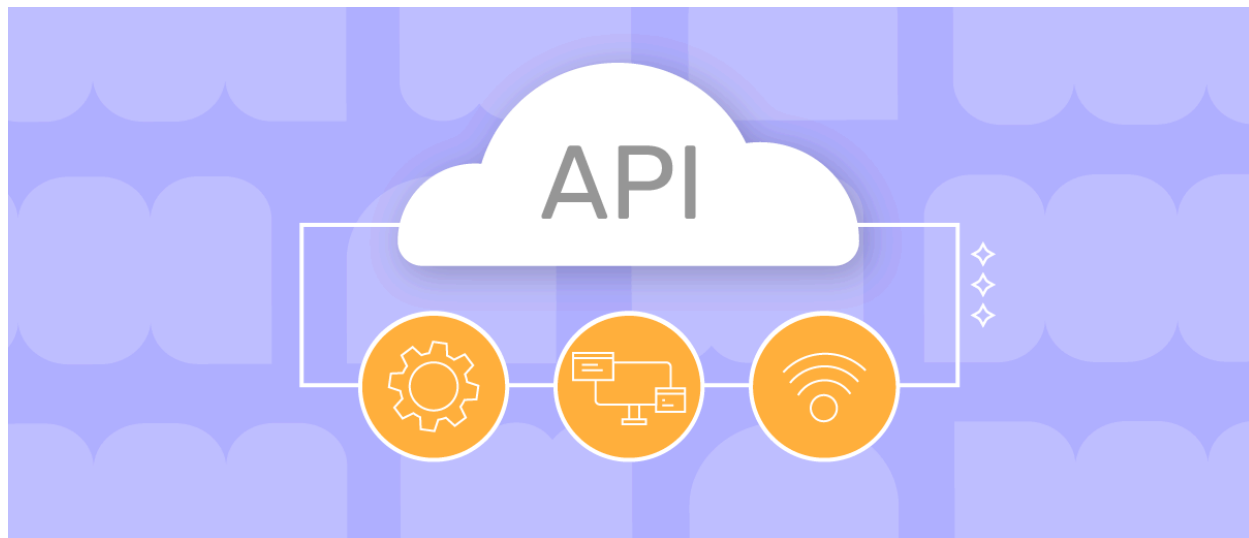


API:



Atividade:

Escolha uma API qualquer da sua escolha, e liste quais componentes estão ou não presentes em sua documentação.

Nome: Júlia Cristina Dos Reis

API Address Validation (Google maps)



Google Maps API

Como funciona?

Com a API Address Validation, é possível determinar se um endereço se refere a um lugar real.

Componentes:

Endpoints: Descrição dos recursos acessíveis pela API, com seus métodos HTTP(GET, POST, PUT, DELETE etc).

API Address Validation: Presente

Endpoint de serviço

Um [endpoint de serviço](#) é um URL de base que especifica o endereço de rede de um serviço de API. Um serviço pode ter vários endpoints de serviço. Este serviço tem o endpoint a seguir, e todos os URIs abaixo são relativos a ele:

- <https://addressvalidation.googleapis.com>

Recurso REST: [v1](#)

Métodos

[provideValidationFeedback](#)

POST /v1:provideValidationFeedback

Feedback sobre o resultado da sequência de tentativas de validação.

Parâmetros: Quais parâmetros são necessários para cada endpoint (parâmetros de consulta, corpo da requisição, cabeçalhos).

API Address Validation: Presente

Na referência REST da API, são definidos os parâmetros esperados (por exemplo, um objeto `ValidateAddressRequest`, campos esperados, etc.).

`validateAddress`

POST `/v1:validateAddress`

Valida um endereço.

Exemplo de requisição e respostas:

Como deve ser uma requisição bem formada para endpoint.

API Address Validation: Presente

A documentação de referência (REST) contém exemplos de requisições JSON e respostas associadas para os endpoints de validação de endereço.

```
estilos de formatação
{
  "basePath": "",
  "name": "addressvalidation",
  "version": "v1",
  "resources": [
    "v1" {
      "methods": {
        "validateAddress": {
          "id": "addressvalidation.validateAddress",
          "path": "v1:validateAddress",
          "flatMap": "v1:validateAddress",
          "httpMethod": "POST",
          "parameters": [],
          "parameterOrder": [],
          "request": {
            "$ref": "GoogleMapsAddressValidationV1ValidateAddressRequest"
          },
          "response": {
            "$ref": "GoogleMapsAddressValidationV1ValidateAddressResponse"
          },
          "scopes": [
            "https://www.googleapis.com/auth/cloud-platform",
            "https://www.googleapis.com/auth/maps-platform.addressvalidation"
          ],
          "description": "Validates an address."
        },
        "provideValidationFeedback": {
          "id": "addressvalidation.provideValidationFeedback",
          "path": "v1:provideValidationFeedback",
          "flatMap": "v1:provideValidationFeedback",
          "httpMethod": "POST",
          "parameters": [],
          "parameterOrder": [],
          "request": {
            "$ref": "GoogleMapsAddressValidationV1ProvideValidationFeedbackRequest"
          },
          "response": {
            "$ref": "GoogleMapsAddressValidationV1ProvideValidationFeedbackResponse"
          },
          "scopes": [
            "https://www.googleapis.com/auth/cloud-platform",
            "https://www.googleapis.com/auth/maps-platform.addressvalidation"
          ],
          "description": "Feedback about the outcome of the sequence of validation attempts. This should be the last call made after a sequence of validation calls for the same address, and should be called once the transaction is concluded. This should only be sent once for the sequence of 'ValidateAddress' requests needed to validate an address fully."
        }
      }
    }
  ],
  "ownerDomain": "google.com",
  "id": "addressvalidationv1",
  "icon": {
    "url": "https://www.googleapis.com/discovery/v1/apis/addressvalidation/v1/icon"
  }
}
```

```
constructor(opts?: ClientOptions, gaxInstance?: typeof gax | typeof gax.fallback);
```

Construct an instance of AddressValidationClient.

Parameters	
Name	Description
opts	ClientOptions
gaxInstance	typeof gax typeof gax.fallback : loaded instance of google-gax . Useful if you need to avoid loading the default gRPC version and want to use the fallback HTTP implementation. Load only fallback version and pass it to the constructor: ``` const gax = require('google-gax/build/src/fallback'); // avoids loading google-gax with gRPC const client = new AddressValidationClient({fallback: 'rest'}, gax); ```

Properties

addressValidationStub

```
addressValidationStub?: Promise<{  
  [name: string]: Function;  

```

apiEndpoint

```
static get apiEndpoint(): string;
```

The DNS address for this API service - same as servicePath(), exists for compatibility reasons.

auth

```
auth: gax.GoogleAuth;
```

descriptors

Código de erro: O que significa cada código de status HTTP retornado pela API (404, 500. etc)

API Address Validation: Não possui

E deu erro!!!



404. That's an error.

The requested URL / was not found on this server. That's all we know.



Autenticação e Autorização: Se a API requer algum tipo de autenticação (como tokens. Chave de API)

API Address Validation: Não possui

A documentação parece focar apenas no uso de chave de API como método de autenticação. Não parece haver suporte ou orientação ampla para usar OAuth ou tokens.