**Primeiros passos para desenvolvimento web**

**TCP x UDP:**

|  |  |
| --- | --- |
| **TCP** | **UDP** |
| Voltado à conexão | Rápido |
| Handshake | Não confiável |
| Integridade, ordem dos dados | “Carro do ovo” (só dispara) |
| Aplicativo de mensagens de texto | Livestream |

**Portas:**

|  |  |  |
| --- | --- | --- |
| 20 | FTP |  |
| 22 | SSH | Conexão segura |
| 25 | SMTP | Email |
| 53 | DNS | Tradutor ip/nome |
| 80 | HTTP | Requisição simples |
| 443 | HTTPS | Requisição segura |

**Modem**

* Modulator-demodulator.
* Converte dados em formato que possa ser transmitido de um computador para outro e lido por outro.

**Roteador**

* Distribui internet para um ou mais dispositivos de uma rede.
* Pode fazer comunicação entre redes.
* Pode ser “burro” (carro do ovo).

**Switch**

* Distribui internet para um ou mais dispositivos de uma rede.
* Criado para ser “inteligente”.

**SMS**

0 custo.

O celular troca, naturalmente, alguns bits com as torres de comunicação.

**MMS**

Transmissão de mensagens multimídia (áudio, vídeo) por meio de uma espécie de conexão de dados primitiva.

**Tabela

Descrição gerada automaticamente**

**Browser**

* Sabe identificar várias LP, linguagens de marcação e conteúdo multimídia
* Existem plug-ins, ou add-ons, que ajudam na navegação
* Cache e cookies

**Web Server**

**Estático**

* É um servidor físico onde são armazenados arquivos, softwares e/ou banco de dados

**Dinâmico**

* Se refere aos softwares que estão presentes no servidor físico

**Web-server dinâmico**

Arquivos (file server)

Aplicações (application server)

Banco de dados (database)

Tudo junto e misturado

**Web-service**

* Interface disponível para fazer requisições e consultas em bancos de dados inacessíveis (Correios, Governo)

**Stacks**

* Pilhas de tecnologia.
* Conjunto de softwares necessários e suficientes para executar um aplicativo/programa.
* Linguagens de programação.
* Ambientes e ferramentas de interação com o app/sw.
* Capacidade e limitação de performance.
* Pontos fortes e fracos do app/sw.

**Front-end:**

* “Parte da frente”.
* Site, software, aplicativo, web service.
* Interface, UI, UX.
* Lógica de programação, HTML, CSS, jQuery/JS/AJAX, PHP, Bootstrap/outros frameworks.

**Back-end:**

* “Parte de trás”.
* Servidores, bancos de dados.
* “meio-de-campo” entre interface e bancos de dados, regras de negócios, validações.
* MySQL, Oracle, protocolos de comunicação, PHP, Java, node,js.

**Full stack:**

* Profissional que sabe trabalhar em todas as camadas das tecnologias de desenvolvimento/execução de um app/sw.



**Principais Linguagens**

* HTML
* CSS
* JavaScript (jQuery, AJAX, diversas libs)
* PHP
* .NET
* ASP
* Java
* Ruby (on Rails)
* Python
* Perl
* C/C++/C#

**Termos comuns**

* 404
* Algoritmo
* ALT
* API
* Aplicação
* Back-end
* Biblioteca/dll (conjunto de ferramentas disponíveis)
* Bootstrap (Framework JavaScript)
* Breakpoints (Pontos chaves de quando montamos uma tela, ou pontos de parada em debug)
* Browser/navegador
* Bug
* Cache
* Código
* Controle de versão
* Cookies
* Debug
* Deploy (pacote enviado para ser publicado)
* Desing adaptativo/responsivo (Se adequa a qualquer proporsão)
* DNS
* Documentação
* DPI (Densidade de pixels)
* Editor de texto
* Estrutura de dados (Fila, array, char, interger)
* Favicon (ícone do site na aba)
* Fontes
* Framework
* Front-end
* FTP (Protocolo de comunicação quase exclusivamente para troca de arquivos)
* Full-stack
* GitHub/SVN
* HTTP(S)
* IP
* Linguagem
* Meta tags (tags html)
* Método ágil
* Mobile
* MVC (Model View Controller)
* MVP (Minimum Viable Product)
* MySQL
* Pixel (Picture Elements)
* Resolução
* Servidor
* Sistema operacional/SO
* Solução
* SSL (Camada de seguraça)
* UI (User Interface)
* UX (User Experience)
* Versão
* WYSIWYG (What You See Is What You Get)