até 2h! Então, o que acontece é que essa pessoa já precisa acordar bem mais cedo, passa um tempão até chegar no trabalho, em geral, num metrô lotado ou num trânsito caótico. Isso é algo extremamente desgastante, porque a pessoa já chega no trabalho cansada, com isso ela produz menos. Aí para compensar, o que a pessoa faz? Isso, ela fica mais tempo no trabalho e sai mais tarde. Só que com isso, ela chega ainda mais tarde em casa e aí tem todas as coisas que precisa fazer da vida pessoal, quando vai ver, acabou o dia e a pessoa sequer descansou ou estudou um pouco. E no dia seguinte segue a mesma rotina.

Esse padrão de comportamento é repetido por muitos brasileiros, mas não ocorre com os europeus. Isso facilita com que o Europeu possa ter uma vida mais saudável e possa também estudar e trabalhar melhor.

## Como o Br é visto no exterior?

Isso é uma parada bem legal. Sempre quando eu conversava com a galera e perguntavam de onde eu vinha e eu falava do Brasil, eu diria que em 90% dos casos eles faziam expressão de impressionados e falavam que conheciam muitos brasileiros ótimos. Já teve muito RH falando comigo se eu não conhecia brasileiros para vir para algum país do lado de cá. Isso é porque nós somos conhecidos como pessoas que trabalham muito e são muito esforçadas. Porque como eu disse acima, o europeu preza muito mais a vida pessoal e nós brasileiros estamos acostumados a "se matar" tanto que até mudar o mindset que não precisa de tudo isso, demora um pouco.

## Conclusão

Como eu falei, o post não seria nenhuma treta nem nada. O que eu concluo a partir de tudo isso é que temos muitos brasileiros fantásticos e bem superiores aos gringos. Os europeus tem mais facilidades para evoluir, mas isso não significa necessariamente que todos aproveitam essas oportunidades. Nós brasileiros temos boa imagem na área de desenvolvimento no exterior e não é impossível trabalhar para uma empresa estrangeira, as vezes é até mais fácil.

# Mudando sua vida através de projetos paralelos

## Fica aqui um relato de alguém que mudou sua vida e quer mudar ainda mais.

## Introdução

Fala pessoal, quando eu escrevi o post [O que ninguém diz para os iniciantes](https://willianjusten.com.br/o-que-ninguem-diz-para-iniciantes/), eu dizia que aquele talvez fosse o post que mais me emocionava. Mas eu acho que estou ficando muito molenga, porque esse post vai ser mais um que vou me emocionar enquanto escrevo.

Para não perder o costume, a playlist que estou escutando hoje é uma de minha autoria e especial para esse post, que é [Side Projects](https://open.spotify.com/user/willianjusten/playlist/031nQBF66JwosHhCH9Degg). Na playlist você vai encontrar, em sua maioria, bandas indies e desconhecidas (sim, ouço muita coisa desconhecida), mas que você talvez curta =)

Esse post deve acabar ficando um pouquinho grande e por isso, vou separá-lo em alguns tópicos principais. Que serão:

* [Um mergulho na minha história](https://willianjusten.com.br/mudando-sua-vida-atraves-de-projetos-paralelos/" \l "historia)
* [Meus cursos](https://willianjusten.com.br/mudando-sua-vida-atraves-de-projetos-paralelos/" \l "meus-cursos)
* [Novos rumos](https://willianjusten.com.br/mudando-sua-vida-atraves-de-projetos-paralelos/" \l "novos-rumos)
* [Novos projetos](https://willianjusten.com.br/mudando-sua-vida-atraves-de-projetos-paralelos/" \l "novos-projetos)
* [Conclusão](https://willianjusten.com.br/mudando-sua-vida-atraves-de-projetos-paralelos/" \l "conclusao)

Bom, sem mais delongas, vamos ao post! Prepara um cafézinho que o post vai ser um pouquinho grande, mas lê até o final please, não sou eu quem está pedindo, é esse lindo gatinho.

## Um mergulho na minha história

Quando eu lancei o meu blog, há mais ou menos 1 ano e meio atrás, eu fiz uma [pequena apresentação sobre mim](https://willianjusten.com.br/making-of-parte-1/), falando que passei por outra área antes de chegar de fato a programação. Mas vou completar essa história com mais alguns pedacinhos. Eu sempre gosto de saber a experiência de vida das pessoas, o que fizeram antes, quais suas escolhas para tal, então por que eu mesmo não fazer né?

Bom, eu como todo nerd/geek sempre fui apaixonado por jogos, como eles funcionavam, a lógica e a física me impressionavam. E por causa disso, lá pelos idos de 2000, acabei conhecendo uma parada chamada [Rpg Maker](http://www.rpgmakerweb.com/), era uma engine beeeem tosca, mas que te permitia criar joguinhos de RPG com interface gráfica e alguns pequenos detalhes de lógica todo fundamentado em condicionais, ali deve ter sido meu "primeiro contato" com "programação". Criei vários joguinhos com meus primos, me divertia muito.

Aquele foi meu primeiro contato, mas depois passei a ser só um player normal. Comecei a me apaixonar por Química, porque ela me permitia entender como as coisas funcionavam, sim, eu sempre fui aquele que futucava tudo para entender. E então eu fiz um curso técnico em Química e logo após, entrei para uma faculdade de Química muito boa.

Naquela época eu jogava um MMORPG chamado [World of Warcraft](http://us.battle.net/wow/pt/), mas jogava no servidor oficial da Blizzard mesmo. Mal entrei na faculdade e conheci uns amigos que jogavam em um servidor pirata chamado [Mod World](https://www.youtube.com/watch?v=RHNkvHmh8Ac). Fiquei impressionado com aquilo, para as pessoas que conhecem o jogo, sabem que ele é gigantesco, fiquei curioso como as pessoas conseguiram fazer o jogo funcionar num servidor não oficial e ainda chamar tantas outras pessoas para jogar. Resolvi jogar nesse servidor pirata para conhecer e também porque era mais divertido jogar com meus amigos de faculdade.

Ao mesmo tempo, na Química eu estava começando a conhecer o pessoal do Laboratório de Química Computacional, eles precisavam de alguém que sabia mexer com Linux, que gostasse de Química e quisesse aprender mais sobre modelagem 3d. Na hora eu me ofereci e acabei ganhando uma bolsa de iniciação científica. Comecei a trabalhar bastante com computação desde então, eu aprendi a mexer em clusters (mas antes eu quase ferrei um =x), aprendi Fortran, Matriz Z, OpenGL e um pouquinho de C/C++.

E eis que o inesperado aconteceu, infelizmente o fundador do servidor de WoW havia falecido de um infarto fulminante e pediram então se alguém se voluntariava a tentar tocar o barco no lugar dele, só enquanto as coisas se ajustavam. Como eu sou maluco, me ofereci. A questão é que eu num tinha a mínima ideia de como era gerenciar um servidor de um jogo, ainda mais um tão grande e com tantas coisas.

A primeira merda que deu, foi que a máquina atual que ficava o jogo era muito cara e sem o fundador, não teríamos como pagar. Então me foi encubida a tarefa de escolher um novo lugar para hospedar o jogo. E vocês acham que eu sabia como escolher? Eu não tinha a mínima ideia! Mas nosso amigo Google é realmente um companheiro, bastou algumas pesquisas, falar com as pessoas certas e nos fóruns corretos e descobri qual seria a melhor configuração. Depois disso, ainda precisei aprender como fazer tudo rodar, conectar o banco de dados, nossa, foram alguns dias intensos. Faltei aula a semana toda só para me dedicar a isso e no final deu tudo certo! Ver a galera gritando nos canais meu nome e agradecendo, foi demais, uma das melhores sensações que eu já tinha tido, eu finalmente tinha sido útil! Foi nesse mesmo servidor que eu fiz meu melhor amigo até hoje, um gaúcho chamado Leonardo Silva, conheci uma guria que namorei por 5 anos e fiz outros grandes amigos até da nossa área como o Yuri Piratello, que sempre me ajuda em umas dúvidas idiotas.

Só que fazer faculdade, iniciação científica e manter o servidor estava ficando muito difícil e eu tive que desistir desse que foi meu primeiro projeto paralelo... Passei mais um ano arrastado na faculdade e decidi que não era mais isso que eu queria, eu gostava mesmo era de computação, esse negócio de fórmulas de Físico Química estava chato demais. Larguei a faculdade prestes a terminar e fiz minha primeira grande mudança, voltar para minha cidade natal e começar um curso de TI!

Quando eu voltei para minha cidade, eu precisava arrumar um emprego, porque não tinha como minha mãe ficar me sustentando. E o meu primeiro serviço na área foi como professor num cursinho de informática, isso mesmo, eu ensinava Office, Linux e Hardware... Na faculdade, como eu já sabia um pouquinho mais dos assuntos, passei quase o curso todo auxiliando meus amigos, cheguei até dar aula de programação para um grupinho. Um amigo, Felipe Borde, sempre falava que eu deveria dar aula, que eu explicava muito bem. Eu não sabia o que isso iria virar né...

Dentro da faculdade de TI, eu resolvi criar esse blog, meu segundo projeto paralelo de grande importância. Meu amigo [Guilherme Louro](http://guilhermelouro.com.br/) ficava me sacaneando para eu criar logo um site e foi assim que eu criei. Nesse blog, eu me dediquei a sempre passar conteúdo que eu achava importante e também para eu fixar melhor algum assunto. Foi através desse blog que eu comecei a ficar mais ativo na comunidade, acabei conhecendo várias e várias pessoas da área, comecei a palestrar, fui trabalhar na [globo.com](http://www.globo.com/) e depois na [HUGE](http://www.hugeinc.com/).

E agora, eu diria que comecei meu grande terceiro projeto paralelo, que é a criação dos meus cursos. Hoje completam 2 meses que lancei meu [curso sobre criação de sites estáticos com Jekyll](https://willianjusten.com.br/um-novo-projeto-um-novo-desafio/), logo depois criei um [curso sobre Git e Github](https://www.udemy.com/git-e-github-para-iniciantes/) e agora estou fazendo a [pré-venda do meu curso de SVG](https://www.udemy.com/aprendendo-svg-do-inicio-ao-avancado/?couponCode=PROMOJUL21).

Isso é um pouquinho da minha história, sempre fui meio maluco de meter as caras, mudei bastante através dos meus projetos paralelos e continuo mudando. E reparei que a cada projeto paralelo, uma grande mudança veio junto.

## Meus cursos

Continuando a falar dos cursos que já lancei, queria agradecer imensamente a todos que já se inscreveram, que compartilharam e principalmente, a todos aqueles que já assistiram e finalizaram o curso, vocês são demais!

Vou passar algumas estátisticas para vocês entenderem o quanto estou feliz e também porque quero sempre deixar tudo transparente para vocês, nessa nova etapa.

Já foram 1957 alunos inscritos! (Editado)

Segue o gráfico referente ao [curso de Jekyll](https://www.udemy.com/criando-sites-estaticos-com-jekyll/)

Como podemos ver, foram mais de 350 horas assistidas, aproximadamente 7500 vídeos tocados e cerca de 40% das pessoas que iniciaram o curso, finalizaram ele. Isso é incrível!! <3

E para o [curso de Git e Github](https://www.udemy.com/git-e-github-para-iniciantes/) os números são ainda mais surpreendentes para mim.

Foram mais de 600 horas assistidas, com mais de 14 mil vídeos tocados, sendo que mais de 50% alunos finalizaram o curso! Ou seja, mais de 800 pessoas melhoraram seu conhecimento em Git, uma ferramenta que é fundamental para o nosso dia a dia como programador.

## Novos Rumos

E assim como no meu primeiro projeto paralelo teve uma grande mudança, que foi sair da Química para TI. No meu segundo projeto paralelo também tiveram as mudanças de empresas e conhecimento. Para fechar meu terceiro projeto paralelo, eu tinha que ter uma grande mudança também!

Anuncio aqui, que saí da HUGE para me dedicar totalmente aos meus projetos paralelos durante um tempo e também um sonho que tenho há muito tempo, que é viajar pelo mundo, conhecendo novas culturas, novas pessoas e claro, vendo como são os programadores em cada lugarzinho. Sempre me imaginei como na cena abaixo:

E para conseguir essa grande mudança, eu venho me planejando já há algum tempinho. Juntei um dinheiro enquanto estive nas empresas grandes que passei e agora comecei com essa iniciativa dos cursos online. Fiz cursos gratuitos para mostrar as pessoas como seria meu conteúdo/didática e então depois lancei cursos pagos, que também tiveram ótima recepção.

## Novos Projetos

Além desse novo rumo que estarei tomando para minha vida, também pretendo criar novos projetos, tudo sempre em prol de ajudar a comunidade e retornar a ajuda que vocês me derem para esse meu sonho. Pretendo sim criar novos cursos de diversos assuntos que vocês tem me pedido e algumas coisas bastante novas que nunca vi sendo feito por aqui. Espero que gostem das surpresas que tem por vir =D

## Black Friday

E se você leu esse post, ou só desceu até aqui, seguem aqui cupons dos meus cursos com promoção destruidora com todos os preços no mínimo! Clique nos links abaixo ou então use o código BLACKFRIDAYJA.

* [Curso de JS com TDD + ES6](https://www.udemy.com/js-com-tdd-na-pratica/?couponCode=PROMOJUL21)
* [Curso de ES6](https://www.udemy.com/curso-completo-de-javascript-moderno-es6/?couponCode=PROMOJUL21)
* [Curso de SVG](https://www.udemy.com/aprendendo-svg-do-inicio-ao-avancado/?couponCode=PROMOJUL21)

## Conclusão

Bom galera, é isso aí, a ideia de criar esse post era passar um pouquinho da minha vida, como ela foi mudando de acordo com meus projetos paralelos. Mostrar para vocês que suas vidas podem mudar e muito fora do seu trabalho principal. Se você faz administração, se você faz química, seja lá o que for, busque o que você ama, mesmo que possa parecer a pior escolha naquele momento, tudo que você faz por amor acaba sendo mais reconfortante e mais valoroso no final.

