**Documentação da API de Tradução**

**1. Endpoints Disponíveis**

1.1. POST /api/translate

- Função: Realizar tradução de texto

-Resposta:

{

"id": 1,

"textoOriginal": "Texto original",

"textoTraduzido": "Texto traduzido",

"idioma": "Código do idioma"

}

1.2. GET /api/translation/:id

- Função: Recuperar uma tradução específica pelo ID

- Parâmetros: id (número) na URL

- Resposta:

{

"id": 1,

"texto\_original": "Texto original",

"texto\_traduzido": "Texto traduzido",

"idioma": "Código do idioma",

"created\_at": "Data de criação"

}

1.3. GET /api/translations

- Função: Listar todas as traduções realizadas

- Resposta: Array de traduções

{

"id": 1,

"texto\_original": "Texto original",

"texto\_traduzido": "Texto traduzido",

"idioma": "Código do idioma",

"created\_at": "Data de criação"

}

]

**2. Tecnologias Utilizadas**

2.1. Express

Função: Framework web para Node.js

- Responsabilidades:

* Gerenciamento de rotas da API
* Processamento de requisições HTTP
* Middleware para parsing de JSON
* Tratamento de erros
* Estruturação da aplicação

2.2. Redis

Função: Sistema de cache em memória

-Responsabilidades:

* Armazenamento temporário de traduções;
* Redução de chamadas à API Azure;
* Melhoria de performance;
* Cache com expiração de 15 minutos.

2.3. MariaDB

Função: Banco de dados relacional

- Responsabilidades:

* Armazenamento persistente das traduções
* Histórico completo de traduções
* Consultas e recuperação de dados
* Garantia de durabilidade dos dados

2.4. Azure Translation API

Função: Serviço de tradução em nuvem

- Responsabilidades:

* Realização das traduções
* Suporte a múltiplos idiomas
* Precisão nas traduções
* Integração com serviços Microsoft

**3. Fluxo Geral das Requisições**

3.1. Fluxo de Tradução

1. Recebimento da Requisição

- API recebe solicitação POST com texto e idioma alvo

2. Verificação de Cache

- Sistema verifica se a tradução existe no Redis

- Se encontrada, retorna imediatamente sem consultar externos

3. Processo de Tradução

-Se não existe no cache:

1. Chama Azure Translation API

2. Recebe tradução

3. Salva no MariaDB

4. Armazena no Redis (cache por 15 min)

5. Retorna resultado ao usuário

4. Armazenamento Duplo

- Redis: armazenamento rápido, temporário

- MariaDB: armazenamento persistente, histórico

3.2. Complementaridade Cache-Banco

Esta arquitetura permite:

- Respostas rápidas para traduções recentes

- Economia de recursos (menos chamadas à API)

- Histórico completo de traduções

- Alta disponibilidade de dados

- Escalabilidade do sistema