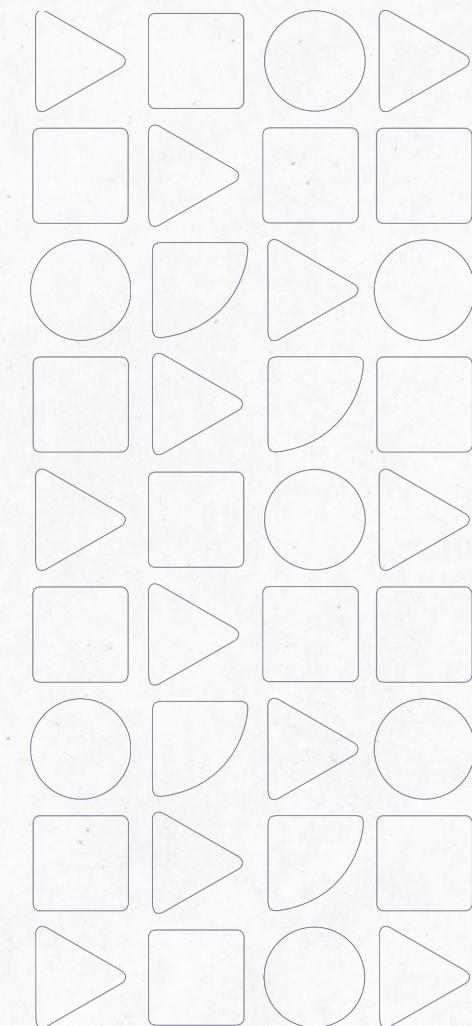


*Front-End, Back-End e Full Stack*

**Disciplina:** Linguagem de Programação



## Conteúdos:

*Front-end, back-end e full stack.*

## Habilidade(s):

- Identificar as áreas de atuação de um desenvolvedor;
- Identificar o melhor *software* de desenvolvimento, de acordo com a necessidade;
- Utilizar ambientes de desenvolvimento de *software* para aplicativos móveis.

# Bloco 1

---

Conhecendo o *front-end*.

# Que profissional eu quero ser?

Quanto aos temas de programação estudados até agora ou que você conhece previamente, responda:

**Quais temas você mais  
gostou?**

**Quais temas você  
menos gostou?**

**No que você é bom?**



## Escolhendo a área profissional

Na hora de escolher em que área seguir, é imprescindível que um indivíduo esteja atento a diversos aspectos, como **afinidade, habilidades, valores pessoais e oportunidade de mercado**.

# Área profissional na programação

Escolher em qual área da programação se concentrar é uma decisão importante, pois isso determinará o seu caminho profissional.

Hoje, conheceremos algumas áreas da programação!



# Como escolher?

Não há um manual de instruções de quais passos são necessários seguir.

Aqui, toda trajetória é válida. Entretanto, você pode estar atento a algumas dicas que podem facilitar o seu caminho!

## Confira

**Autoavaliação:** identifique os seus interesses e as suas paixões.

**Converse com profissionais:** são experientes no assunto!

**Habilidades:** quais são as suas técnicas atuais?

**Recursos:** invista em cursos introdutórios.

**Pesquisa de mercado:** explore as tendências da indústria.

**Projetos:** inicie projetos pessoais em diferentes áreas.



# Conhecendo os tipos de desenvolvedores

Embora a área da programação seja vasta, hoje nós vamos conhecer os três tipos de profissionais mais comuns quando o assunto é desenvolvimento. São eles:



*Front-end*

Especialistas em interagir com a interface.



*Back-end*

Aqui, são mais focados nos servidores.