Você não tem como saber onde os seus projetos paralelos vão te levar. Quando você começa a trabalhar neles, você ainda não consegue ver claramente os benefícios que ele está te trazendo. Mas esses projetos podem trazer várias coisas. Seja um novo portfólio que vai trazer um próximo trabalho ou cliente. Pode ser algo que você se apaixone tanto que te revigore e faça voltar a ter amor pelas coisas e pela sua vida. Enfim, um projeto paralelo pode significar várias coisas e tudo que você precisa é começar!

# Minha trajetoria até me tornar Desenvolvedor

## Uma pequena entrevista onde falo sobre meus primeiros trabalhos, desafios e meu futuro.

## Introdução

Fala pessoal, o [William Oliveira](https://twitter.com/w_oliveiras) me pediu para participar de uma entrevista para o [Training Center](https://github.com/training-center/sobre), demorei demaaaais para fazer, mas acabei enviando para ele esses dias. Você pode ler no [Medium](https://medium.com/trainingcenter/como-é-trabalhar-como-desenvolvedor-front-end-remoto-por-willian-justen-e7cdf7fe2e64) ou só seguir o post aqui, que a entrevista está na íntegra também.

Enquanto escrevi a entrevista e até agora, estou ouvindo a mesma banda, que é [Mt. Wolf](https://open.spotify.com/album/4WBiO5swd0JJnrDyo34CPg), eles tem um som muito legal e ótimo para relaxar, se curte um som mais calmo e atmosférico, vai sem medo =D

Bom, vamos ao post que interessa, mais um pouquinho da minha história em palavras curtas.

## Apresentação



Olá, meu nome é [Willian Justen](https://twitter.com/Willian_justen) e sou de uma cidade na serra do Rio de Janeiro chamada Petrópolis (carinhosamente apelidada de Silent Hill brasileira). Tenho 27 anos e atualmente sou Desenvolvedor Front-End no Core Team da [Toptal](http://toptal.com/), onde trabalho remotamente com pessoas do mundo todo, mas antes de trabalhar nela, trabalhei também em algumas empresas bem interessantes como [HUGE](http://www.hugeinc.com/), [Globo.com](http://www.globo.com/) e [Queremos/WeDemand](https://queremos.com.br/).

Escrevo num [blog pessoal](https://willianjusten.com.br/) e também crio [cursos](https://willianjusten.com.br/cursos/), que hoje estão sendo transferidos para a plataforma da [Udemy](https://www.udemy.com/user/willian-justen-de-vasconcellos/).

### Como você conheceu a área de Desenvolvimento?

No meu blog eu fiz um [post](https://willianjusten.com.br/mudando-sua-vida-atraves-de-projetos-paralelos/" \l "historia) falando um pouquinho da minha história e como passei a me interessar por Desenvolvimento. Mas basicamente foram vários pequenos encontros, que foram me deixando cada vez mais e mais interessado pela área.

Quando mais novo eu amava brincar com o [RPG Maker](http://www.rpgmakerweb.com/) e sonhava em trabalhar numa Nintendo ou Blizzard da vida, criando jogos. Mas acabou que foi chegando a época de vestibular e como eu era muito bom em Química, eu decidi ir para Química. Sim meus amigos, eu fiz 4 anos de Química Industrial e pasmem ainda mais, eu desisti no último período =D

Dentro da Química, eu acabei descobrindo uma área que era de Modelagem Molecular, onde basicamente utilizávamos grandes computadores (clusters) para fazer cálculos e predizer estruturas moleculares para então criar esses elementos em 3D. Foi paixão a primeira vista e eu comecei a aprender ainda mais de programação, como Fortran (que vida...), OpenGL, C/C++, Python, além de aprender mais sobre Linux e grandes servidores.

E como você pode notar, eu fui deixando a Química totalmente de lado. Quando surgiu a oportunidade de administrar um servidor privado de World of Warcraft (não só instalar e rodar, mas mexer no código mesmo), foi o momento que eu repensei e falei para mim, ok, é isso que eu quero. Quero ser desenvolvedor! Voltei a morar na minha cidade natal, iniciei um curso de TI e estamos aí até hoje.

### Por que você escolheu ser Desenvolvedor Remoto?

É engraçado dizer isso, mas eu resolvi ser Desenvolvedor por causa das frustrações/felicidades que eu tinha cada vez que eu tentava fazer algo. Como eu comecei na área como um curioso, eu passava horas e horas fazendo um monte de merda e tentando entender por que não funcionavam e, as vezes, para entender por que funcionavam também. E sempre depois dessas frustrações, quando eu entendia, vinha uma felicidade tão grande, que me fez ver que era isso que eu queria para mim!



Outro grande motivo de eu me interessar pela área de Desenvolvimento foi a comunidade, apesar de sempre ter um ou outro idiota, tem muita gente querendo se ajudar. Isso é algo que eu não vejo tanto em nenhuma outra área!

E o desejo de trabalhar remoto começou quando eu decidi sair da empresa que estava, para iniciar meus projetos paralelos e viajar um pouco pelo mundo. Minha primeira forma de trabalhar remoto, foi criando meus cursos, que uniam o útil ao agradável, eu gosto muito de ensinar e poderia usar disso, para tentar juntar uma graninha para viajar. Fiz até o post [Mudando sua vida através de projetos paralelos](https://willianjusten.com.br/mudando-sua-vida-atraves-de-projetos-paralelos/) explicando esse desejo de mudança.

Muita gente da área vislumbra poder trabalhar remoto, alguns porque preferem ter mais tempo em casa, outros porque acham que assim não vão se chatear com chefes, outros para tentar receber um salário melhor, enfim, são vários os motivos para se querer trabalhar remoto e são outros vários para não se querer também.

### Como foi o seu primeiro trampo (alocado)?

Eu costumo dizer que tive 2 primeiros trampos na área. O primeiro de todos foi exatamente quando eu saí da Química, eu tinha voltado para minha cidade natal, mas eu não queria ficar parado, já estava "velho" e tinha acabado de desistir de uma faculdade no final, se eu ficasse parado, a família iria encher o saco.

Eu fui virar então, um professor de cursinho de informática, ensinando Windows, Office, Linux e Redes/Hardware. Pode parecer um emprego bem ruim (na real era mesmo), mas aprendi muitas coisas com ele. Se hoje dou cursos e tenho didática, é porque observei demais como os meus alunos, desde crianças de 7-8 anos até idosos de 70 anos, iam aprendendo e absorvendo as coisas.

O problema desse emprego é que eu não estava necessariamente aprendendo nada novo da área e isso me chateava um pouco. Então na primeira oportunidade eu fui tentar uma vaga de estágio numa agência web em Petrópolis mesmo. E bom, foi lá que tudo começou.

A vaga era para desenvolvedor web, mas eu não sabia praticamente nada de web! O que eu fiz? Bom, uma semana antes de fazer a entrevista, eu resolvi montar um CV diferente, para tentar chamar a atenção, já que eu não tinha tanto conhecimento em web, mas eu já tinha um pouco de conhecimento de Photoshop e outras linguagens. Eu criei então um CV, basicamente uma imagem, utilizando o formato do Windows 8, aquela interface Metro, com ícones e cores flat.

Viram meu CV e resolveram me chamar para uma entrevista, como a agência era pequena, o próprio dono da agência me entrevistou. Ele me perguntou se eu já tinha criado alguns sites, o que eu já sabia. E eu basicamente falei que nunca tinha feito nenhum site na vida, mas que eu já tinha mexido com programação e que poderia aprender qualquer coisa muito rápido, contanto que ele me desse uma chance. Ele era meio carrasco e cético, mas falou que iria me dar uma chance num contrato de estagiário por 1 mês e se ele gostasse, poderia continuar mais tempo.

Na semana seguinte a entrevista, eu comecei nessa agência no mesmo dia que meu mentor e grande amigo, [Guilherme Louro](https://github.com/guilouro). Ele, apesar de já ser bastante experiente, também estava em seu primeiro dia e então sentou do meu lado e fomos batendo papo e organizando as coisas para poder começar a trabalhar. Acho que a cada 5min eu chamava ele para me explicar alguma coisa, me impressiono como ele não me mandou a merda xD

E foi ali que eu comecei a dar meus primeiros passos na web. Eu aprendi muita coisa e muito rápido. Mas a coisa mais importante que aprendi foi:

Um bom mentor pode te fazer se apaixonar pelo que faz e pode fazer você querer crescer e fazer ainda mais.

Eu fiquei nessa empresa por 1 ano e meio, quando decidi que queria tentar alguma coisa na capital (RJ), pois lá tinham mais coisas, melhores salários e trabalhos mais interessantes e foi assim que fui parar na Queremos. O detalhe é que eu descobri essa vaga num tweet do Bernard De Luna.

### Como foi o seu primeiro trampo (remoto)?

Bom, essa é uma área relativamente nova para mim. Minha primeira experiência remota foi para uma empresa americana, que eu trabalhei como freela contratado por 3 meses para ajudar a construir um site para um evento. Foi muito legal e muito frustrante ao mesmo tempo. Foi a primeira vez que eu trabalhei tendo que falar numa outra língua que não o Português e bom, eu ficava extremamente nervoso para me comunicar, mas foi aí que eu descobri um primeiro truque.

Se você não conseguir se comunicar bem falando. Escreva! Você vai pensar 2x no que quer falar e aí vai ser mais fácil de ser entendido.

Sempre que tinha alguma coisa que eu não tinha entendido claramente, eu pedia para me mandarem um email e com isso, tudo ficava mais claro e a comunicação fluía. E outra coisa importante também é:

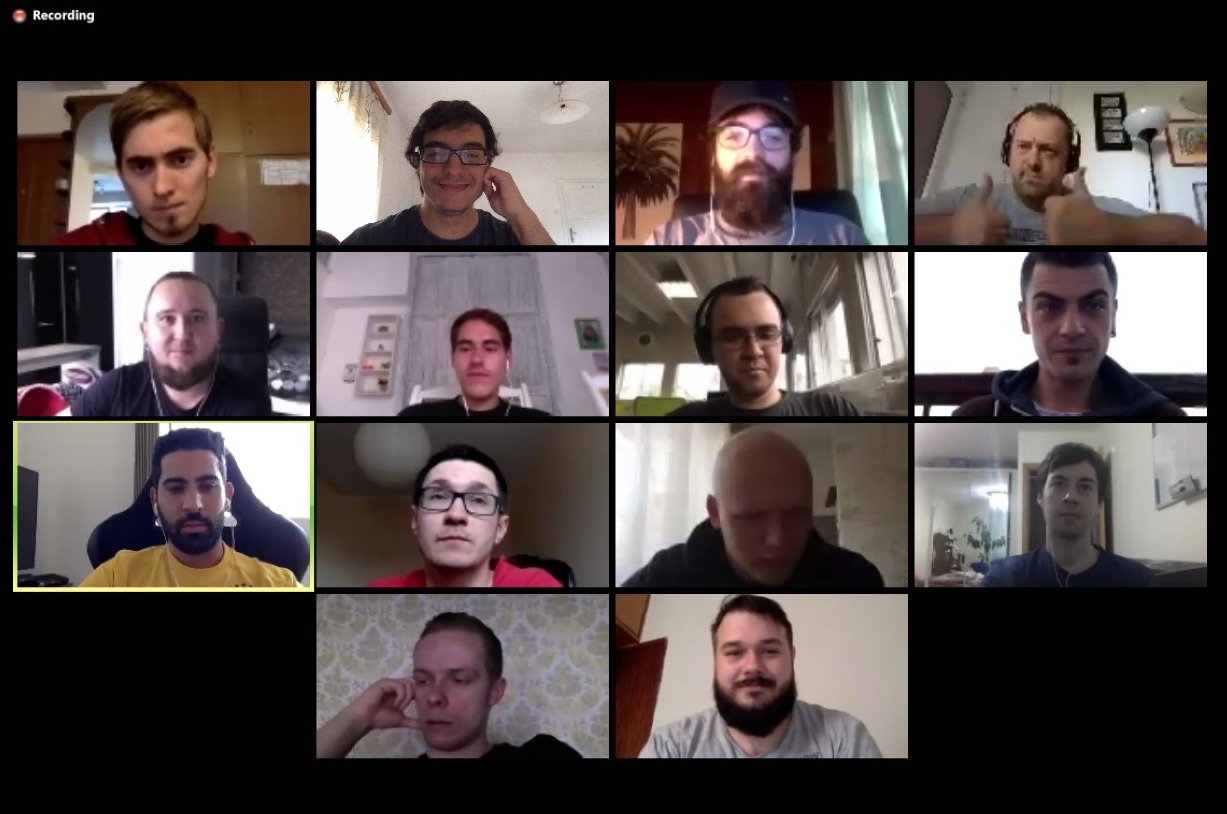
As pessoas não esperam que você fale como Shakespeare, eles querem saber se você entendeu, suas opiniões e o que você vai fazer.

E bom, o meu primeiro trabalho remoto de forma oficial tem sido o meu atual trabalho na Toptal. Muita gente confunde a [Toptal](http://toptal.com/), achando que ela é só uma rede de freelancers que prestam serviço para outras empresas. Mas não é bem assim não, existem váaarias pessoas que trabalham para manter a plataforma, como desenvolvedores, designers, diretores e outras váaaarias áreas.

