



**Shortly**

---



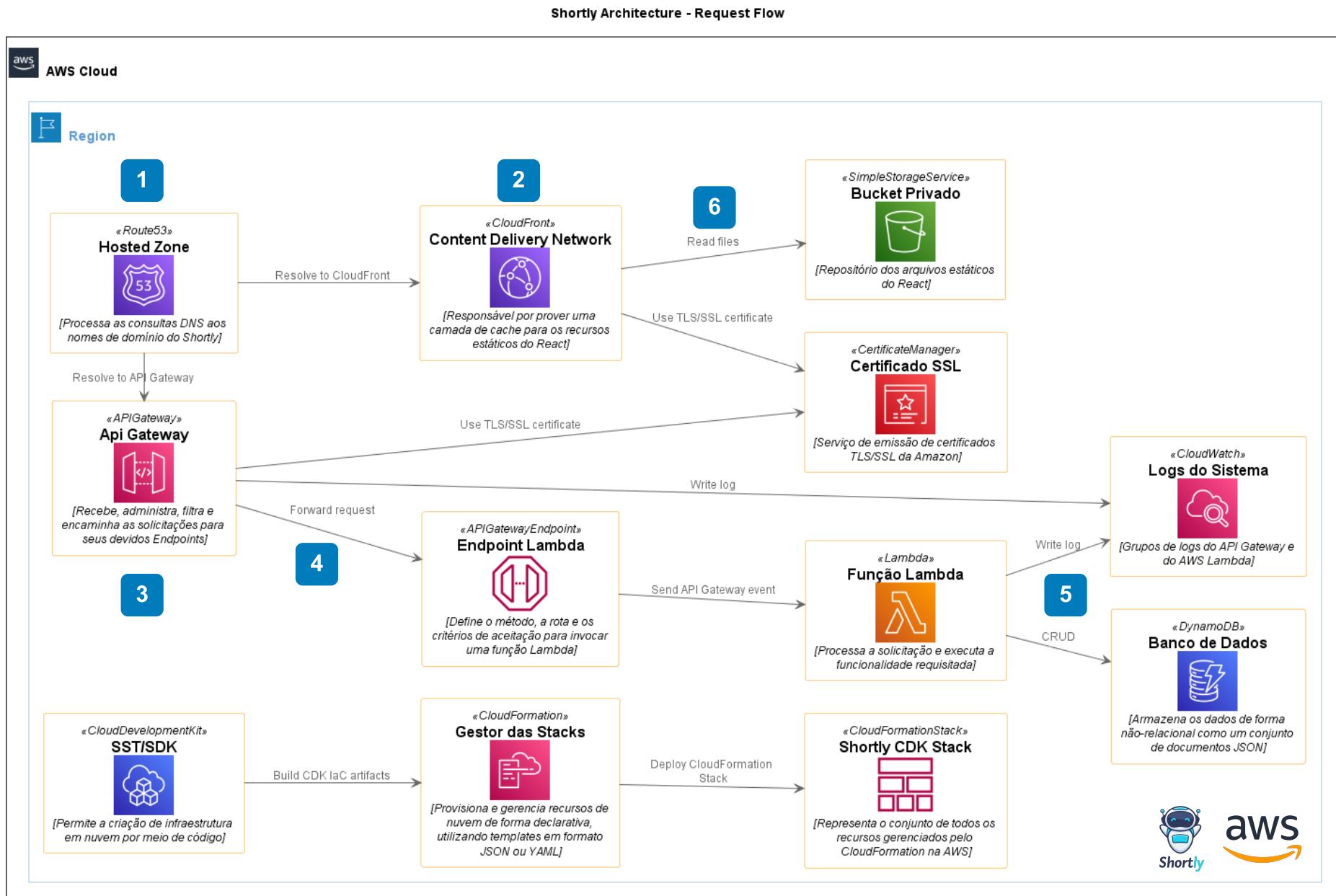
## Arquitetura de Referência

Versão Serverless (AWS Lambda)

