

# DESENVOLVIMENTO DE SOFTWARE PARA WEB

Hermeson Richardson Pereira de Oliveira <sup>1</sup>

<sup>1</sup> Desenvolvimento de Software para Web – UNINORTE

[hermeson.richardison@gmail.com](mailto:hermeson.richardison@gmail.com)

**Abstract.** *The development software comes from a range of instructions and relatively denotes from a modern data structure to the simplest of programs, however, it should be noted that nowadays, we often need systems that are increasingly capable of giving us facilities in our day-to-day, be it a page of banks, payments, social networks, everything gives a software capable of keeping the necessary information for a good development. For each situation, the developer follows a journey in its mystical storyline as a different journey with each attempt and they grant us varied information in unique ways, as each developer, thinks differently, each idea, implementation, transformation, it all comes down to its real mission of passing knowledge to the end user.*

**Keywords:** *Development, Developer, Software, Information.*

**Resumo.** *Os softwares de desenvolvimento decorrem de uma gama de instruções e relativamente denotam desde uma moderna estrutura de dados até os mais simples dos programas, contudo, há de se notar que nos tempos atuais, necessitamos com frequência de sistemas cada vez mais capazes de nos atribuir facilidades em nosso dia a dia, seja uma página de bancos, pagamentos, redes sociais, tudo confere a um software capaz de guardar as informações necessárias para um bom desenvolvimento. Para cada situação, o desenvolvedor percorre uma jornada em seu enredo místico como uma viagem diferente a cada tentativa e nos concedem informação variada de formas únicas, pois cada desenvolvedor, pensa de uma forma diferente, cada ideia, implementação, transformação, tudo se resume a sua real missão de passar conhecimento ao usuário final.*

**Palavras-chave:** *Desenvolvimento, Desenvolvedor, Software, Informações.*

## Sumário

1.	Introdução .....	3
1.1	Objetivos .....	3
1.1.1	Objetivos Gerais.....	3
1.1.2	Objetivos Específicos.....	3
1.2	Justificativa .....	4
2.	Os conceitos das Web's .....	5
2.1	Diferenças entre Web 1.0, 2.0 e 3.0. ....	6
2.1.1	Web 1.0 .....	6
2.1.2	Web 2.0 .....	6
2.1.3	Web 3.0 .....	7
3.	Softwares de criação Web.....	8
4.	IDEs e Editores de Textos.....	8
4.1	Exemplos de IDE's .....	8
5.	Front-end Frameworks .....	11
6.	Web App Frameworks .....	12
6.1	Exemplos de Web App Frameworks .....	12
7.	Terminal de Linha de Commando .....	14
8.	Versionador de Código .....	14
9.	Web Deploy .....	15
10.	Outras Tecnologias .....	16
11.	Comunidades de Desenvolvimento Web .....	17
11.1	Exemplos de comunidades.....	17
12.	Conclusão.....	19
13.	Bibliografia .....	20

## 1. Introdução

O contexto deste artigo apresentado de forma sucinta, demonstra em equivalência os resultados que podem ser obtidos de uma tecnologia que está em abrangência no mercado visando cada vez mais negócios na área onde oportunidades não faltarão ao longo dos anos, para os que se prepararem ao que está acontecendo e o que ainda está por vir.

Desenvolver para a web deixou de ser apenas um trabalho, virou recurso de experiências e aprendizados onde disseminam-se em áreas distintas. De certa forma a consistência nas formulações de ideias e seus parâmetros para um desenvolvimento cabem não só aos programadores, mas também aos recursos que serão utilizados, uma mistura correta de linguagens e uma dosagem temperamental de criatividade, afinal de contas, do que adianta saber criar páginas na web, se elas não conseguem chamar a atenção dos usuários que nem mesmo quiseram adentrar a fundo no seu ambiente virtual.

Criações como facebook, youtube, instagram entre outros famosos softwares foram criados com a intenção de segurar o internauta em seus mundos virtuais, e após esta façanha, através de propagandas postas de formas estratégicas, lucrar com as merchandising de seus parceiros e patrocinadores. Até mesmo páginas da web que passam a ideia de aprendizagem intuitiva, fazem proveito de propagandas, pois são necessários recursos para pagar pela manutenção de uma página que está sendo hospedada em um servidor.

