



Venkatesh Mogili

V M



Venkatesh Mogili
#WebGuru

తెలుగు లో



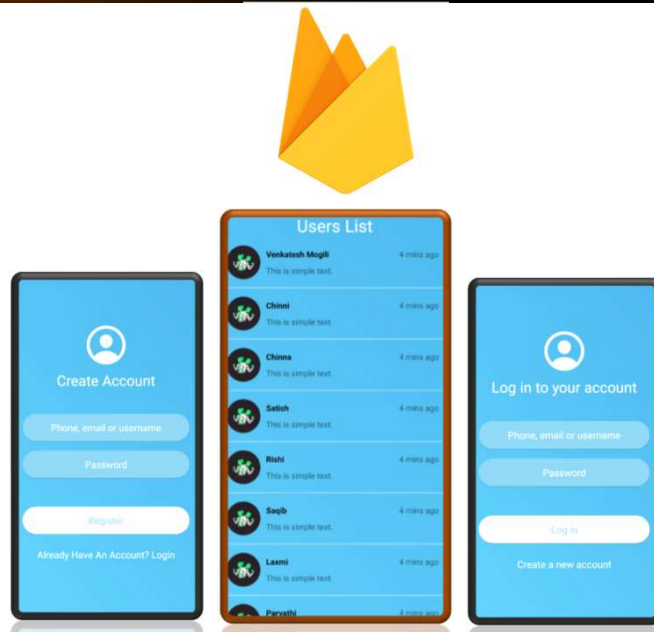
PART-6

React Native Crash Course

Firebase Authentication, Realtime Database

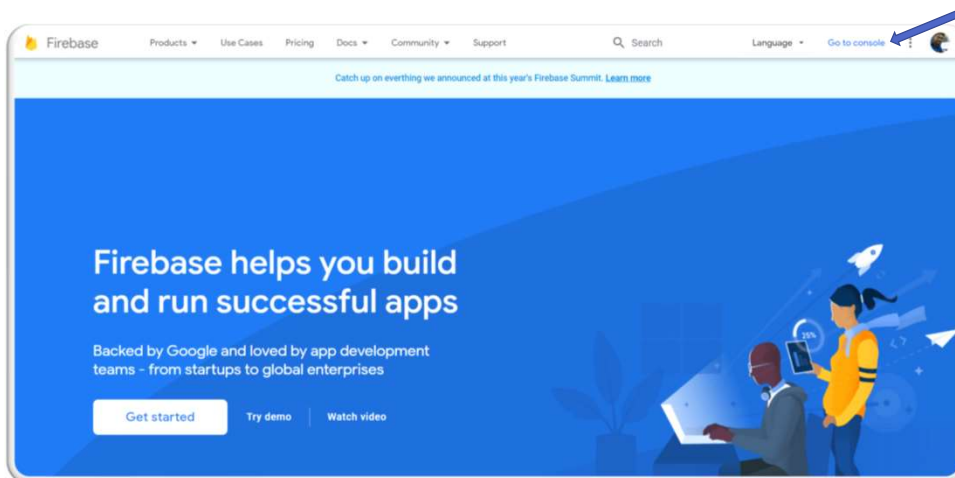
Contents

- Firebase Setup
- Firebase Configuration
- Firebase Realtime Database
- Registration
- Login
- Users List
- CRUD Operations



Firestore Setup

1. create a project in a firestore console. (<https://firebase.google.com/>)

