



Get unlimited access

Open in app



Published in Databox Technologies



Nuwan Abeywickrama

Follow

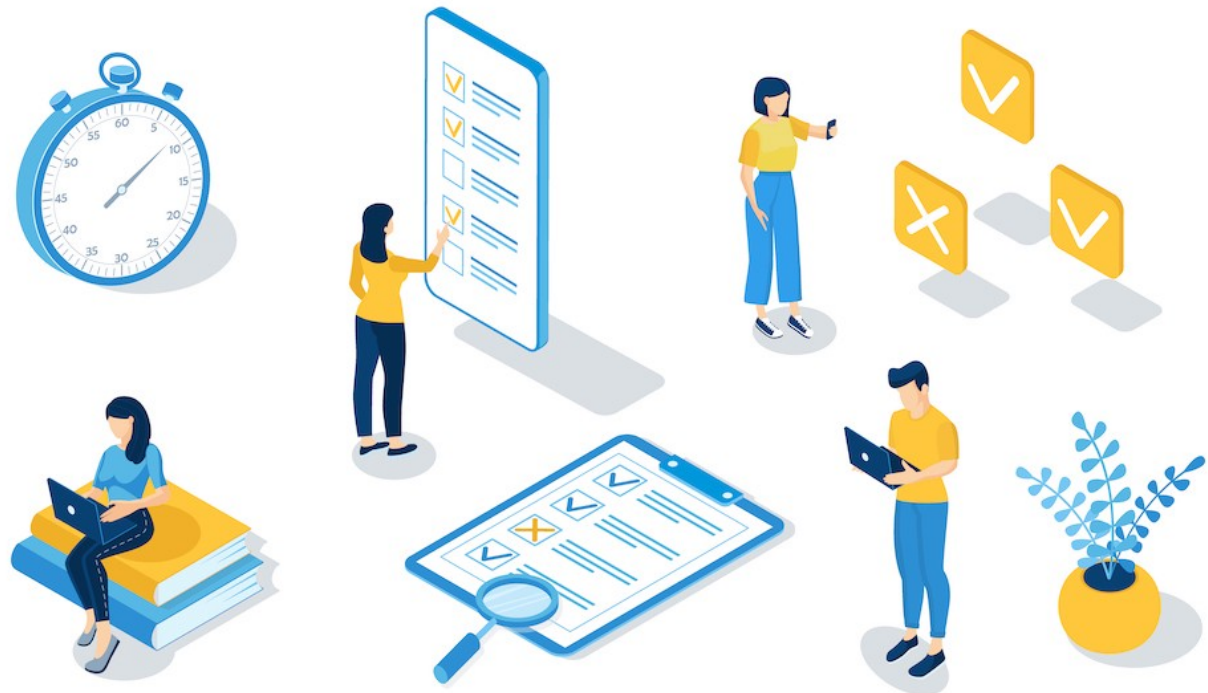
Mar 16, 2021 · 8 min read · Listen



Save



Creating Quiz Game App Using IONIC framework.



