***Módulo 1 - Introdução***

***Mercado de TI***

Você sabia que o mercado de tecnologia demanda uma grande quantidade de profissionais qualificados? Segundo a Brasscom (Associação Brasileira das Empresas de Tecnologia da Informação e Comunicação), “**A oferta de 46 mil formandos ao ano é insuficiente para atender os 70 mil profissionais demandados ao ano até 2024**”.

Muitas empresas e especialistas apontam que essa demanda aumentou após o início da pandemia, pois as empresas aceleraram o processo de transformação digital e o número de startups baseadas em tecnologia não para de crescer. Esses movimentos acarretam em uma maior demanda por profissionais na área de tecnologia.

Um outro dado interessante apontado pela Brasscom é que o profissional da área de software recebe **quase 3 vezes mais, que a média salarial registrada no Brasil**, ou seja, enquanto a média salarial nacional é de R$ 1.836,00, o profissional da área de software recebe R$ 5.066,00, em média.

Além de todas as oportunidades existentes hoje, muitas das profissões relacionadas a área de tecnologia estão entre as chamadas **profissões do futuro**.

**Módulo 2 - HTML & CSS**

***Desenvolvedor Front End & Back End***O que faz um Desenvolvedor Web?

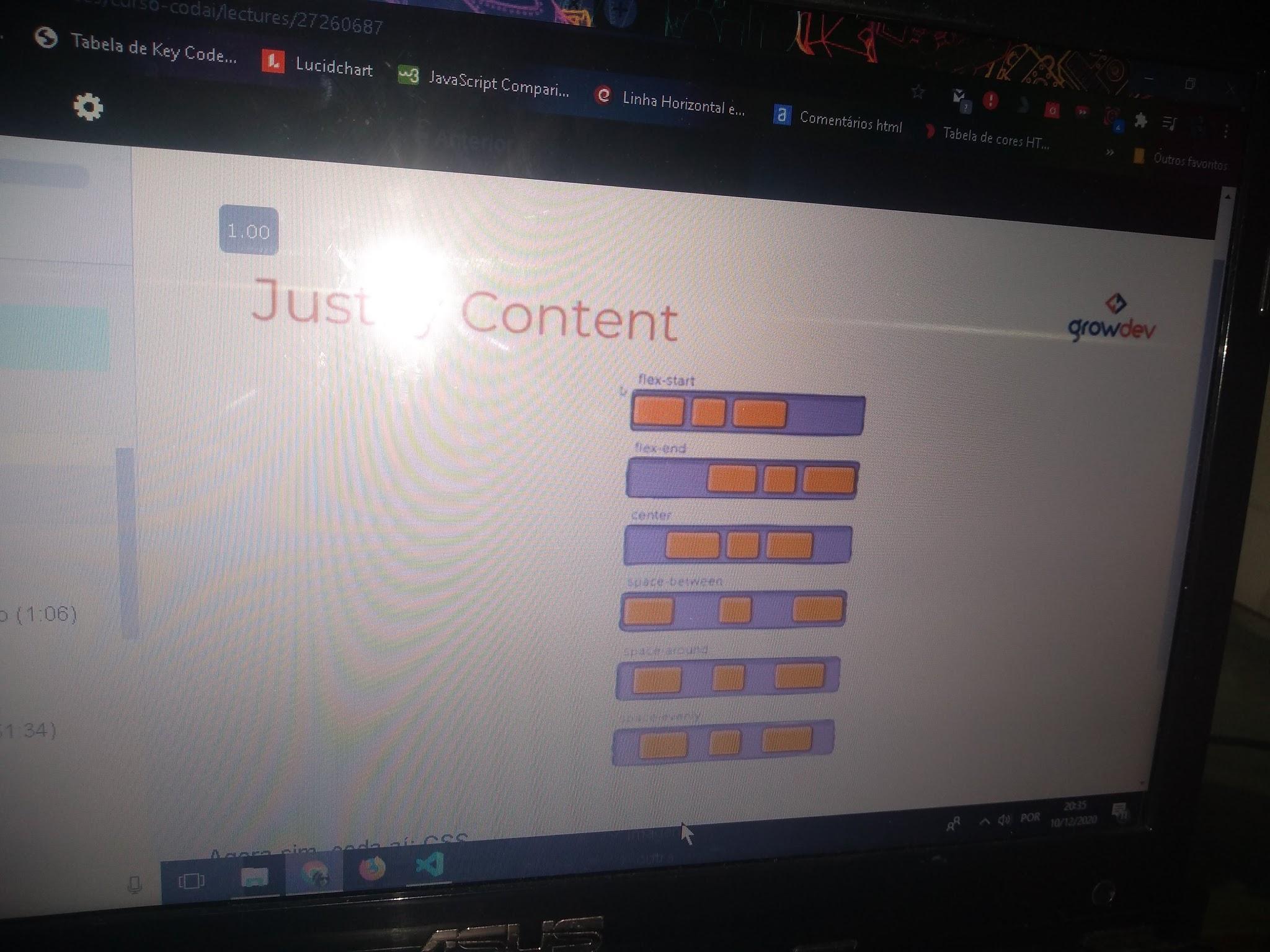
Essencialmente, um desenvolvedor web realiza o desenvolvimento de websites ou aplicações na internet como google, facebook e portais de notícia.