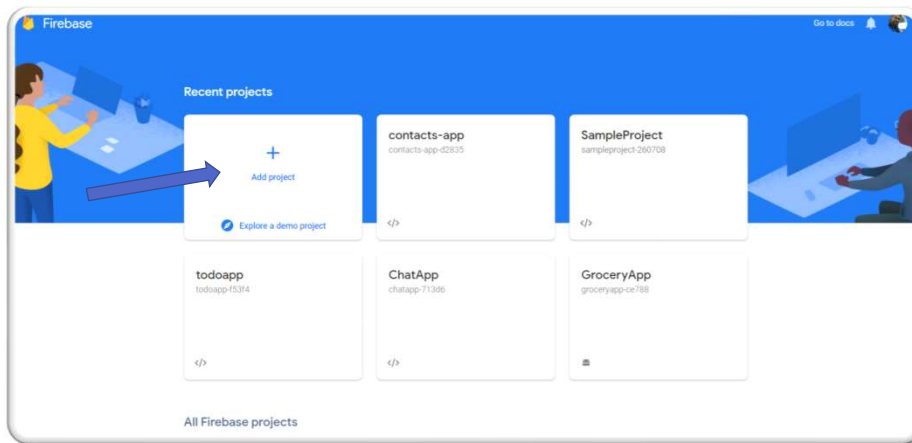




Firebase Setup



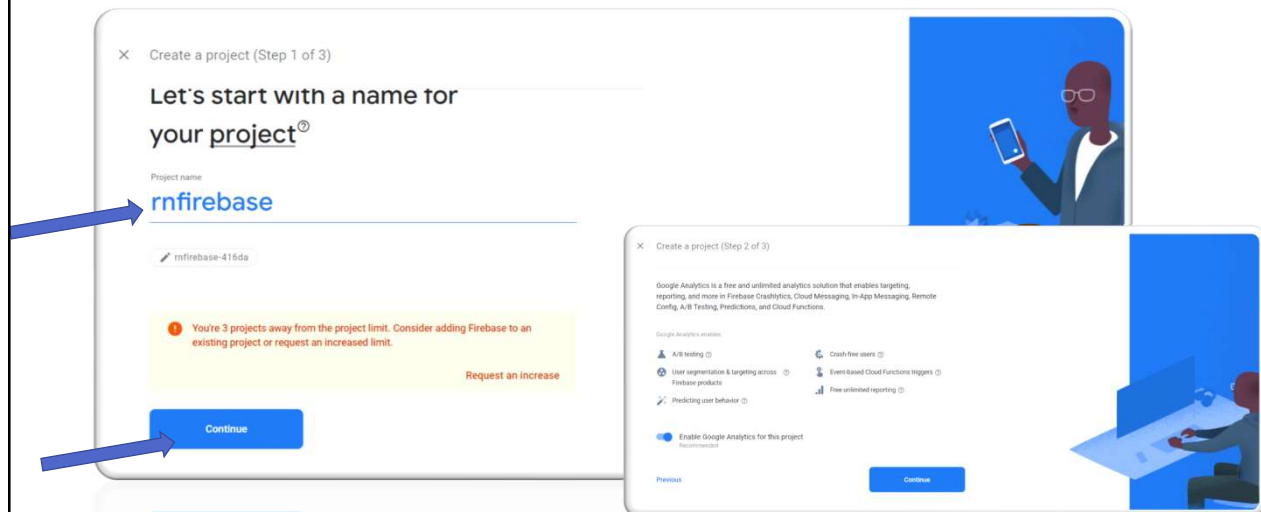
2. Create a **new project** if you do not have already otherwise click on any **existing project** you want to use. (<https://console.firebase.google.com/>)




