

Web API

Davi Augusto Neves Leite 191027383
Luiz Fernando Sementille 191021032



1.

O que é uma API?

O que é?

- API é um acrônimo de **Application Programming Interface** ou, em português, Interface de Programação de Aplicações;
- Representa um conjunto de definições e protocolos usado no desenvolvimento e na integração de software de aplicações;
- Geralmente está associada a uma função específica e que sua construção é de alta complexidade;

Como surgiu?

- O surgimento de API se deu para resolver dois problemas: o aumento da presença e uso de aplicações acessadas pela Web e a necessária atualização constante de dados dessas aplicações;
- Necessidade de trocar informações com provedores de dados que resolvem problemas específicos;

Políticas de Lançamento/Tipos de API

- **API's de parceiros:** A API é compartilhada com parceiros de negócios específicos. Isso pode fornecer fluxos de receita adicionais sem comprometer a qualidade;
- **API's públicas:** A API é disponibilizada para todos. Terceiros podem desenvolver aplicações que interajam com a sua API e isso pode se tornar uma fonte de inovação;
- **API's privadas:** A API é usada apenas internamente. Isso oferece às empresas um maior controle;

Quais os benefícios do seu uso?

- Permite com que os desenvolvedores criem funcionalidades complexas mais facilmente;
- Abstraem o código mais complexo, proporcionando o uso de sintaxes mais simples em seu lugar;
- Permite que a solução ou serviço se comunique com outros produtos e serviços sem precisar saber como eles foram implementados;

Quais os benefícios do seu uso?

- As empresas não precisam perder seu tempo e dinheiro desenvolvendo essas funcionalidades, sendo que elas já existem - o famoso “Reinventar a roda”;

E quais os problemas?

- O principal problema está em como a empresa proprietária da API a manuseia:
 - Ela pode desligar a API cujo o software estava inteira ou parcialmente dependente dela;
 - Ela pode querer monetizar o produto de forma que prejudique o software que a utiliza;



2.

O que é uma Web API?

O que é?

- É uma API que expõe suas funcionalidades sistêmicas através da internet e usando um conjunto de protocolos.
- Utiliza o modelo cliente-servidor;
- Pode ser desenvolvida utilizando diferentes tecnologias como Java e ASP.NET;

Protocolos

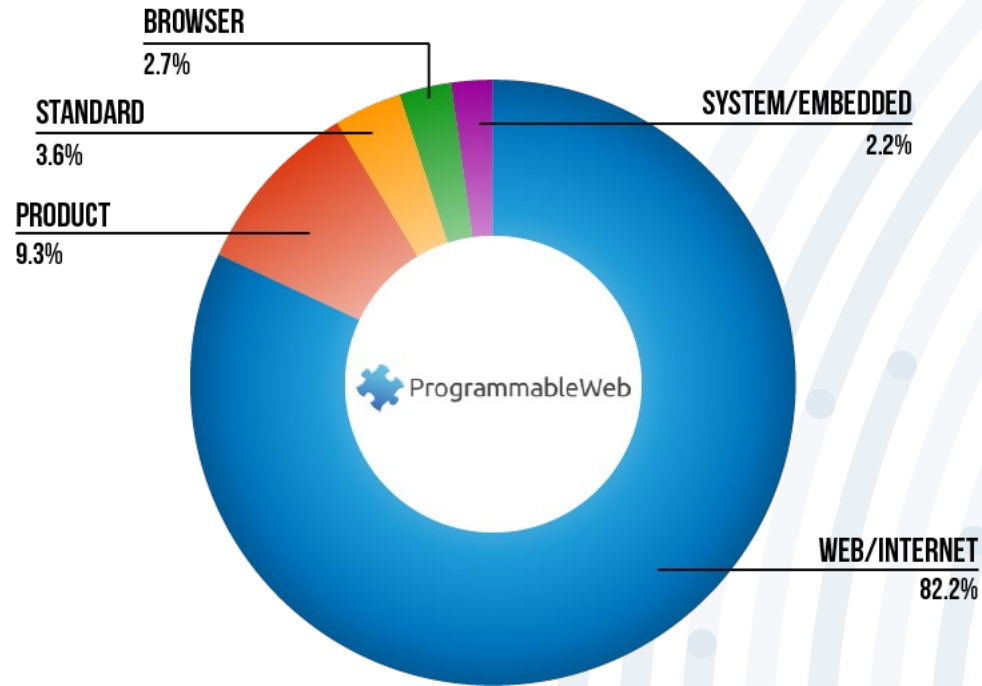
- O HTTP é o principal protocolo na obtenção e troca de recursos, como documentos HTML. Em outras palavras, é a base de qualquer troca de dados na Web e de um protocolo cliente-servidor (geralmente o TCP);
- Dessa forma, é possível o cliente enviar mensagens por meio de um navegador Web e o servidor atender e responder essas solicitações quando possível;
- Outros protocolos muito comuns são o SOAP e o REST, ambos utilizam o HTTP como base;

