d

| Author(s) : [Mir Ijazur Rahaman](mailto:mi.rahaman@bassetti-group.com) [Pallab Roy](mailto:pallab.roy@bassetti-group.com) | Date created: 10/04/2025  Number of pages: |
| --- | --- |

| Solution Specification |
| --- |

| Summary |
| --- |
| A real-time chat application built with Angular 18 and Firebase Firestore featuring public and private messaging, user status tracking, and responsive UI.  It supports real-time updates, Firestore integration, and can be extended with features like read receipts and authentication. |

| Background | | |
| --- | --- | --- |
| **Date** | **Who** | **Description** |
| 10/04/2025 | [Mir Ijazur Rahaman](mailto:mi.rahaman@bassetti-group.com)[Haque Rizwan](mailto:haque.rizwan@bassetti-group.com)  [Pallab Roy](mailto:pallab.roy@bassetti-group.com) | Version 1.0  Creation of the document. |

# 

Summary

[Features](#_vbydmutjjmcr)

[How It Works](#_6tdq1lno5nvo)

[User Authentication](#_vr8nre6ydwi)

[Public Chat](#_w1uuquiso56r)

[Private Messaging](#_pl1ji6ixddxv)

[Real-time Updates](#_pblr5bqgm8yc)

[Data Structure](#_9chvymq8926)

[Users](#_pp1yd3y9o50k)

[Messages](#_lvlcczkf3tlr)

[Prerequisites](#_n8sghbhoskxf)

[Installation](#_3qb6y0spug8a)

[1. Install node globally:](#_6wl85ahvm8zw)

[2. Install Angular CLI globally (if not already installed):](#_w1di34iovpom)

[3. Clone this repository or create a new Angular project:](#_u7rax52t9ohh)

[4. Install dependencies, including Firebase:](#_x9v3d14rg5uq)

[5. Set up Firebase:](#_ijcf3stof6yy)

[Project Structure](#_iisa9apnbki8)

[Development Server](#_iw80eidwa5iy)

[1. Set Up Firebase Project:](#_tm3wcdhiwiby)

[2. Create Firestore Database:](#_pk8hphzfk2i5)

[3.Configure Firestore Security Rules (Development Example)](#_tn8ikbmljbri)

[4.Configure firebase-config.ts](#_2yefjeetn0se)

[5.(Optional) Create Initial Collections & Documents:](#_odkbi0no3ddx)

[Authentication](#_nsk0szekf1l)

[Additional Features](#_ufi28rjhfnxz)

[Possible Enhancements](#_mpejl2dtpgh6)

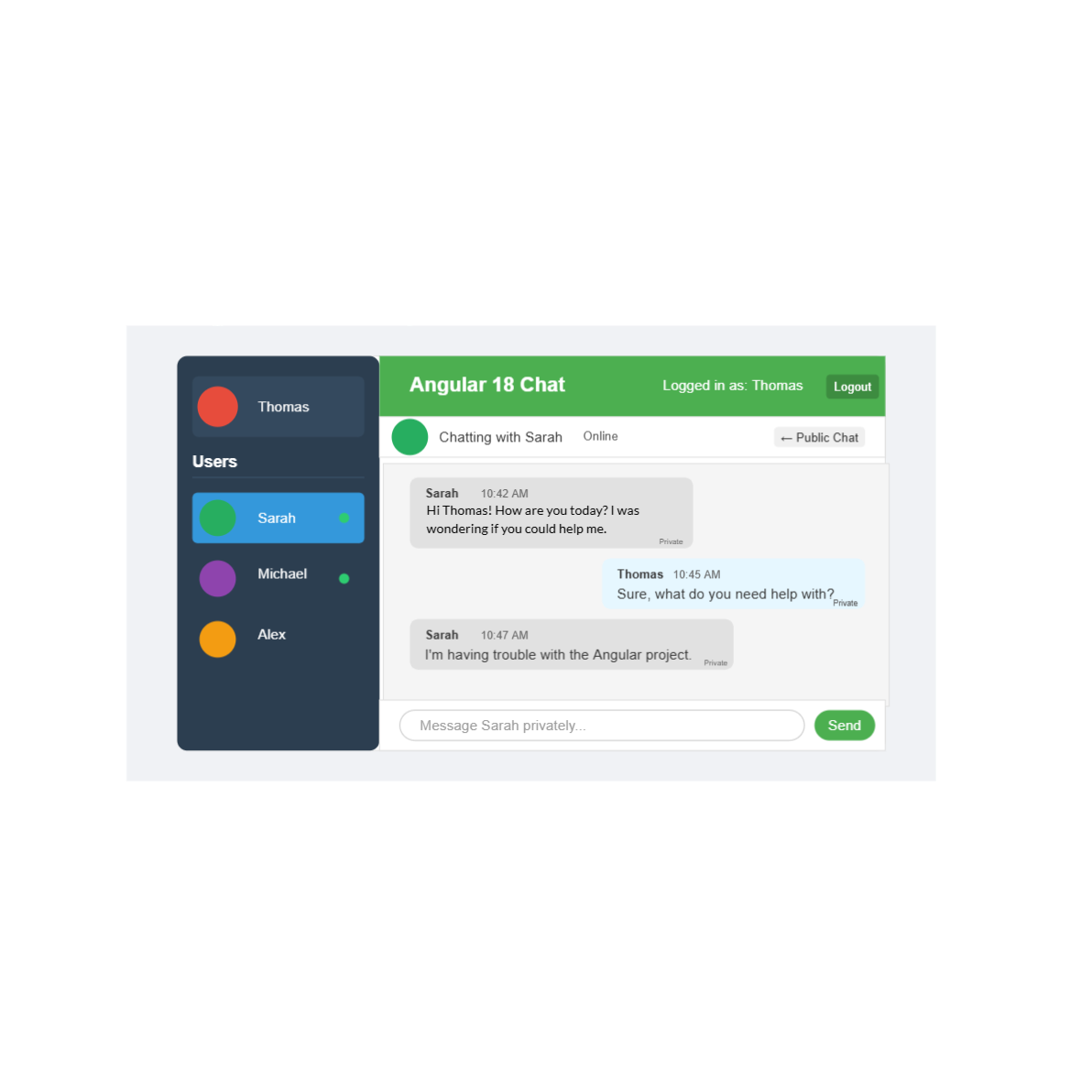
[Angular 18 Features Used](#_14a1vsckvzbx)