<https://shortly.com.br/>

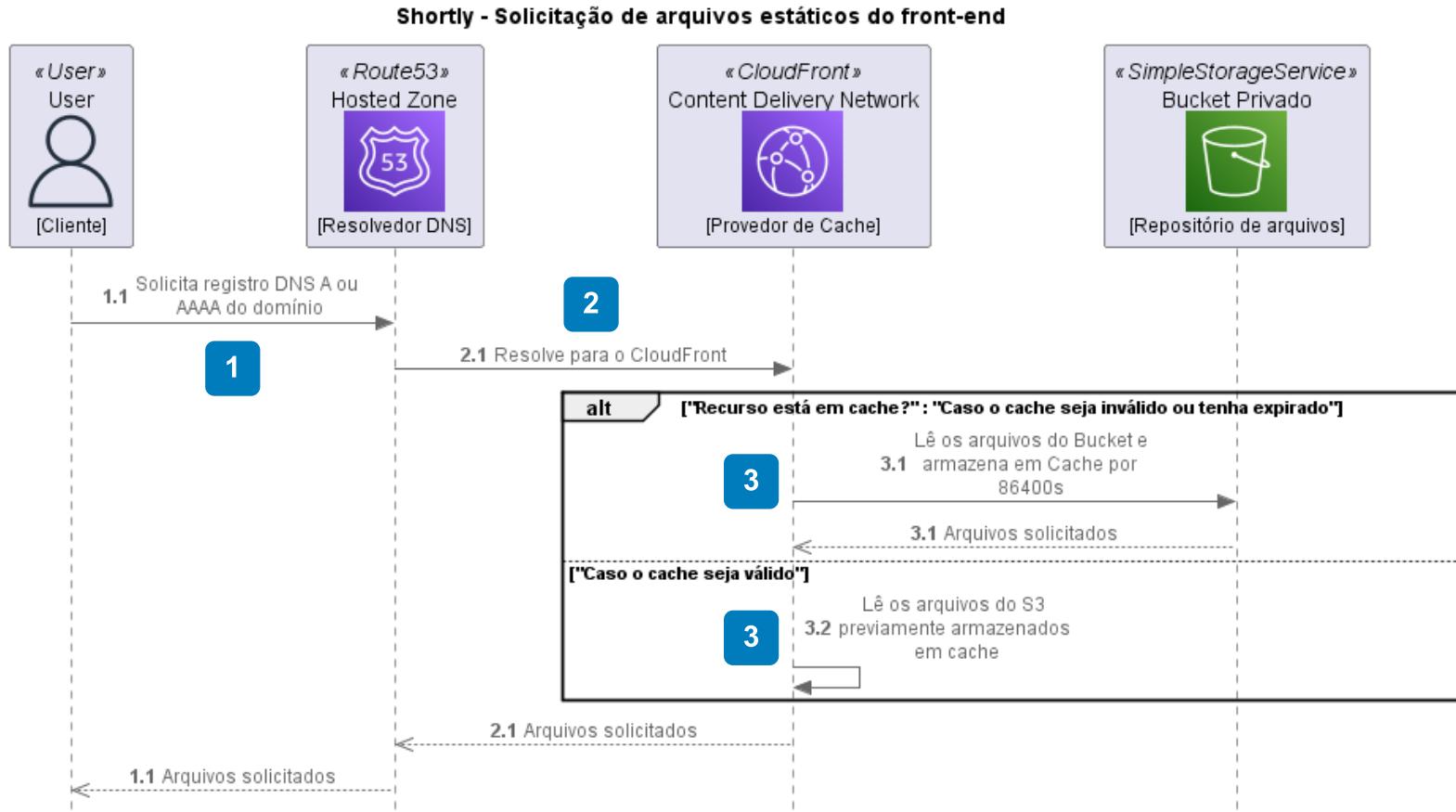
# Visão Geral da Arquitetura do Shortly, Parte 1 de 8

O diagrama a seguir representa a arquitetura do Shortly, um sistema que utiliza o AWS Lambda em conjunto com o DynamoDB. O Lambda é responsável pelo processamento das solicitações, enquanto o DynamoDB armazena os dados de forma escalável e sem servidor. Essa combinação permite uma arquitetura eficiente, com processamento sob demanda e escalabilidade automática, proporcionando uma experiência de baixa latência e alta disponibilidade para os usuários além de baixos custos de infraestrutura.