Firebase Setup




3. Provide project **name** and click **continue**.

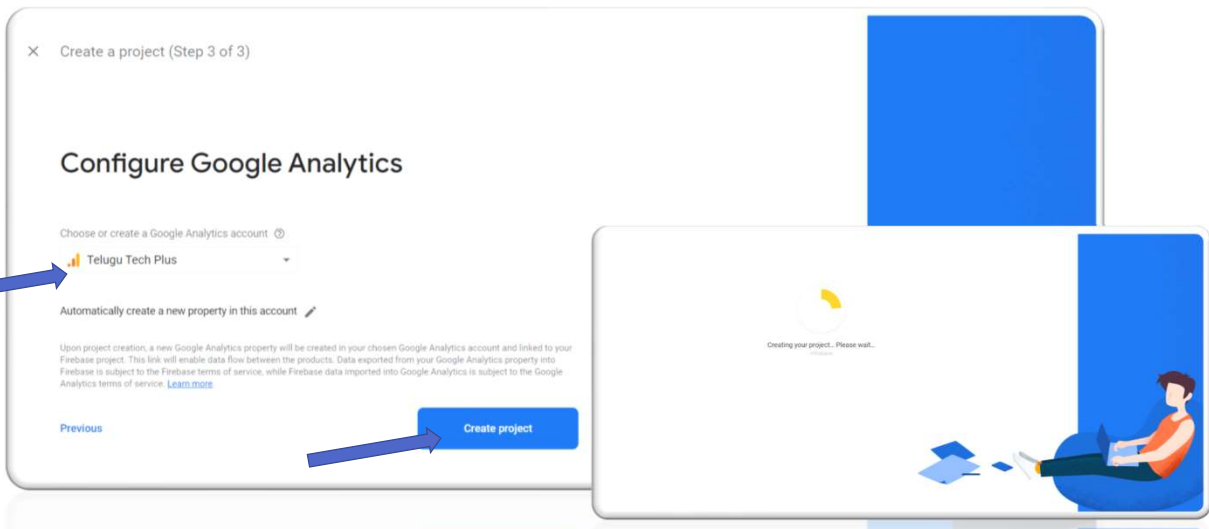





Firebase Setup




4. **Select** your account click **Create project**.

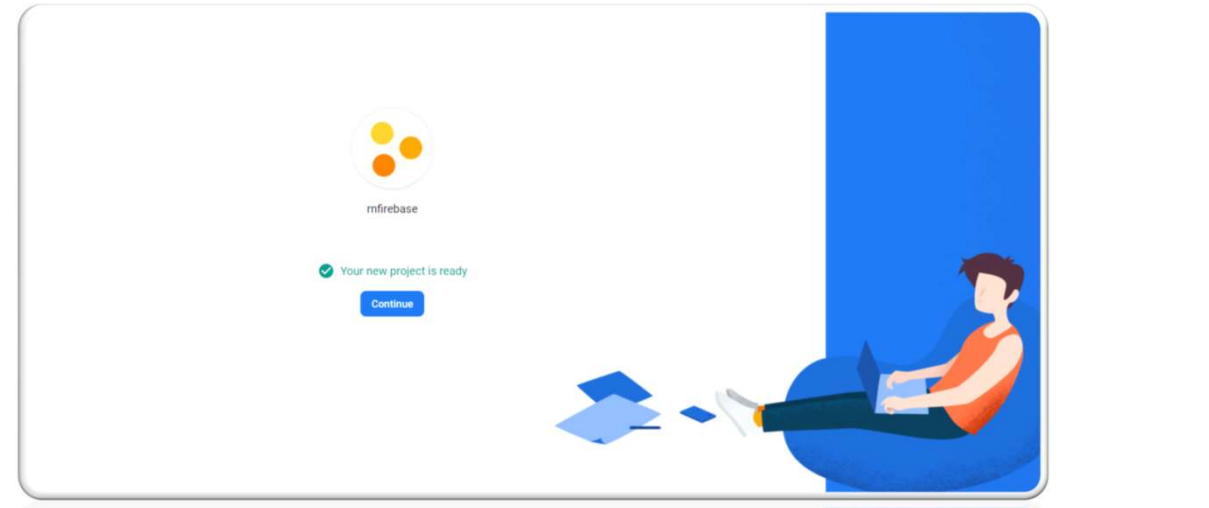




Firebase Setup

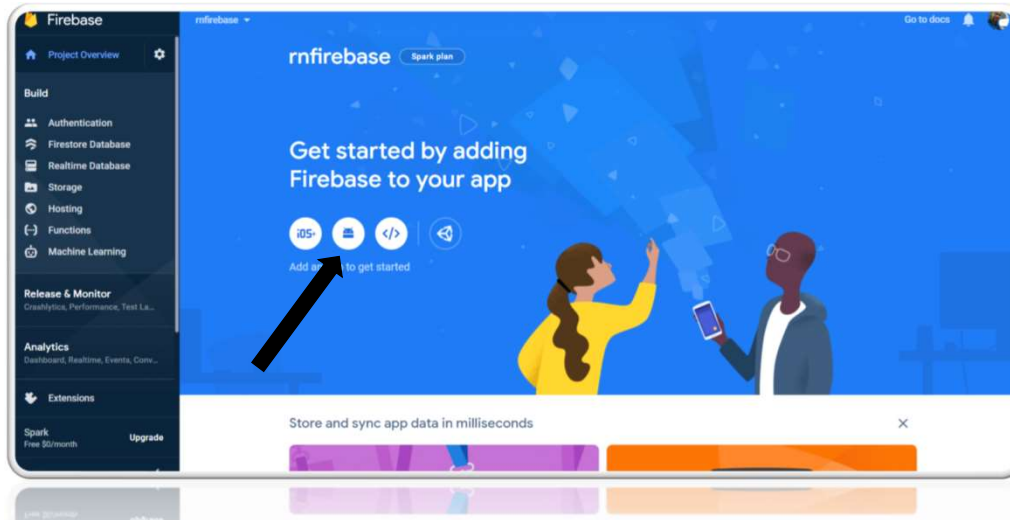


5. **Wait** for few minutes, project will be created and click **continue**.



