**Vue-Cordova-Firebase-Push-Notifications**

**Create Cordova and Vue Project:**

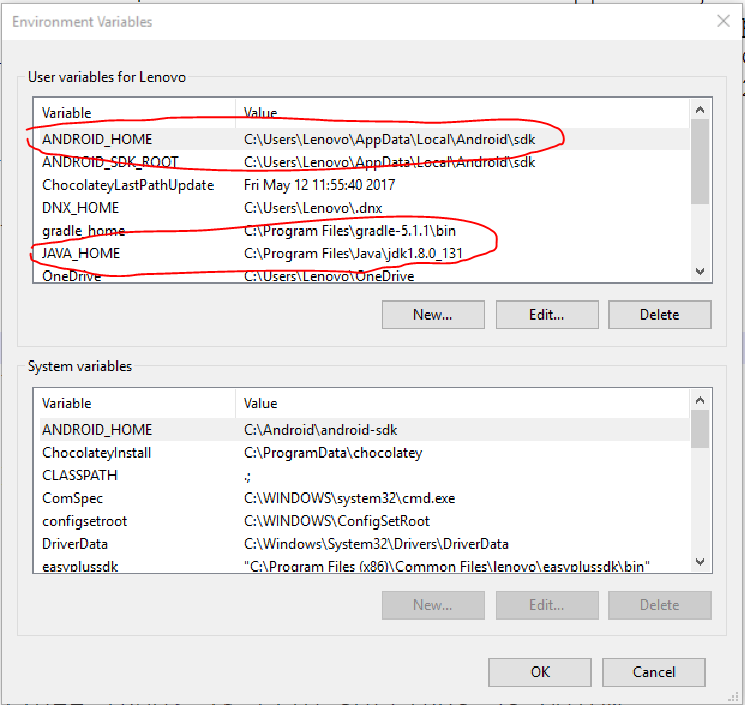
**Step1: First Install Android Studio and JDK file**

<https://developer.android.com/studio/>

<https://www.oracle.com/technetwork/java/javase/downloads/jdk11-downloads-5066655.html>

**Step2: Edit Environment Variables**

Add ANDROID\_HOME and JAVA\_HOME



**Step3: Install node.js , Cordova and Vue CLI**

* <https://nodejs.org/en/>
* **npm install -g cordova**
* **npm install -g vue-cli**

**Step4: Create Cordova-Vue Project**

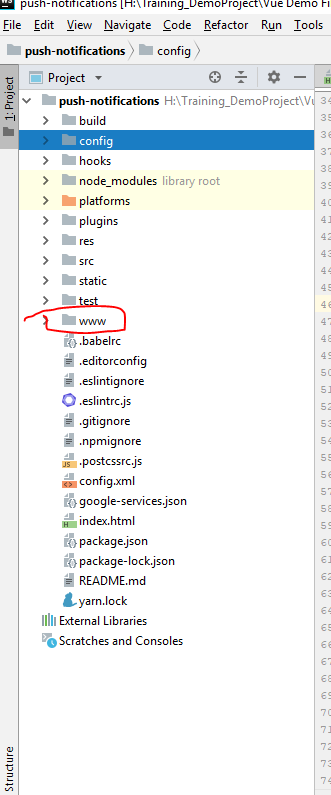
Create Cordova and Vue Project in the Same Folder with Same Name

* **cordova create push-notifications**
* **vue init webpack push-notifications**

You will be asked: ? Target directory exists. Continue? you should answer “Yes” to this as the directory was created in the previous Cordova step.

**Step5: Open Project and remove all files and folder from the www folder using following command:**

* **rm -r www/\***



**Step6: Change Your Config 🡪 index.js file settings inside build object as below:**

**// Change index path as below,add www in front of path name**

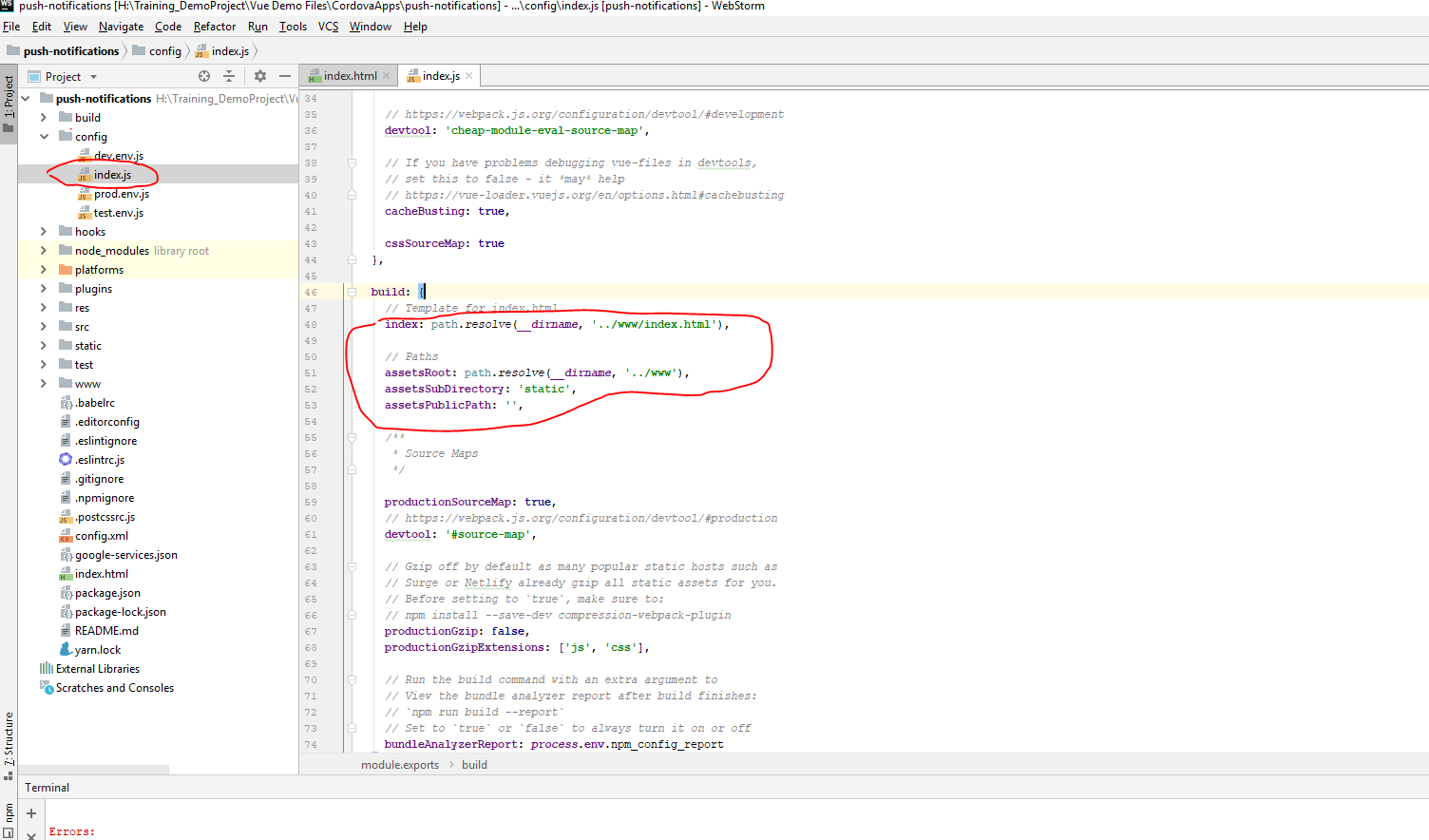
index: path.resolve(\_\_dirname, '**../www/index.html'**),