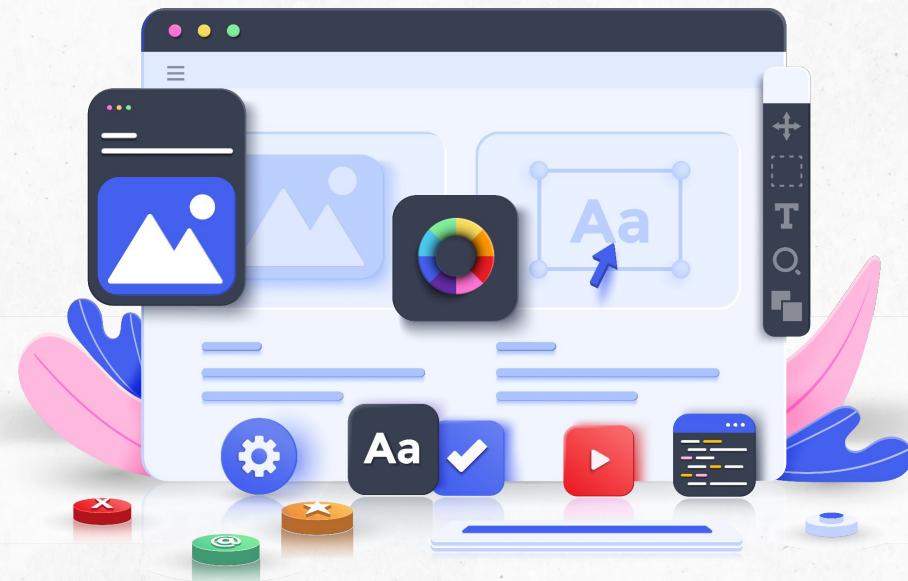
*Full stack*

São experientes nas duas áreas!

# Front-end

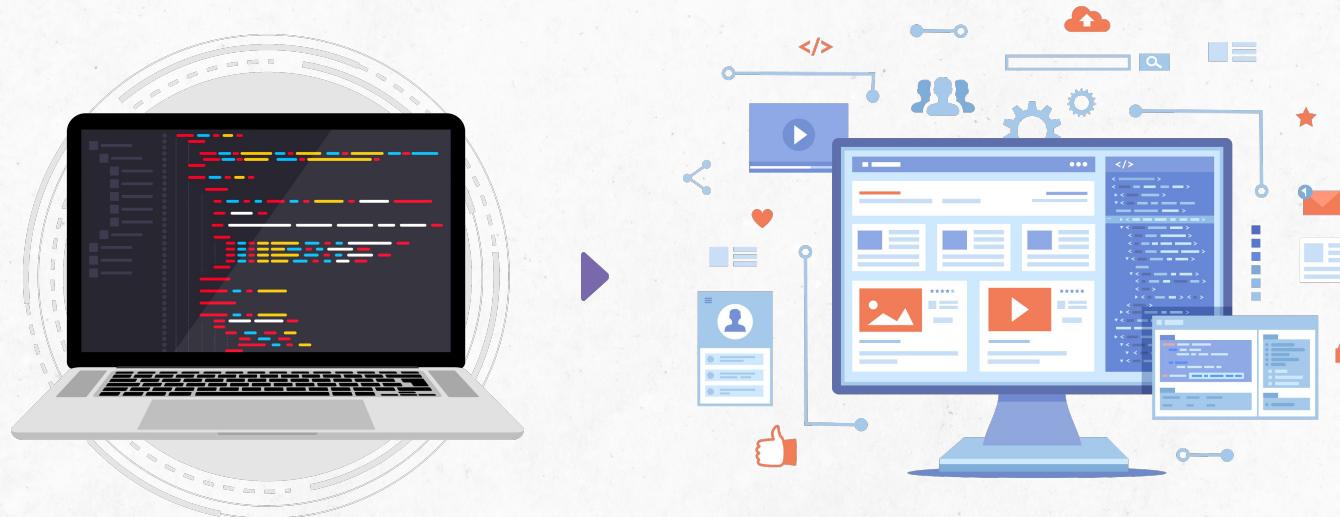
É responsável pelo desenvolvimento de toda a parte visual de um *site* ou aplicativo, para que seja possível haver interação com o usuário.

Quando você clica em um botão de **curtir**, por exemplo, é graças ao trabalho do *front-end*!



# Principais responsabilidades do front-end

Ele é responsável por implementar os códigos de uma maneira que seja acessível para os usuários.



# Quais conhecimentos um *front-end* deve ter?

Esse profissional deve possuir uma variedade de conhecimentos técnicos e habilidades para desempenhar as suas responsabilidades com eficácia.

A seguir, vamos ver os principais.