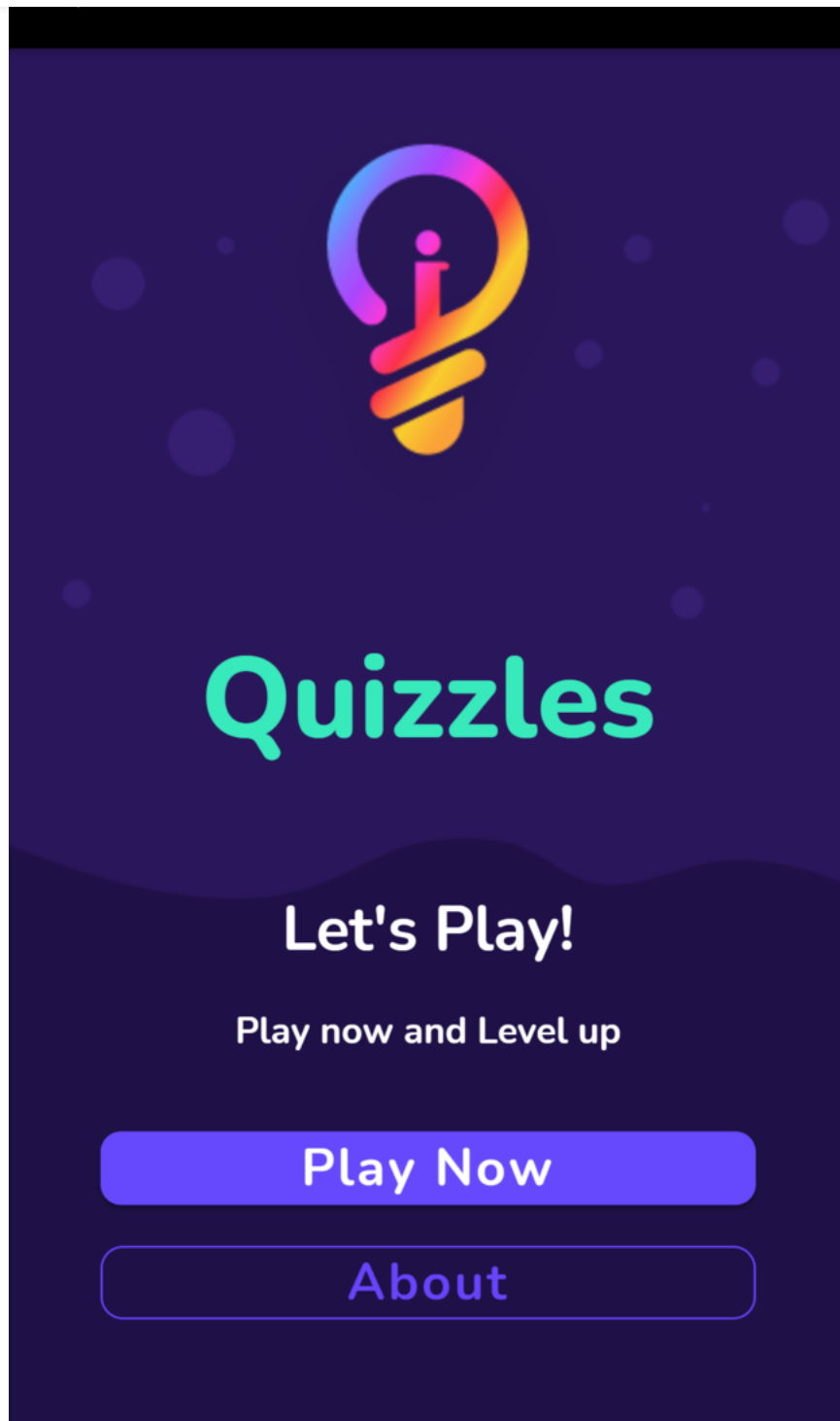
This is a journey about one of my project in my internship period which is a creating Android quiz game App using Ionic framework. I wanted to share my story of working on this app so it would help other developers who takes on the same journey. So, I hope this post helps and inspires other programmers who want to create android app using ionic.





Get unlimited access

Open in app



Intro for quick start

As First step, You must refer [Ionic Framework Documentation](#) and select one of web application framework such as [Angular](#), [React](#) or [Vue](#). In my project I used Angular with ionic and VS Code as code editor.

Using above documentation you will get idea how to setup environment for ionic development. After setup environment completely you can start continue with the article.

Before design and make logics you must create a new ionic project. You can use below command on your terminal/cli to do so.

ionic start myApp tabs

You can use your app name instead “myapp”. Then your terminal shows like this **Type Y at the prompt Would you like to integrate your new app with Cordova to target native iOS and Android? (y/N)** in here type y and press enter key. After run command below I





After run this commands, If your setup and other installations completed correctly you can show your sample app open in the browser. Some times your browser will not open automatically don't worry about that copy link address in your terminal and past in your browser. You can see your ionic app in the browser. Now, you have ionic app and you can develop it as you need..

Create Quiz App

We are building this app for Android OS to start with. So first add android platform using command "***cordova platform add android***". Before starting developments take few minutes to read below. <https://ionicframework.com/docs/developing/android> It will resolve most of the problems you might get. Important thing is you must separate your project to pages, components, modules and Services.

Now you have to make specific folders to separate your code very clean. For that go to **src>app** and create folder called "models". In same directory go to **tab1** and open integrated terminal to that directory. Use commands in below

ionic generate component start

ionic generate component view-question

ionic generate page user-level

ionic generate page view-quiz

In this game I used images to levels and all of these images in assets folder. following this directory open assets folder **src>assets** . Now you have to make folders called fonts, icon, images, quizzes, levels, in this assets folder. Font folder has all of external fonts that use in app. Icon folder has images such as app icon, logo like that. Images folder has level images, background images etc. Quizzes folder has json files and All of these json files have all of information related to each level questions. User folder has userData json file which has information about each level such as level id, level name, level image link.

In models folder I used different models for different tasks. It will be different from person to person within their experience.

Lets move to Service folder which is the most valuable things in this app. In service folder we can use to store our CRUD operations and any other operations which wants to different places in our application. In this application I used firebase Realtime database for store users meta data such as User's device id, current level, completed levels with marks like that. For using firebase, we must configure app with firebase.

1. Install Angular Fire and Firebase

You should connect your ionic application with the firebase database. install Angular Fire and Firebase from npm by running the following command.

