Migrate FLOE and Fluid project websites to a Modern Static Site Generator

My details

- Name - Kumar Chaitanya
- Email - chaitanyas0101@gmail.com, chinuon@circlepanda.io
- Telephone - +91-8572896685
- Timezone - GMT +0530
- Institution - Maharishi Markandeshwar(Deemed to be University)
- Course - B.tech. CSE
- Location - Mullana, Ambala, Haryana, India
- IRC - @chaitanya
- GSoC nick - Chaitanya
- Twitter - https://twitter.com/Chinu282828777
- Blog - https://sites.google.com/view/braveknightgame/home

Links to profile

- GitHub - https://github.com/Chaitanyassr

Inclusive Design Institute and fluid-project Community participation status

- #fluid-work IRC username ‘chaitanya’.
- I have done minor contributions in fluid-project to understand the community.
- I have started learning infusion as it has been used in nearly all the projects of Inclusive Design Institute and fluid-project.

Introduction

My name is Kumar Chaitanya, I am a 3rd year student, pursuing B.tech in computer science and engineering from Maharishi Markandeshwar(Deemed to be University), India. My main areas of interest are game development and web technologies. I also do android development. I am an avid lover of open source and a heavy GitHub user. Apart from this I also have a keen interest in reading, writing, acting, music, art and dance.

Some of my notable projects are -
Brave Knight-@ since 2019 sept(100+ downloads on Google play) - https://github.com/Chaitanyassr/BraveKnight--
Obesity(android game) - https://github.com/Chaitanyassr/Obesity

I am a core contributor at Circlepandalabs and a co - creator of **web framework Pandakit** - https://github.com/CirclepandaLab/pandakit and also contributed to a few of the projects of FOSSASIA.

Education & Experience

My specialization is in the field of game development and web technologies. The courses that I have done are -

- Responsive web design - https://www.freecodecamp.org/certification/fcc38946725-e399-4985-916a-2ace6b5e8485/responsive-web-design
- Vanilla JS course
- Introduction to the internet of things
- Game development using unreal engine 4(C++)
- Introduction to Hugo

I had also led my team to Smart India Hackathon 2019 finals where we made a Web scraper using Jsoup library.

I had also organized a workshop on micro-controllers.

I am currently working on many cool projects related to the web. Few of them are related - Picsearch(A reverse image search engine), Pandakit(web framework) etc.I am also working on few of my gaming projects like Dark_matter which is a FPS game with very extraordinary visuals. I use Godot and unreal for game development.

I have been working with CirclepandaLabs for 4 months so I am very much comfortable working with people with different time-zones.

I have been self taught in JAMstack and Hugo. Programming languages that I know are - C++, Python(flask), java ,php, JavaScript, and GD-script.

I have already gone through the whole source code of the FLOE and Fluid project websites. I had also gone through the Social Justice Repair Kit and Coding to Learn and Create.

Have you ever worked with Git ?

- Yes,I am very much proficient with Git. (https://github.com/Chaitanyassr)

Project Plan

Abstract

- Migrating floe website to static site generator.
- Migrating fluid project website to static site generator.

Technology - stack

- Modern static site generator - Hugo
- For deploying static site - Netlify

Why Hugo ?

- As new content will be added in both Fluid and FLOE sites. It should be well insured that the site building should be fast and Hugo is much **faster** than other static site generators**.**
- Hugo supports YAML , TOML and JSON which gives flexibility to use whatever we are comfortable with.
- Built-in live reload server.
- Built-in support for dynamic API driven content without using any external plugin.
- Shortcodes - Hugo offers a flexible alternative to markdown assets handling.
  Syntax - {{< shortcode-name parameter >}}
- By default, Hugo creates new content files with date, title and draft = true. This promotes consistency for sites using multiple content and saves time.
- Hugo directory system is so well formatted

```
archetypes/
content/
data/
layouts/
static/
themes/
config.toml
```

.