# 

Angular 18 Chat Application with Firebase

A real-time chat application built with Angular 18 and Firebase Firestore featuring both public and private messaging.

The following is a visual representation of the intended application functionality.



# Features

* User authentication with user selection or creation
* Public chat room for all users
* Private messaging between users
* Real-time message updates with Firebase Firestore
* User online/offline status tracking
* Clean, responsive UI with status indicators
* Message timestamps and read receipts
* Modern UI with animations
* Auto-scroll to latest messages
* Standalone components architecture

# How It Works

## User Authentication

* Users can create a new profile or select an existing user.
* User status (online/offline) is tracked in real time.

## Public Chat

* By default, all messages are sent to the public chat.
* All online users can see these messages.

## Private Messaging

* Click on a user in the sidebar to start a private conversation.
* Messages are only visible to the sender and recipient.
* Private messages are indicated with a unique style.

## Real-time Updates

* All messages and user statuses update instantly across all clients.
* Firebase Firestore provides automatic synchronization.

# Data Structure

The application uses Firestore with two main collections:

## Users

* Stores user information:
  + User name
  + Online status
  + Last active timestamp
  + Avatar color

## Messages

Stores all message data, including:

* Content of the message
* Sender's name
* Recipient's name (or "everyone" for public messages)
* Timestamp
* Privacy status (private or public)

# Prerequisites

* Node.js v18.x or later
* npm v9.x or later
* A Firebase account (free tier is sufficient)

# Installation

## 1. Install node globally:

Download Node js from: <https://nodejs.org/en/download>  


then install the msi file

to check is node is install properly open cmd and run:

# Verify the Node.js version:

node -v # Should print "v22.14.0".

# Verify npm version:

npm -v # Should print "10.9.2".

## 2. Install Angular CLI globally (if not already installed):

*npm install -g @angular/cli*

## 3. Clone this repository or create a new Angular project:

*# Option 1: Start from scratch*

*ng new angular-chat-app*

cd *angular-chat-app*

*# Option 2: Clone repository (if available)*

*# git clone <repository-url>*

*# cd angular-chat-app*

## 4. Install dependencies, including Firebase:

*npm install*

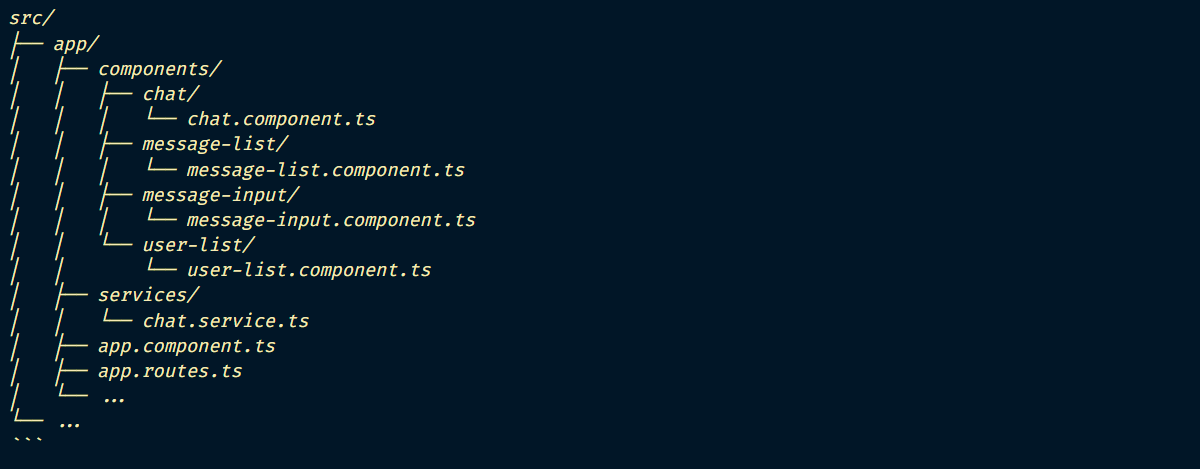
*npm install @angular/fire firebase*

## 5. Set up Firebase:

To get started with your Firebase chat application, follow these steps:

1. Create a new Firebase project.
2. Register a web app within your Firebase project.
3. Enable the Firestore database for your project.
4. Copy your Firebase configuration details into the 'src/app/firebase-config.ts' file.
5. Set up the necessary Firestore collections: 'users' and 'messages'.

# Project Structure

The application is organized as follows:  


# Development Server

Run the development server:

*ng serve*

Navigate to http://localhost:4200/ in your browser. The application will automatically reload if you change any of the source files.

# Production Build

Run the development server:

*ng build --configuration production*

The build artifacts will be stored in the **`dist/`** directory.

# 

# Important Angular subjects to know

**Components & Data Binding**

Core building blocks with property, event, and two-way binding.

**Dependency Injection (DI)**

A Powerful mechanism for managing services and shared logic.

**Routing & Navigation**

Managing app navigation, lazy loading, and route guards.

**Reactive Forms**

For complex, scalable form handling and validation.

**HTTP Communication**

Handling API calls, interceptors, and error handling.

**RxJS & Observables**

Reactive programming with streams and operators.

**Performance Optimization**

Change detection strategies, lazy loading, OnPush, and best practices.

*\*\*\*Note: This list covers the essentials, but Angular has a wide range of additional features for more advanced use cases and features.*

Firebase Firestore Setup for Angular 18 Chat App

## 1. Set Up Firebase Project:

1. Go to [Firebase Console](https://console.firebase.google.com/)
2. Click **“Add project”** and follow the setup wizard
3. Once your project is created, click **“Web” (</>)** to add a web app
4. Register the app (e.g., angular-chat-app)
5. Copy the **Firebase configuration object**

## 2. Create Firestore Database:

1. In Firebase Console, navigate to **Firestore Database**
2. Click **“Create database”**
3. Choose either **Production mode** or **Test mode** (for development)
4. Select your **location**
5. Click **“Enable”**

## 3.Configure Firestore Security Rules (Development Example)

Navigate to: **Firestore Database > Rules**, and apply:

*rules\_version* *=* '2';

*service* *cloud.*firestore {

*match* */databases/*{*database*}*/documents* {

*// Messages Collection*

*match* */messages/*{messageId} {

*allow* *read:* *if* true;

*allow* *write:* *if* true; *// Change to: request.auth != null;*

}

*// Users Collection*

*match* */users/*{userId} {

*allow* *read:* *if* true;

*allow* *write:* *if* true; *// Change to: request.auth != null;*

}

}

}

## 4.Configure firebase-config.ts

Create or update the firebase-config.ts file in your Angular project:

*export const firebaseConfig = {*

*apiKey:* "YOUR\_API\_KEY"*,*

*authDomain:* "YOUR\_PROJECT\_ID.firebaseapp.com"*,*

*projectId:* "YOUR\_PROJECT\_ID"*,*

*storageBucket:* "YOUR\_PROJECT\_ID.appspot.com"*,*

*messagingSenderId:* "YOUR\_MESSAGING\_SENDER\_ID"*,*

*appId:* "YOUR\_APP\_ID"

*}*;

## 5.Create Initial Collections & Documents:

**Collection:** messages  
 Example Document:

{

"text": "Welcome to the chat app!",

"sender": "System",

"timestamp": "<server timestamp>"

}

### Add indexing to “message” Collection:

1. In Firebase Console, navigate to **Firestore Database**
2. Goto, Indexes tab
3. Click on “Add Index”
4. Click on “Create Index”

**Collection:** users

Example Document:

{

"name": "Alice",

"isOnline": true,

"lastActive": "<server timestamp>"

}

Extending the Application

# Authentication

The current implementation does not include user authentication. To add authentication:

1. Enable Authentication in your Firebase project.
2. Implement login/signup components.
3. Update Firestore security rules to require authentication.
4. Add user profile management.

# Additional Features

You can extend this application with:

## Possible Enhancements

* Message read receipts
* Typing indicators
* Image and file sharing
* User profiles
* Group chats
* Message search functionality
* Notifications
* Emoji support

# Angular 18 Features Used

* Standalone components (default in Angular 18)
* Simplified imports
* Modern control flow with @for, @if, etc.
* Improved performance with signals (can be implemented)
* Lightweight architecture