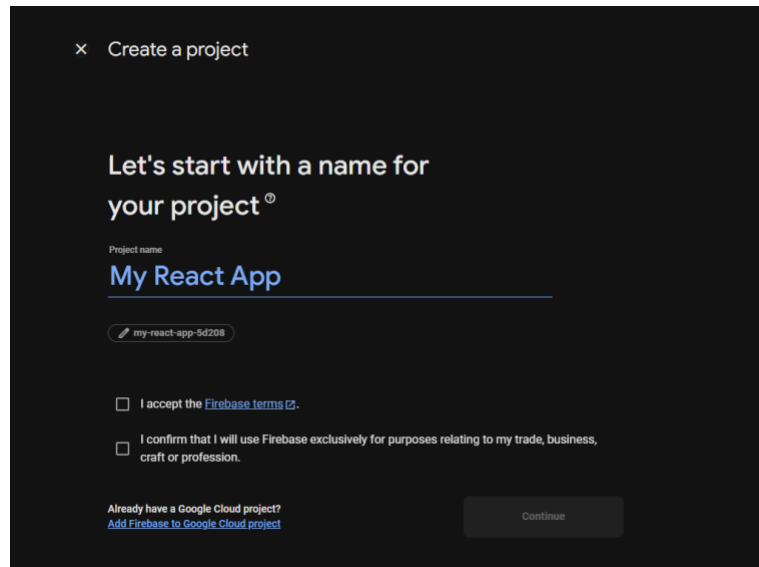


CLOUD COMPUTING

Practical :How to host and deploy a react app with firebase

Step 1: Firebase Console Dashboard.

- Go to your Firebase console dashboard and “create a project”.



× Create a project

Let's start with a name for your project[®]

Project name

My React App

my-react-app-5d208

☐ I accept the [Firebase terms](#).

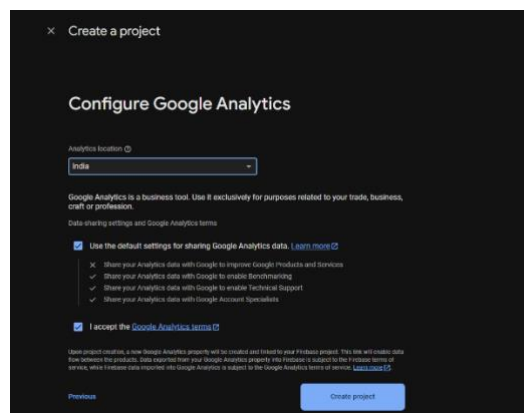
☐ I confirm that I will use Firebase exclusively for purposes relating to my trade, business, craft or profession.

Already have a Google Cloud project?
[Add Firebase to Google Cloud project](#)

Continue

Step 2: Fill in Project Details

- In this example, I named the project "My React APP".
- Click the "Create Project" button and it should start creating your Firebase project.



× Create a project

Configure Google Analytics

Analytics location

India

Google Analytics is a business tool. Use it exclusively for purposes related to your trade, business, craft or profession.

Data sharing settings and Google Analytics terms

☒ Use the default settings for sharing Google Analytics data. [Learn more](#)

- ☒ Share your Analytics data with Google to improve Google Products and Services
- ☒ Share your Analytics data with Google to enable Benchmarking
- ☒ Share your Analytics data with Google to enable Technical Support
- ☒ Share your Analytics data with Google Account Specialists

☒ I accept the [Google Analytics terms](#)

Start project creation. A new Google Analytics property will be created and linked to your Firebase project. This site will enable data flow between the products. Data reported from your Google Analytics property into Firebase is subject to the Firebase terms of service, which includes data processing. This Google Analytics is subject to the Google Analytics terms of service. [Learn more](#)

