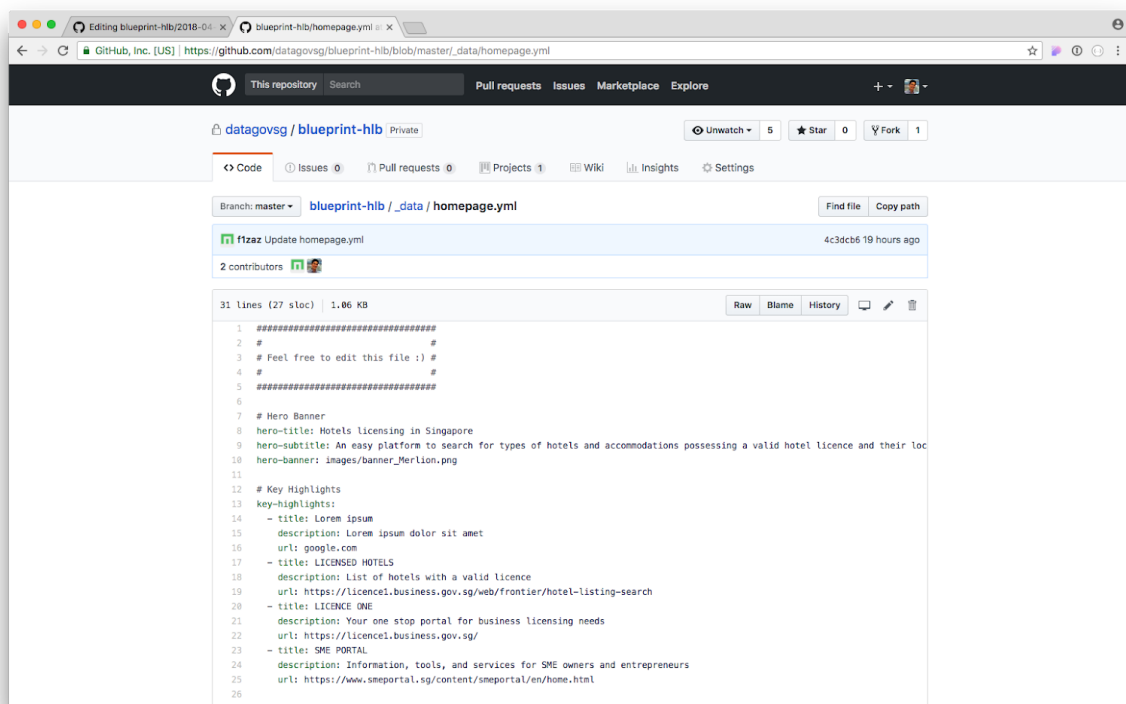


Task 1: Update the Homepage

There are two sub-parts to this task: a) changing the hero banner image, and b) editing the key highlights in the hero banner.

The Homepage is controlled by a single YAML (.yaml) file. The file is in the `_data` folder and is called `homepage.yaml`.

The YAML file tells the website generator what information to display on the homepage. It looks something like this.

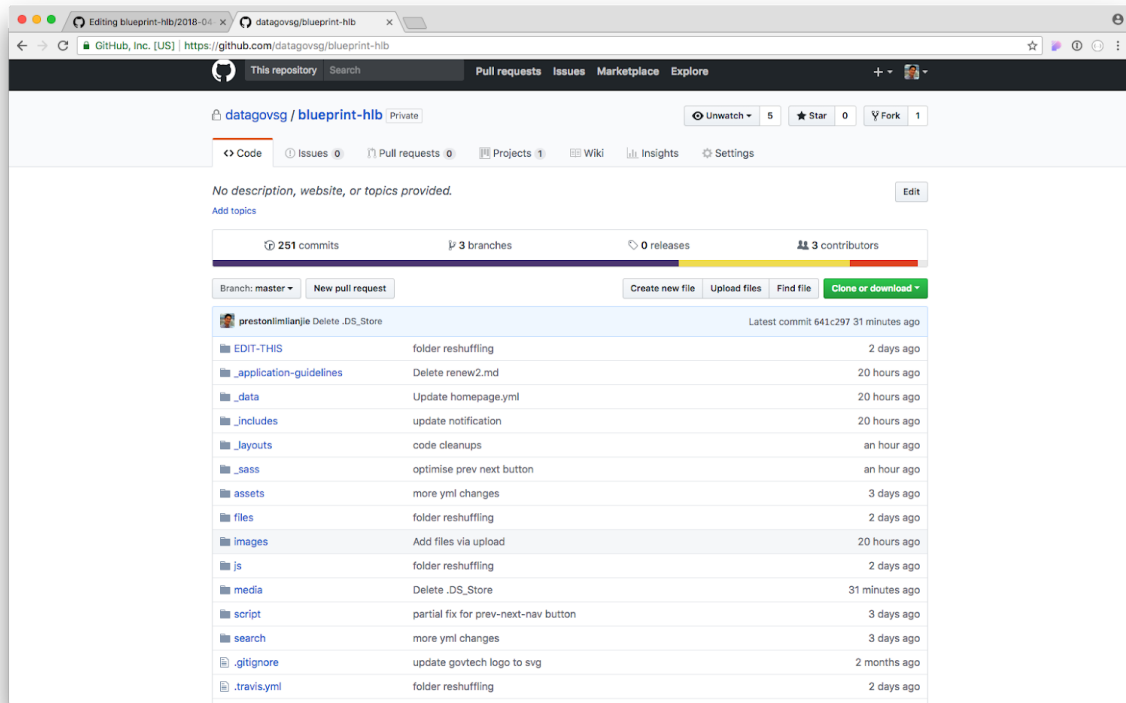


The screenshot shows a web browser displaying a GitHub repository page for 'datagovsg/blueprint-hlb'. The file 'blueprint-hlb / _data / homepage.yaml' is selected. The file content is a YAML configuration for a homepage, including sections for Hero Banner, Key Highlights, and Media Section. The Hero Banner section includes fields for hero-title, hero-subtitle, and hero-banner. The Key Highlights section includes a list of highlights with fields for title, description, and url. The Media Section is partially visible at the bottom.

```
1 #####
2 #
3 # Feel free to edit this file :) #
4 #
5 #####
6
7 # Hero Banner
8 hero-title: Hotels licensing in Singapore
9 hero-subtitle: An easy platform to search for types of hotels and accommodations possessing a valid hotel licence and their loc
10 hero-banner: images/banner_Merlion.png
11
12 # Key Highlights
13 key-highlights:
14   - title: Lorem ipsum
15     description: Lorem ipsum dolor sit amet
16     url: google.com
17   - title: LICENSED HOTELS
18     description: List of hotels with a valid licence
19     url: https://licencel.business.gov.sg/web/frontier/hotel-listing-search
20   - title: LICENCE ONE
21     description: Your one stop portal for business licensing needs
22     url: https://licencel.business.gov.sg/
23   - title: SME PORTAL
24     description: Information, tools, and services for SME owners and entrepreneurs
25     url: https://www.smeportal.sg/content/smeportal/en/home.html
26
27 # Media Section
```

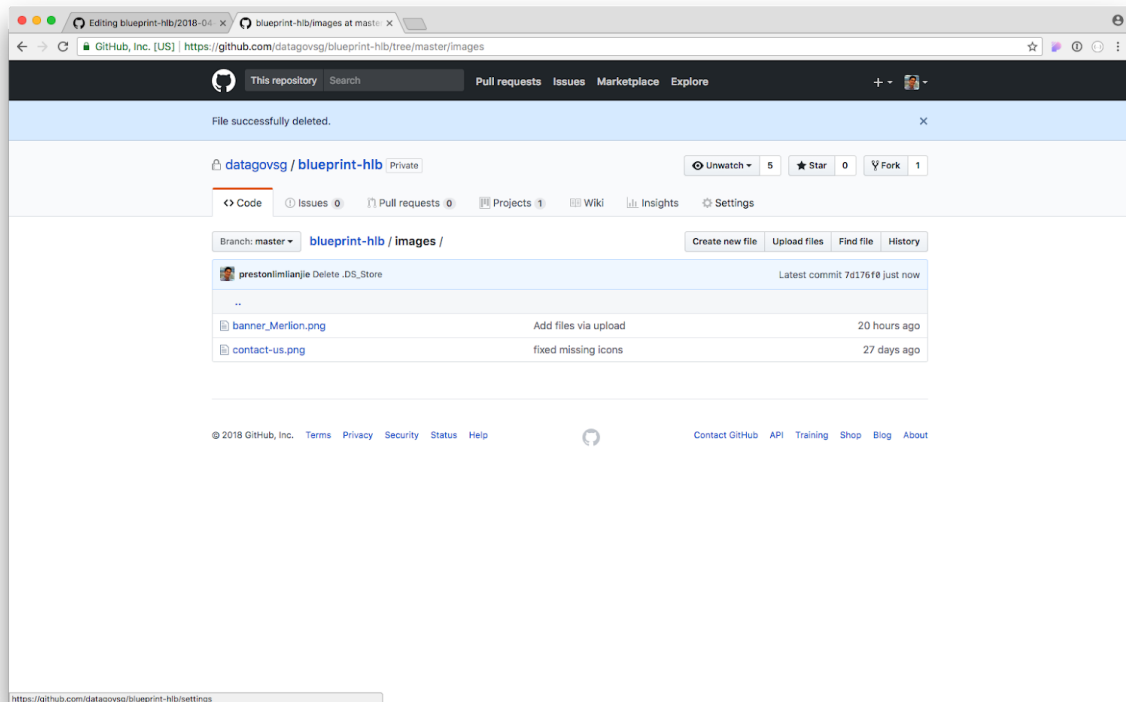
Task 1a: Change the Hero Banner Image

Step 1: Click on the images folder.



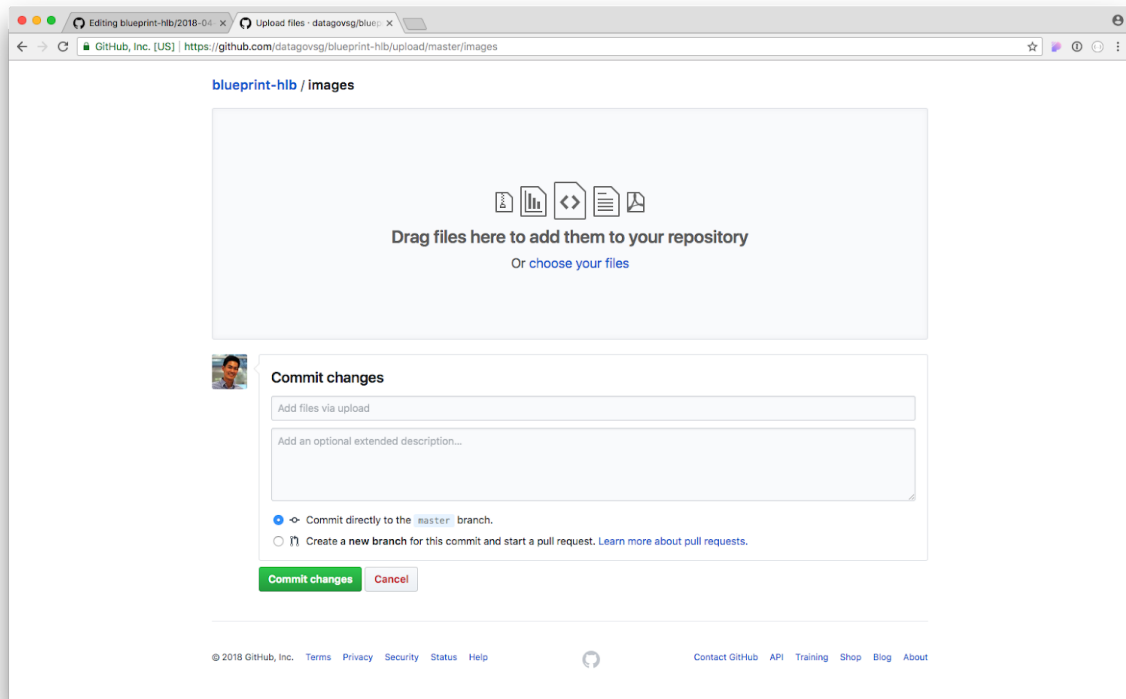
Main directory of the website repository

Step 2: Click on the Upload files button.



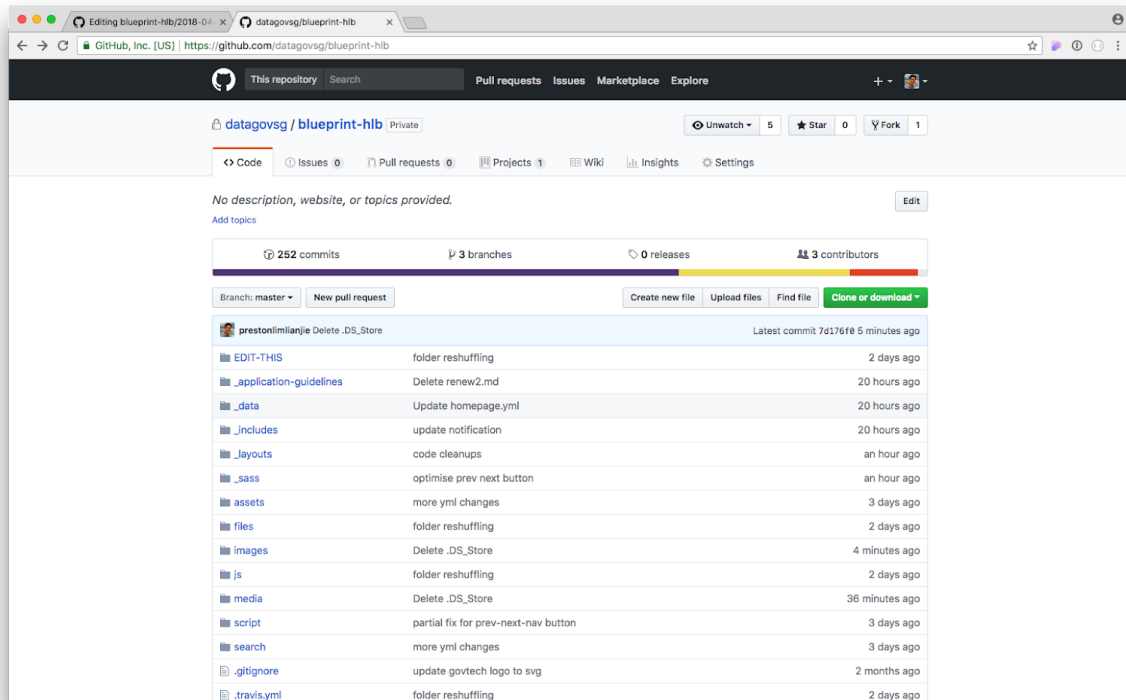
The images folder contains all the images used in the website.

Step 3: Upload the banner_GardenBay.png. Click on the Commit changes button to save the file.



