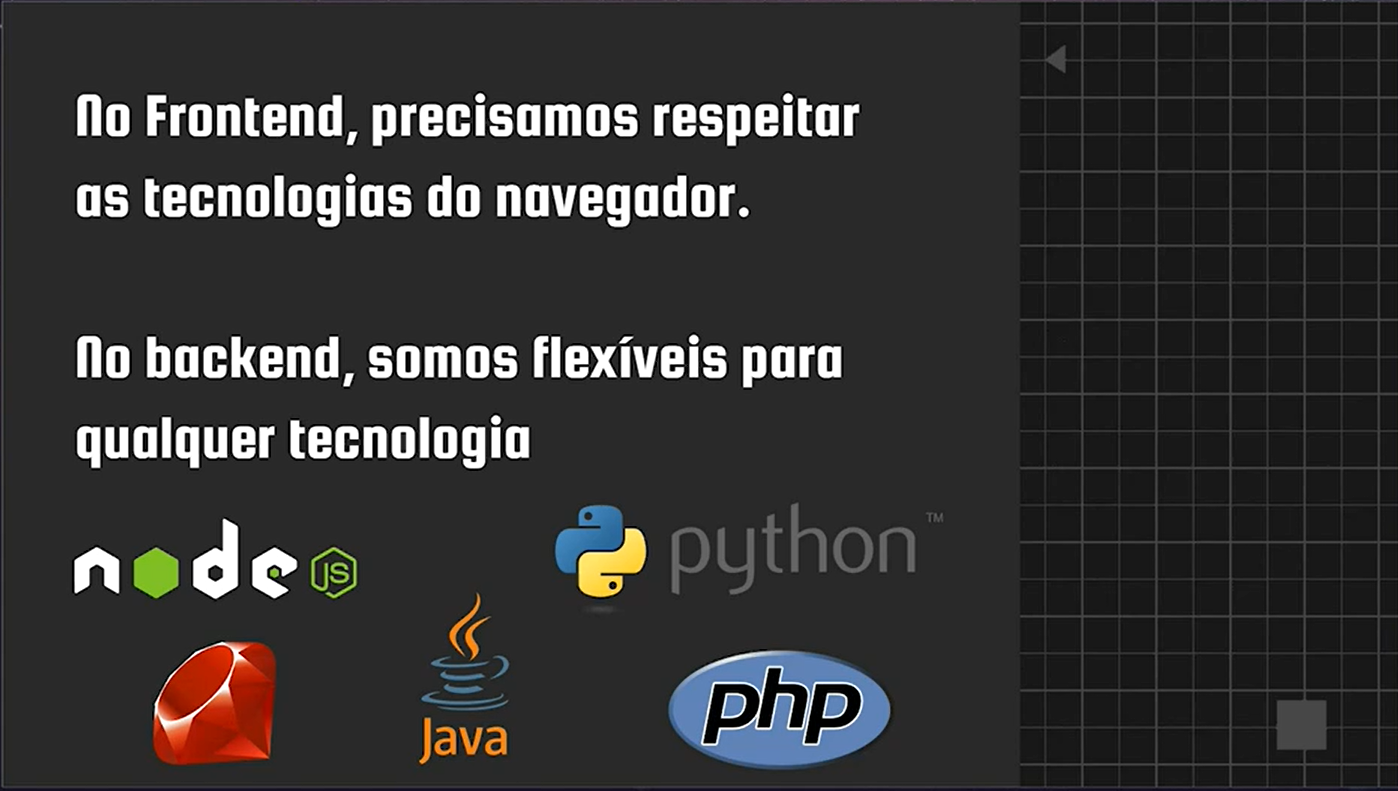
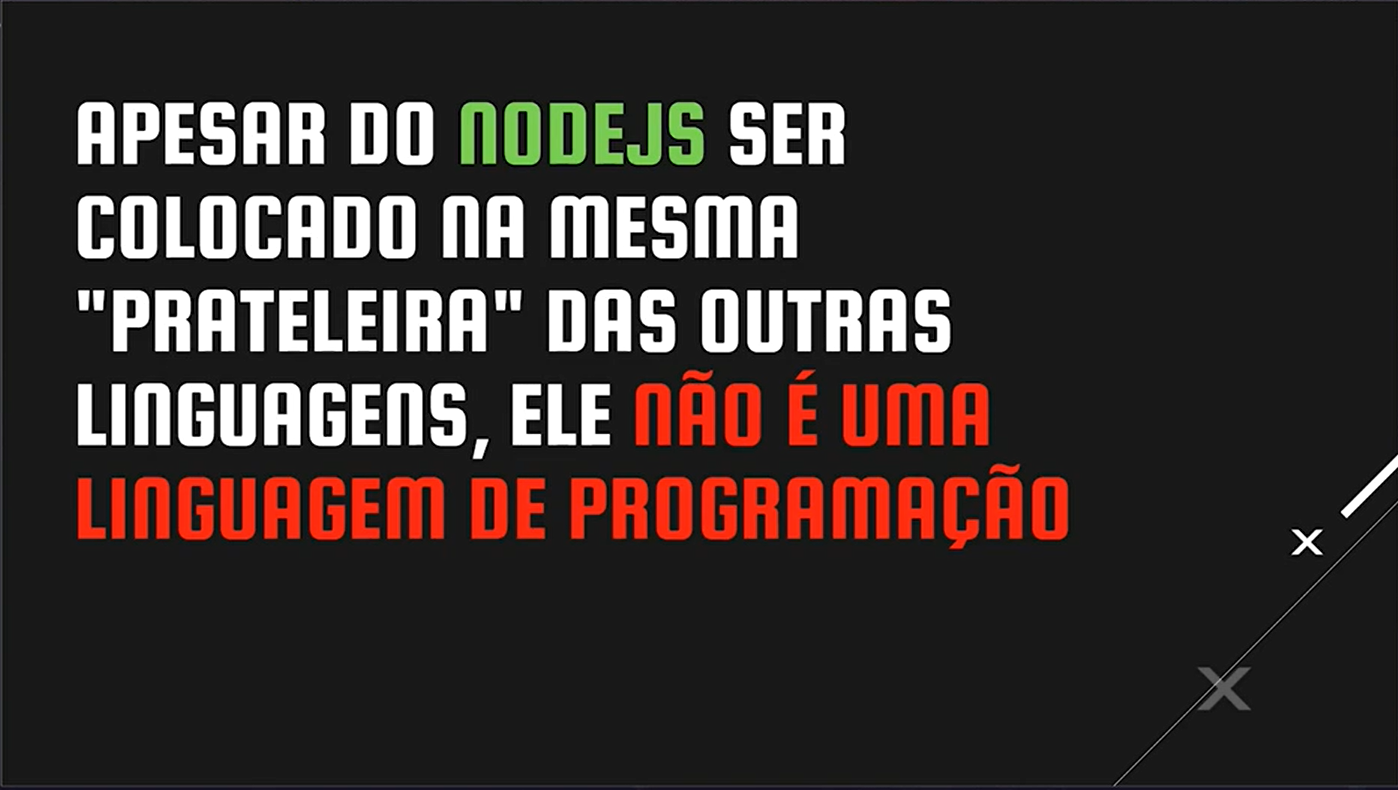
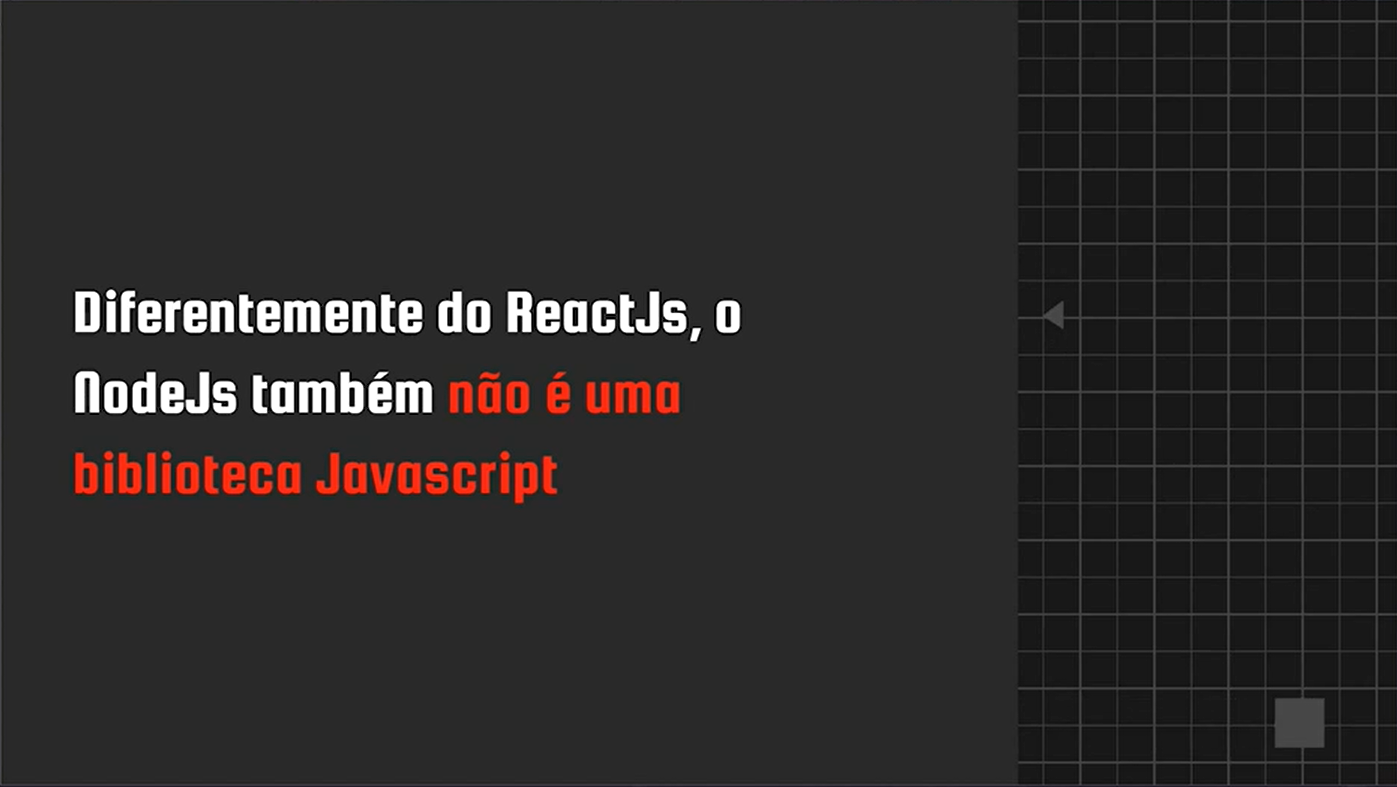
**NODE JS**

