

# Custom domain for GitHub project pages

Asked 12 years, 10 months ago   Modified 9 months ago

Viewed 148k times



526

I have a `gh-pages` branch in one of my <http://github.com> repos. The GitHub project pages works fine if I go to <http://myuser.github.com/myrepo>



I want to setup a custom domain (`myexample.com`) that will serve up these project pages. I want **both**



`myexample.com` and `www.myexample.com` to serve up these project pages.



[GitHub pages help](#) says to make an A record and a CNAME record in your DNS. The A record makes sense, but I do not know what CNAME record to make in my DNS.

The `gh-pages` docs say to make a `CNAME` record for 'charlie.github.com' which is a user page repository. I do not have a user page repository - I only have a project repository and a `gh-pages` branch that I want to use for `myexample.com` and `www.myexample.com`.

*Do I need to make a user page repository just so I can use my project page for [www.myexample.com](http://www.myexample.com) and `myexample.com`?*

I would just try it, but I want to make sure this will work as I already have [www.myexample.com](http://www.myexample.com) live and don't want to make a mistake.

I emailed GitHub support and their response was

You can't have both point to the same gh-pages as far as I know.

I find it hard to believe they would only support A records for project pages.

Has anyone successfully done this before?

github

dns

Share

Improve this question

Follow

edited Mar 23 at 23:38



danronmoon

3,873 ● 5 ● 35 ● 58

asked Jan 31, 2012 at 15:50



rynop

53.4k ● 27 ● 105 ● 117

---

10 I've set up [lacewing-project.org](http://lacewing-project.org) with just an A record pointing to 207.97.227.245 (no CNAME records), and it works fine. I also put a file called CNAME in the root of the gh-pages branch containing the domain name. — James M Jan 31, 2012 at 15:56 ✎

---

In my case, just placing the CNAME file in gh-pages branch did the trick, thank you! – [jarandaf](#) Apr 26, 2016 at 14:47

9 Answers

Sorted by:

Highest score (default)



**1/23/19 UPDATE:**

**664**

Things have changed quite a bit (for the better) since my last answer. This updated answer will show you how to configure:



1. Root apex (example.com)
2. Sub-domain ([www.example.com](http://www.example.com))
3. HTTPS (optional but strongly encouraged)

+50



In the end, all requests to `example.com` will be re-directed to <https://www.example.com> (or `http://` if you choose NOT to use HTTPS). I always use `www` as my final landing. Why([1](#),[2](#)), is for another discussion.

This answer is long but it is **not** complicated. I was verbose for clarity as the GitHub docs on this topic are not clear or linear.

## Step 1: Enable GitHub pages in GitHub settings

1. From your repo, click on the  Settings tab

2. Scroll down to the **GitHub Pages** section. You have two options:

## GitHub Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository. [Learn more.](#)

3. Choosing **master branch** will treat **/README.md** as your web **index.html**. Choosing **master branch /docs folder** will treat **/docs/README.md** as your web **index.html**.

4. Choose a theme.

5. Wait a minute while GitHub publishes your site.

Verify it works by clicking on the link next to **Your site is ready to be published at**

## Step 2: Specify custom domain in GitHub settings

Enter your custom domain name here and hit **save** :

## GitHub Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

✓ Your site is published at <https://rynop.github.io/aws-blueprint/>

**Source**  
Your GitHub Pages site is currently being built from the /docs folder in the master branch. [Learn more.](#)  

master branch /docs folder ▼ **Save**

**Theme Chooser**  
Select a theme to publish your site with a Jekyll theme. [Learn more.](#)  
Your site is currently using the Cayman theme.  

**Change theme**

**Custom domain**  
Custom domains allow you to serve your site from a domain other than rynop.github.io. [Learn more.](#)  

**Save**

This is a subtle, but important step.

- If the custom domain you added to your GitHub Pages site is `example.com`, then `www.example.com` will redirect to `example.com`
- If the custom domain you added to your GitHub Pages site is `www.example.com`, then `example.com` will redirect to `www.example.com`.