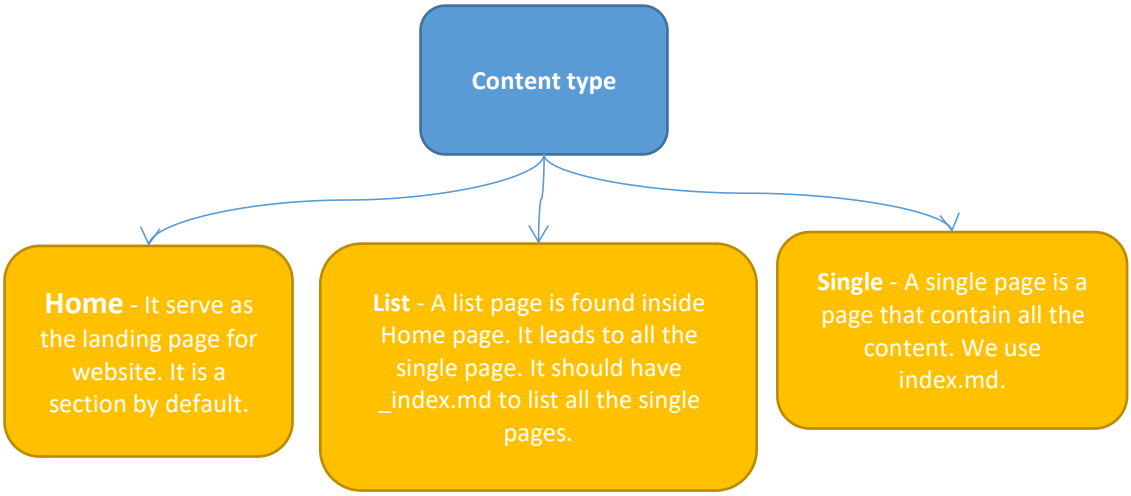
- As the project demands that the site should have well defined processes for updating and adding new content and Hugo has the most powerful content model.

Example given below -

```
Example -
 |── content/
    ├── main           <-- https://example.com/main/
        ├──_index.md
        ├── about         <-- https://example.com/about/
           |── _index.md
           |---- a.md <---- https://example.com/about/a/
        └── footer.md   <-- https://example.com/main/
```

Where main and about are sections and other are markdown files. A section should have an _index.md file for it to be called as a section and to list all the pages. Example - https://example.com/about/ will not go to list the page if it didn't have _index.md file. Main would not be needing an _index.md file as it is at the root level of the content so it is a predefined section.

- Hugo is very scalable. Hugo separates all content into three types of template - Home, List and single page.



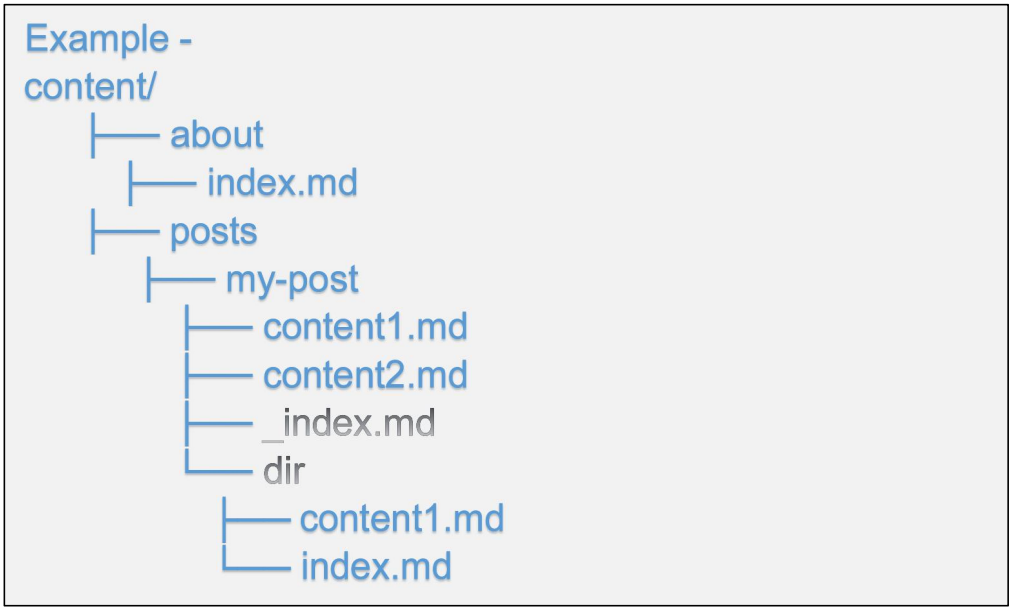
There is another kind of template apart from these that is **404**.