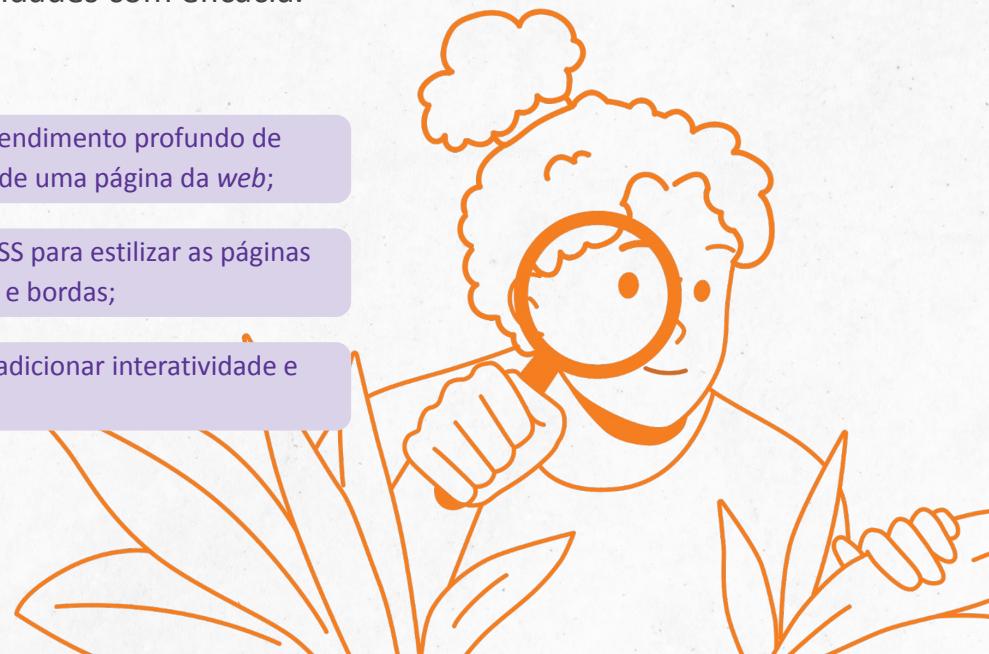
**HTML (*Hypertext Markup Language*):** deve ter um entendimento profundo de HTML, a linguagem usada para criar a estrutura básica de uma página da web;



**CSS (*Cascading Style Sheets*):** deve ser capaz de usar CSS para estilizar as páginas da web, incluindo formatação de texto, cores, margens e bordas;



**JavaScript:** deve ter conhecimento em JavaScript para adicionar interatividade e dinamismo às páginas da web.



# Principais responsabilidades do front-end

A seguir, confira no que ele está envolvido.

1

Criação de *layouts*, elementos gráficos, botões, menus e todos os elementos visuais.

2

Garante que a interface seja compatível com diferentes navegadores (Chrome, Firefox, Safari etc.) e dispositivos (computadores, tablets e mais).

3

Concentra-se na experiência do usuário, tornando a interface do usuário intuitiva e fácil de usar.

4

Trabalha em estreita colaboração com *designers* gráficos, buscando traduzir conceitos de *design* em código funcional.

5

Ao desenvolver a interface do usuário, ele tem em mente a acessibilidade.

6

Precisa acompanhar as últimas tendências e atualizações para se manter atualizado.

# Qual é a diferença de um *designer* e um *front-end*?

## *Designer*

Concentra-se, principalmente, em questões estéticas e visuais, como *layout*, cores e tipografia.

Usam ferramentas como Adobe Photoshop, Illustrator, Sketch e Figma.

Garantem que o *design* seja atraente e eficaz.

Levam em consideração a acessibilidade visual, como o contraste de cores.

# VS

## *Front-end*

Concentram-se na implementação do *design*, tornando-o uma realidade funcional.

Usam HTML, CSS e JavaScript para criar a estrutura, estilo e interatividade.

Criam menus, botões e formulários para garantir que a página responda a eventos.

Garantem que a página seja responsiva, ou seja, que funcione bem em diferentes tamanhos.

## Média salarial de um *front-end*

Segundo o site **talent.com**, que baseou-se em 80 salários do Brasil, um *front-end* ganha, em média:

R\$2.750/mês.



# Bloco 2

---

O que é *back-end*?

## Interface que eu admiro

Você já parou para pensar quem está por trás dos aplicativos que você mais gosta?