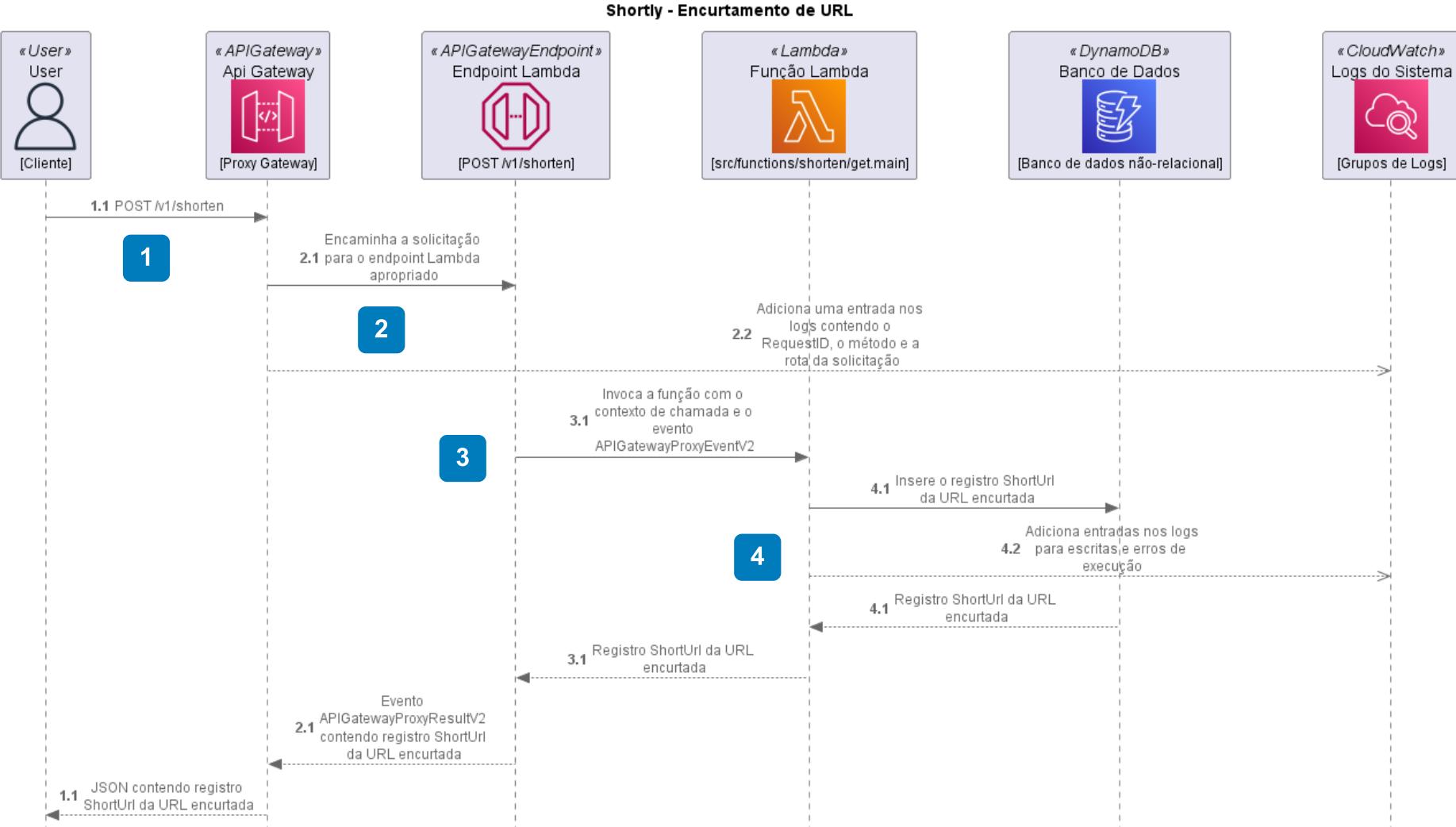
# Visão Geral da Arquitetura do Shortly, Parte 2 de 8

O diagrama de sequência a seguir representa a interação de um usuário com um front-end web hospedado em um Bucket S3 privado, fornecido por uma distribuição CloudFront. O processo envolve quatro entidades principais: o ator (usuário), o resolvedor DNS do Route53, o Amazon CloudFront e o Bucket Privado do Amazon S3.