De tudo, nota-se que a gama de direções onde são levadas as oportunidades são contudo, infinitas, bastando assim o cliente dispor sua ideia de negócio e assim começar a lucrar com mais um recurso que pode agir de forma paralela aos aspirantes de compra na web.

### 1.1 Objetivos

A seguir serão apresentados os objetivos gerais e específicos do trabalho.

#### 1.1.1 Objetivos Gerais

Desenvolver um artigo de desenvolvimento de software web que permita exemplificar parâmetros de criação de especificações de softwares, permitindo o assim abranger o conteúdo requisitado.

#### 1.1.2 Objetivos Específicos

Dentre os objetivos específicos do trabalho, destacam-se:

- Demonstrar historicamente a evolução das Web's.

- Realizar a análise dos softwares destacados e seus requisitos, identificando as características essenciais para o desenvolvedor.
- Visualizar as estruturas básicas dos softwares por suas linguagens e pretensões.
- Exposição superficial de frameworks e suas linguagens de programação web.

## 1.2 Justificativa

Como afirma Bento ( 2013, p. 17)

**“O segredo da famosa web 2.0 é a capacidade de interação entre as pessoas e os serviços online. Mas, para que esta interação seja possível, é necessário que os sites sejam capazes de receber informações dos internautas e também de exibir conteúdo personalizados para cada um ou de mudar seu conteúdo automaticamente, sem que o desenvolvedor precise criar um novo HTML para isso. “**

Diante da afirmação, dispomos da idéia de que o futuro nos reserva tantas reformulações quanto paradigmas de como chegaremos ao limite das sucessões entre novas ondas tecnológicas, pois já estamos na onda web 3.0 e muitos já especulam a web 4.0 batendo em nossa porta. É inegável a evolução contínua da sociedade para abranger seus conhecimentos e fazer florir novas fontes de renda as quais prometem ser inovadoras e revolucionárias diante de tantas criações. A falta de planejamento na criação e desenvolvimento de um software para web pode acarretar em danos irreparáveis estando o serviço em funcionamento e não tendo viabilização de mão de obra profissional para tais reparos, contudo as ferramentas hoje disponíveis quase que fazem todo o processo de forma automática por seu operador, ao qual apenas direciona os parâmetros para o caminho correto.

## 1.3 Metodologia

Para a consolidação do conteúdo, foi realizada pesquisa em livros e sites que disponibilizam citações, matérias e até mesmo o próprio serviço por reconhecimento ao abranger usuários que queiram utilizar os seus softwares pagos, trial ou mesmo open source.

A união das informações a seguir, demonstra que o conteúdo elaborado tem como visto uma gama de derivações postas em sites informativos para orientação aos seus internautas.

O presente trabalho, do ponto de vista da natureza pode ser classificado como uma pesquisa exploratória aplicada. Do ponto de vista técnico esse trabalho pode ser classificado como um estudo de caso.

As metodologias utilizadas consistem em coletar e analisar informações sobre um determinado software e/ou framework, a fim de estudar aspectos relacionados ao tema de desenvolvimento web.

Neste método foi buscada a aplicação prática dos conhecimentos para uma solução que resultou em um artigo para análise dos parâmetros das ferramentas propostas e exemplificação do conteúdo.

#### 1.4 Estrutura do Trabalho

O trabalho está dividido em 12 capítulos. Destes, o primeiro apresenta a introdução, objetivos e justificativa para a realização deste trabalho.

No Capítulo 2 apresenta o referencial teórico sobre os conceitos e características dos documentos de especificação de softwares, e linguagens de programação utilizadas no desenvolvimento do sistema proposto e apresenta também um sistema similar ao proposto pelo presente trabalho.

O Capítulo 3 apresenta os materiais e métodos utilizados para a realização deste trabalho.

O Capítulo 4 apresenta os resultados obtidos através de um comparativo de utilização entre o sistema proposto e o similar.

Finalizando, no capítulo 5 são apresentadas as conclusões, as dificuldades encontradas e as sugestões de trabalhos futuros.