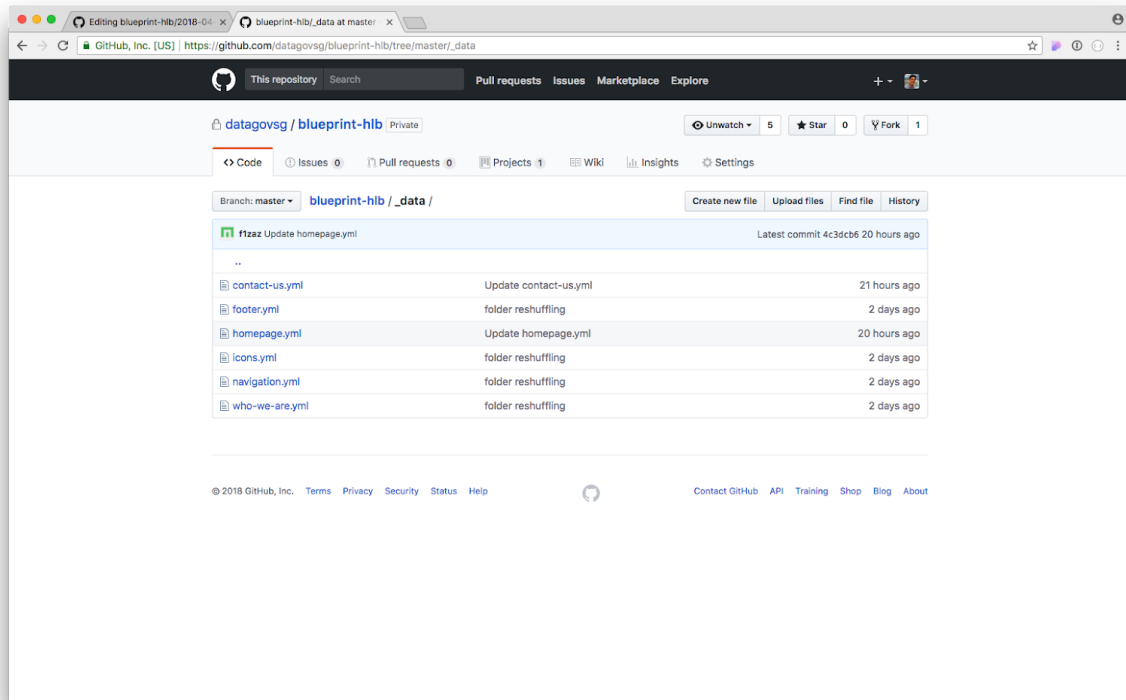
Upload and save the image file.

Step 4: Go back to the main directory of the repository. Click on the `_data` folder.



Go to the `_data` folder

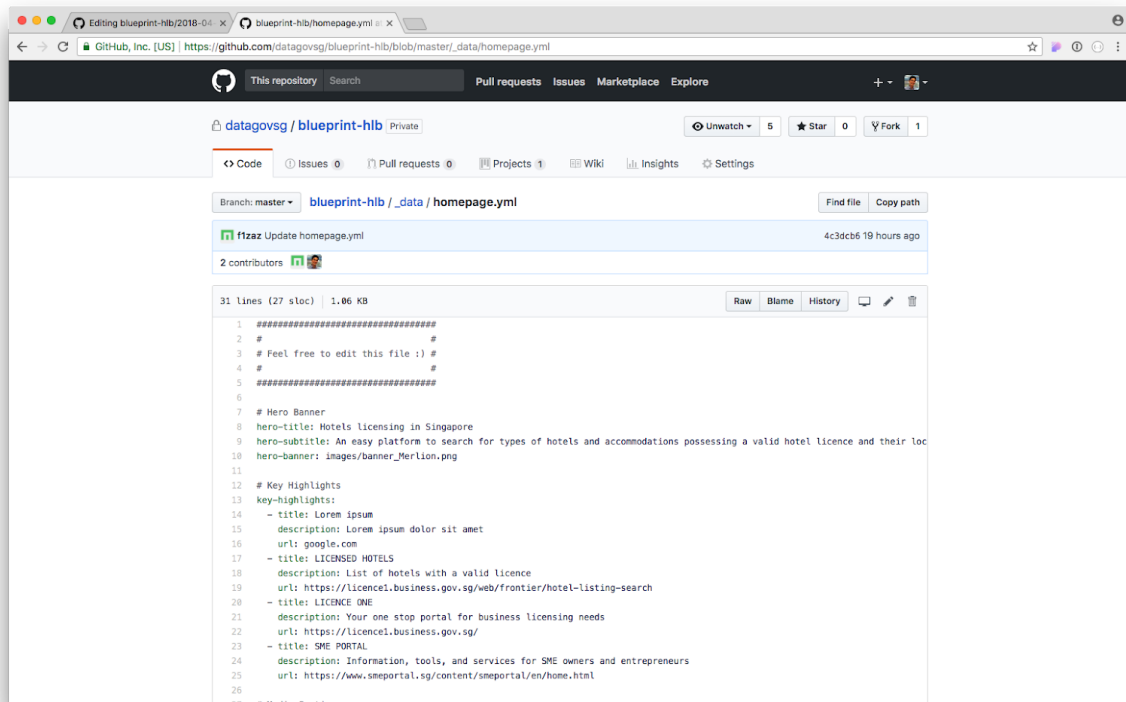
Step 5: Click on the homepage.yml file.



Click on homepage.yml

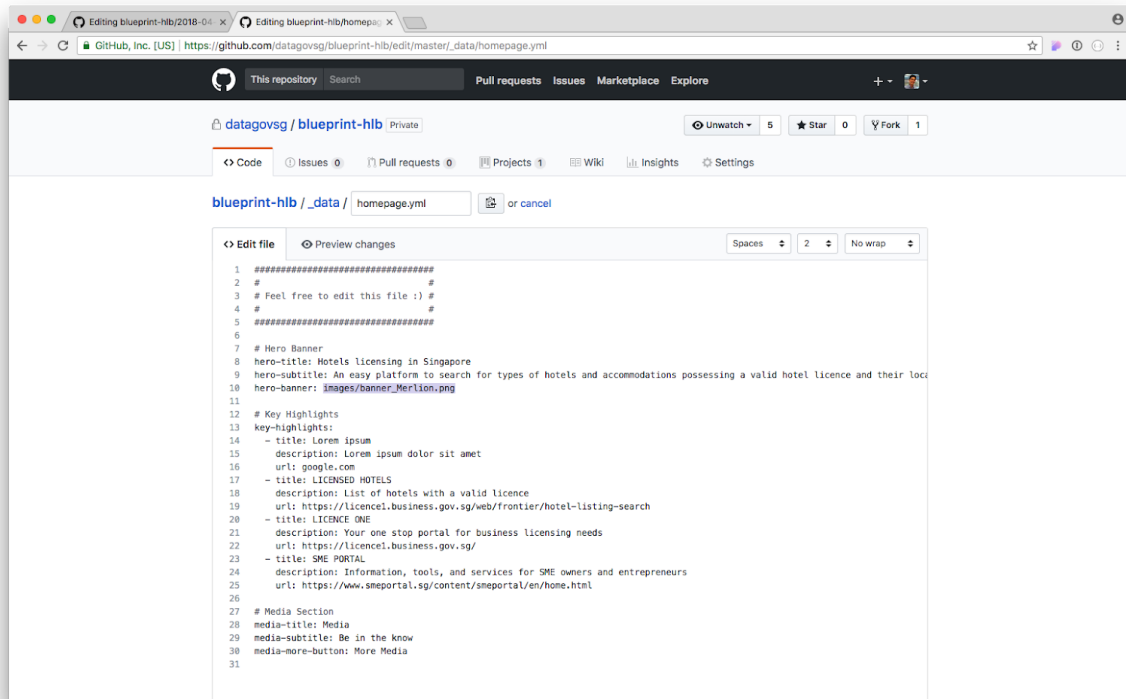
Step 6: We want to change the hero-banner image URL to the one that we just uploaded. Hero-banner tells the website generator where to find the hero banner image.

Click on the edit (pencil icon) button.



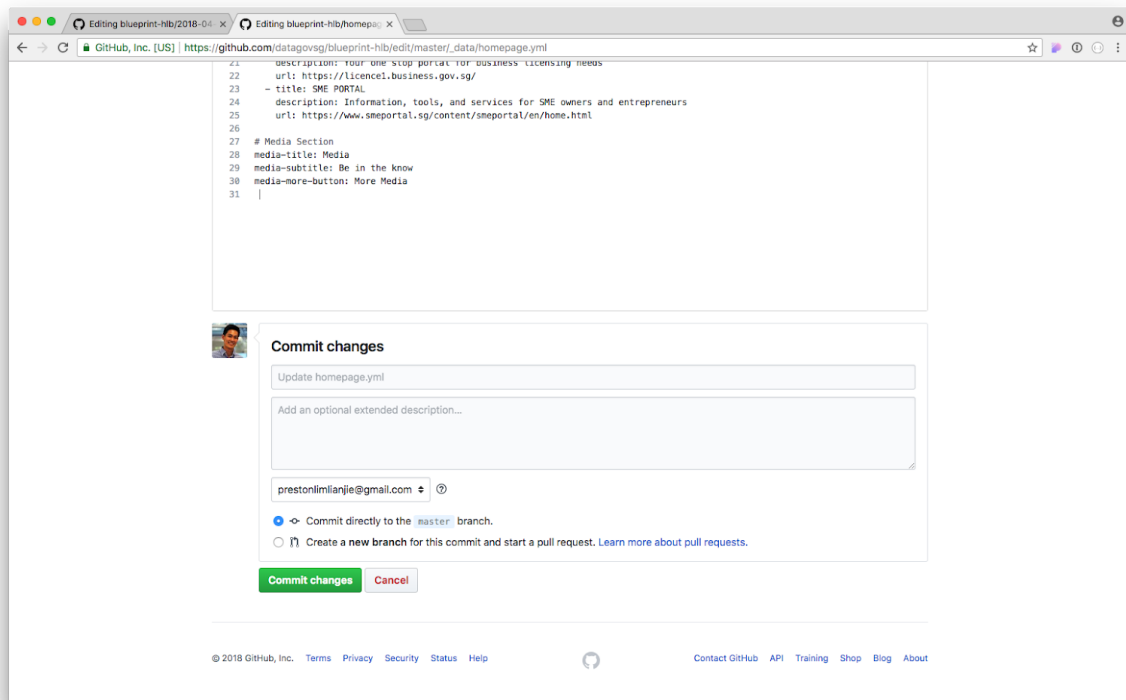
Click on homepage.yml

Step 7: Replace the hero-banner URL with images/banner_GardenBay.png.



Replace the hero-banner URL

Step 8: Click on the Commit changes button to save the file. The image change should be reflected on the website in a minute or so.



And we're done!

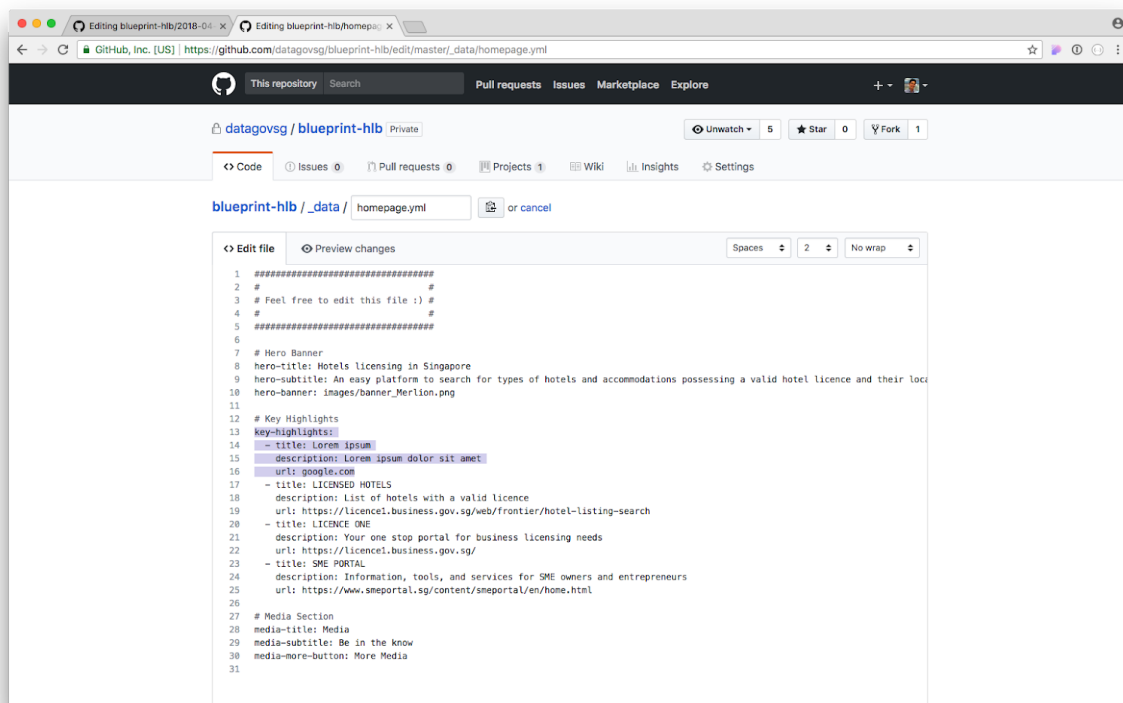
Task 1b: Edit a Key Highlight

Step 1: Go back to editing the `_data/homepage.yml` file. Replace the first key highlight section with:

```
title: TEMPLATES AND FORMS
description: Downloadable templates and forms
url: /media/forms-&-templates/
```

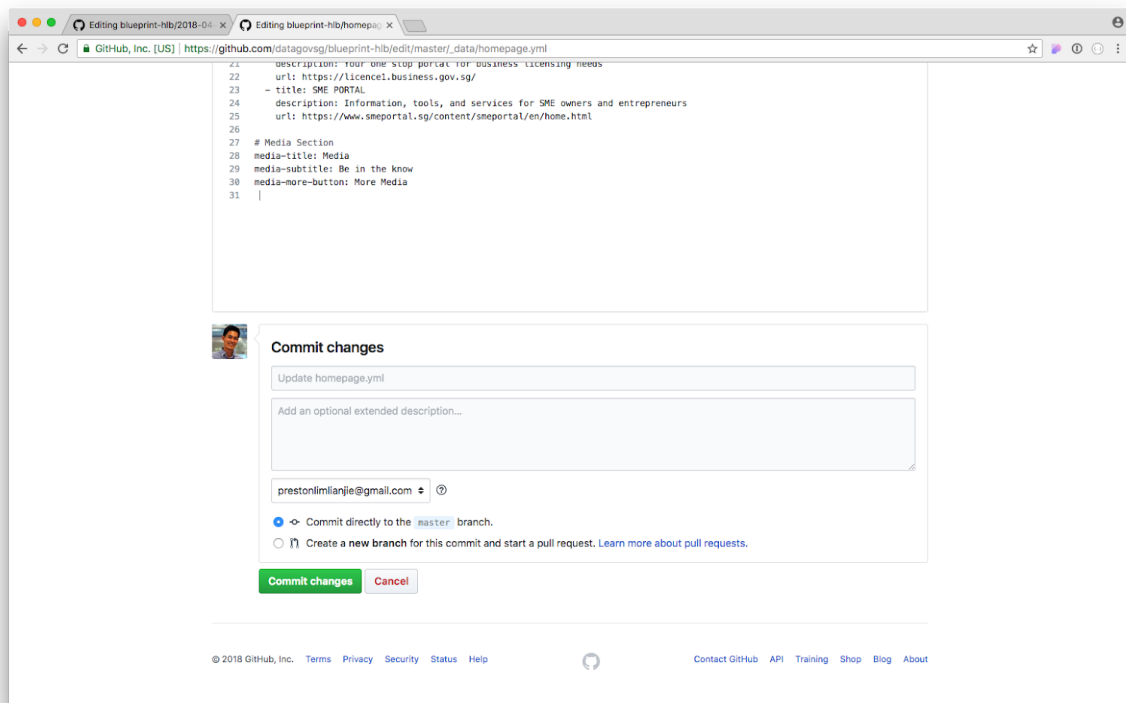
The title is the title of the key highlight. It is case insensitive.
The description is the description of the highlight. It is also case insensitive.