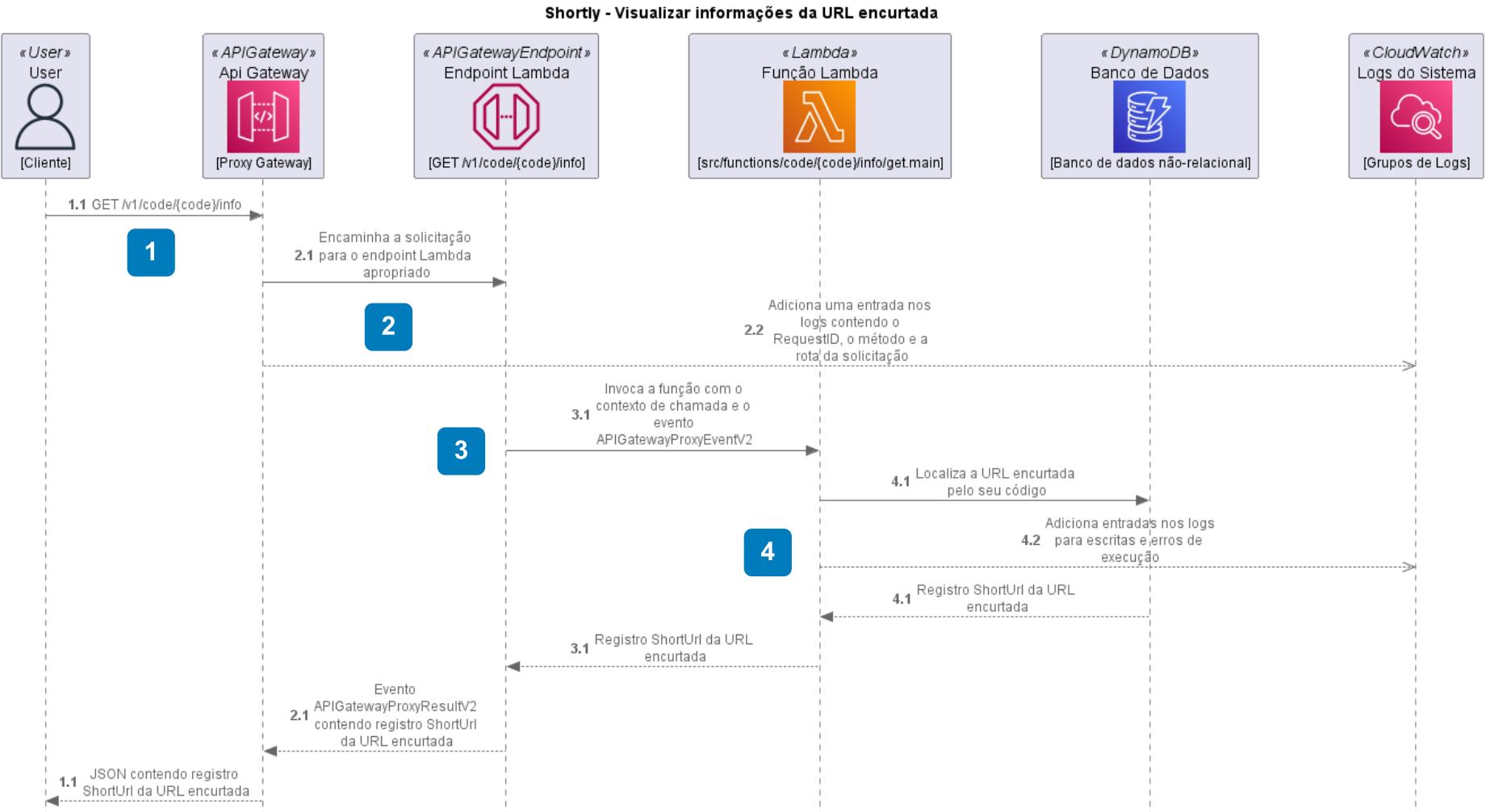
# Visão Geral da Arquitetura do Shortly, Parte 3 de 8

O diagrama de sequência a seguir mostra a interação de um usuário com um endpoint provido pelo API Gateway em conjunto com uma FaaS que interage com um banco de dados NoSQL Amazon DynamoDB. No esquema abaixo vemos a interação do usuário para **Encurtar uma URL** a partir do Shortly.



