IT5007 SOFTWARE ENGINEERING ON APPLICATION ARCHITECTURE

PROJECT PROPOSAL

The Research Collaborator

Aishwarya Jayagopal A0228508B

Srinivasan Kolumam Nandakumar A0228510N

INTRODUCTION

Any research project, during its lifecycle goes through various stages - the inception, problem formulation, experimentation, paper write-up, review and publication. Today, a large number of these activities are done across a plethora of platforms. Although these tools have proven very useful in the pandemic struck world, it has also introduced challenges in the form of the number of tools that are to be monitored regularly to even get an overall estimate of the project's progress.

For example, let us consider the workflow of a deep learning research project undertaken at a university:

- The initial step involves identifying a problem statement, and this involves understanding where the gap lies. This is achieved by a literature survey, in which the main stakeholders identify previous publications that have been performed in their domain of interest. The literature survey usually spans across various publishers like Elsevier, Springer etc.
- This stage is followed by the formulation of a tentative problem statement or hypothesis that needs to be proved.
- Proving the hypothesis is achieved through targeted experiments. In the research domain of deep learning, the experimentation is often done on remote servers that meet the system requirements needed to run such jobs.
- Once the results to these experiments are obtained, they are shared amongst the other members of the research team on Google Sheets or as Microsoft Excel workbooks.
- If any comments or suggestions are to be communicated within the team, it could be done through emails or in the form of comments added on Google Sheets.
- After the experimental results are found to be satisfactory, the hypothesis, the experiment and other details are drafted as a paper. This is often done on platforms like Google Docs or Microsoft Word etc.A more formal version is usually maintained on platforms like LaTeX, Overleaf etc.
- The associated proof to substantiate the hypothesis is maintained on source code management repositories like GitHub or Bitbucket.
- The resulting paper is then submitted to conferences or journals, undergoes reviews and is revised by the authors. The revision stage may involve new experiments too.
- Often the entire project needs to be split into manageable chunks, and task assignment, prioritizing and management may need tools like JIRA or Trello.

The overall progress also cannot be directly tracked, which can prove cumbersome for researchers involved in a whole gamut of projects. The various stages mentioned above can be summarized as in Figure [1].
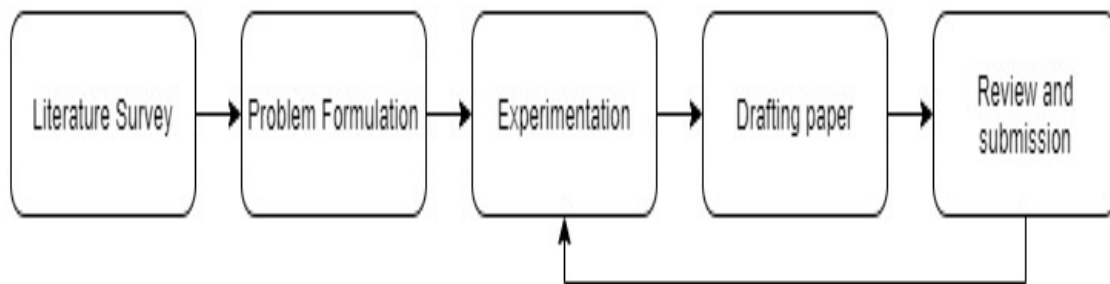

Figure 1: A general workflow followed in a research project

The Research Collaborator app is envisioned to be a one-stop solution for all stages related to a research project, from the inception to its publication. It is meant to ease the overhead that researchers often face while working on multiple projects, where they often end up distributing various pieces of the same project across multiple platforms.Through this application, we aim to integrate all these pieces and bring them all under one umbrella.

The only other application we could identify that served a similar set of end users, with the purpose of collaboration is the OSF offered by Center for Open Science. OSF is a free open source collaboration platform that is also targeted at researchers. While OSF considers projects as any form of collaboration – a paper or an experiment or a lab group, it does not consider the entire workflow that forms the basis of a paper publication process. The focus in Research Collaborator is the workflow involved in a paper publication. As a result, identifying the progress of a project in the context of this workflow is possible, which may not be obvious in case of OSF. Another key feature in Research Collaborator is the ability to create tasks under each project and assign them to specific contributors.