The url is the URL that the user will be re-directed to when they click on the key highlight button. For *external* links, we need to use the absolute URL (e.g. if you wanted to redirect your user to google.com, you need to use `https://google.com`)



Replace the key highlight section

Step 2: Scroll down and click on the Commit changes button to save the file. The change should be reflected on the website in a minute or so.

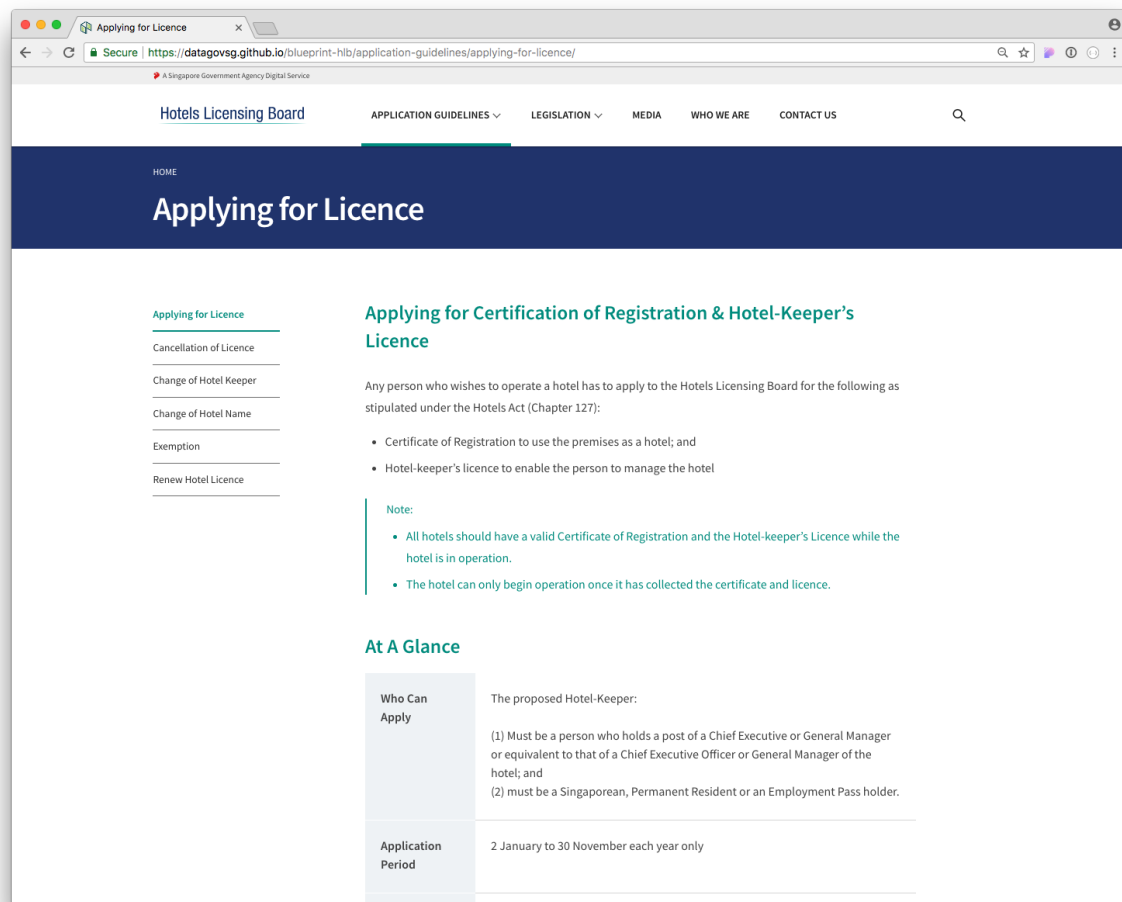


And we're done!

Application Guideline Files

Short detour: Understanding Application Guideline Markdown Files

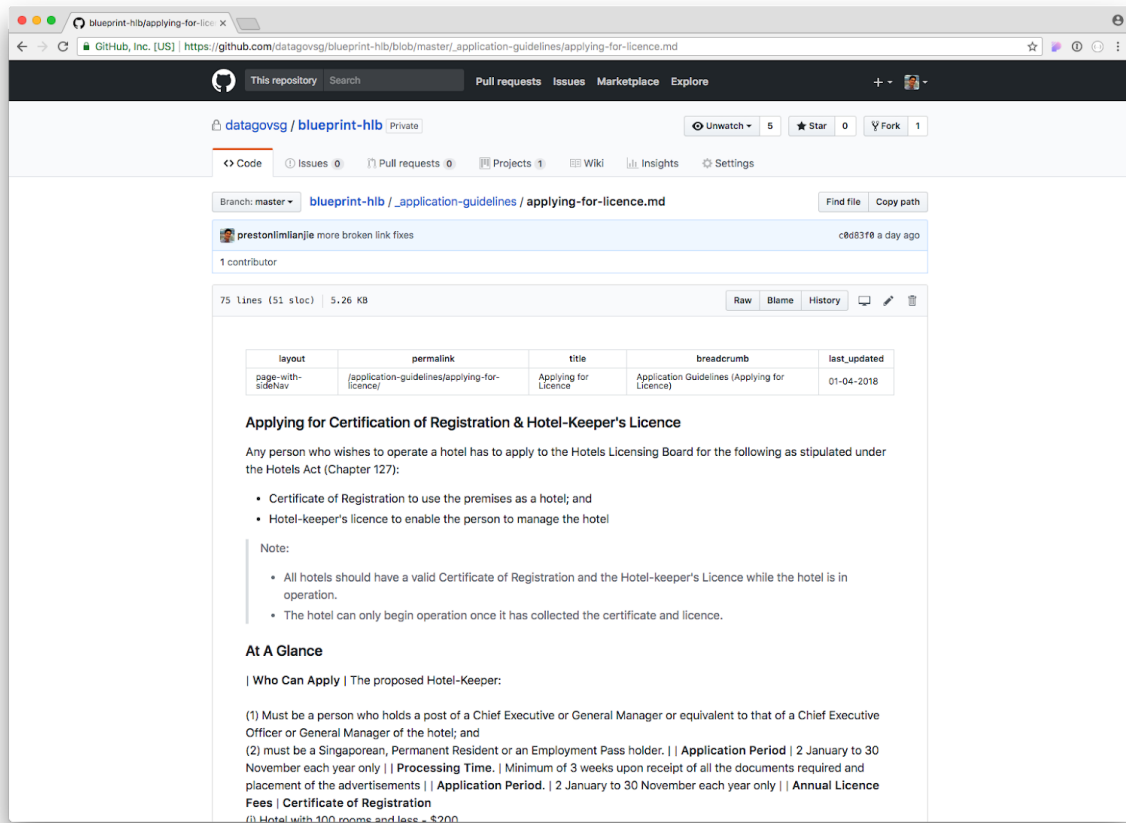
This section explains what a markdown file is, and specifically what has to go into an Application Guideline markdown file. Please skip this if you are familiar with both of these concepts.



Sample application guideline on the Blueprint website

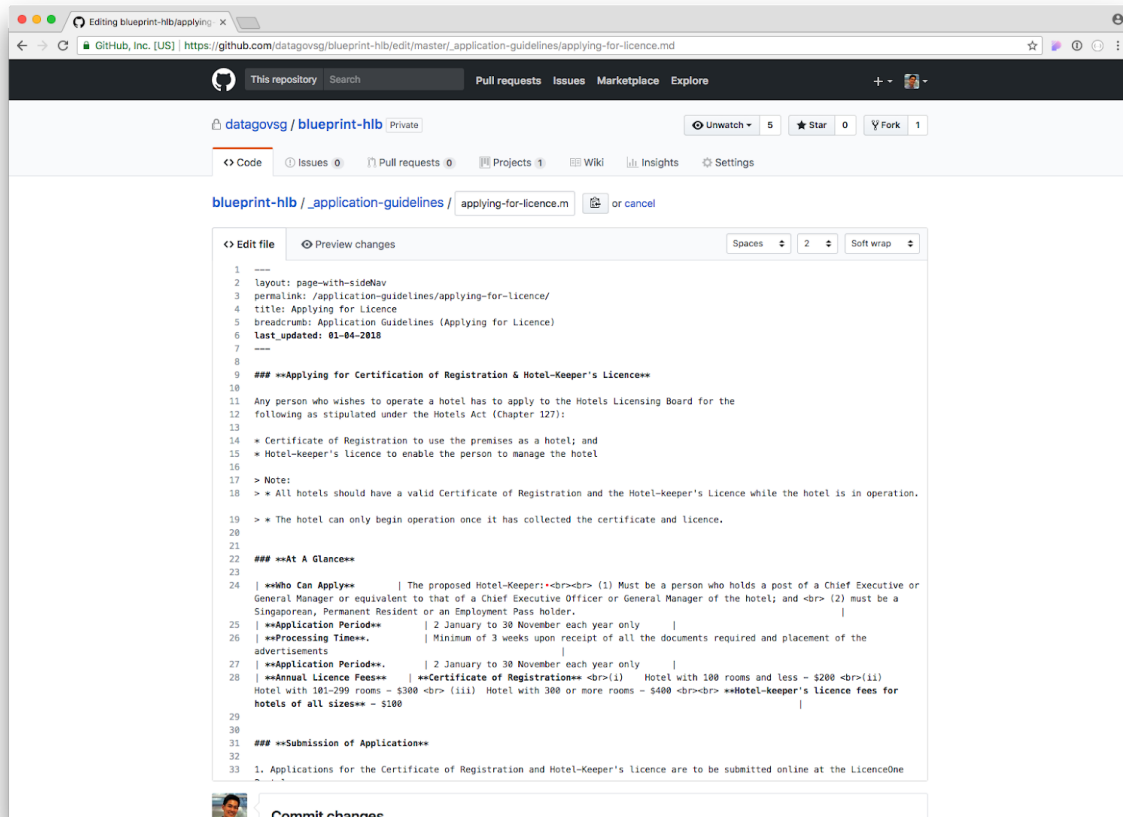
Before we create a new application guideline similar to the one above, let us take a look at how an application guideline markdown file looks like. Let's click on the `applying-for-licence.md` file.

You should see a preview screen similar to the one shown below.



Application guideline preview screen

Click on the edit icon (the pencil icon) at the right of the page (next to Raw, Blame, and History). You should see a raw markdown file similar to that shown below.



The screenshot shows a GitHub web interface for the repository 'datagovsg/blueprint-hlb'. The file being viewed is 'applying-for-licence.md' in the '_application-guidelines/' directory. The file content is a raw markdown document with the following structure:

```
1 ---
2 layout: page-with-sidenav
3 permalink: /application-guidelines/applying-for-licence/
4 title: Applying for Licence
5 breadcrumb: Application Guidelines (Applying for Licence)
6 last_updated: 01-04-2018
7 ---
8
9 ### **Applying for Certification of Registration & Hotel-Keeper's Licence**
10
11 Any person who wishes to operate a hotel has to apply to the Hotels Licensing Board for the
12 following as stipulated under the Hotels Act (Chapter 127):
13
14 * Certificate of Registration to use the premises as a hotel; and
15 * Hotel-keeper's licence to enable the person to manage the hotel
16
17 > Note:
18 > * All hotels should have a valid Certificate of Registration and the Hotel-keeper's Licence while the hotel is in operation.
19 > * The hotel can only begin operation once it has collected the certificate and licence.
20
21
22 ### **At A Glance**
23
24 | **Who Can Apply** | The proposed Hotel-Keeper: <br> (1) Must be a person who holds a post of a Chief Executive or
25 | | General Manager or equivalent to that of a Chief Executive Officer or General Manager of the hotel; and <br> (2) must be a
26 | | Singaporean, Permanent Resident or an Employment Pass holder.
27 |
28 | **Application Period** | 2 January to 30 November each year only
29 |
30 | **Processing Time** | Minimum of 3 weeks upon receipt of all the documents required and placement of the
31 | | advertisements
32 |
33 | **Application Period** | 2 January to 30 November each year only
34 |
35 | **Annual Licence Fees** | **Certificate of Registration** <br> (i) Hotel with 100 rooms and less - $200 <br> (ii)
36 | | Hotel with 101-299 rooms - $300 <br> (iii) Hotel with 300 or more rooms - $400 <br> **Hotel-keeper's licence fees for
37 | | hotels of all sizes** - $100
38
39
40 ### **Submission of Application**
41
42 1. Applications for the Certificate of Registration and Hotel-Keeper's licence are to be submitted online at the LicenceOne
43 ~ ~ ~
```

At the bottom of the editor, there is a 'Commit changes' button.

Raw application guideline markdown file

There are two main components to every markdown file: (1) header and (2) content.

The header (also known as YAML front-matter) is the component enclosed between two triple dashes '---'. The header contains important metadata about the page; it tells the website generator what this page's title, URL, and updated date is.

```
---
layout: page-with-sideNav
permalink: /application-guidelines/applying-for-licence/
title: Applying for Licence
breadcrumb: Application Guidelines (Applying for Licence)
last_updated: 01-04-2018
---
```

Sample markdown header

The crucial components of the application guidelines header are:

- layout
 - o Always leave it as *page-with-sideNav*. What this does is tell the website generator to create a side navigation bar for users to navigate to other application guidelines.
 - *layout: page-with-sideNav*
- permalink
 - o This is the website URL. It should be as follows
 - *permalink: /application-guidelines/<page-title>/*
 - o The title of your page has to be in lower case and must not contain special characters (i.e. only alpha-numeric and no whitespace).
 - o For example, if I wanted the page to be *Applying for Licence*. My page title would be *applying-for-licence*, and my resulting permalink would be:
 - *permalink: /application-guidelines/applying-for-licence/*
- title
 - o This is the title of the page. It can contain special characters and is case-insensitive.
 - o For example, if I wanted the page to be *Applying for Licence*. My page title would be *Applying for Licence*.
 - *title: Applying for Licence*
- breadcrumb
 - o This is the breadcrumb to tell your users where they are on the website. It can contain special characters and is case-insensitive.
 - o For example, if I wanted the user to know that they were on the *Applying for Licence* page, my breadcrumb would be *Application Guidelines (Applying for Licence)*
 - *breadcrumb: Application Guidelines (Applying for Licence)*
- last-updated

- o This tells your users when this page was last updated. It follows the format *DD-MM-YYYY*.
 - *last_updated: 01-04-2018*

The content is the actual meaty bit of the page. It is written in the markdown language – a simple to read format. The purpose of the markdown language is to help you write website content without having to delve into icky HTML.