Previous Create project

Step 3: Install Firebase and Firebase Tools

- The next step is to go to your project terminal on VS code, your Command Line Interface, or any code editor you're using. Ensure you're in the main folder of the project you want to deploy and then install Firebase into the project using this command: `npm install firebase`.
- Next, install the Firebase tools we'll be using for hosting and deployment using this command: `npm install -g firebase-tools`.

```
PS C:\Users\Admin\Desktop\project1> npm install firebase
[#####.....] / idealTree:Admin: timing idealTree:root Completed in 3608ms

PS C:\Users\Admin\Desktop\project1> npm install -g firebase-tools
>>
added 604 packages, and changed 2 packages in 36s

68 packages are looking for funding
  run `npm fund` for details
PS C:\Users\Admin\Desktop\project1> firebase --version
13.22.1
PS C:\Users\Admin\Desktop\project1> 
```

Step 4: Log in to Firebase Using the Terminal

- After configuring the Firebase project and installing the necessary dependencies, you'll have to log in to Firebase on the terminal using this command: `firebase login`.
- A prompt requiring you to select Yes or No to a question on whether you should "Allow Firebase to collect CLI and Emulator Suite usage and error reporting information" will appear. Select the "Yes" option.

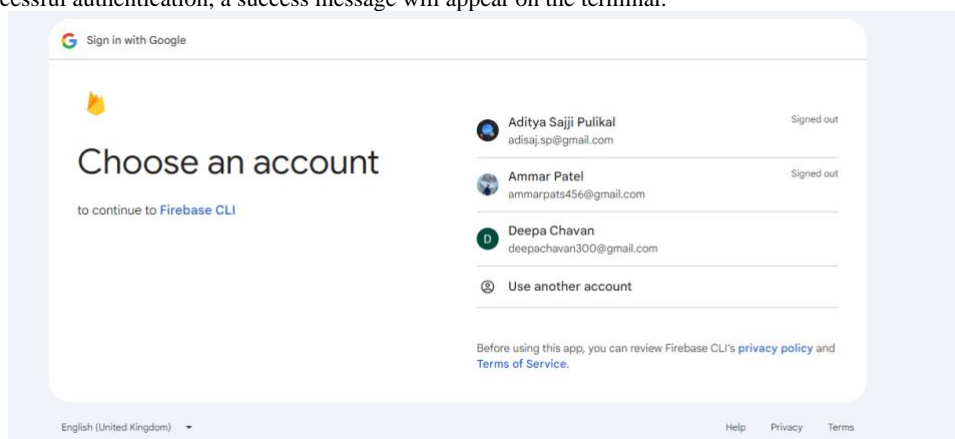
```
13.22.1
PS C:\Users\Admin\Desktop\project1> firebase login
i Firebase optionally collects CLI and Emulator Suite usage and error reporting information to help improve our products. Data is collected in accordance with our privacy policy (https://firebase.google.com/privacy) and is not used to identify you.

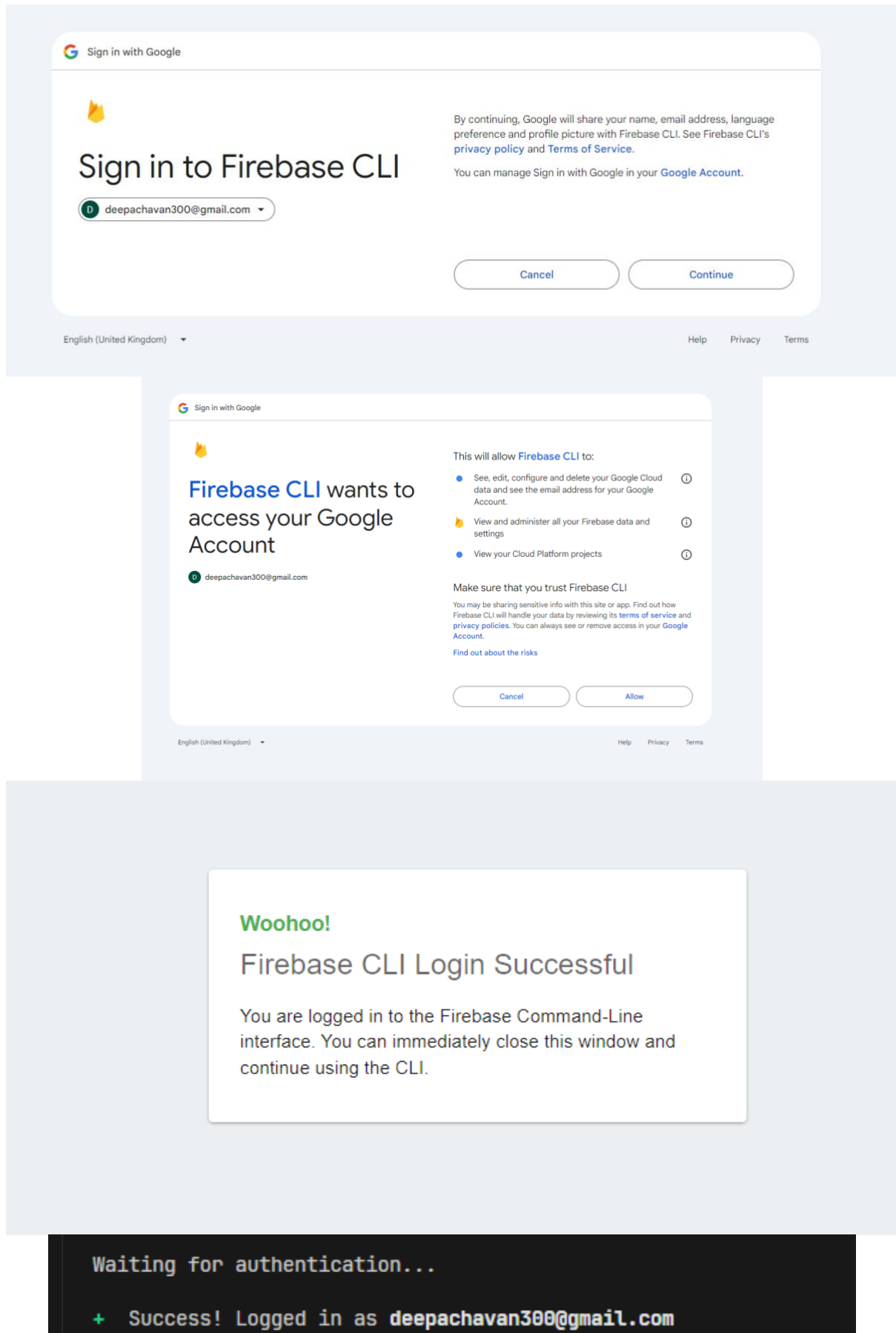
? Allow Firebase to collect CLI and Emulator Suite usage and error reporting information? Yes
i To change your data collection preference at any time, run 'firebase logout' and log in again.

Visit this URL on this device to log in:
https://accounts.google.com/o/oauth2/auth?client_id=563584335869-fgrhqm47bqnekij5i8b5pr03ho849e6.apps.googleusercontent.com&scope=email%20openid%20https://www.googleapis.com/auth/cloud-platform&response_type=code&state=cts.readonly%20https%3A%2F%2Fwww.googleapis.com%2Fauth%2Ffirebase%20https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fcloud-platform
Waiting for authentication...
```

Step 5: Select Account

- A window will open up on the default browser that will require you to select your Firebase account for login. After successful authentication, a success message will appear on the terminal.