## 2. Os conceitos das Web's

Como afirma Evaldo Junior Bento (2013, p. 17)

“O segredo da famosa web 2.0 é a capacidade de interação entre as pessoas e os serviços online. Mas, para que esta interação seja possível, é necessário que os sites sejam capazes de receber informações dos internautas e também de exibir conteúdo personalizados para cada um ou de mudar seu conteúdo automaticamente, sem que o desenvolvedor precise criar um novo HTML para isso.”

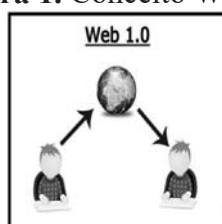
Porém, o conceito de web 2.0 já está começando a ser ultrapassado pela nova onda chamada Web 3.0 anunciada como a terceira onda da Internet, projeta estruturar todo o conteúdo disponível na rede mundial de computadores dentro dos conceitos de “compreensão das máquinas” e “semântica das redes”. Esta nova Web também pode ser chamada de “A Web Inteligente”.

## 2.1 Diferenças entre Web 1.0, 2.0 e 3.0.

A democracia impera na internet em forma de comunicação, é possível encontrar diferentes conteúdos e acessar de maneira ilimitada tantos recursos que passaríamos várias vidas para vê-los. O conteúdo já publicado até hoje é incrivelmente enorme, muita informação para poder assimilar, mas o que seria na verdade a WEB 1.0, 2.0 e 3.0? veremos a seguir as diferenças:

### 2.1.1 Web 1.0

**Figura 1. Conceito Web 1.0**



Fonte: <https://rosedreamer22.wordpress.com/2015/08/05/web-1-0-and-web-2-0/>

As principais diferenças entre esses formatos de WEB estão na dinâmica e interatividade. Um site com o formato de WEB 1.0 é estático e sem nenhuma forma de interatividade com os leitores. É possível visitá-lo muitas vezes, mas não haverá nada de novo em todas as novas visitas e não há motivo para retornar a ele.

### 2.1.2 Web 2.0

**Figura 2. Conceito Web 2.0**



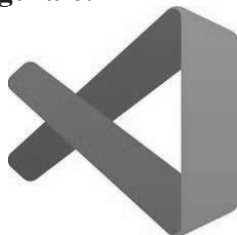
Fonte: <https://medium.com/@CarlaMariaMart6/web-2-0-cd53ce686a00>

Com o tempo, a maior parte dos sites migrou do formato 1.0 para o formato 2.0, que consiste em uma maior interação dentro de cada página. A partir desse novo formato de WEB, foi possível a criação de blogs e da Wikipédia, em que o leitor não é passivo em relação ao que está publicado.



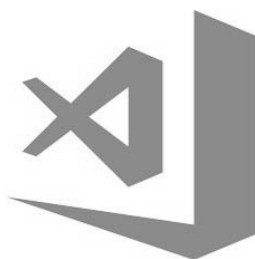




**Figura 6. Visual Studio**

Fonte: <https://visual-studio-code.br.uptodown.com/windows>

**Visual Studio Code (VS Code)** – Visual Studio Code é um editor leve mas poderoso e com uma interface bem pensada que funciona em seu desktop e está disponível para Windows, Mac e Linux. Ele tem suporte a JavaScript, TypeScript e Node.js e possui um rico ecossistema de extensões para outras linguagens (como C ++, C #, Python, PHP) e runtimes. O VS Code é ideal se você precisa um editor leve que não usa muito espaço e recursos.

**Figura 7. Visual Studio Code**

Fonte: [https://pt.wikipedia.org/wiki/Ficheiro:Visual\\_Studio\\_Code\\_1.17\\_icon.svg](https://pt.wikipedia.org/wiki/Ficheiro:Visual_Studio_Code_1.17_icon.svg)

**Atom** – Atom é um editor de texto moderno, acessível, mas ‘hackable to the core’, isso significa que você pode personalizar tudo para fazer qualquer coisa, sem tocar a mão nos arquivos de configurações. Ele é um aplicativo de desktop construído com HTML, JavaScript, CSS, e integração com node. Ele roda em Electron, um framework para construir aplicativos multiplataforma usando tecnologias web. Ele possui gerenciador de pacotes e pacotes extensas (como plugins, definições de sintaxes, menus, snippets, etc.) escritos em CoffeeScript.