As mentioned before, I recommend always landing at `www` so I entered `www.example.com` as pictured above.

## Step 3: Create DNS entries

In your DNS provider's web console, create four **A** records and one **CNAME**.

1. **A** Records for **@** (aka root apex):

Some DNS providers will have you specify **@**, others (like AWS Route 53) you will [leave the sub-domain blank](#) to indicate **@**. In either case, [these](#) are the **A** records to create:

```
185.199.108.153
185.199.109.153
185.199.110.153
185.199.111.153
```

[Check the Github docs](#) to confirm these are the most up-to-date IPs.

2. Create a **CNAME** record to point [www.example.com](http://www.example.com) to `YOUR-GITHUB-USERNAME.github.io`.

This is the most confusing part.

Note the `YOUR-GITHUB-USERNAME` **NOT** the GitHub repo name! The value of `YOUR-GITHUB-USERNAME` is determined by [this chart](#).

For a **User pages** site (most likely what you are), **CNAME** entry will be `username.github.io`, ex:

**Name:**  .example.com.

**Type:**

**Alias:** ☐ Yes ☒ No

**TTL (Seconds):**

**Value:**

For a **Organization pages** site, **CNAME** entry will be `orgname.github.io`, ex:

**Name:**  .example.com.

**Type:**

**Alias:** ☐ Yes ☒ No

**TTL (Seconds):**

**Value:**

## Step 5: Confirm DNS entries

1. Confirm your **A** records by running `dig +noall +answer example.com`. It should return the four `185.x.x.x` IP addresses you entered.
2. Confirm your **CNAME** record by running `dig www.example.com +nostats +nocmd`. It should return a **CNAME** `YOUR-GITHUB-USERNAME.github.io`

It may take an hour or so for these DNS entries to resolve/propagate. Once they do, open up your browser to `http://example.com` and it should re-direct to `http://www.example.com`

## Step 6: SSL (HTTPS) Configuration. Optional, but

# highly recommended

After you have the custom domain working, go back to the repo settings. If you already have the settings page open, hard refresh the page.

If there is a message under the **Enforce HTTPS** checkbox, stating that it is still processing you will need to wait. You may also need to hit the **save** button in the **Custom domain** section to kick off the **Enforce HTTPS** processing.

Once processing is completed, it should look like this:

## GitHub Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

✓ Your site is published at <http://www.clipguard.org/>

### Source

Your GitHub Pages site is currently being built from the master branch. [Learn more.](#)

master branch ▾

Save

### Theme Chooser

Select a theme to publish your site with a Jekyll theme. [Learn more.](#)

Your site is currently using the Cayman theme.

Change theme

### Custom domain

Custom domains allow you to serve your site from a domain other than `rynop.github.io`. [Learn more.](#)

Save

☐ **Enforce HTTPS**

HTTPS provides a layer of encryption that prevents others from snooping on or tampering with traffic to your site. When HTTPS is enforced, your site will only be served over HTTPS. [Learn more.](#)

Just click on the **Enforce HTTPS** checkbox, and point your browser to `https://example.com`. It should re-direct and



open `https://www.example.com`

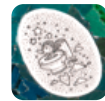
## THATS IT!

GitHub will automatically keep your HTTPS cert up-to-date AND should handle the apex to `www` redirect over HTTPS.