assetsRoot: path.resolve(\_\_dirname, **'../www/'**),

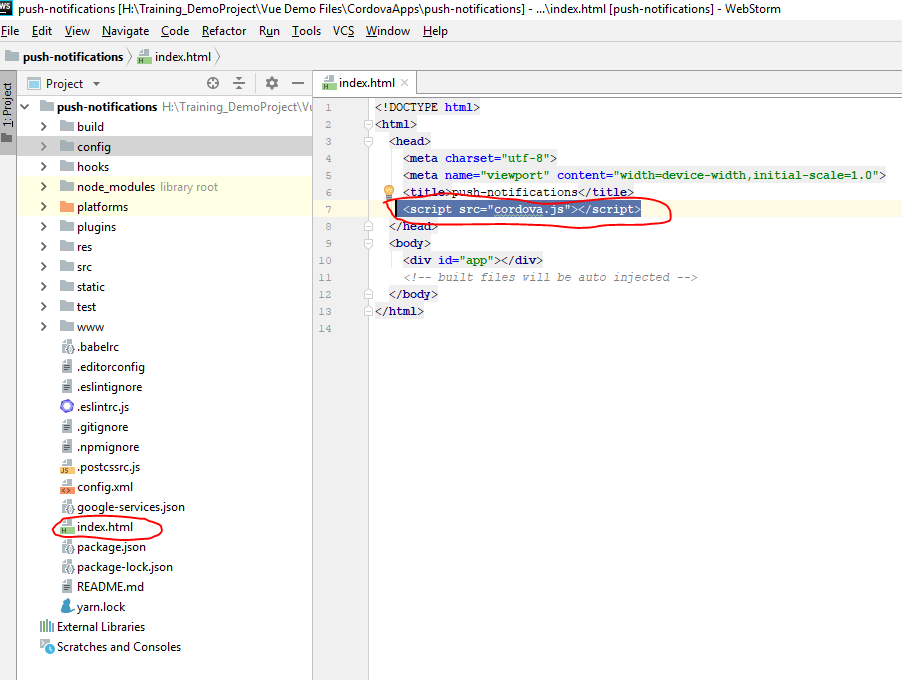
assetsSubDirectory: 'static',

**//remove back slash from assetsPublicPath**

assetsPublicPath: '',



**Step7: Add Cordova.js script in the header section of the index.html**



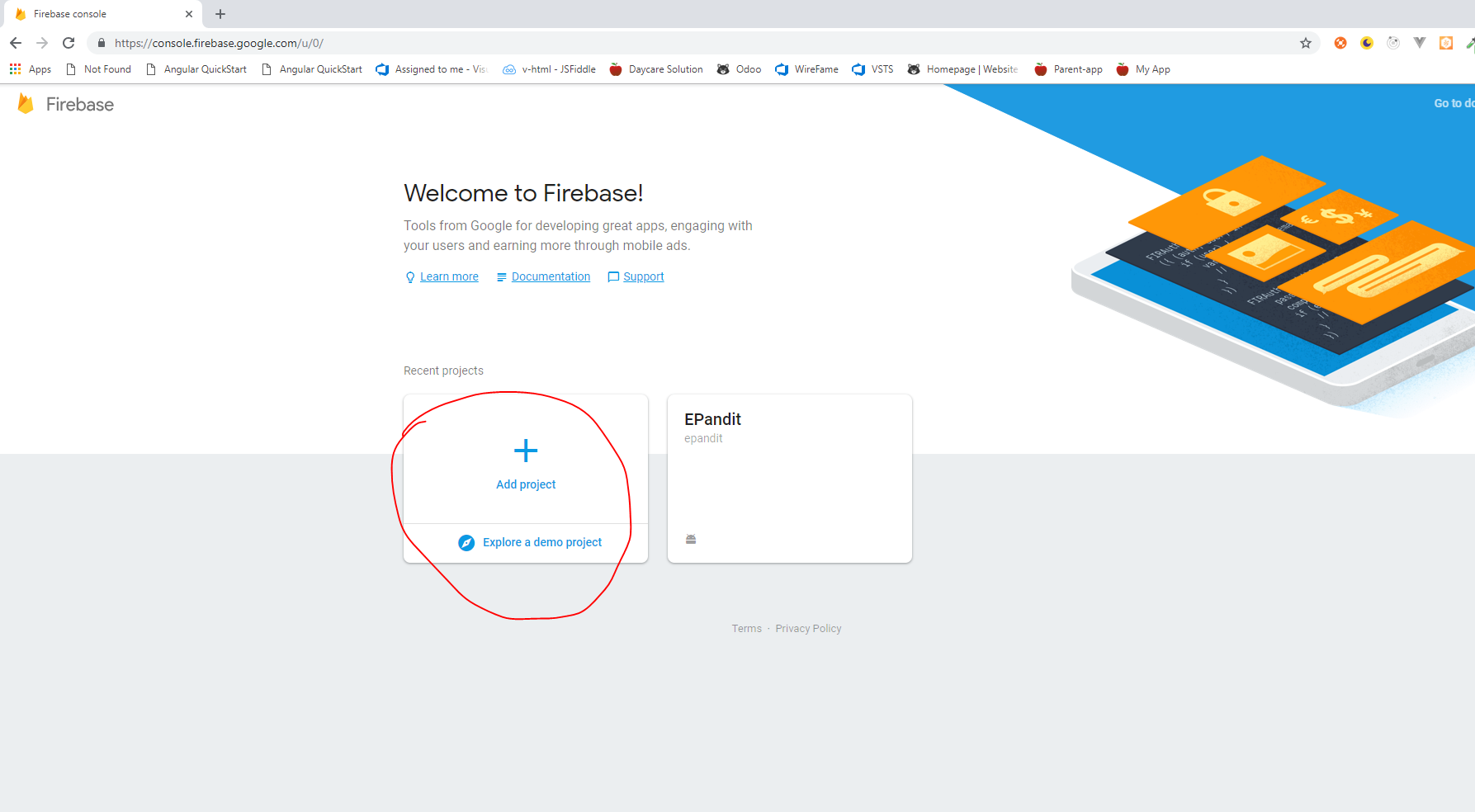
**Step8: run following commands**

* **npm install**
* **npm run build**
* **cordova add platform android**
* **cordova build android**
* **cordova run android**