Como fazer uma requisição HTTP?

- Para mandar uma requisição HTTP é necessário indicar:
 - Protocolo de comunicação saiba, que no caso é o HTTP.
 - Host
 - Caminho dentro do servidor
 - Uma query (Opcional)
- Ao final, teremos uma requisição do seguinte formato:
`http://www.exemplo.com/api/exemplo.html?flag=true`

Presença: ProgrammableWeb API Directory

API TYPES RECORDED IN PROGRAMMABLEWEB DIRECTORY (NON-NULL VALUES)



Quais os benefícios do seu uso?

- São a melhor opção para criar serviços orientados a recursos;
- Possuem um ótimo desempenho e sua integração é simples;
- Suporta qualquer padrão de texto, como JSON (JavaScript Object Notation) e o XML (Extensible Markup Language);

Como usar?

- Escolher a Web API;
- Ler a documentação para conhecer os requisitos de acesso e algumas sintaxes;
- Ao tentar estabelecer conexão, a maioria das API's pedem uma chave de acesso;
- Construir uma URL de comunicação à partir da documentação da API;



3.

Paradigmas dos Web API's

API's Pedido-Resposta (Request-Response)

- As API's de Pedido-Resposta (ou Solicitação-Resposta) expõem, em sua grande maioria, uma interface por meio de um servidor Web baseado em HTTP, definindo o conjunto dos chamados endpoints;
- Os clientes fazem requisições HTTP de dados para esses endpoints e o servidor retorna as respostas geralmente em forma de JSON ou XML;
- Existência de três paradigmas: REST, RPC e GraphQL.

Paradigma REST

- Utilizado por empresas como Google, Twitter e GitHub;
- Mais popular para o desenvolvimento de API's;
- Inteiramente baseada nos **recursos** da aplicação: exposição de dados como recursos e utilização de métodos padrões de HTTP para CRUD;

Paradigma REST

- RESTful: capacidade de um sistema aplicar a arquitetura REST;
- Baseada em cinco princípios (disponíveis no artigo “O que é API REST” da empresa RedHat);
- Considerada mais fácil de usar do que um protocolo prescrito, sendo implementada em aplicações mobile ou IoT.