**Figura 8. Atom IDE**

Fonte: <https://fossbytes.com/atom-ide-github-how-install/>

**Sublime** – O Sublime é um editor de texto sofisticado para código, marcação e texto com grande desempenho. Ele é útil especialmente para arquivos individuais porque ele executa mais rápido do que o VS Code. Oferece um alto nível de personalização, quase

nunca falha e inclui plugins bacanas e pacotes como Origami e GitGutter que tornam o desenvolvimento realmente divertido.

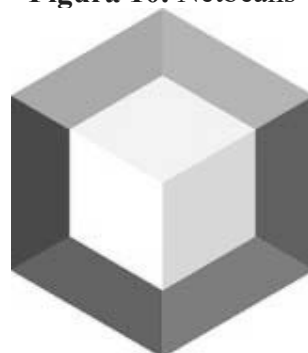
**Figura 9.** Sublime IDE



Fonte: <https://worldvectorlogo.com/logo/sublime-text>

**Netbeans** – O NetBeans é uma das melhores IDE open source (código-fonte aberto) de PHP. Ele é enriquecido com recursos e é gratuito oferece suporte em várias línguas como Inglês, Japonês, Português BR, Russo e Chinês simplificado. A versão mais recente de Netbeans (versão 8.2) é leve, mais rápido e suporta tudo em relação de PHP. Ele suporta frameworks de PHP mais populares como Zend, Doctrine, Smarty e Symfony2 e também suporta Laravel através de laravel-ide-helper e outros frameworks como Yii, CakePHP, FuelPHP e WordPress CMS.

**Figura 10.** Netbeans



[https://commons.wikimedia.org/wiki/File:Apache\\_NetBeans\\_Logo.svg](https://commons.wikimedia.org/wiki/File:Apache_NetBeans_Logo.svg)

**Cloud9** – Cloud 9 é o IDE na nuvem mais usado e mais famoso para todas as linguagens de programação. Se você usa PHP, então o Cloud9 é ainda melhor para você porque ele oferece depurador (built-in debugger) para inspeção de código em tempo real.

**Figura 11.** Cloud9

Fonte: <https://www.crunchbase.com/organization/ajax-cloud9-ide>

## 5. Front-end Frameworks

Os Frameworks de front-end ou “CSS frameworks” são pacotes com código préescrito e padronizado em arquivos e pastas. Eles fornecem uma base para construir e ainda permitem flexibilidade com o design final. Eles ajudam a aumentar a sua produtividade sem sacrificar a qualidade do seu trabalho.

Um framework front-end típico contém os seguintes componentes:

- Um layout em grid para organizar os elementos de design de seu site
- Estilos de fonte pré-definidos e dimensionamento que variam baseado na sua função (por exemplo, tipografias diferentes para títulos e parágrafos)
- Componentes prontos do site, como painéis laterais, botões e barras de navegação

Exemplos de frameworks CSS Bootstrap – O framework mais popular de HTML, CSS e JS, Bootstrap facilita o desenvolvimento de sites responsivos e projetos mobile na web. Ele é open source, hospedado no GitHub e é perfeito para iniciantes, pois é o framework mais fácil de aprender e usar.

**Figura 12.** Bootstrap

Fonte: <https://getbootstrap.com.br/docs/4.1/getting-started/introduction/>

**Foundation** – Um framework que oferece muita flexibilidade e personalização, o Foundation é ideal se você é um veterano de desenvolvimento de front-end procurando um framework que seja superflexível. Como o Bootstrap, o Foundation também é gratuito e open source. Feito pelo pessoal da ZURB (uma empresa de product design), ele tem muitos recursos para apoiar o seu ciclo de desenvolvimento.

**Figura 13.** Foundation

Fonte: <https://webdesign.tutsplus.com/articles/how-they-did-it-the-foundation-framework--webdesign-17739>

**Materialize** – O Materialize é baseado no Material Design do Google para facilitar a construção de aplicações web responsivas e com visual elegante. Ele é um framework HTML, CSS e JavaScript que conta com recursos para simplificar o desenvolvimento de páginas responsivas que utiliza as filosofias de design do Material Design do Google. O Material Design combina os princípios clássicos de design bem sucedido, junto com a inovação e tecnologia.