Para realizar a dinâmica, escolha um programa, um *site*, uma rede social ou um aplicativo que você goste devido à sua **interatividade**. Em seguida, pesquise a pessoa (ou pessoas) que estiveram por trás do seu *front-end* e compartilhe o que encontrou com o restante da sala!



# O que é um *back-end*?

É um especialista em desenvolvimento de *software* que se concentra nas partes “invisíveis” de um aplicativo ou *site*, isto é, as partes que não são visíveis aos usuários, mas são essenciais para o seu funcionamento.

Imagine que um *site* está sofrendo erros constantes no carregamento de sua página. A tarefa do *back-end* é de investigar o erro e diagnosticar a causa do problema.



# Principais responsabilidades do *back-end*

**Desenvolvimento de servidores:** criar e manter os servidores que hospedam os aplicativos ou *sites*, configurando o ambiente de servidor, escolhendo sistemas operacionais e garantindo que os servidores estejam configurados.

**Lógica de negócios:** desenvolver a lógica de negócios dos aplicativos, escrevendo códigos que processam solicitações do cliente, executam cálculos complexos, tomam decisões baseadas em dados e interagem com sistemas externos.

**Gerenciamento de banco de dados:** projetar e manter o banco de dados que armazena informações críticas do aplicativo, como dados de usuários, conteúdo do *site* e registros de transações.

**APIs (Interfaces de Programação de Aplicativos):** criar APIs que permitam a comunicação entre o *front-end* e o *back-end*;

**Segurança:** garantir a segurança do aplicativo implementando medidas de proteção contra ameaças, como autenticação de usuários, criptografia de dados e prevenção de ataques cibernéticos e proteção.

# Quais conhecimentos um *back-end* deve ter?

Veja só o que é necessário:

1

Proficiência em linguagens de programação usadas no desenvolvimento *back-end*, como Python, Java, Ruby, PHP, C# e Node.js;

3

Conhecimento de *frameworks back-end* populares, como Django (Python), Ruby on Rails (Ruby), Spring (Java), Laravel (PHP), Express.js (Node.js), entre outros;

2

Compreensão de **bancos de dados relacionais** (como MySQL, PostgreSQL, SQL Server) e **não relacionais** (como MongoDB, Redis, Cassandra);

4

Habilidade para projetar, construir e documentar APIs RESTful ou GraphQL para a comunicação entre o *back-end* e o *front-end*.

## Relação entre *back-end* e *front-end*

Algumas pessoas utilizam a metáfora do *iceberg* para explicar o trabalho de um *front-end* e de um *back-end*.

Aqui, a metáfora destaca a parte visível (*front-end*) e a parte invisível (*back-end*) de uma página web.



## Média salarial de um *back-end*

Segundo o site **talent.com**, que baseou-se em 695 salários do Brasil, um *back-end* ganha, em média:

R\$5.000/mês.



# Bloco 3

---

E o *full stack*?

# Vamos recapitular?

Ainda que o *back-end* e o *front-end* trabalhem em conjunto, algumas responsabilidades estão designadas a cada um deles.

Observe as descrições abaixo e indique se é a função de um *back-end* ou de um *front-end*.

Monitorar o desempenho do servidor

Análise de banco de dados

Otimização de imagens e recursos

Integração com outros navegadores

Avaliar o carregamento da página

Analizar erro em produtos específicos do site

# Gabarito

E aí, acertou?

## *Back-end*

Monitorar o desempenho do servidor

Análise de banco de dados

Analizar erro em produtos específicos do *site*

## *Front-end*

Otimização de imagens e recursos

Avaliar o carregamento da página

Integração com outros navegadores

## Profissional *full stack*

É alguém que possui conhecimento e habilidades em todas as camadas de desenvolvimento de aplicativos, incluindo o *front-end* e o *back-end*.



# Benefícios de ser *full stack*

## Fique ligado!



**Versatilidade:** é capaz de trabalhar em todas as partes de um projeto.



**Autossuficiência:** pode assumir uma ampla gama de responsabilidades.



**Compreensão completa:** útil para entender um ciclo de vida de *software*.



**Comunicação facilitada:** capacidade de se relacionar com ambas as áreas.



**Desenvolvimento de protótipos rápidos:** pode desenvolver ideias funcionais rapidamente.



**Liderança de projetos:** consegue liderar projetos do início ao fim.



**Maior empregabilidade:** alta valorização pelas empresas.



**Adaptação às mudanças:** tende a estar em constante evolução.

## Média salarial de um *full stack*

Segundo o site [talent.com](#), que baseou-se em 1316 salários do Brasil, um *full stack* ganha, em média:

R\$5.250/mês.