**Applying for Certification of Registration & Hotel-Keeper's Licence**

Any person who wishes to operate a hotel has to apply to the Hotels Licensing Board for the following as stipulated under the Hotels Act (Chapter 127):

- * Certificate of Registration to use the premises as a hotel; and
- * Hotel-keeper's licence to enable the person to manage the hotel

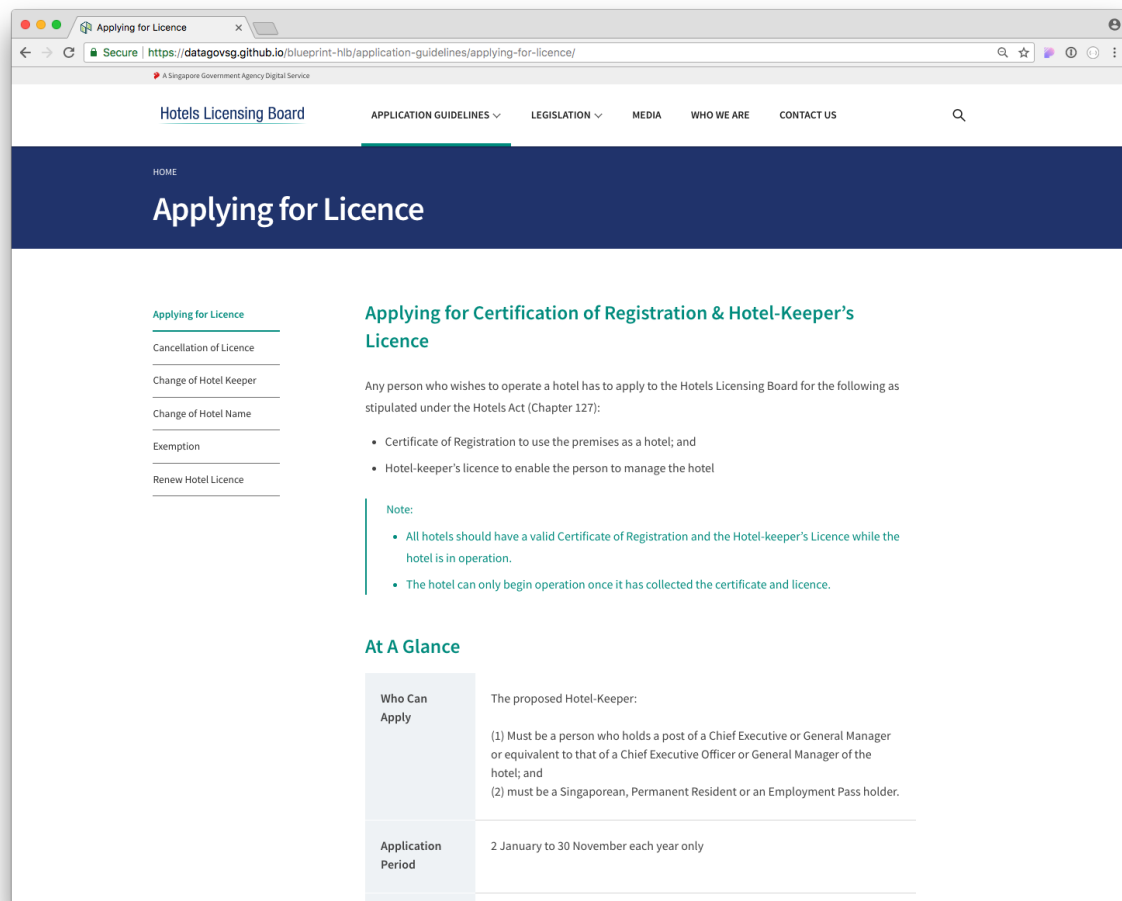
Sample markdown content

For more on markdown styling, please refer to our markdown styling guide.

Task 2: Create a new Application Guideline

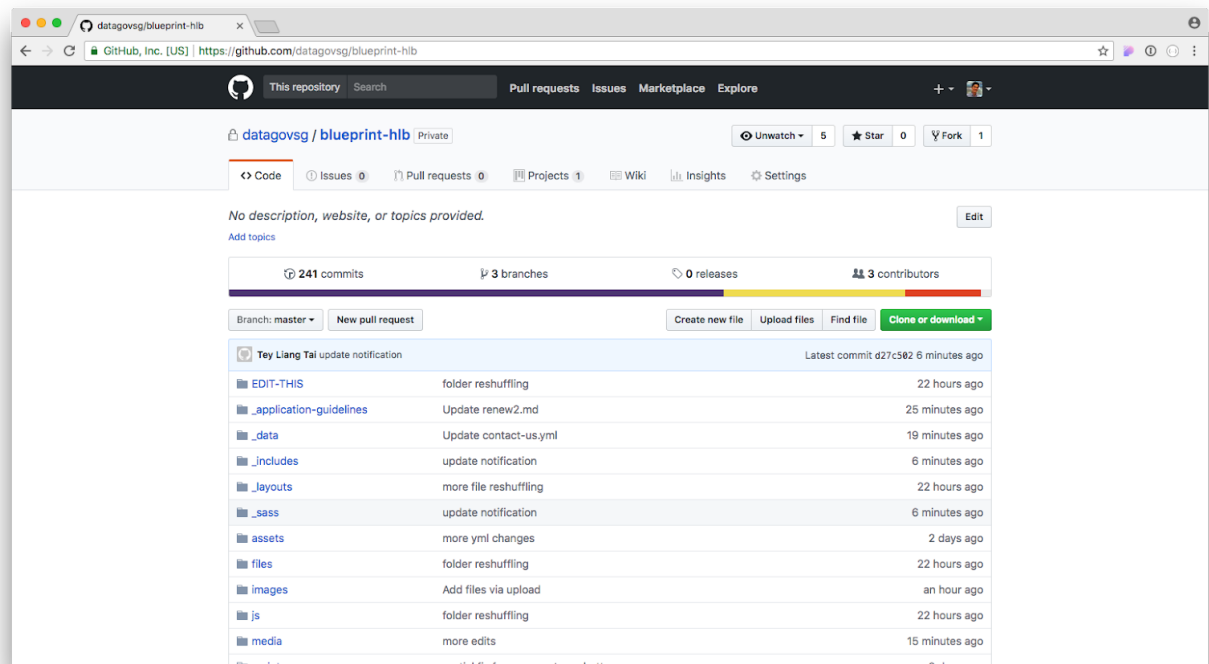
Application guideline files are markdown (.md) files that are found in the _application-guidelines folder.

In this task, we will create a new Application Guideline.



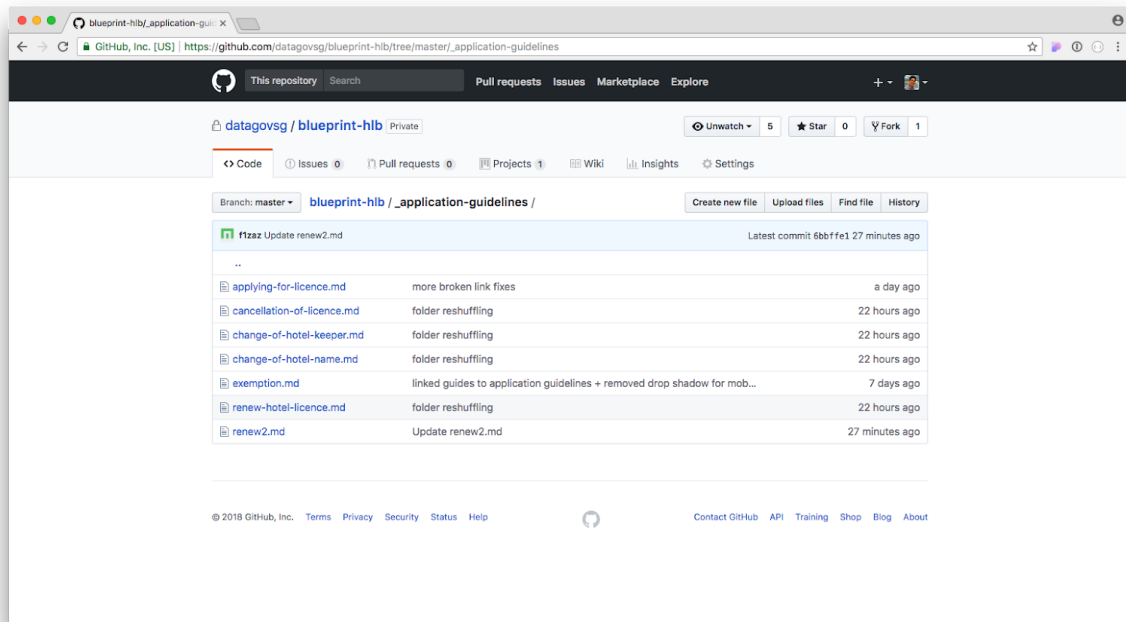
Sample application guideline on the Blueprint website

Step 1: Click on the _application-guidelines folder.



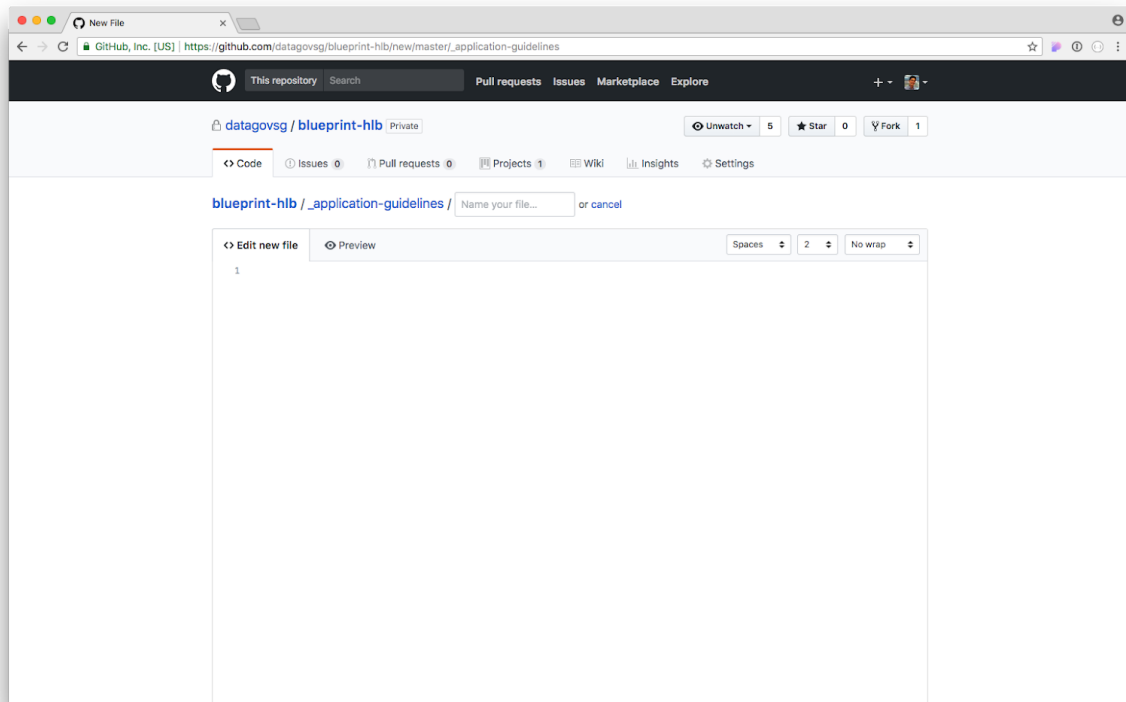
A view of the main repository

Within the `_application-guidelines` folder, there are several application guideline markdown files.



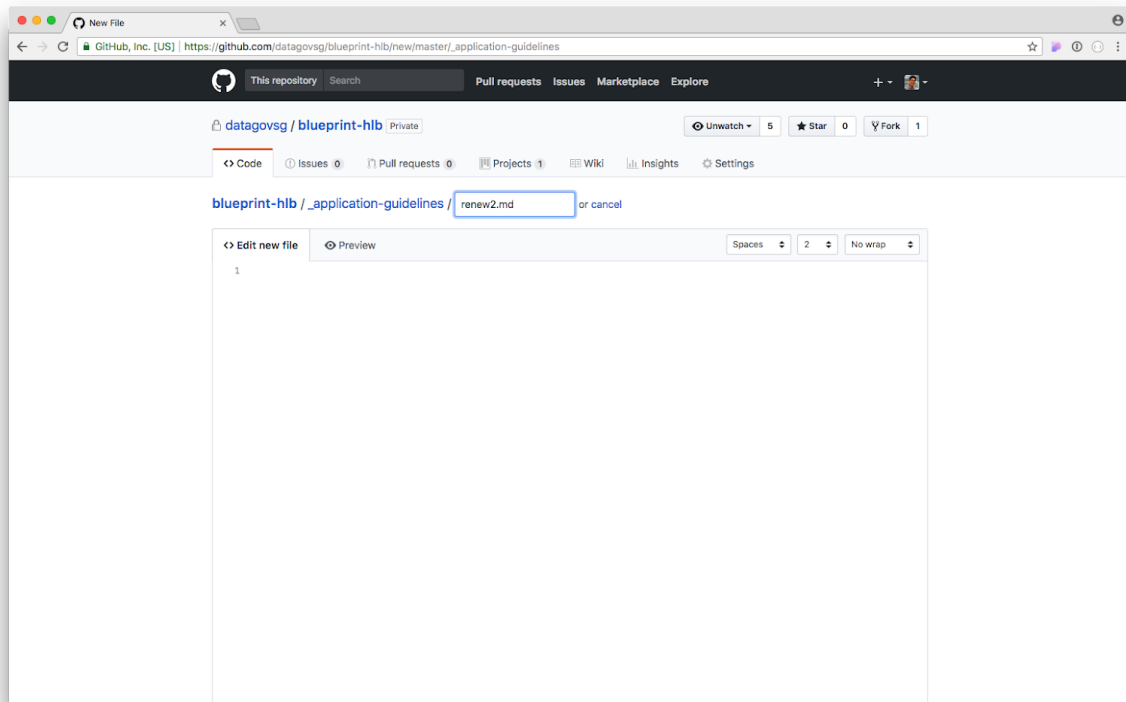
Markdown files in the `_application-guidelines` folder

Step 2: Click on the Create new file button.