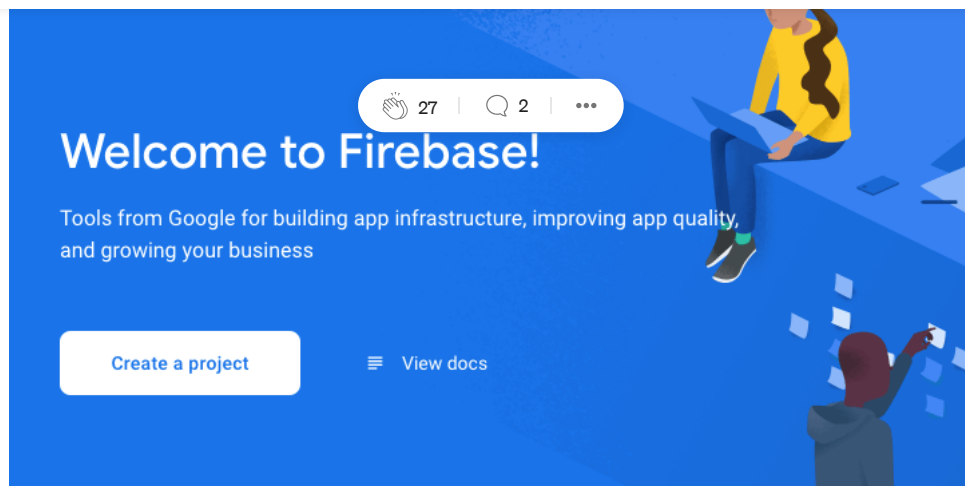
npm install @angular/fire firebase - -save

As a next step you have to config your app with firebase.

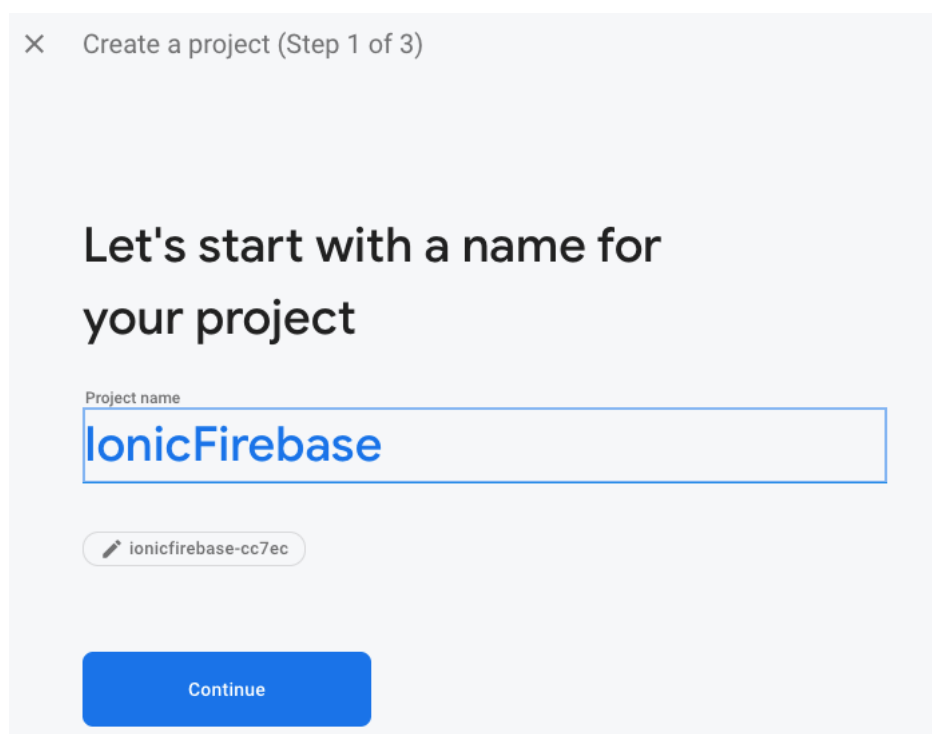
2. Create Firebase Account

Visit console.firebase.google.com and sign in using your Gmail account.