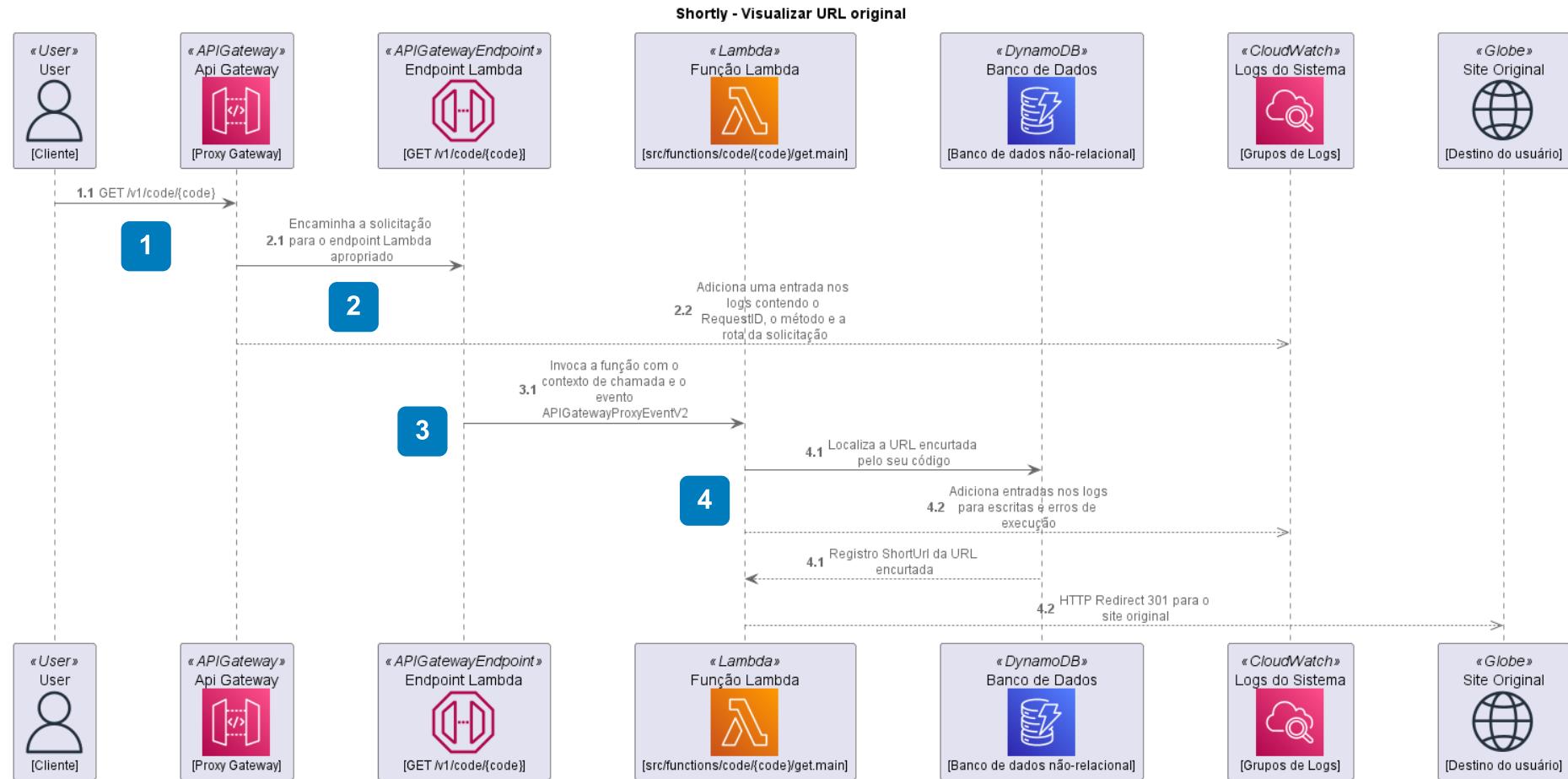
# Visão Geral da Arquitetura do Shortly, Parte 4 de 8

O diagrama de sequência a seguir mostra a interação de um usuário com um endpoint provido pelo API Gateway em conjunto com uma FaaS que interage com um banco de dados NoSQL Amazon DynamoDB. No esquema abaixo vemos a interação do usuário para **Visualizar informações da URL encurtada** a partir do Shortly.



# Visão Geral da Arquitetura do Shortly, Parte 5 de 8

O diagrama de sequência a seguir mostra a interação de um usuário com um endpoint provido pelo API Gateway em conjunto com uma FaaS que interage com um banco de dados NoSQL Amazon DynamoDB. No esquema abaixo vemos a interação do usuário para **Visualizar a URL original** a partir do Shortly.