Empty new file

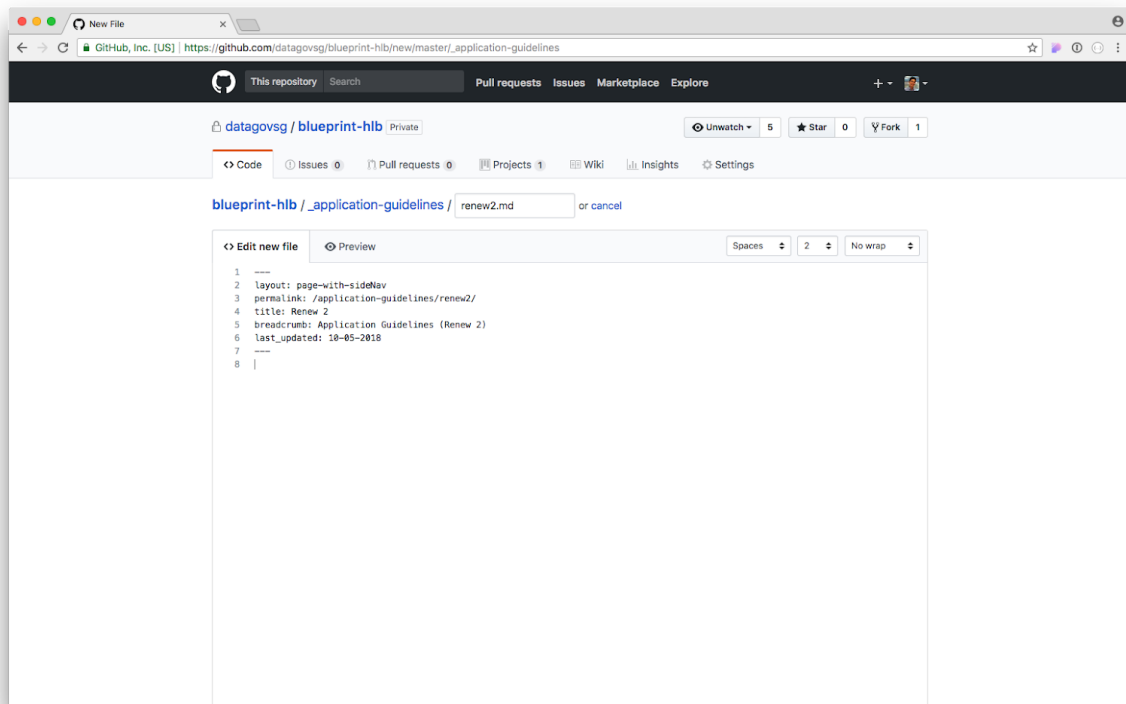
Step 3: Give your new markdown file a name, for example, `renew2.md`. Do this by typing into the *Name your file* input box at the top.



Hurrah, you named your file!

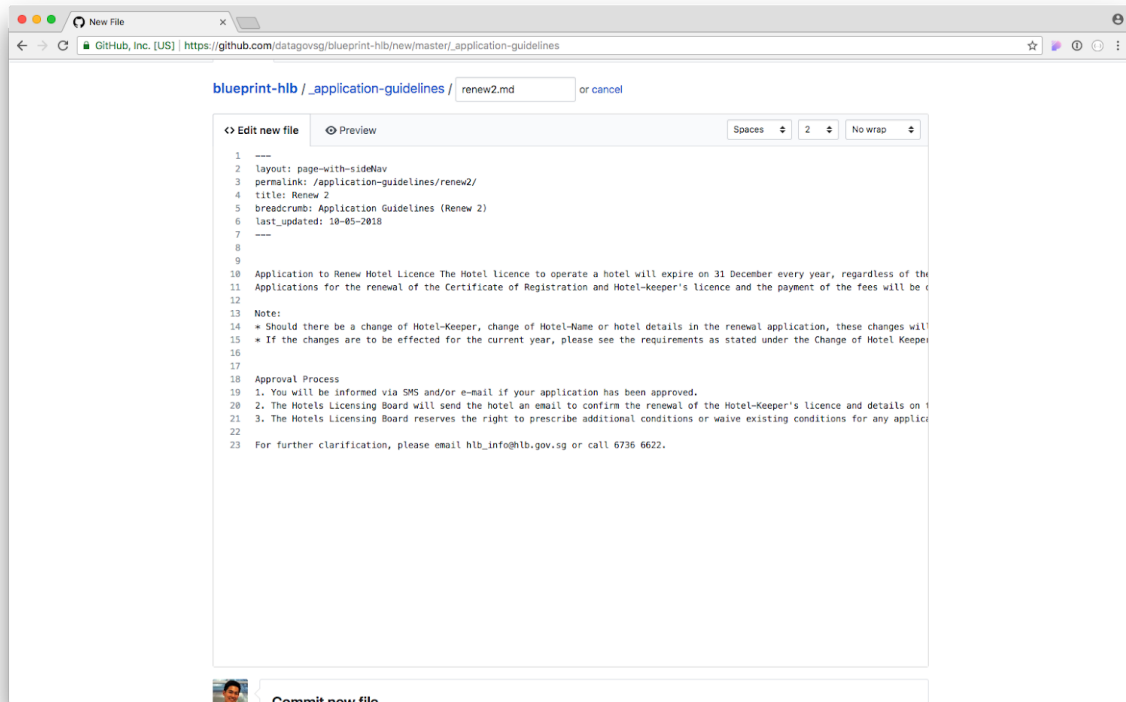
Step 4: Create your header for the markdown file. For example:

```
---
layout: page-with-sideNav
permalink: /application-guidelines/renew2/
title: Renew 2
breadcrumb: Application Guidelines (Renew 2)
last_updated: 10-05-2018
---
```



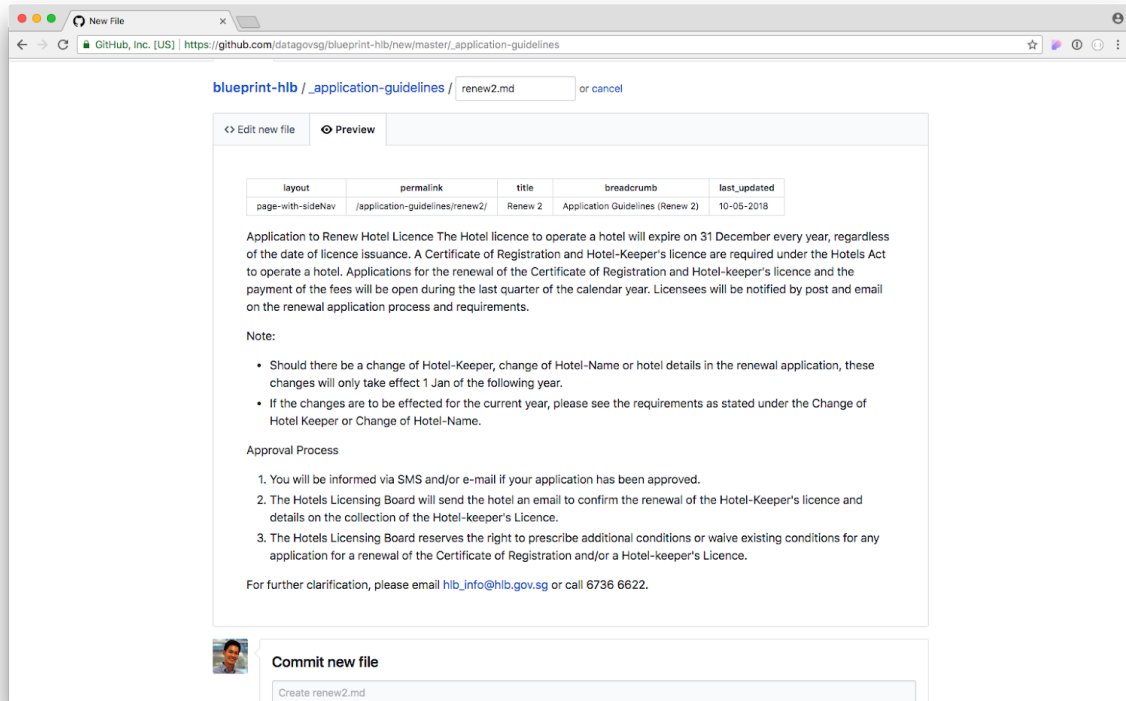
Header added to markdown file

Step 5a: Add your content. You should realize that the content becomes unformatted in the GitHub editor. We will have to add the markdown styles in the next step.



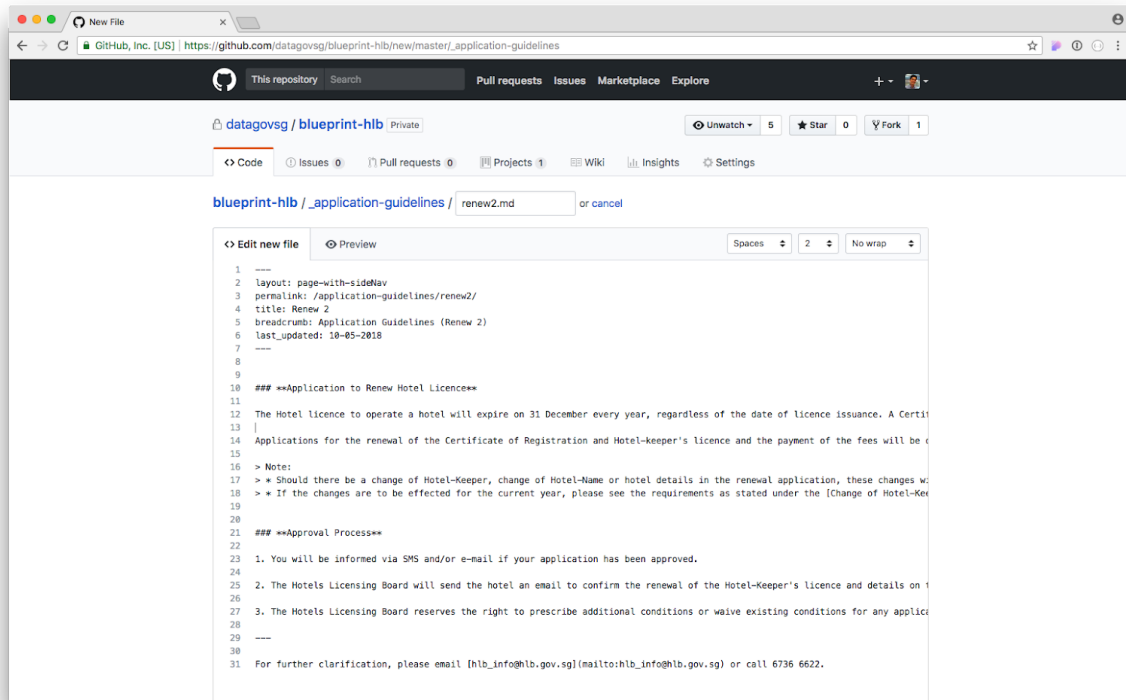
Content added to markdown file

Step 5b: Click on the Preview button. This shows you a preview of how your markdown will likely look when it is rendered – though it is not an entirely accurate representation of the page.



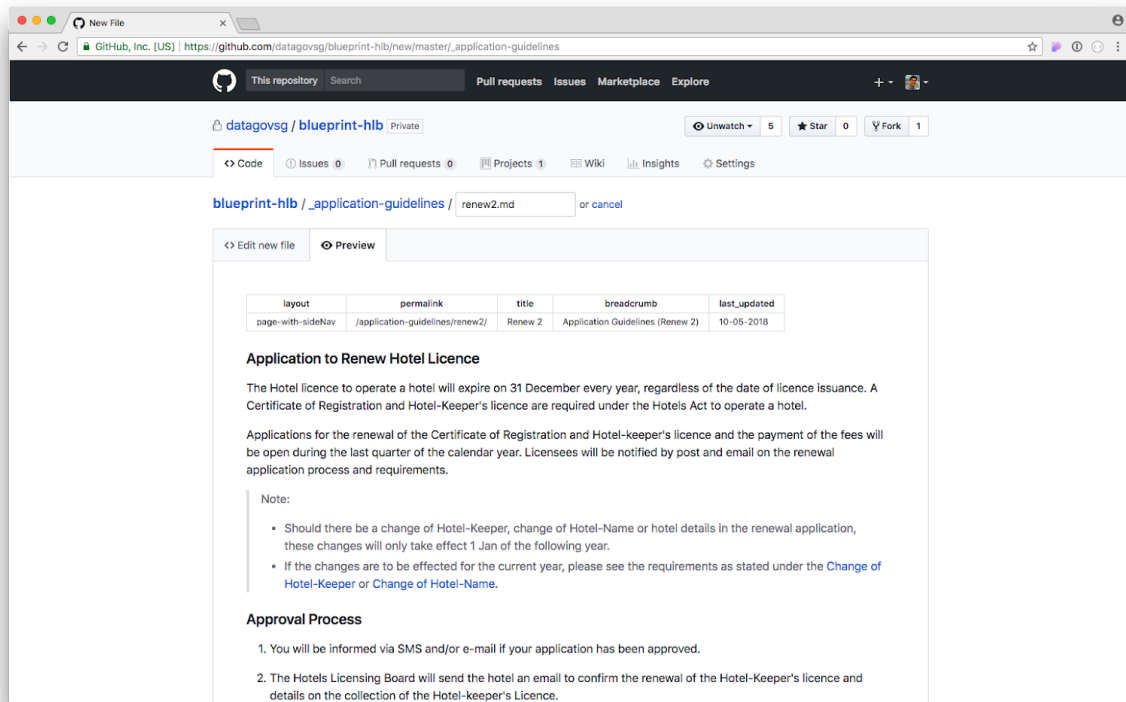
Preview of markdown file

Step 6a: Click on the Edit new file button to return to the editor. Add the relevant GitHub markdown styling.