Paradigma RPC

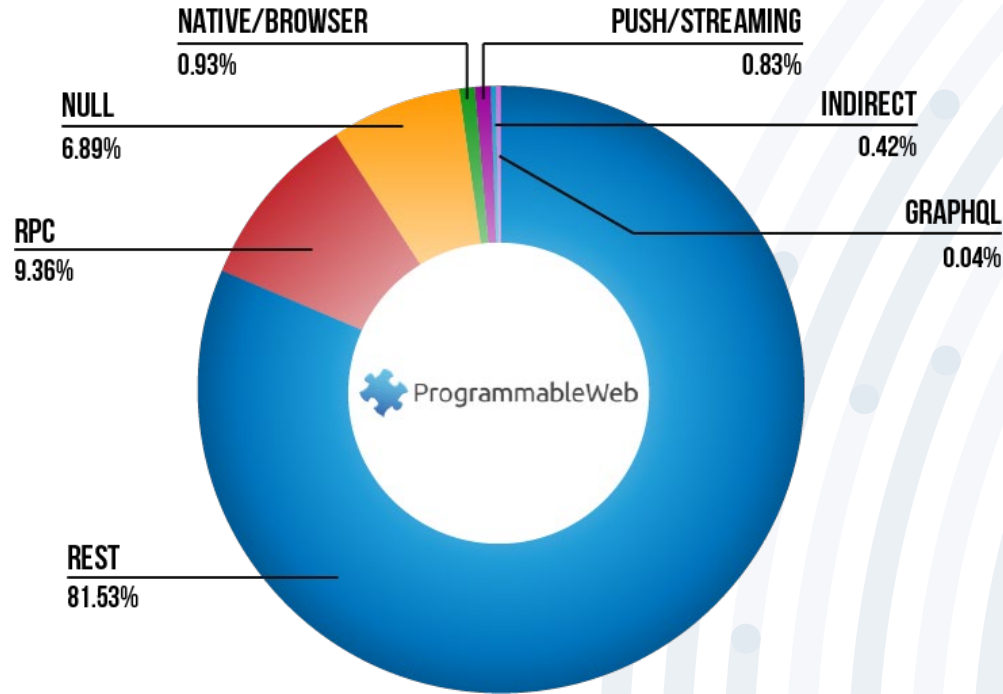
- Protocolo para execução remota de métodos: foco nas **ações** do que nos recursos (como no REST);
- Clientes passam um nome de método e seus argumentos para um servidor e recebem o processamento no formato JSON ou XML;
- Baseado em dois princípios: nome da operação e simplicidade nas requisições HTTP;
- Utilizada em API's em que possui métodos que podem ter complicações se utilizado o CRUD;

Paradigma GraphQL

- Linguagem de consulta adotada por provedores como Facebook, GitHub e Pinterest;
- O mais simples entre os paradigmas citados até então: permite ao cliente definir a estrutura de dados e o servidor, diante disso, deve retornar exatamente a estrutura desejada;
- Utilizado quando é necessário uma flexibilidade de consulta e uma manutenção consistente dos dados.

Presença: ProgrammableWeb API Directory

ARCHITECTURAL STYLES RECORDED IN PROGRAMMABLEWEB DIRECTORY



API's Dirigidas a Eventos (Event-Driven)

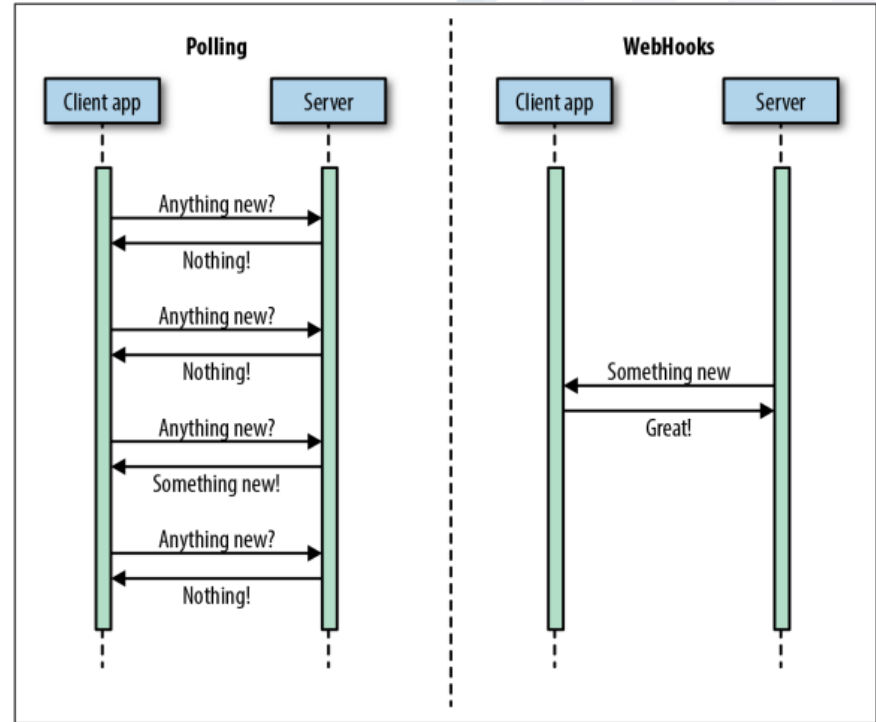
- São API's que foram desenvolvidas para suportar serviços com constante mudança de dados;
- Eliminam a necessidade do pooling e de se procurar pela melhor frequência;
- Existem 3 principais paradigmas associados: WebHooks, WebSockets e HTTP Streaming;