1 O usuário envia uma solicitação para **visualizar a URL original** que é então recebida pelo API Gateway do Shortly.

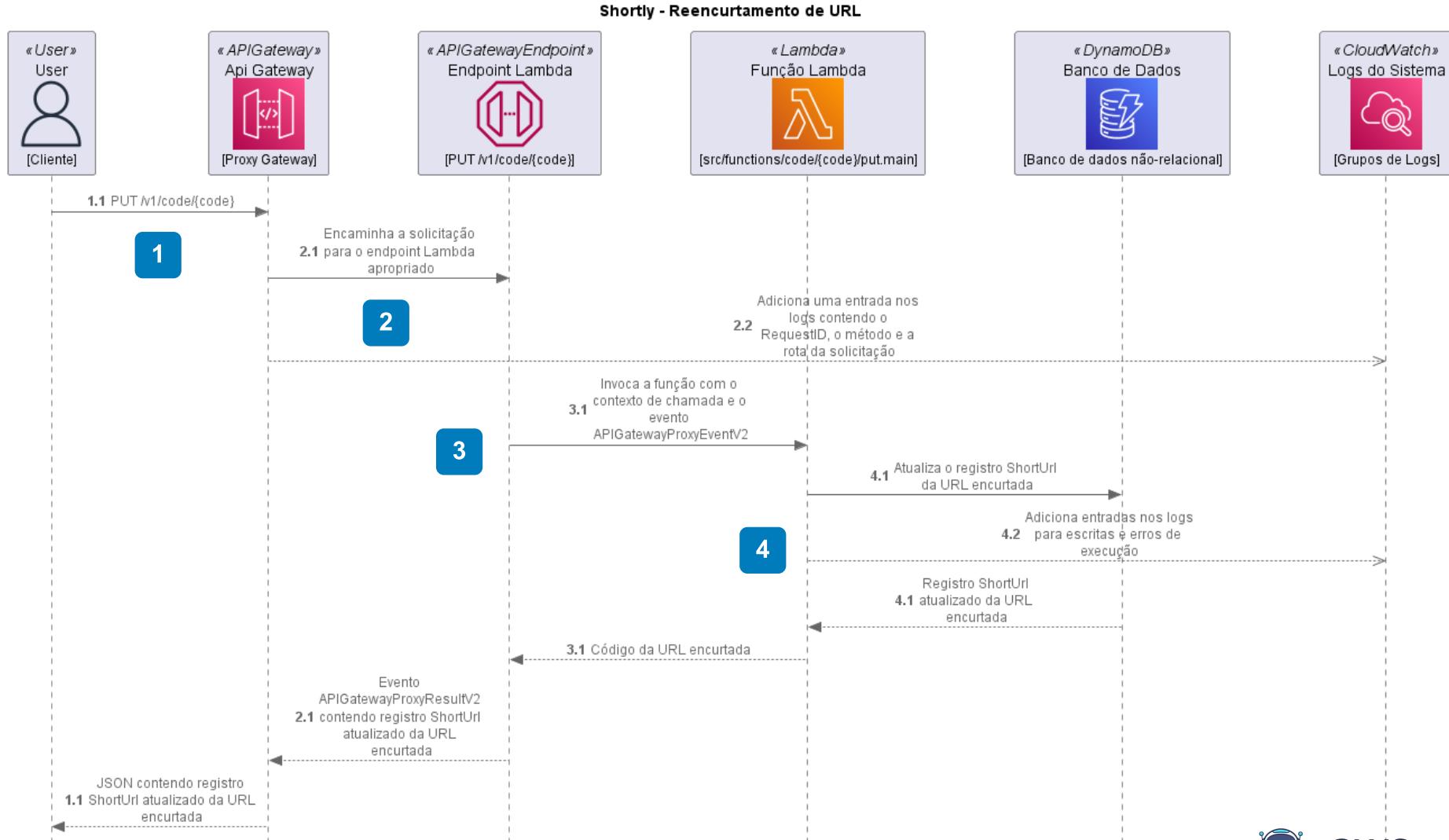
2 O API Gateway, por sua vez, **encaminha a solicitação para o Endpoint Lambda apropriado**, especificado pelas regras de método, rota e nome de domínio solicitados. Esse Endpoint Lambda, por sua vez, recebe a solicitação e verifica se as regras do Endpoint foram atendidas.

3 O Endpoint Lambda que recebe a solicitação do API Gateway elabora uma carga útil para disparar um evento na FaaS **src/functions/code/{code}/get.main** que é invocada com essa carga útil de evento.

4 A FaaS **src/functions/code/{code}/get.main** recebe a carga útil do evento originado pelo Endpoint Lambda e processa a solicitação lendo um registro no Amazon DynamoDB e escrevendo algumas entradas nos grupos de logs do CloudWatch. Ao final do processo, a FaaS retorna um evento que força o API Gateway a enviar o status HTTP 301 com o cabeçalho Location para redirecionar o usuário para a URL original.

