

Toucan Static Site Generator: Quick Installation Guide

Toucan is a Swift-based static site generator designed for simplicity and performance. This guide walks you through the installation process and helps you get started with creating your first static website.

Disclaimer

Toucan product is currently in its beta phase, which means you might encounter issues if you attempt to custom modify the template using Toucan. While we're actively refining and improving the experience, some features may be unstable or require manual adjustments. TRY-O template is fully operational as is! We appreciate your flexibility as we work towards a more polished version! 🚀

Installation Overview

You can install Toucan on macOS or Linux using the following methods:

- [Homebrew](#)
 - [Mint \(macOS only\)](#)
 - Prebuilt binary from [GitHub Releases](#)
 - `.pkg` installer for macOS
 - `.deb` or `.rpm` packages for Linux
 - Compile from source
-

macOS Installation

Homebrew

```
brew tap toucansites/toucan
brew install toucan
```

Or directly:

```
brew install toucansites/toucan/toucan
```

Mint

```
mint install toucansites/toucan
```

Or:

```
mint run toucansites/toucan
```

GitHub Releases (Binary or PKG)

Download from: [Toucan GitHub Releases](#)

- For binary:

```
unzip toucan-macos-1.0.0.beta.4.zip
sudo mv toucan /usr/local/bin/
```

- For PKG installer:

- GUI: double-click `.pkg` file
- CLI:

```
sudo installer -pkg toucan-1.0.0-beta.4.pkg -target /
```

Compile from Source

```
git clone https://github.com/toucansites/toucan.git
cd toucan
make install
./scripts/run-chmod.sh
which toucan
```

Linux Installation

Homebrew (Linuxbrew)

```
brew tap toucansites/toucan  
brew install toucan
```

Or:

```
brew install toucansites/toucan/toucan
```

Native Packages

Debian / Ubuntu

```
wget https://github.com/toucansites/toucan/releases/download/1.0.0.beta.4/toucan-linux-amd64-1.0.0.beta.4.deb  
sudo dpkg -i toucan-linux-amd64-1.0.0.beta.4.deb
```

Fedora / RHEL / openSUSE

```
wget https://github.com/toucansites/toucan/releases/download/1.0.0.beta.4/toucan-linux-x86_64-1.0.0.beta.4.rpm  
sudo rpm -i toucan-linux-x86_64-1.0.0.beta.4.rpm
```

Compile from Source

```
git clone https://github.com/toucansites/toucan.git  
cd toucan  
make install  
./scripts/run-chmod.sh  
which toucan
```

Getting Started with Toucan

Step 1: Initialize a New Site

1. Create a new site by running:
 - Run the following command in your terminal, replacing **my-site** with your desired project folder name:

```
toucan init my-site
```

- Toucan will generate a folder with your chosen name, including default configuration files and templates.

2. Navigate to your new site folder:

```
cd my-site
```

Once your site is initialized, your project will have the following structure:

```
my-site/
├── .github/
│   └── workflows/
│       └── deploy.yml      # Deploy file for Github Actions
├── LICENSE                 # License file for the project
├── Makefile                # Makefile for building and managing the project
├── README.md               # Project README file
└── src/                    # Source files for content and templates
```

Step 2: Generate Your Site

1. Run the following command to generate the static files:

```
toucan generate
```

- Toucan processes your content and templates, producing static HTML files in the `dist` directory.

Step 3: Watch for changes (only for macOS for now)

The watch command monitors a source directory for changes and automatically regenerates the site whenever changes are detected.

1. Run the following command to enable file watching:

```
toucan watch
```

Step 4: Serve Your Site Locally

1. Start a local development server:

```
toucan serve
```

2. Open your browser and navigate to:

```
http://localhost:3000
```

- This allows you to preview your site locally.

