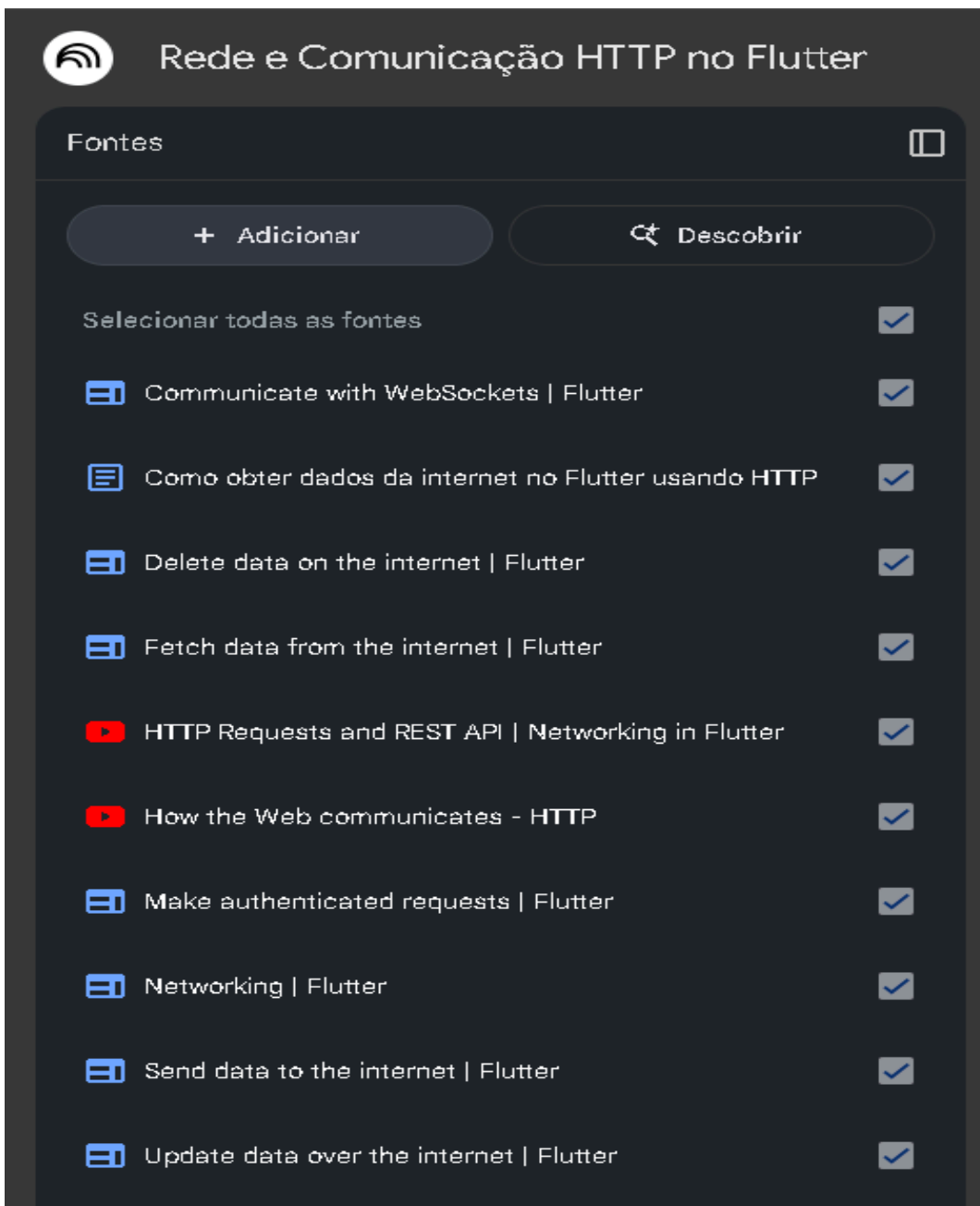


# Avaliação 07 - NotebookLM

Criar um NotebookLM sobre o tema NETWORKING e em <https://docs.flutter.dev/data-and-backend/networking>. Usar como fonte o link acima, videos no youtube relacionados ao tema, e PDF. Criar um questionário com 20 questões (múltipla escolha 4 opções de resposta) no final relacionar as respostas. Evidenciar no Github a avaliação-07 com prints da tela do Notebooklm com as fontes, e o questionário em PDF com as respostas.

- Links adicionados ao notebookLM



## • Resultado das questões que foram pedidas

Com certeza! Abaixo está um questionário com 20 questões de múltipla escolha sobre comunicação de dados em Flutter, baseado nas fontes fornecidas, com o gabarito no final.

-----

Questionário: Comunicação de Dados em Flutter

Responda às questões de múltipla escolha com base nas informações das fontes.