ARCHITECTURE AND APPROACH

Architecture

The architecture we propose to use involves hosting a database and the application running on a remote server. This server would cater to most of the data storage and business logic, with only minimal business logic implemented at the end user side, along with the display. Figure [2] shows the proposed architecture.
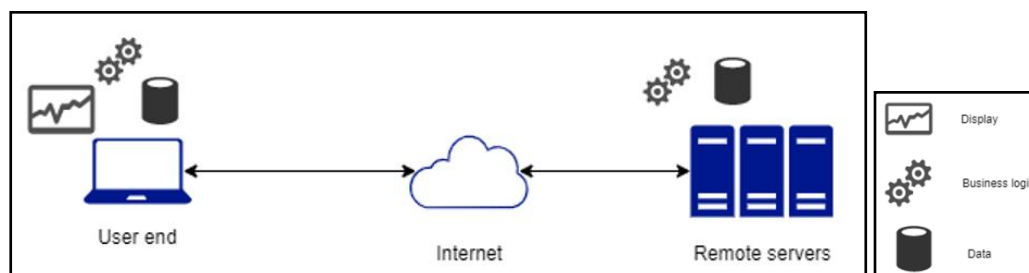

Figure 2: Proposed architecture for Research Collaborator

Approach

The largest entity in the business logic is the project. Any user can create an account with a valid email ID. The organization associated with the user can be used as a key for identification. Each user has the ability to create a new project, search for other registered users from across multiple organizations and invite users to collaborate on a project. The user who creates the project can also assign roles to the invited user. The roles available are – Principal Investigator, Research Staff, Students and Other investigators (usually for users from another organization). Each user can monitor their dashboard to view the status of all the projects they are a part of.

Organizations are only a construct used to ease the search for users and do not form a part of the hierarchy. The reason behind this is to allow even users who do not belong to any organization to register an account and collaborate with researchers spread across organizations. Figure [3] depicts the business logic and the hierarchy.
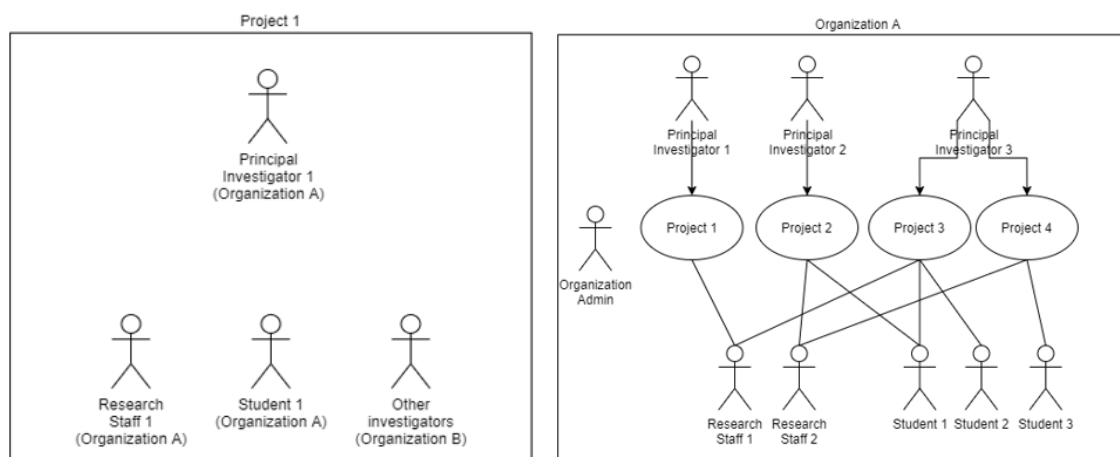
Figure 3: Hierarchy followed in the business logic