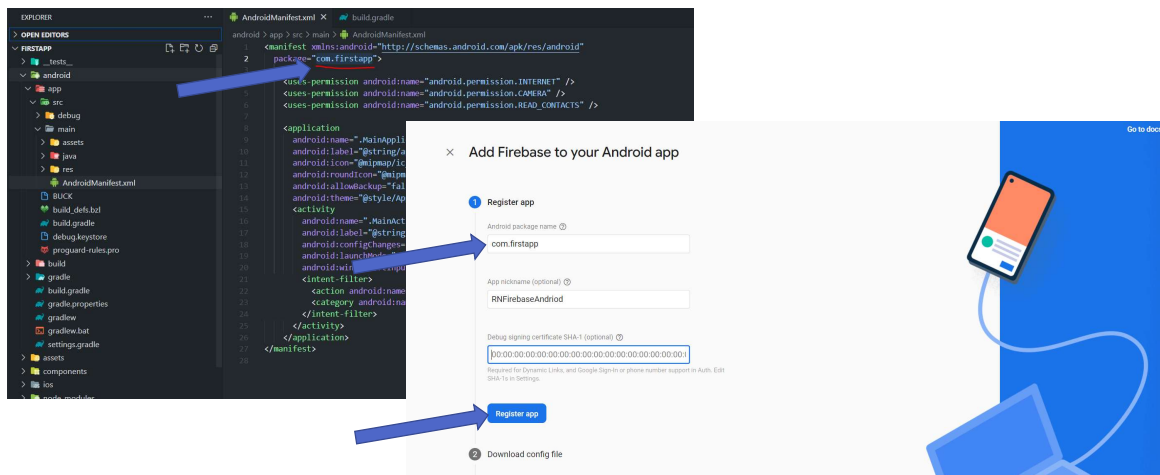
Firebase Configuration

1. Once project is created, you can see the dashboard, **select Android**.



Firebase Configuration

2. Fill the **Android package name**. That will be available in **AndroidManifest**



Firebase Configuration

3. Download **google-services.json** and paste in **android/app** folder.

The screenshot shows the 'Add Firebase to your Android app' wizard in Android Studio. The 'Download config file' step is active, with a blue arrow pointing to the 'Download google-services.json' button. Another blue arrow points to the 'Next' button. To the right, the Android Studio project structure is visible, with a blue arrow pointing to the 'google-services.json' file in the 'android/app' directory.

Firebase Configuration

4. Add classpath 'com.google.gms:google-services:4.3.10' in **build.gradle**

The screenshot shows the 'Add Firebase SDK' wizard in Android Studio. The 'Project-level build.gradle (<project>/build.gradle):' section is visible, with a blue arrow pointing to the 'classpath' line in the 'dependencies' block. The code is as follows:

```
buildscript {
    repositories {
        // Check that you have the following line (if not, add it):
        google() // Google's Maven repository
    }
    dependencies {
        ...
        // Add this line
        classpath 'com.google.gms:google-services:4.3.10'
    }
}

allprojects {
    repositories {
        // Check that you have the following line (if not, add it):
        google() // Google's Maven repository
    }
}
```



Firebase Configuration



5. Add **apply plugin: 'com.google.gms.google-services'** in *app/build.gradle*

The screenshot shows an IDE window with the `app/build.gradle` file open. The file contains the following code:

```
apply plugin: 'com.android.application'
// Add this line
apply plugin: 'com.google.gms.google-services'

dependencies {
    // Import the Firebase BoM
    implementation platform('com.google.firebase:firebase-bom:29.0.1')

    // Add the dependency for the Firebase SDK for Google Analytics
    // When using the BoM, don't specify versions in Firebase dependencies
    implementation 'com.google.firebase:firebase-analytics'

    // Add the dependencies for any other desired Firebase products
    // https://firebase.google.com/docs/android/setup#available-libraries
}
```

A blue arrow points to the `apply plugin: 'com.google.gms.google-services'` line. Below the code, a message says "Finally, press 'Sync now' in the bar that appears in the IDE." and there is a "Sync now" button. At the bottom, there are "Previous" and "Next" buttons.