1. Qual é o propósito principal do pacote `http` no Flutter? a) Para criar interfaces de usuário interativas [Não especificado nas fontes, mas o `http` é para rede]. b) Para gerenciar o estado da aplicação localmente [Não especificado nas fontes]. c) Para realizar requisições HTTP como GET, POST, PUT e DELETE **1** .... d) Para construir jogos em 2D [Não especificado nas fontes].
2. Para qual tipo de comunicação de dados os WebSockets são utilizados no Flutter? a) Requisições HTTP unidirecionais **5** **6** . b) Comunicação bidirecional com um servidor sem polling **5** . c) Armazenamento de dados localmente no disco [Não especificado nas fontes]. d) Apenas para enviar dados ao servidor [Não especificado nas fontes].
3. Qual pacote é necessário para se conectar a um servidor WebSocket no Flutter? a) `flutter_socket_io` [Não especificado nas fontes]. b) `web_socket_channel` **7** . c) `http_channel` [Não especificado nas fontes]. d) `dart:io` [Não especificado nas fontes].
4. Como se adiciona o pacote `http` a um projeto Flutter? a) Adicionando-o diretamente ao arquivo `main.dart` [Não especificado nas fontes]. b) Executando `flutter pub add http` e importando-o **3** .... c) Não é necessário adicionar, ele vem por padrão com o Flutter [Não especificado nas fontes]. d) Através de um widget `HttpPackageBuilder` [Não especificado nas fontes].
5. Qual método HTTP é utilizado para buscar dados de um servidor? a) `http.post()` **2** **11** . b) `http.delete()` **2** **12** . c) `http.put()` **2** **13** . d) `http.get()` **2** ....
6. No contexto de requisições HTTP em Flutter, o que representa um objeto `Future`? a) Um widget que exibe dados assíncronos [Não é o objeto `Future` em si, mas sim o `FutureBuilder`]. b) Um valor ou erro potencial que estará disponível no futuro **12** .... c) Uma resposta HTTP bem-sucedida [Não, isso é `http.Response`]. d) Uma coleção de dados em tempo real [Não, isso é mais um `Stream`].
7. Por que não é recomendado fazer chamadas de API diretamente no método `build()` de um widget Flutter? a) O método `build()` é chamado apenas uma vez e não recarregaria os dados **19** . b) O `build()` é chamado repetidamente, causando lentidão ao refazer a chamada da API **20** . c) As chamadas de API só podem ser feitas em `initState()` **19** . d) O `build()` não tem acesso ao contexto da aplicação para chamadas de rede [Não especificado nas fontes].
8. O que é necessário para converter o corpo da resposta JSON de uma requisição HTTP em um `Map Dart`? a) A classe

Chat

8. O que é necessário para converter o corpo da resposta JSON de uma requisição HTTP em um `Map Dart`? a) A classe `Image.network` [Não especificado nas fontes]. b) O método `Uri.parse()` **14** . c) O pacote `dart:convert` e a função `jsonDecode()` **16** .... d) O widget `StreamBuilder` [Não especificado nas fontes].
9. Qual HTTP status code é tipicamente retornado para uma requisição GET bem-sucedida? a) 404 Not Found **26** . b) 500 Internal Server Error **27** . c) 200 OK **22** .... d) 201 Created **33** .
10. Qual HTTP status code é geralmente retornado quando os dados são enviados com sucesso e um novo recurso é criado (POST)? a) 200 OK [Não é o mais comum para criação]. b) 400 Bad Request [Não especificado nas fontes]. c) 201 CREATED **24** **33** . d) 405 Method Not Allowed **26** .
11. Como o Flutter lida com a exibição de dados assíncronos (como os obtidos de uma API) na tela? a) Usando o widget `Text` diretamente [Não especificado nas fontes]. b) Através do widget `FutureBuilder` **14** .... c) Com o método `setState()` sem um `FutureBuilder` [Não especificado nas fontes]. d) Exclusivamente com `ListView.builder` **37** .
12. Qual é a principal diferença entre `Stream` e `Future` no Dart, conforme descrito para WebSockets? a) `Future` lida com muitos eventos ao longo do tempo, enquanto `Stream` retorna uma única resposta assíncrona **38** . b) `Stream` lida com muitos eventos ao longo do tempo, enquanto `Future` retorna uma única resposta assíncrona **38** . c) Ambos são idênticos e usados para o mesmo propósito [Não especificado nas fontes]. d) `Stream` é para dados locais e `Future` é para dados da internet [Não especificado nas fontes].
13. Ao lidar com erros de rede, qual boa prática é recomendada? a) Ignorar exceções, pois o Flutter lida com elas automaticamente [Não especificado nas fontes]. b) Usar `try/catch` para capturar exceções e tratar erros de rede **14** **27** . c) Apenas exibir um ícone de carregamento indefinidamente [Não é uma boa prática para erros]. d) Não é necessário, a conexão de rede é sempre estável [Não especificado nas fontes].
14. Como você enviaria dados para um servidor usando o pacote `http` no Flutter? a) Usando `http.get()` com um corpo [Não é o método correto para enviar dados]. b) Usando `http.post()` e fornecendo um corpo JSON **11** .... c) Usando `http.fetch()` [Não especificado nas fontes]. d) Usando `channel.sink.add()` [Não é para HTTP, é para WebSockets].
15. Qual permissão é necessária no arquivo `AndroidManifest.xml` para que um aplicativo Flutter no Android possa buscar dados da internet? a) `<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />` [Não especificado nas fontes]. b) `<uses-permission android:name="android.permission.CAMERA" />` [Não especificado nas fontes]. c) `<uses-permission android:name="android.permission.INTERNET" />` **9** .... d) `<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />` [Não especificado nas fontes].
16. O que significa "REST" no contexto de REST APIs? a) Rapidly Evolving Software Technology [Não especificado nas fontes]. b) Representational State Transfer **27** . c) Real-time Event Streaming [Não especificado nas fontes]. d) Remote Execution System [Não especificado nas fontes].
17. Uma das propriedades chave de uma REST API é ser "stateless". O que isso implica? a) O servidor mantém o histórico de todas as requisições anteriores de um cliente **44** . b) Cada requisição do cliente ao servidor deve conter todas as informações necessárias para ser processada, sem depender de contexto anterior **27** **44** . c) As requisições são sempre bem-sucedidas [Não especificado nas fontes].