FEATURES
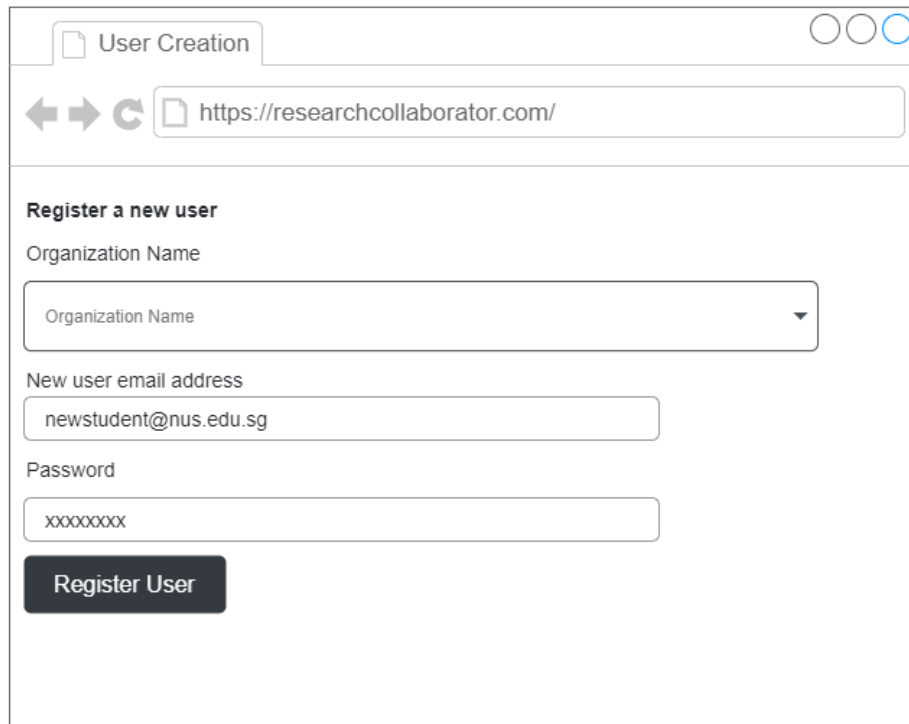
Research Collaborator has features split into three categories based on the focus area – features that do not require a user sign in, general features a user can perform and project specific features. The project specific features form the crux of the application.

No Sign-in Features:

1. User registration -
   A new user can create an account with a valid email address, and can either select an organization from an available list of organizations or as an independent researcher. Figure [4] is a mock-up of a user registration page.
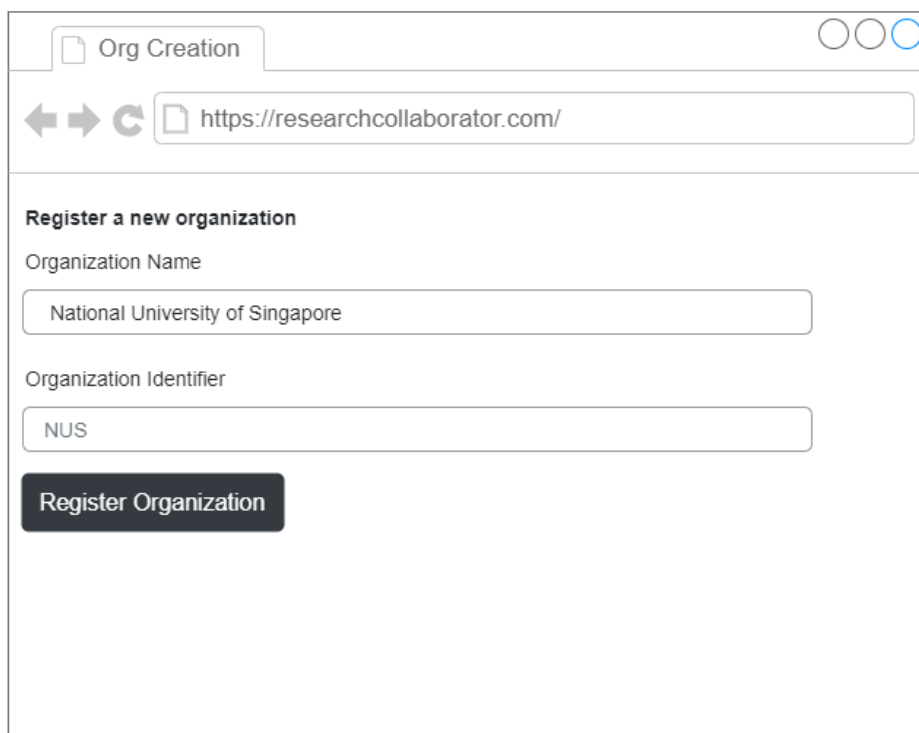
Figure 4: Mock-up of user registration page

