signalr_netcore Package Overview

1. What Is signalr_netcore?

A Dart/Flutter client for ASP .NET Core SignalR, enabling Flutter apps to invoke server-side hub methods, receive broadcasts, and auto-reconnect on network drops.

2. Core Features

Transports	WebSocket; Server-Sent Events; Long Polling (fallbacks)
Protocols	JSON and MessagePack
Auto-Reconnect	Built-in retry policy
Streaming RPC	Server and client streaming
Logging	Integrates with Dart logging package

3. Installing

Add to pubspec.yaml:

```yaml
dependencies:
  flutter: sdk: flutter
  signalr_netcore: ^1.4.3
```

Then run `flutter pub get`.

4. Basic Usage

Create & Configure	```dart
	final hub = HubConnectionBuilder()
	   .withUrl("http://server/hub")
	   .withAutomaticReconnect()
	   .configureLogging(Logger("SignalR"))
	   .build();
	hub.onclose((error) { /* handle close */ });
	``

Start & Invoke	```dart
	await hub.start();
	await hub.invoke('AddTask', args: ['Buy milk']);
	``

Receive Server Calls	```dart
	hub.on('TasksUpdated', (args) {
	 final list = args![0] as List;
	 // update UI
	});
	``

5. Advanced Tips & Important Info

MessagePack	Use `MessagePackHubProtocol()` for compact binary messages. Requires server to add MessagePack protocol.
Custom Headers & Auth	Use `HttpConnectionOptions` to attach JWT or custom headers.
CORS	Configure server CORS with `AllowCredentials()` and `WithOrigins()` to permit Flutter web or emulator connections.
Error Handling	Use `onclose`, `onreconnecting`, `onreconnected` to handle disconnects and retry logic.

6. Why Choose signalr_netcore?

- Matches ASP.NET Core SignalR feature set
- Cross-platform: Android, iOS, Web, Desktop
- Built-in auto-reconnect
- Pure Dart, no native plugins
- Open-source (MIT) with active community

7. Further Reading & Resources

- Pub.dev: https://pub.dev/packages/signalr_netcore
- ASP.NET Core SignalR Docs: https://docs.microsoft.com/aspnet/core/signalr
- MessagePack Protocol: https://docs.microsoft.com/aspnet/core/signalr/messagepack