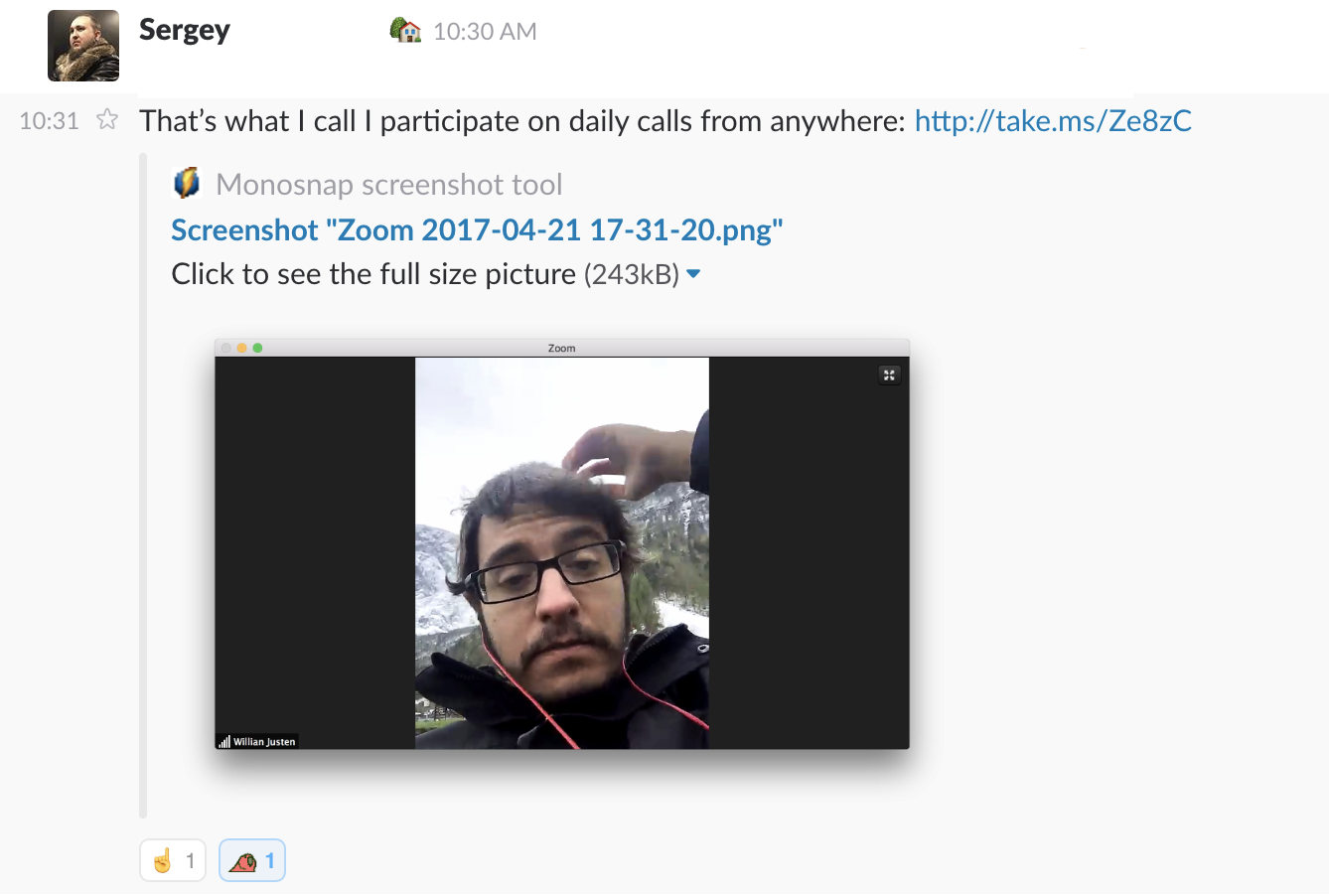
Farei depois um post falando do processo da Toptal em si, mas ele é basicamente dividido em: entrevista de inglês/perfil, entrevista+live coding com um desenvolvedor, outro projeto live coding com 2 desenvolvedores, conversa com o VP da empresa. Essa última etapa eu fiz na estrada, enquanto estava viajando pela Irlanda, foi bastante engraçado que parei na primeira cidadezinha que tinha, não devia ter nem 200 habitantes ali.

Na Toptal são vários times, cada um é responsável por algum pedaço da plataforma e/ou produtos internos. Nos comunicamos pelo Slack o dia todo, fazemos bastante pair programming usando o Screenhero ou qualquer outro app que compartilhe tela, temos daily como qualquer equipe, retrospectivas e updates semanais da empresa como um todo. Utilizamos o Gìthub para versionar nosso código e possuímos um processo de code review muito legal, onde todos tentam ajudar para ter o melhor código entregue no final.

Por mais que estejamos em todos os lugares do mundo, a gente tá sempre perto devido aos canais de comunicação e é bastante divertido saber onde cada um está, a Toptal influencia bastante a galera viajar, ir a eventos e etc. Abaixo segue uma foto de uma Daily do time:



E segue aqui um belo print que um amigo do time resolveu fazer, quando eu estava trabalhando sentado numa cadeirinha com as montanhas da Austria no fundo...



### Quais são as skills de quem trabalha nesta área?

Eu poderia falar uma lista imensa de tecnologias e coisas que saem a cada segundo, mas não é bem isso que se é preciso. O importante está nas soft skills, é ter vontade de aprender, não ter medo de errar ou não saber algo, ter paciência e disciplina.

Para ser um desenvolvedor você precisa se acostumar a se frustrar bastante até conseguir algo, mas a consquista final vale por toda a frustração. Mas calma, não precisa achar que ser desenvolvedor é um sofrimento sem fim, muito pelo contrário, essa busca por entender as coisas e por fazer funcionar, te molda e te melhora a cada segundo, por isso é bom ter paciência e dedicação.

Para quem deseja trabalhar remoto, o mais importante de tudo é disciplina e comunicação. Trabalhar de casa ou onde for não é uma tarefa fácil, você possui muito mais coisas que podem te atrapalhar e te tirar o foco. E claro, sua mentalidade não pode mudar, trabalhar remoto não significa que você vai poder dormir até tarde, ficar só viajando e jogando videogame. Você precisa criar seus horários, precisa de um espaço que consiga trabalhar adequadamente e comunicar qualquer coisa aos seus parceiros. Se você tiver algum problema/dificuldade, seja rápido e avise, assim o time pode te ajudar com isso. Se você não falar, ninguém terá como descobrir e isso é altamente prejudicial.

### Quais são os principais desafios da área?

O primeiro grande desafio é saber gerenciar seu tempo! E digo isso não só para quem trabalha remoto, mas para quem trabalha alocado também. Nossa área está em constante evolução e precisamos estar sempre estudando e nos atualizando (não, não estou falando para seguir a modinha do momento). A procrastinação é um problema grave e para isso existem várias formas que você pode fazer para tentar diminuir isso, escrevi um [post bem antigo](https://willianjusten.com.br/tecnicas-de-aprendizado/) sobre isso aliás.

E aliado a um bom gerenciamento do tempo, vem a disciplina, um desenvolvedor remoto precisa manter essa disciplina, por mais que tenham distrações, focar no que é importante e necessário, para então poder fazer as outras coisas que desejar.

### Quais são as principais recompensas da área?

A primeira coisa que eu destaco como uma recompensa é a comunidade, temos uma área muito legal, que tá sempre se ajudando e criando coisas novas todos os dias.

A outra são as oportunidades de trabalho, existem em todos os cantos e com salários muito bons, seja para trabalhar no Brasil como fora dele.

E focando no trabalho remoto, as pessoas costumam dizer que só isso já é uma recompensa. E bem, é uma recompensa e tanto. Você consegue ter uma melhor qualidade de vida, economizar tempo não fazendo deslocamento (pense que uma pessoa pode perder 4h indo e vindo do trabalho), estar mais perto da família, viajar com mais frequência e trabalhar de onde quiser, enfim, são várias vantagens.

### Você pensa em mudar de área?

Nossa área é tão ampla e tão legal, que não posso dizer que não posso estar fazendo uma outra coisa no futuro. Eu confesso que o sonho de criança/adolescente de ser um Desenvolvedor de Games ainda é bem grande, então, quem sabe um dia? Mas hoje, eu diria que estou bem feliz com o que eu faço.

### Por que alguém deveria se tornar um(a) desenvolvedor(a) remoto?

Se você é uma pessoa disciplinada, que tem um lado auto-didata forte e tem desejo de estar um pouco mais livre de escritórios e montar seu próprio tempo, o trabalho remoto é perfeito para você.

Mas uma dica que acho muito importante é, o trabalho remoto não foi feito para quem está começando e não é para qualquer um, pode frustrar e prejudicar mais do que ajudar. Eu digo isso porque é muito importante que no início você tenha contato com pessoas da área que poderão te orientar e ajudar, além de entender como são as dinâmicas de trabalho. Só assim você estará apto para adquirir mais disciplina e enfrentar o trabalho remoto, que como dito antes, tem muitas vantagens porém desafios bem grandes também.

# Será que sou bom o suficiente?

## Essa é uma pergunta bastante comum de nós desenvolvedores, e por que será isso?

## Introdução

E aí meu povo, com o intuito de voltar a postar com mais frequência, estou eu aqui para falar de um assunto não técnico novamente, mas que julgo bem importante e ligado a nossa área.

E para seguir na vibe desse meu post, já vai abrindo essa playlist chamada [Zen Focus](https://open.spotify.com/user/spotify/playlist/70X9CPfVWXmFqX5ObjSvOP), vários acústicos bem relaxantes e bons de se ouvir. Altamente recomendado se você precisar de foco para trabalhar numa tarefa estressante =)

## Primeiro algumas perguntinhas...

Só para ter uma ideia de como você se sente e se tem algumas similiaridades com isso, vamos lá:

### Opiniões sobre você

* Você, em geral, acha o seu trabalho pior do que os outros acham?
* Você fica em dúvida se deveria estar de fato no lugar que ocupa?
* Você se sente preocupado com seu conhecimento e se vê sempre abaixo dos seus amigos ou colegas de trabalho?

### Atitudes perto de seus superiores

* Você fica nervoso ou ansioso diante deles?
* Você fica preocupado se em algum momento seu superior vai descobrir que você não é tão bom e te demitir?

### Seu processo de desenvolvimento

* Você demora mais tempo tentando deixar tudo perfeito do que de fato finalizar as necessidades do projeto?
* Quando você sobe um código, você fica tão preocupado com o que seus colegas vão pensar que atrapalha o seu dia de trabalho?

Se você respondeu sim para boa parte das perguntas, isso é um sinal que você pode ter a Síndrome do Impostor. Mas calma! Não é o fim do mundo e você não é o único a ter isso, existem vários como você, quer um exemplo bem próximo? Eu mesmo, sim eu tenho Síndrome do Impostor há alguns anos e é claro, isso me complica bastante, mas não me impede de viver e crescer profissionalmente.

## Síndrome do Impostor

A síndrome do impostor (também conhecida como fenômeno de impostor ou síndrome de fraude) é um conceito que descreve indivíduos de alto desempenho que são marcados por uma incapacidade de internalizar suas realizações e um medo persistente de serem expostos como uma "fraude".

Quando você sente que nunca é bom o suficiente em algo e tem o constante medo de ser "descoberto" é o exato momento que a síndrome do impostor está atacando. Existe um gráfico bem interessante que mostram os dois opostos, que são o Efeito Dunning-kruger que é quando a pessoa se acha melhor e superior a pessoas mais capacitadas (~~os famosos analistas de facebook~~) e a Síndrome do Impostor, que é quando você é habilidoso, mas incapaz de ver isso.

É comprovado que essa Síndrome é bastante comum em profissões que possuem muitas cobranças e análises, como jornalistas, escritores e claro, programadores!

Há pouco tempo o Hubspot fez uma pesquisa sobre essa Síndrome na área de Desenvolvimento e encontrou resultados muito interessantes.

Praticamente todos os desenvolvedores já passaram ou passam por momentos da Síndrome do Impostor. E isso independe a experiência, pois até programadores com 10 anos ou mais já relataram passar por isso.

É claro que é normal os iniciantes passarem por mais momentos assim, pois eles vêem um mundo gigante diante deles e acabam se sentindo incapazes de evoluir e não percebem o quão rápido evoluem. Lembro de ter feito um post chamado [O que ninguém diz para os iniciantes](https://willianjusten.com.br/o-que-ninguem-diz-para-iniciantes/), onde vi várias pessoas com esse aspecto e que tentei mostrar que não precisava ser o fim do mundo.

## E por que nós programadores nos sentimos assim?

Bom, isso vária muito de pessoa para pessoa, mas existem algumas coisas comuns em nossa área:

* Nossa "competição" não é só em nossa cidade, é com qualquer um que tenha computador e uma boa conexão a internet
* Toda rotina de programação pode ser medida, o que significa que alguém pode fazer a sua tarefa ser 1000% mais eficiente e fazer você se sentir ainda pior
* Tudo se atualiza tão rápido que nunca conseguimos nos sentir 100% aptos aquilo que fazemos
* Usuários podem ser muito burros, mas a culpa sempre será do programador
* Mesmo depois de trabalhar, costumamos continuar usando computador, o que dá a sensação de estarmos trabalhando demais
* Boa parte de nós somos introvertidos, então buscar por ajuda pode ser algo bem difícil

E por aí vai, existem N razões que provocam essas sensações...

## E como lidar com isso?

### Veja a realidade como ela realmente é!

É claro que existem grandes desenvolvedores, mas você já parou para pensar em quanto tempo eles gastaram para estar ali? E já parou para pensar que duas vidas são muito distintas e que ele pode ter tido várias influências que colaboraram com o crescimento dele? Tente não se comparar, até porque no final todos somos muitos parecidos, bons em alguns aspectos e não tão bons em outros.

### Diminua seu medo de errar

Já parou para pensar que você só vai conseguir evoluir (e ver evolução) se você errar e te corrigirem para não fazer mais? Pense que a cada erro é um acerto no futuro e que além disso, a pessoa que te corrigiu irá enxergar que você já não erra mais naquilo e evoluiu.

Eu tinha muito medo de review de código, demorava horas/dias trabalhando em um mínimo trecho de código com um enorme medo de quem alguém pudesse olhar e falar que estava tudo errado ~~ou falar que tava uma merda mesmo~~.

Isso só me prejudicava pois eu demorava muito para entregar e os erros que eu cometeria sem saber, eu continuava cometendo, pois para mim não eram erros. Quando percebi que reviews são importantes para que você além de terminar sua tarefa melhor, seja guiado para o caminho certo, comecei a notar uma grande evolução no meu código entregado e também passei a aprender muito mais.