**Figura 14.** materialize

Fonte: <https://twitter.com/materializecss>

## 6. Web App Frameworks

Os frameworks de web apps simplificam sua vida de desenvolvimento e existem para melhorar e facilitar o processo de construir um site. Normalmente eles fornecem ferramentas que cobrem os casos mais comuns de CRUD (Create, Read, Update, Delete ou Criar, Ler, Atualizar, Excluir). Existem muitos frameworks de web disponíveis.

### 6.1 Exemplos de Web App Frameworks

**ASP.NET** – Um framework de web app grátis e completamente suportado que ajuda a criar soluções de Web baseados em padrões. Ele cria sites baseados em HTML5, CSS, e JavaScript. O .NET Core é uma plataforma rápida, e leve para criar aplicativos e serviços de web que rodam em Windows, Linux e Mac. Ao contrário do framework .NET que funciona só em Windows, o .NET Core é de código-fonte aberto e multiplataforma, que você pode criar aplicativos de .NET Core que rodam em Windows, Linux and Mac OS.

**Figura 15.** ASP.NET Framework



Fonte: <https://networkencyclopedia.com/active-server-pages-asp/>

**Ruby on Rails** – o Ruby é grátis, open source e oferece uma integração boa com JS, que é perfeito para sites pesados de AJAX. Ele também inclui o Prototype Javascript Library que você pode integrar diretamente.

**Figura 16.** Ruby on Rails Framework



Fonte: [https://commons.wikimedia.org/wiki/File:Ruby\\_on\\_Rails-logo.png](https://commons.wikimedia.org/wiki/File:Ruby_on_Rails-logo.png)

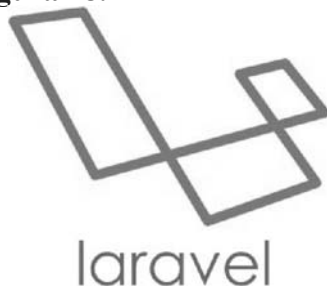
**Django** – O Django é um framework Web de alto nível em Python que facilita o desenvolvimento rápido e o design limpo e pragmático. Ele oferece uma seção de administração de site autogerada para seus usuários (de site) para criar, editar e atualizar o conteúdo. Ele também inclui ferramentas para armazenar dados em cache e construir URLs flexíveis. Se você usa Python, então confira este blog extremamente útil sobre programação com Python.

**Figura 17.** Django Framework



Fonte: <https://www.treinaweb.com.br/blog/django-ou-flask-eis-a-questao/>

**Laravel** – O framework mais famoso de PHP, Laravel é simples e muito fácil de usar. Ele também oferece maior compatibilidade e desenvolvimento mais rápido do que qualquer outro framework de PHP. Ele é simples, tem tudo o que precisa para desenvolver uma aplicação robusta, inclusive funcionalidades importantes de outros frameworks, como database migrations. Tudo isso sem muitas configurações.

**Figura 18.** Laravel Framework

Fonte: <https://seeklogo.com/vector-logo/311758/laravel-framework>

**Phalcon** – Se a sua preocupação principal é a velocidade e desempenho, então escolhe o Phalcon, o framework mais rápido do PHP.

**Figura 19.** Phalcon framework

Fonte: <http://blog.discustecnologia.com.br/?p=19>

## 7. Terminal de Linha de Comando

A ferramenta mais básica (para usuários Windows apenas) é CMD (Command Line Prompt) – o processador de comandos do Windows. Nossos especialistas recomendam a usar ferramentas poderosas como o Powershell (para o Windows), o Bash (para o Linux) e o Terminal (para o Mac). É importante aprender como usar elas porque ferramentas como o Git, o MongoDB, o MySQL, entre outras “te obrigam” a usar linha de comando.

## 8. Versionador de Código

Um passo muito importante para qualquer desenvolvedor é aprender a usar um versionador de código, que basicamente vai ajudar a guardar histórico da evolução do desenvolvimento, guardar o código fonte em um local seguro para evitar perdas, além de ser essencial quando se trabalha em equipe. Existem vários versionadores, como o Team Foundation Server (TFS), Git, SVN, dentre outros, além de produtos SaaS como Visual Studio Online, GitHub e BitBucket.

**Figura 20.** Git 1.1

Fonte: <https://pt.wikipedia.org/wiki/Git>

## 9. Web Deploy

Escrever o código para criar aplicativos de app é uma coisa, mas fazer um deploy de aplicativo e fazê-lo funcionar sem problemas para o seu público é uma coisa completamente diferente. A seguir, veremos algumas ferramentas de web deploy que realmente vão reduzir dores de cabeça.

