Em Flutter, você pode usar o pacote "socket\_io\_client" para se conectar a um servidor Node.js usando o protocolo Socket.IO. Para fazer isso, você precisará fazer o seguinte:

Adicionar o pacote "socket\_io\_client" ao arquivo pubspec.yaml do seu projeto e importá-lo no arquivo onde deseja usá-lo:

yaml

Copy code

dependencies:

socket\_io\_client: ^0.9.4

dart

Copy code

import 'package:socket\_io\_client/socket\_io\_client.dart' as IO;

Criar uma instância do socket e se conectar ao servidor Node.js:

dart

Copy code

IO.Socket socket = IO.io('http://localhost:3000', <String, dynamic>{

'transports': ['websocket'],

});

Adicionar ouvintes para eventos específicos do servidor, como "connect", "disconnect", "error", etc.

dart

Copy code

socket.on('connect', (\_) {

print('connected');

});

socket.on('event', (data) => print(data));

socket.on('disconnect', (\_) {

print('disconnected');

});

Enviar dados para o servidor usando o método "emit":

dart

Copy code

socket.emit("message", "Hello from Flutter!");

Fechar a conexão quando você terminar de usar o socket

dart

Copy code

socket.disconnect();

Note que este é um exemplo básico e pode ser modificado de acordo com suas necessidades específicas. Certifique-se de que o servidor Node.js está configurado corretamente para lidar com o protocolo Socket.IO e que as configurações de endereço e porta estão corretas.