### Diminua sua ansiedade

A maioria dos engenheiros, principalmente os mais novos ~~ou gente como eu~~, tem uma necessidade gigante de produzir muito código bom em muito pouco tempo. Vá devagar, escreva seu código com calma, tente não se preocupar tanto com o tempo, escreva alguns testes, revise seu código, faça um pair programming se tiver como. Todos esses processos que "demoram" vão te ajudar a ter uma entrega melhor.

### Não tenha medo de perguntar

Muitas pessoas deixam de perguntar as coisas por medo de se "exporem" e serem "descobertas" como um impostor. Mas hey! Ninguém nasceu sabendo e muitas das vezes a outra pessoa já se ferrou tantas vezes com aquilo, que vai ser um alívio para ela saber que alguém também está passando pela mesma coisa.

O aliado das perguntas na nossa área é o pair programming, que é uma técnica onde duas pessoas programam juntas, como mamãe dizia "duas cabeças pensam melhor que uma".

### Dê um desconto a você

A área de desenvolvimento é gigantesca, existem milhares de linguagens, bibliotecas e outras coisas mais sendo lançadas na velocidade da luz. E você não precisa aprender tudo! Isso mesmo, não é porque ontem 5 desenvolvedores legais da Google resolveram lançar a máquina do tempo que você precisa já ser o mestre dela. Vá devagar, quando você tiver necessidade, você vai lá e aprende. Não se sinta mal se você não souber, você conhece alguém que sabe programar em todas as linguagens do mundo e bem?

### Procure por bons feedbacks

A Síndrome do Impostor é uma coisa muito interna, mas coordenadores e colegas de trabalho são bastante importante para ajudar em quão confiante você pode ser com seu trabalho e suas habilidades. Sempre escute os bons feedbacks e aprenda com eles, isso significa que você não precisa chegar amanhã na empresa se achando o melhor do mundo, mas também não precisa se sentir um impostor por estar ali. Escute bem as dicas e absorva tudo que for de bom e, caso você ouça algo ruim e que não lhe fará bem só deixe para lá e busque outros feedbacks melhores.

## Mais leitura

Se você ficou curioso com o assunto e quer conhecer outros programadores que também relataram esse problema, seguem alguns links:

* [Overcoming Impostor Syndrome or How I Learned to Stop Worrying and Love Coding by Alicia Liu](https://medium.com/@aliciatweet/overcoming-impostor-syndrome-bdae04e46ec5" \l ".g63job35d)
* [I’m an Impostor by David Walsh](https://davidwalsh.name/impostor-syndrome)

## Conclusão

Bom pessoal, é basicamente isso que eu queria passar para vocês. Estamos aí quase acabando o ano, geral definindo suas metas e se preocupando com sua vida e analisando como foi seu ano. Peço que vocês analisem o ano que passou, em quais momentos você se sentiu assim e o que você pode fazer para melhorar, essa será sua meta de 2017 a partir de então.

Eu não posso dizer que estou curado da Síndrome do Impostor e muito menos que não sinto nada, pelo contrário, ao mesmo tempo que escrevia eu fui lembrando dos várias momentos que tive esse ano. Mas aos poucos eu vou tentando melhorar isso, vou tentando me aliviar mais e ver que muitos dos problemas sou só eu quem crio e por isso eu posso deixá-los de lado.

Faça isso você também, enfrente essa dificuldade de se sentir sempre abaixo dos outros. Estude sim, não deixe de se dedicar, mas não persiga isso deixando tudo de lado só porque você se sente incapaz e um impostor.

# O que ninguém diz para os iniciantes

## Análise de uma famosa entrevista com Ira Glass sobre iniciantes e o processo criativo.

## Introdução

Fala pessoal, o post de hoje é totalmente inspirado em um vídeo bastante legal, que eu acho motivacional para caramba e que eu gostaria de ter assistido há mais tempo. O vídeo em questão se chama Nobody Tells This To Beginners e se trata de uma montagem em cima da entrevista de Ira Glass sobre esse assunto.

## Vídeo Inspirado

Se você estiver tendo problemas de visualizar o vídeo acima completo, tente [esse aqui](https://www.youtube.com/watch?v=D_gSFWt6ito)

## Transcrição Livre

"Ninguém diz isso para os iniciantes. Eu desejava que alguém tivesse me dito. Todos nós que fazemos trabalhos criativos, entramos nisso porque temos bom gosto. Mas existe uma lacuna. Nos nossos primeiros anos que fazemos coisas, elas não são tão boas. Elas tentam ser boas, tem potencial, mas elas não. Mas o seu bom gosto, a coisa que fez você entrar nesse jogo, ainda é matador. E o seu bom gosto é o que faz o seu trabalho te desapontar. Várias pessoas nunca passam dessa fase, elas desistem. Várias pessoas que eu sei que fazem trabalhos criativos e interessantes passaram por anos disto. Nós sabemos que nosso trabalho não tem aquela coisa especial que gostaríamos que tivesse. Todos nós passamos por isso. E se você está apenas começando ou ainda está nessa fase, você precisa saber que isso é normal e a coisa mais importante que você precisa fazer é um monte de trabalhos. Coloque uma meta para você e toda semana você vai terminar um projeto. Só passando por um grande volume de trabalho que você irá diminuir essa lacuna e o seu trabalho vai ser tão bom quanto suas ambições. E eu levei mais tempo para descobrir como fazer isso do que qualquer um que eu já conheci. Vai demorar. É normal que demore. Você só tem que lutar contra seu caminho."

## Análise e Introspecção

Enquanto fui escrevendo essa análise, fui ouvindo uma [puta playlist do Spotify](https://open.spotify.com/user/nickfearless/playlist/6EF56fuiUgN2GOMVZIiXpq), que aconselho a você ir escutando junto, talvez ela te inspire e te anime também.

Você já teve a sensação de que tudo que faz não é tão bom como você gostaria que fosse? Que pessoas de 20 anos fazem coisas incríveis, que você jamais se imaginou tendo capacidade de fazer? Que por mais que você estude, vire horas e horas, nunca é o suficiente para alcançar aquilo que você acha que é o bom? E aí você começou a se desgastar, se sentir péssimo, com vontade de desistir de tudo e tentar outra coisa?

Como o Ira Glass diz, isso é normal, você tem bom gosto, você sabe olhar bons trabalhos e dizer o quanto eles são bons.

O que você não pode fazer é comparar o seu trabalho com o dos outros e se diminuir, cada um teve o seu tempo de estudo, seu tempo de aprimoramento, melhoria e principalmente, todos passaram pela mesma fase que você está passando. O que você deve fazer é identificar e entender essa fase que você está passando, pois quando ela passar, você terá superado tudo e seus trabalhos terão alcançado o nível que você tanto almejava.

Você pode pensar que eu já passei por essa fase e me sinto ótimo, que eu tenho trabalhos legais, escrevo num blog que um monte de gente já conhece, que faço palestras. Mas é aí que você se engana, todo dia eu acordo com a sensação de que tenho muito a fazer ainda, que tudo que eu já fiz e faço, ainda não valeu a pena.

Mas eu finalmente estou começando a identificar isso e ver que é só uma fase. E o que estou fazendo para me sentir melhor e para melhorar isso? Começo projetos e projetos. Tenho colocado deadlines para eles e com isso vou aprendendo coisas que eu sempre quis e que se eu não tivesse criado esses desafios, talvez nunca teria começado.

Esses projetos podem ser qualquer coisa. Sempre teve vontade de escrever um blog? COMECE! Sempre quis contribuir no github? COMECE! Sempre teve vontade de palestrar e passar o seu conhecimento para a comunidade? COMECE!

Você pode passar o resto da vida achando que não tem o "conhecimento suficiente", que não tem a "experiência necessária", que as pessoas vão rir de você. Ou você pode começar a se mostrar, começar a fazer aquilo que sempre quis fazer, mas tinha medo. Se suas primeiras críticas não forem boas, CONTINUE, ninguém disse que seria fácil, mas ninguém disse que seria impossível também.

Existia um post do Daniel Filho chamado Queime seus ídolos. O que eu prefiro entender do post dele é que não é para você deixar de ter pessoas que você se espelha, mas elas são pessoas como você, elas também erraram bastante, elas passaram pelas mesmas fases que você. Não pense que elas estão lá por serem muito melhores que você, ninguém sabe pouco o suficiente. Aquele seu ídolo pode ser um gênio no JS e mal saber CSS, isso não faz dela ruim. Digo o mesmo para você, você está começando ou já está há um bom tempo na área, não se diminua, tente passar o pouco que você sabe para os outros e você verá que o pouco que você sabe já é muito para outros.

Há alguns dias eu postei na timeline do meu Facebook um vídeo bem interessante, mostrando o criador do Node.JS na sua primeira apresentação.

Um cara franzino, tímido, embolado com as palavras, que nem sabia se iria funcionar seu side project. E que hoje é só conhecido por ter ajudado na criação da ferramenta de desenvolvimento que mais cresce no mundo.

Ou seja, se você é tímido, tem vergonha de se apresentar, tem vergonha do que faz e nunca mostra nada para os outros. Perca esse medo! Contribua no Github, faça um blog, grave vídeos, se mostre! Você pode ser o mais novo grande desenvolvedor da década.

## Conclusão

Posso ter escrito um monte de baboseiras que você deve ter lido e falado super mal, mas eu coloquei aqui, eu não tive medo. Eu poderia ter guardado esses sentimentos e pensamentos só para mim. Mas uma das coisas que eu mais amo da comunidade Dev e que não vejo em nenhuma outra área é a disposição que todos nós temos de ajudar uns aos outros. Eu poderia citar aqui diversos programadores que me ensinaram muito e que me fizeram ser o que sou hoje. E você que está lendo, pode ser mais um a me ajudar, a ajudar os outros e ajudar a si mesmo.

# 3 meses de globo.com

## Um pouco do que passei desde o primeiro contato, processo seletivo, contratação, minha mudança e o dia-a-dia.

## Introdução

Como faço em todos os posts, a trilha sonora desse post é da cantora [Banks](https://open.spotify.com/album/5tMskcB2i9chjb4zcHx20n). Escrevo também enquanto desço minha querida serra de Petrópolis, num agradável frio de 8 graus.

Algumas pessoas já haviam me pedido para falar um pouco sobre como foi o processo seletivo e como é trabalhar na globo.com, eu demorei um pouco para escrever esse post, pois queria capturar o máximo de coisas possíveis e também esperar um tempo suficiente para que eu pudesse analisar tudo.

Pretendo fazer um post simples, com as minhas experiências, mas isso não quer dizer que tudo que aconteceu comigo, acontece com quem é contratado ou passa pelo processo seletivo, são só as coisas que ocorreram comigo.

P.S.: também esperei os 3 meses de experiência antes de escrever, vai que me demitem né? Meu time me zoava todo dia sobre isso xD

## Contatos

Eu trabalhava no [Queremos](https://www.queremos.com.br/), um lugar muito legal, com várias pessoas legais e uma ideia incrível. A empresa fica no Humaitá, bairro da zona sul do Rio de Janeiro. Como eu sou de uma cidade serrana, bastante afastada, não tinha muitos contatos, mas quando desci para trabalhar no Queremos, acabei conhecendo muitas pessoas e uma delas foi o [Davidson Fellipe](https://twitter.com/davidsonFellipe).

Conversa vai, conversa vem, ele perguntou se eu um dia não teria interesse em trabalhar na globo.com. Na hora eu fiquei receoso, porque a gente ouve sempre que os caras que trabalham na globo.com são todos gênios (e são mesmo!) e eu, um pobre iniciante nesse ramo de desenvolvimento, achava que nunca iria passar, mas falei que tinha interesse sim.

Não deu muito tempo e entraram em contato comigo, perguntando se eu teria interesse de participar do processo seletivo. Eu fiquei meio bolado e fui conversar com o Davidson sobre, ele falou para eu tentar, que não custava nada e ele acreditava bastante em mim. Como eu realmente não tinha nada a perder e queria saber meu nível, aceitei passar pelo teste.

## Processo Seletivo - Parte 1

Ao aceitar participar, recebi um projeto com várias coisinhas para fazer e que eu deveria entregá-lo no tempo que eu desejasse. Me pegaram numa semana complicada, onde estava cheio de coisas para fazer, mandei um email avisando sobre isso e foram totalmente tranquilos quanto a isso, falando para eu não me preocupar. Mas como sou um cara desesperado e preocupado por natureza (um defeito que venho tentando corrigir), peguei uma madrugada desesperada e fiz tudo. Não era nada super difícil não, mas possuía alguns detalhes interessantes e podia fazer com qualquer linguagem e stack. Eu escolhi fazer em Javascript, usei uma stack que já estava bastante habituado a usar, que é o meu [Fast](https://github.com/willianjusten/fast).

Passou por volta de 1 semana até eu receber uma resposta, que no caso foi positiva. Disseram que gostaram do que eu fiz e tinham interesse em fazer um entrevista agora por Skype e marcaram um dia comigo.

## Processo Seletivo - Parte 2

Do dia que marcamos até o dia da entrevista no Skype, eu estava uma pilha de nervos, como sempre preocupado achando que não iria passar, me rebaixando. Dessa vez, foi a hora de conversar com um outro amigo que já estava na globo.com também, mas que já havia trabalhado no Queremos comigo, [Rodrigo Machado](https://twitter.com/rcmachado). Ele falou que eu podia ir para o apartamento dele ficar lá para a entrevista, para assim não me atrasar para o trabalho depois, eu ainda descia a serra de Petrópolis todos os dias.

Na hora que iria ter o teste, não teve! Isso mesmo que você leu, eu fiquei desesperado para caramba, nem tinha dormido direito e aí o teste precisou ser cancelado devido a alguns problemas. Ele foi então remarcado para uma outra data. Foi desse dia que eu resolvi que deveria parar de ficar tão preocupado/desesperado. Eu já trabalhava num lugar legal, não haveria problema se eu não passasse também, era só um teste.