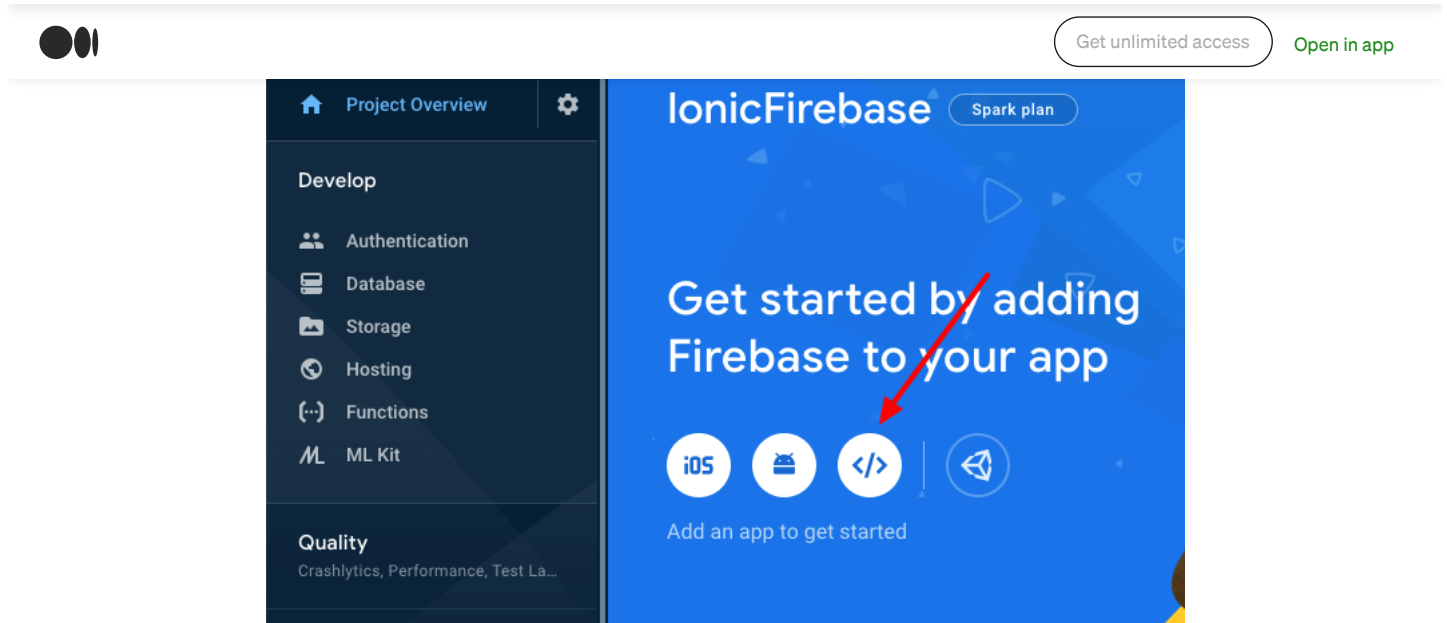
[Get unlimited access](#)[Open in app](#)

Click on the “create a project” button then click on continue button.

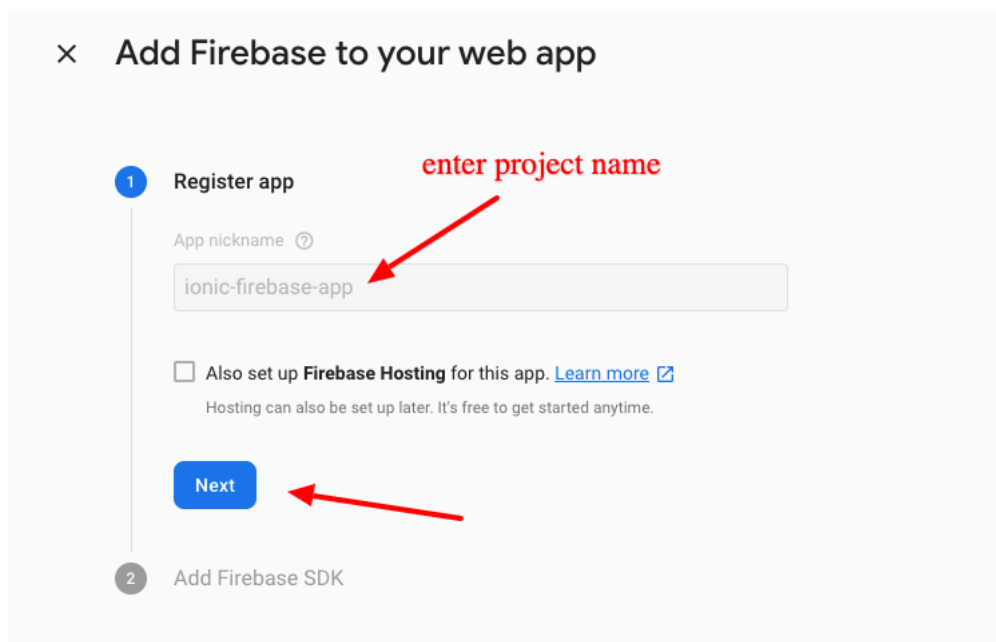


Next, click on the web icon as shown in the screenshot.





Next, add Firebase to your web app. Enter the app's nickname and then click on the next button.



It will take you to the screen where you will see Firebase configuration, copy the red marked Firebase configuration keys keep it in the notepad or something else. You will need these keys to register in your Ionic project.