Step 7: Run Project Build

- Use the **npm run build** command to build the project scripts. (Create a build command with express JS)

```
PS C:\Users\Admin\Desktop\project1> npm install express
>>

50 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities

package.json > ...
1  {
2    "name": "project1",
3    "version": "1.0.0",
4    "description": "A simple project",
5    "main": "server.js",
6    "scripts": {
7      "start": "node server.js",
8      "build": "echo \"No build process is required for this project\""
9    },
10   "author": "",
11   "license": "ISC",
12   "dependencies": {
13     "express": "^4.18.2"
14   }
15 }
16

server.js > ...
1  const express = require("express");
2  const app = express();
3  const port = 1000;
4
5  // Serve static files from the "src" directory
6  app.use(express.static("src"));
7
8  app.get("/", (req, res) => {
9    res.sendFile(__dirname + "/src/index.html");
10  });
11
12  app.listen(port, () => {
13    console.log(`Server running at http://localhost:${port}`);
14  });
15

PS C:\Users\Admin\Desktop\project1> npm run build

> project1@1.0.0 build
> echo "No build process is required for this project"

"No build process is required for this project"
```

How to Initialize Firebase

Step 1: Initialize Firebase

- Initialize Firebase for this project by using this command on the terminal: **firebase init**.
- Some prompts that will come up after this command are: "Are you ready to proceed?", to which you will type "Y" for "Yes".
- The next prompt is: "Which Firebase features do you want to set up for this directory?". Use the arrow down key on

your keyboard to point to the "Hosting:Configure files for Firebase Hosting and (optionally) set up GitHub Action deploys" option. Press the space bar and then hit enter.

```
PS C:\Users\Admin\Desktop\project1> firebase init

##### 
##      ## ##      ## ##      ##      ## ##      ##      ##
##### 
##      ## ##      ## ##      ##      ## ##      ##      ##
##      ##### 

You're about to initialize a Firebase project in this directory:

C:\Users\Admin\Desktop\project1

? Are you ready to proceed? (Y/n) ☐
```

```
C:\Users\Admin\Desktop\project1
> Are you ready to proceed? Yes
> Which Firebase features do you want to set up for this directory? Press Space to select features, then Enter to confirm your choices. (Press <space> to select, <a> to toggle all, <i> to invert selection, and <enter> to proceed)
(*) Data Connect: Set up a Firebase Data Connect service
() Firestore: Configure security rules and indexes files for Firestore
() Genkit: Set up a new Genkit project with Firebase
() Functions: Configure a Cloud Functions directory and its files
() Hosting: Configure files for Firebase Hosting and (optionally) set up GitHub Action deploys
() Storage: Configure a security rules file for Cloud Storage
() Emulators: Set up local emulators for Firebase products
(Move up and down to reveal more choices)

C:\Users\Admin\Desktop\project1

? Are you ready to proceed? Yes
? Which Firebase features do you want to set up for this directory? Press Space to select features, then Enter to confirm your choices and <enter> to proceed)
() Firestore: Configure security rules and indexes files for Firestore
() Genkit: Set up a new Genkit project with Firebase
() Functions: Configure a Cloud Functions directory and its files
>(*) Hosting: Configure files for Firebase Hosting and (optionally) set up GitHub Action deploys
() Storage: Configure a security rules file for Cloud Storage
() Emulators: Set up local emulators for Firebase products
() Remote Config: Configure a template file for Remote Config
(Move up and down to reveal more choices)
```

Step 2: Project Setup

- This step connects your project directory with the Firebase project. When prompted to select a project, choose the "Use an existing project" option.
- Then when prompted to "select a default Firebase project for the directory", select the particular Firebase project you created in part 1 of this process.
- You might see some other project options if you have more than one Firebase project on your Firebase console.

```
=== Project Setup

First, let's associate this project directory with a Firebase project.
You can create multiple project aliases by running firebase use --add,
but for now we'll just set up a default project.

? Please select an option: (Use arrow keys)
> Use an existing project
  Create a new project
  Add Firebase to an existing Google Cloud Platform project
  Don't set up a default project
```

=== Project Setup

First, let's associate this project directory with a Firebase project. You can create multiple project aliases by running `firebase use --add`, but for now we'll just set up a default project.

```
? Please select an option: Use an existing project
? Select a default Firebase project for this directory: (Use arrow keys)
> my-react-app-5d208 (My React App)
```

Step 3: Setup Hosting

- This process will bring up some prompts you'll have to answer.
- The first one is: "What do you want to use as your public directory?", to which you'll choose or type in "build".
- Next, when asked if you want to "Configure as a single-page app (rewrite all urls to /index.html)", select the "Y" or "Yes" option.
- When asked to "Set up automatic builds and deploys with GitHub?", choose the "Yes" option. Also, when prompted with the "File build/index.html already exists. Overwrite?" question, choose "No".