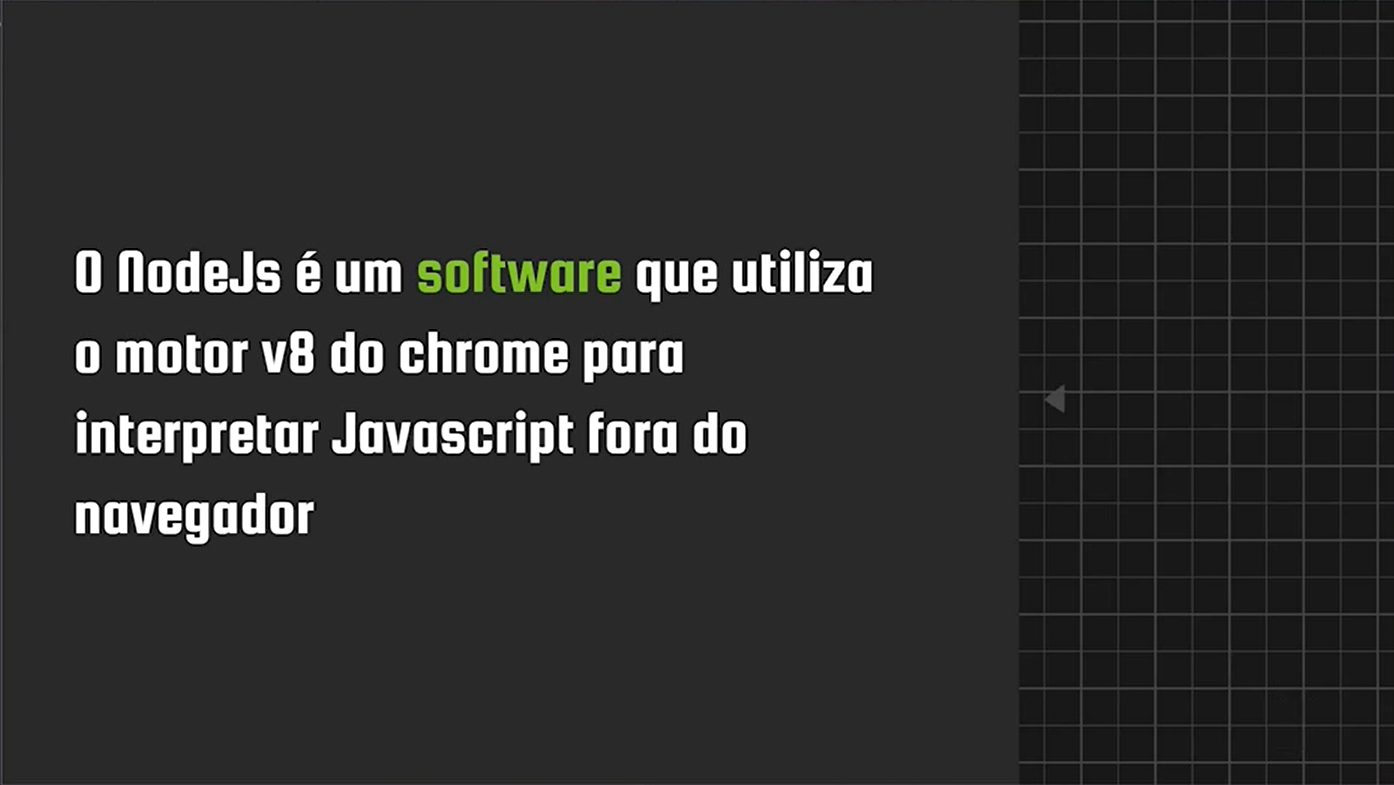
****

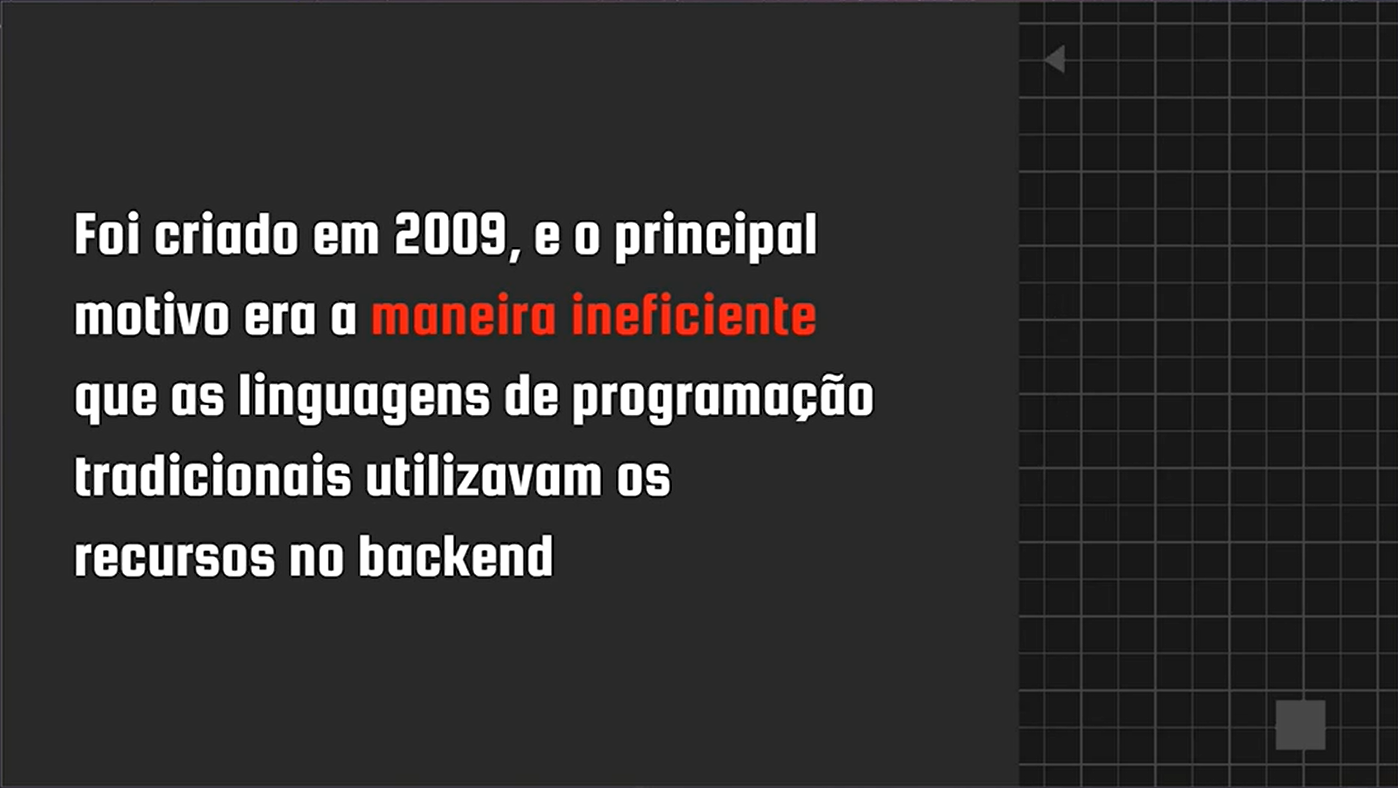
****

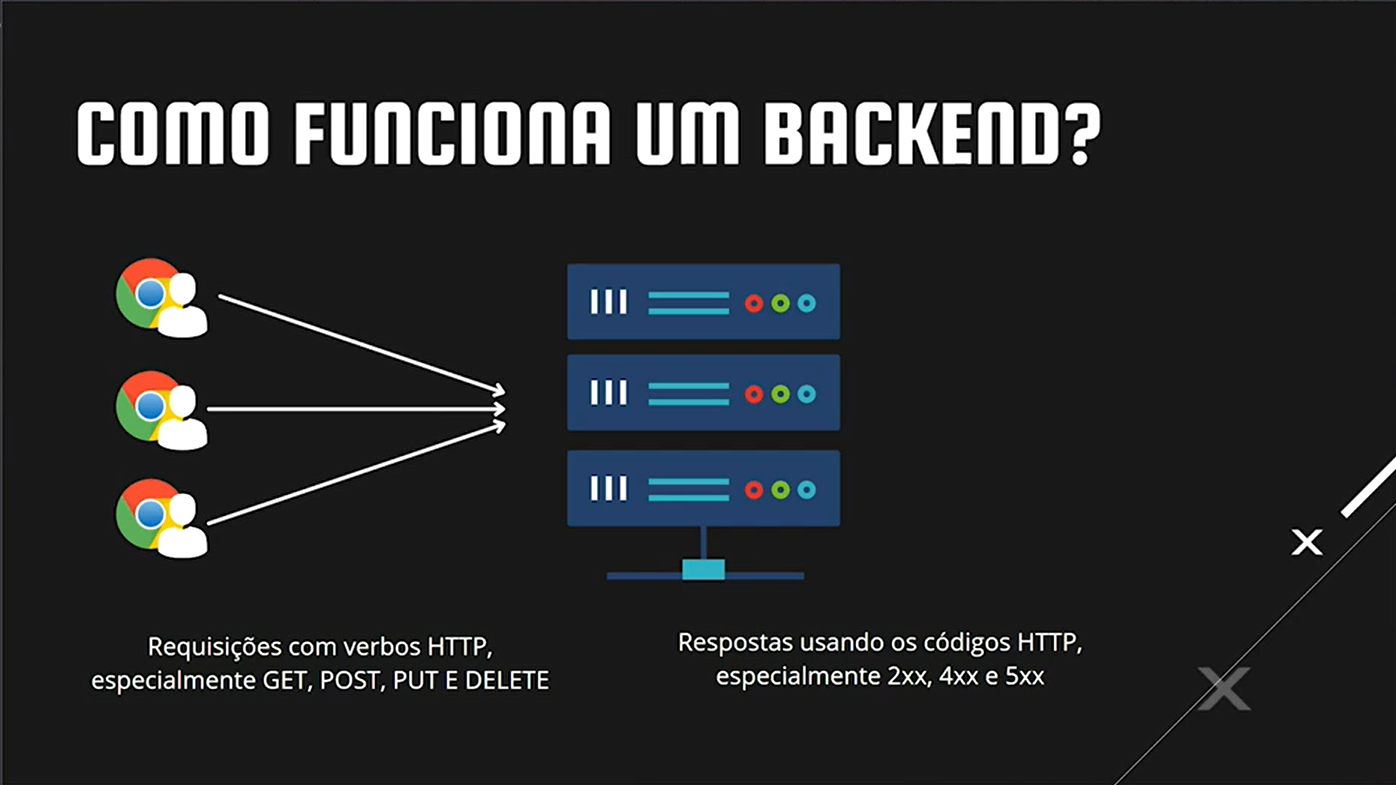
****

****

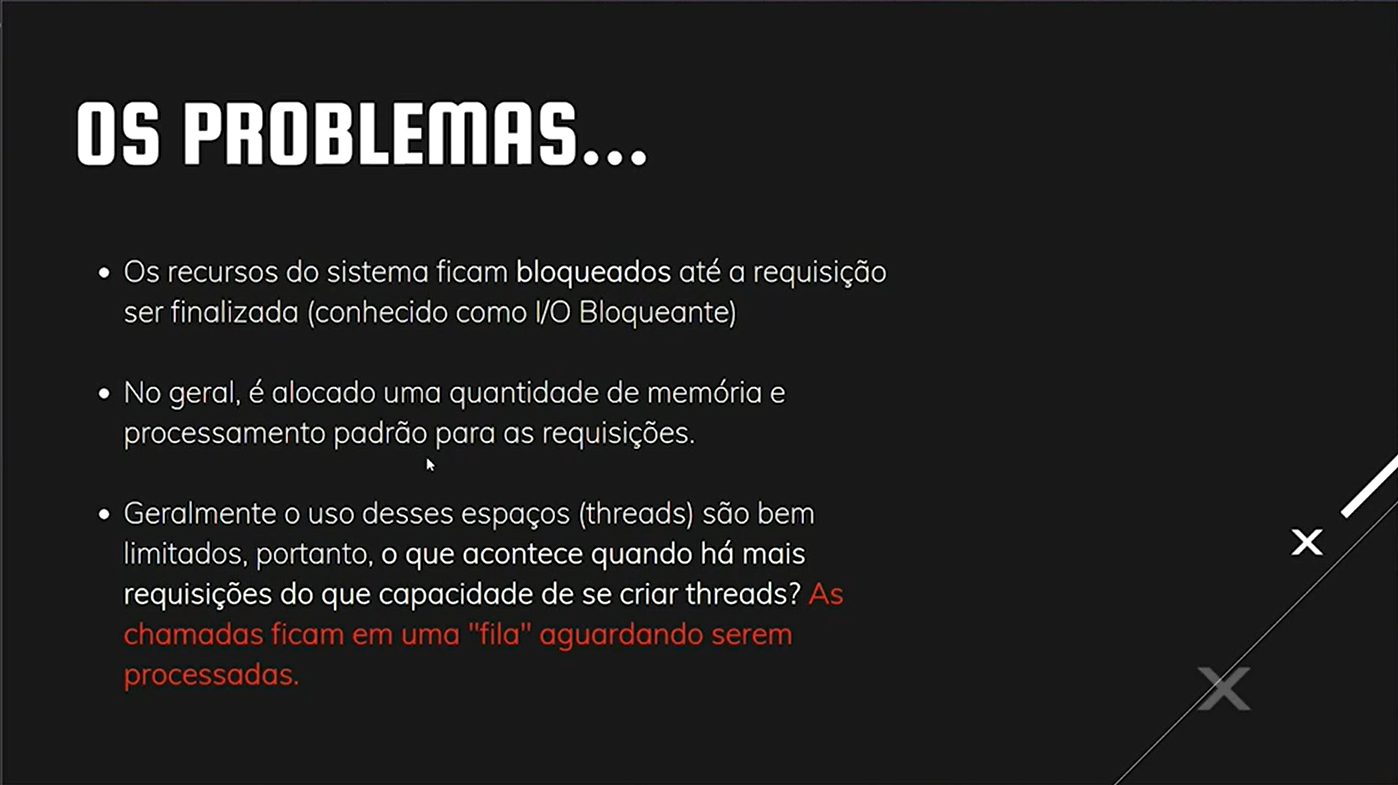