## Chat

15. Qual permissão é necessária no arquivo `AndroidManifest.xml` para que um aplicativo Flutter no Android possa buscar dados da internet? a) `<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />` [Não especificado nas fontes]. b) `<uses-permission android:name="android.permission.CAMERA" />` [Não especificado nas fontes]. c) `<uses-permission android:name="android.permission.INTERNET" />` 9 ... d) `<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />` [Não especificado nas fontes].

16. O que significa "REST" no contexto de REST APIs? a) Rapidly Evolving Software Technology [Não especificado nas fontes]. b) Representational State Transfer 27 . c) Real-time Event Streaming [Não especificado nas fontes]. d) Remote Execution System [Não especificado nas fontes].

17. Uma das propriedades chave de uma REST API é ser "stateless". O que isso implica? a) O servidor mantém o histórico de todas as requisições anteriores de um cliente 44 . b) Cada requisição do cliente ao servidor deve conter todas as informações necessárias para ser processada, sem depender de contexto anterior 27 44 . c) As requisições são sempre bem-sucedidas [Não especificado nas fontes]. d) A comunicação é mantida aberta por um longo período 6 .

18. Qual método HTTP é usado para atualizar dados existentes em um servidor? a) `http.delete()` 2 . b) `http.get()` 2 . c) `http.post()` 2 . d) `http.put()` 2 13 .

19. Para adicionar um cabeçalho de autorização a uma requisição HTTP no Flutter, qual cabeçalho é comumente utilizado? a) `Content-Type` 11 ... b) `Authorization` 45 . c) `User-Agent` [Não especificado nas fontes]. d) `Accept` [Não especificado nas fontes].

20. Ao usar o `WebSocketChannel` para enviar dados a um servidor, qual objeto ou método é usado para "empurrar" as mensagens? a) `channel.stream.listen()` [Não, isso é para ouvir]. b) `channel.data.send()` [Não especificado nas fontes]. c) `channel.sink.add()` 46 47 . d) `channel.connect()` [Não, isso é para conectar].

### Gabarito:

1. c) Para realizar requisições HTTP como GET, POST, PUT e DELETE 1 ... .
2. b) Comunicação bidirecional com um servidor sem polling 5 .
3. b) `web_socket_channel` 7 .
4. b) Executando `flutter pub add http` e importando-o 3 ... .
5. d) `http.get()` 2 ... .
6. b) Um valor ou erro potencial que estará disponível no futuro 12 ... .
7. b) O `build()` é chamado repetidamente, causando lentidão ao refazer a chamada da API 20 .

11. b) Através do widget `FutureBuilder` 14 ... .

12. b) `Stream` lida com muitos eventos ao longo do tempo, enquanto `Future` retorna uma única resposta assíncrona 38 .

13. b) Usar `try/catch` para capturar exceções e tratar erros de rede 14 27 .

14. b) Usando `http.post()` e fornecendo um corpo JSON 11 ... .

15. c) `<uses-permission android:name="android.permission.INTERNET" />` 9 ... .

16. b) Representational State Transfer 27 .

17. b) Cada requisição do cliente ao servidor deve conter todas as informações necessárias para ser processada, sem depender de contexto anterior 27 44 .

18. d) `http.put()` 2 13 .

19. b) `Authorization` 45 .

20. c) `channel.sink.add()` 46 47 .