=== Hosting Setup

Your **public** directory is the folder (relative to your project directory) that will contain Hosting assets to be uploaded with `firebase deploy`. If you have a build process for your assets, use your build's output directory.

```
? What do you want to use as your public directory? build
? Configure as a single-page app (rewrite all urls to /index.html)? (y/N) █
```

Step 4: Authorize Firebase with GitHub

- You will have to authorize Firebase with your GitHub account. A window will open up on your browser that will require you to authorize Firebase into your GitHub, and input your GitHub password. After a successful authentication, you'll get a success message on your terminal with your GitHub username.

Set Up GitHub Workflow

Step 1: Select GitHub Repo

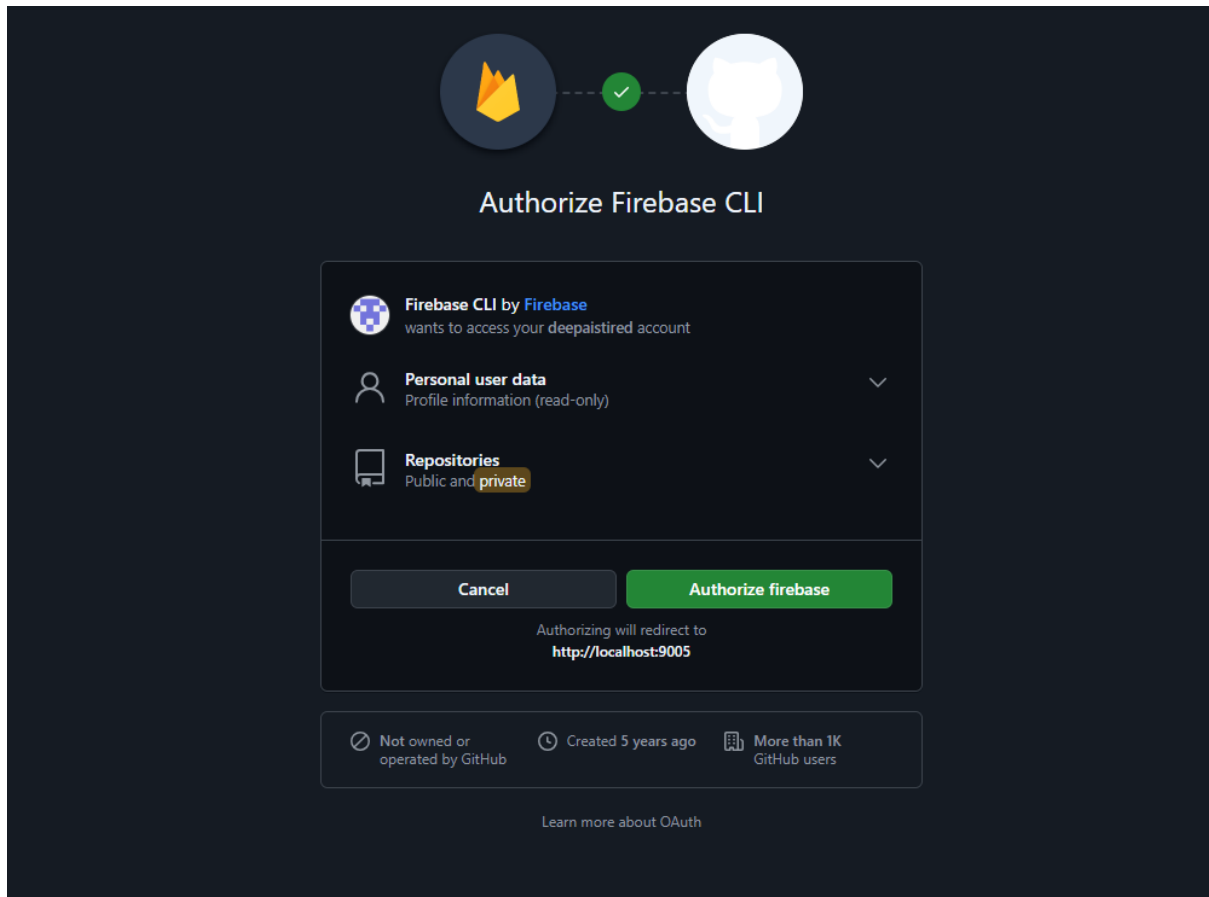
- First, you'll need to type in the GitHub repository you'd like to use to set up a GitHub workflow for Firebase deployments.
- The format to do this is "username/repository". Remember in part 1 of this process, you built a project and pushed it to GitHub. That GitHub repo is what you'll use.

```
? What do you want to use as your public directory? build
? Configure as a single-page app (rewrite all urls to /index.html)? Yes
? Set up automatic builds and deploys with GitHub? Yes
+ Wrote build/index.html

i Didn't detect a .git folder. Assuming C:\Users\Admin\Desktop\project1 is the project root.
i Authorizing with GitHub to upload your service account to a GitHub repository's secrets store.

Visit this URL on this device to log in:
https://github.com/login/oauth/authorize?client_id=89cf58f02ac6aaed3484&state=554625609&redirect_uri=http%3A%2F%2Flocalhost%3A9085&scope=read%3Auser%20repo%20public_repo

Waiting for authentication...
```