****

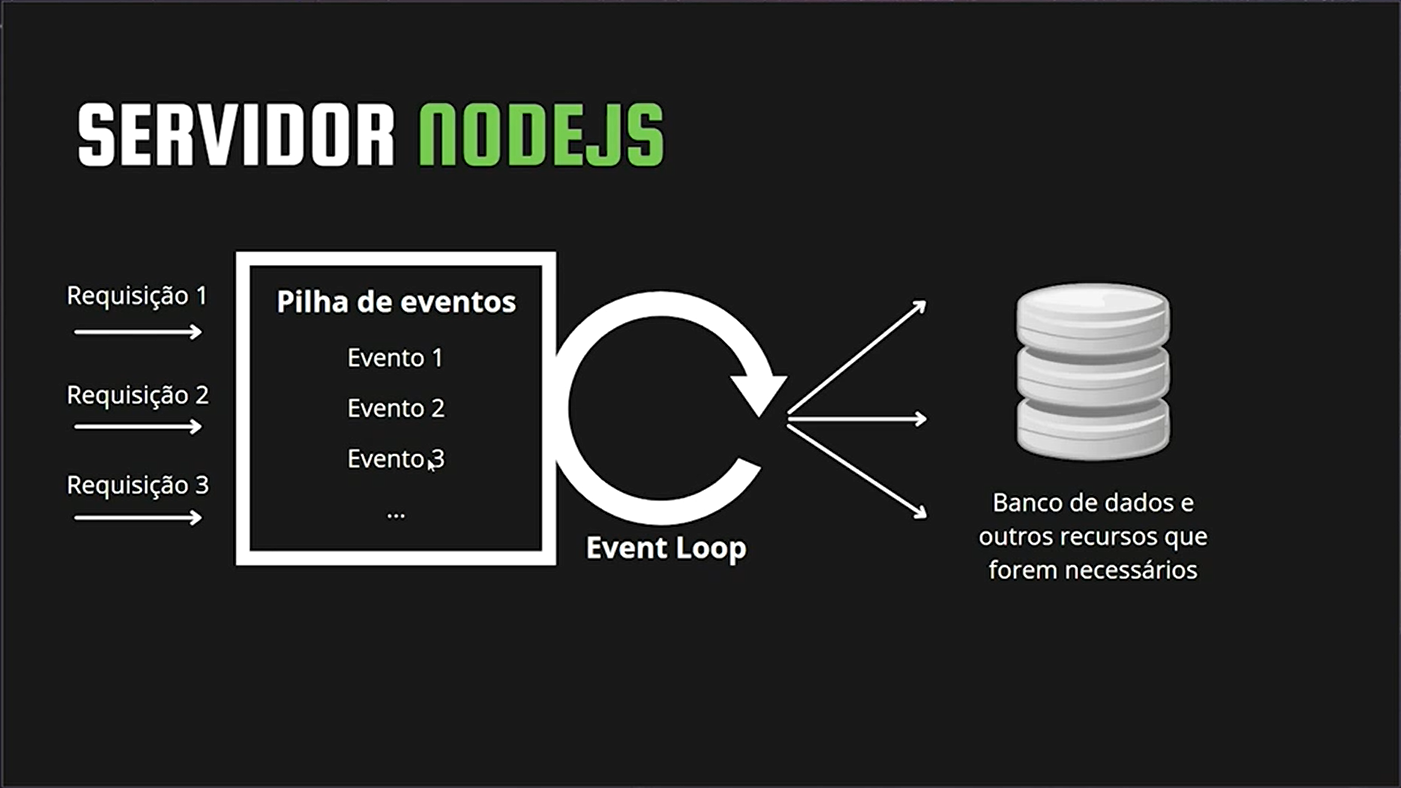
****

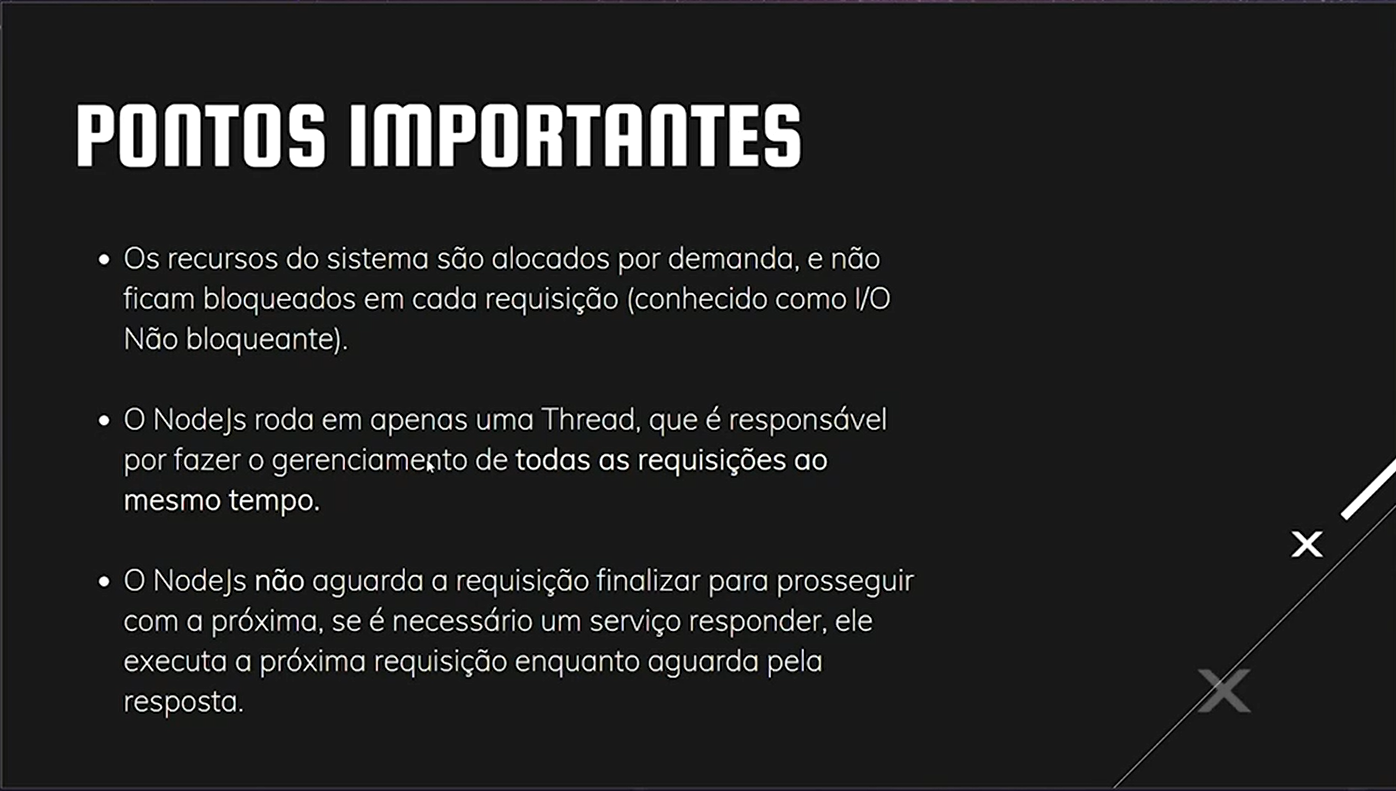
****

****

****

****

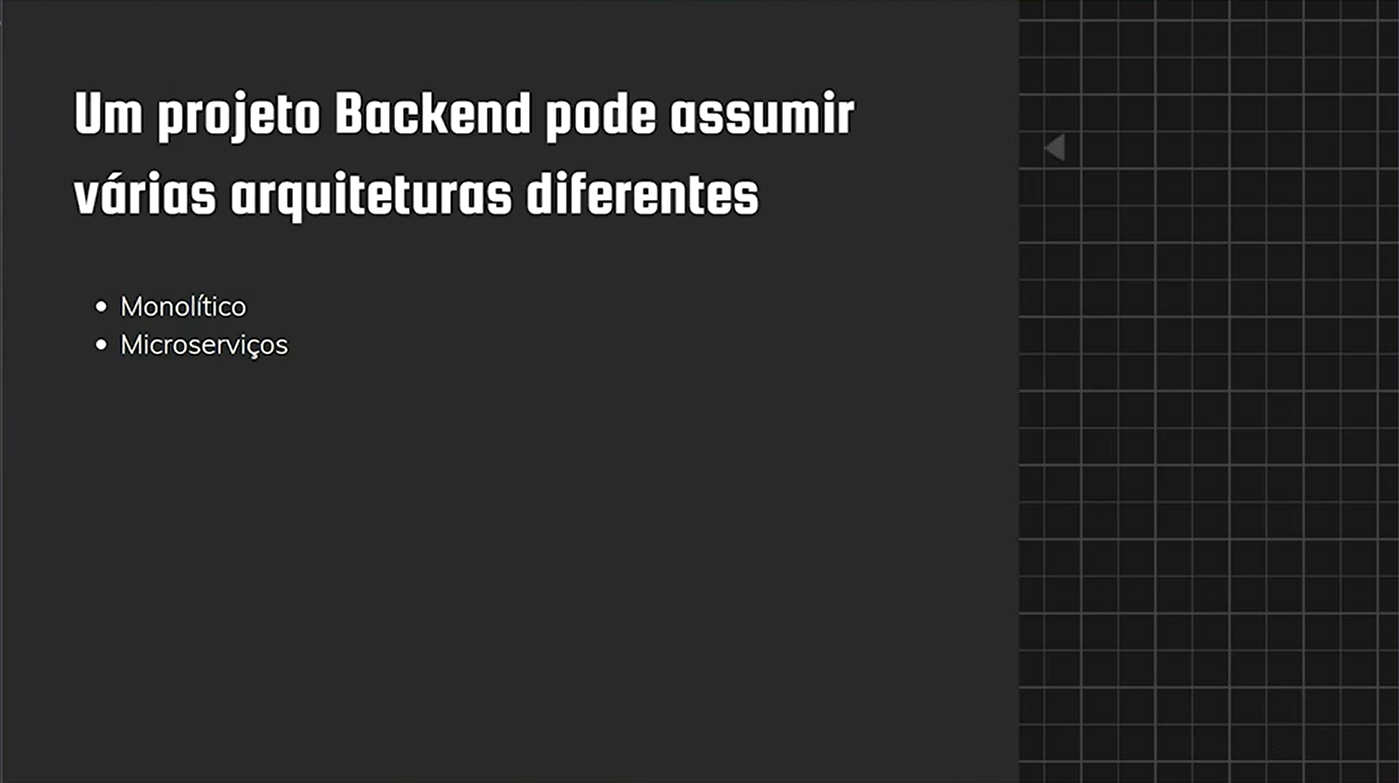
****

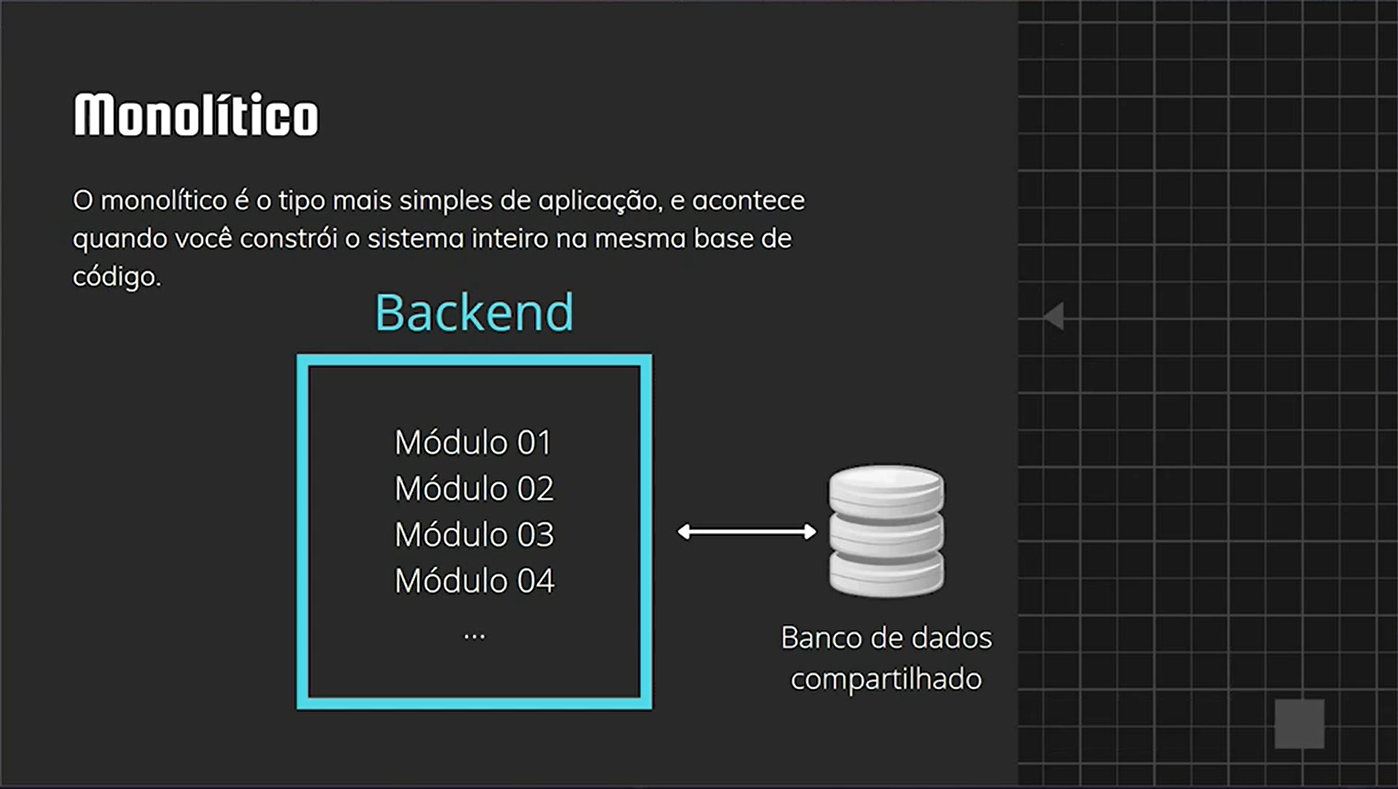
