

## GUIDE

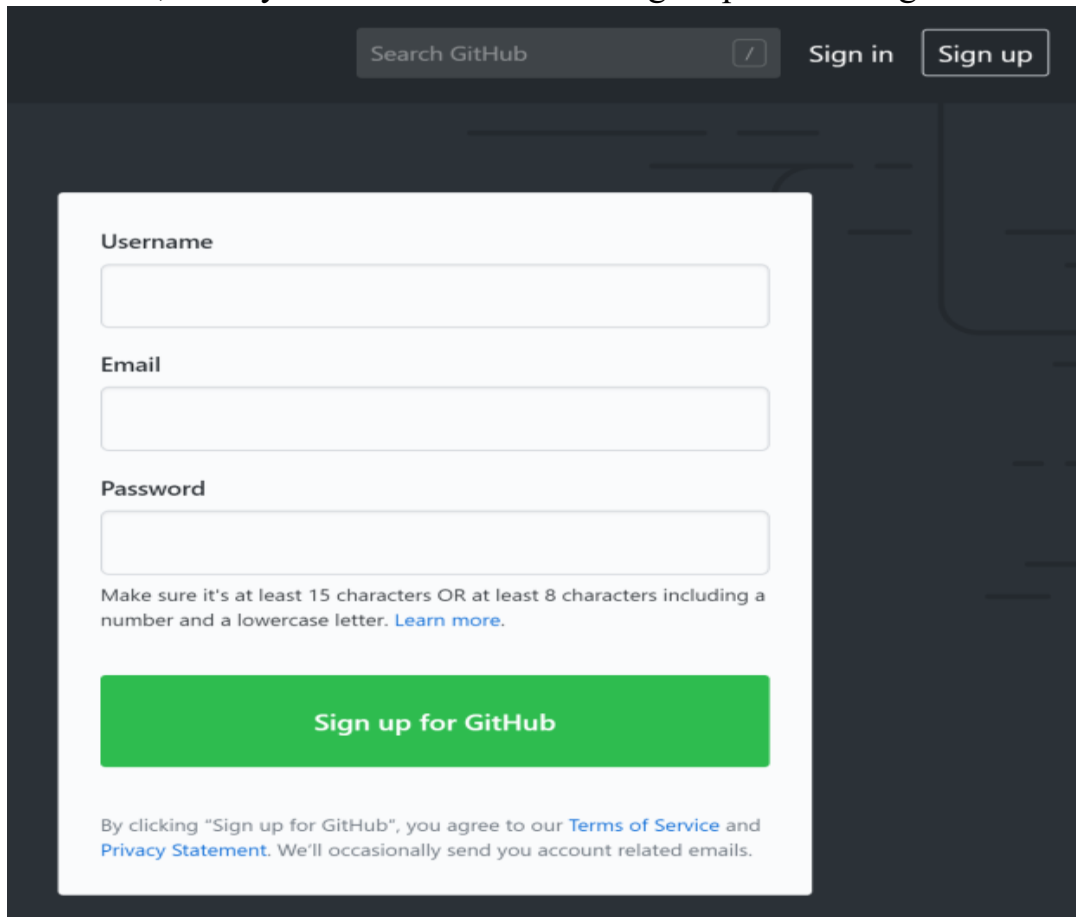
### TO DEPLOYING A REACT APP TO GITHUB PAGES

GitHub Pages is a free service from GitHub that allows you to host a website directly from a repository on GitHub. It is a great tool for deploying static websites such as blogs, personal pages, or project documentation.

#### I. Create a Repository on GitHub.

##### 1. Create an account and log in to GitHub.

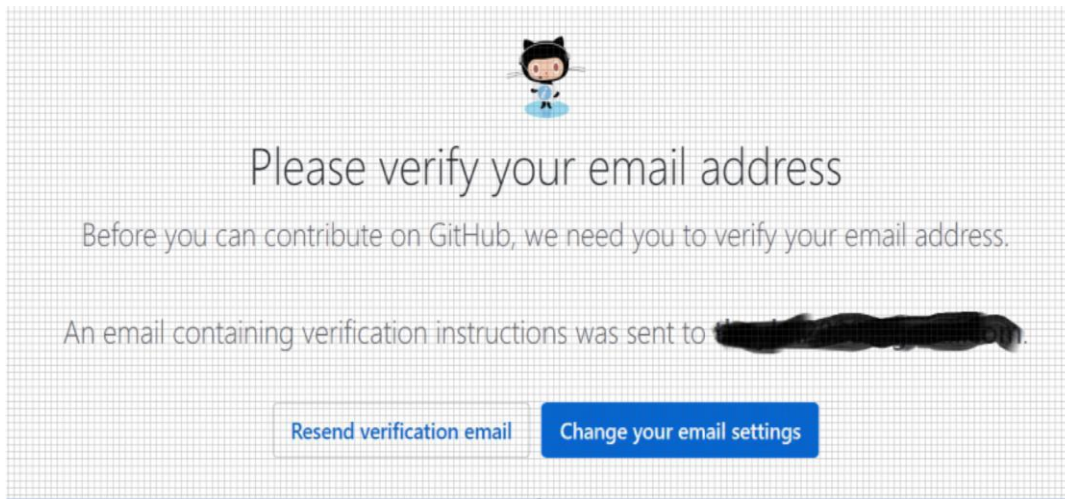
Go to <https://github.com/> to log in or sign up for an account. If you don't have an account, enter your information in the Sign Up form to register.