Share Improve this answer

Follow

edited Dec 22, 2022 at 10:36



starball

48.3k ● 28 ● 187 ● 848

answered Feb 3, 2012 at 5:01



rynop

53.4k ● 27 ● 105 ● 117

---

14 As of Apr 19, 2012, [GitHub's documentation](#) uses `204.232.175.78` as the IP address. – [Aseem Kishore](#) Apr 20, 2012 at 2:24

---

31 Checked feb 6th 2014 and github docs say: Create DNS A records which point to the following IP addresses: 192.30.252.153, 192.30.252.154. – [Danny](#) Feb 6, 2014 at 23:08 ✎

---

2 "If you are using an A record that points to 207.97.227.245 or 204.232.175.78, you will need to update your DNS settings, as we no longer serve Pages directly from those servers." [help.github.com/articles/my-custom-domain-isn-t-working](https://help.github.com/articles/my-custom-domain-isn-t-working) - 3/8/14 – [eddywashere](#) Mar 9, 2014 at 6:15

---

11 I don't think it's possible to set up a CNAME record from the naked @ domain. That's why you need the A records, to handle the redirect from the naked domain to the www

subdomain. It's a problem with any cloudhost. No guarantee of a fixed IP. – [superluminary](#) Mar 12, 2014 at 22:19

---

2 @rynop GoDaddy doesn't allow @ and www to yourusername.github.io, only one CNAME record per subdomain :( – [Tejas Manohar](#) Jul 31, 2014 at 5:20

---



## Overview

259



The [documentation](#) is a little confusing when it comes to **project pages**, as opposed to **user pages**. It feels like you should have to do more, but actually the process is very easy.



It involves:



1. Setting up 2 static A records for the naked (no www) domain.
2. Creating one CNAME record for www which will point to a GitHub URL. This will handle www redirection for you.
3. Creating a file called CNAME (capitalised) in your project root on the gh-pages branch. This will tell Github what URL to respond to.
4. Wait for everything to propagate.

## What you will get

Your content will be served from a URL of the form <http://nicholasjohnson.com>.

Visiting <http://www.nicholasjohnson.com> will return a 301 redirect to the naked domain.

The path will be respected by the redirect, so traffic to <http://www.nicholasjohnson.com/angular> will be redirected to <http://nicholasjohnson.com/angular>.

You can have one project page per repository, so if your repos are open you can have as many as you like.

Here's the process:

## 1. Create A records

For the A records, point @ to the following ip addresses:

```
@: 185.199.108.153
@: 185.199.109.153
@: 185.199.110.153
@: 185.199.111.153
```

These are the static Github IP addresses from which your content will be served.

## 2. Create a CNAME Record

For the CNAME record, point www to yourusername.github.io. Note the trailing full stop. Note also, **this is the username, not the project name**. You don't need to specify the project name yet. Github will use

the CNAME **file** to determine which project to serve content from.

e.g.

```
www: forwardadvance.github.io.
```

The purpose of the CNAME is to redirect all www subdomain traffic to a GitHub page which will 301 redirect to the naked domain.

Here's a screenshot of the configuration I use for my own site <http://nicholasjohnson.com>:

@	3600	A	192.30.252.153
@	3600	A	192.30.252.154
www	3600	CNAME	forwardadvance.github.io.

### 3. Create a CNAME file

Add a file called CNAME to your project root in the gh-pages branch. This should contain the domain you want to serve. Make sure you commit and push.

e.g.

```
nicholasjohnson.com
```

This file tells GitHub to use this repo to handle traffic to this domain.

## 4. Wait

Now wait 5 minutes, your project page should now be live.

Share Improve this answer

Follow

edited Dec 21, 2018 at 3:31



cremaster

50 ● 7

answered Mar 13, 2014 at 9:43



superluminary

49k ● 26 ● 153 ● 150

---

1 No, the apex domain has a static IP. The www subdomain has the CNAME. – [superluminary](#) Mar 27, 2014 at 21:23

---

4 @superluminary, i repeated your process, and i have two issues, 1) I'm using freeDNS afraid.org for DNS sharing, which does not allow to add trailing . in username.github.io.. 2) when i added github.io url without trailing ., mydomain.com is working, but i cannot acces with www.mydomain.com ?? – [bistaumanga](#) Apr 22, 2014 at 10:04 ✎

---

4 The part that I couldn't figure out via GitHub docs was what URL to use for the project. Thanks for this answer: "this is the username, not the project name." so use `yourUserName.github.io.` – [pkamb](#) Oct 1, 2015 at 20:30