Get unlimited access

Open in app

2 Add Firebase SDK

Copy and paste these scripts into the bottom of your <body> tag, but before you use any Firebase services:

```
<!-- The core Firebase JS SDK is always required and must be listed first -->
<script src="https://www.gstatic.com/firebasejs/7.5.0/firebase-app.js"></script>

<!-- TODO: Add SDKs for Firebase products that you want to use
https://firebase.google.com/docs/web/setup#available-libraries -->
<script src="https://www.gstatic.com/firebasejs/7.5.0/firebase-analytics.js"></script>

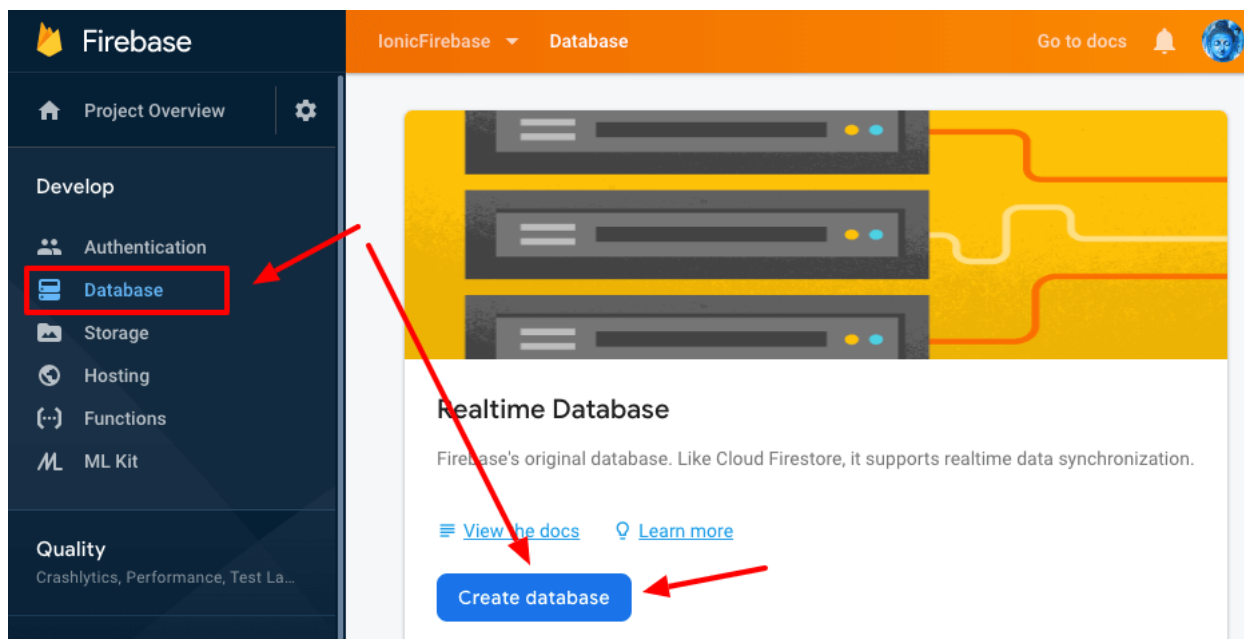
<script>
  // Your web app's Firebase configuration
  var firebaseConfig = {
    apiKey: "AIzaSyAYcgDXJVTa_p6vM_8KvU1JLW4dzbd6mZQ",
    authDomain: "ionicfirebase-cc7ec.firebaseio.com",
    databaseURL: "https://ionicfirebase-cc7ec.firebaseio.com",
    projectId: "ionicfirebase-cc7ec",
    storageBucket: "ionicfirebase-cc7ec.appspot.com",
    messagingSenderId: "224644312062",
    appId: "1:224644312062:web:93bf980415eadc6feaaeaf",
    measurementId: "G-L3F62TP6WF"
  };
  // Initialize Firebase
  firebase.initializeApp(firebaseConfig);
  firebase.analytics();
</script>
```

Firebase configuration

Learn more about Firebase for web: [Get Started](#), [Web SDK API Reference](#), [Samples](#)