Quando finalmente chegou o dia do teste no skype, alguns programadores do TechTudo conversaram comigo, perguntaram um pouco do meu histórico, projetos que eu tinha orgulho de ter participado, coisas interessantes que eu gostaria de contar, etc. Curiosamente eu estava bem relaxado, a conversa fluiu super bem e falei para caramba (sim, eu falo demais!!!).

Após a conversa, falaram que iriam fazer alguns testes comigo. Os 2 primeiros testes eram para escrever pequenos programinhas para resolver alguns problemas, resolvi fazer um em javascript e o outro em python. O primeiro problema eu demorei até começar, dei uma travada bonita, aquela preocupação de fazer algo com geral vendo do outro lado. O segundo já foi bem mais tranquilo e saiu mais fácil. Depois destes 2 problemas, me passaram acesso a um web app, para eu fazer a correção de pequenos trechos de código em Javascript. Não eram erros de sintaxe e sim erros de performance, como manipulação de DOM dentro de um loop, entre outros problemas graves. Essa parte já foi bem mais simples para mim, já que gosto bastante de performance e já havia visto aquelas falhas de performance, inclusive em códigos mais velhos meus.

Depois de tudo feito, falaram que gostaram bastante, mas que iriam analisar um pouco mais e entrariam em contato para me avisar se eu passei e a próxima etapa.

## Processo Seletivo - Parte 3

Passado uma semana, recebi um email avisando que haviam gostado bastante do meu teste, que eu havia passado para a próxima etapa e que ela consistia em passar um dia na globo.com.

Eu já havia sido avisado que teria essa etapa, mas na hora que recebi o aviso real, fiquei mega feliz, conhecer uma empresa bastante grande e que todos falam! Marcamos um dia e, é claro, fiquei super ansioso até o dia chegar. Mal dormi naquele dia, acordei extremamente cedo para descer a serra e não ter problemas de atraso. Cheguei lá com quase 2 horas de antecedência e nesse meio tempo acabei escrevendo um post, de curiosidade fui ver que horas eu comecei a escrever o post:

"Como ter Domínio Proprio no Github Pages" date: 2015-01-29 04:16:36. Sim, eu praticamente não dormi!!

Após a espera, entrei e fui apresentado a algumas pessoas e guiado por dentro da globo.com. Para quem vinha de uma startup com umas 8-9 pessoas, foi um grande baque, tem muita gente lá dentro! Vários times de diferentes áreas, isso porque eu só tinha visto um andar, depois fui apresentado ao andar inferior, com mais um monte de gente também.

Pela manhã tive uma conversa com um dos times de Esportes, basicamente a mesma coisa que tive no Skype, só que agora presencial. Finalizando a conversa, me passaram um probleminha para resolver. Foi uma das piores soluções que eu fiz na vida, tanto que nem finalizei direito, fiquei bastante nervoso e frustrado... Me pediram para explicar o código e enquanto eu explicava, fui tendo outras ideias e fui explicando o motivo de ter pensando em outras coisas e acho que isso que me ajudou, porque eles viram que eu até que sabia, mas estava nervoso.

Na parte da tarde conversei com o time do TechTudo, falei um pouco sobre o problema que eu estava resolvendo e também sobre outras coisas que eles iam me perguntando. A conversa foi super legal e com isso adquiri até contato com o [Igor Canedo](https://twitter.com/Caned0) e simpatia com o resto do time. Depois dessa conversa, fui ter uma conversa um pouco mais técnica com o Gerente Técnico, falamos sobre muitas coisas e ele também me propôs 2 problemas mais teóricos para eu dizer como faria a arquitetura do sistema para comportar esses 2 problemas, indo de banco de dados, até número de requisições a uma API. Não foi totalmente técnico, mas também não foi tão fácil, mas consegui expor vários pontos que ele gostou.

Já no finalzinho da tarde, conversei com o pessoal do RH, sobre benefícios da empresa e outras coisas mais burocráticas e então dei tchau para todos, com pelo menos a sensação de dever cumprido. Dali era só esperar a resposta, mas que já nem era a mais importante das coisas, já que eu estava feliz de ter participado de tudo.

## Contratação

Deu uns 2-3 dias e me ligaram falando que eu havia passado! Fiquei mega feliz, mas sabia que agora seriam várias mudanças. Disse que aceitava sim, mas que primeiramente precisava me desvincular da empresa em que trabalhava e conversar com eles, para não causar nenhum problema para eles. Conversei com o pessoal da empresa, alguns ficaram chateados, mas como todos são legais, me desejaram boa sorte nesse meu novo rumo.

## Conhecendo meu novo time

Ao chegar na globo, já para trabalhar. Fui apresentado ao meu novo time, que seria o Time 1 de Esportes, um dos responsáveis pelo globoesporte. Acho que eu não poderia ter caído num melhor time (puxando saco =p). Meu time é relativamente pequeno e é composto por:

[Kenji Yamamoto](https://twitter.com/kenjiyamamoto), não tem aquela frase: "você pode ser bom, mas sempre vai ter um asiático melhor do que você", então...O cara é sinistro em milhares de coisas, brincamos até que ele é o pau para toda obra do time. Sempre com uma calma nipônica e um conhecimento tanto de linguagem quanto de produto (por já estar há quase 10 anos na globo.com), sempre me ajuda nas minhas dúvidas, por mais bobas que elas sejam.

[Rômulo Jales](https://twitter.com/romulojales), é um cabra da peste arretado, lá de Pernambuco, é o cara do Backend. Parece que respira programação e sempre te dá uma solução para um problema. Profundo conhecedor de Python e também um grande conselheiro e amigo. Não tem nem muito tempo que o conheço, mas já me acolhe e me ajuda como se nos conhecessemos há anos.

[André Cataldo](https://twitter.com/andrecataldo), o SM (Scrum Master) do time. Sempre preocupado com o time e o andamento dos projetos, atento para nos ajudar e nos atender em nossos problemas. Sabe como ser justo e comemorar uma conquista. Também um dos caras mais experientes da globo.com, com quase 14 anos só lá dentro, sendo que a globo.com fez 15 anos esse ano!

## Dia-a-dia no globoesporte

No globoesporte trabalhamos com uma variação do método ágil chamado [Scrum](http://www.desenvolvimentoagil.com.br/scrum/). Com isso temos nossos Plannings Semanais e Dailys, usando uma board para anotar nossas histórias, sempre na base (To Do, Doing, Done), com algumas coisinhas a mais, que variam de time para time.

Existem projetos só para o produto (esportes ou jornalismo ou entretenimento), mas também existem os projetos integrados a todas as áreas, em geral, nesse tipo de projeto, são feitas várias reuniões para entender as necessidades de cada um, para que assim a construção seja adequada a todos.

Dentro da globo, os projetos, em sua maioria, são feitos em Python/Django, mas também já vi coisas em Ruby, outras em Go e algumas áreas mais legadas usando Java. A preocupação é sempre em solucionar o problema de forma eficiente e útil para todos, então as linguagens e frameworks utilizados não são fixos, podendo variar bastante.

Prezamos muito pela comunicação, tanto dentro do time como em outras áreas também. Como o Rômulo Jales diz: "Não existe desenvolvimento sem comunicação".

A empresa é bastante legal e as pessoas são ótimas, semana passada eu pude participar do meu primeiro HackDay na globo e foi super divertido! Meu time desenvolveu um bolão bolado, utilizando a API do Tempo Real do globoesporte, para dar os resultados durante a Copa América.

Dava para ver na galera a animação e empolgação nos projetos, pelo que falaram foi um recorde de pessoas participantes e de projetos desenvolvidos, muitos deles eu queria ter participado, de tão legais que eram.

## Mudança e Conclusão

Bom, como disse lá no início, apesar de eu ser do Estado do Rio de Janeiro, sou de uma cidade serrana, chamada Petrópolis, que é cerca de 2:30 da globo.com (isso já contando o trânsito que sempre existe). Esse processo de subida todo dia, gastando cerca de 5h era muito cansativo. Mesmo indo num ônibus confortável e "dormindo" boa parte do trajeto, eu precisava acordar bastante cedo. Foi aí que resolvi me mudar para mais próximo do trabalho, é claro que toda mudança é cansativa e estressante também, mas já venho notado os benefícios de fazer isso. Então, se você é de longe, não tenha medo de se mudar, a empresa é legal e tenho certeza que você será bem acolhido.

Se você tem interesse, não tenha medo de tentar e basta enviar seu currículo no [github da globo](https://talentos.globo.com/).

Uffa, escrevi demais já! Enfim, espero que tenha respondido a maioria das dúvidas que me fizeram nesse tempo e caso queiram comentar mais alguma coisa, só falar nos comentários.

# Como se tornar um desenvolvedor Front End

## Aprenda os caminhos para virar um desenvolvedor Front End de qualidade.

## Índice

* [Introdução](https://willianjusten.com.br/como-se-tornar-um-desenvolvedor-front-end/" \l "intro)
* [O que faz um desenvolvedor Front End?](https://willianjusten.com.br/como-se-tornar-um-desenvolvedor-front-end/" \l "oqfaz)
* [Qual a importância do Front End?](https://willianjusten.com.br/como-se-tornar-um-desenvolvedor-front-end/" \l "import)
* [Mercado de Trabalho](https://willianjusten.com.br/como-se-tornar-um-desenvolvedor-front-end/" \l "mercado)
* [Níveis de conhecimento](https://willianjusten.com.br/como-se-tornar-um-desenvolvedor-front-end/" \l "niveis)
* [Eita lascou, como aprendo tudo isso?](https://willianjusten.com.br/como-se-tornar-um-desenvolvedor-front-end/" \l "eita)
* [Inglês](https://willianjusten.com.br/como-se-tornar-um-desenvolvedor-front-end/" \l "ingles)
* [Editores de Código](https://willianjusten.com.br/como-se-tornar-um-desenvolvedor-front-end/" \l "editores)
* [Básico do Terminal](https://willianjusten.com.br/como-se-tornar-um-desenvolvedor-front-end/" \l "terminal)
* [HTML](https://willianjusten.com.br/como-se-tornar-um-desenvolvedor-front-end/" \l "html)
* [CSS](https://willianjusten.com.br/como-se-tornar-um-desenvolvedor-front-end/" \l "css)
* [Javascript](https://willianjusten.com.br/como-se-tornar-um-desenvolvedor-front-end/" \l "js)
* [Controle de Versão](https://willianjusten.com.br/como-se-tornar-um-desenvolvedor-front-end/" \l "git)
* [Seo](https://willianjusten.com.br/como-se-tornar-um-desenvolvedor-front-end/" \l "seo)
* [Performance](https://willianjusten.com.br/como-se-tornar-um-desenvolvedor-front-end/" \l "perf)
* [Automação de Tarefas](https://willianjusten.com.br/como-se-tornar-um-desenvolvedor-front-end/" \l "task)
* [Mantenha-se Atualizado](https://willianjusten.com.br/como-se-tornar-um-desenvolvedor-front-end/" \l "update)
* [Dicas](https://willianjusten.com.br/como-se-tornar-um-desenvolvedor-front-end/" \l "dicas)

## Introdução

Bom, já faz um tempinho que eu não escrevo no blog, mas é por um bom motivo, estou me mudando de empresa e com isso eu estou precisando organizar algumas coisas da minha vida, inclusive uma mudança de cidade.

Algum tempo eu já pensava em fazer um post parecido e como estou precisando também deixar alguns guias para os futuros desenvolvedores que vão entrar no meu lugar, resolvi unir o útil ao agradável e criar esse post. Espero que ajude a quem está começando e para quem já é experiente, que possa acrescentar com outras dicas, dar alguns detalhes de sua experiência, seria bastante legal.

O post será separado em algumas partes, onde irei responder algumas dúvidas comuns, links para estudo, requisitos básicos, enfim, um compilado grandão de informações. A ideia é passar só um pouco de conteúdo para vocês se guiarem, é claro que não vou colocar todos os meus favoritos, senão seria um post gigantesco.

## O que faz um desenvolvedor Front End?

O desenvolvedor front end é o profissional responsável por projetar, construir e otimizar as interfaces de um projeto web.

### Responsabilidades

* Trabalhar junto a equipe de design para criar a melhor experiência possível
* Desenvolver código limpo e bem documentado
* Seguir os web standards
* Se preocupar com a performance no client-side
* Aplicar técnicas de SEO
* etc...

É claro que as responsabilidades mudam de acordo com a empresa em que você for trabalhar, mas essa é uma base comum na maioria dos lugares.

## Qual a importância do Front End?

É claro que a maioria já deve saber a necessidade desse profissional, mas vale ressaltar alguns detalhes:

* 80% do tempo de carregamento é gasto no frontend
* Cada vez mais as interfaces vem ficando complexas e detalhadas, fazendo ainda mais necessária a presença de um especialista
* O visual é a primeira coisa a ser vista, se ele estiver quebrado, o risco do usuário sair e nunca mais voltar é maior
* etc...

## Mercado de trabalho

Acho que uma imagem já fala mais que mil palavras né? A busca por desenvolvedores front end só está crescendo, isso permite salários maiores para quem é qualificado de verdade, boas empresas para se trabalhar, etc.

## Níveis de conhecimento

Olha, definir níveis de conhecimento, seja para qualquer profissão é algo bastante complicado, porque depende do que você vai se especializar. Eu meio que fiz uma separação mais ou menos de alguns assuntos que julgo necessários para cada nível de desenvolvedor. Terão alguns extras, que são assuntos interessantes, mas não fundamentais para aquele nível. Lembrando que os níveis mais altos compreendem tudo dos níveis anteriores.

Lembre-se: essas informações podem variar bastante de lugar para lugar.