---

1 I have updated this solution and really recommend doing it. You can get it to work in other ways, for example using a CNAME for the @ host, but it will break e-mail, etc. This is the best solution I have found. Thank you. – [Steven L.](#) Dec 11, 2015 at 15:36

---

1 Thank you! I was stuck on the correct value for the CNAME record in DNS - just organization.github.io, no matter what project under that organization you're trying to serve up. (i.e.,



## Short answer

29



These detailed explanations are great, but the OP's (and my) confusion could be resolved with one sentence:

**"Direct DNS to your GitHub *username* or *organization*, ignoring the specific project, and add the appropriate CNAME *files* in your project repositories: GitHub will send the right DNS to the right project based on *files* in the *repository*."**

Share Improve this answer

Follow

edited Mar 18, 2022 at 11:37



[blacktide](#)

12k ● 8 ● 40 ● 58

answered Dec 15, 2016 at 14:28



[Jim Pivarski](#)

5,944 ● 3 ● 36 ● 51

---

3 I have no idea what "sending DNS" means. – [ceving](#) May 16, 2017 at 8:02

---

Redirect DNS queries, so that HTTP requests with particular domain names get the right documents as responses.

– [Jim Pivarski](#) May 16, 2017 at 12:08

---



If you are wondering how to get your domain to appear as `www.mydomain.com` instead of

21

redirecting the `www` request to `mydomain.com`,  
try this:



**CNAME file on gh-pages** branch will have one line:

`www.mydomain.com` (instead of `mydomain.com`)

No matter your preference on redirection (in other words, no matter what is in your CNAME file on the gh-pages branch), **with your DNS provider**, you should set it up like this:

```
A      @      192.30.252.154
A      @      192.30.252.153
CNAME  www    username.github.io
```

Share Improve this answer

Follow

edited Apr 29, 2017 at 9:28



**Simon Wicki**

4,049 ● 5 ● 23 ● 25

answered Jan 24, 2016 at 21:52



**Phil**

2,790 ● 20 ● 20

---

@zwacky why can't I just point the CNAME at @? why should "www.mydomain.com" and "mydomain.com" resolve to potentially different IPs? – [aaa90210](#) Jun 28, 2017 at 5:53



---

so im guessing what you type in the CNAME file should match whats in the custom domain box on github settings page? aka either example.com or www.example.com depending which redirect you want? Either way CNAME on

DNS settings is the same, its www -> username.github.io  
– [MadeByDouglas](#) Mar 15 at 5:24

---

---





15



As of Aug 29, 2013, [Github's documentation](#) claim that:

Warning: Project pages subpaths like <http://username.github.io/projectname> will not be redirected to a project's custom domain.

Share Improve this answer

answered Aug 29, 2013 at 9:03

Follow



Amio

185 ● 1 ● 5

---

3 I'm confused why they added this behavior. I want my .com to point to github pages, not the other way around.  
– [Jordan Scales](#) Oct 4, 2013 at 16:35

---

9 I was really confused by this too. As I now understand it, you can still point yoursite.com to github pages, but yourusername.github.io/yoursite won't redirect to yoursite.com. That's fine because yoursite.com will still work, just make sure to update your dns records and wait for that to resolve. [help.github.com/articles/...](http://help.github.com/articles/...) – [eddywashere](#) Mar 9, 2014 at 6:21

---

This should have SEO (Search Engine Optimization) ramifications for the website. Google is said to demote the results ranking of content that is accessible from multiple domains, which is one of several reasons for applying 301 ("permanent") redirects from `username.github.io/projectname` to `www.projectname.com` . – [BorromeanNot](#) Nov 28, 2016 at 16:47

---



3



I just discovered, after a bit of frustration, that if you're using [PairNIC](#), all you have to do is enable the "Web Forwarding" setting under "Custom DNS" and supply the `username.github.io/project` address and it will automatically set up both the apex and subdomain records for you. It appears to do exactly what's suggested in the accepted answer. However, it won't let you do the exact same thing by manually adding records. Very strange. Anyway, it took me a while to figure that out, so I thought I'd share to save everyone else the trouble.

