# A statically generated blog example using Next.js and TinaCMS

This example showcases Next.js's [Static Generation](#) feature using [TinaCMS](#) as the CMS and editor.

> *This boilerplate demonstrates a basic usage and best practices. If you are looking for a more feature rich Tina experience with contextual editing. check out [tina-cloud-starter](#).*

## Related examples

- [WordPress](#)
- [DatoCMS](#)
- [Sanity](#)
- [TakeShape](#)
- [Prismic](#)
- [Contentful](#)
- [Strapi](#)
- [Agility CMS](#)
- [ButterCMS](#)
- [Storyblok](#)
- [GraphCMS](#)
- [Kontent](#)
- [Umbraco Heartcore](#)
- [Blog Starter](#)
- [Builder.io](#)

## How to use

Execute [`create-next-app`](#) with [npm](#) or [Yarn](#) to bootstrap the example:

```
npx create-next-app --example cms-tina cms-tina-app
# or
yarn create next-app --example cms-ghost cms-tina-app
# or
pnpm create next-app -- --example cms-ghost cms-tina-app
```

### Setp 1. Run Next.js in development mode

To get started, no configuration is needed for local development and editing.

```
npm install
npm run tina-dev

# or

yarn install
yarn tina-dev
```

Your blog should be up and running on [http://localhost:3000](http://localhost:3000)! If it doesn't work, post on [GitHub discussions](#).

**Step 2. Editing blog posts.**

Tina is git backed and uses markdown, JSON or MDX to power websites. To enter edit mode locally you just need to visit http://localhost:3000/admin

You can then select the collection "Blog Posts" and then the content you would like to edit.

Once you hit save, Tina will use our graphQL modify the content on your filesystem.

**Step 4. Deploy on Vercel**

You can deploy this app to the cloud with Vercel (Documentation).

**Deploy Your Local Project**

To deploy your local project to Vercel, push it to GitHub. Once you have pushed to GitHub, sign up for your Tina Cloud account at https://app.tina.io/register. Then follow the steps below:

1. Select new project
2. Select Import your site
3. Follow steps to connect your GitHub repo.
4. Copy your Client ID

Then import to Vercel.

**Important**: When you import your project on Vercel, make sure to click on **Environment Variables** and set NEXT_PUBLIC_TINA_CLIENT_ID to the client ID above.

Once you have successfully deployed to Vercel, go back to your Tina dashboard and under the project configuration enter the url in the Site URL(s) for example: https://tina-cms.vercel.app.