### Iniciante Básico

* HTML (estrutura e tags)
* CSS (propriedades básicas)
* JS (pequenas noções, chamadas de scripts prontos)
* Básico de Photoshop (recortes, cores, fontes)

### Intermediário

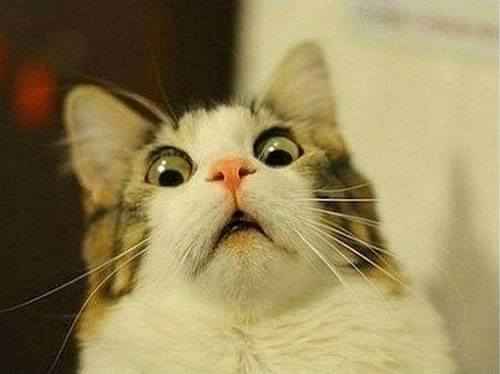
* HTML (Semântica e Acessibilidade)
* CSS (propriedades avançadas do CSS3, Design Responsivo, funcionamento de grids e noção de frameworks)
* JS (Entendimento da linguagem, noções de orientação a objeto)
* Ferramenta de Versionamento (Git)
* Noções de performance
* SEO (Extra)

### Avançado

* CSS (Entender pre-processadores e suas funcões)
* JS (Conhecimento forte na linguagem, entender algumas arquiteturas e Testes)
* NodeJS (Saber como funciona e alguns conhecimentos basicos para automatização e afins)
* Grunt/Gulp (saber como automatizar processos)
* Bons conhecimentos de performance
* Noções de UX/UI (Extra)

### Master

* CSS (conceitos avançados de arquitetura - BEM, Organic, Atomic)
* JS (Total domínio na linguagem, design patterns na palma da mão)
* NodeJS (saber como criar módulos, chamá-los tanto no front quanto no back)
* Ter sólida experiência em sistemas front como o MEAN Stack



## Eita lascou, como aprendo tudo isso?

Estava achando que ser desenvolvedor front end é moleza? Como disse o [Nicholas C. Zakas](https://twitter.com/slicknet)

Cada dia uma nova tecnologia é lançada e precisamos nos manter sempre atualizados, a questão é, como fazer tudo isso? Fiz um post há um tempo atrás falando sobre [técnicas de aprendizado](https://willianjusten.com.br/tecnicas-de-aprendizado/), que podem ajudar um pouco nessa longa caminhada.

Vou separar aqui as principais áreas que precisamos estudar, o porquê delas e alguns links para auxiliá-los no estudo.

## Inglês

Sim, uma das coisas mais importantes para se tornar um bom desenvolvedor é saber inglês. É claro que eu não estou falando que você precisa ser fluente como um americano legítimo de início, mas saber ler e assistir vídeos vai te ajudar muito, visto que a maioria do material vai estar em inglês, você querendo ou não. Não fique mendigando informações só na sua língua, saia do comodismo e vá aprender inglês, vai por mim, vai ser o melhor investimento que você pode fazer. Sobre como aprender inglês, existem vários métodos, desde os cursos tradicionais até programinhas para te ensinar. Dependendo do seu nível, eu recomendo fortemente um professor particular, pois ele poderá focar exatamente nas suas deficiências e você conseguirá evoluir bem rápido. Aqui vão alguns links legais falando do assunto:

* [Como aprender inglês](https://carlosschults.net/pt/como-aprender-ingles-parte-1/)
* [Será que realmente o profissional de TI deve aprender inglês?](http://www.adamsilva.com.br/carreira/sera-que-realmente-o-profissional-de-ti-deve-aprender-ingles/)

## Editores de código

Eles são basicamente o nosso braço direito, é importante escolher um editor que seja confortável para você e agilize o seu workflow. Não entre em guerrinhas de qual é melhor ou pior, instale, use e veja qual te agrada mais. Os iniciantes sempre caem na tentação de perguntar o melhor e bom, cada um vai puxar sardinha para o seu lado.

* [Sublime Text](http://www.sublimetext.com/)
* [Brackets](http://brackets.io/)
* [Atom](https://atom.io/)
* [Vim](http://www.vim.org/)
* Procure que tem muitos!

#### Sublime Text

* [12 Most-Wanted Sublime Text Tips And Tricks](http://www.hongkiat.com/blog/sublime-text-tips/)
* [7 Plugins do sublime que você deveria conhecer](https://tableless.com.br/7-plugins-sublime-text-que-voce-deveria-conhecer/)

#### Brackets

* [Brackets Course](http://css-snippets.com/brackets-course/)

#### Vim

* [12 Vim Tips](http://dalibornasevic.com/posts/43-12-vim-tips)
* [Powe Vim Usage](http://www.ukuug.org/events/linux2004/programme/paper-SMyers/Linux_2004_slides/vim_tips/)
* [Vim Adventures](http://vim-adventures.com/) - aprenda Vim jogando!!! Demais! <3
* [Vim Tutorials Screencasts and Resources](http://code.tutsplus.com/articles/25-vim-tutorials-screencasts-and-resources--net-14631)

## Básico do Terminal

Se você quer ser realmente um programador bom, vai ter que largar esse medo e colocar a mão na massa. O terminal se torna ferramenta FUNDAMENTAL para níveis do intermediário ao avançado. Seja para usar seu controle de versão, seja para utilizar ferramentas de automação e outras várias ferramentas.

* [Unix: Conceitos e Comandos Básicos](http://www.ime.usp.br/~ueda/ldoc/rb.html)
* [Unix Course](http://www.tutorialspoint.com/unix/)
* [Learn Shell](http://learnshell.org/)
* [Learning the Shell](http://linuxcommand.org/lc3_learning_the_shell.php)
* [Blog do Vitor Britto](http://vitorbritto.com.br/blog/) - Esse cara vira e mexe solta alguma dica muito legal sobre.

## HTML

A base de tudo, que vai governar a estrutura de todo o sistema. Qualquer desenvolvedor precisa saber ao menos o básico de html, inclusive a galera de backend. Já para galera de frontend, temos que nos atentar a mais detalhes, como: acessibilidade e semântica.

#### Básico

* [Codecademy HTML](https://www.codecademy.com/catalog/language/html-css)
* [Dash](https://dash.generalassemb.ly/)
* [Teamtreehouse](http://teamtreehouse.com/library/topic:learn-html/type:course)
* [CodeSchool](https://www.codeschool.com/paths/html-css)
* [30 Days to Learn Html and CSS](https://www.youtube.com/watch?v=yTHTo28hwTQ&list=PLgGbWId6zgaWZkPFI4Sc9QXDmmOWa1v5F)
* [Learn to code Html & CSS](https://learn.shayhowe.com/)
* [Apostila Web da Caelum](http://www.caelum.com.br/apostila-html-css-javascript/)
* [Dive into Html5 em Português](https://diveintohtml5.com.br/)

#### Semântica

* [Use Html5 sectioning elements](http://blog.teamtreehouse.com/use-html5-sectioning-elements)
* [SEO, Semântica e Html5](http://www.slideshare.net/fabioflat/seo-semantica-e-html5-12901651)
* [A semântica do HTML](https://speakerdeck.com/diegoeis/a-semantica-do-html)

### Acessibilidade

* [Slides de Acessibilidade na web](https://tableless.com.br/slides-para-devs-9-acessibilidade-na-web/)
* [WebAIM](http://webaim.org/intro/)

## CSS

No início era só um modo de deixar as páginas um pouquinhos mais bonitas, hoje já é uma tecnologia que faz milagres, permitindo animações, melhores interações e até mesmo substituir alguns tipos de imagens. Para o desenvolvedor frontend é uma das áreas mais importantes rivalizando com JS, dependendo de onde você deseja se especializar. Para os que preferem mais a parte de UX/UI e design, aqui será onde você deve se aprimorar cada vez mais.

Dica: CSS funciona muito na base de teste e acerte, veja o que cada propriedade faz, brinque com elas até se sentir confortável. Depois disso procure códigos alheios e tente entender o funcionamento e comece a aplicar em seus próprios códigos. Não há nada de errado olhar código alheio, open-source está aí para isso mesmo.

#### Básico

* [Site do Maujor](http://www.maujor.com/)
* [Aprenda Layout com CSS](http://pt-br.learnlayout.com/)
* [CSS Cross-country - Code Scholl](https://www.codeschool.com/courses/css-cross-country)

#### CSS3

* [Css3 for web Designers - Book](http://www.abookapart.com/products/css3-for-web-designers)
* [The Guide to CSS Animation](http://www.smashingmagazine.com/2011/09/14/the-guide-to-css-animation-principles-and-examples/)
* [Transition e Animation](https://tableless.com.br/transition-e-animation/)
* [MDN: CSS Animations](https://developer.mozilla.org/pt-BR/docs/CSS/Using_CSS_animations)

#### Frameworks / Grids / Responsive

* [Utilizando o bootstrap de forma consistente](https://www.felipefialho.com/blog/2014/utilizando-o-bootstrap-de-forma-consistente)
* [Desenvolvendo com Bootstrap 3](http://thiagonasc.com/desenvolvimento-web/desenvolvendo-com-bootstrap-3-um-framework-front-end-que-vale-a-pena)
* [Snippets para Bootstrap](http://bootsnipp.com/)
* [Foundation 5 para iniciantes - video](http://dev.rbtech.info/zurb-foundation-5-iniciantes-aula-1/)
* [Jeet](http://jeet.gs/) - aconselho fortemente
* [Skeleton](http://getskeleton.com/)
* [Responsive Design Workflow - Video](https://vimeo.com/45915667)
* [Responsive Patterns](http://bradfrost.github.io/this-is-responsive/patterns.html)

#### Pré-processadores

* [Less](http://lesscss.org/)
* [Sass](http://sass-lang.com/)
* [Stylus](http://learnboost.github.io/stylus/)
* [Por que usar pré-processadores?](https://speakerdeck.com/romulomachado/por-que-usar-pre-processadores-css)
* [Slide comparando os 3 pré-processadores](https://speakerdeck.com/willianjusten/pre-processadores-css)

#### Metodologias / Boas Práticas

* [How to structure a Sass project](http://thesassway.com/beginner/how-to-structure-a-sass-project)
* [Css Performance Organization](https://learn.shayhowe.com/advanced-html-css/performance-organization/)
* [OOCSS, SMACSS](http://www.slideshare.net/maxdesign/css-oocss-and-smacss)
* [Understanding BEM syntax](https://csswizardry.com/2013/01/mindbemding-getting-your-head-round-bem-syntax/)
* [Atomic Design](http://bradfrost.com/blog/post/atomic-web-design/)

#### Inpirações / Tutoriais

* [Codrops](http://tympanus.net/codrops/) - só ame <3
* [CSS-Tricks](http://css-tricks.com/)
* [Codepen](http://codepen.io/)
* [24 ways](http://24ways.org/topics/code/)

## Javascript

Basicamente a única real linguagem no client side e que agora já se extende para o server side também. Essa linguagem tem crescido muito e hoje permite fazer praticamente tudo, seja na web e até fora dela. Se você quer ser considerado um desenvolvedor front end completo é bom que tenha essa linguagem como sua melhor amiga. Como um amigo diz:

Desenvolvedor Front End que não sabe javascript é recortador de layout. - Suissa

Tome bastante cuidado ao querer tomar atalhos, como utilizar JQuery para tudo ou já tentar ir direto para Angular, que é modinha.

No início é comum utilizarmos plugins prontos para realizar ações específicas (sliders, lightbox), mas com o tempo vai se fazendo necessário criar soluções únicas e para isso que precisamos entender bem a linguagem.

O Vitor Britto também fez outro [incrível post](http://www.vitorbritto.com.br/blog/the-book-is-on-the-table/) falando sobre livros de JS.

* [Codecademy - JS](https://www.codecademy.com/catalog/language/javascript)
* [MDN - JS](https://developer.mozilla.org/en-US/Learn/JavaScript)
* [How to Learn JS Properly](http://javascriptissexy.com/how-to-learn-javascript-properly/)

## Controle de Versão

Quantas vezes você já se pegou fazendo 1000 backups só para não perder certas versões dos arquivos? E se isso pudesse ser feito de uma forma mais fácil e ainda permitisse muito mais? Para isso temos ferramentas de controle de versão, sendo o [Git](http://git-scm.com/) e o [SVN](https://tortoisesvn.net/) os mais famosos. Se você quiser brincar com o [github](http://github.com/), o git será a pedida.

A maioria das empresas utiliza um desses dois para poder organizar melhor seu código, além de poder versionar, o git permite um controle dos colaboradores, controle de flow, separação em branches e muito mais.

#### Básico

* [Meu curso de Git e Github para iniciantes](https://www.udemy.com/git-e-github-para-iniciantes/)
* [Try Git - Code School](https://try.github.io/levels/1/challenges/1)
* [Guia prático](http://rogerdudler.github.io/git-guide/index.pt_BR.html)
* [Introdução ao git e github - Vídeo](https://www.youtube.com/watch?v=yNwh0S0S0bU)

#### Intermediário/Avançado

* [A web-focused git workflow](http://joemaller.com/990/a-web-focused-git-workflow/)
* [Deploy web applications easily with svn](http://taylorjasko.com/deploy-web-applications-easily-with-svn/)

## SEO

Search Engine Optimization ou SEO é um conjunto de estratégias e métodos para melhorar o posicionamento de páginas da internet nos mecanismos de buscas, como o Google e Bing.

E qual a importância disso? Estar bem posicionado nos mecanismos de busca lhe permite ganhar mais acessos e com isso divulgar mais o seu produto.

Nem todo desenvolvedor Front End precisa ser o mestre de tudo em SEO, podemos deixar isso para especialistas, mas algumas informações básicas é importante também.

* [Agência Mestre - MestreSEO](https://www.agenciamestre.com/)
* [Como melhorar meu SEO?](https://willianjusten.com.br/como-melhorar-meu-seo/)
* [Dicas de SEO para Front-end](https://tableless.com.br/dicas-de-seo-para-front-end/)

## Performance

Como disse no início do artigo, uma das responsabilidades do desenvolvedor front end é cuidar da performance da aplicação, visto que 80% do carregamento está envolvido diretamente com o client side. Sendo assim, se você quiser se destacar entre os desenvolvedores, foque bastante em performance, os resultados são muito satisfatórios e recompensadores.

