

Setting up Slidev on GitHub Pages

This tutorial walks you through the full process of setting up Slidev presentations that are built automatically on GitHub Pages — so you can edit `slides.md` right on GitHub, no local build required.



1. Fork This Repository

The easiest way to get started is to **fork this repository**:

1. Click the **Fork** button (top-right corner of this page).
2. Choose your GitHub account.
3. Wait a few seconds — a copy of this repo will appear under your account.
4. That's it! You're now ready to customize your own Slidev presentation.

You'll automatically have all necessary files (`slides.md` , `package.json` , GitHub Actions workflow, etc.) configured for you.



No need to download or upload anything manually — everything runs in the cloud.



2. Understand the Folder Structure

Your forked repository contains these files:

```
slidev-template/
├── .github/
│   └── workflows/
│       └── deploy.yml           ← GitHub Action for auto-deploy
├── public/
│   └── favicon.svg             ← Optional favicon
├── slides.md                   ← Your presentation content (editable on GitHub)
├── package.json                ← Defines scripts & dependencies
├── package-lock.json           ← Auto-generated dependency lock file
├── README.md                   ← Repo overview and usage
├── .gitignore                  ← Ignores build/dependency files
└── (generated later)
    └── dist/                   ← Build output created by GitHub Actions
```

Everything happens automatically — you only edit `slides.md`.



3. GitHub Actions – Build & Deploy

The workflow file (`.github/workflows/deploy.yml`) does three key things:

1. Installs dependencies with `npm ci`.
2. Builds your slides using Slidev (`npm run build`).
3. Publishes the result to the `gh-pages` branch.

Each time you push or edit your slides, GitHub Actions builds and deploys your deck.

It automatically detects whether your repo is a `user/org` site (`username.github.io`) or `project` site (`username.github.io/repo/`).



4. Enable GitHub Pages

Once your first deployment finishes:

1. Go to **Settings** → **Pages** in your repository.
2. Set the **Source** to the `gh-pages` branch.
3. Click **Save**.


After a minute or two, your slides are live at:

- **User/Org Site:** `https://username.github.io/`
- **Project Site:** `https://username.github.io/repo/`



5. Edit Your Slides

Simply edit the `slides.md` file directly on GitHub:

1. Go to the file → click the  **Edit** icon.
2. Modify your content (Markdown syntax).
3. Scroll down → **Commit changes**.

The workflow runs again and your updated slides appear online.

6. Markdown Basics for Slidev

Each slide is separated by `---`.

```
# Welcome to Slidev 🖐️
```

```
This is your first slide.
```

```
---
```

```
## Next Slide
```

- Use bullet points
- Add code snippets:

```
```ts {all|1|2-3}  
export function greet(name: string) {
 return `Hello, ${name}!`
}
```

```
```
```

Other features:

- Press P for presenter mode
- Press D to draw on slides



7. Customize the Theme

Slidev uses themes from NPM packages (like `@slidev/theme-default`). You can switch themes in the frontmatter of `slides.md`:

```
---  
theme: seriph  
---
```

Available themes: <https://sli.dev/themes>



8. Custom Domain (Optional)

To use a custom domain:

1. Add a file named `CNAME` in your `public/` folder.
2. Inside, put your domain name, e.g. `slides.example.com`.

GitHub Pages will handle it automatically.



9. Troubleshooting

If your deck doesn't load correctly:

- Check that GitHub Pages is set to `gh-pages` branch.
- Wait 1–2 minutes after a commit for deployment.
- Check the **Actions** tab for logs.



10. Done!

You now have a fully working **Slidev** presentation running on **GitHub Pages** — automatically built and deployed from your Markdown file.



Now edit `slides.md` to create your first deck and publish it instantly!