Woohoo!

Firebase CLI GitHub Login Successful

You are logged into GitHub via the Firebase CLI. You can immediately close this window and continue using the Firebase CLI.

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
? Set up automatic builds and deploys with GitHub? Yes
i build/index.html is unchanged
i Didn't detect a .git folder. Assuming C:\Users\Admin\Desktop\project1 is the project root.
i Authorizing with GitHub to upload your service account to a GitHub repository's secrets store.
Visit this URL on this device to log in:
https://github.com/login/oauth/authorize?client_id=89cf58f82ac6aaed3484&state=887427164&redirect_uri=http%3A%2F%2Flocalhost%3A9005&scope=read%3Auser%20repo%20public_repo
Waiting for authentication...
+ Success! Logged into GitHub as deepastired
? For which GitHub repository would you like to set up a GitHub workflow? (format: user/repository) |
```

Step 2: GitHub secret token

- After setting up a GitHub workflow, it will create a service account with Firebase Hosting admin permissions and will upload the service account JSON to GitHub as a secret token.
- You can also view this secret token on GitHub. To do this, go to the repository of the project and switch to the "Settings" tab.

On the left-hand panel of the settings page, click on the "Secrets and variables" dropdown and select the "Actions" option.

The screenshot shows the GitHub Actions secrets page. At the top, there are tabs for Security, Insights, and Settings. The main heading is 'Actions secrets / New secret'. Below this, there is a form to create a new secret. The 'Name' field is filled with 'FIREBASE_SERVICE_ACCOUNT'. The 'Secret' field is empty. Below the form, there is a section titled 'Repository secrets' with a 'New repository secret' button. A table lists the existing repository secrets, showing 'FIREBASE_SERVICE_ACCOUNT' as the only one, with a 'now' timestamp and edit/delete icons. At the bottom, a terminal window shows the command to run the workflow and the resulting JSON output for the service account.