Markdown content with styling

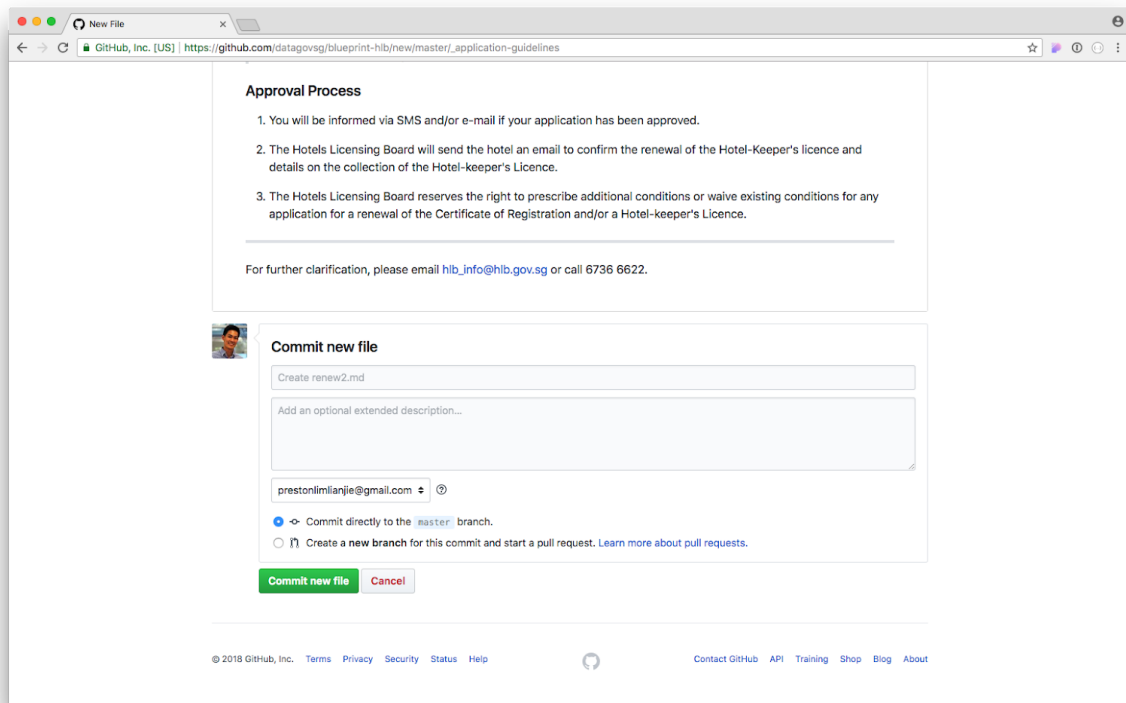
Step 6b: Click on the Preview button to check out the rendered page. This is close to how the website will look.



Preview of styled markdown file

Step 7: Scroll down to the bottom of the page. Leave an optional commit message (explaining why you did what you did) and click on the Commit button to save.

GitHub pages will then generate your website. It can take up to 1-2 minutes for the new version of the website to become live.

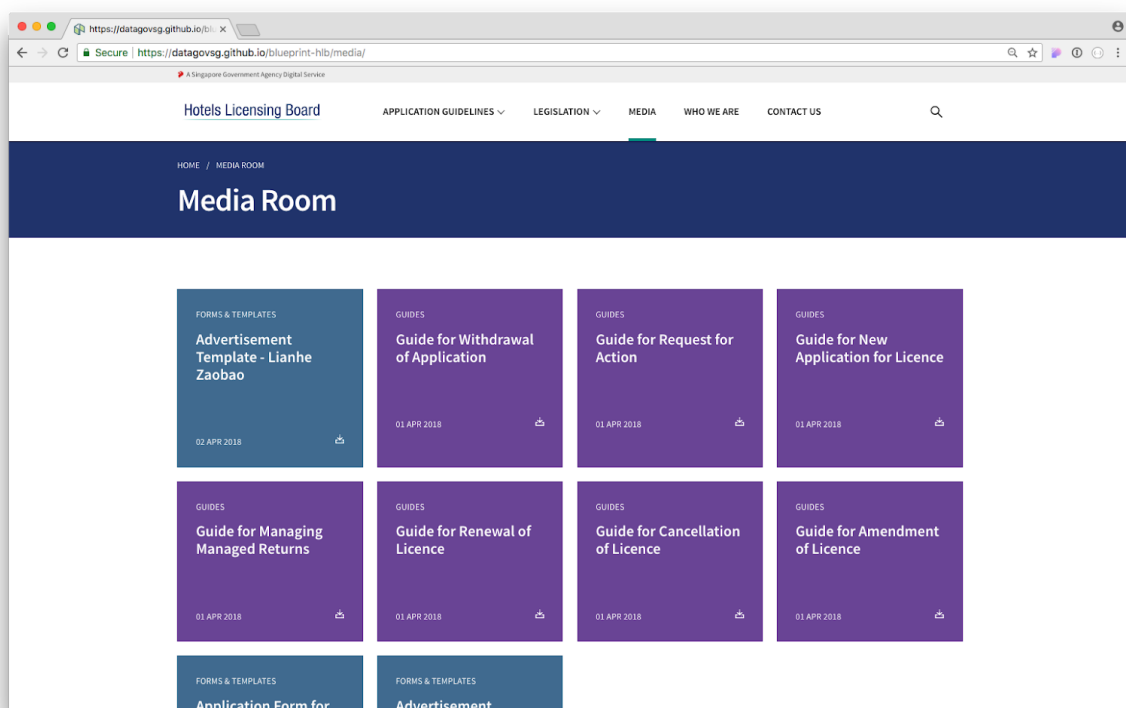


Task 3: Create a new Media post (Forms & Templates)

The Media section is a collection of markdown posts. These posts can either be: (1) files (e.g. PDF or Word Doc) or (2) articles (a website page containing words and images).

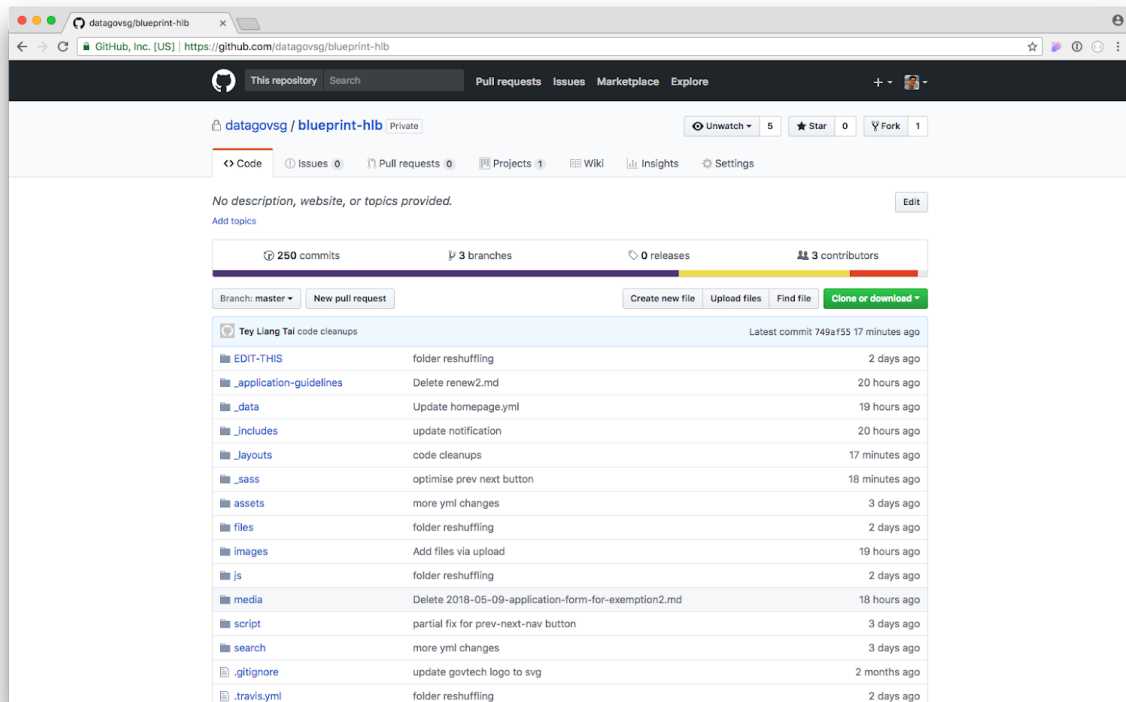
In this task, we will create a new Media post containing a .doc file.

Each Media post is a markdown file and contains a header and content, similar to that of the Application Guidelines. However, there is a slight difference in the header syntax – we will go into detail below.



Media page containing all the posts on the website

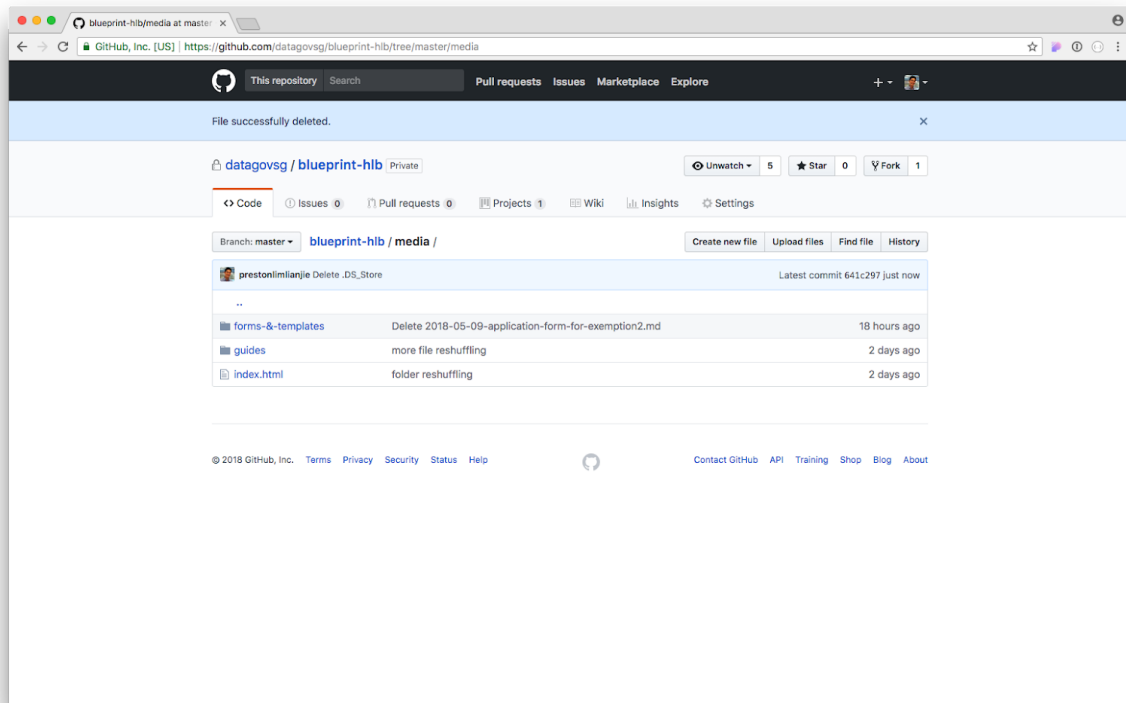
Step 1: Click on the media folder.



The media folder contains all the relevant media files

Step 2: You should see two folders: forms-&-templates and guides. These are the two categories of media files present on this website.

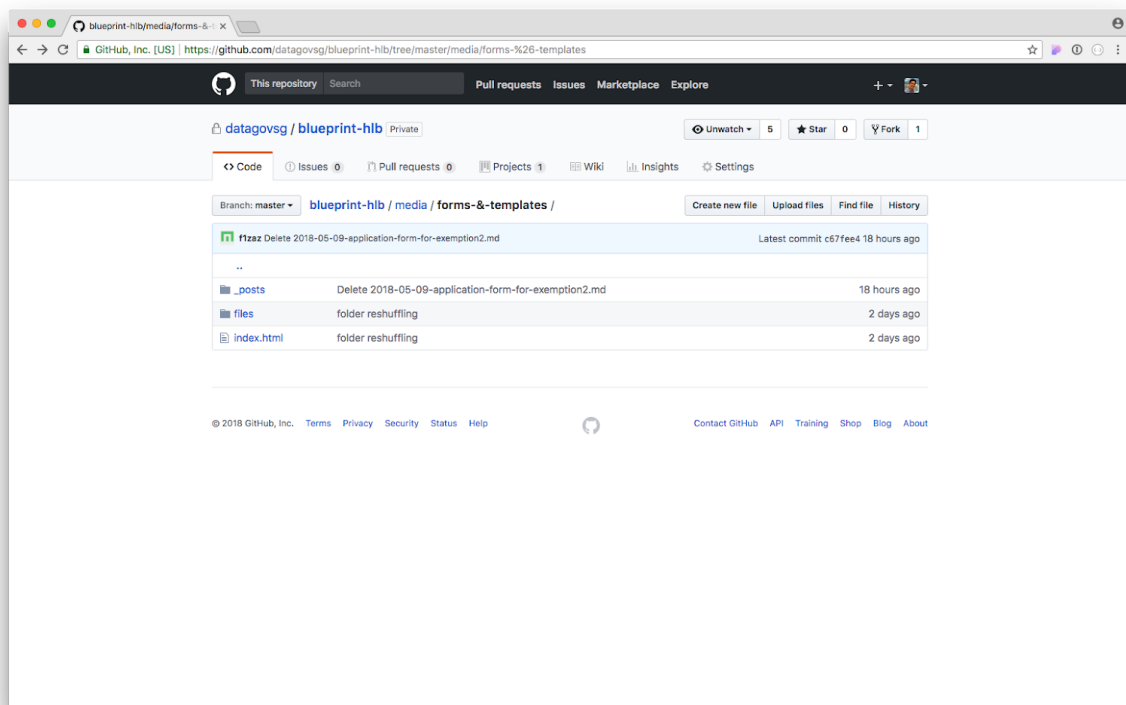
Click on the forms-&-templates folder.



There are two categories of media posts for this particular website

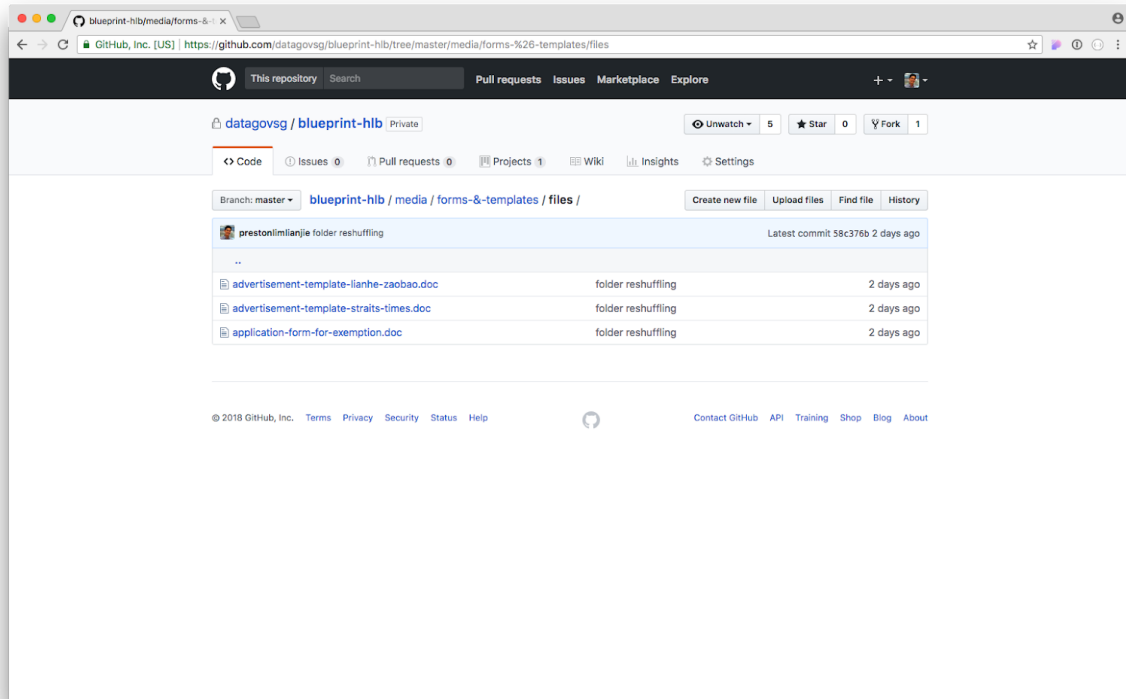
Step 3: At this point we see two folders: `_posts` and `files`. The `_posts` folder contains the markdown files denoting each post, while the `files` folder contains the actual files that the posts refer to.