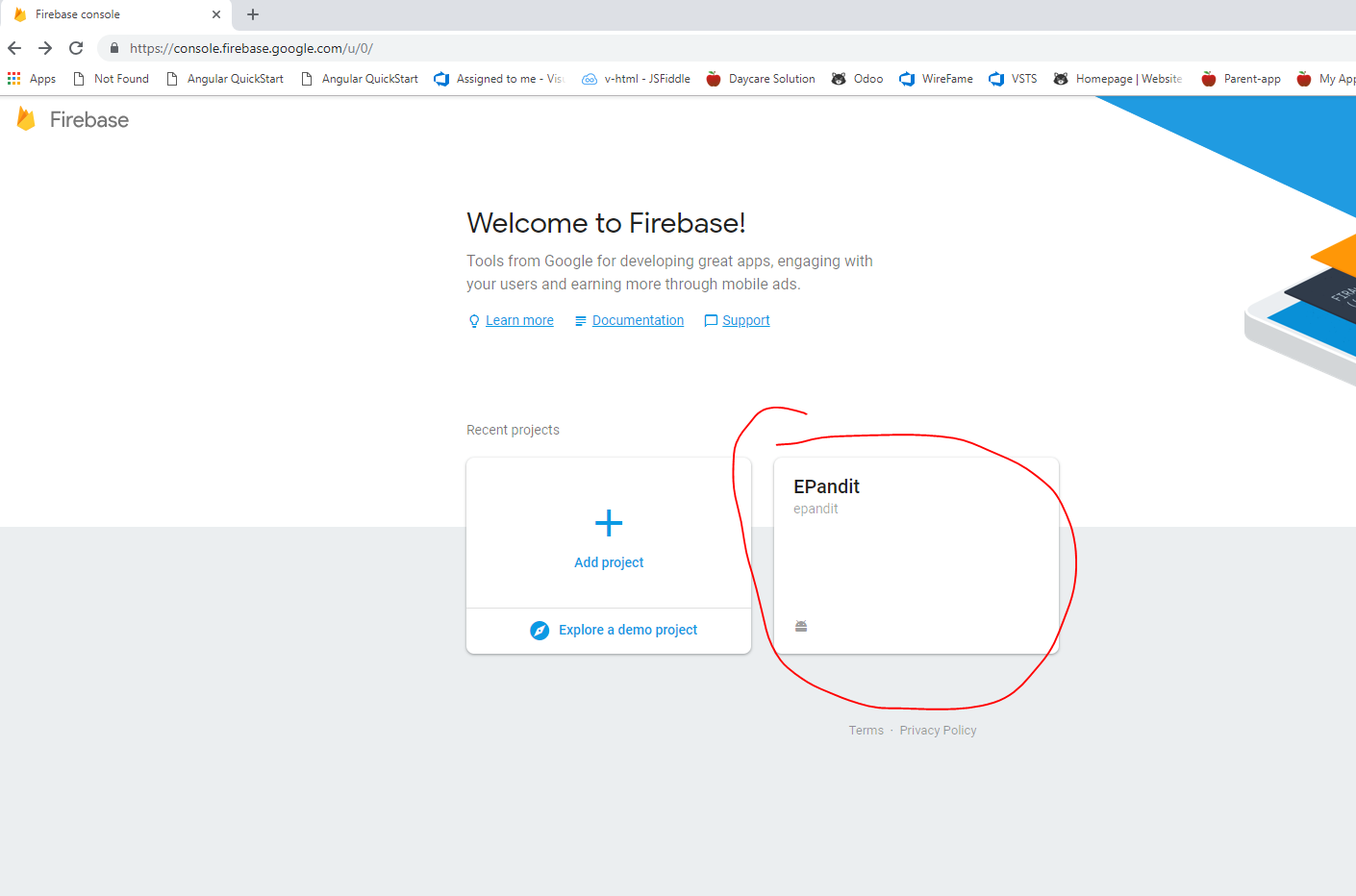
**Integration of Firebase Push Notification in the project:**