```

> Users > Admin > Downloads > <C> my-react-app-5d208-firebase-adminsdk-ctsk-d8615ec0c4.json > ...
{
  "type": "service_account",
  "project_id": "my-react-app-5d208",
  "private_key_id": "d8615ec0c448f5c331dc0840ce9685ba984ebd67",
  "private_key": "FIREBASE_SERVICE_ACCOUNT",
  "client_email": "firebase-adminsdk-ctsk@my-react-app-5d208.iam.gserviceaccount.com",
  "client_id": "101803709115223633105",
  "auth_uri": "https://accounts.google.com/o/oauth2/auth",
  "token_uri": "https://oauth2.googleapis.com/token",
  "auth_provider_x509_cert_url": "https://www.googleapis.com/oauth2/v1/certs",
  "client_x509_cert_url": "https://www.googleapis.com/robot/v1/metadata/x509/firebase-adminsdk-ctsk%40my-react-app-5d208.iam.gserviceaccount.com",
  "universe_domain": "googleapis.com"
}

```

Step 3: Set up Workflow

- You'll be prompted with the question "Set up the workflow to run a build script before every deploy?". Choose "Yes" for this.
- You'd also be asked, "What script should be run before every deploy? (npm ci && npm run build) npm run build". Type this into the terminal: **npm ci && npm run build**. This will create a workflow file in the project directory.
- You'll now see the "firebase-hosting-pull-request.yml" file inside a ".github/workflows" folder in your project folder structure.

```

https://github.com/login/oauth/authorize?client_id=89cf58f02ac6aaed3484&state=615759858&redirect_uri=http%3A%2F%2Flocalhost%3A9095&scope=read%3Auser%20repo
Waiting for authentication...
+ Success! Logged into GitHub as deepaistired
? For which GitHub repository would you like to set up a GitHub workflow? (format: user/repository) deepaistired/firebase-development
+ Created service account github-action-877072371 with Firebase Hosting admin permissions.
+ Uploaded service account JSON to GitHub as secret FIREBASE_SERVICE_ACCOUNT_MY_REACT_APP_50208.
i You can manage your secrets at https://github.com/deepaistired/firebase-development/settings/secrets.
? Set up the workflow to run a build script before every deploy? Yes
? What script should be run before every deploy? (npm ci && npm run build) npm ci && npm run build

```



```
? For which GitHub repository would you like to set up a GitHub workflow? (format: user/repository) deepaistired/firebase-development
+ Created service account github-action-877072371 with Firebase Hosting admin permissions.
+ Uploaded service account JSON to GitHub as secret FIREBASE_SERVICE_ACCOUNT_MY_REACT_APP_50208.
i You can manage your secrets at https://github.com/deepaistired/firebase-development/settings/secrets.

? Set up the workflow to run a build script before every deploy? Yes
? What script should be run before every deploy? (npm ci && npm run build) npm run build
? What script should be run before every deploy? npm ci && npm run build

+ Created workflow file C:\Users\Admin\Desktop\project1\.github\workflows/firebase-hosting-pull-request.yml
? Set up automatic deployment to your site's live channel when a PR is merged? (Y/n) █
```

Step 4: Automatic deployment

- Next, you'll be asked if you want to "Set up automatic deployment to your site's live channel when a PR is merged". Select "Yes".
- When asked to enter the name of the GitHub branch associated with your site's live channel, type or select "main". This will create another workflow file "firebase-hosting-merge.yml" inside the ".github/workflows" folder.

```
+ Created workflow file C:\Users\Admin\Desktop\project1\.github\workflows/firebase-hosting-pull-request.yml
? Set up automatic deployment to your site's live channel when a PR is merged? Yes
? What is the name of the GitHub branch associated with your site's live channel? main

