



Core Concepts

Key Terminology

Posts	The primary content type in Ghost, representing articles or blog entries.
Pages	Static content like 'About' or 'Contact' pages.
Tags	Used to categorize and organize posts.
Authors	Users who create and publish content.
Members	Users who can subscribe to your content.
Themes	Control the visual appearance of your site.

Theme Development

Theme Structure

Ghost themes are built using Handlebars.js for templating.

Key files include:

- `default.hbs`: The base template for all pages.
- `index.hbs`: Template for the homepage/post listings.
- `post.hbs`: Template for individual posts.
- `page.hbs`: Template for individual pages.
- `tag.hbs`: Template for tag archives.
- `author.hbs`: Template for author archives.
- `partials/`: Directory for reusable template snippets.

API & Integrations

Content API

Ghost provides a Content API for fetching posts, pages, tags, and authors.

This allows you to use Ghost as a headless CMS and build custom frontends using any technology.

Access requires an API key, which you can generate in the Ghost admin panel.

Example request using the Content API:

```
GET /ghost/api/content/posts/?key=[your_api_key]
```

Command Line Interface (Ghost-CLI)

Installation

Install Ghost-CLI globally using npm:

```
npm install ghost-cli -g
```

Admin Interface

Access the Ghost admin panel by navigating to `/ghost` on your domain (e.g., `yourdomain.com/ghost`).

This is where you create, edit, and manage your content, users, and settings.

The admin interface is divided into sections for Posts, Pages, Members, Settings, and Explore.

Handlebars Helpers

<code>{{@site}}</code>	Access site-wide data (e.g., name, description, logo).
<code>{{@post}}</code>	Access post-specific data (e.g., title, content, url).
<code>{{@page}}</code>	Access page-specific data.
<code>{{url}}</code>	Output the URL of a post, page, tag, or author.
<code>{{content}}</code>	Output the post or page content.
<code>{{excerpt}}</code>	Output the post excerpt.

Admin API

The Admin API allows you to manage content, users, and settings programmatically.

It requires authentication using JWT (JSON Web Tokens).

Use cases include building custom integrations and automating content management tasks.

Common Commands

<code>ghost install</code>	Install a new Ghost instance.
<code>ghost update</code>	Update an existing Ghost instance to the latest version.
<code>ghost start</code>	Start the Ghost process.
<code>ghost stop</code>	Stop the Ghost process.
<code>ghost restart</code>	Restart the Ghost process.
<code>ghost config</code>	Configure Ghost settings.

Content Editor (Koenig)

Ghost uses a block-based editor called Koenig for creating posts and pages.

Each piece of content is a 'card', allowing for flexible layout and content types.

Common card types include Markdown, HTML, image, embed, and callout.

Theme Settings

Ghost themes can be configured using `package.json` to define settings that can be customized in the Ghost admin panel.

This allows users to easily change things like accent colors, social media links, and other theme-specific options without directly editing the theme files.

Webhooks

Ghost supports webhooks, allowing you to trigger actions in other applications when certain events occur (e.g., post published, member created).

This enables seamless integration with other services and workflows.

Configuration

Ghost-CLI helps you configure your Ghost installation, including setting up the database, mail server, and SSL certificate.

Run `ghost config` to interactively set these options.