**Step1: Login in to Firebase**

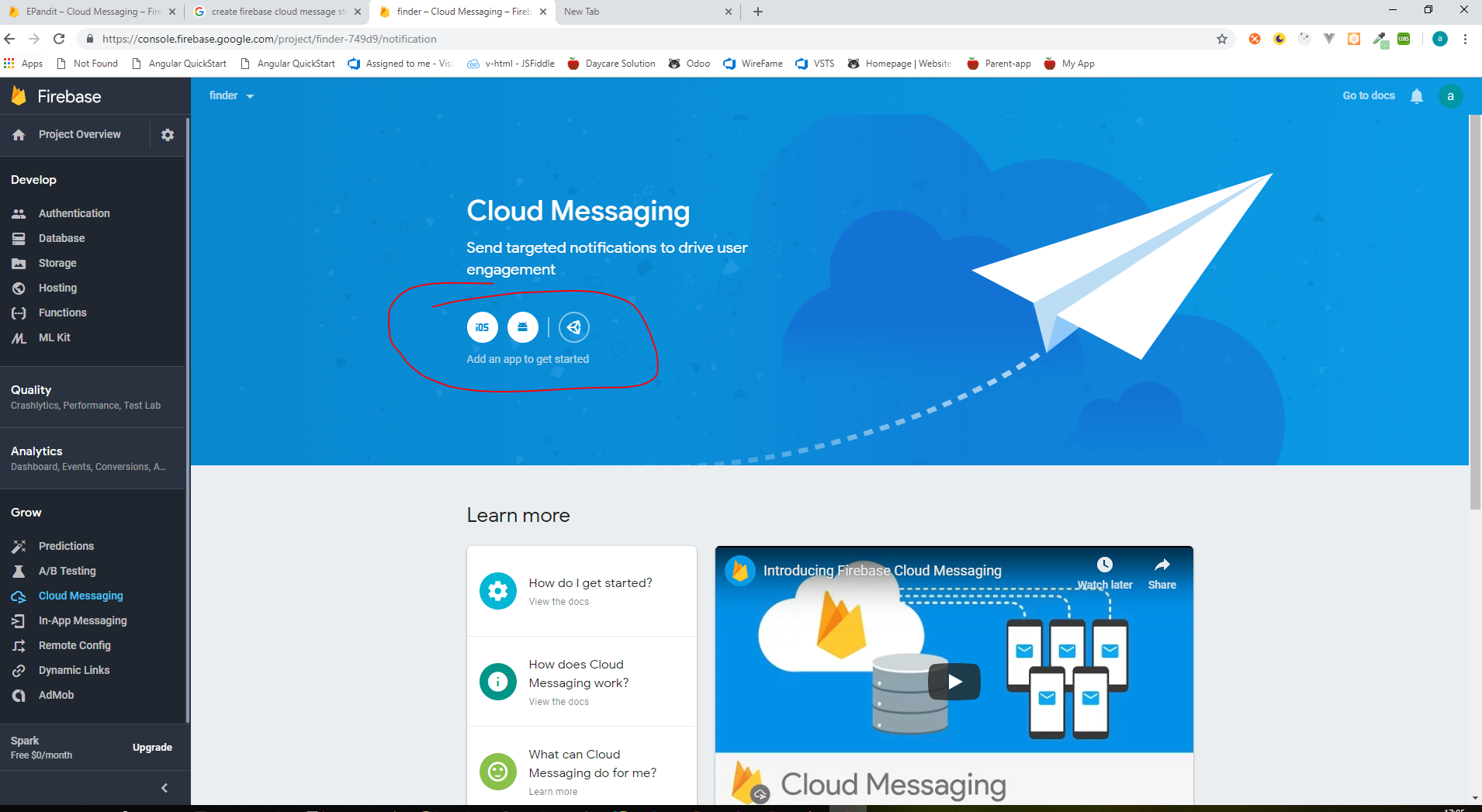
**Step2: Add New Project**



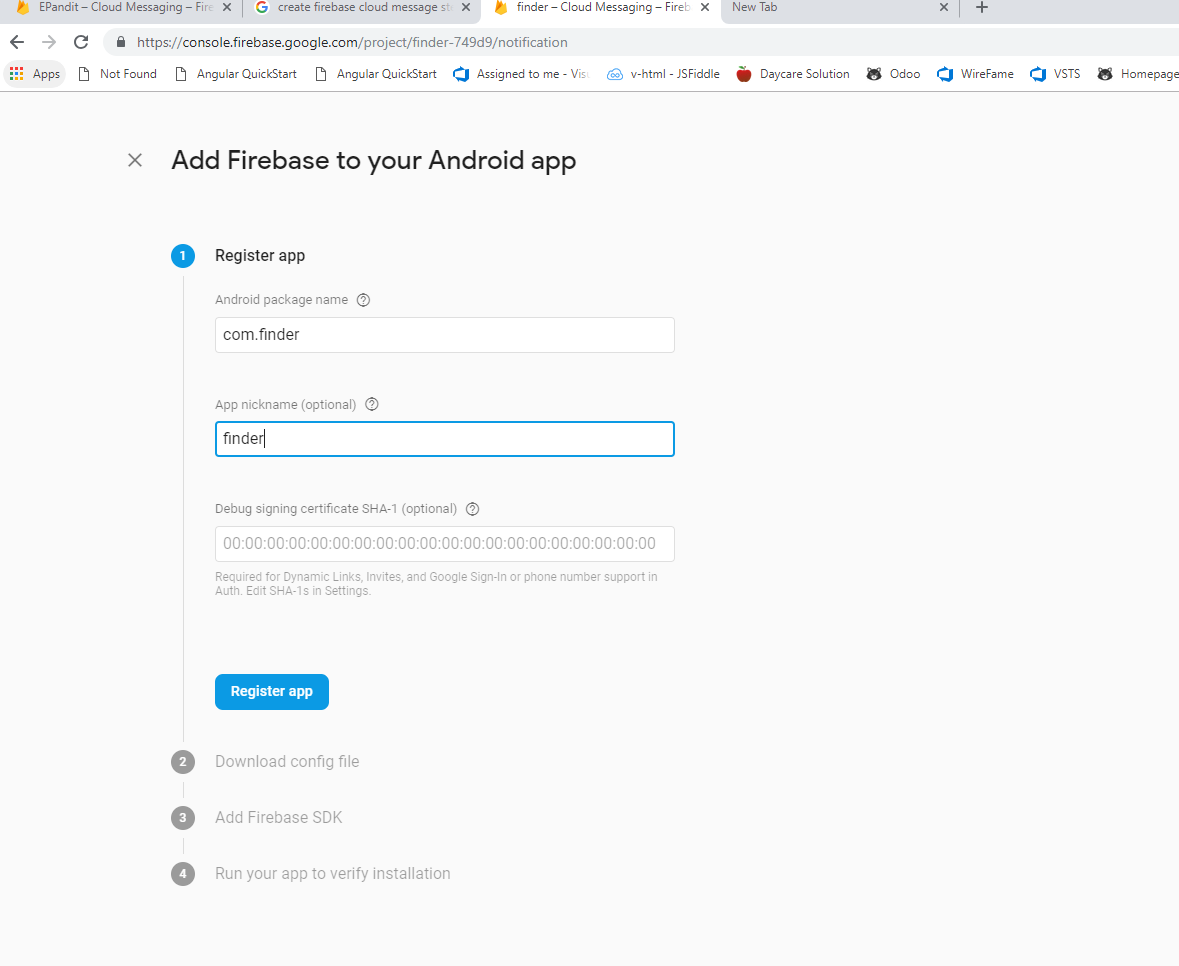
**Step3: open Project**



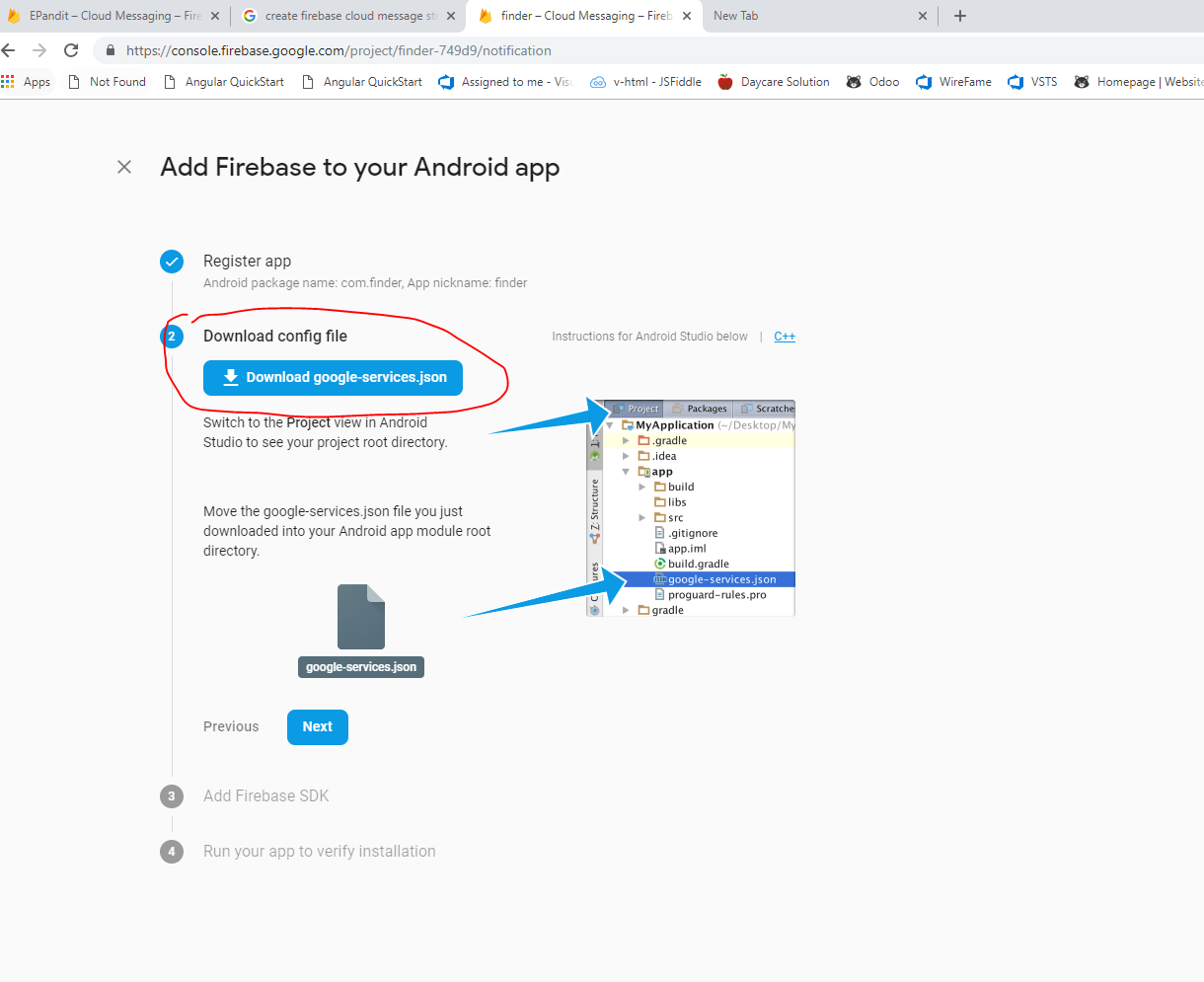
**Step4: Go to Grow 🡪 Cloud Messaging 🡪Create Cloud Message for respective platform**