**Filezilla** – Se você é um iniciante, apenas está começando com desenvolvimento web, você pode usar um FTP simples como Filezilla para publicar seus recursos. Ele é uma ferramenta de FTP muito simples de troca de arquivos, que pode ser útil para fazer deploy de sites.

**Figura 21.** Filezilla



Fonte: <https://filezilla-project.org/>

**Git** – O FTP não possui todos os recursos necessários para o deploy e controle eficiente de aplicações web. Então você deveria usar uma ferramenta como o Git e construir seu processo de deploy com isso. O Git facilita controle de versão – sempre será possível rastrear cada alteração, e fazer deploy em múltiplos ambientes.

**Figura 22.** Git 1.2



Fonte: <https://pt.wikipedia.org/wiki/Git>

**Jenkins** – Jenkins é uma ferramenta de automação (CI – Continuous Integration) de deploy escrito em Java. Se você tem todo seu código numa máquina e publica usando o FTP, não valeria a pena usar o Jenkins. Mas se você mantê-lo num repositório e faça testes, valeria a pena usar um CI como Jenkins. Assim você pode automatizar toda essa parte, fora que deixa o processo de deploy muito mais fluido. Outra vantagem é de segurança.

**Figura 23. Jenkins**

Fonte: <https://www.jenkins.io/>

## 10. Outras Tecnologias

Além disso, existem várias outras ferramentas e tecnologias que facilitam bastante a vida de um desenvolvedor web.

**Postman** – O Postman é como um canivete suíço para devs de serviços da web. Ele é um cliente de HTTP muito poderoso que você pode testar serviços web construídos sob o modelo REST principalmente, mas funciona para a web em geral. O Postman é um App do Chrome que se destaca por ter um ótimo recurso de compartilhamento entre usuários.

**NewRelic** – Saber exatamente o que está acontecendo com seu site ou aplicação em produção é algo muito valioso no mundo de hoje. Devs sabem que o mundo localhost é maravilhoso, lá temos informações em abundância para debugar códigos e otimizar scripts, conseguimos ter controle total sobre o que está acontecendo. Porém, quando se coloca uma aplicação para um ambiente de produção, muitas vezes perdemos o controle das coisas.

**Chrome Dev Tools** – Não tem como se aventurar no desenvolvimento front-end sem usar o Chrome DevTools, o tão famoso F12.

**AngularJS** – O AngularJS é um framework open source mantido principalmente pelo Google e usa o conceito de SPA (Single Page Application). Nesse tipo de aplicação, apenas uma parte dela muda e a outra permanece fixa, fazendo parte de um template compartilhando entre as páginas. Ele possui ferramentas para buscar dados, processá-lo, exibi-lo na tela e manipular a entrada do usuário (como cliques ou pressionamentos de teclas) de uma forma inteligente. Ele simplifica desenvolvimento fornecendo um framework confiável no estilo MVC (Model View Controller). Ele é ótimo para construir SPAs que não precisam muitas manipulações de DOM (aplicativos que não envolvem jogos e editores de GUI).

**Vue.js** – Vue.JS é uma biblioteca de JavaScript para desenvolver componentes reativos para interfaces web modernas. Desenvolver componentes que são reativos envolve observar um objeto javascript e refletir suas alterações no DOM do HTML. Fazer isso usando javascript puro ou jQuery não é fácil. VueJS torna isso extremamente simples.

**ReactJS** – Desenvolvido pelo Facebook, o ReactJS é um concorrente muito forte de AngularJS. Ele pode construir aplicativos em grande escala que envolvem muitas manipulações de DOM. É porque ao contrário de AngularJS (versão 1), o ReactJS fornece algo como “Virtual-server side-DOM”, junto com o client-side DOM. A presença de dois DOM resulta em maior eficiência e capacidade de manipulações complexas de DOM. Os sites de Facebook e Instagram foram construídos usando ReactJS.



