

The following is a **one-page GitHub Pages deployment checklist** that can be used as a step-by-step guide.

✓ GitHub Pages Deployment Checklist

1. Prepare Your Local Project

- Make sure your index.html file is inside your project folder.
 - (Optional) Add README.md with instructions/tutorial.
 - Test locally by opening index.html in a browser.
-

2. Create GitHub Repository

- Go to [GitHub](#).
 - Click **New Repository**.
 - Name it (e.g., tribute-page).
 - Keep it **Public**.
 - Do *not* initialize with README, .gitignore, or license (you already have files).
-

3. Connect Local Folder to GitHub

In your terminal (Git Bash or VS Code terminal):

```
cd path/to/your/project
```

```
git init
```

```
git remote add origin https://github.com/YOUR-USERNAME/tribute-page.git
```

```
git add .
```

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

```
git branch -M main
```

```
git push -u origin main
```

4. Enable GitHub Pages

- Go to your repo on GitHub.
 - Click **Settings** (top menu).
 - Scroll down to **Pages** (left sidebar).
 - Under *Source*, select:
 - **Branch:** main
 - **Folder:** / (root)
 - Save.
-

5. Get Your Live Link

- Wait 1–2 minutes for GitHub to build your site.
 - Refresh the **Pages** section → copy your link (e.g., <https://yourusername.github.io/tribute-page/>).
 - Paste the link into your README.md under **Project Preview**.
-

6. Share 🎉

- Test your site on desktop and mobile.
 - Share the link on Facebook or LinkedIn!
-

👉 That's it. You now have a repeatable process you can use for every future project (CSS, JS, etc.).

Would you like me to also adapt this into a **ready-to-paste section at the bottom of your README** (so it lives inside the repo too), or keep it as your private checklist?