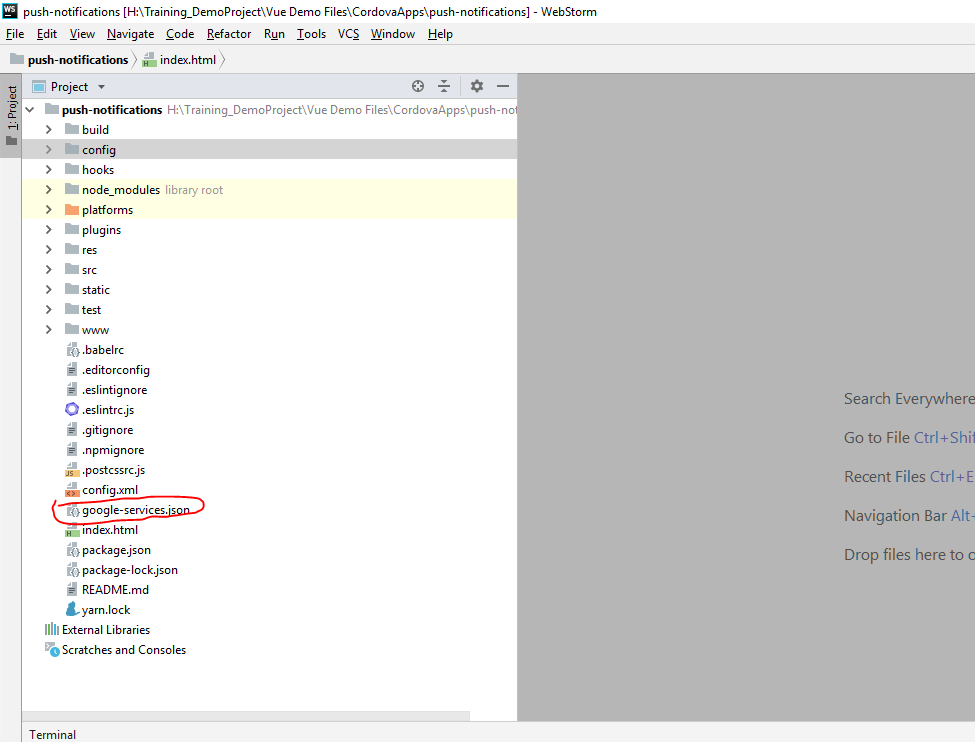
**Step5: Enter all Required Field**



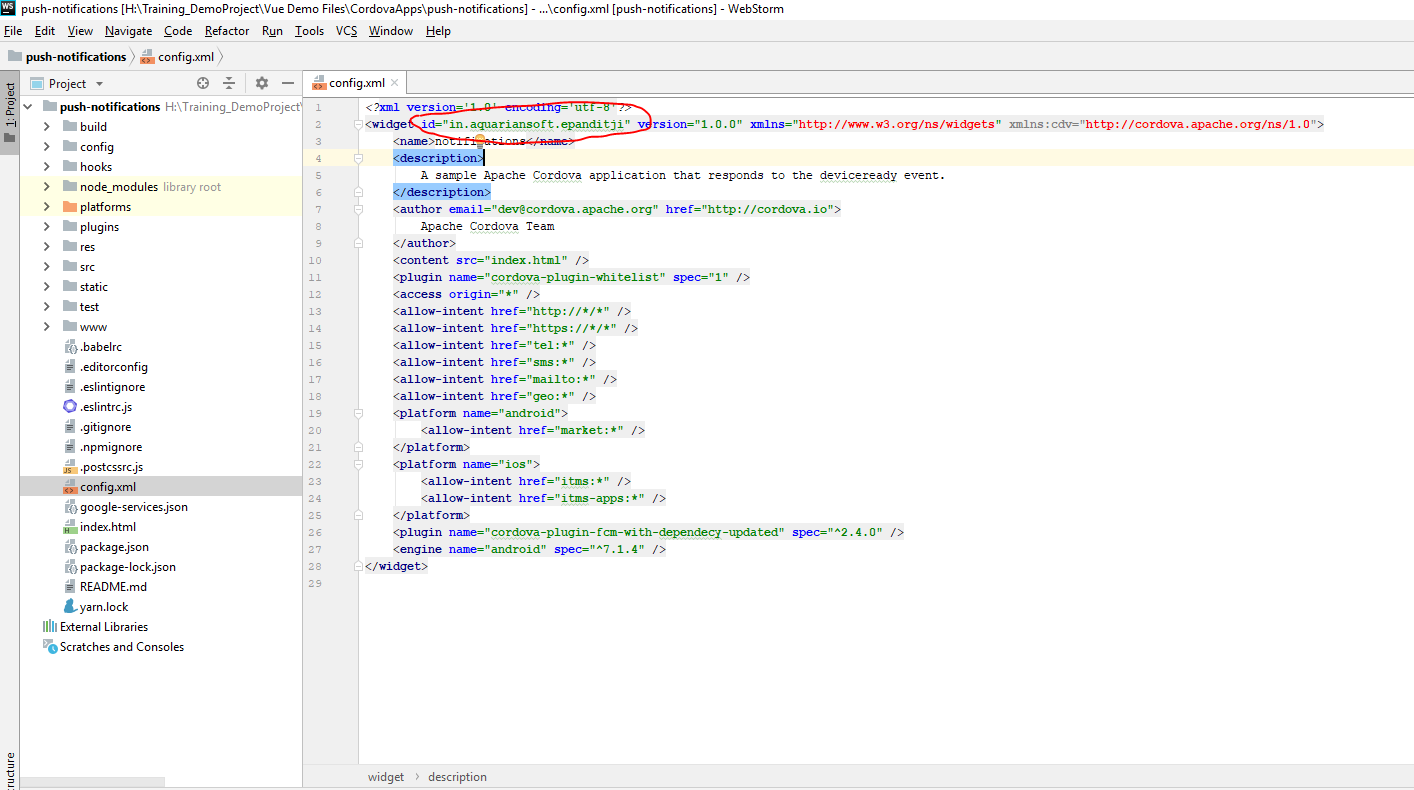
**Step6: Download “google-services.json “ File**