Este profissional normalmente divide-se em duas áreas de atuação: **front-end e back-end**.

* **Front-end** — Este desenvolvedor é responsável por toda a parte que você pode efetivamente ver em um site, a chamada "camada gráfica". O foco é na experiência que o usuário tem ao fazer um acesso no sistema, cuidando desde o layout até pequenos detalhes de menus e rodapés, por exemplo. Costuma trabalhar com HTML, CSS e JavaScript. Trabalha com a parte da aplicação que interage diretamente com o usuário;
* **Back-end** — Esse é o profissional responsável pela lógica do negócio do site, garantindo que os dados enviados para o navegador estejam corretos no sistema e em seu dia-a-dia atua desenvolvendo cálculos e manipulação de banco de dados. Trabalha com os dados e informações, responsável por tudo o que está “atrás dos panos” no código.  
    
  ***IDE - Ambiente de Desenvolvimento Integrado***Para podermos escrever nossas primeiras linhas de código, precisamos de um editor de código-fonte. É bem comum este editor estar dentro de uma IDE (Integrated Development Environment), que significa Ambiente de Desenvolvimento Integrado.   
    
  ***Introdução ao HTML***O HTML é uma linguagem de marcação. Sua primeira versão foi publicada em 1991, e ela se tornou tão popular que se tornou a linguagem padrão da web. Foram surgindo novas versões, com melhorias e avanços, e por isso você pode já ter visto que o HTML5 é a versão que nós utilizamos atualmente, e que iremos aprender neste módulo. E afinal, o que quer dizer estas siglas do HTML? Do inglês HiperText Markup Language, traduzindo Linguagem de Marcação de Hipertexto. Se formos comparar uma página na internet como um corpo humano, o HTML seria como o esqueleto, os ossos que dão a estrutura. E depois com o estilo de formas, cores, utilizamos o CSS!   
  Em uma página web, o HTML formará a primeira camada que apresenta o conteúdo ao usuário. O CSS formará a segunda camada que dará forma e estilos aos elementos. Finalmente, o Javascript formará a terceira camada que adicionará os comportamentos e funcionalidades à página.   
  O HTML é uma linguagem baseada em marcação. Nós marcamos os elementos para mostrar quais informações a página exibe.  
    
  **TAGS HTML:**

<h1> Título em Destaque </h1>  
  
**ESTRUTURA BÁSICA:**<!DOCTYPE html>  
<html lang=”pt-br”>  
<head> //cabeçalho da página  
 <meta charset=”UTF - 8”>  
 <title> Document </title>  
</head>  
<body> //corpo da página  
 <h1> Título em destaque </h1>  
</body>  
</html>  
height=altura  
width=largura  
ul=lista não ordenada(círculo)  
ol=lista ordenada(número à esquerda ordenando)  
dl=lista descritiva;dd=detalhar um termo  
tr=primeira linha(cabeçalho da página)  
th=célula do cabeçalho da tabela  
td=célula normal do corpo da tabela  
rowspan=mesclagem de linhas e quantas  
mesclar na horizontal=colspan  
thead=cabeçalho  
tbody=corpo  
tfoot ou footer(CSS)=rodapé  
style="width: 100%" //tabela ocupando toda a página  
section=sessão  
small=texto menor  
  
  
***Introdução ao CSS*O CSS (Cascading Style Sheet ou Folha de Estilo em Cascata)** é usado para dar estilo aos elementos escritos em linguagem de marcação, como HTML. Ele separa o conteúdo que representa a parte visual do site. Utilizando o CSS é possível alterar a cor do texto e do fundo, fontes de texto e espaçamento entre parágrafos, além de muitos outros.

