

Hosting a static website on Google App Engine

You can use Google App Engine to host a static website. Static web pages can contain client-side technologies such as HTML, CSS, and JavaScript. Hosting your static site on App Engine can cost less than using a traditional hosting provider, as App Engine provides a free tier.

Sites hosted on App Engine are hosted on the `appspot.com` subdomain, such as `[my-project-id].appspot.com`. After you deploy your site, you can map your own domain name to your App Engine-hosted website.

Before you begin

Before you can host your website on Google App Engine:

1. Create a new GCP Console project or retrieve the project ID of an existing project to use:

[GO TO THE PROJECTS PAGE](#)

Tip: You can retrieve a list of your existing project IDs with the [gcloud command line tool](#) (`#before_you_begin`).

2. Install and then initialize the Google Cloud SDK:

[DOWNLOAD THE SDK](#)

Creating a website to host on Google App Engine

Basic structure for the project

This guide uses the following structure for the project:

- `app.yaml`: Configure the settings of your App Engine application.
- `www/`: Directory to store all of your static files, such as HTML, CSS, images, and JavaScript.

- `css/`: Directory to store stylesheets.
 - `style.css`: Basic stylesheet that formats the look and feel of your site.
- `images/`: Optional directory to store images.
- `index.html`: An HTML file that displays content for your website.
- `js/`: Optional directory to store JavaScript files.
- Other asset directories.

Creating the `app.yaml` file

The `app.yaml` file is a configuration file that tells App Engine how to map URLs to your static files. In the following steps, you will add handlers that will load `www/index.html` when someone visits your website, and all static files will be stored in and called from the `www` directory.

Create the `app.yaml` file in your application's root directory:

1. Create a directory that has the same name as your project ID. You can find your project ID in the [Console](https://console.cloud.google.com/) (<https://console.cloud.google.com/>).
2. In directory that you just created, create a file named `app.yaml`.
3. Edit the `app.yaml` file and add the following code to the file:

```
runtime: python27
api_version: 1
threadsafe: true

handlers:
- url: /
  static_files: www/index.html
  upload: www/index.html

- url: /(.*)
  static_files: www/\1
  upload: www/(.*)
```



More reference information about the `app.yaml` file can be found in the [app.yaml reference documentation](https://cloud.google.com/appengine/docs/standard/python/config/appref) (<https://cloud.google.com/appengine/docs/standard/python/config/appref>).

Creating the `index.html` file

Create an HTML file that will be served when someone navigates to the root page of your website. Store this file in your `www` directory.

```
<html>
  <head>
    <title>Hello, world!</title>
    <link rel="stylesheet" type="text/css" href="/css/style.css">
  </head>
  <body>
    <h1>Hello, world!</h1>
    <p>
      This is a simple static HTML file that will be served from Google App
      Engine.
    </p>
  </body>
</html>
```

Deploying your application to App Engine

When you deploy your application files, your website will be uploaded to App Engine. To deploy your app, run the following command from within the root directory of your application where the `app.yaml` file is located:

```
gcloud app deploy
```

Optional flags:

- Include the `--project` flag to specify an alternate GCP Console project ID to what you initialized as the default in the `gcloud` tool. Example: `--project [YOUR_PROJECT_ID]`
- Include the `-v` flag to specify a version ID, otherwise one is generated for you. Example: `-v [YOUR_VERSION_ID]`

To learn more about deploying your app from the command line, see [Deploying a Python 2 App](https://cloud.google.com/appengine/docs/python/tools/uploadinganapp) (<https://cloud.google.com/appengine/docs/python/tools/uploadinganapp>).

Viewing your application

To launch your browser and view the app at `https://[YOUR_PROJECT_ID].appspot.com`, run the following command:

```
gcloud app browse
```



What's next

[Serve your App Engine-hosted website from a custom domain](#)

(<https://cloud.google.com/appengine/docs/standard/python/console/using-custom-domains-and-ssl>).

Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 4.0 License](https://creativecommons.org/licenses/by/4.0/) (<https://creativecommons.org/licenses/by/4.0/>), and code samples are licensed under the [Apache 2.0 License](https://www.apache.org/licenses/LICENSE-2.0) (<https://www.apache.org/licenses/LICENSE-2.0>). For details, see our [Site Policies](https://developers.google.com/terms/site-policies) (<https://developers.google.com/terms/site-policies>). Java is a registered trademark of Oracle and/or its affiliates.

Last updated August 8, 2019.