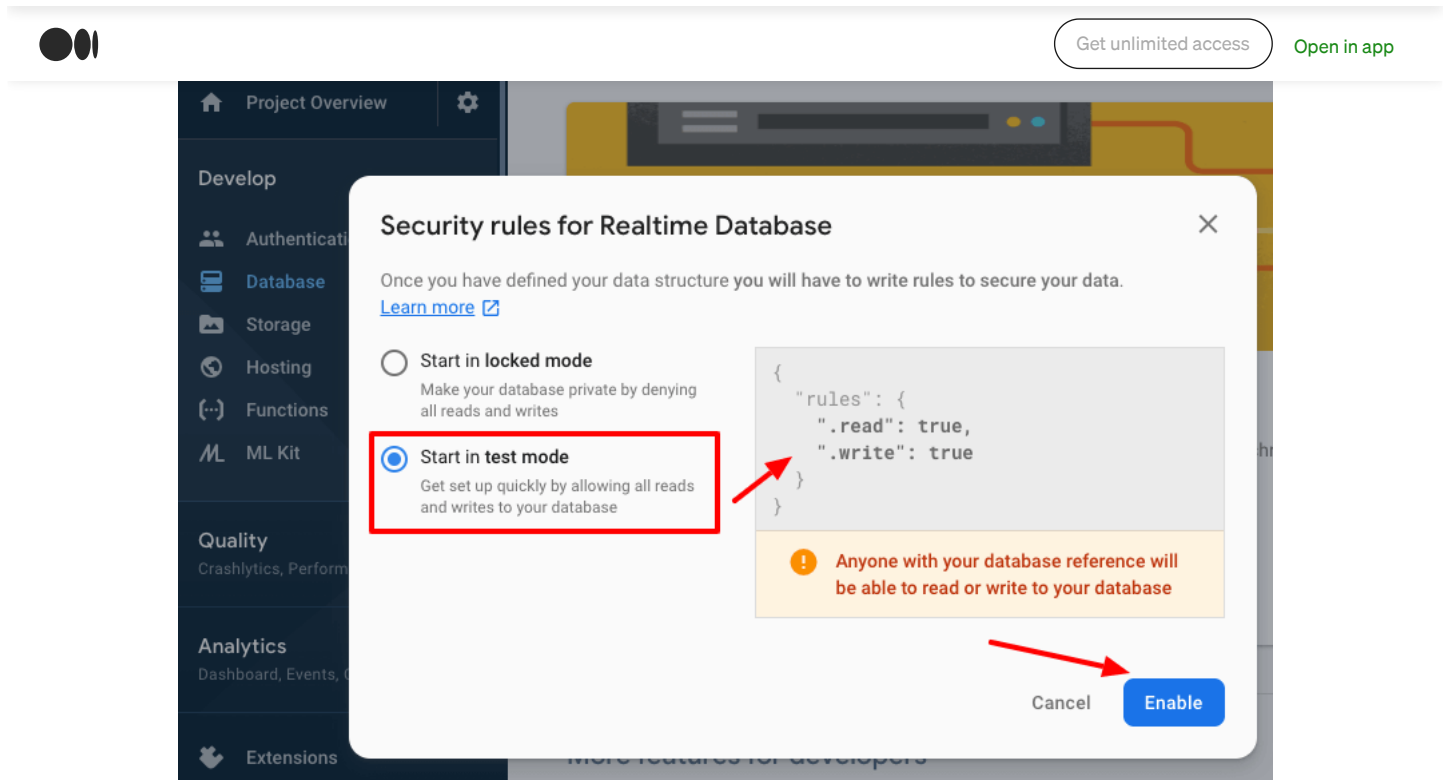
Continue to console

Next, we will click on the “database” from the left side navbar. Then, look for Realtime Database and click on the “create database” button.



It will open the security rules model popover, select “start in test mode” option. Remember we are setting up these rules for testing purpose. In real world app be careful about database rules.





Now, we are all set to use the Firebase Real-time Database.

3. Configure Firebase Config Keys

In this step, we will register the Firebase config rules inside the `environment.prod.ts` and `environment.ts` file, you can find these files in the `src > environments` folder.

Add the following code inside the `environment.prod.ts` file. In previous section I mentioned to keep this details in notepad.

```
export const environment = {
  production: true,
  firebaseConfig: {
    apiKey: "<your-api-key>",
    authDomain: "<your-auth-domain>",
    databaseURL: "<your-database-url>",
    projectId: "<your-cloud-firestore-project>",
    storageBucket: "<your-storage-bucket>",
    messagingSenderId: "<your-sender-id>",
    appId: "<your-app-id>",
    measurementId: "<your-measurement-id>"
  }
};
```

Place the following code inside the `environment.ts` file.

```
export const environment = {
  production: false,
  firebaseConfig: {
    apiKey: "<your-api-key>",
    authDomain: "<your-auth-domain>",
    databaseURL: "<your-database-url>",
    projectId: "<your-cloud-firestore-project>",
    storageBucket: "<your-storage-bucket>",
    messagingSenderId: "<your-sender-id>",
    appId: "<your-app-id>",
    measurementId: "<your-measurement-id>"
  }
};
```