# Bloco 4

---

Vamos explorar?

# Descobrindo o meu caminho

A partir do que foi abordado até então, vocês deverão desenvolver um **plano de carreira** para si mesmos.

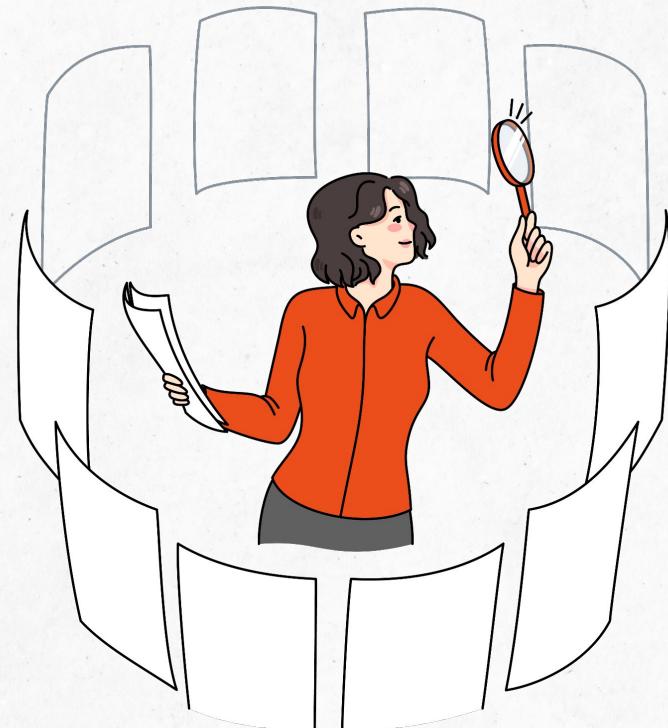
Esse plano consiste em analisar três perguntas fundamentais e, em seguida, realizar um exercício de autorreflexão para descobrir o que mais se assemelha com o seu perfil.

Em seguida, compartilhe com o restante da sala o que você descobriu e os seus motivos que te fizeram escolher a área.

Qual área parece mais alinhada com os meus interesses e as minhas habilidades atuais?

Quais tarefas ou responsabilidades de cada área eu acho mais atraentes?

Eu me sinto confortável em aprender e trabalhar com várias tecnologias?



## Discussões e reflexões



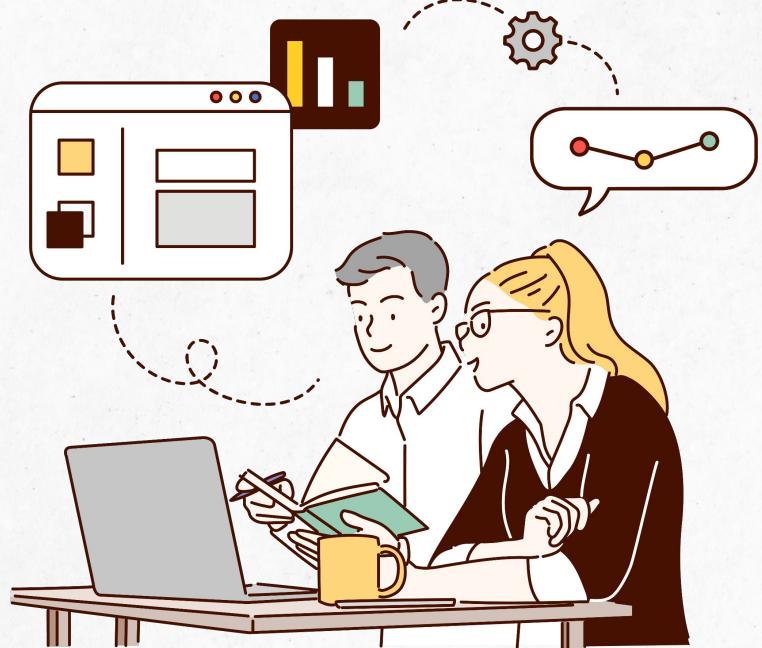
# Bloco 5

---

Descobrindo algumas plataformas de *software*.