Overlaid on the right is the "Add Firebase to your Android app" dialog. It shows a progress list with four steps: "Register app", "Download config file", "Add Firebase SDK", and "Next steps". The "Next steps" step is currently selected. Below the list, there is text: "You're all set! Make sure to check out the [documentation](#) to learn how to get started with each Firebase product that you want to use in your app. You can also explore [sample Firebase apps](#). Or, continue to the console to explore Firebase." At the bottom of the dialog, there are "Previous" and "Continue to console" buttons. A large green checkmark is positioned above the dialog.




Firebase Configuration




6. Install react-native-firebase packages and run the project.

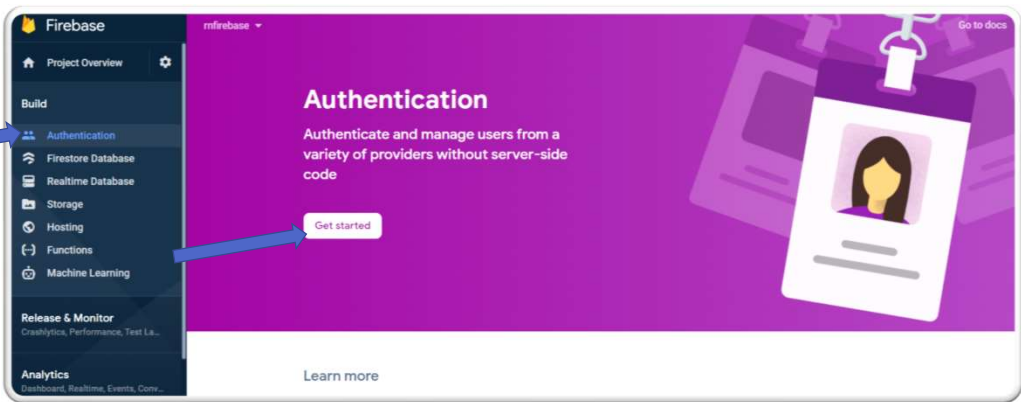
- **npm install @react-native-firebase/app**
- **npm install @react-native-firebase/database**
- **npm install @react-native-firebase/auth**




Firebase Authentication




1.Select Authentication and Click Get Started.



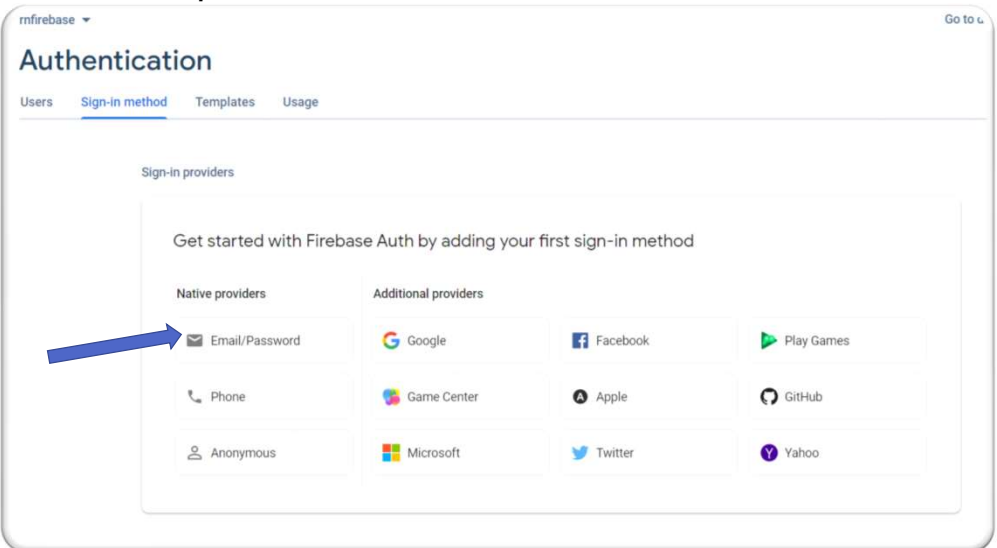
The screenshot shows the Firebase console interface. On the left, a dark sidebar contains a menu with categories: 'Project Overview', 'Build', 'Release & Monitor', and 'Analytics'. The 'Build' category is expanded, showing sub-items like 'Authentication', 'Firestore Database', 'Realtime Database', 'Storage', 'Hosting', 'Functions', and 'Machine Learning'. A blue arrow points to the 'Authentication' link. The main area has a purple header with the title 'Authentication' and a subtitle 'Authenticate and manage users from a variety of providers without server-side code'. Below this is a 'Get started' button, which is also pointed to by a blue arrow. There is also a 'Learn more' link at the bottom.