WebHooks

- É apenas uma URL que aceita mensagens HTTP POST, GET, PUT ou DELETE;
- Uma resposta HTTP será enviada para a URL configurada sempre que um evento acontecer;
- É de fácil implementação por parte do desenvolvedor da aplicação, basta criar um HTTP endpoint para receber os eventos;

WebHooks

- Utilizando o **Pooling** a aplicação do cliente 'procura' por dados novos de tempos em tempos;
- Utilizando **WebHooks** o próprio servidor envia a resposta sempre que algum dado novo estiver disponível;



Diferença entre o Pooling e o WebHooks

WebSockets

- É um protocolo usado para estabelecer um canal de comunicação full duplex sobre um único protocolo TCP (Transport Control Protocol);
- Permite que uma conexão criada entre o navegador e o servidor seja contínua e sem a necessidade de polling.
- É suportado pela maioria dos navegadores;
- Aplicações como Slack e o Trello suportam WebSockets;

WebSockets

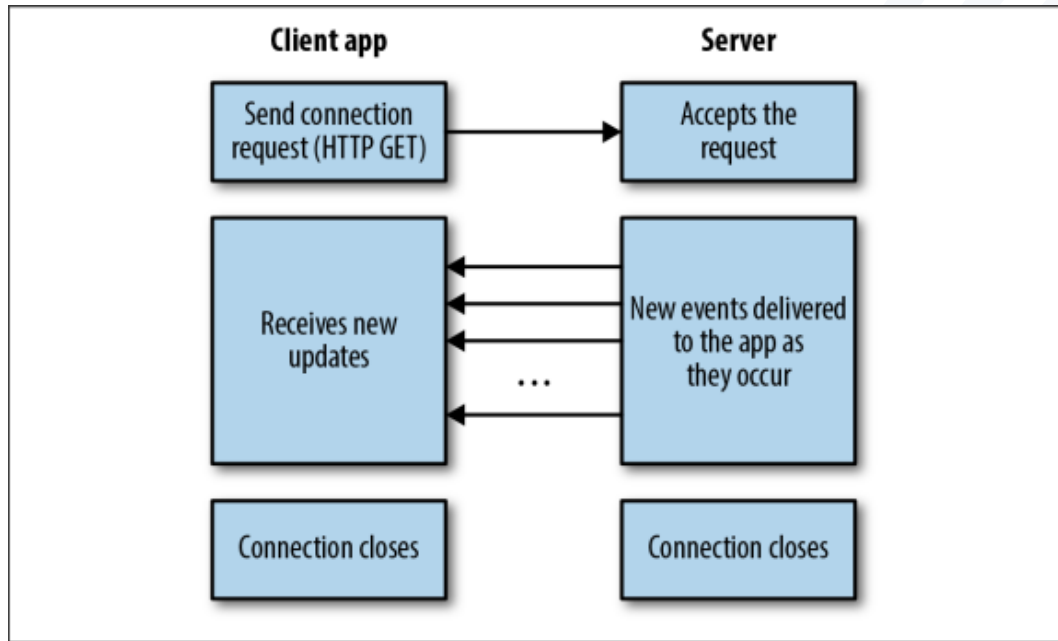
| | |
|---|--------------|
| ↑ {"type":"tickle","id":16392} | 28 18:12... |
| ↑ {"type":"typing","channel":"C0GEV71UG","id":16393} | 50 18:12... |
| ↑ {"type":"message","channel":"C0GEV71UG","text":"new message","id":16394} | 72 18:12... |
| ↓ {"ok":true,"reply_to":16394,"ts":"1519870324.000289","text":"new message"} | 74 18:12... |
| ↑ {"type":"typing","channel":"C0GEV71UG","id":16395} | 50 18:12... |
| ↑ {"type":"message","channel":"C0GEV71UG","text":"hi","id":16396} | 63 18:12... |
| ↓ {"ok":true,"reply_to":16396,"ts":"1519870325.000226","text":"hi"} | 65 18:12... |
| ↓ {"type":"channel_marked","channel":"C0GEV71UG","ts":"1519870325.000226","unread_count":0,"unread_count_disp... | 231 18:12... |
| ↑ {"type":"typing","channel":"C0GEV71UG","id":16397} | 50 18:12... |
| ↑ {"type":"message","channel":"C0GEV71UG","text":"Hey","id":16398} | 64 18:12... |
| ↓ {"ok":true,"reply_to":16398,"ts":"1519870326.000314","text":"Hey"} | 66 18:12... |
| ↑ {"type":"typing","channel":"C0GEV71UG","id":16399} | 50 18:12... |
| ↑ {"type":"message","channel":"C0GEV71UG","text":"what's up","id":16400} | 70 18:12... |
| ↓ {"ok":true,"reply_to":16400,"ts":"1519870328.000116","text":"what's up"} | 72 18:12... |
| ↓ {"type":"channel_marked","channel":"C0GEV71UG","ts":"1519870328.000116","unread_count":0,"unread_count_disp... | 231 18:12... |
| ↑ {"type":"ping","id":16401} | 26 18:12... |
| ↓ {"type":"pong","reply_to":16401} | 32 18:12... |
| ↓ {"type":"reaction_added","user":"U0H4TC2U8","item":{"type":"message","channel":"C0GEV71UG","ts":"1519870326.... | 227 18:12... |
| ↓ {"type":"pref_change","name":"emoji_use","value":{"train":1,"two":3,"one":2,"memo":1,"wave::skin-tone-3":3,... | 318 18:12... |
| ↓ {"type":"reaction_added","user":"U0H4TC2U8","item":{"type":"message","channel":"C0GEV71UG","ts":"1519870326.... | 227 18:12... |