**Step7: Add “google-services.json” File in the project root folder**



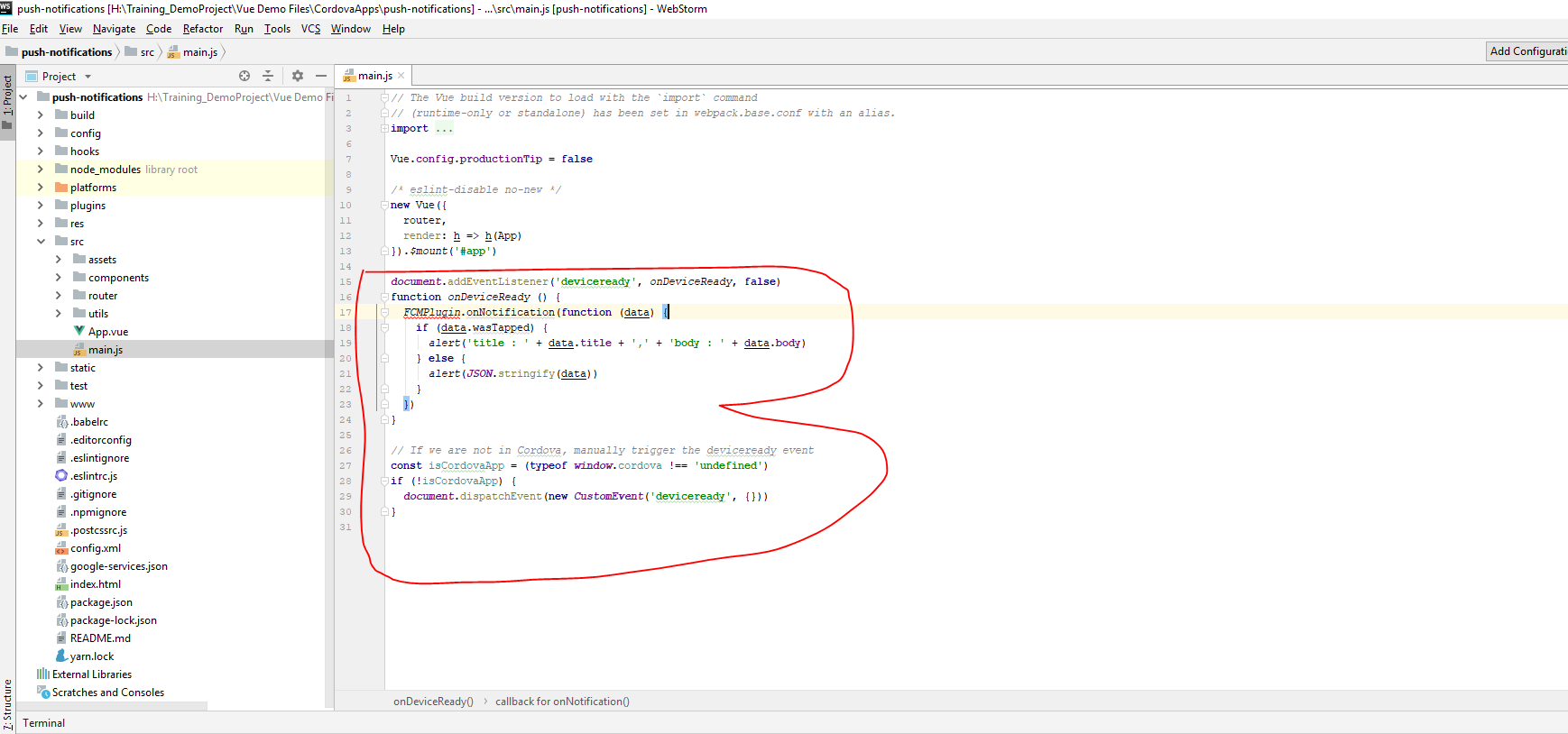
**Step8: Change id of “config.xml” file as your package name**