****

****

****

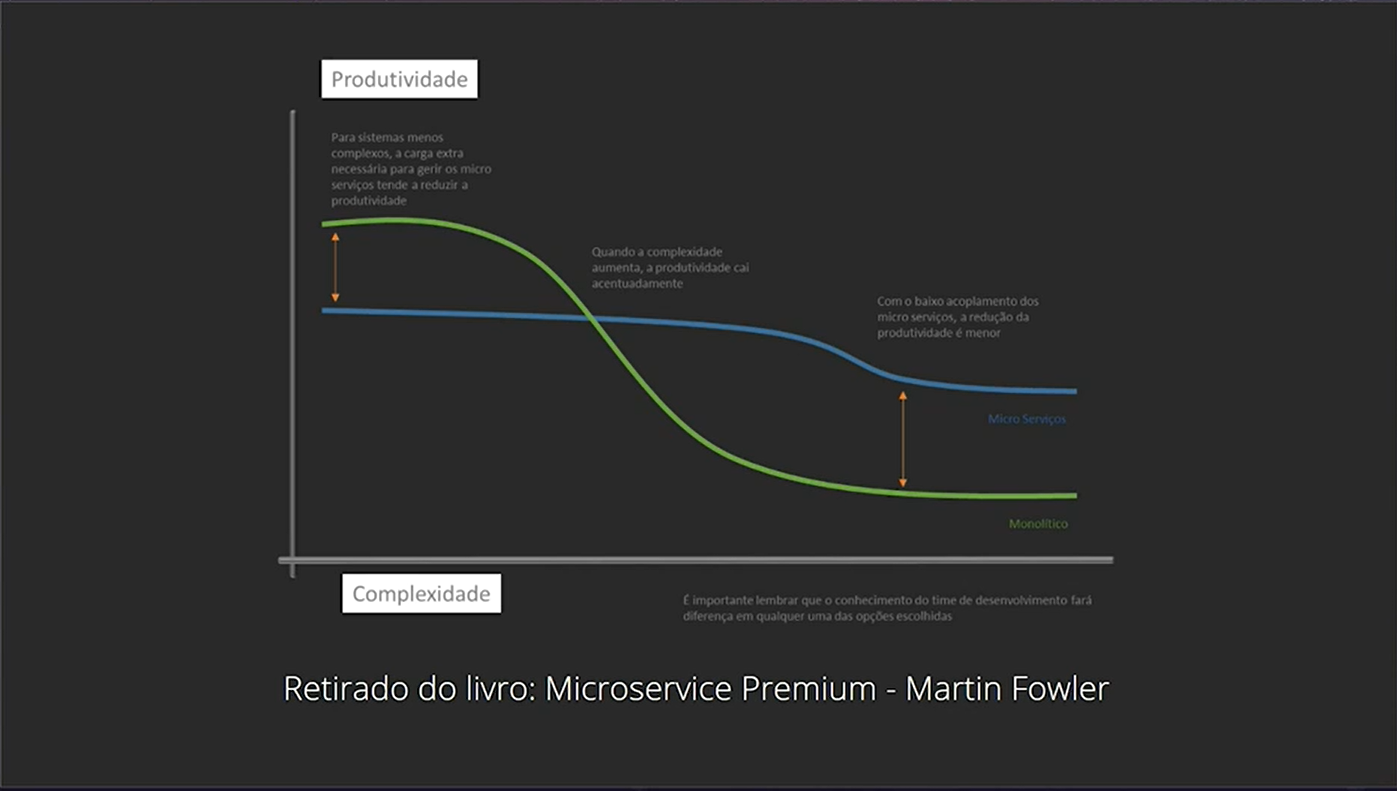
****

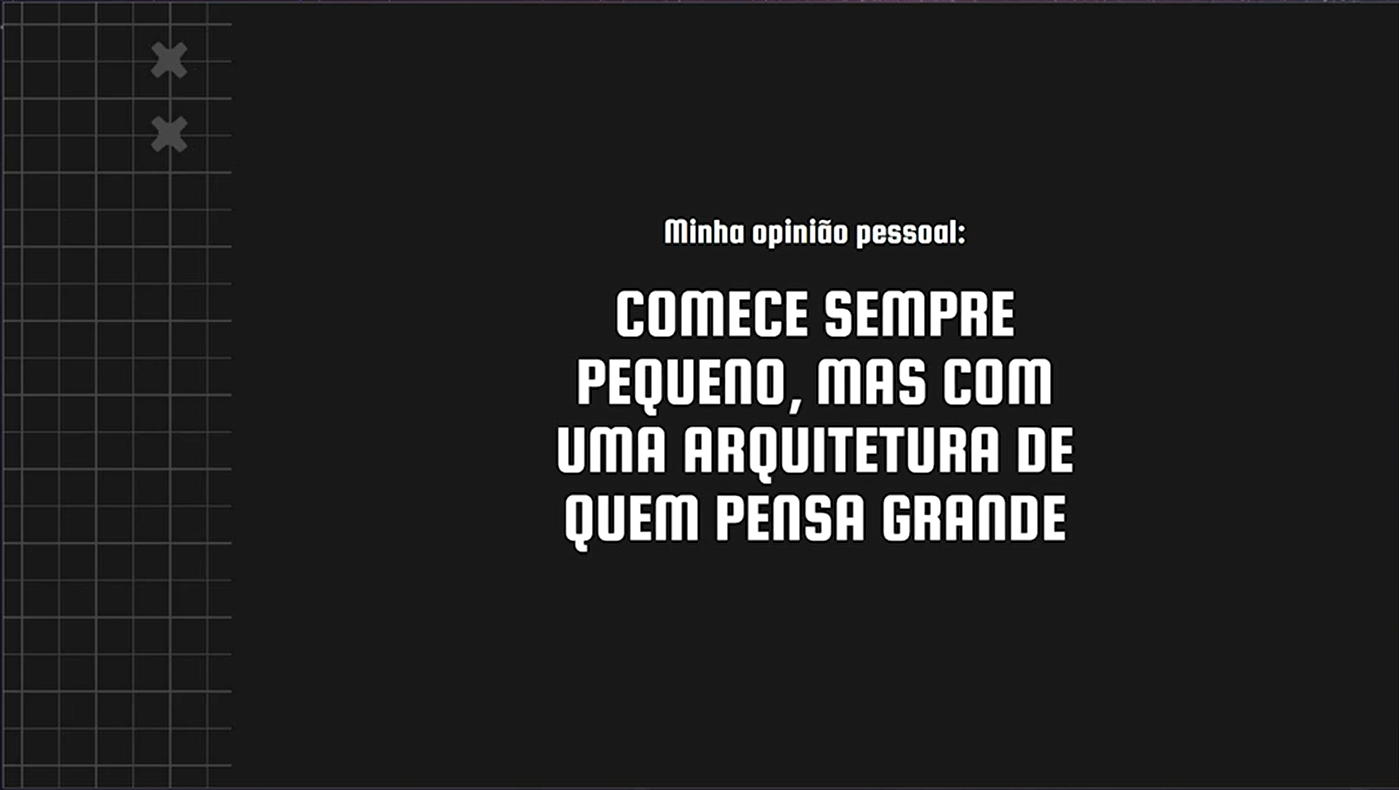
****

****

****

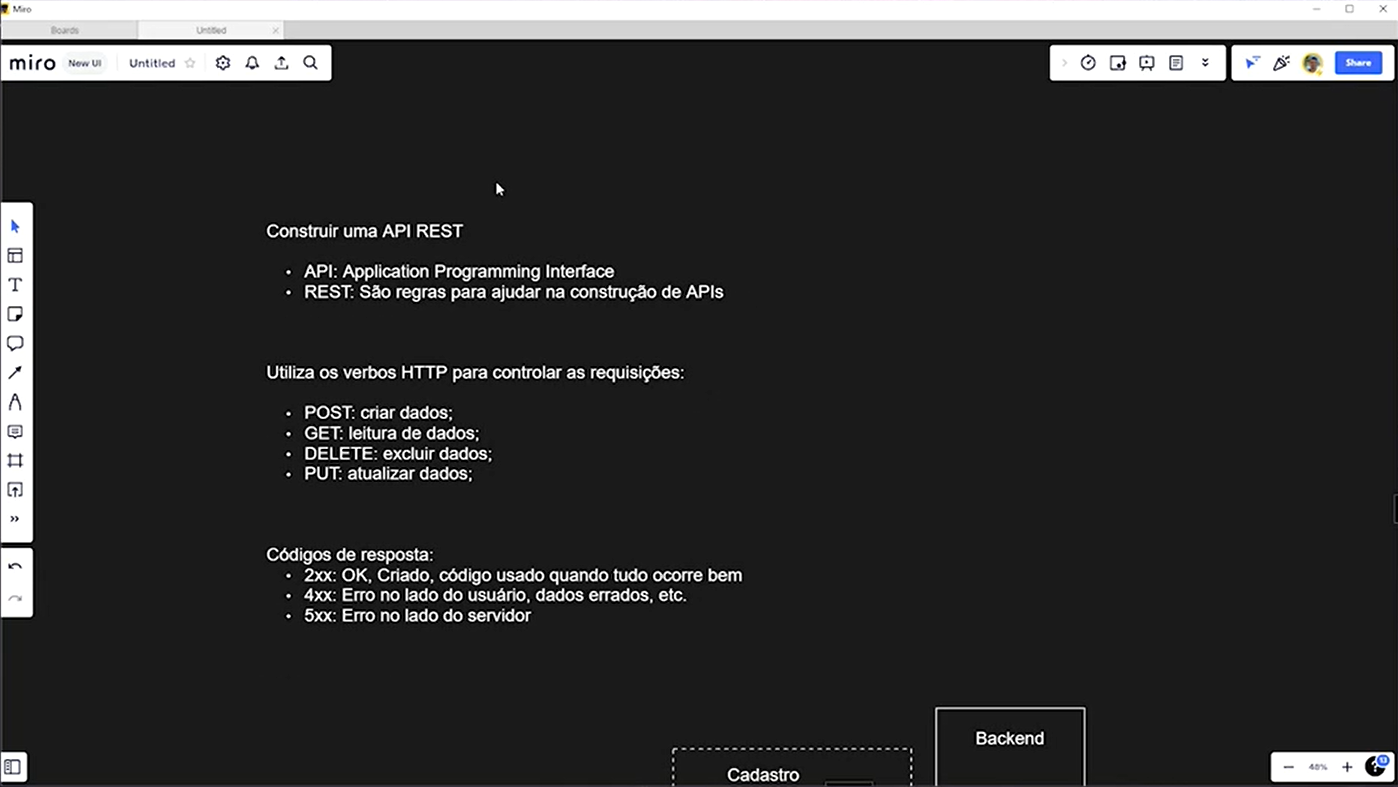
****

****

****

****

Resumo do módulo: **CONSTRUÇÃO DE UM BACKEND**

****

O que é uma API?

APIs são mecanismos que permitem que dois componentes de software se comuniquem usando um conjunto de definições e protocolos. Por exemplo, o sistema de software do instituto meteorológico contém dados meteorológicos diários. O aplicativo meteorológico em seu telefone “fala” com este sistema por meio de APIs e mostra atualizações meteorológicas diárias no telefone.