2. Organization registration-

   This page lets the user create a new organization if it does not already exist. Figure [5] shows the mock-up for this.
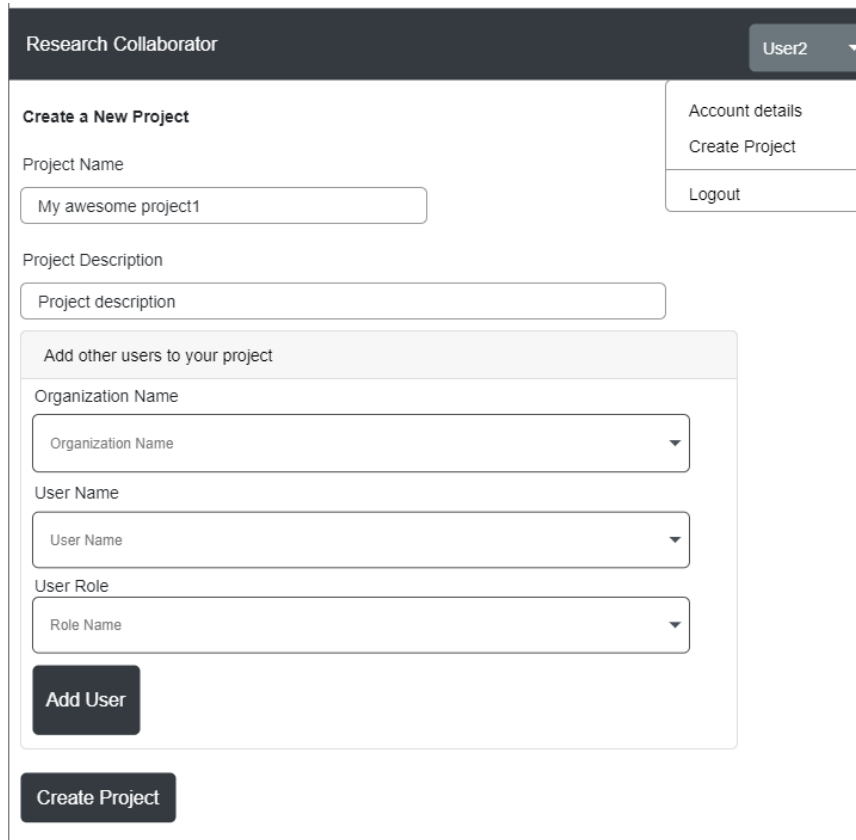

Figure 5: Mock-up of organization registration page

General User Features:

1. Project creation -
   Each user can create a project of their choice, by default the user creating the project is assigned the Principal Investigator role. The user can search for and invite users to collaborate as well. Figure [6] shows the project creation page.