Firebase Authentication



1. Select Email/Password method.



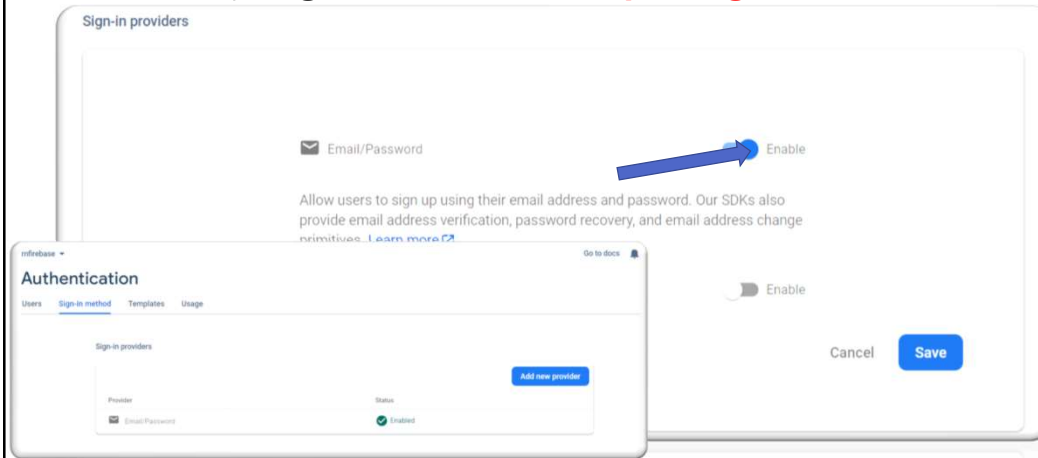
The screenshot shows the 'Sign-in method' page in the Firebase Authentication console. The page has tabs for 'Users', 'Sign-in method', 'Templates', and 'Usage'. The 'Sign-in method' tab is active. Below the tabs, there's a section titled 'Sign-in providers' with a sub-header 'Get started with Firebase Auth by adding your first sign-in method'. This section is divided into two columns: 'Native providers' and 'Additional providers'. In the 'Native providers' column, the 'Email/Password' provider is selected, indicated by a blue arrow. Other providers in this column include 'Phone' and 'Anonymous'. The 'Additional providers' column lists various social and third-party providers like Google, Facebook, Play Games, Game Center, Apple, GitHub, Microsoft, Twitter, and Yahoo.



Firestore Authentication



1. Select Email/Password method. If you have CORS chrome extension, disable it, if you get error as **“Error updating Email/Password”**.



Login & Registration



npm i react-native-vector-icons

Log in to your account

Phone, email or username

Password

Log in

Create a new account

Create Account

Phone, email or username

Password

Register

Already Have An Account? Login



Registration using Firebase.



- `import auth from '@react-native-firebase/auth';`
- `const createUser = async (email, password) =>`
`{`
 `try { let response = await auth().createUserWithEmailAndPassword(email, password)`
 `if (response) { console.log(response) } }`
 `catch (e) { console.error(e.message) }`
`}`
- `npm run android`



Login using Firebase.