- In Hugo, Content organization is done by using page bundles. They are of two types -
  - **Leaf bundle** (in which index file name is **index.md**. **It is use for single page**)

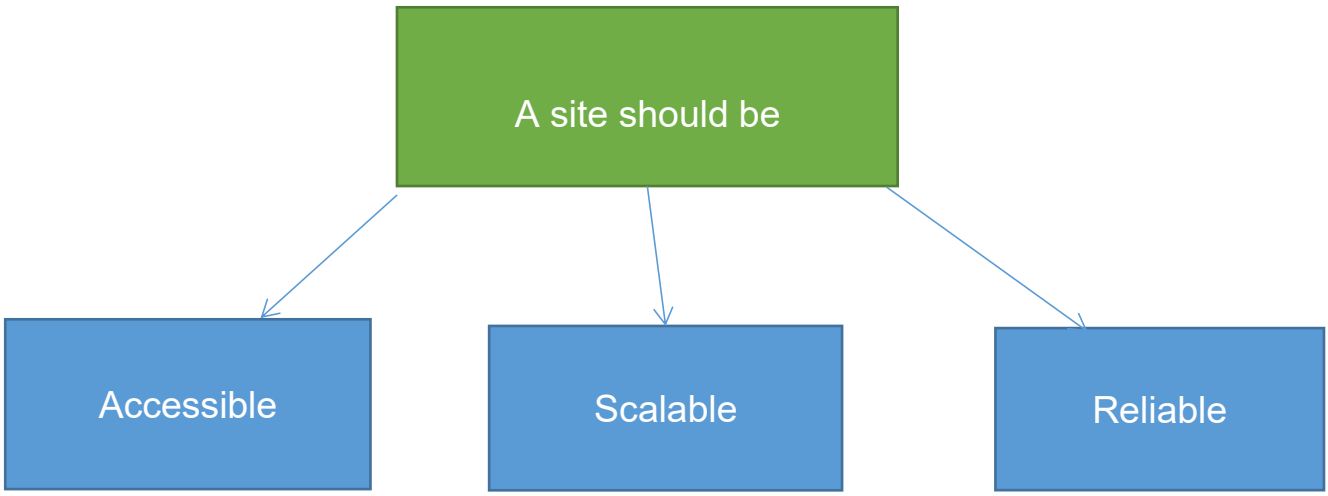
- **Branch bundle**(in which index file name is **_index.md**. **It is used for the list page**.)

Leaf Bundle - It does have any children. It us used for the single page.

Branch Bundle - It is like a leaf bundle inside a leaf. Suppose we want to add a directory in my-post in the above example. Then we have to use _index.md instead of index.md.



- We can also have _default in the layouts folder which will contain baseof.html which has a predefined layout that we will be using in every page.

