

Deployment Document

Step 1: Sign Up / Log In to Vercel

I go to Vercel's website.

I click Sign Up (if I don't have an account) or Log In (if I already have an account).

I sign up/log in using GitHub

Step 2: Push My Project to GitHub

I open my terminal and navigate to my Next.js project folder.

I run the following commands to initialize git in my project folder:

```
bash
```

CopyEdit

```
git init
```

```
git add .
```

```
git commit -m "Initial commit"
```

Step 3: Connect My GitHub Repository to Vercel

I go to my Vercel Dashboard.

I click on New Project.

Vercel asks me to connect a Git provider, so I select GitHub.

I authorize Vercel to access my GitHub repositories.

Vercel shows a list of my GitHub repositories. I select the repository I just pushed my Next.js project to.

I click Import.

Step 4: Configure My Project Settings

Project Name: Vercel automatically uses my repository name, but I can change it if I want.

Root Directory: Since my Next.js project is at the root of my repository, I leave this blank.

Framework Preset: Vercel automatically detects it's a Next.js app, so I don't need to change anything.

I click Continue.

Step 6: Deploy My Project

After configuring everything, I click Deploy.

Vercel starts the build and deployment process.

I wait a few moments for the deployment to finish, and Vercel provides me with a live URL

Step 8: Continuous Deployment

Vercel automatically sets up

Continuous Deployment. This means that every time I push new code to my GitHub repository, Vercel redeploys my app.

If I create a Pull Request on GitHub, Vercel generates a Preview Deployment URL so I can test the changes before merging.