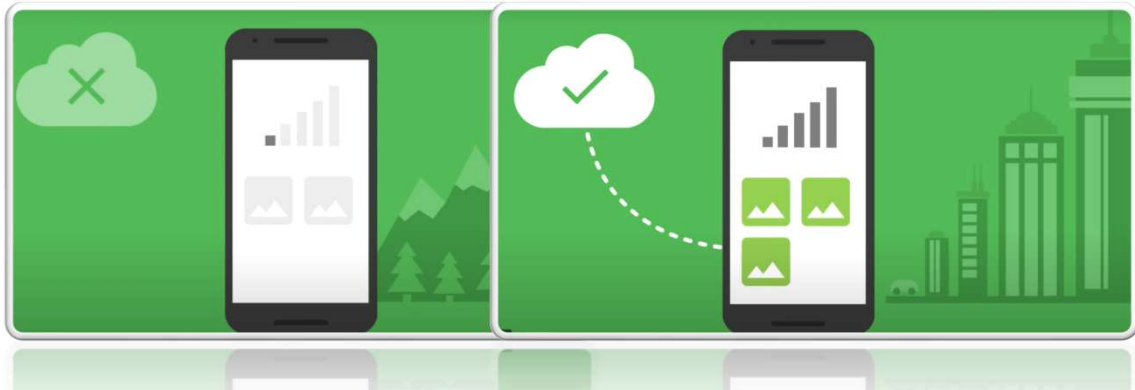
- `import auth from '@react-native-firebase/auth';`
- `const loginUser = async (email, password) =>`
`{`
 `try { let response = await auth().signInWithEmailAndPassword(email, password)`
 `if (response) { console.log(response) } }`
 `catch (e) { console.error(e.message) }`
`}`



What is Firebase Realtime Database?



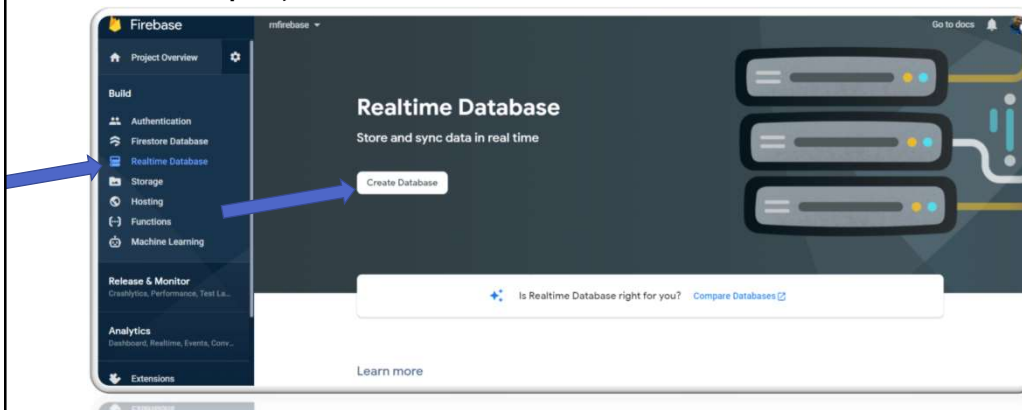
- Firebase **Realtime Database** will sync with the data across all the devices even if there is **no network** connection by using **cache method** and sync with the **data** when **network is available**.




Firebase Realtime Database Setup




- Go to the Firebase Console, and click the **Realtime Database** from the menu bar. If you are creating a realtime database for the first time in your Firebase project, click the **Create Database** button.

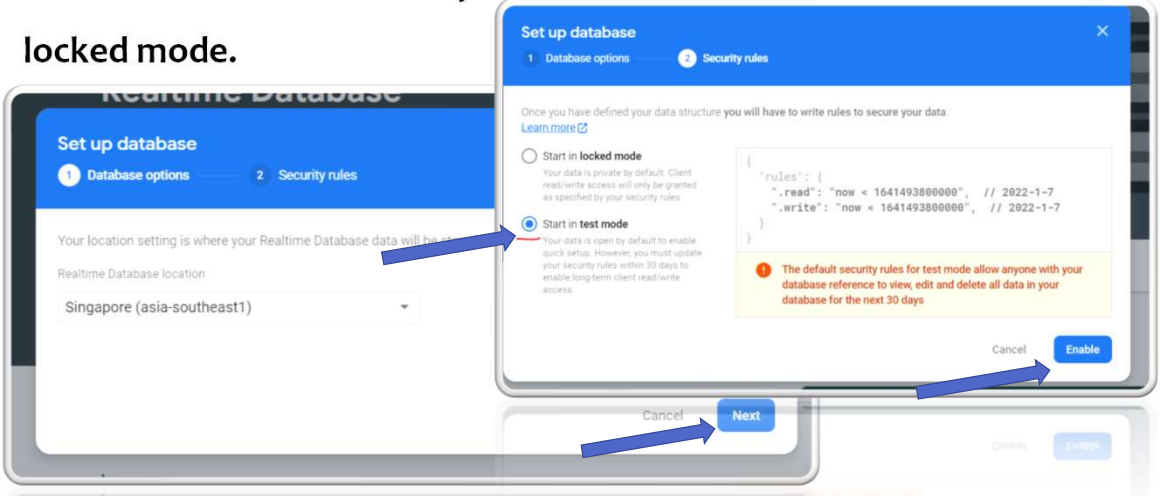





Firestore Realtime Database Setup




➤ Select **database location** any of your choice and select **test mode** or **locked mode**.