Share Improve this answer

answered Oct 21, 2014 at 23:34

Follow



[Brandon Fosdick](#)

16.1k ● 4 ● 23 ● 20



3



Things are lot easier nowadays!

1. Update your Apex domain (@) record to point

192.30.252.154

192.30.252.153

2. Edit your **Custom domain** field in your github repo settings.

## GitHub Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

✓ Your site is published at <http://khedmate-nsan.org/>

### Source

Your GitHub Pages site is currently being built from the master branch. [Learn more.](#)

master branch ▾

Save

### Theme Chooser

Select a theme to publish your site with a Jekyll theme. [Learn more.](#)

Choose a theme

### Custom domain

Custom domains allow you to serve your site from a domain other than [khedmate-nsan.org](#). [Learn more.](#)

[khedmate-nsan.org](#)

Save

☐ **Enforce HTTPS** — Unavailable for your site because your domain is not properly configured to support HTTPS ([khedmate-nsan.org](#))

HTTPS provides a layer of encryption that prevents others from snooping on or tampering with traffic to your site. When HTTPS is enforced, your site will only be served over HTTPS. [Learn more.](#)

3. www and other subdomains can be updated as CNAME to apex domain.

Share Improve this answer

edited May 14, 2018 at 6:30

Follow

answered May 14, 2018 at 6:24

Rafaf Tahsin

8,496 ● 4 ● 30 ● 45

The selected answer is the good one, but is long, so **you might not read the key point:**

3

I got an error with the SSL when accesign [www.example.com](http://www.example.com) but it worked fine if I go to example.com



If it happens the same to you, probably your error is that in the DNS configuration you have set:

```
CNAME www.example.com --> example.com (WRONG)
```

But, what you have to do is:

```
CNAME www.example.com --> username.github.io  
(GOOD)
```

or

```
CNAME www.example.com --> organization.github.io  
(GOOD)
```

That was my error

Share Improve this answer

answered Jan 28, 2021 at 12:52

Follow



Rafa Nogales

664 ● 9 ● 13



0



I'd like to share [my steps](#) which is a bit different to what offered by [rynop](#) and [superluminary](#).

- for **A** Record is exactly the same but
- instead of creating **CNAME** for **www** I would prefer to redirect it to my blank domain ( **non-www** )



This configuration is referring to guidance of [preferred domain](#). The domain setting of **www** to **non www** or vise

versa can be different on each of the domain providers. Since my domain is under GoDaddy, so under the *Domain Setting* I set it using the [Subdomain Forwarding](#) (301).

As the result of pointing the domain to Github repository, it will then give all the URLs for both of `master` and `gh-pages`.

As addition to the `CNAME` file above, you may need to completely [bypass Jekyll processing](#) on GitHub Pages by creating a file named `.nojekyll` in the root of your pages repo.

Share Improve this answer

edited Jun 30, 2021 at 21:56

Follow

answered Aug 14, 2015 at 10:20



eQ19

10.6k ● 3 ● 71 ● 80

---

4 I would recommend **not** to use godaddy, but just personal opinion – [MMachinegun](#) Mar 4, 2016 at 18:19 ✎

---

You have no CNAME in the master branch. – [Rubel Biswas](#) Jun 29, 2021 at 18:51

---

@Rubel First: I have pointed out by "instead of creating CNAME." Second: The domain is no more exist so I have removed the live sample.. Thanks to have reminded me.

– [eQ19](#) Jun 30, 2021 at 22:02 ✎

---



**Highly active question.** Earn 10 reputation (not counting the [association bonus](#)) in order to answer this question. The reputation requirement helps protect this question from spam and non-answer activity.