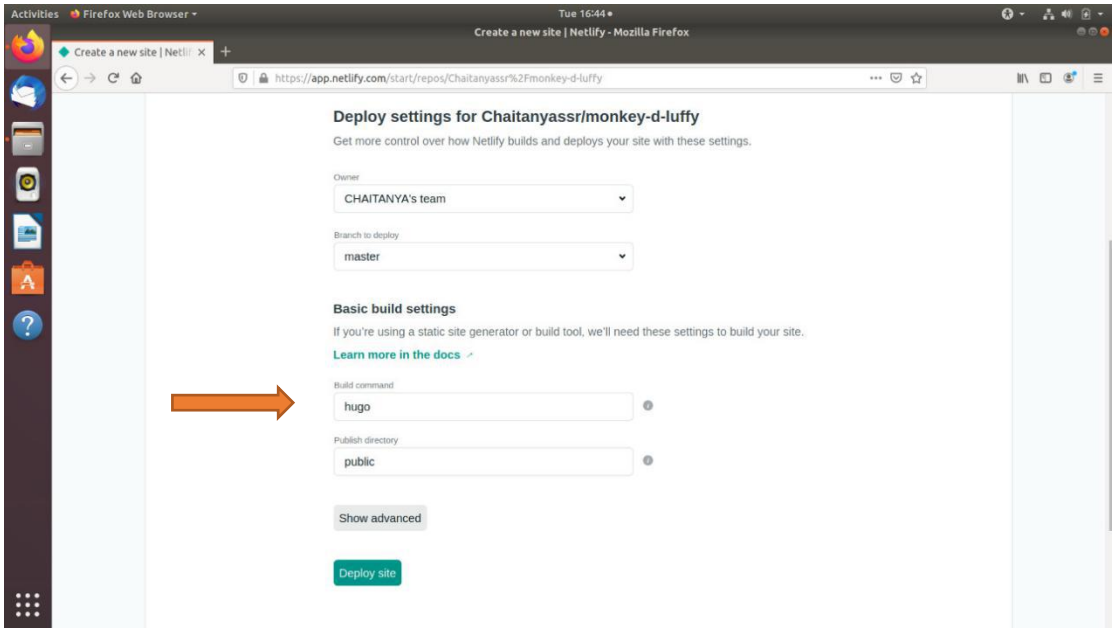


Comparing a few popular static site generators with Hugo on the basis of these three basic needs.

	Hugo	Jekyll	Gatsby
Accessible	Very **easy to install**, So accessible to all.	Setting up for windows users might be difficult as it needs a Ruby environment, So not that accessible.	has 2182 dependencies (according to npm ls --parseable | wc -l), that is 2181 more than Hugo.
Scalable	**Well defined process for adding new content**. **Lighting fast**(written in Golang)-It takes only milliseconds to build a site with 10k+pages.	We have draft and post but the directory structure is not that vast when compared to Hugo. It takes multiple seconds to minutes to build a site.	It has a well based directory structure that is automatic routing based. It takes multiple seconds to minutes to build a site.
Reliable	**Hugo is an old project, widely used, smashing magazine uses it, so it is much reliable**.	Jekyll is losing popularity these days, reliable but not better than Hugo.	Gatsby is a new project, can be said reliable but has a lot of moving parts so tough to work with.It also has a lot of bugs.

Why Netlify ?

- Netlify automatically updates the site when you push the commit from GitHub.
- The experience of deploying using Netlify is better than using GitHub pages as you can see the deploy logs and build failure message instead of getting email saying something is wrong.
- It's very easy to use Netlify. I have been using it for quite a long to deploy my Hugo build site. It automatically identifies that the repository is Hugo build.



Goals

- Migrate FLOE project website -

1. Modifying archetype file.
2. Putting files in the assets folder.
3. Adding markdown file in content folder.
4. Putting news content in a data folder.
5. Adding HTML files in layout.
6. Putting CSS,JS files in the static folder.
7. Fixes for issues #76,#94,#105,#106,#116.
8. Enhancement

- Migrate Fluid-project website -

1. Putting HTML files in the layout folder in the site generated by Hugo.
2. Modifying archetype file.
3. Putting the files in the static folder of Hugo after taking out the appropriate file form files from the SRC folder in DOCPAD.
4. Putting markdown in the content folder.
5. Putting codes in the static folder.
6. Putting JSON in the data folder of Hugo generated static site.
7. Fixes for issue #27,#28, and #29.