CSS foi desenvolvido em 1996, por uma razão bem simples: o HTML não havia sido projetado para receber a formatação de estilos em uma página. Então, para trazer personalidade e vida a cada página web construída com HTML, se torna essencial saber como usar o CSS na prática.   
  
cor estilizada: hexadecimal (#000… preto)  
link:css  
online color packer no Google  
text-align: justify=texto longo  
font-size=tamanho da fonte/aumentar texto  
para mudar a cor de um dos parágrafos do texto: criar classe  
  
**CSS FLEXBOX:** A ideia é simples: os filhos de um elemento com flexbox podem se posicionar em qualquer direção e podem ter dimensões flexíveis para se adaptar. Você pode posicionar os diversos elementos independente da sua posição na estrutura do HTML, o que é muito bom.  
  
**Display**

* Esta propriedade define um container flex;
* Habilita o container flex para todos os seus filhos diretos.  
  .container {  
   display: flex; /\* or inline-flex \*/  
  }  
    
  **Flex Direction**
* Esta propriedade estabelece o eixo principal, definindo assim a direção em que os flex itens são colocados no container.  
  .container {  
   flex-direction: row | row-reverse | column(coluna de cima p/ baixo) | column-reverse(de baixo p/cima);   
  } //da esquerda p/ direita, da direita p/ esquerda, da horizontal p/ vertical e da vertical p/ horizontal  
    
  **Justify Content**
* Define o alongamento ao longo do eixo principal. Ajuda a distribuir o espaço livre extra restante quanto todos os itens flexíveis em uma linha são inflexíveis ou são flexíveis, mas atingiram seu tamanho máximo. Ele também exerce algum controle sobre o alinhamento dos itens quando eles ultrapassam a linha.  
  .container {  
   justify-content: flex-start | flex-end | center | space-between | space-around(entre cada um dos elementos) | space-evenly(distribui o espaço uniformemente | start | end | left | right … \* safe | unsafe;   
  }  
    
    
  align-items: similar à justify-content mas trabalha com eixo cruzado

**Módulo 3 - Framework Bootstrap**

## ***Framework* O que é um framework?**

## O framework é um pacote de códigos prontos que podem ser utilizados no desenvolvimento de uma aplicação web. A proposta de uso dessa ferramenta é aplicar funcionalidades, comandos e estruturas já prontas para garantir qualidade e produtividade no projeto.

## É possível encontrar diversas definições para framework, algumas simples, outras mais elaboradas, mas o ponto comum entre todas é a **reusabilidade(adaptar uma classe existente em um sistema a outro sistema, para executar um mesmo conjunto de ações)**.

## Um framework tem como principal objetivo resolver problemas recorrentes com uma abordagem genérica, permitindo focar seus esforços na resolução do problema em si, e não ficar reescrevendo código.

## Um framework no [desenvolvimento de](https://pt.wikipedia.org/wiki/Desenvolvimento_de_software) uma aplicação web, é uma [abstração](https://pt.wikipedia.org/wiki/Abstra%C3%A7%C3%A3o) que une [códigos](https://pt.wikipedia.org/wiki/C%C3%B3digo) comuns provendo uma funcionalidade genérica para serem usadas em vários projetos diferentes.

## Por exemplo, se construíssemos um prédio com estruturas pré-moldadas, como paredes e tudo mais, precisaríamos apenas ajustar seus posicionamentos, não precisando colocar tijolo por tijolo, facilitando o trabalho e o tempo de construção.

## Essa é a ideia de um framework, gerar produtividade, qualidade e padronização na construção de uma aplicação. Tem como objetivo agilizar o desenvolvimento do seu código.

## **O que é um framework front-end?** Nesse contexto, os frameworks front-end servem como ferramentas que auxiliam na criação de telas e na forma como os dados são exibidos, ou seja, ajudam a definir como será o visual desde um botão até o posicionamento de cada elemento na tela. **Introdução ao Bootstrap** O **Bootstrap** foi desenvolvido por engenheiros do Twitter, com objetivo de otimizar mais o desenvolvimento. Em resumo, são códigos prontos que podem ser customizados, para ser usado e acelerar seu desenvolvimento. Uma das características principais é que ele auxilia na **responsividade**, algo essencial nos dias de hoje.

## Atualmente, temos muitos dispositivos com tamanhos de telas diferentes usados pelos usuários, então cada página precisa se adaptar para que o usuário consiga ter a melhor experiência.

## Acesso ao framework: <https://getbootstrap.com/> h1: boostrap heading h2: boostrap heading h3: boostrap heading h4: boostrap heading h5: boostrap heading h6: boostrap heading

## 

## 

## 