O [Davidson Fellipe](https://twitter.com/davidsonFellipe) fez uma awesome só disso e bom, ela está tão completa, mas tão completa, que basta o link dela, para você achar todo o resto. [Segue aqui esse ótimo compilado](https://github.com/davidsonfellipe/awesome-wpo).

## Automação de Tarefas

Durante o percurso de frontend, principalmente na parte de performance você irá ver que muitas tarefas se tornam bastante repetitivas, como compilar o css, minificar javascript e etc. Para facilitar essas tarefas, temos essas ferramentas de automação. Existem várias, mas dentre as puramente Javascript e que se tem mais utilizado na área de Front End, temos o [Grunt](https://gruntjs.com/) e o [Gulp](http://gulpjs.com/).

* [State of Grunt](https://cowboy.github.io/state-of-grunt-fe-summit-2014-talk)
* [Gulp - The vision, history and future of the project](https://medium.com/@contrahacks/gulp-3828e8126466)
* [Grunt vs Gulp - Beyond Numbers](https://jaysoo.ca/2014/01/27/gruntjs-vs-gulpjs/)
* [Building with Gulp](https://www.smashingmagazine.com/2014/06/11/building-with-gulp/)
* [Slides of Gulp](https://slides.com/contra/gulp)
* [Gulp, Grunt, Whatever](https://ponyfoo.com/articles/gulp-grunt-whatever)
* [Gulp: O novo automatizador](https://tableless.com.br/gulp-o-novo-automatizador/)
* [Bye bye Grunt.js, hello Gulp.js!](https://blog.caelum.com.br/bye-bye-grunt-js-hello-gulp-js/)

## Mantenha-se Atualizado

Como já deve ter visto, a área de Front End é uma das que mais se movimenta. Do dia para a noite podem nascer 20 ferramentas novas para você conhecer e aprender. Para você não perder nada de novo que está acontecendo, aconselho muito a seguir as pessoas certas, assinar weeklys e seguir blogs da área. Vou colocar aqui só alguns, porque só dessa parte temos muita coisa!

#### Pessoas para seguir

* [Addy Osmani](https://twitter.com/addyosmani)
* [Ben Alman](https://twitter.com/cowboy)
* [Brad Frost](https://twitter.com/brad_frost)
* [Chris Coyer](https://twitter.com/chriscoyier)
* [John Resig](https://twitter.com/jeresig)
* [Sara Soueidan](https://twitter.com/SaraSoueidan)

#### Weeklys

* [Web Design Weekly](http://web-design-weekly.com/)
* [Frontend Focus](https://frontendfoc.us/)
* [Css Weekly](http://css-weekly.com/)
* [Javascript Weekly](http://javascriptweekly.com/)

#### Blogs

* [Codrops](http://tympanus.net/codrops/)
* [EchoJS](http://www.echojs.com/)
* [Google’s Web Fundamentals Program](https://developers.google.com/web)
* [Css-tricks](http://css-tricks.com/)
* [Smashing Magazine](http://www.smashingmagazine.com/)
* [Da2k Blog](http://blog.da2k.com.br/) - PtBr
* [Leandro Oriente Blog](http://leandrooriente.com/) - PtBr
* [Felipe Fialho Blog](https://www.felipefialho.com/blog/) - PtBr

#### Eventos

Nada melhor que um bom networking para te ajudar a conhecer mais coisas. E nada melhor que os eventos para isso, além de palestras ótimas, sempre acaba conhecendo alguém para te ajudar no futuro.

Falando nisso, vai rolar o GDG DevFest em BH, se quiser que eu vá, [vote aqui](http://call4paperz.com/events/gdg-devfest-extended-bh-2015--2)

## Dicas

Para quem está começando é sempre um pouquinho complicado saber por onde começar, como fazer e o que fazer. Aqui vão algumas dicas que o [Vitor Britto](http://vitorbritto.com.br/) deu uma vez na Timeline dele e que eu concordo 110%.

* Não adianta querer saber tudo em um curto espaço de tempo. Mas saiba que você pode aprender muito com o passar do tempo e se estiver disposto a aprender.
* Vai com calma, estude no seu tempo, aprecie um tempo de estudo, mas também seu lazer. O mundo não vai acabar se você não conseguir estudar tudo que quer.
* Se afobar para querer o aprendizado a todo custo é uma péssima ideia. Aprenda na medida do possível, respeitando o seu limite e praticando sempre que puder.
* Pegue as coisas que você gosta, separe, estude a base daquilo e faça pequenos experimentos, porque a prática leva a perfeição. E se puder, passe adiante o que aprendeu, que será ainda mais recompensado. -Na programação: a leitura vale prata e a prática vale ouro.

## Conclusão

É claro que existem outros links, outras coisas para aprender e outros métodos, nossa área é bastante grande. Mas espero que isso já lhe auxilie um pouco nessa longa jornada que é se tornar um desenvolvedor front end.

Para os mais experientes, faltou alguma coisa muito importante que esqueci de falar? Explica aí embaixo nos comentários ou manda um Pull Request, críticas são sempre boas, principalmente para ajudar a quem está começando.

## Fontes

* [O Papel do Desenvolvedor Front End](http://pt.slideshare.net/raelmax/o-papel-do-desenvolvedor-front-end)
* [Guia Front-end das Galáxias](http://pt.slideshare.net/davidsonfellipe/guia-do-front-end-das-galaxias)
* [Como me tornar um desenvolvedor front end](http://leandrooriente.com/como-me-tornar-um-desenvolvedor-front-end/)
* [Como se tornar um dev front-end](https://tableless.com.br/tornar-dev-front-end/)

# Devo fazer faculdade?

## Essa dúvida sempre acontece, principalmente para quem trabalha com programação. Vamos conversar mais sobre o assunto.

## Introdução

Ontem conversando com o [Fernando Daciuk](http://blog.da2k.com.br/), começamos a falar de um assunto que eu julgo bastante importante, ainda mais para nossa área. É engraçado como esse assunto pode diferir de pessoa para pessoa, conheço pessoas que dizem que uma boa universidade é o mais importante de tudo e outras que acham uma bobeira. Por esse ser um assunto tão dividido, vou contar um pouco da minha história e também um pouco da minha opinião sobre o assunto e também levantar algumas dúvidas comuns em relação a isso.

## Histórico

Como disse no meu [primeiro post](https://willianjusten.com.br/making-of-parte-1/), meu primeiro curso foi bem diferente do que eu faço atualmente, foi Química Industrial, mas vou falar um pouco dele, porque foi fundamental para mim. Eu tinha 17 anos quando passei para a faculdade, resolvi me mudar de cidade (Petrópolis -> Niterói) só para poder fazer o curso. Foi uma experiência completamente nova, eu nunca tinha saído de casa, ainda mais tão novo e eu estudava em lugares bem pequenos, quando vi a imensidão da UFF, com seus inúmeros Campus, foi algo bastante diferente para mim.

Sempre fui um cara que passava em tudo com grande facilidade, fiz o vestibular praticamente sem me preparar como meus amigos e mesmo assim passei em várias faculdades e principalmente a que eu queria, que era a UFF. E esse foi o meu primeiro baque na faculdade. Meu primeiro período teve uma das matérias conhecidas como a mais carrasca de todas, que é Cálculo I e é claro que eu levei bomba na primeira prova, mas não foi coisa pouca não, eu tirei 0,3 de 10,0. Aquilo feriu demais o meu orgulho e a partir dali comecei a criar uma pequena rotina de estudos e fui um dos poucos da turma que conseguiu passar. Isso me ajudou a ver a importância dos estudos, dedicação e foco, coisa que eu nem dava bola antes.

Durante o curso de Química tive a oportunidade de conhecer professores muito bons, que se esforçavam para que o aluno saísse da turma não só com o conhecimento da matéria, mas com o conhecimento da vida, para buscar mais conhecimento e crescer ainda mais. Mas também tive professores péssimos, que só estavam ali por estarem obrigados por algum motivo oculto. Se você está começando a faculdade ou desistiu de alguma faculdade por causa disso, saiba que isso é muito comum, em todo lugar você é capaz de encontrar os bons e os péssimos profissionais, cabe a você analisar e escolher o seu lado. Essa foi outra grande lição que aprendi na faculdade, como analisar pessoas e ambientes.

Lá ná Química mesmo eu comecei a me aproximar mais da computação, já que eu trabalhei no laboratório de Modelagem Molecular, onde fazíamos cálculos matemáticos para tentar buscar solucionar ou descobrir novas informações acerca de compostos químicos. E foi aí que eu comecei a me interessar mais por computação e resolvi fazer TI no IST em Petrópolis.

Durante o meu curso de TI, eu já estava mais experiente, sabia sobre o funcionamento de uma faculdade, quais disciplinas necessitariam de mais foco, quais professores eu deveria forçar mais ou menos e o principal, com quais pessoas eu deveria me relacionar. Foi um curso relativamente mais fácil, devido a essa experiência de uma faculdade passada, mas muito proveitoso. Consegui passar um pouco da minha experiência para alguns dos colegas da turma, que hoje posso dizer que são bons amigos e que também aprendi muito com eles.

Para mim, as faculdades serviram para meu amadurecimento pessoal e também profissional. Me permitiram pensar mais distante e não só uma visão de 1-2 meses. Fizeram com que eu fosse mais analítico a problemas e soluções, nunca mais resolvo um problema grande sem fazer uma análise completa dele e separá-lo por etapas e detalhes importantes. Também me ajudaram a me relacionar melhor com as pessoas e o mais importante, as pessoas certas. Tive milhares de trabalhos que necessitavam de longos textos a serem escritos e com isso, acabei tomando bastante gosto por escrever e por uma gramática correta. Enfim, eu tenho certeza que não seria o mesmo sem minhas faculdades e tenho muito a agradecer por isso.

## Faculdades e "faculdades"

O primeiro ponto a se entender é que as faculdades não vão te preparar para tudo na vida, seja para o mercado de trabalho, seja para a área científica.

Quem faz a faculdade são os alunos!

* Autor desconhecido

Essa frase faz todo o sentido do mundo, se você quer fazer uma faculdade só para ganhar um diploma, já desiste agora, pois muito provavelmente você não irá conseguir e se conseguir, não terá valido de nada.

Faculdade não é pra te dar conhecimento técnico. Pra isso vc tem cursos, livros, tutoriais na internet e afins. Faculdade não acompanha mercado. Se for assim, faculdade vai mudar de 2 em 2 semanas.

A faculdade foi feita pra te mostrar a base daquilo que vc estuda, te fazer evoluir como pessoa e profissional. Além disso, faculdade está longe de se resumir a ir a aula. É todo o ambiente ao seu redor.

A faculdade irá servir para te mostrar situações e oportunidades para que você adquira experiência e busque mais conhecimento. E nesse caso temos dois tipos bem diferentes: as faculdades mais tradicionais (Federais e Estaduais) e as vendedoras de diploma. Se você está numa faculdade que promete diploma rápido e fácil em dois anos, não vá esperando algo muito bom por aí. E isso é o que geralmente acontece com a maioria das pessoas na nossa área, fazem uma faculdade qualquer, tem uma péssima experiência e aí acaba falando mal.

## Já sei programar, para que vou fazer isso?

Essa é uma das frases mais equivocadas que eu vejo por aí. A faculdade não está lá para te ensinar necessariamente a programar, você terá diversos tipos de disciplinas, algumas aplicadas a programação e outras até bastante distantes a primeira vista. Muitos dos melhores programadores já sabiam programar antes de entrar na faculdade, isso acontece porque temos cada vez mais contato com computadores e tutoriais na internet e isso é excelente, porque já mostra a força de vontade de aprender.

Na faculdade aprenderemos conceitos e paradigmas que são mais difíceis de serem vistos no cotidiano, como estudo de grafos, algoritmos eficazes, matemática discreta aplicada, dentre outros vários assuntos, dependendo da sua instituição de ensino. E isso irá te ajudar a ter uma cabeça mais aberta quando novos assuntos aparecerem.

## Famosos nem fizeram faculdade...

Você não é todo mundo. - Sua mãe

Assim como você pode acabar não fazendo faculdade e se tornar um cara de sucesso como Bill Gates, Steve Jobs, Mark Zuckerberg...Você também pode fracassar feio e não conquistar nada na sua vida. Não devemos comparar casos a parte com a nossa realidade, não que fazendo faculdade você será bem sucedido, isso será conquistado com o seu esforço.

Existe uma [palestra muito famosa do Jobs](https://www.youtube.com/watch?v=66f2yP7ehDs), em que ele fala um pouquinho da sua experiência e como ele deixou a faculdade para estudar caligrafia, vale assistir.

## Já sou velho e trabalho com isso tem 20 anos

Nunca se é velho para qualquer coisa =)

Realmente se você já trabalha há tanto tempo na área, uma faculdade de programação pode não lhe trazer tantos benefícios. Mas por que não um curso diferente? Talvez administração, se quiser focar talvez em criar uma startup ou até mesmo Design, sempre vai ter alguma coisa legal para se aprender e aprimorar.

## Ok, me convenceu, mas qual eu faço?

O primeiro de tudo é ter uma noção do que deseja seguir para frente, se quer algo mais focado em infra ou desenvolvimento, se quer algo bastante acadêmico ou não.

Tendo um pouco disso em mente, leia as ementas dos cursos, existem milhares de cursos e cada um tem suas peculiaridades.

## Mas não se esqueça...

Por mais que você esteja na faculdade, todo o seu sucesso irá depender de você, então siga algumas dicas importantes:

* Leia bastante: não se prenda ao dado nas aulas, busque conhecer mais sobre aquele assunto.
* Discuta: não seja só mais um na sala que escuta e acredita em tudo que o professor diz, discuta opiniões, mostre seus pontos e espere receber mais conhecimento.
* Estude Certo: estudar 16h por dia não te fará mais inteligente e nem estudar 1 dia antes da prova. Estude um pouco sempre, quando chegar uma prova ou trabalho, você já saberá tudo.
* Tenha bons grupos: escolha pessoas que compartilhem do seu pensamento e não carregue ninguém, divida as tarefas e aprenda bastante com todos.
* Viva o ambiente: não entre já querendo sair, conviva lá, retire o máximo que puder do lugar.