+ Created workflow file C:\Users\Admin\Desktop\project1\.github\workflows/firebase-hosting-merge.yml

i Action required: Visit this URL to revoke authorization for the Firebase CLI GitHub OAuth App:
https://github.com/settings/connections/applications/89cf50f02ac6aaed3484
i Action required: Push any new workflow file(s) to your repo

i Writing configuration info to firebase.json...
```

Step 5: Generate Folders

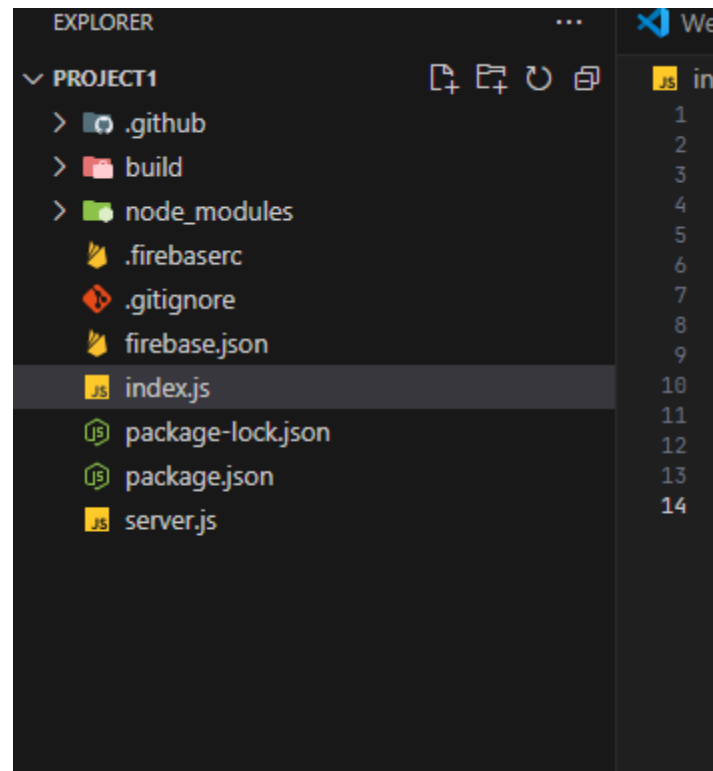
- The two operations performed above will generate two folders in your project directory. One named "firebase.json" is where the configuration information will be written in, and the other named ".firebaserc" is where the project information will be written in.

```
+ Created workflow file C:\Users\Admin\Desktop\project1\.github\workflows/firebase-hosting-merge.yml

i Action required: Visit this URL to revoke authorization for the Firebase CLI GitHub OAuth App:
https://github.com/settings/connections/applications/89cf50f02ac6aaed3484
i Action required: Push any new workflow file(s) to your repo

i Writing configuration info to firebase.json...
i Writing project information to .firebaserc...
i Writing gitignore file to .gitignore...

+ Firebase initialization complete!
PS C:\Users\Admin\Desktop\project1> █
```



Run the Deployment Command

Run the deployment command **firebase deploy**. Wait for it to deploy. After it is done, a success message will be displayed on your terminal with a hosting URL. That is the project's live link with a "web.app" domain extension.

```
+ Firebase initialization complete!
PS C:\Users\Admin\Desktop\project1> firebase deploy

=== Deploying to 'my-react-app-5d208'...

i deploying hosting
i hosting[my-react-app-5d208]: beginning deploy...
i hosting[my-react-app-5d208]: found 1 files in build
+ hosting[my-react-app-5d208]: file upload complete
i hosting[my-react-app-5d208]: finalizing version...
+ hosting[my-react-app-5d208]: version finalized
i hosting[my-react-app-5d208]: releasing new version...
+ hosting[my-react-app-5d208]: release complete
```

A screenshot of a web browser window. The address bar shows 'my-react-app-5d208.web.app'. The browser has tabs for 'YouTube', 'Maps', and 'New Tab'. The main content area displays a white box with the text 'Welcome Firebase Hosting Setup Complete'. Below this, it says 'You're seeing this because you've successfully setup Firebase Hosting. Now it's time to go build something extraordinary!'. At the bottom of the box is a blue button with the text 'OPEN HOSTING DOCUMENTATION'. Below the white box, there is a small line of text: 'Firebase SDK loaded with auth, database, firestore, functions, messaging, storage, analytics, remoteConfig, performance'.