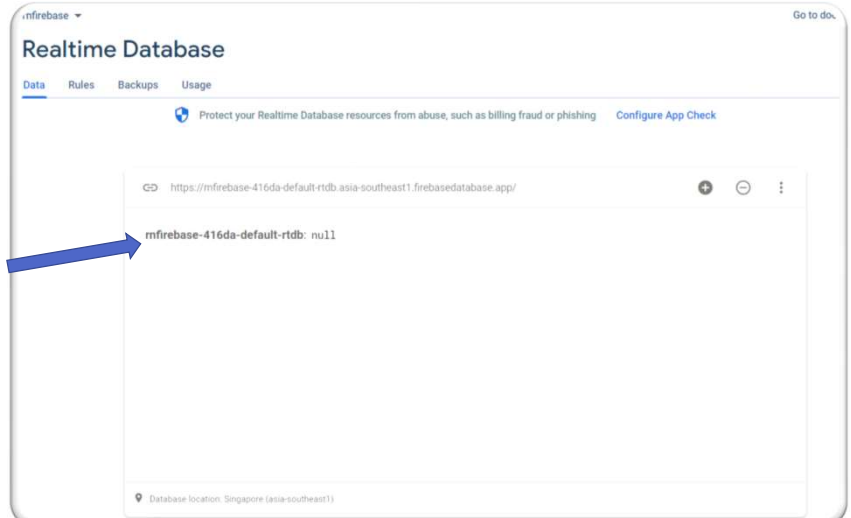




Firestore Realtime Database Setup



➤ Database will be created like this.





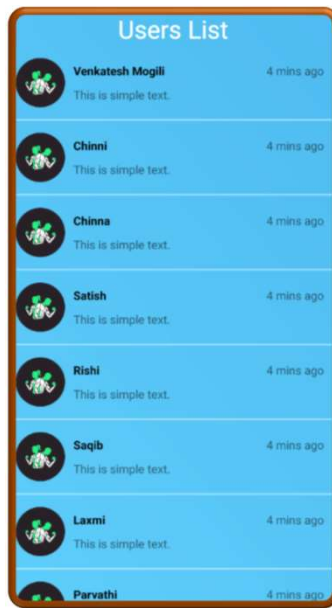
Registration using Firebase Realtime Database



- `import {firebase} from '@react-native-firebase/database';`
- `let addUser = user => {
 firebase.app().database("database_url").ref('/users').push({ email:
 user.email, password: user.password}); };`
- Reference: <https://rnfirebase.io/database/usage>



Users List

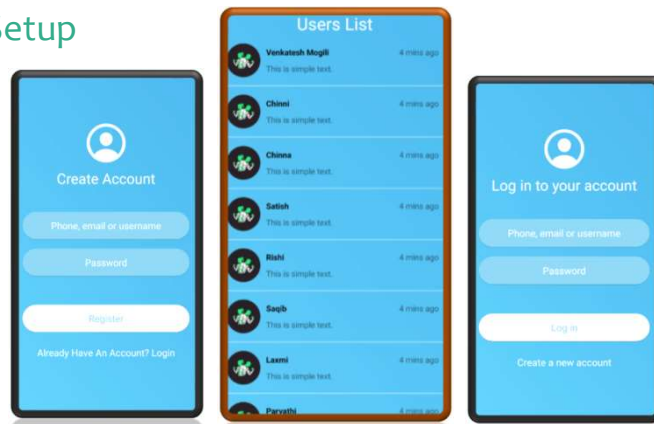




Summary



- ✓ Firebase Setup
- ✓ Firebase Configuration
- ✓ Firebase Realtime Database Setup
- ✓ Registration
- ✓ Login
- ✓ Users List
- ✓ CRUD Operations



Thank you and Don't forget to Subscribe



Check out other courses



<https://vmtraining.netlify.app>



Watch



Implement



Try Experiments

VM Online Training