**Step9: Add “Google Firebase Cloud Messaging Cordova Push Plugin” plugin in the project as:**

**cordova-plugin-fcm-with-dependecy-updated**

**Step10: For Receive Message Add the Following Code in the main.js file as:**

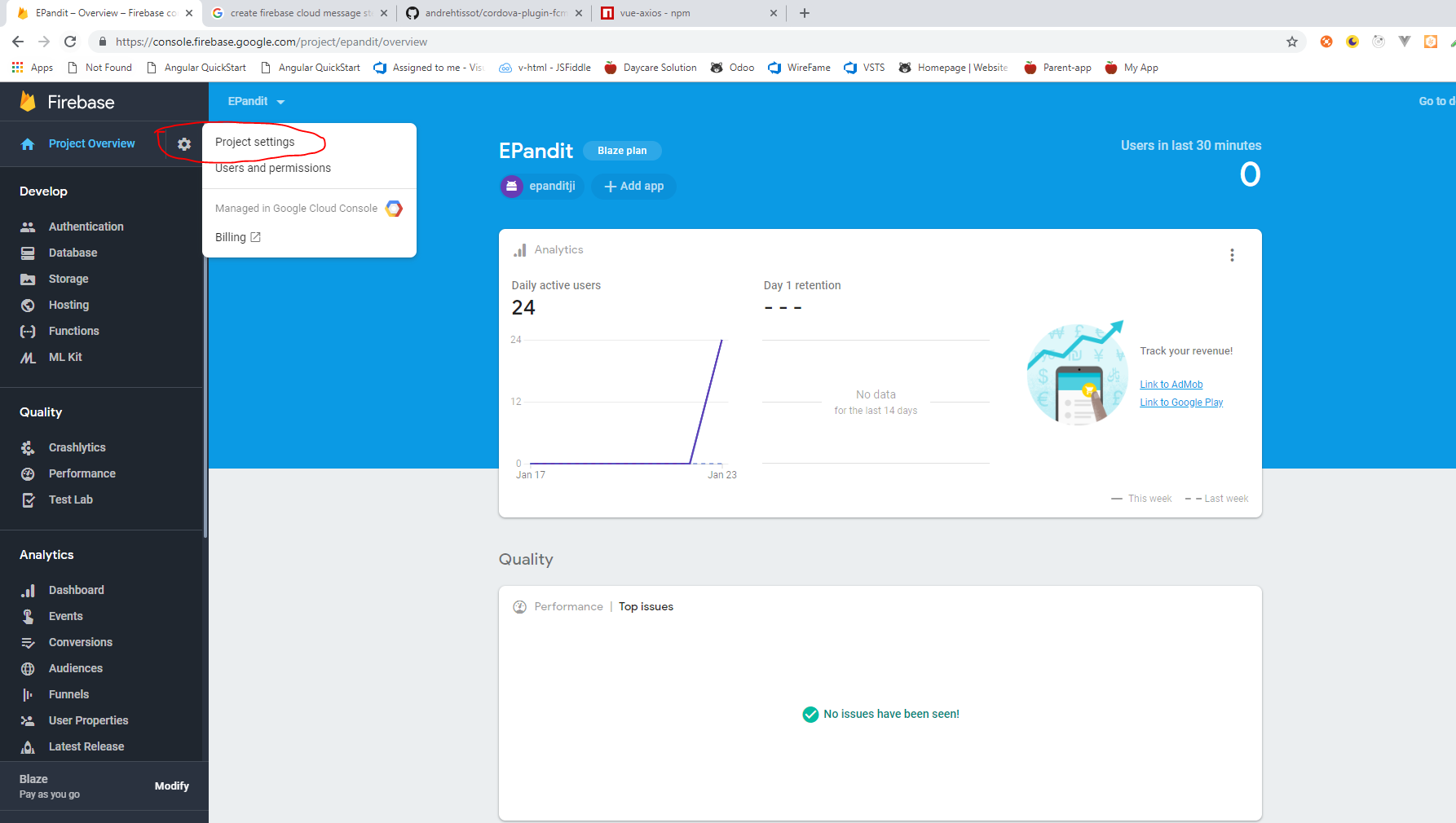


**Step:11 For Send Message we need to Post messages in following api**

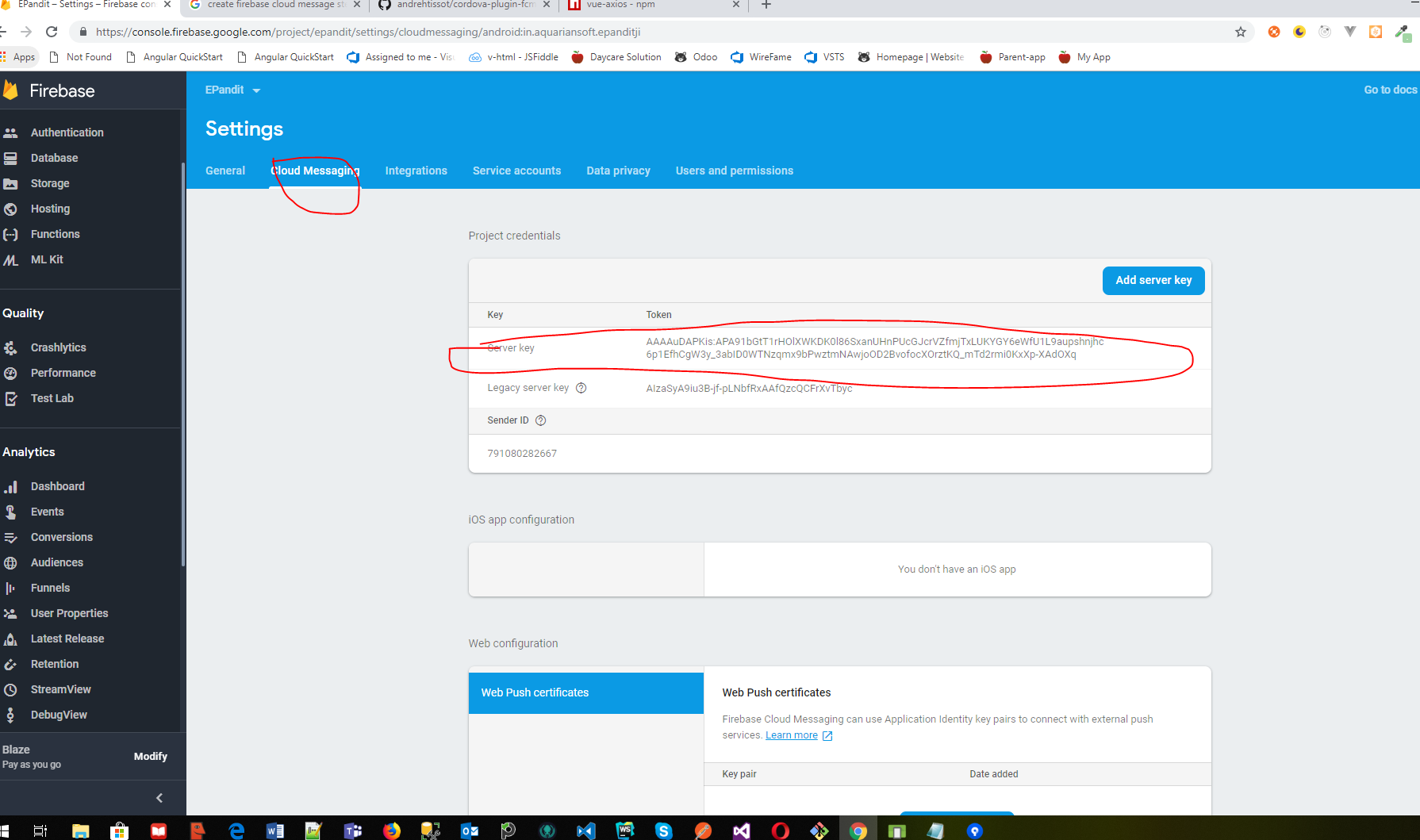
**https://fcm.googleapis.com/fcm/send** ---post call

To Get Authorization key value to pass in header section of api is

Step1 **:** Go to Firebase Project setting



Step2: Get **Server Key** from inside setting 🡪Cloud Messaging and pass this **Server Key** to the api Header Authorization Section as given code in below



In api we need to send object Folowing Object:

{

"notification":{

"title":"Notification title",

"body":"Notification body",

"sound":"default",

"click\_action":"FCM\_PLUGIN\_ACTIVITY",

"icon":"fcm\_push\_icon"

},

"data":{

"param1":"value1",

"param2":"value2"

},

"to":"/topics/topicExample",

"priority":"high",

"restricted\_package\_name":""

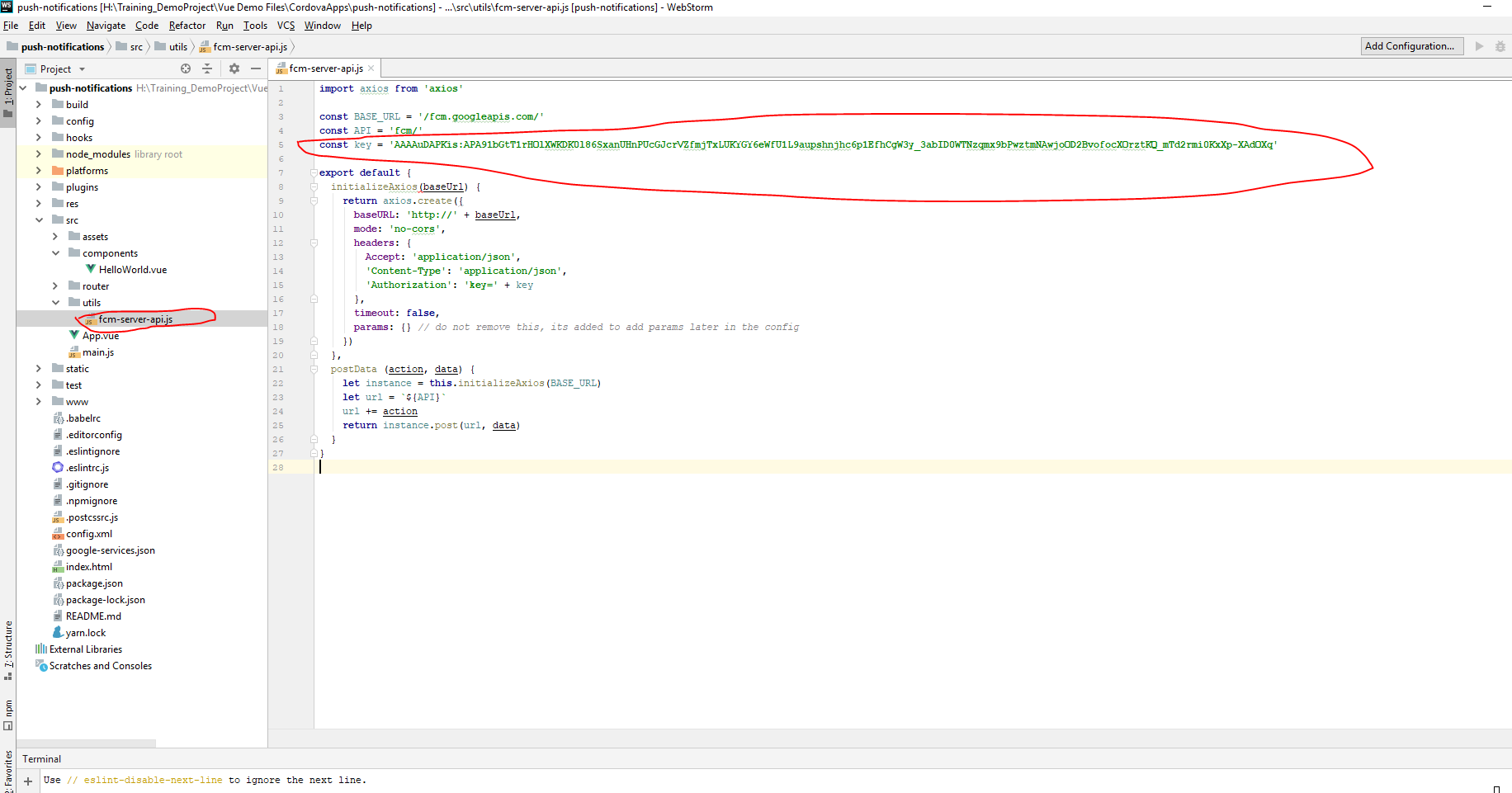
}

**Add the following code as below:**

1: Install axios for api call:

