

Hosting Your Website for Free Using GitHub. A Beginner's Guide to Hosting Websites on GitHub

By cmd@hacker~\$

www.tca.ac.tz



About the Author:

cmd@hacker~\$ is a cybersecurity expert, consultant, ethical hacker, and academician committed to helping individuals and organizations understand digital vulnerabilities and strengthen their security posture. With expertise in penetration testing, network defense, secure software development, and cybersecurity research, cmd@hacker~\$ works to create a safer digital landscape for all users.

GitHub:

GitHub is a platform for hosting, managing, and collaborating on code repositories. It uses Git, a version control system, to track changes in projects. GitHub Pages, a feature of GitHub, allows users to host static websites directly from a repository for free.

GitHub Pages is a powerful feature provided by GitHub that allows users to host static websites (HTML, CSS, and JavaScript) directly from their GitHub repositories. Let's assume the GitHub username **Kasanga** to guide you step by step through the process of hosting a website.

This guide will walk you through hosting a website using GitHub Pages.

Create a GitHub Account:

Step 1: Create a GitHub Account (if you don't have one)

1. Go to [GitHub](#).
2. Sign up for a free account by filling in your details.
3. Choose your unique username carefully—this will appear in your website URL.
 - For this guide, the username is **Kasanga**, so the hosted website will eventually be available at:
<https://kasanga.github.io>
4. Confirm your email address to fully activate your account.

Step 2: Create a New Repository for Your Website

1. Log in to your GitHub account and click the + icon in the top-right corner, then select **New repository**.
2. Fill in the repository details:
 - **Repository Name:** Enter a name, e.g., `kasanga.github.io`
NB: The name before `(name).github.io` must be the same as the username of the account in GitHub.

- **Description:** Optionally describe your project (e.g., "Personal website hosted on GitHub Pages").
 - **Visibility:** Choose **Public** (so others can view your website).
3. (Optional) Check the box **Add a README file** to initialize the repository. This file can provide an overview of your project.
 4. Click **Create repository**.

Step 3: Upload Your Website Files

Your website needs basic files like index.html to function. These files can include:

- index.html: The main webpage file (**mandatory**).
- style.css: Optional file for styling your page.
- script.js: Optional file for JavaScript functionality.

To upload your files:

1. **Direct Upload via GitHub Interface:** (skip 2 below if you prefer to use this)
 - On the repository page (e.g., <https://github.com/kasanga/kasanga.github.io>), click **Add file > Upload files**.
 - Drag and drop your index.html and any other files into the upload area.
 - Write a **commit message** (e.g., "Add website files") and click **Commit changes**.
2. **Using Git Locally:** (skip 1 above if you prefer to use this)

If you're familiar with Git, you can upload your files using the following steps:

 - Clone the repository locally:

```
$git clone https://github.com/kasanga/kasanga.github.io.git
$cd kasanga.github.io
```

 - Add your website files to the project directory.
 - Stage and commit your changes:

```
$git add .
```

```
$git commit -m "Add website files" #replace with your website files
```

- Push the changes back to GitHub:

```
$git push origin main
```

Step 4: Enable GitHub Pages

Once your files are in the repository, enable GitHub Pages to make your website live:

1. Go to your repository page (e.g., <https://github.com/kasanga/kasanga.github.io>).
2. Click the **Settings** tab at the top.
3. In the left-hand menu, find and click **Pages** (under "Code and automation").
4. Under the **Source** section:
 - Select the branch where your files are stored (e.g., **main**).
 - Choose **Root** as the folder if your files are in the **main** directory.
5. Save your changes.

GitHub will display the URL of your website, which will look like:

<https://kasanga.github.io>

Step 5: Verify Your Website is Live

1. Wait a few minutes for GitHub Pages to build your site and Propagation of your web pages.
2. Open the provided URL (e.g., <https://kasanga.github.io>) in your browser.
3. You should see your website live!

With these steps, **Kasanga** can host a fully functional, professional-looking website for free!