Figure 6: Mock-up of projectcreation page

2. Projects Dashboard-
   The user can view the current status of all the projects they are a part of on this page, as shown in Figure [7].
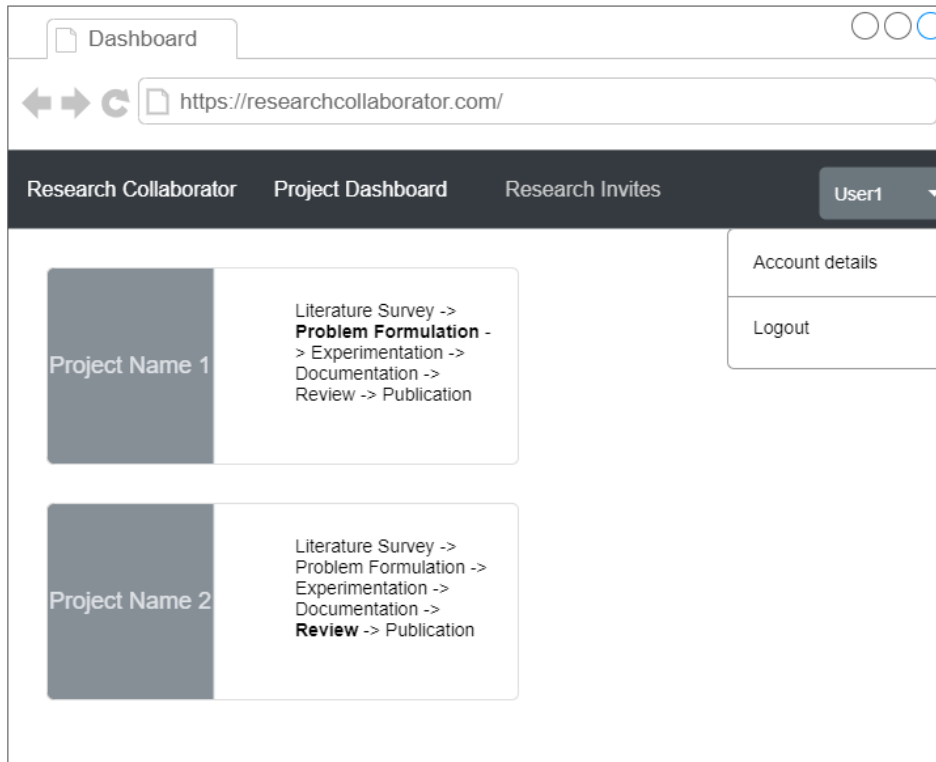
Figure 7: Mock-up of projectsdashboard page

3. Research Invites page -

This page shows a list of projects the user has been invited to participate in. Figure [8] depicts this.
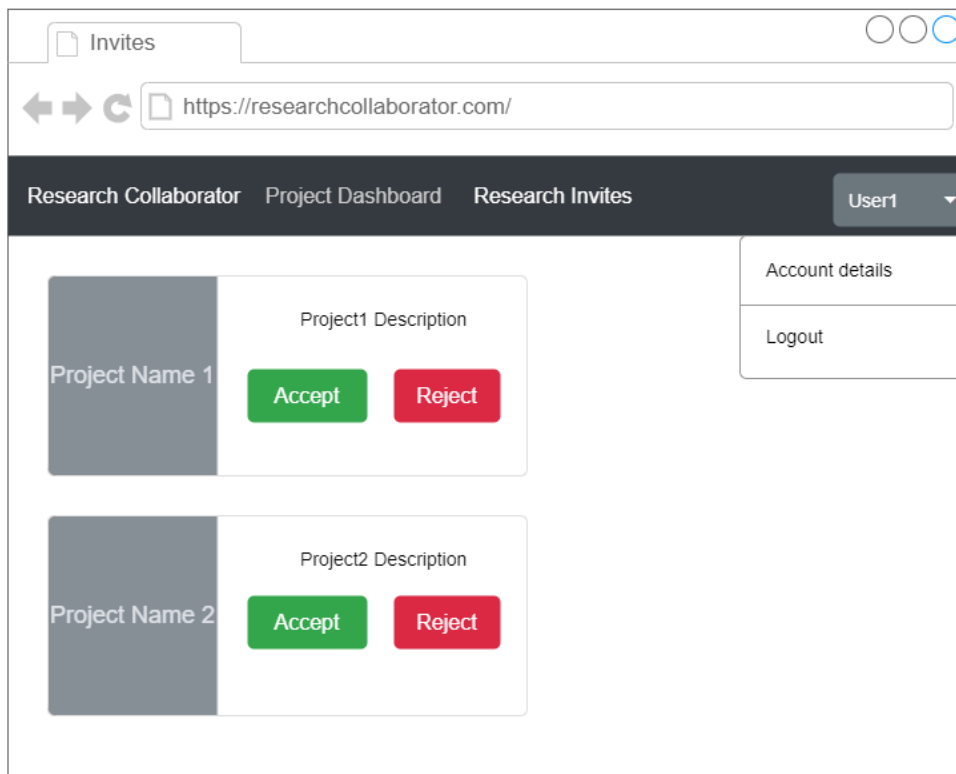

Figure 8: Mock-up of research invites page

4. Account details –
This page lets the user update their personal particulars like username, organization, reset password etc. The email address used while registering is the unique identifier for each user and cannot be modified.

