Collaborative Website Project Preliminary Documentation

Brandon Kane
https://github.com/B08waffles
brandonkane23167@gmail.com
Web Developer

Contents

1 Introduction

The idea is simple, a bunch of us from 2022's Diploma of Website Development want to create a collaborative website where we can post our interview recounts, technological findings and learn through doing the process from which we will need to become familiar with in our chosen industries. This document aims to serve as a prelude to the discussions that have been organized on the following Monday from now.

2 Purpose

We want to learn and grow as developers and what better way to do that than to create a new project, collaboratively. We also have allot to gain from the potential content and visibility of such a project. Everyone will self assign themselves to various tasks according to their preferences and/or strengths. The goal is to familiarize ourselves with Git, including all the common commands and processes such as Pull Requests, Merging, Issues and so on. In addition to Git, other technologies such as Jira will be immensely helpful to submerge ourselves into during the development of this project. We will learn about "Epics", "Sprints" and more.

Depending on what technologies have been chosen, e may use this as an opportunity to learn new frameworks and languages to better equip ourselves for our future endeavors.

Ultimately we will produce a website that is capable of hosting our own blog style posts about our favorite technologies and interview recounts for the purpose of informing others and each other about our unique experiences and conclusions, in addition to allowing other like minded individuals to do the same by signing up, posting articles and comments.

3 Requirements

Here is a brief and potentially incomplete criteria for our project:

- All members have GitHub and/or GitLab accounts depending on our chosen remote repository system to use;
- Discussions must be had about what lanuages and technologies to utilize in this project;
- Members will have to assign themselves tasks or be assigned tasks to do in an efficient and democratic nature;
- Discussions about production and staging environments will need to be had;
- A formalized "todo" system will need to be implemented, such as Jira and/or Trello;
- Deadlines may be discussed in the group meeting;
- Wire-frames may be made to help all hone in on the desired outcome of the front-end;
- Other supporting documentation such as UML Diagrams and/or ERD may or may not be deemed essential for this project.

4 Technologies

4.1 Languages

One of the key decisions to be made will be the choice of language(s) that will be used for this project. There are a few important factors to consider when choosing the project's language(s), some of which include:

- Popularity, how common is this language in the industry? Will I use this language after this project?
- Employability, is there currently allot of job postings that require this language?
- Familiarity, how far out of my comfort zone am I willing to go for this project and my learning?
- Time, how long will it probably take to learn all this? Will it hinder our proposed deadlines to try this?
- Frameworks, what frameworks does this language support?

4.2 Frameworks

Another core decision to be made is what framework will be used in this project? There are quite allot to choose from after all, here's a few things worth considering when choosing a framework:

- Popularity, how common is this framework in the industry? Will I use this framework after the project?
- Employability, am I more employable for learning this framework? How many jobs are there for this framework currently?
- Familiarity, am I willing to learn something completely alien or would I do I have preferences for how my frameworks should be?
- Time, will learning this framework help or hinder my participation in this project?
- Languages, what languages does this framework support?

4.3 Production

When all is said and done, the code is working locally, everyone is ready to show the world what we've got, where do we host it? There are a few options, these are just some:

- Virtual Private Servers, we hire or purchase a server to setup manually via SSH and the command line;
- Heroku, they have an easy to use system and offer cheap/free deals depending on our needs;

- Siteground, they have a backup system of their own and they offset their carbon emissions which is nice;
- Netlify, while usually considered a static site environment, running a "server-less" build makes this a viable option;
- Network Attached Storage, does someone want to host the site on their own hardware setup locally?
- Another consideration is that would we like to have our own custom domain and if so how do we finance that between the group?

5 Collaboration

There is the possibility of using Discord, Matrix, Element, Signal etc to organise remote meetings and voice calls for collaborative purposes. Something I want to mention here is that in Visual Studio Code there are a few interesting extensions that could help us code, most notably I'd like to introduce you to "Live Share", its an extension that allows for multiple people to view and edit the same file/project. There are other standalone tools that offer the same capabilities as well although at the time of writing this the names of them escape me.

In essence, I want us all to consider how we could best deal with collaboration. Code together? Code over Discord? Run with a Jira ticket on our own? There are many options available to us.

6 Conclusion

Thanks for reading! I hope we can all say soon enough that we have worked as part of a development team within the IT industry and are proud of the results that we achieved.