

GitHub Deployment Guide for Ian Wu Portfolio

Quick Start - Push to GitHub

Step 1: Navigate to the deployment directory

```
cd /home/ubuntu/ian_wu_portfolio/github-deploy
```

Step 2: Initialize Git and push to GitHub

```
# Initialize Git repository
git init

# Configure Git (use your GitHub email)
git config user.name "Ian Wu"
git config user.email "ian.kin.wu@gmail.com"

# Add all files
git add .

# Create initial commit
git commit -m "Initial commit: Ian Wu Portfolio"

# Add your GitHub repository (replace with your actual repository URL)
git remote add origin https://github.com/YOUR_USERNAME/YOUR_REPO_NAME.git

# Push to GitHub
git branch -M main
git push -u origin main
```

Deployment Options for Your Next.js Portfolio

★ Option 1: Vercel (RECOMMENDED - Easiest & Best for Next.js)

1. Go to <https://vercel.com> (<https://vercel.com>)
2. Sign in with your GitHub account
3. Click "Add New Project"
4. Import your portfolio repository from GitHub
5. Vercel will automatically:
 - Detect it's a Next.js project
 - Install dependencies
 - Build and deploy your site
6. Your site will be live at: `https://your-project.vercel.app`
7. **Connect your custom domain:**
 - Go to Project Settings → Domains
 - Add your custom domain
 - Follow the DNS configuration instructions

Benefits:

- Zero configuration needed
 - Automatic deployments on every push to main
 - Built-in SSL certificate
 - Global CDN
 - Perfect Next.js support (SSR, API routes, etc.)
 - Free for personal projects
-

Option 2: Netlify

1. Go to <https://netlify.com> (<https://netlify.com>)
 2. Sign in with GitHub
 3. Click “Add new site” → “Import an existing project”
 4. Select your repository
 5. Configure build settings:
 - Build command: `npm run build`
 - Publish directory: `.next`
 6. Click “Deploy”
 7. Connect your custom domain in Domain settings
-

Option 3: GitHub Pages (Static Export Only)

 **Note:** GitHub Pages only supports static sites. You’ll need to export your Next.js app.

Step-by-Step for GitHub Pages:

1. **Modify your `next.config.js` file:**

```
/** @type {import('next').NextConfig} */
const nextConfig = {
  output: 'export',
  images: { unoptimized: true },
  trailingSlash: true,
  basePath: '/YOUR_REPO_NAME', // Only if deploying to username.github.io/repo-name
};

module.exports = nextConfig;
```

1. **Build the static export:**

```
cd /home/ubuntu/ian_wu_portfolio/github-deploy
npm install
npm run build
```

1. **Deploy the out directory:**

```
# Create gh-pages branch
git checkout --orphan gh-pages
git rm -rf .
cp -r out/* .
git add .
git commit -m "Deploy to GitHub Pages"
git push origin gh-pages --force
```

1. Enable GitHub Pages:

- Go to your repository on GitHub
- Settings → Pages
- Source: Select `gh-pages` branch
- Save
- Your site will be at: `https://YOUR_USERNAME.github.io/YOUR_REPO_NAME`

Connecting Your Custom Domain

For Vercel:

1. Project Settings → Domains
2. Add your domain
3. Update your DNS records:
 - Add A records or CNAME as instructed by Vercel
 - Wait for DNS propagation (can take up to 48 hours)

For Netlify:

1. Domain settings → Add custom domain
2. Update DNS records as instructed
3. Netlify will automatically provision SSL

For GitHub Pages:

1. Create a `CNAME` file in your `gh-pages` branch:

```
echo "your-domain.com" > CNAME
git add CNAME
git commit -m "Add custom domain"
git push origin gh-pages
```

1. In your domain registrar, add:
 - A records pointing to GitHub's IPs
 - Or CNAME record pointing to `YOUR_USERNAME.github.io`
-

Common Git Commands

```
# Check status of your files
git status

# Add specific files
git add filename.txt

# Add all changed files
git add .

# Commit changes
git commit -m "Description of changes"

# Push changes to GitHub
git push origin main

# Pull latest changes from GitHub
git pull origin main

# View commit history
git log --oneline

# Create a new branch
git checkout -b new-feature

# Switch back to main branch
git checkout main

# View remote repositories
git remote -v
```

Updating Your Portfolio

After making changes to your portfolio:

```
cd /home/ubuntu/ian_wu_portfolio/github-deploy
git add .
git commit -m "Update portfolio content"
git push origin main
```

If you're using Vercel or Netlify, they will automatically detect the changes and redeploy your site!

Troubleshooting

“Permission denied (publickey)” error

- You need to set up SSH keys for GitHub
- Or use HTTPS URL with personal access token
- Guide: <https://docs.github.com/en/authentication>

Build fails on deployment

- Make sure all dependencies are in `package.json`
- Test the build locally first: `npm run build`
- Check the build logs on your deployment platform


Custom domain not working

- DNS changes can take 24-48 hours to propagate
- Check your DNS records are configured correctly
- Make sure SSL is enabled in your deployment platform

Images not showing on GitHub Pages

- GitHub Pages requires `images: { unoptimized: true }` in `next.config.js`
- Make sure image paths are correct

My Recommendation

 **Use Vercel** - It's the easiest and most powerful option for Next.js sites:

1. Push your code to GitHub (instructions at the top)
2. Go to vercel.com and import your repository
3. Done! Your site will be live in minutes
4. Connect your custom domain in Vercel settings

No build configuration needed, automatic deployments, and perfect Next.js support!

Resources

- **Git Documentation:** <https://git-scm.com/doc>
- **GitHub Guides:** <https://guides.github.com>
- **Next.js Documentation:** <https://nextjs.org/docs>
- **Vercel Documentation:** <https://vercel.com/docs>
- **Netlify Documentation:** <https://docs.netlify.com>
- **GitHub Pages Documentation:** <https://docs.github.com/pages>

Good luck with your deployment! 🚀