O que significa API?

API significa Application Programming Interface (Interface de Programação de Aplicação). No contexto de APIs, a palavra Aplicação refere-se a qualquer software com uma função distinta. A interface pode ser pensada como um contrato de serviço entre duas aplicações. Esse contrato define como as duas se comunicam usando solicitações e respostas. A documentação de suas respectivas APIs contém informações sobre como os desenvolvedores devem estruturar essas solicitações e respostas.

Como as APIs funcionam?

A arquitetura da API geralmente é explicada em termos de cliente e servidor. A aplicação que envia a solicitação é chamada de cliente e a aplicação que envia a resposta é chamada de servidor. Então, no exemplo do clima, o banco de dados meteorológico do instituto é o servidor e o aplicativo móvel é o cliente.

Existem quatro maneiras diferentes pelas quais as APIs podem funcionar, dependendo de quando e por que elas foram criadas.

APIs SOAP

Essas APIs usam o Simple Object Access Protocol (Protocolo de Acesso a Objetos Simples). Cliente e servidor trocam mensagens usando XML. Esta é uma API menos flexível que era mais popular no passado.

APIs RPC

Essas APIs são conhecidas como Remote Procedure Calls (Chamadas de Procedimento Remoto). O cliente conclui uma função (ou um procedimento) no servidor e o servidor envia a saída de volta ao cliente.

APIs WebSocket

[API Websocket](https://docs.aws.amazon.com/apigateway/latest/developerguide/apigateway-websocket-api-overview?pg=wianapi&cta=websocketapi) é outro desenvolvimento de API da Web moderno que usa objetos JSON para passar dados. Uma API WebSocket oferece suporte à comunicação bidirecional entre aplicativos cliente e o servidor. O servidor pode enviar mensagens de retorno de chamada a clientes conectados, tornando-o mais eficiente que a API REST.

APIs REST

Essas são as APIs mais populares e flexíveis encontradas na Web atualmente. O cliente envia solicitações ao servidor como dados. O servidor usa essa entrada do cliente para iniciar funções internas e retorna os dados de saída ao cliente. Vejamos as APIs REST em mais detalhes abaixo.

O que são APIs REST?

REST significa Transferência Representacional de Estado. REST define um conjunto de funções como GET, PUT, DELETE e assim por diante, que os clientes podem usar para acessar os dados do servidor. Clientes e servidores trocam dados usando HTTP.

A principal característica da [API REST](https://docs.aws.amazon.com/apigateway/latest/developerguide/http-api-vs-rest?pg=wianapi&cta=restapi) é a ausência de estado. A ausência de estado significa que os servidores não salvam dados do cliente entre as solicitações. As solicitações do cliente ao servidor são semelhantes aos URLs que você digita no navegador para visitar um site. A resposta do servidor corresponde a dados simples, sem a renderização gráfica típica de uma página da Web.

O que é API Web?

Uma API Web ou API de serviço da Web é uma interface de processamento de aplicações entre um servidor da Web e um navegador da Web. Todos os serviços da Web são APIs, mas nem todas as APIs são serviços da Web. A API REST é um tipo especial de API Web que usa o estilo de arquitetura padrão explicado acima.

Os diferentes termos que abrangem as APIs, como API Java ou APIs de serviço, existem porque, historicamente, as APIs foram criadas antes da World Wide Web. As APIs Web modernas são APIs REST e os termos podem ser usados de forma intercambiável.

O que são integrações de API?

As integrações de API são componentes de software que atualizam automaticamente os dados entre clientes e servidores. Alguns exemplos de integrações de API são quando os dados automáticos são sincronizados com a nuvem por meio da galeria de imagens do seu telefone ou a data e a hora são sincronizadas automaticamente no seu laptop quando você viaja para um local com outro fuso horário. As empresas também podem usá-las para automatizar com eficiência muitas funções do sistema.

Quais são os benefícios das APIs REST?

As APIs REST oferecem quatro principais benefícios:

1. Integração

As APIs são usadas para integrar novas aplicações com sistemas de software existentes. Isso aumenta a velocidade de desenvolvimento porque cada funcionalidade não precisa ser escrita do zero. Você pode usar APIs para aproveitar o código existente.

2. Inovação

Setores inteiros podem mudar com a chegada de uma nova aplicação. As empresas precisam responder rapidamente e oferecer suporte à rápida implantação de serviços inovadores. Elas podem fazer isso fazendo alterações no nível da API sem precisar reescrever todo o código.

3. Expansão

As APIs apresentam uma oportunidade única para as empresas atenderem às necessidades de seus clientes em diferentes plataformas. Por exemplo, a API de mapas permite a integração de informações de mapas por meio de sites, Android, iOS etc. Qualquer empresa pode fornecer acesso semelhante aos seus respectivos bancos de dados internos usando APIs gratuitas ou pagas.

4. Facilidade de manutenção

A API atua como um gateway entre dois sistemas. Cada sistema é obrigado a fazer alterações internas para que a API não seja afetada. Dessa forma, qualquer alteração futura de código feita por uma parte não afetará a outra parte.

Quais são os diferentes tipos de API?

As APIs são classificadas de acordo com sua arquitetura e escopo de uso. Já exploramos os principais tipos de arquiteturas de API, agora, vamos dar uma olhada no escopo de uso.

APIs privadas

Elas são internos a uma empresa e são usadas apenas para conectar sistemas e dados dentro da empresa.

APIs públicas

Estas são abertas ao público e podem ser usadas por qualquer pessoa. Pode ou não haver alguma autorização e custo associado a esses tipos de APIs.

APIs de parceiros

Estas são acessíveis apenas por desenvolvedores externos autorizados para auxiliar as parcerias entre empresas.

APIs compostas

Estas combinam duas ou mais APIs distintas para atender a requisitos ou comportamentos complexos do sistema.

O que é um endpoint de API e por que ele é importante?

Os endpoints da API são os pontos de contato finais no sistema de comunicação da API. Estes incluem URLs de servidores, serviços e outros locais digitais específicos de onde as informações são enviadas e recebidas entre sistemas. Os endpoints da API são fundamentais para as empresas por dois motivos principais:

1. Segurança

Os endpoints da API tornam o sistema vulnerável a ataques. O monitoramento da API é crucial para impedir o uso indevido.

2. Performance

Os endpoints da API, especialmente os de alto tráfego, podem causar gargalos e afetar a performance do sistema.

Como proteger uma API REST?

Todas as APIs devem ser protegidas por meio de autenticação e monitoramento adequados. As duas principais maneiras de proteger APIs REST incluem:

1. Tokens de autenticação

Eles são usados para autorizar os usuários a fazer a chamada de API. Os tokens de autenticação verificam se os usuários são quem afirmam ser e se têm direitos de acesso para aquela chamada de API específica. Por exemplo, quando você faz login em seu servidor de e-mail, seu cliente de e-mail usa tokens de autenticação para acesso seguro.

2. Chaves de API

As chaves de API verificam o programa ou a aplicação que faz a chamada de API. Elas identificam a aplicação e garantem que ela tenha os direitos de acesso necessários para fazer a chamada de API específica. As chaves de API não são tão seguras quanto os tokens, mas permitem o monitoramento da API para coletar dados sobre o uso. Você pode ter notado uma longa sequência de caracteres e números no URL do seu navegador ao visitar diferentes sites. Essa string é uma chave de API que o site usa para fazer chamadas de API internas.

Como criar uma API?

Diligência prévia e esforços são necessários para criar uma API com a qual outros desenvolvedores desejem trabalhar e na qual queiram confiar. Estas são as cinco etapas necessárias para o design de uma API de alta qualidade:

1. Planejar a API

As especificações da API, como OpenAPI, fornecem o esquema para o design da sua API. É melhor pensar em diferentes casos de uso com antecedência e garantir que a API esteja de acordo com os padrões atuais de desenvolvimento de API.

2. Criar a API

 Os designers de APIs prototipam APIs usando código boilerplate. Depois que o protótipo é testado, os desenvolvedores podem personalizá-lo de acordo com as especificações internas.

3. Testar a API

O teste de API é o mesmo que o teste de software e deve ser feito para evitar bugs e defeitos. As ferramentas de teste de API podem ser usadas para testar a resistência da API contra ataques cibernéticos.

4. Documentar a API

Embora as APIs sejam autoexplicativas, a documentação da API funciona como um guia para melhorar a usabilidade. APIs bem documentadas que oferecem uma série de funções e casos de uso tendem a ser mais populares em uma arquitetura orientada a serviços.