Copy Theme and Default Contents

Delete files

Delete all files and sub-folders from the:

1. **my-site/src/assets** folder
2. **my-site/src/contents** folder
3. **my-site/src/pipelines** folder
4. **my-site/src/themes/default** folder
5. **my-site/src/types** folder

Copy files

1. Copy all files from the **try-o-theme-1.0.0-beta.2/assets** folder in the ZIP file and paste them into the **my-site/src/assets** folder.
2. Copy all files from the **try-o-theme-1.0.0-beta.2/contents** folder in the ZIP file and paste them into the **my-site/src/contents** folder.
3. Copy all files from the **try-o-theme-1.0.0-beta.2/pipelines** folder in the ZIP file and paste them into the **my-site/src/pipelines** folder.
4. Copy all files from the **try-o-theme-1.0.0-beta.2/theme** folder in the ZIP file and paste them into the **my-site/src/themes/default** folder.
5. Copy all files from the **try-o-theme-1.0.0-beta.2/types** folder in the ZIP file and paste them into the **my-site/src/types** folder.
6. Create a folder **blocks**, if not exists inside the **my-site/src/** folder.
7. Copy all files from the **try-o-theme-1.0.0-beta.2/blocks** folder in the ZIP file and paste them into the **my-site/src/blocks** folder.
8. Copy the file **try-o-theme-1.0.0-beta.2/config.yml** in the ZIP file and override/replace it in the **my-site/src** folder.
9. Copy the file **try-o-theme-1.0.0-beta.2/site.yml** in the ZIP file and override/replace it in the **my-site/src** folder.

After successful file copies, your project will have the following structure:

```

my-site/
├── .github/
│   └── workflows/
│       └── deploy.yml
├── dist
├── src/
│   ├── assets/
│   ├── blocks/
│   ├── contents/
│   │   ├── 404
│   │   ├── about
│   │   ├── assets
│   │   ├── authors
│   │   ├── home
│   │   ├── lists
│   │   ├── posts
│   │   ├── tags
│   │   └── site.yml
│   ├── pipelines/
│   │   ├── 404.yml
│   │   ├── api.yml
│   │   ├── html.yml
│   │   ├── redirect.yml
│   │   ├── rss.yml
│   │   └── sitemap.yml
│   ├── themes/
│   │   └── default/
│   │       ├── assets
│   │       ├── blocks
│   │       ├── templates
│   │       └── types
│   ├── types/
│   ├── config.yml
│   └── site.yml

```

Once all files are successfully copied, regenerate your site to apply the changes:

- First run the following command:

```
toucan generate
```

- After successful generation, run the following command:

```
toucan serve
```

The theme comes with a variety of sample content to help you quickly understand how to

structure and manage your site. These include:

Default Contents

1. Tags:

- Multiple tags are provided in **src/contents/tags/**.
- This demonstrates how tags are created and displayed.

2. Authors:

- A test author is included in **src/contents/authors/**.
- This shows how to add and customize author profiles.

3. Posts:

- Several sample posts are included in **src/contents/posts/**.
- These showcase how to write Markdown-based posts with metadata and images.

4. 404 Page:

- A preconfigured “Page Not Found” template is located in **src/contents/404/**.

5. About Page:

- A sample “About” page is included in **src/contents/about/** to demonstrate static page creation.

6. Assets:

- Default images and icons are stored in **src/contents/assets/**.
- Includes logos, placeholders, and theme-specific assets.

7. Home Page Content:

- A preconfigured home page setup is included in **src/contents/home/**.

You can use these sample files as a reference or starting point for your project. Once you're familiar with the structure, replace them with your own custom content.

Happy building with Toucan!

Troubleshooting

If you encounter any issues during installation or setup, try the following steps:

- Ensure Swift is installed and correctly configured.
- Check your `PATH` to ensure the Toucan binary is accessible.
- Review the logs for any error messages and report issues on the [Toucan GitHub Issues page](#).

Resources

- **Official Repository:** <https://github.com/toucansites/toucan>
- **Swift Downloads:** <https://swift.org/download/>