Exemplo de informações sendo trocadas entre o Slack e o navegador

HTTP Streaming

- É um upgrade das API's pedido-resposta, pois o tamanho da resposta HTTP não é finito;
- Nesse mecanismo, o servidor pode continuar a mandar novos dados em uma única conexão de longo prazo aberta pelo cliente;
- É suportado pela maioria dos navegadores;
- É utilizado por aplicações como o Twitter e o Facebook;

HTTP Streaming



Interação Cliente- Servidor com uma API HTTP Streaming



4.

Web Services x Web API

Web Service: o que é?

- Solução utilizada na integração de sistemas e na comunicação entre aplicações diferentes (utilizando a tecnologia Web);
- Transferência de dados por meio de protocolos de comunicação (geralmente HTTP) e codificação em XML;
- Dependência de componentes como XML-RPC e SOAP;

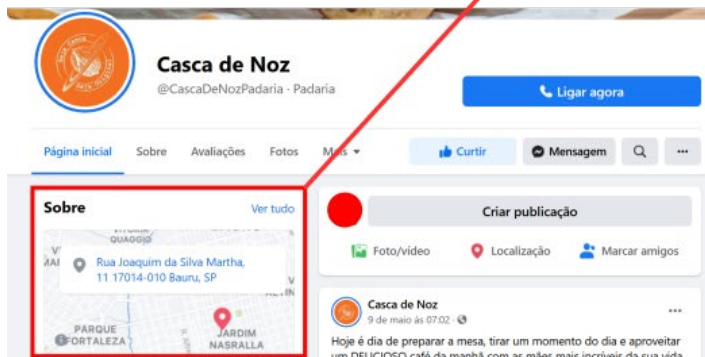
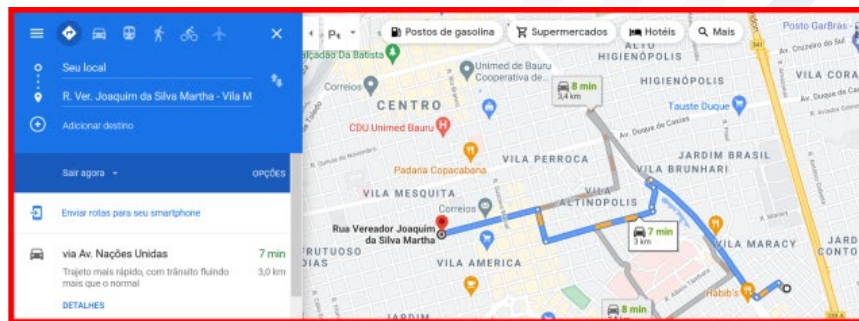
Principais Diferenças