**Node.js** – O Node.js é uma plataforma construída sobre o motor JavaScript do Google Chrome para facilmente construir aplicações de rede rápidas e escaláveis. Ele usa um modelo de IO chamado ‘event-driven non-blocking IO model’ e tem um mecanismo JavaScript extremamente rápido do Google, o mecanismo V8. Usamos JavaScript normalmente como a linguagem de browser. O Node.js ajuda a usar o JavaScript no lado do server. Muitas ferramentas da web (web tools ou task runners) como Bower, Grunt, Gulp, etc usam o Node.js. Você também pode construir aplicações completas apenas usando o Node.js.

**Fontella** – O Fontella, nosso principal Front-end Designer, também usa o Node para seu trabalho, “O Node tem um papel fundamental no meu workflow front-end.

**Studio 3T (anteriormente o MongoChef)** – O Studio3T (anteriormente era conhecida como MongoChef). Ele suporta totalmente a integração de MongoDB shell e oferece todos os recursos para ajudá-lo a dominar seus bancos de dados de nível enterprise. Instalar e usar o Studio3T é super fácil. Você pode adicionar bancos de dados, usuários, coleções e muito mais, sem entrar no MongoDB shell.

**Genymotion** – O Genymotion é um emulador rápido de terceiros que pode ser usado em vez do emulador padrão do Android. Ele roda como uma máquina virtual em seu computador, então você obtém um desempenho muito melhor, e é muito mais como usar um dispositivo real em vez do emulador Android. O Genymotion é gratuito para desenvolvedores (uso não comercial) para usar, mas também tem versões pagas, dependendo de suas necessidades.

## 11. Comunidades de Desenvolvimento Web

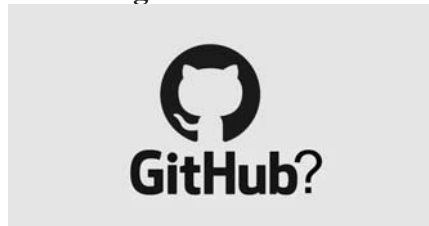
Cada desenvolvedor de web já enfrentou um problema. Quando você tem um problema, o que você faz? Você pesquisa no Google para encontrar uma resposta rápida. A web oferece tanto conteúdo certo na ponta dos dedos que torna mais fácil diagnosticar e solucionar problemas quando surgem.

É exatamente por isso existem várias comunidades para desenvolvedores ensinar, aprender e crescer.

### 11.1 Exemplos de comunidades

**Github** – O Github é o maior comunidade para desenvolvedores de web e app para compartilhar códigos e projetos. Ele também é uma comunidade que ajuda aumentar seu conhecimento e nível. Além disso, o GitHub oferece centenas de códigos prontos que você pode usar como um trampolim para seus projetos de desenvolvimento.

**Figura 24.** Github



Fonte: <https://screenrant.com/github-explained-what-is/>

**Stackoverflow** – O Stackoverflow é uma comunidade de devs, code gurus e entusiastas de webdesign que estão lá para ajudar uns aos outros com os problemas que podem acontecer dentro do processo de criação de sites. Você pode enviar suas dúvidas ou procurar respostas dentro da comunidade.

**Figura 25.** Stack over flow



Fonte: [https://pt.m.wikipedia.org/wiki/Ficheiro:Stack\\_Overflow\\_logo.svg](https://pt.m.wikipedia.org/wiki/Ficheiro:Stack_Overflow_logo.svg)

## 12. Conclusão

Em termos claros, vivemos sob uma constante evolução, e todos os dias somos obrigados a nos reinventar, amadurecer e criar novos meios de sermos úteis a sociedade, impactando direta ou indiretamente no futuro tecnológico.

Vivemos num mundo consumista onde diariamente muito conteúdo informativo oriundos de jornal, revistas ou ainda programas de televisão, mas isso acabou se tornando chato com o tempo, e com linguagens estáticas como HTML e CSS isso tornou-se um relacionamento monótono, mas com os recursos que o Java Script, PHP e MySQL, entre outras linguagens mais abrangentes trazem consigo uma variedade de funcionalidades, e as páginas começaram a criar vida tornando-se intuitivas e como uma via de mão dupla onde o usuário dispõe de conteúdo estático e maleável tornando a experiência mais intuitiva fazendo ser possível não só leitura como permitir interação com os internautas. A adequação dos montantes de informações bem alocadas viabiliza o interesse dos usuários a interagir e se fixarem por um longo tempo no conteúdo fazendo assim o dono do serviço dispor de seus advindos investimentos de merchandising. Vimos que há um montante de softwares capazes de criar tantas possibilidades que se torna quase impossível utilizar todos. O programador precisa realizar uma preparação anterior e consultar exatamente o interesse do cliente para a realização de criação do conteúdo ao qual será desenvolvido.