Project Specific Features:

1. Project Status -
On selecting a particular project from the user dashboard, the project status can be viewed. The progress can be tracked here and can even be updated. Figure [9] displays this.
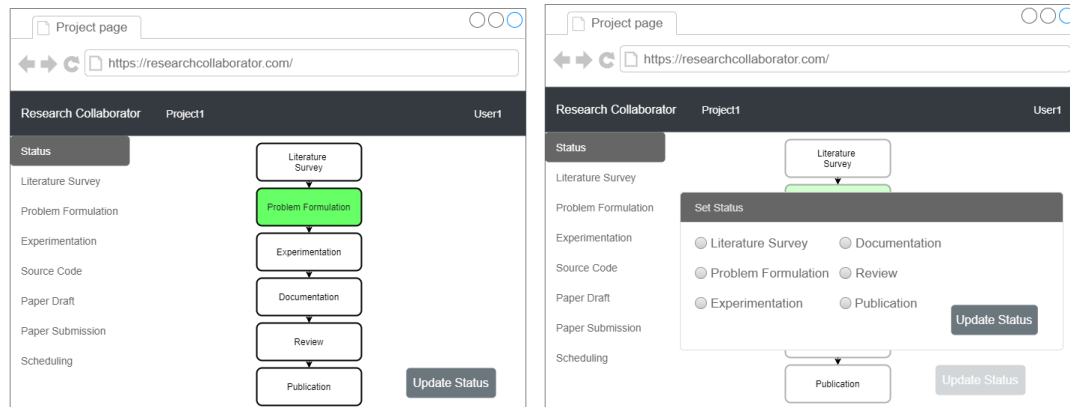

Figure 9: Project status page and update.

2. Literature Survey-
This page contains tiles that link to the publications referred during the course of research. More links can also be added by the user, as in Figure [10].
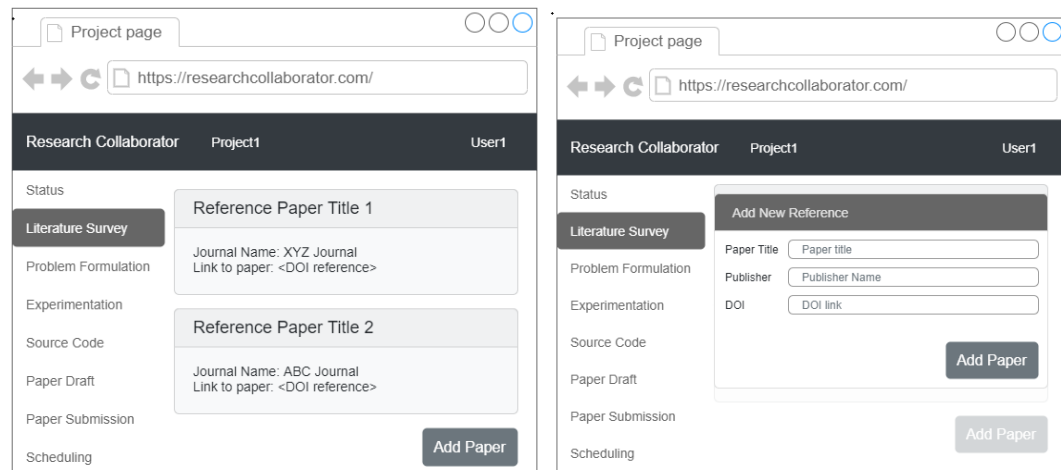