Technical details

Migrate FLOE project website -

1.
   A. As Archetypes are content template files in the archetypes directory that contain preconfigured front matter so all the files and code related to footer will be put here.

B. I will also put a few of the content of preference shown in the FLOE website.

2.
A. As seen in https://github.com/fluid-project/sojustrepairit.org assets folder contain CSS related files. So I would be putting all the CSS work in it.
B. I would be creating a folder named resources if necessary.

3.
A. Content folder will have an index.md for the main FLOE website.
B. Content folder will contain plain text or markdown of all the 3 pages linked with the FLOE page that is of News,Resources and UI Option.

4.
A. Data folder would be used to store data or news in JSON format.
B. An array will be created in JSON for news.

5.
A. Layout folder will have one list page for the main page and another 3 single pages for News,Resources and UI Option.
B. A baseof.html will be created inside _default folder inside layout.
C. An entity called block will be used for common code.

6.
A. Static folder will contain all the JavaScript file and all the infusion code inside lib(static/js/lib/infusion)
B. As in https://github.com/codelearncreate/c2lc-website we have all the images inside static/images. So, I would be putting all the images inside this folder for the FLOE project website.

7.
A. #76(back to the top) - A PR is already submitted but it needs some design consideration. I would be working on this issue but will discuss this with my mentors beforehand.
B. #94 - A PR is already present but not merged. Would discuss with my mentors if I need to incorporate it or not.

C. #105,#106,#116 - these issues were created by me and we also have PRs but they are not merged. I would be fixing it when migrating sites.

8.
A. Issue #133 I created this issue and would be working on it. https://github.com/fluid-project/floeproject.org/issues/133. In this we would going to have one button for expanding and collapsing the item inside class - floe-resources-main (web page - https://floeproject.org/resources.html). ToggleCategory will set to true when expanded and become false when collapsed.
B. Issue # 134 (https://github.com/fluid-project/floeproject.org/issues/134)(web page - https://floeproject.org/resources.html)I created this issue as an enhancement. Currently clicking on “learning to learn” button displays the content quickly, instead of this I would be making them display smoothly by using transition in CSS or JS(infusion).

Migrate Fluid-project website -

1.
A. Putting index file from layout of DOCPAD to layout in Hugo directory after rendering from ECO to HTML.
B. Putting HTML file for ABOUT, PROJECTS, INFUSION, NEWS in layout from documents.
C. Adding more HTML files from src/files.

2.
A. Adding preconfigured front matter of every page and content of preference from src/partials and src/files/html respectively.

3.
A. Forming Static folder in the site generated by Hugo.
B. Storing relevant files in it like putting the files in the static folder of Hugo after taking out the appropriate file form files from the SRC folder in DOCPAD.

C. I would be putting JavaScript and image files in it as done in the sojustrepairit.org repository.

4.

A. I would be putting all the markdown files in the content folder of the site generated by Hugo from src/documents/blog.

5.

A. Adding content in the static folder like all the JS and images as done in sojustrepairit.org(form files in src folder).

6.

A. Modifying the data folder by adding a JSON file that will store news in it.
B. Currently news is located in documents folder of src folder and stored in markdown files.
C. I will be creating an array in JSON file for news.
Example -

```
var arr = [{id:"1",Date:"",News:"",},

    { id:"2", Date:"", News:"",} …]
```