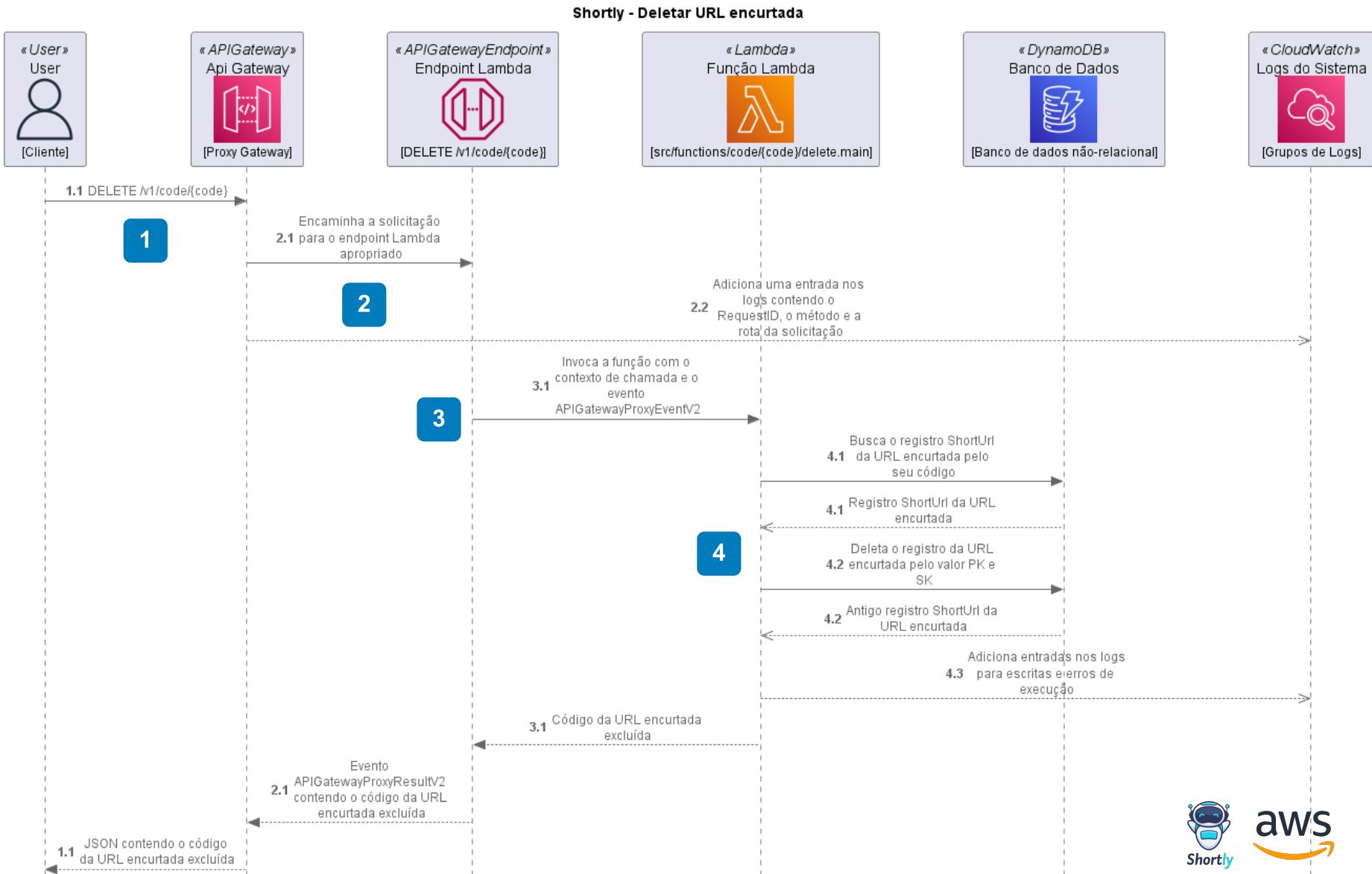
# Visão Geral da Arquitetura do Shortly, Parte 6 de 8

O diagrama de sequência a seguir mostra a interação de um usuário com um endpoint provido pelo API Gateway em conjunto com uma FaaS que interage com um banco de dados NoSQL Amazon DynamoDB. No esquema abaixo vemos a interação do usuário para **Reencurtar uma URL** a partir do Shortly.