## Busquem conhecimento

Durante a criação do post, acabei achando alguns vídeos legais e excelentes posts sobre ~~mil vezes melhores que o meu~~.

* [O que aprendi em Harvard - Jorge Paulo Lemann](https://www.youtube.com/watch?v=rhaeYj7cln0) - ASSISTA
* [Faculdade de informática. fazer ou não fazer? - Leandro Oriente](http://leandrooriente.com/faculdade-de-informatica-fazer-ou-nao-fazer/)
* [Devo fazer faculdade? - Fabio Akita](http://www.akitaonrails.com/2009/04/17/off-topic-devo-fazer-faculdade" \l ".VNYHTFPF88Y)

# Meu dia a dia como dev frontend

## Um pouco do meu dia a dia como dev frontend. Minha máquina, meus programas, arquivos, configurações.

Como alguns devem saber o [Luiz Felipe Tartarotti Fialho](https://github.com/felipefialho) criou um [repositório](https://github.com/felipefialho/1-post-por-dia) no github bastante legal, para que pessoas possam pedir alguns temas.

E teve esse pedido do [Matheus Moraes](https://github.com/matheusmmo):

Galera, sempre tive muita curiosidade e imagino que outros também tenham, de saber como é o dia a dia de cada dev. Ou seja, como é o seu workflow, qual a estrutura de pastas que o cara usa quando vai fazer uma aplicação x. Qual as ferramentas que usa todos os dias, qual frameworks que gostam de usar esporadicamente. E até saber como funciona os workflows das suas empresas. Um papo um pouco mais descontraído, sobre experiências. Eu que sou do interior de SP e não tenho muito contato direto com a galera que não trabalhe em agência sempre tenho curiosidade de saber essas coisas. Quem já trabalhou em agência sabe que o negócio é correria e na maioria das vezes não fazemos o que precisamos pra deixar um projeto legal. Enfim, é isso. Fica a dica ai pra quem quiser fazer ^^

O [Fernando Daciuk](https://github.com/fdaciuk) já fez o [post dele](http://blog.da2k.com.br/2015/01/15/como-ser-um-dev-frontend-usando-linux/) e agora farei o meu =)

## Meu cantinho

Eu uso um Macbook Air 13", tanto em casa, quanto no [Queremos!](https://www.queremos.com.br/), não ligo muito para guerrinha Apple x PC, mas dei preferência ao Air por ser bastante leve e fino, como eu ando bastante, é bem melhor carregar uma coisa levinha do que um trambolho gigante e pesado.

Utilizei durante muito tempo notebooks da Dell e meu último foi da Samsung (acho que ele me traumatizou)... Neles eu sempre utilizei Linux, comecei com o [Ubuntu](http://www.ubuntu.com/) e depois fui parar no lindo [Elementary](https://elementaryos.org/) (que inclusive é bastante similar ao Mac OS e facilitou a transição).

Praticamente não consigo trabalhar em uma tela só, então no trabalho eu uso um monitor de 23" FullHD da LG e em casa utilizo um de 25" da Samsung. Não sou altamente crítico com monitor não, contanto que tenha uma resolução e tela grandes, está ótimo.

Tenho um quadro em que faço algumas anotações e trabalho com meu [Personal Kanban](http://en.wikipedia.org/wiki/Kanban_board), que nada mais é que uma listinha de tarefas a fazer, sendo feitas e já feitas. Para quem não pode ter um quadro, existem vários apps que permitem fazer essa organização, o que eu mais aconselho é o [Trello](http://trello.com/), eu só tenho um quadro mesmo, porque eu acordo e já olho para ele, se fosse aplicativo, acabaria me esquecendo de olhar, sério, eu só não esqueço a cabeça porque está presa no corpo.

## Aplicativos

Como terminal padrão eu uso o [iterm 2](http://iterm2.com/) com umas [configurações](https://github.com/willianjusten/dotfiles) diferentes, para mudar um pouco as cores e também a forma de mostrar as informações.

Para codar, utilizo o [Sublime Text 3](http://www.sublimetext.com/3) com alguns [plugins fundamentais](https://github.com/willianjusten/sublime-preferences) para minha vida.

Meu browser principal é Google Chrome, mas também tenho utilizado bastante do Firefox Developer Edition, até que ele é bem interessante e tem algumas ferramentas bastante úteis.

#### Outras coisinhas que uso

* [Mou](http://25.io/mou/) - para escrever Markdown e ter um preview bonitinho.
* [Vagrant](https://www.vagrantup.com/) - para criar ambientes de desenvolvimento separados.
* [Dropbox](https://www.dropbox.com/) - meu salvador da pátria para guardar todas as minhas coisas.
* [Hipchat](https://www.hipchat.com/) - programa de chat que utilizo no trabalho, ele é integrado ao new relic e a cada deploy ou commit, ele envia uma mensagem para todo o grupo.
* [Spotify](https://www.spotify.com/br/) - sério, ele é muito importante para mim, sem ele, não trabalho direito.

## Workflow

Eu mudo bastante as ferramentas e tecnologias de acordo com os projetos, mas o que não falta nunca:

* [Gulp](http://gulpjs.com/) - meu automatizador de tarefas padrão
* [Stylus](http://learnboost.github.io/stylus/) - meu pre-processador css favorito

Eu tenho um boilerplate que uso bastante e adapto bastante também, que é o [Fast](https://github.com/willianjusten/Fast)

Em alguns projetos gosto de utilizar o MEAN (MongoDB, Express, AngularJS, NodeJS), também curto Python, então já me aventurei um pouco com o [Flask](http://flask.pocoo.org/), [Bottle](http://bottlepy.org/) e o [Django](https://www.djangoproject.com/). No Queremos! tem uma parte rodando o [Symfony](http://symfony.com/), que é um Framework PHP. Enfim, não tenho medo de aprender coisas novas, seja no Frontend, seja no Backend, o importante é se aventurar! \o/

# Técnicas de aprendizado

## Se você tem costume de procrastinar, tem dificuldade de aprender algo rápido. Aprenda técnicas que te permitem aprender melhor e aumentar sua produtividade.

Você já se pegou deixando uma tarefa para depois só porque estava cansado ou com preguiça de fazer naquele momento? Já ficou se punindo por não ter feito algo que deveria fazer? Ou então sentou para estudar e até a parede parecia mais interessante? Se você respondeu não para todas as perguntas, ~~some daqui seu ET~~, brincadeiras a parte, é comum do ser humano achar certas tarefas tediantes/desagradáveis e por isso perdemos nosso foco e deixamos de fazer nossas tarefas e isso se chama Procrastinação.

Procrastinação: é o ato de adiar tarefas, realizando outras, menos importantes ou mais prazerosas, deixando as mais importantes para depois.

Este que vos escreve tem um enorme problema de concentração e uma vontade absurda de procrastinar. Eu trabalho em outra cidade, então faço uma viagem um pouco desgastante, acabo chegando em casa cansado e tudo que eu quero é descansar e foi assim que eu demorei mais de 1 ano e meio falando que iria fazer um site/blog e não fiz. Na antiga empresa em que trabalhei, conheci um cara, que para mim é um dos programadores mais fodas que conheço, mas que se esconde, que é o [Guilherme Louro](https://github.com/guilouro) e esses dias ele postou sobre uma das técnicas que vou falar abaixo e bom, apesar de conhecer, eu resolvi dar uma chance e eis que criei meu blog!!

## Introdução

No intuito de auxiliar as pessoas a melhorarem e controlarem esse estado de vontade de procrastinar, alguns pesquisadores desenvolveram certos métodos de estudo e eu irei falar um pouco sobre alguns deles. A ideia é te dar o caminho das pedras aqui, você escolher um método que mais combina com você e partir para cima. A meta é: Deixe de procrastinar!

### Técnica do Pomodoro

A [Técnica do Pomodoro](http://pomodorotechnique.com/) foi criada nos anos 90 por um desenvolvedor chamado [Francesco Cirillo](http://cirilloconsulting.de/). Ele deu esse nome graças aos timers em formato de tomate que ele usou enquanto estudava na universidade. Essa é uma das metodologias mais simples e o conceito está em separar grandes tarefas em pequenas porções de tempo, os chamados "Pomodoros" e a cada pomodoro fazer uma breve pausa, para que o cérebro recupere um pouco das energias, relaxe e permita voltar a se concentrar melhor novamente. Isto permite melhorar a concentração e o foco com o tempo. Essa é considerada a técnica favorita dos programadores. Se eu não me engano o [Ney Simões](https://github.com/neysimoes) utiliza este método, se você utilizar esse método também, fala nos comentários, é sempre bom compartilhar experiências =)

#### Como funciona?

1. Escolha uma tarefa para ser completada
2. Inicie o Pomodoro (o tempo padrão é de 25 minutos)
3. Trabalhe somente nesta tarefa até que o tempo acabe
4. Descanse por um pequeno período de tempo (o tempo padrão é de 5 minutos)
5. Retorne ao Pomodoro
6. A cada 4 Pomodoros uma pausa maior deve ser feita (em geral são de 15-30 minutos)

#### Links para começar

* [Site oficial](http://pomodorotechnique.com/) - fala basicamente tudo sobre o método
* [Tirinha genial explicando o método](http://mel-meow.com/uma-longa-noite-aprendendo/) - a tirinha que o Guilherme compartilhou
* [Pomodoro One](https://itunes.apple.com/us/app/pomodoro-one/id907364780?mt=12) - Aplicativo para Mac com um timer
* [Pomotodo](https://pomotodo.com/app/) - App online que mapeia os pomodoros e tarefas executadas (utilizo ele)
* [Tomato Timer](http://tomato-timer.com/) - outro app online
* [Tomighty](http://www.tomighty.org/) - App para Mac/Windows

### Getting Things Done (GTD)

O GTD é uma invenção do David Allen, dono do site [gettingthingsdone.com](http://gettingthingsdone.com/). Esse método foi feito com o intuito de organizar o caos que a nossa cabeça faz as vezes. Ele tem como objetivo encorajar o usuário a criar metas no seu dia-a-dia e realizá-las. Essas metas podem ser desde arrumar o seu quarto até terminar uma feature importante de um sistema.

#### Como funciona?

1. Etapa de Coleta: use um caderno de anotações, um quadro de tarefas, qualquer coisa, simplesmente anote todas as tarefas que você precisa executar.
2. Etapa de Processamento: verifique as tarefas, veja quais são possíveis, quais não são e as elimine.
3. Etapa de Organização: organize as listas de tarefas e crie categorias para cada tipo de ação.
4. Etapa de Reflexão: Sempre revise constantemente as suas tarefas, limpe e atualize as listas, veja tarefas que para executar depois.
5. Etapa de Execução: simplesmente faça suas tarefas, utilize o sistema da melhor maneira possível.

#### Links para começar

* [Site oficial](http://gettingthingsdone.com/) - lá tem as etapas, links para livros e etc.
* [Produzindo.net](http://www.produzindo.net/gtd-getting-things-done-parte-1-introducao/) - Um blog falando mais a fundo sobre o GTD

### Seinfeld Secret Method (Don't Break the Chain)

Esse é um método simples de implementar e tem como objetivo marcar as coisas que são importantes para nós e permitindo assim, alcançar nossos objetivos. O conceito é simples, gaste um pouco do seu tempo fazendo uma tarefa, ao final do dia, marque um X no calendário. Se você deixar de fazer algum dia, você não pode marcar o X e com isso "quebra a corrente".

#### Como funciona?

1. Marque seus objetivos, não comece com mais que 3, você pode aumentar depois para 4, caso tenha certeza que irá conseguir.
2. Defina tarefas mínimas para cada objetivo, por exemplo, "correr 1km".
3. Defina suas regras e limites
4. Imprima um calendário e vá marcando o X a cada dia que fizer uma tarefa.

#### Links para começar

* [How Seinfeld's Productivity Secret Fixed My Procrastination Problem](http://lifehacker.com/5886128/how-seinfelds-productivity-secret-fixed-my-procrastination-problem) - um cara contanto sua experiência sobre o método
* [Writerstore.com](https://www.writersstore.com/dont-break-the-chain-jerry-seinfeld/) - um artigo falando um pouco mais do método
* [jamesclear.com](https://jamesclear.com/stop-procrastinating-seinfeld-strategy) - mais um artigo falando sobre o método

### Seu próprio método

As vezes você não quer seguir a risca nenhum desses métodos, mas quer parar de procrastinar, quer começar a aprender, a se concentrar melhor, então não desiste não cara, cria o seu método e parte para cima dele, existem várias histórias muito legais de pessoas que resolveram fazer isso, uma das que mais gosto é do [John Resig](https://github.com/jeresig) criador do JQuery e um dos fundadores da [Khan Academy](https://www.khanacademy.org/).

* [Write Code Every Day](http://ejohn.org/blog/write-code-every-day/) - John Resig diz como resolveu programar um pouco a cada dia.
* [How I finally learned to build stuff with Rails](https://medium.com/@mackenziechild/how-i-finally-learned-rails-95e9b832675b) - Mackenzie Child dá algumas dicas de como fez para aprender Rails
* [Jennifer Dewalt](http://blog.jenniferdewalt.com/) - essa programadora mostra a sua experiência de fazer 1 site por dia durante 180 dias!

### Conclusão

Se existe coisa pior que a procrastinação, eu não conheço. Hoje de manhã estava conversando com o [Fernando Daciuk](http://blog.da2k.com.br/) exatamente sobre procrastinação, é uma parada muito complicada, todo mundo já viveu ou vive um tempo em que fazer qualquer coisas fora do horário de expediente já é cansativo e desanimador. Espero que com esse texto você se anime mais, busque conhecimento, compartilhe o que aprendeu e se dedique mais. Quem fala aqui é um ex-químico que se apaixonou tanto pela área que desistiu de tudo só para começar a aprender a mexer com web. Boa sorte galera! E se você tem uma história legal, compartilha aí com a gente!!