**npm install --save axios**

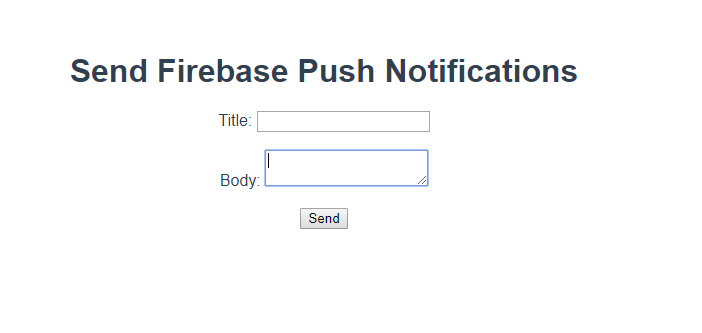
2: Create **fcm-server-api.js** file for api header setting as below



3: Design Page for Send Notification and Method for Send Code is as follow:

**import** axios **from 'axios'  
  
const** BASE\_URL = **'/fcm.googleapis.com/'  
const** API = **'fcm/'  
const** key = **'AAAAuDAPKis:APA91bGtT1rHOlXWKDK0l86SxanUHnPUcGJcrVZfmjTxLUKYGY6eWfU1L9aupshnjhc6p1EfhCgW3y\_3abID0WTNzqmx9bPwztmNAwjoOD2BvofocXOrztKQ\_mTd2rmi0KxXp-XAdOXq'  
  
export default** {  
 initializeAxios(baseUrl) {  
 **return** axios.create({  
 **baseURL**: **'http://'** + baseUrl,  
 **mode**: **'no-cors'**,  
 **headers**: {  
 **Accept**: **'application/json'**,  
 **'Content-Type'**: **'application/json'**,  
 **'Authorization'**: **'key='** + key  
 },  
 **timeout**: **false**,  
 **params**: {} *// do not remove this, its added to add params later in the config* })  
 },  
 postData (action, data) {  
 **let** instance = **this**.initializeAxios(BASE\_URL)  
 **let** url = **`**${API}**`** url += action  
 **return** instance.post(url, data)  
 }  
}

<**template**>  
 <**div class="hello"**>  
 <**h1**>{{ **msg** }}</**h1**>  
  
 <**div**>  
 <**span**>Title:</**span**>  
 <**input v-model="title"**>  
 </**div**>  
  
 <**br**/>  
  
 <**div**>  
 <**span**>Body:</**span**>  
 <**textarea v-model="body"**></**textarea**>  
 </**div**>  
  
 <**br**/>  
  
 <**div**>  
 <**button @click="**sendNotification()**"**>Send</**button**>  
 </**div**>  
  
 </**div**>  
</**template**>  
  
<**script**>  
**import** fcmApi **from '@/utils/fcm-server-api'  
export default** {  
 **name**: **'HelloWorld'**,  
 data () {  
 **return** {  
 **msg**: **'Send Firebase Push Notifications'**,  
 **title**: **''**,  
 **body**: **''** }  
 },  
 **methods**: {  
 sendNotification () {  
 **let** notificationObj = {  
 **'notification'**: {  
 **'title'**: **this**.**title**,  
 **'body'**: **this**.**body**,  
 **'sound'**: **'default'**,  
 **'click\_action'**: **'FCM\_PLUGIN\_ACTIVITY'**,  
 **'icon'**: **'fcm\_push\_icon'** },  
 **'data'**: {  
 **'title'**: **this**.**title**,  
 **'body'**: **this**.**body** },  
 **'to'**: **'/topics/android'**,  
 **'priority'**: **'high'**,  
 **'restricted\_package\_name'**: **'in.aquariansoft.epanditji'** }  
 **return new *Promise***((resolve, reject) => {  
 fcmApi.postData(**'send'**, notificationObj).then(res => {  
 resolve(res)  
 **this**.**title** = **''  
 this**.**body** = **''** *alert*(**'msg successfully sent'**)  
 })  
 .catch(err => {  
 **if** (!err.**response**) {  
 *alert*(**'error'**)  
 } **else** {  
 *alert*(**'error'** + err)  
 }  
 reject(err)  
 })  
 })  
 }  
 }  
}  
</**script**>  
  
*<!-- Add "scoped" attribute to limit CSS to this component only -->*<**style scoped**>  
</**style**>



**Last Step: run following commands**

* **npm run build**
* **cordova build android**
* **cordova run android**

**For Help:**

**For Reference Use the below link:**

[**https://github.com/andrehtissot/cordova-plugin-fcm-with-dependecy-updated**](https://github.com/andrehtissot/cordova-plugin-fcm-with-dependecy-updated)

**We can also add Some other functionality by using below code:**

#### Receiving Token Refresh

//FCMPlugin.onTokenRefresh( onTokenRefreshCallback(token) );

//Note that this callback will be fired everytime a new token is generated, including the first time.

FCMPlugin.onTokenRefresh(function(token){

alert( token );

});

#### Get token

//FCMPlugin.getToken( successCallback(token), errorCallback(err) );

//Keep in mind the function will return null if the token has not been established yet.

FCMPlugin.getToken(function(token){

alert(token);

});

#### Subscribe to topic

//FCMPlugin.subscribeToTopic( topic, successCallback(msg), errorCallback(err) );

//All devices are subscribed automatically to 'all' and 'ios' or 'android' topic respectively.

//Must match the following regular expression: "[a-zA-Z0-9-\_.~%]{1,900}".

FCMPlugin.subscribeToTopic('topicExample');

#### Unsubscribe from topic

//FCMPlugin.unsubscribeFromTopic( topic, successCallback(msg), errorCallback(err) );

FCMPlugin.unsubscribeFromTopic('topicExample');

#### Receiving push notification data

//FCMPlugin.onNotification( onNotificationCallback(data), successCallback(msg), errorCallback(err) )

//Here you define your application behaviour based on the notification data.

FCMPlugin.onNotification(function(data){

if(data.wasTapped){

//Notification was received on device tray and tapped by the user.

alert( JSON.stringify(data) );

}else{

//Notification was received in foreground. Maybe the user needs to be notified.

alert( JSON.stringify(data) );

}

});

#### Send notification. Payload example (REST API)

Full documentation: <https://firebase.google.com/docs/cloud-messaging/http-server-ref>  
Free testing server: [https://cordova-plugin-fcm.appspot.com](https://cordova-plugin-fcm.appspot.com/)

//POST: https://fcm.googleapis.com/fcm/send

//HEADER: Content-Type: application/json

//HEADER: Authorization: key=AIzaSy\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

{

"notification":{

"title":"Notification title",

"body":"Notification body",

"sound":"default",

"click\_action":"FCM\_PLUGIN\_ACTIVITY",

"icon":"fcm\_push\_icon"

},

"data":{

"param1":"value1",

"param2":"value2"

},

"to":"/topics/topicExample",

"priority":"high",

"restricted\_package\_name":""

}

//sound: optional field if you want sound with the notification

//click\_action: must be present with the specified value for Android

//icon: white icon resource name for Android >5.0

//data: put any "param":"value" and retreive them in the JavaScript notification callback

//to: device token or /topic/topicExample

//priority: must be set to "high" for delivering notifications on closed iOS apps

//restricted\_package\_name: optional field if you want to send only to a restricted app package (i.e: com.myapp.test)