What we need to first do is to upload the file onto the website. Click on the `files` folder.



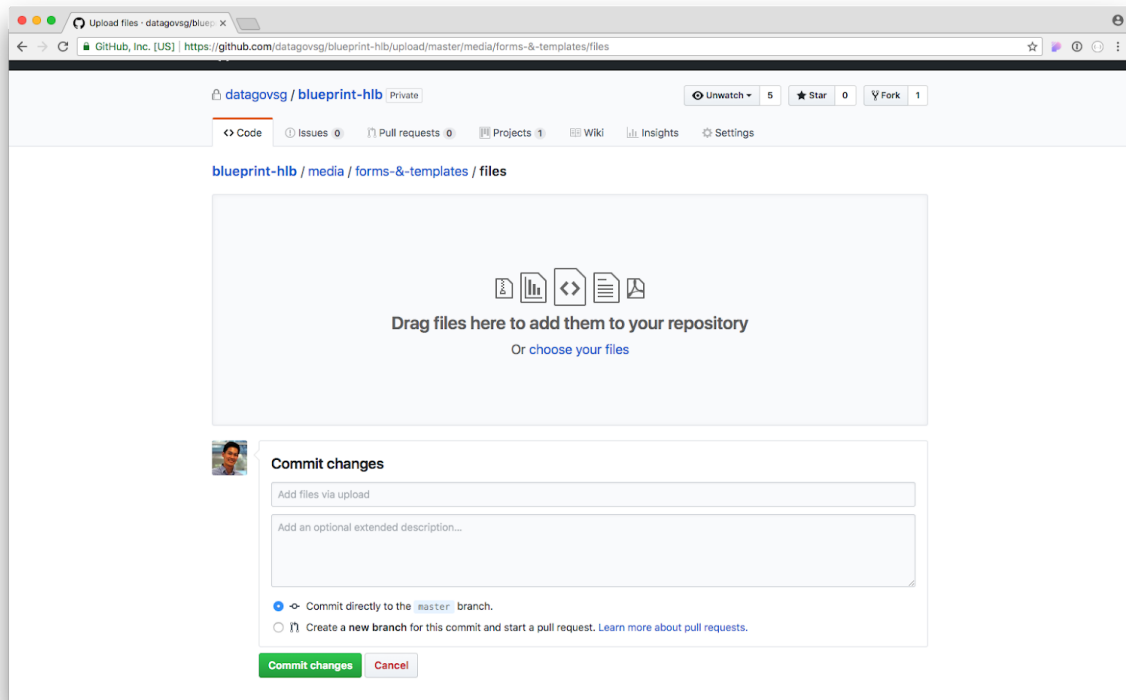
Click on the files folder

Step 4: Click on the Upload files button.



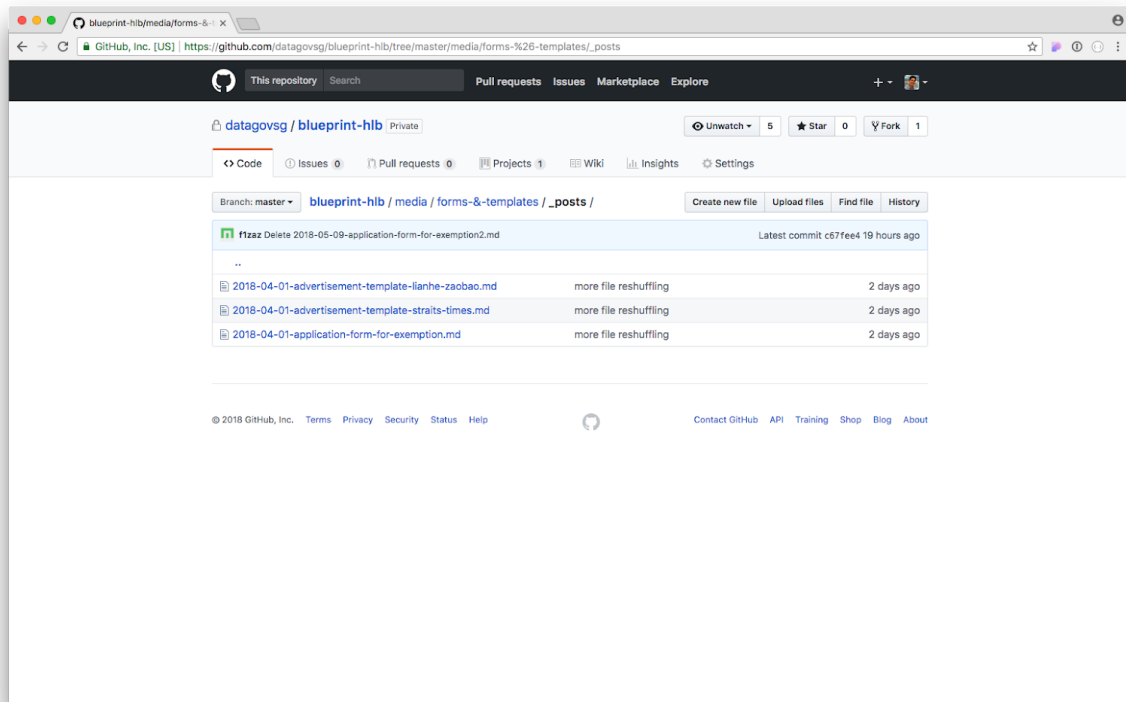
The files folder for forms-&-templates

Step 5: Upload the application-form-for-exemption2.doc file and click on the Commit changes button to save the file.



Upload and save your .doc file

Step 6: Now go to the `_posts` folder that we saw previously. Click on Create new file.

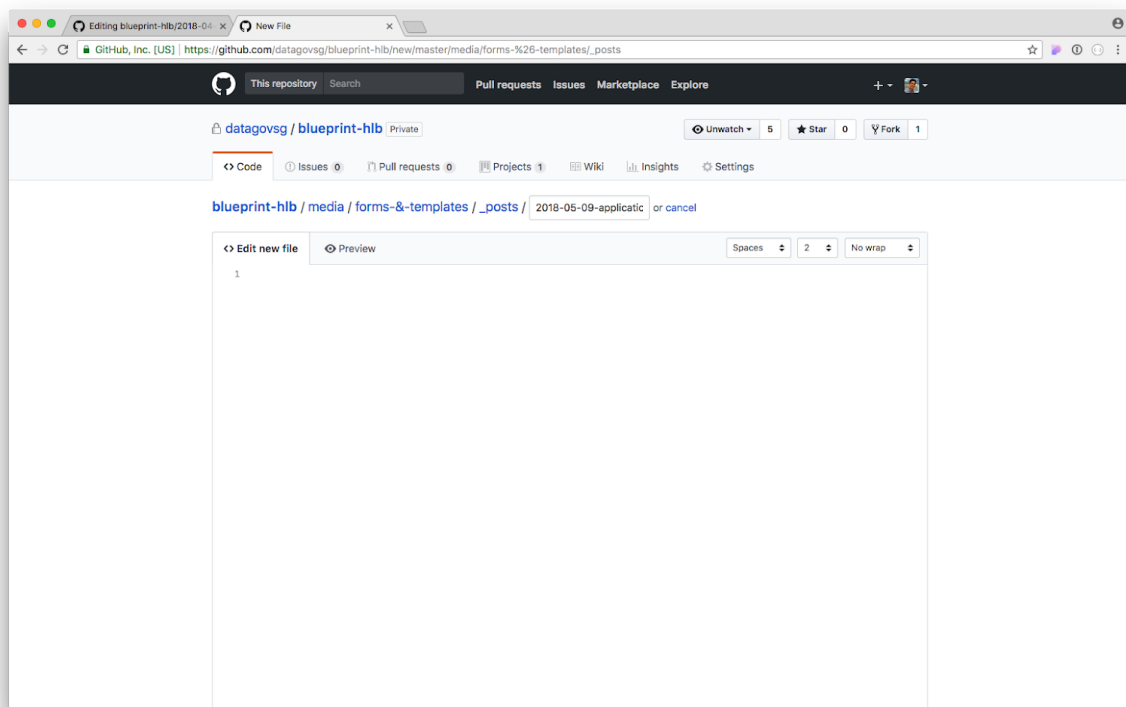


The `_posts` folder contains the post markdown files

Step 7: Give your markdown file the following name:
2018-05-09-application-form-for-exemption2.md.

The Media post file naming convention is as follows:
YYYY-MM-DD-your-file-name.md.

There must be no special characters (including whitespace) and the words must be in lower case.



Give your file a name – pay attention to the Media post naming convention!

Step 8: Type in the following header.

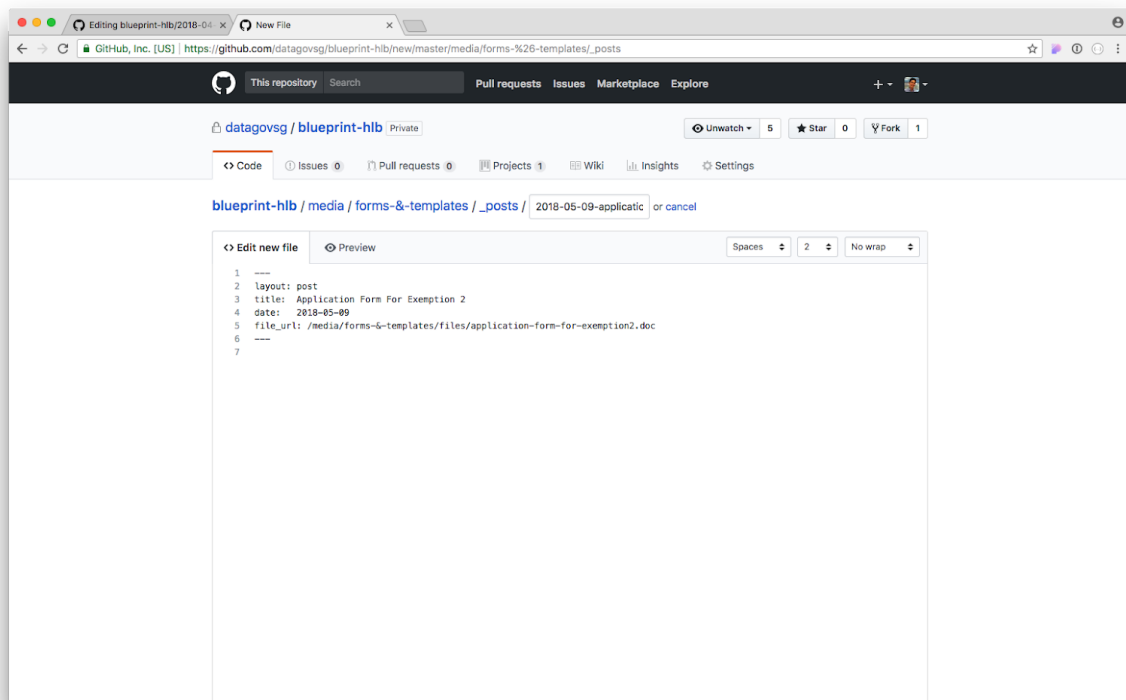
```
---
layout: post
title: Application Form For Exemption 2
date: 2018-05-09
file_url: /media/forms-&-templates/files/application-form-for-exemption2.doc
---
```

The layout tells the website generator what the page should look like (in this case, it should look like a post).

The title – as the name suggests – is the title of the post. It is case insensitive.

The date is the date of upload; it follows the format: *YYYY-MM-DD*

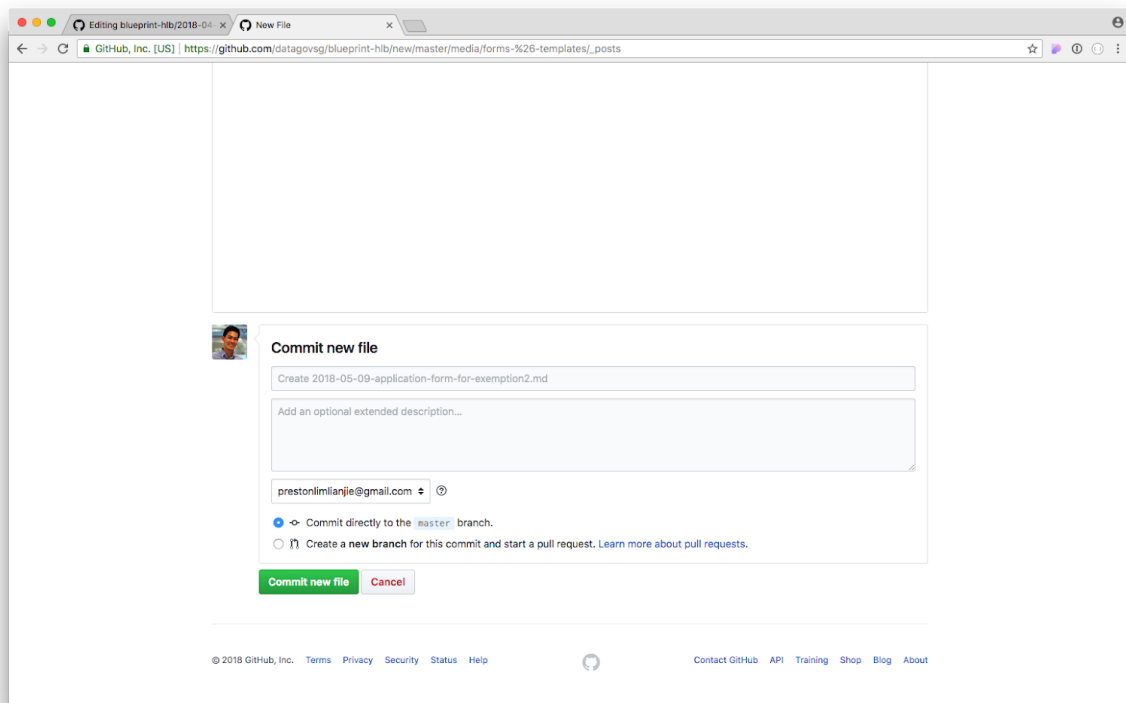
The file_url is the folder where we had previously uploaded our file.



Type in the header

Step 9: Unlike the Application Guideline markdown, the Media post that contain files don't need to have content. Our work is done!

