

## UI/Frontend decisions and options

For requirement of A2, we decide to display the UI design of our website using HTML and CSS since they have a very shallow learning curve, which even our teammate with no previous software development experience could write nice pages after learning it for several days.

For future frontend development, after finalizing our topic, we explored various approaches for building a web interface to showcase the public abstracts of our applied research internship projects. The front-end developers will use React to create a dynamic and user-friendly web interface as it is adept at managing complex user interfaces and is compatible with the Javascript and Typescript languages, which most of our teammates are familiar with. Following listed more justments on why React is chosen:

**Ease of development:** React's component-based architecture allows its ease of development. As shown in [this blog](#), React has a shallower learning curve compared to Angular, another popularly used frontend technology, due to its building of reusable UI components, which can be easily combined to form complex user interfaces, speeds up the development process, and allows easy maintenance and update.

**Availability of libraries & Maturity of the technology:** The React ecosystem is home to a vast array of open-source libraries and tools, which can be used to simplify common development tasks such as routing, state management, and form validation, as mentioned in [this blog](#). Many of these libraries have been developed and maintained by the React community for several years and have reached a high degree of maturity and stability. For instance, [React Bootstrap](#) is a widely used library that builds React components from scratch, without the need on dependency to jQuery. It has been around since 2011 and is now considered a mature and stable library but keeps evolving at the same time.

**Domains:** React is also widely used in various domains, including e-commerce, social networking, media, and healthcare. It has been adopted by many companies such as Facebook, Airbnb, and Netflix, who have built large-scale web applications using React (Olga Melnyk, 2022). For instance, Airbnb has a large-scale React application that handles millions of bookings per year.

**Performance:** React's virtual DOM and its efficient rendering mechanism help to optimize the performance of web applications built using React (Olga Melnyk, 2022). The virtual DOM is a lightweight representation of the actual DOM, which allows React to quickly compare changes to the UI and update only the necessary parts. React also uses a technique called reconciliation, which optimizes the way components are rendered by minimizing the number of DOM updates required. These performance optimizations can significantly improve the performance of web development using React.

**Cooperation with Backend:** Another main reason for choosing React is that it is compatible with Django, which is the backend technology that the other sub-team of our project team

would choose. By integrating React into the Django environment, we can create a single-page application that uses React for front-end rendering and Django for back-end API services. This strategy can result in a more interactive and smooth user experience while also facilitating better code maintenance.

## **User Story Accomplished**

As a student, while browsing the front page of the website, I came across a project that piqued my interest. I clicked on the project link to access the full details of the project. Once I finished reading, I wanted to return to the home page of the website and therefore clicked on the Home button located on the top-left navigation bar.

Requirements for the user story are: the link should take the user to the corresponding page, which contains the relevant information they clicked on. Additionally, the project page must include a Home button, which, upon clicking, should redirect the user to the homepage.

## **Individual Contributions**

Stephanie Chen has a role of a Frontend Developer and UX UI Designer. She works on HTML, CSS and javascript to develop the user interface. She also has to design the user interface and provide assistance in writing the frontend options and decisions made for the big team report.

ChengYue Zhang is a UX/UI designer. She works on writing HTML, CSS, Javascript to implement the basic features of the website, deploy the website, and notice other teammates to complete the report and work on time. Besides, she also writes the website instructions for A2 report, and make comment on the report which Ray writes to improve it.

Ray Xu has a role as a Frontend Developer and works for several tasks, including HTML and CSS development. Ray also contributes on giving advice for the UI design such as the navigation bar margin, box border, etc. In addition, Ray contributes to writing sub-team paper reports and planning on future development.

## **Instructions for web application**

The project page can be directly accessed through url:

<https://csc301-2023-winter.github.io/assignment-2-10-1-chenx608-xuray2-zhan7289/src/index.html>

The source code is listed on github repository. Simply git clone the repository using this link: <https://github.com/csc301-2023-winter/assignment-2-10-1-chenx608-xuray2-zhan7289.git> , and then click on src/index.html file to open up the homepage of the website.

## User interactions

The link above will direct the user to the home page of the website. Several faked internship projects would show up at the bottom of the page. If user is interested in one of the project and click on the project name, the user would be directed to the project page, which includes more specific details such as project abstract, organization and supervisor. Alternatively, if the user click on the link of organization or supervisor, the user would be prompt to the company's website or the supervisor's profile page (not implemented for A2).