| Web Services | Web API |
|---|--|
| Web Services são um tipo de API que deve ser acessado por meio de uma conexão de rede. | As API's são interfaces de aplicações, o que implica que um aplicativo pode se comunicar com outro de maneira padronizada. |
| Web Services são usados por REST, SOAP e XML-RPC para comunicação. | API é usada para qualquer estilo de comunicação. |
| Todos os Web Services são API's | API's não são necessariamente Web Services. |
| Não tem design leve, necessitando de uma convenção SOAP para enviar/receber dados do sistema. | Tem uma arquitetura leve, sendo utilizado em dispositivos que restringem a capacidade de transmissão (como smartphones). |
| Oferece suporte apenas para o protocolo HTTP. | Oferece suporte para o protocolo HTTP/s. |
| Não é de código aberto, mas pode ser adaptado por qualquer cliente que compreenda XML. | É de código-aberto e pode ser criado com .NET Framework (ASP.NET) |
| Suporta apenas XML. | Suporta XML e JSON. |



5. **Web API's Populares**

Google Maps



Twitter

CADASTRE-SE

ENTRAR

Acesse via Número de Telefone

endereço de e-mail ou ID de membro

Senha

INICIAR SESSÃO

Esqueceu sua Senha?

Acesso rápido com

Ver mais ▾

Autorizar o acesso de
AliExpress.com à sua conta?

Usuário ou e-mail

Senha

☐ Lembrar-me · [Esqueceu sua senha?](#)

Entrar

Cancelar

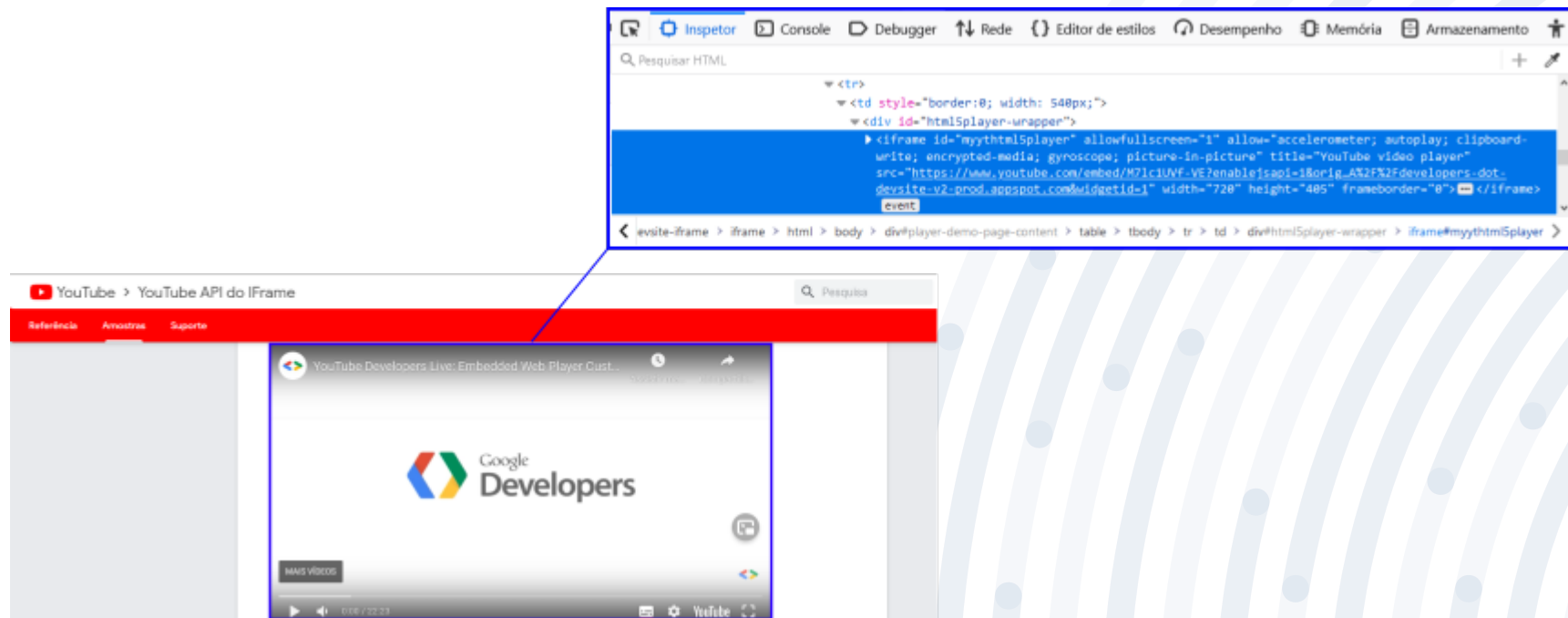
Este aplicativo poderá:

- Ver os Tweets da sua timeline (inclusive os Tweets protegidos) e suas listas e coleções.
- Ver as informações do seu perfil e as configurações da sua conta do Twitter.
- Ver as contas que você segue, silenciou e bloqueou.
- Ver seu endereço de e-mail.

[AliExpress.com](#)
[www.aliexpress.com](#)
[AliExpress](#)
[Política de Privacidade](#)
[Termos e Condições](#)

35

Youtube



6. Referências

JIN, Brenda *et al.* What's an API? In: JIN, Brenda; SAHNI, Saurabh; SHEVAT, Amir. **Designing Web APIs**: building apis that developers love. Sebastopol: O'Reilly, 2018. Cap. 1. p. 1-8.

MDN WEB DOOCS. **Introdução às Web APIs**. Disponível em: https://developer.mozilla.org/pt-BR/docs/Learn/JavaScript/Client-side_web_APIs/Introduction. Acesso em: 18 de maio de 2021.

RED HAT (org.). **O que é API?** Disponível em: <https://www.redhat.com/pt-br/topics/api/what-are-application-programming-interfaces>. Acesso em: 18 de maio de 2021.

GEEKS FOR GEEKS. What is Web API and why we use it?. Disponível em: <https://www.geeksforgeeks.org/what-is-web-api-and-why-we-use-it/>. Acesso em: 19 de maio de 2021.

GEEKS FOR GEEKS. Differences between Web Services and Web API. Disponível em: <https://www.geeksforgeeks.org/differences-between-web-services-and-web-api/?ref=rp>. Acesso em: 19 de maio de 2021.

DEVELOPERS GOOGLE. Demonstração do player do YouTube. Disponível em: https://developers.google.com/youtube/youtube_player_demo?authuser=1. Acesso em: 19 de maio de 2021.

RED HAT (org.). O que é API REST?. Disponível em: <https://www.redhat.com/pt-br/topics/api/what-is-a-rest-api>. Acesso em: 19 de maio de 2021.

WAGNER, Janet.

The Increasing Importance of APIs in Web Development.

Disponível em: <https://code.tutsplus.com/articles/the-increasing-importance-of-apis-in-web-development--net-22368>. Acesso em: 19 de maio de 2021.

SANTOS, Wendell.

Which API Types and Architectural Styles are Most Used?.

Disponível em:

<https://www.programmableweb.com/news/which-api-types-and-architectural-styles-are-most-used/research/2017/11/26>.

Acesso em: 19 de maio de 2021.