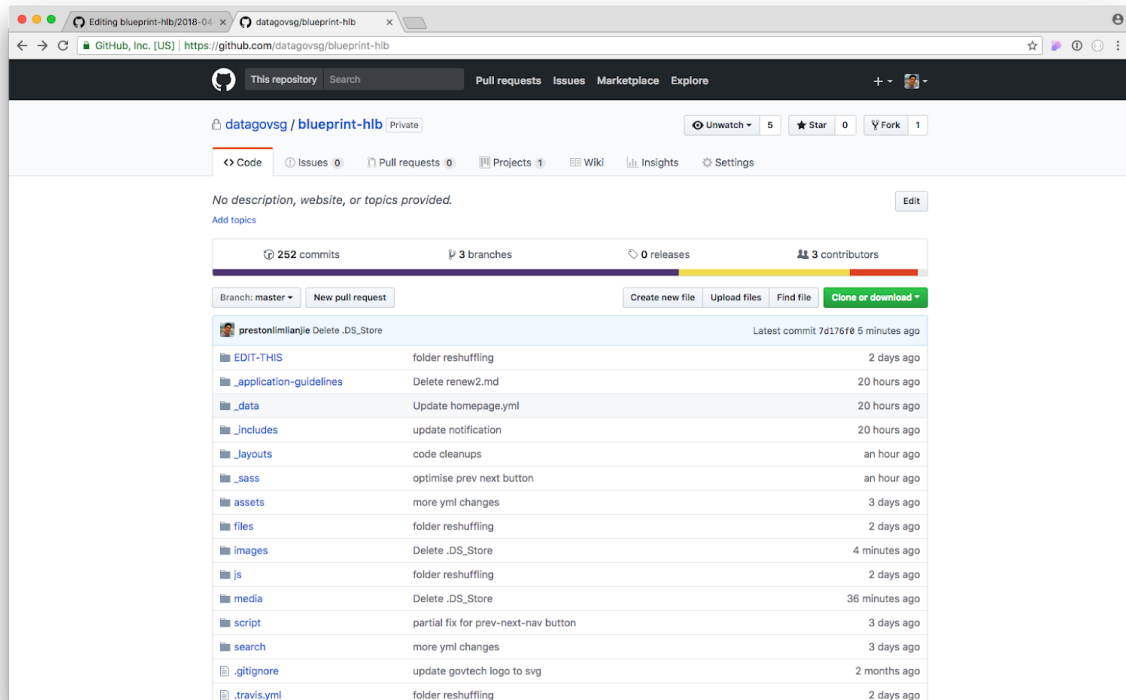
Scroll down and click on the **Commit new file** button to save your post. You should be able to see the new file on the website in a minute or so!



Click on Commit new file to save your file.

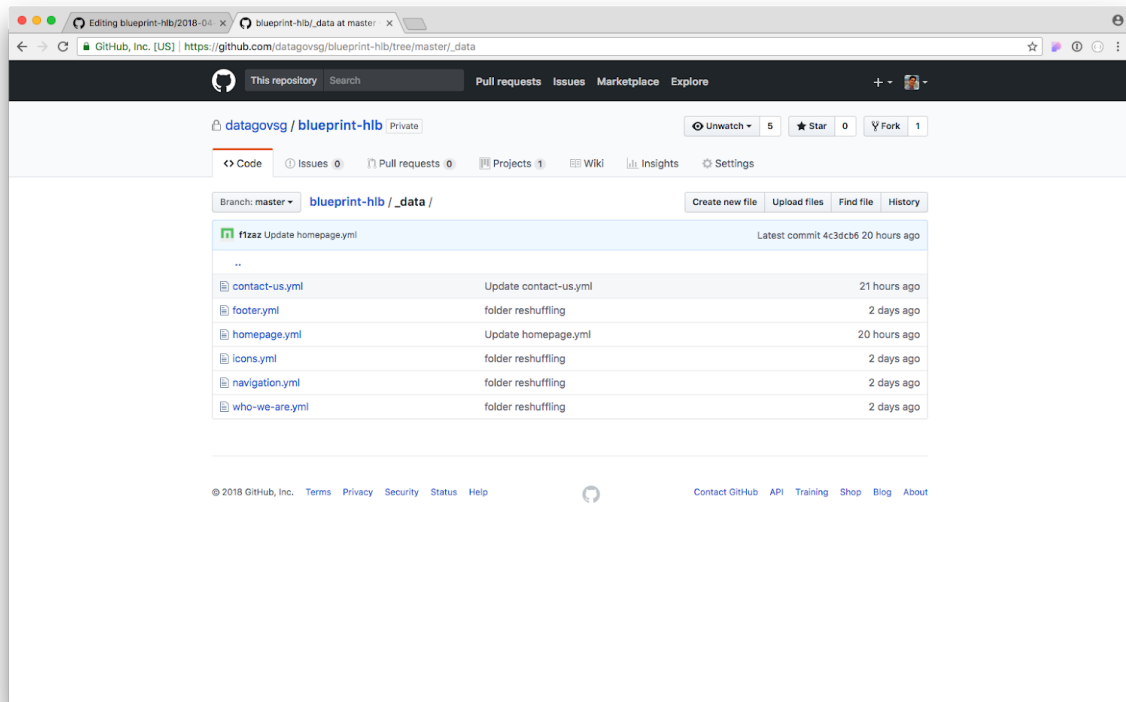
Task 4: Edit the Contact Us Information

Step 1: Click on the `_data` folder.



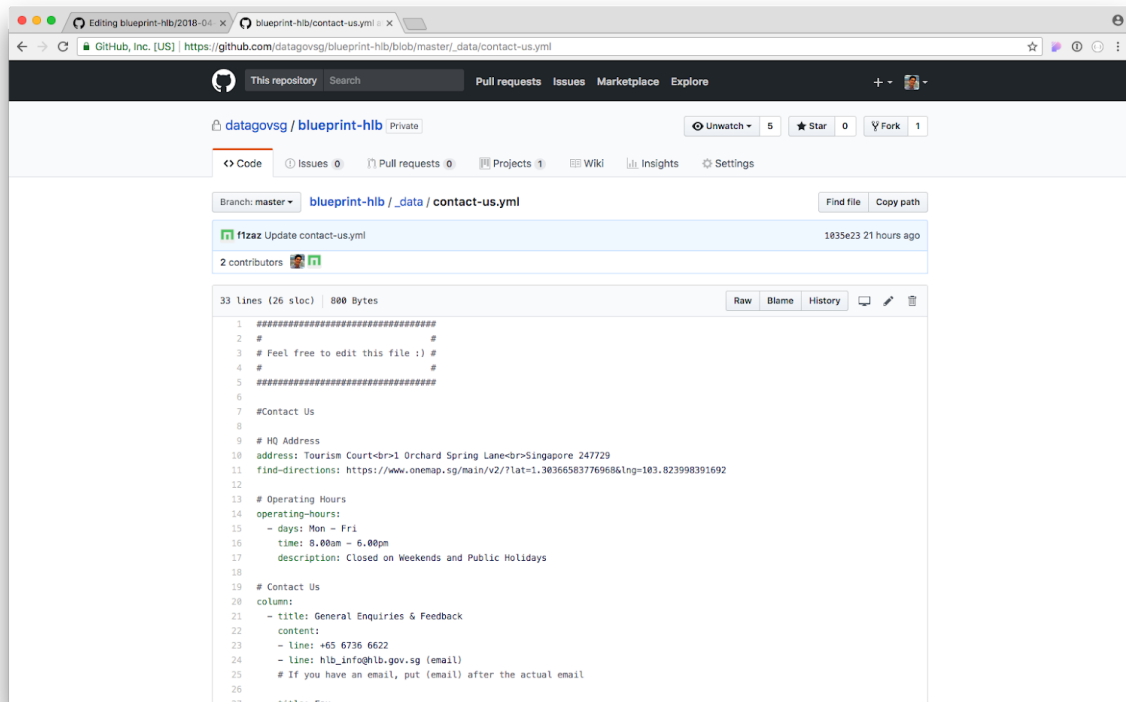
Go to the `_data` folder

Step 2: Click on the contact-us.yml file.



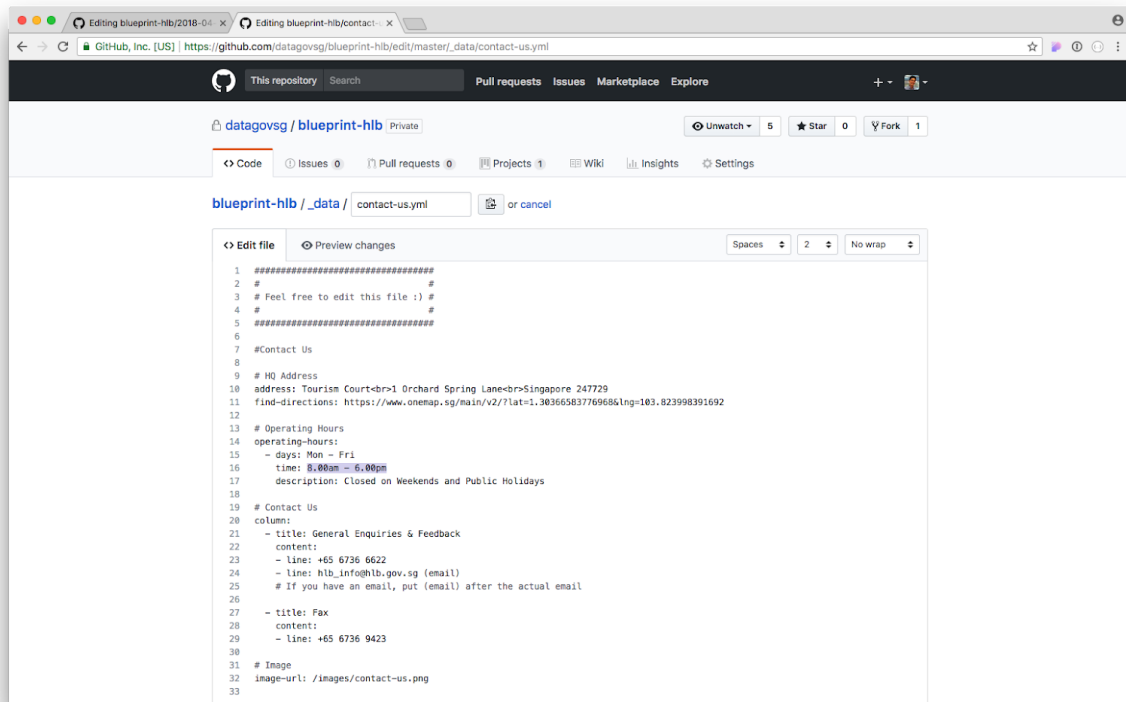
Click on the contact-us.yml file

Step 3: We should now see a preview screen for the contact-us.yml file. Click on the edit (pencil icon) button.



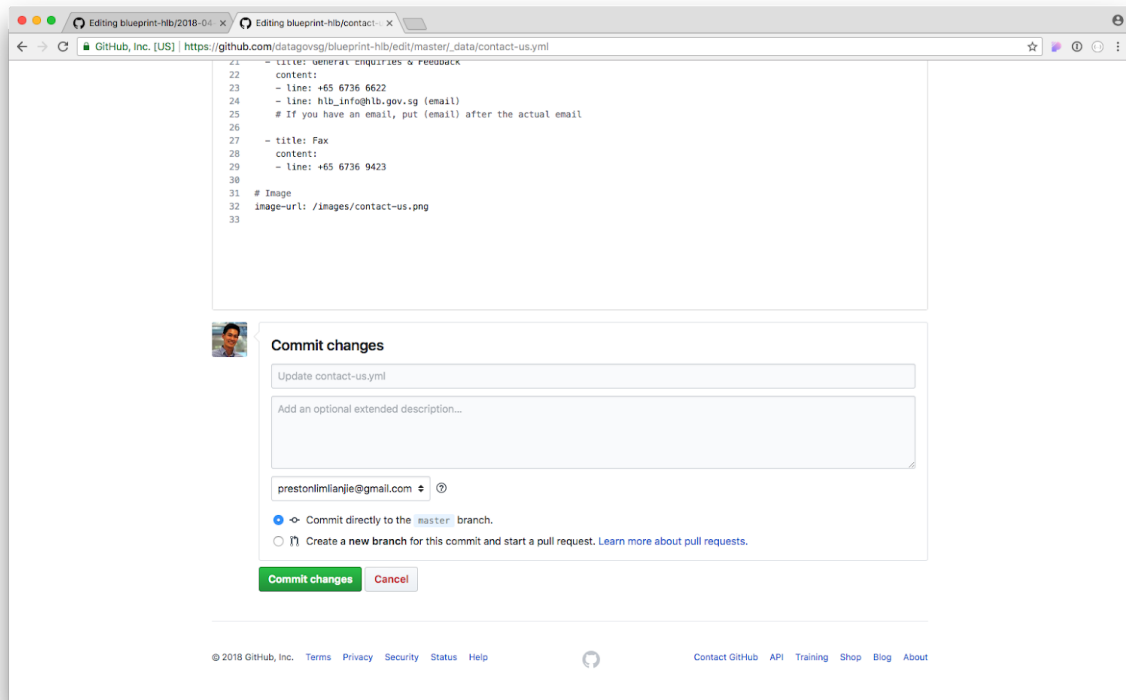
Click on the edit button

Step 4: Replace the time of the operating-hours for Mon-Fri with:
8.30am – 6.00pm



Replace the operating hours

Step 5: Scroll down and click on the Commit changes button to save. The change should be reflected on the website in a minute or so.



We're done!

Cheat Sheet

Why Markdown and YAML?

Traditionally, in order to edit a website, one had to write in pure HTML. This is challenging for website administrators who had limited experience with HTML as it is not immediately readable or intuitive to non-experienced users.

Our team decided to go with Markdown (.md) – more human readable syntax – so as to empower non-technical website administrators.

While Markdown is appropriate for writing long segments of text, it alone does not allow users to edit small details of the website (e.g. change the hero-banner image) without diving into the HTML.

We therefore chose to supplement the Markdown syntax with YAML (.yaml) – yet another human readable syntax.

What can I edit?

_application-guidelines

Contains the Application Guidelines

media

Contains the media-related files, e.g. Forms and Templates and Guides

EDIT-THIS

Contains the rest of the markdown files, e.g. Privacy Statement, Terms of Use, Who We Are, Contact Us

_data

Contains the YAML files to make fine tweaks to the website.

Images

Contains the images used in the website.

How should I go about editing my website?

The best practice is to always look for the Markdown file first. The Markdown file tells the website generator what to do.

Sometimes, if the edits are to be made to the YAML file that is related to the Markdown file, the Markdown file will contain instructions on where this YAML file is.

For example, if I wanted to edit the Who We Are page on the website. I would go into the EDIT-THIS/who-we-are/who-we-are.md file. I could make changes to the paragraphs of text there. However, if I wanted to change the names of the board members, I would – as instructed by the Markdown file – have to go to the YAML file to do so (_data/who-we-are.yml).