

# 🗨 TCP রিয়েল-টাইম চ্যাট অ্যাপ

এটি একটি রিয়েল-টাইম **TCP চ্যাট অ্যাপ** যেখানে একাধিক ক্লায়েন্ট একই সার্ভারের মাধ্যমে মেসেজ পাঠাতে ও পেতে পারে।

তুমি যা মেসেজ পাঠাবে, সেটি **সার্ভার সবার কাছে ব্রডকাস্ট করবে**।

## ✳ ধারণা

- **Chat Server:** সব ক্লায়েন্ট থেকে মেসেজ গ্রহণ করে অন্য সবার কাছে পাঠায়।
- **TCP Socket:** নিরাপদ ও নির্ভরযোগ্য কানেকশন।
- **Dart:** সার্ভার কোডের জন্য ব্যবহৃত।
- **Flutter:** ক্লায়েন্ট UI তৈরির জন্য ব্যবহৃত।

## 🚀 প্রজেক্ট চালানোর ধাপ

### ১. সার্ভার চালানো

1. **server** ফোল্ডারে ঢুকো:

```
cd server
```

2. **chat\_server.dart** চালাও:

```
dart chat_server.dart
```

3. আউটপুটে দেখাবে:

```
Chat Server started on 0.0.0.0:8082
```

### ২. ক্লায়েন্ট চালানো (Flutter App)

1. **client\_flutter** ফোল্ডারে ঢুকো:

```
cd client_flutter
```

2. **lib/main.dart** ফাইলে তোমার সার্ভারের IP Address দাও:

- যদি সার্ভার **একই কম্পিউটারে** থাকে: **127.0.0.1**
- যদি **ফোনে ক্লায়েন্ট চালাও**: তোমার ল্যাপটপের IP Address (যেমন **192.168.0.3**)।

3. অ্যাপ চালাও:

```
flutter run
```

4. একাধিক ডিভাইসে চালালে সবাই একে অপরের মেসেজ রিয়েল-টাইমে দেখতে পাবে।

---

## সার্ভার কোড (server/chat\_server.dart)

```
import 'dart:io';

List<Socket> clients = [];

void main() async {
  final server = await ServerSocket.bind(InternetAddress.anyIPv4, 8082);
  print('Chat Server started on ${server.address.address}:${server.port}');

  await for (Socket client in server) {
    print('Client connected: ${client.remoteAddress.address}');
    clients.add(client);

    client.listen((data) {
      String message = String.fromCharCodeList(data.toList());
      print('Received: $message');

      for (var c in clients) {
        if (c != client) {
          c.write('User: $message\n');
        }
      }
    }, onDone: () {
      print('Client disconnected: ${client.remoteAddress.address}');
      clients.remove(client);
    });
  }
}
```

---

## ক্লায়েন্ট কোড (client\_flutter/lib/main.dart)

```
import 'dart:io';
import 'dart:ui';
import 'package:flutter/material.dart';
```

```

void main() {
  runApp(const ChatApp());
}

class ChatApp extends StatelessWidget {
  const ChatApp({super.key});

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      title: 'TCP Chat Client',
      theme: ThemeData(
        colorScheme: ColorScheme.fromSeed(seedColor: Colors.deepPurple),
      ),
      home: const ChatPage(),
    );
  }
}
// সম্পূর্ণ main.dart কোড প্রোজেক্টের ভিতরে রয়েছে।

```

## 🌀 UI বৈশিষ্ট্য

- **Animated Gradient Background** – রঙের অ্যানিমেশন।
- **Glassmorphism Chat Bubbles** – প্রতিটি মেসেজ বাবল স্বচ্ছ এবং স্মার্ট লুক।
- **Auto Scroll** – নতুন মেসেজ এলে লিস্ট স্বয়ংক্রিয়ভাবে নিচে যায়।
- **Keyboard Friendly Layout** – কীবোর্ড ওপেন হলে UI ঝাঁকুনি হয় না।

## 💡 গুরুত্বপূর্ণ টিপস

- সার্ভার আগে চালাও, তারপর Flutter ক্লায়েন্ট চালাও।
- ফোন ও ল্যাপটপ একই Wi-Fi নেটওয়ার্কে থাকতে হবে।
- IP Address সঠিক না দিলে অ্যাপ Timeout দেখাবে।