# *Softwares*

Os softwares desempenham um papel fundamental na vida dos profissionais de Tecnologia da Informação (TI) e são de grande importância em diversas dimensões do trabalho deles.



# Node.js

## O que é

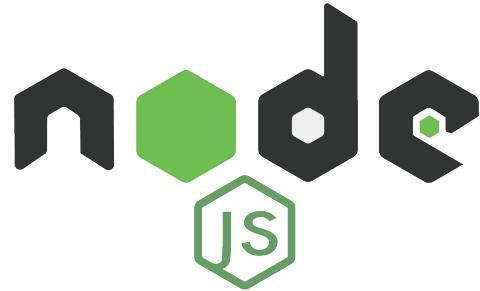
É uma plataforma de aplicação que permite que os desenvolvedores usem JavaScript no lado do servidor para criar aplicativos web e serviços.

## Características

É conhecido por ser assíncrono e orientado a eventos, o que o torna altamente eficiente para operações de I/O intensivas, como redes e acesso a bancos de dados.

## Diferencial

Uma das principais características do Node.js é o uso do JavaScript como linguagem de desenvolvimento, tanto no lado do cliente quanto no do servidor.



# Vue.js

## O que é

É um *framework* progressivo para a construção de interfaces de usuário. Você pode usá-lo para criar componentes de interface em partes do seu aplicativo, sem a necessidade de reconstruí-lo completamente.

## Características

Permite que os desenvolvedores criem componentes reutilizáveis e gerenciem o estado da interface do usuário de maneira eficaz.

## Diferencial

É fácil de integrar com outros projetos ou bibliotecas.



# Angular.js

## O que é

É um *framework* JavaScript desenvolvido pelo Google. Ele é usado para construir aplicativos da *web* de página única (SPA).

## Características

É conhecido por seu sistema de ligação de dados bidirecional, o que permite que as alterações no modelo de dados sejam refletidas automaticamente na interface do usuário e vice-versa.

## Diferencial

É um projeto de código aberto, o que significa que a comunidade de desenvolvedores pode contribuir para seu desenvolvimento, sua atualização e as suas correção de erros.



ANGULARJS

# Bloco 6

---

Hora de praticar!

## Vamos exercitar?

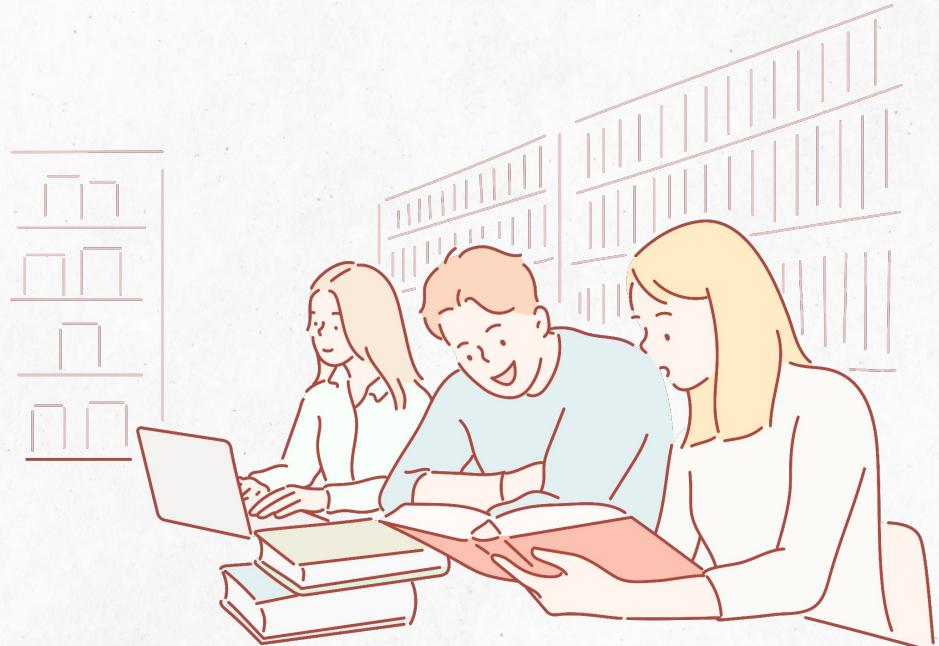
Separem-se em três grupos. Cada equipe deve ficar designado para um dos temas: **Node.js, Angular.js e Vue.js**.