Figure 10: Literature survey – addition of papers and view

3. Problem Formulation-
This page contains a brief description akin to the abstract. This can be updated by the user. This page also allows other users to add comments, as shown in Figure [11].
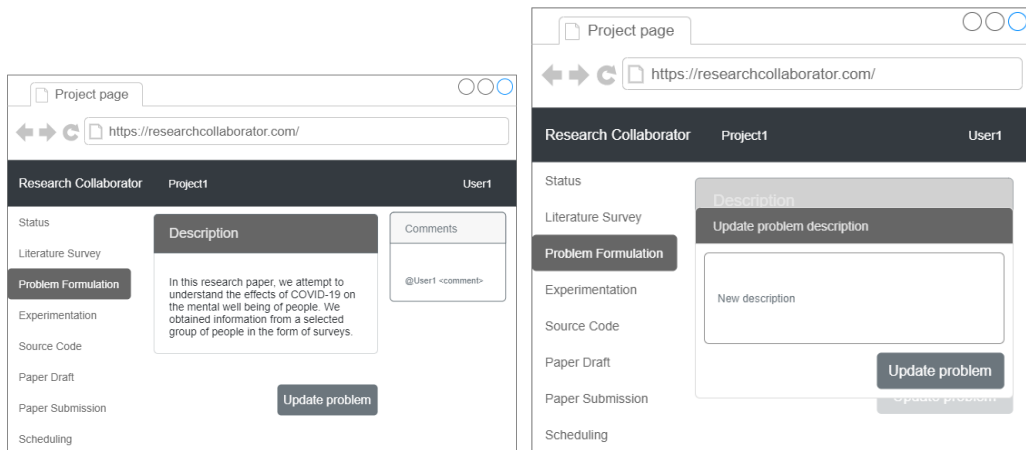
Figure 11: Problem formulation and updates

4. Experimentation-
This page contains information on the kind of experiments, their description and links to any results obtained from these experiments. It also has the option for comments as in Figure [12].
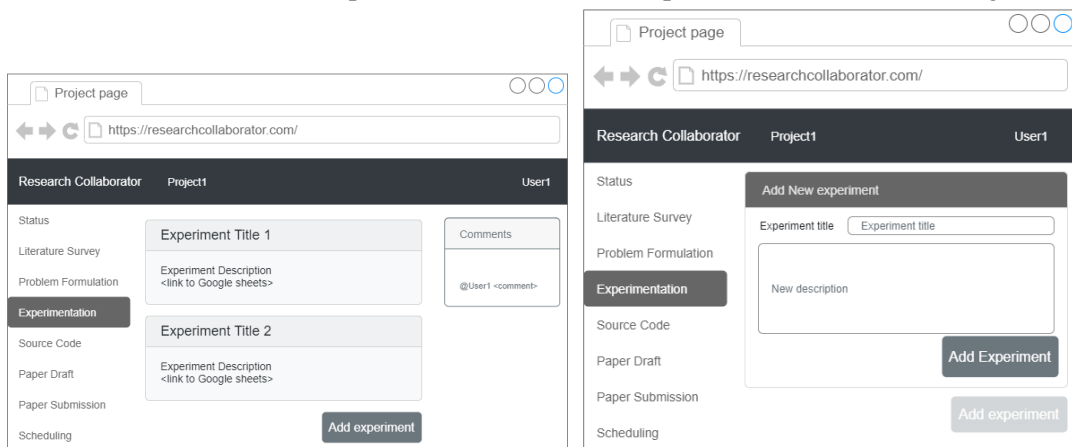

Figure 12: Experiment description and updates

5. Source Code -
This page links to the Github or Bitbucket repositories that contain the code used in the project. Figure [13] shows this.
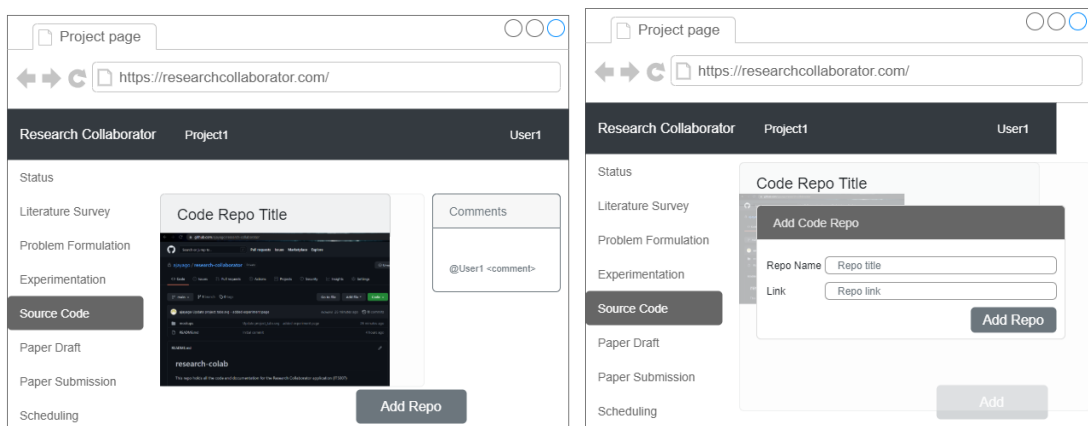