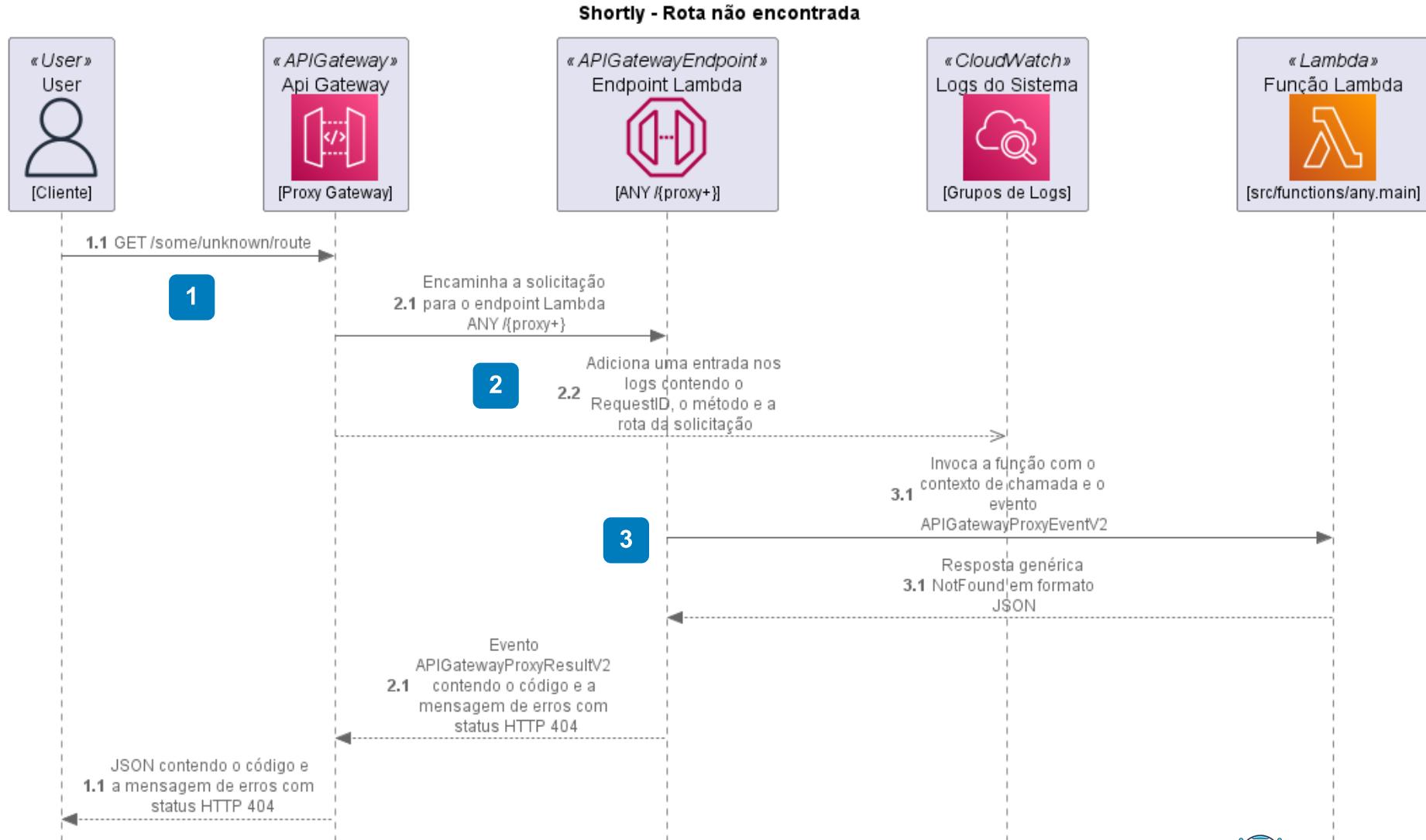
# Visão Geral da Arquitetura do Shortly, Parte 7 de 8

O diagrama de sequência a seguir mostra a interação de um usuário com um endpoint provido pelo API Gateway em conjunto com uma FaaS que interage com um banco de dados NoSQL Amazon DynamoDB. No esquema abaixo vemos a interação do usuário para **Deletar uma URL encurtada** a partir do Shortly.



# Visão Geral da Arquitetura do Shortly, Parte 8 de 8

O diagrama de sequência a seguir mostra a interação de um usuário com um endpoint provido pelo API Gateway em conjunto com uma FaaS que interage com um banco de dados NoSQL Amazon DynamoDB. No esquema abaixo vemos a interação do usuário para **Acessar uma rota não encontrada** no Shortly.



- 1 O usuário envia uma solicitação para uma rota inválida que não existe no sistema que é então recebida pelo API Gateway do Shortly.
- 2 O API Gateway, por sua vez, encaminha a solicitação para o Endpoint Lambda que trata das solicitações enviadas para qualquer método em qualquer rota que não tenha sido especificada por nenhum Endpoint Lambda. Esse Endpoint Lambda, por sua vez, recebe a solicitação e verifica se as regras do Endpoint foram atendidas.
- 3 O Endpoint Lambda que recebe a solicitação do API Gateway elabora uma carga útil para disparar um evento na FaaS **src/functions/any.main** que é invocada com essa carga útil de evento. Como nenhuma rota foi encontrada, uma FaaS padrão é executada, devolvendo uma resposta de erro 404.





---

# Arquitetura de Referência

Versão Serverless (AWS Lambda)