Get unlimited access

Open in app

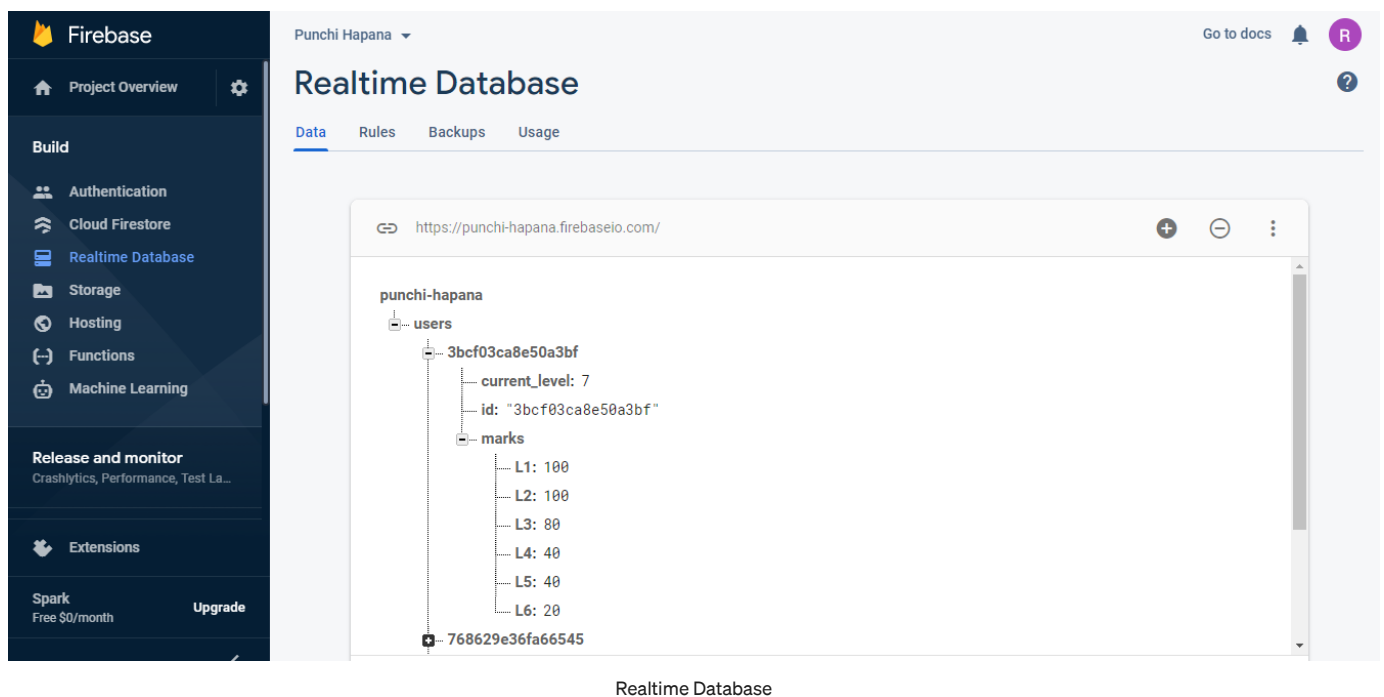
```
// firebase imports, remove what you don't require
import { AngularFireModule } from '@angular/fire';

// environment
import { environment } from '../environments/environment';

@NgModule({
  declarations: [AppComponent],
  entryComponents: [],
  imports: [
    BrowserModule,
    IonicModule.forRoot(),
    AppRoutingModule,
    HttpClientModule,
    AngularFireModule.initializeApp(environment.firebaseConfig),
  ],
  providers: [
    StatusBar,
    SplashScreen,
    { provide: RouteReuseStrategy, useClass: IonicRouteStrategy }
  ],
  bootstrap: [AppComponent]
})

export class AppModule { }
```

Then create database in you Realtime database. Using demo data.



Now, you can connect your Ionic app with Firebase Realtime database.

We want more plugins for our other implementations. There are implementing countdown and get user's device id (UUID). For that following commands.

```
ionic cordova plugin add cordova-plugin-device
```

```
npm install @ionic-native/device
```

```
npm install ngx-countdown — save
```

At the moment now you have to define your CRUD operations and file read operations in side the service folder. First of all you have to make different service files as you need. In here I used three service file called **level**, **quiz**, **userdata**. You can make services like a

