

Personal Website, Requirements

Author: Yunfei Luo

May 15, 2020

1. Functionalities.

1.1. Welcome page

When user navigate to the website, there is a clean-style page, with brief introduction, links to the documents, and nice pictures. There is also a footer at the bottom of the webpage, containing the linkedin, github, and my contact information. This footer will be shown in all the webpages on the website.

1.2. Navigation to documents

User click on the link with either:

- i) navigate to software engineering documentations
- ii) navigate to reports of open source projects
- iii) navigate to blogs
- iv) navigate to search area! :)

For the first three cases, the list of corresponding documents will be shown with a clean style. When user click on the title, the documents need to be shown explicitly. AND, there need to be a back button to let the user navigate back to the list page.

1.3. Search for relevant document(s)

User click into the search area. They will search with key words: blogs, reports, image process, language model, what is search engine, what is this, to be or not to be, hey what's up! ...

The retrieved list will be shown in the same form mentioned in 1.2. The search bar will keep at the top of the webpage. The user could either click the document for reading (also same scenario in 1.2.), or try another query, or close the browser.

2. Non-Functionalities.

2.1. Development Schema: Waterfall, CI/CD

Since this is a personal project, waterfall pattern is a proper way to keep the progress going forward stably and correctly. Continuous Integration enable the consistency of codes among local development branch and the remote master branch.

2.2. Front-end Tool(s): React

The render functionality of React enables the concise response the quick and temporary click actions. The JSX syntax enables the effective development, and makes the code more readable.

2.3. Back-end Tool(s): Flask, AWS

Flask is used for local testing, i.e. setup a local server, and test the interactions between front-end

and back-end.

AWS provide cloud services that enables the quick deployment. The services that will be used include: S3 bucket for server, Lambda functions for handling events, API Gateway for the communication between front-end and back-end, and Relational Database Service for indexing the articles.

2.4. Version Control: Github

A widely recognized tool for version control. Github offer clean interface to trace the issues, and push/pull requests.