### 13. Bibliografia

BENTO, Evaldo Junior. **Desenvolvimento Web com PHP eMySQL**. São Paulo: Casa do Código, 2013.

ZEMEL, Tércio. **Web Design Responsivo**. São Paulo: Casa do Código, 2014.

<<https://blog.umbler.com/br/ferramentas-para-desenvolvedores-da-web/>>. Acesso em: 07 de jun. de 2020.

<<https://tecnologia.culturamix.com/internet/a-diferenca-entre-web-2-0-e-web-1-0/>>. Acesso em: 07 de jun. de 2020.

<<https://rosedreamer22.wordpress.com/2015/08/05/web-1-0-and-web-2-0/>>. Acesso em: 07 de jun. de 2020.

<<https://medium.com/@CarlaMariaMart6/web-2-0-cd53ce686a00>>. Acesso em: 07 de jun. de 2020.

<<http://marcosmucheroni.pro.br/blog/?p=9663>>. Acesso em: 07 de jun. de 2020.

<<https://www.idealmarketing.com.br/blog/o-que-e-web-3-0/>>. Acesso em: 07 de jun. de 2020.

<<https://blog.umbler.com/br/ferramentas-para-desenvolvedores-da-web/>>. Acesso em: 07 de jun. de 2020.

<<https://visual-studio-code.br.uptodown.com/windows>>. Acesso em: 07 de jun. de 2020.

<[https://pt.wikipedia.org/wiki/Ficheiro:Visual\\_Studio\\_Code\\_1.17\\_icon.svg](https://pt.wikipedia.org/wiki/Ficheiro:Visual_Studio_Code_1.17_icon.svg)>. Acesso em: 07 de jun. de 2020.

<<https://worldvectorlogo.com/logo/sublime-text>>. Acesso em: 07 de jun. de 2020.

<[https://commons.wikimedia.org/wiki/File:Apache\\_NetBeans\\_Logo.svg](https://commons.wikimedia.org/wiki/File:Apache_NetBeans_Logo.svg)>. Acesso em: 07 de jun. de 2020.

<<https://www.crunchbase.com/organization/ajax-cloud9-ide-cloud9>>. Acesso em: 07 de jun. de 2020.

<<https://getbootstrap.com.br/docs/4.1/getting-started/introduction/>>. Acesso em: 07 de jun. de 2020.

<<https://webdesign.tutsplus.com/articles/how-they-did-it-the-foundation-framework--webdesign-17739>>. Acesso em: 07 de jun. de 2020.

<<https://twitter.com/materializecss>>. Acesso em: 07 de jun. de 2020.

<<https://networkencyclopedia.com/active-server-pages-asp/>>. Acesso em: 07 de jun. de 2020.

<[https://commons.wikimedia.org/wiki/File:Ruby\\_on\\_Rails-logo.png](https://commons.wikimedia.org/wiki/File:Ruby_on_Rails-logo.png)>. Acesso em: 07 de jun. de 2020.

<<https://www.treinaweb.com.br/blog/django-ou-flask-eis-a-questao/>>. Acesso em: 07 de jun. de 2020.

<<https://seeklogo.com/vector-logo/311758/laravel-framework>>. Acesso em: 07 de jun. de 2020.

<<https://pt.wikipedia.org/wiki/Git>>. Acesso em: 07 de jun. de 2020.

<<https://filezilla-project.org/>>. Acesso em: 07 de jun. de 2020.

<<https://www.jenkins.io/>>. Acesso em: 07 de jun. de 2020.

<<https://screenrant.com/github-explained-what-is/>>. Acesso em: 07 de jun. de 2020.

<[https://pt.m.wikipedia.org/wiki/Ficheiro:Stack\\_Overflow\\_logo.svg](https://pt.m.wikipedia.org/wiki/Ficheiro:Stack_Overflow_logo.svg)>. Acesso em: 07 de jun. de 2020.