Em seguida, o(a) professor(a) apresentará uma situação de caso. Os alunos devem defender qual seria o melhor ambiente de desenvolvimento para lidar com o caso.

Cada grupo terá 20 minutos para fazer a pesquisa. Depois, terão cinco minutos para defender os seus argumentos.

Preparados?

Uma empresa está querendo desenvolver um *site* de compras *on-line*. Qual é o *software* mais indicado para ser utilizado nesse caso?

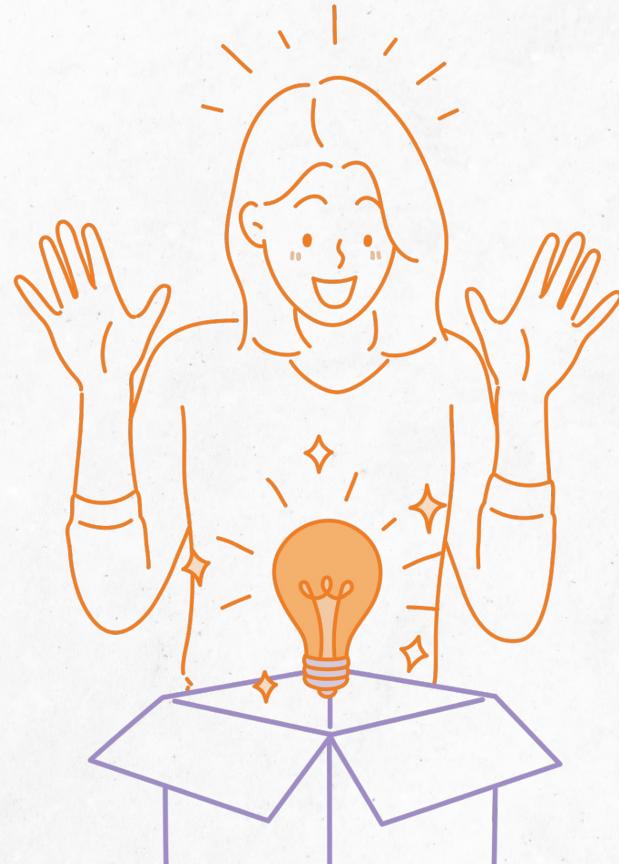


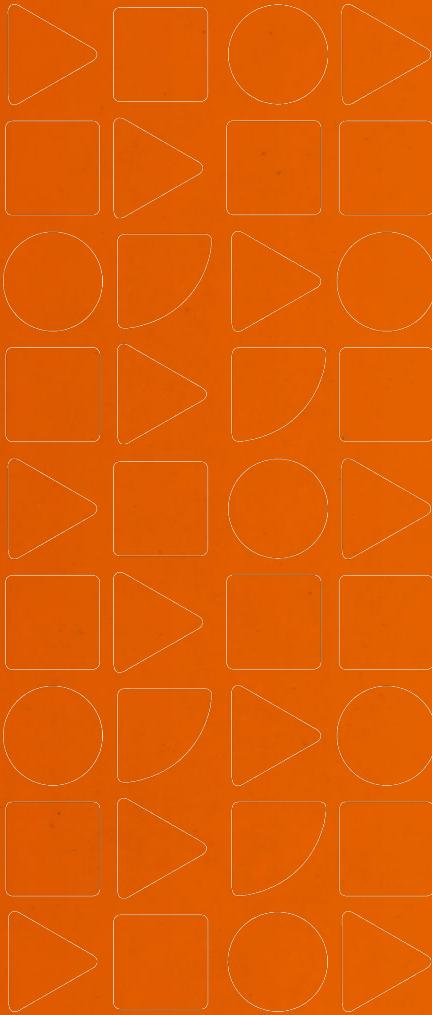
# Hora do debate argumentativo!

Apresentem os seus argumentos!



E agora? O que você sabe sobre  
áreas da programação?





# Referências Bibliográficas

PROZ EDUCAÇÃO. *Apostila de Desenvolvimento para Dispositivos Móveis I.* 2023.