(A)

The screenshot shows the MScAC homepage. At the top is a navigation bar with links for Home, Contact, and About. Below this is the MScAC logo. The main section contains three filters: 'Internship Year' with a dropdown set to 2020, 'Project Organization' with a 'Choose organization' button, and 'Academic Supervisor' with a 'Choose supervisor' button. Below these filters is a search bar with the placeholder text 'Enter a search term' and a 'Search' button. Under the search bar, there are two suggested search terms: 'machine learning' and 'robotics'. Below the search section, the heading '2022 Applied Research Internship Projects' is displayed. Under this heading, there are three project cards: 'Project 1' (Organization 1, Supervisor 1), 'Project 2' (Organization 2, Supervisor 2), and 'Project 3' (Organization 3, Supervisor 3).

(B)



Figure 1. (A) Image of the homepage. (B) Project box would zoom out and show blue when move the cursor to it.

At the top of the page is the navigation bar, where “Home”, “Contact” and “About” are listed. Clicking on each button would prompt to the corresponding pages for information about the website (fake information for now).



Figure 2. Navigation bar options

There are three filters for user to use. The functions of real filtering are not implemented yet, only the UI design is shown. If the user click on the filter, the available options would show up, and by clicking on one of the option, the selected option would replace the word in the filtering box to let the user know which option is chosen.

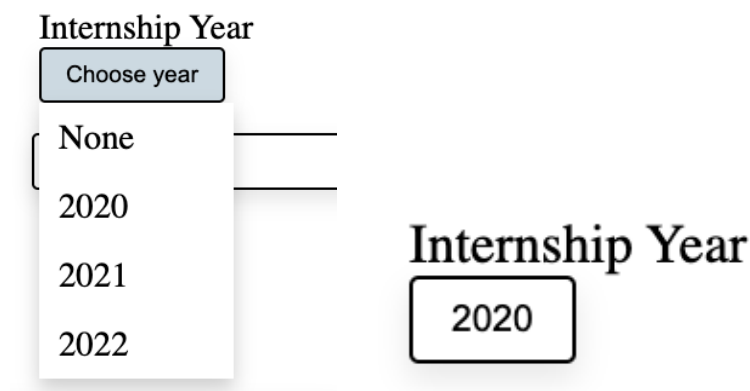
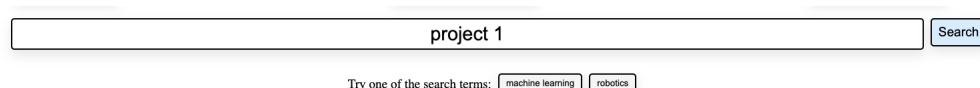


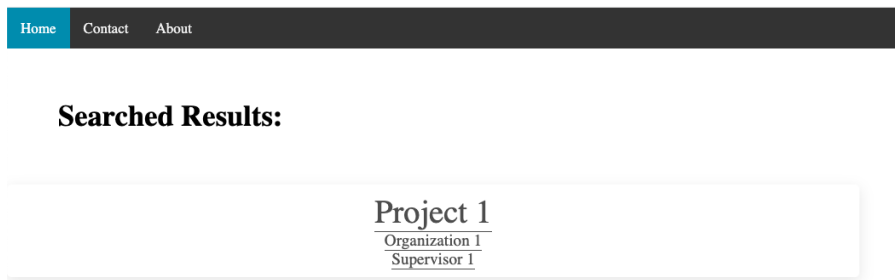
Figure 3. Options for the internship year filter. The chosen option would shown in the filter box, as shown in the figure at right side.

If the user want to search for a project by name, type the search term in the searching bar and then click “Search”. For A2 displaying purpose, it would prompt the user to the results page where the projects that includes the search term would be listed. If no results are found, the page would show “No matched results” and a “Go Back” button to return to the previous page. If the searching bar is empty but user clicked on search, a message would show up to notice the user to input a search term.

(A)



(B)



(C)



Figure 4. (A-B) Figure shows a search term of “project 1”. The result page would contain the project whose name is “project 1”. (C) A message would show up when search term is empty.

Below the search bar listed several tags for quick search. By clicking on the tag, it would automatically search for the projects that contain the tag values.

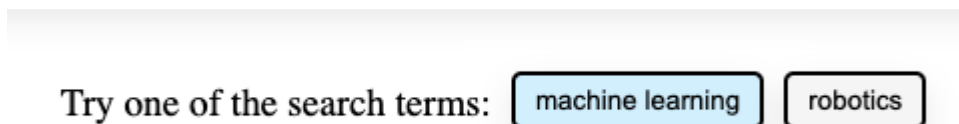


Figure 5. Tags that would show blue when the cursor is moved onto it.