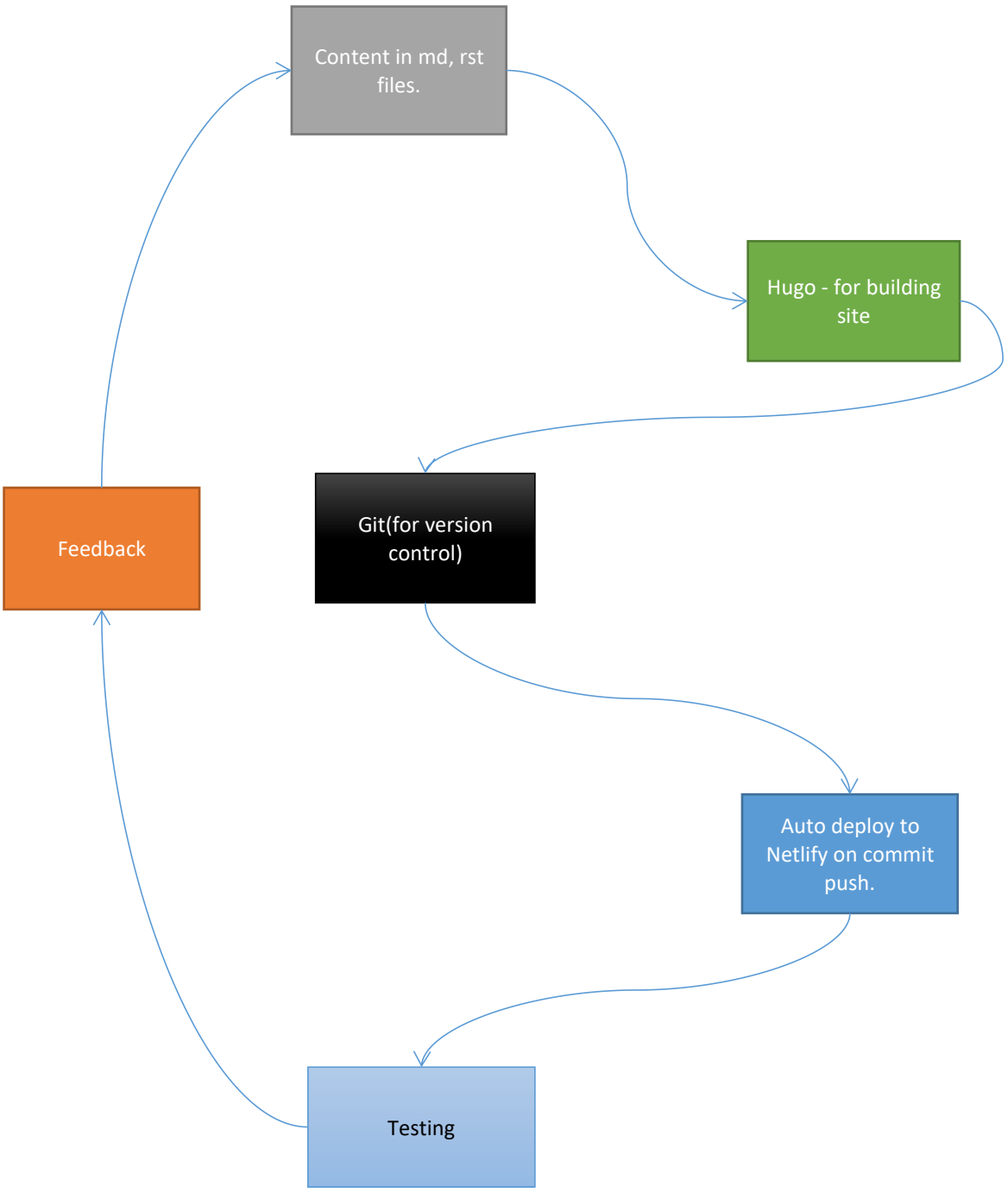
And than accessing it in HTML file in layout folder using -

```
{{ range .Site.Data.name_of_file.json }} … {{ end }}
```

7.

A. #27 I would provide a fix to this issue when creating the site form Hugo.
B. #28, #29 - these issues were created by me. Background image of both the main page and the project page(https://fluidproject.org/projects.html) get disappeared when colour and contrast is changed from the preference. I would be providing fixes for both these two issues.