The screenshot shows the GitHub Sign Up page. At the top, there is a search bar labeled "Search GitHub" and two buttons: "Sign in" and "Sign up". Below these is a large white form box. Inside the form box, there are three input fields: "Username", "Email", and "Password". Below the "Password" field, there is a note: "Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. [Learn more.](#)". At the bottom of the form box is a large green button labeled "Sign up for GitHub". Below the button, there is a line of text: "By clicking 'Sign up for GitHub', you agree to our [Terms of Service](#) and [Privacy Statement](#). We'll occasionally send you account related emails."

After providing all the required information and completing the registration process.

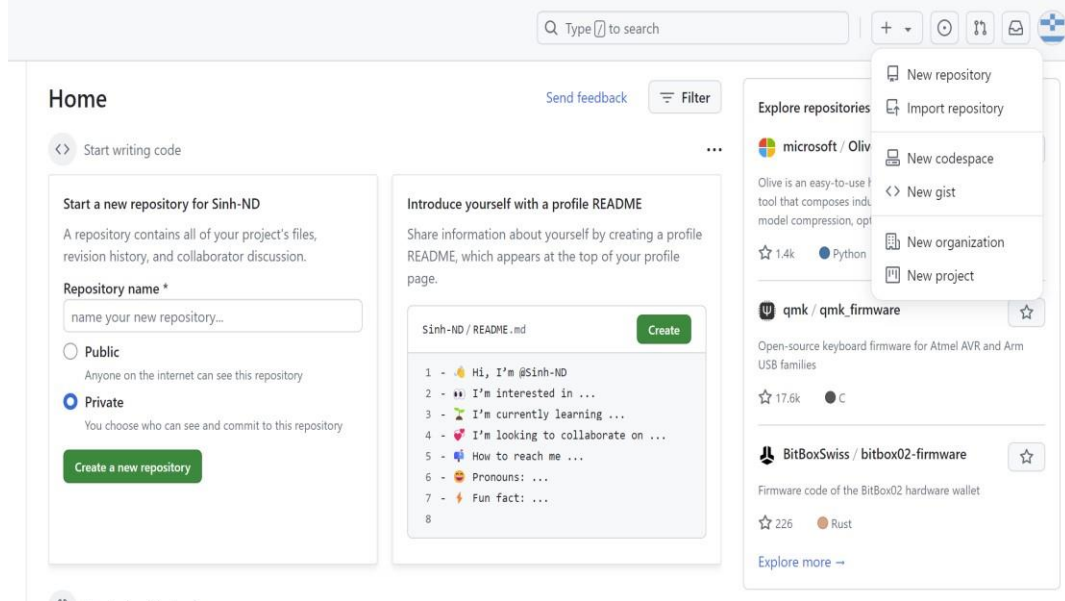
# GITHUB SETUP



Check your email to see if GitHub has sent a verification link. If so, click on [Verify email address] to proceed with verification. If you haven't received it, go back to the GitHub homepage and click [Resend verification email].

## 2. Create a New Repository.

- Click the plus sign icon in the upper-right corner of the screen and select [New Repository].




- Name your repository, for example: project2
- Choose between Public or Private depending on your needs.
- Click Create repository to complete the process.

## Create a new repository



A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk (\*).

Owner \*  
 Biubietnam ▾ / Repository name \*  
  
✔ project2 is available.

Great repository names are short and memorable. Need inspiration? How about **redesigned-sniffle** ?

Description (optional)

- ☒  **Public**  
Anyone on the internet can see this repository. You choose who can commit.
- ☐  **Private**  
You choose who can see and commit to this repository.

Initialize this repository with:

- ☐ **Add a README file**  
This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore

.gitignore template: None ▾

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license

License: None ▾

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

 You are creating a public repository in your personal account.

Create repository

## II. Deploying a React App to GitHub Pages

Open the terminal in Visual Studio Code:

1. **Install GitHub Pages** `npm install --save gh-pages`

2. **Add to package.json File**

```
"homepage": "https://biubietnam.github.io/project2 ",  
"predeploy": "npm run build",  
"deploy": "gh-pages -d build",
```

```
"homepage": "https://biubietnam.github.io/project2",  
"name": "eproject",  
"version": "0.1.0",  
"private": true,  
"dependencies": {  
  "@emotion/react": "^11.13.3",  
  "@emotion/styled": "^11.13.0",  
  "@mui/icons-material": "^6.1.5",  
  "@mui/material": "^6.1.4",  
  "@testing-library/jest-dom": "^5.17.0",  
  "@testing-library/react": "^13.4.0",  
  "@testing-library/user-event": "^13.5.0",  
  "gh-pages": "^6.2.0",  
  "react": "^18.3.1",  
  "react-dom": "^18.3.1",  
  "react-router-dom": "^6.27.0",  
  "react-scripts": "5.0.1",  
  "web-vitals": "^2.1.4"
```

```
"scripts": {  
  "predeploy": "npm run build",  
  "deploy": "gh-pages -d build",  
  "start": "react-scripts start",  
  "build": "react-scripts build",  
  "test": "react-scripts test",  
  "eject": "react-scripts eject"  
},  
"eslintConfig": {  
  "extends": [  
    "react-app",  
    "react-app/jest"  
  ]  
},  
"browserslist": {  
  "production": [  
    ">0.2%",  
    "not dead",  
    "not op_mini all"  
  ],  
  "development": [  
    "last 1 chrome version",  
    "last 1 firefox version",  
    "last 1 safari version"  
  ]  
}
```

### 3. Push Code to GitHub

- git init ○ git add .
- git commit -m "first commit" ○ git branch -M main
- git remote add origin <https://github.com/Biubietnam/project2>
- git push -u origin main

## 4. Run the deploy command ○

npm run deploy

After a successful deployment, go back to GitHub: Open your repository on GitHub, click on the "Settings" tab at the top, and scroll down to the "Pages" section. Wait for about 1-5 minutes and you should see the results as shown below.