5. Comercializar a API

Assim como a Amazon é um marketplace online para varejo, existem marketplaces de API para desenvolvedores comprarem e venderem outras APIs. Catalogar sua API pode permitir que você ganhe dinheiro com ela.

O que é teste de API?

As estratégias de teste de API são semelhantes a outras metodologias de teste de software. O principal foco é validar as respostas do servidor. O teste de API inclui:

* Fazer várias solicitações a endpoints de API para testes de performance.
* Escrever testes de unidade para verificar a lógica de negócios e a correção funcional.
* Testar a segurança simulando ataques ao sistema.

Como escrever a documentação da API?

Escrever uma documentação de API abrangente faz parte do [processo de gerenciamento de API](https://aws.amazon.com/pt/api-gateway/api-management/?pg=wianapi&cta=apimgtprcs). A documentação da API pode ser gerada automaticamente usando ferramentas ou escrita manualmente. Algumas práticas recomendadas incluem:

* Escrever explicações em inglês simples e fácil de ler. Documentos gerados por ferramentas podem se tornar prolixos e exigir edição.
* Usar exemplos de código para explicar a funcionalidade.
* Fazer a manutenção da documentação para que seja precisa e atualizada.
* Visar o estilo de escrita para iniciantes
* Abranger todos os problemas que a API pode solucionar para os usuários.

Como usar uma API?

As etapas para implementar uma nova API incluem:

1. Obter uma chave de API. Isso é feito criando uma conta verificada com o provedor de API.
2. Configure um cliente de API HTTP. Essa ferramenta permite estruturar solicitações de API facilmente usando as chaves de API recebidas.
3. Se você não tiver um cliente de API, tente estruturar a solicitação por conta própria em seu navegador consultando a documentação da API.
4. Quando estiver familiarizado com a nova sintaxe da API, você poderá começar a usá-la em seu código.

Onde posso encontrar novas APIs?

Novas APIs Web podem ser encontradas em marketplaces de APIs e diretórios de APIs. Os marketplaces de API são plataformas abertas nas quais qualquer pessoa pode catalogar uma API para venda. Os diretórios de APIs são repositórios controlados e regulamentados pelo proprietário do diretório. Designers de API experientes podem avaliar e testar uma nova API antes de adicioná-la ao diretório.

Alguns sites de API populares incluem:

* RapidAPI: o maior mercado global de APIs com mais de 10.000 APIs públicas e 1 milhão de desenvolvedores ativos no local. O RapidAPI permite que os usuários testem APIs diretamente na plataforma antes de fazer a compra.
* Public APIs: a plataforma agrupa APIs remotas em 40 categorias de nicho, tornando mais fácil navegar e encontrar a API certa para atender às suas necessidades.
* APIForThat e APIList: ambos os sites têm listas de mais de 500 APIs Web, juntamente com informações detalhadas sobre como usá-las.

O que é um gateway de API?

Um API Gateway é uma ferramenta de gerenciamento de APIs para clientes empresariais que usam uma grande variedade de serviços de backend. Os gateways de API geralmente lidam com tarefas comuns, como autenticação de usuários, estatísticas e gerenciamento de taxas que são aplicáveis a todas as chamadas de API.

O [Amazon API Gateway](https://aws.amazon.com/pt/api-gateway/?pg=wianapi&cta=amzapigtwy) é um serviço totalmente gerenciado que permite que desenvolvedores criem, publiquem, mantenham, monitorem e protejam APIs em qualquer escala com facilidade. Ele lida com todas as tarefas envolvidas na aceitação e no processamento de milhares de chamadas de API simultâneas, incluindo gerenciamento de tráfego, suporte a CORS, autorização e controle de acesso, controle de utilização, monitoramento e gerenciamento de versão de API.

O que é GraphQL?

GraphQL é uma linguagem de consulta desenvolvida especificamente para APIs. Ela prioriza fornecer aos clientes exatamente os dados que eles solicitam e nada mais. A linguagem foi projetada para tornar as APIs rápidas, flexíveis e amigáveis ao desenvolvedor. Como alternativa ao REST, o GraphQL oferece aos desenvolvedores de frontend a capacidade de consultar vários bancos de dados, microsserviços e APIs com um único endpoint de GraphQL. As organizações optam por criar APIs com o GraphQL porque isso as ajuda a desenvolver aplicações mais rapidamente. [Leia mais sobre o GraphQL aqui](https://aws.amazon.com/graphql/).

O AWS AppSync é um serviço totalmente gerenciado que facilita o desenvolvimento de APIs GraphQL ao lidar com o trabalho pesado de conectar-se com segurança a fontes de dados como AWS DynamoDB, AWS Lambda e muito mais. O AWS AppSync pode enviar atualizações de dados em tempo real por Websockets a milhões de clientes. Para aplicativos móveis e aplicações Web, o AppSync também fornece acesso a dados locais quando os dispositivos ficam offline. Depois de implantado, o AWS AppSync aumenta e reduz automaticamente a escala na vertical do mecanismo de execução da API GraphQL para atender aos volumes de solicitação da API.

Como obter serviços de API da Amazon?

O gerenciamento da Interface de Programação de Aplicação é uma parte importante do desenvolvimento de software moderno. Vale a pena investir em infraestrutura de API, incluindo ferramentas, gateway e arquitetura de microsserviços para usuários internos e externos.

O [Amazon API Gateway](https://aws.amazon.com/pt/api-gateway/?pg=wianapi&cta=amzapugateway) inclui uma [gama completa de recursos](https://aws.amazon.com/pt/api-gateway/features/?pg=wianapi&cta=fullrangefeatures) para gerenciar várias APIs simultaneamente e com eficiência. Você pode fazer até um milhão de chamadas de API gratuitamente [cadastrando-se no Portal da AWS](https://aws.amazon.com/pt/api-gateway/pricing/?pg=wianapi&cta=awssignupportal).

O [AWS AppSync](https://aws.amazon.com/pt/appsync/?pg=wianapi&cta=awsappsync) oferece configuração, administração e manutenção de API GraphQL totalmente gerenciadas, com infraestrutura sem servidor de alta disponibilidade integrada. Você paga apenas pelo que usa, sem taxas mínimas ou uso obrigatório de serviços. Para começar, faça login no console do AWS AppSync.

**Vídeo 05 –** Para criação de qualquer projeto node, devemos sempre começar pelos comandos:

**npm init -y** (cria o arquivo package.json)

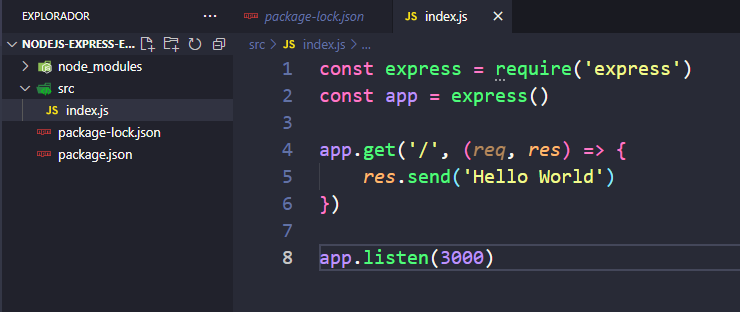
**git init** (inicia o repositório git)

**npm install** [**express@4.17.2**](mailto:express@4.17.2)(baixa a biblioteca express)

Documentação do express: https://www.npmjs.com/package/express

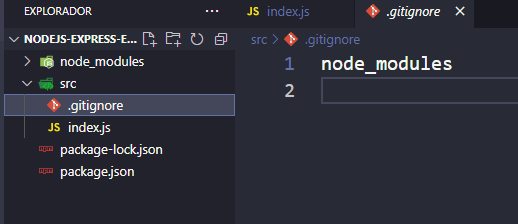
****

Criação de pasta **src**, arquivo **index.js** e primeiros comandos:

****

O navegador sempre usa o verbo **get** para fazer requisições ao **backend**.

Criar arquivo **gitignore** e colocar o arquivo **node\_modules** nele para não subir para o **github**

****

Trocar a **branch** para **main**: **git branch -M main**

Para criar um novo **commit** devemos iniciar um novo repositório no **github:** [**https://github.com/new**](https://github.com/new)

Copiar link do github para incluir arquivos e envio para o repositório:

*git remote add origin https://github.com/MarcosPassos-git/nodejs-express-exemplos.git*

Criar o **commit** clicando no **change**, no maizinho, renomear o arquivo, clicar **control** + **enter**