Work-flow


Content in md, rst files.
Hugo - for building site
Git(for version control)
Auto deploy to Netlify on commit push.
Testing
Feedback

Timeline

Learning infusion and perfecting myself in other technology stack that is required.	April 1 - May 4
Community Bonding.	May 4 - June 1
*Exams.	April 28 - May 12 (approx)
Getting completely comfortable with existing code base, research and code documentation.	May 4 - May 18
Migrate FLOE project website - Adding markdown files in content folder	May 19 - June 1
GsoC Coding phase.	June 1 - Aug - 24
Migrate FLOE project website - Modifying archetype file	June 2 - June 5
Migrate FLOE project website - Putting files in assets folder.	June 6 - June 9
Migrate FLOE project website - Adding news content in data folder.	June 10 - June 14
Migrate FLOE project website - Adding HTML files in layout folder.	June 14 - June 20
Migrate FLOE project website - Putting code in static folder.	June21 - June 27
Migrate FLOE project website - Fixing issues.	June 28
Evaluation.	June 29 - July 3
Migrate FLOE project website - Fixing issues.	July 4 - July 5
Migrate FLOE project website - Enhancement.	July 6 - July 8

Migrate Fluid-project website - Putting markdown in content folder.	July 9 - July 12
Migrate Fluid-project website - Putting HTML files in the layout folder.	July 12 - July 20
Migrate Fluid-project website - Modifying archetype file.	July 21 - July 27
Evaluation.	July 27 - July 31
Migrate Fluid-project website - Making a Static folder.	Aug 1 - Aug 6
Migrate Fluid-project website - Putting JSON in the DATA folder.	Aug 6 - Aug 12
Deploying both the sites.	Aug 12 - Aug 15
Testing	Aug 15 - Aug 23
Project submission	Aug 24 - Aug 31

*Exams - The data of the exams are still not final as recent covid-19 pandemic. It can even get canceled. I am aware of the time I am going to lose during this period. I will be learning and perfecting myself in the technology stack that I will be using in this time period while also following the timeline that I have given.I will also try to participate in issues and discussions and will be in know-how of what is happening in the community. Once it is over, I have no other obligations and will be available full-time. I will always be ready to work for extra hours to stay on my schedule.

Final outcome

Fluid and FLOE-project websites will be migrated into Hugo.

Questions

How will you document your weekly progress ?

I will send emails to my mentors(Jon Hung(jhung), Gregor Moss (gmoss), Colin Clark(colinclark)) weekly writing about my progress. Although I am planning to do daily mails to my mentors but if you want weekly, I can also do that as well.

I am online most times of the day and will keep my IRC active while working. I will also be in touch with my mentors through IRC.

I am also going to use a task management system like Wunderlist so that mentors would be able to keep track of my work.

How will you ensure your project code continues to be maintained, after GSoC ?

I would be there to resolve issues and to manage the pull request. I would also do enhancement from time to time.

I will also contribute to other projects of the Inclusive Design Institute and fluid-project.

What is your GitHub account ?

https://github.com/Chaitanyassr

Will you be working on your own, or will you have partners ?

I will be working on my own.

What is your working environment ?

My working environment is a ASUS laptop having Intel i5 7th Gen processor with 8 GB RAM running Ubuntu 18.04 and Windows 10. My timezone is GMT +0530.

What are your other summer plans ?

I don't have any other plans. I will be free this summer as I am not taking any internships or courses so I will be making this as my primary task.I will be able to dedicate minimum 40 hrs per week.

How will you reach out to other developers and support other developers to join your project ?

I will try my level best to stay connected with the community and will not hesitate to ask them for help when needed. I will ask my friends to contribute to join the fluid-projects and Inclusive Design Institute on GitHub.

Are you applying to any other organization ?

No

Have you applied to GSoC in the past ?

No

Are you willing to do other projects instead ?

Yes

Conclusion

I am very much interested in taking part in GSoC with Inclusive Design Institute. Also working with Inclusive Design Institute will be memorable and I am sure that it would be a great learning experience.

Regards,
Kumar Chaitanya