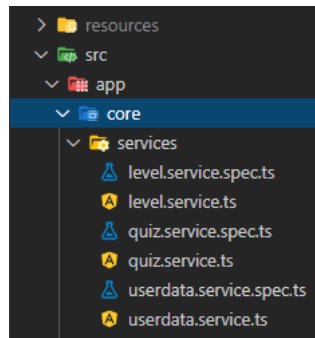




Get unlimited access

Open in app

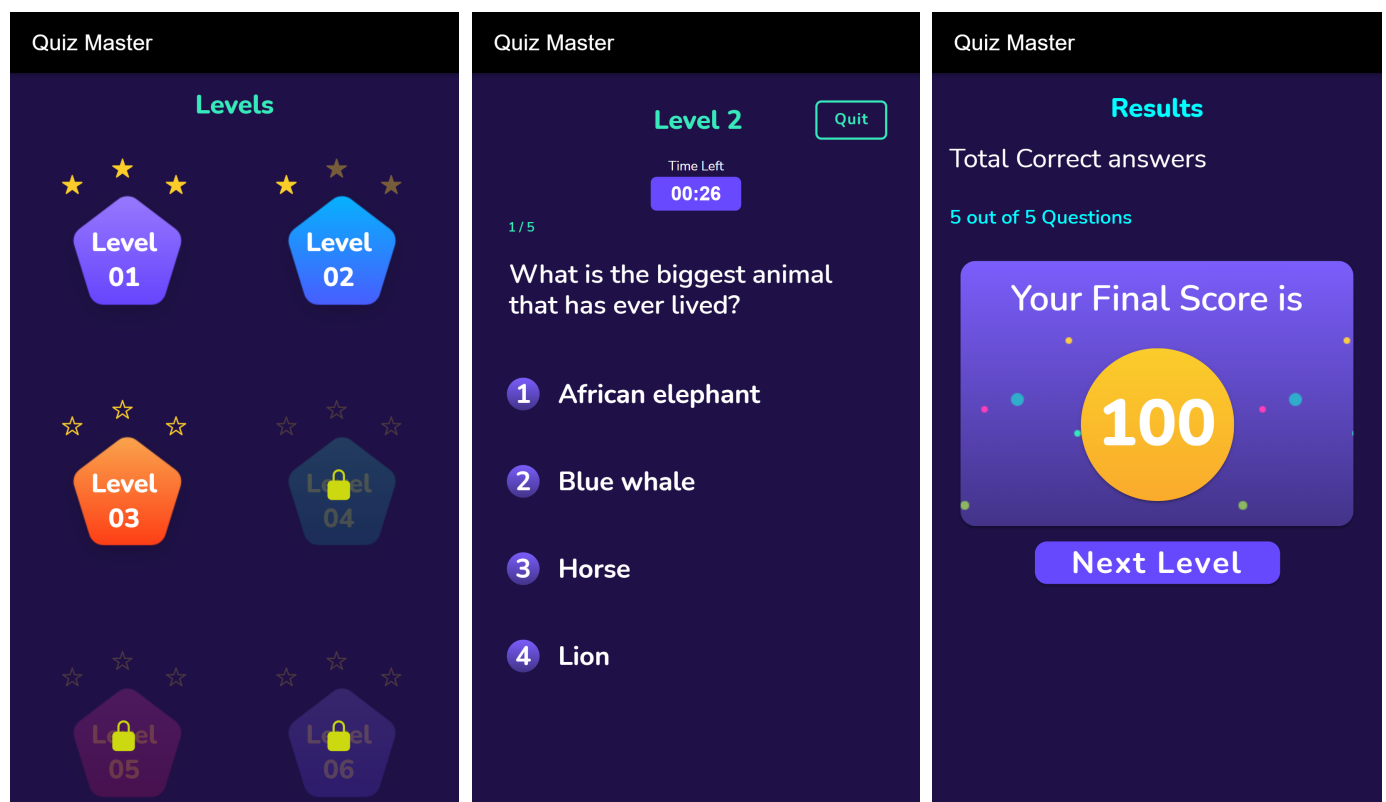
ionic generate service userdata



services folder

In level service, I defined get data from json file that store in inside assets folder. Quiz service is used to get quizzes data from firebase which have all details about each level questions. User data service work with firebase which has lot of implementations such as get all users, update user, update marks, get one user's data like that.

Let's talk about UI and flow of this app. As a first UI I used start component which has two buttons for play and get information about app. Then user navigates to user level page while starting play game. In this page has user levels and show stars related to marks of each levels. Next user go to questions view page and has questions related to each level. Each questions has limited time. That shows top of the questions. After User completed answering all questions in that level shows the marks and provided a feature to try again or go to next level.



Flow of this App

After install and User open app first time, App will get user's device id (UUID). It will help to save user data in firebase. Then user click "play now" button and go to game level screen. If user play game first time, he/she has able to play only level one and other levels are locked. Other levels will be unlock as how user win that level. Each level display three stars related to marks percentage obtained by



[Get unlimited access](#)[Open in app](#)

about each game level question. After user clicks an answer then it will come next question. This process will continue until end of all questions in that level. In final step will calculate user's total marks and will save it in firebase with it's level and display whether if user pass or fail. If user pass the level, can go to next level but user fail the level, must try again that level. After click "next level" or "try again" button again user will go to game level screen. Now, user can look three stars in each level which are completed by user. That three stars displayed related to percentage of marks obtained by user for that level. User can complete all game levels in this game like that.

I attached my GitHub repo link below. Anyone who is interested in making an android app using ionic will be helped to refer for their practice. In this article, I explained special topics related to my app. In this journey, I used a lot of documentations which are on their official sites. I linked all of these websites beginning of this article.

GitHub Link: https://github.com/Ramesh-Nuwan/Quiz_App_ionic

Thank You.

Read more

[How to make smart BMI Calculator App using Flutter](#)

Thanks to Prabath Perera

Get an email whenever Nuwan Abeywickrama publishes.

Emails will be sent to sanogotech@gmail.com. [Not you?](#)



Subscribe