Figure 13: Code repositories and commenting

6. Paper Draft –
The initial drafts of the paper are done via Google Docs and this is linked up on this page. It also allows for user comments, as shown in Figure [14].
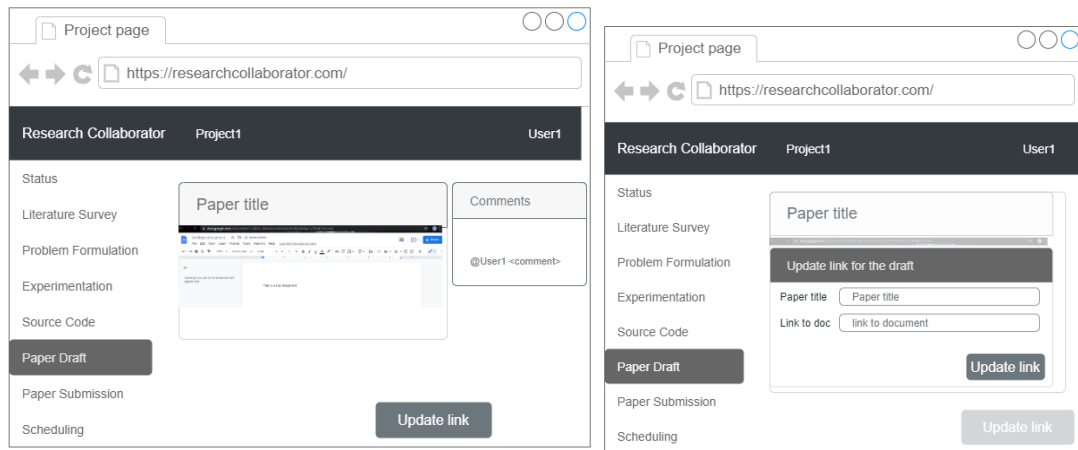
Figure 14: Paper drafting and comments

7. Paper submission -
This page can be used to link to the journal or conference where the paper has been submitted, and can be updated. This is depicted in Figure [15].
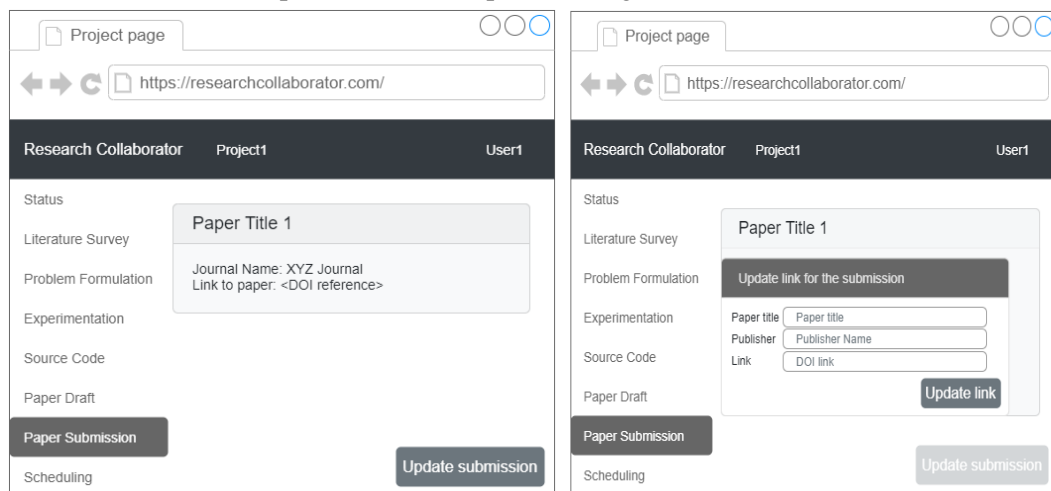

Figure 15: Paper submission and updates

8. Scheduling and task management -
This page lets users create tasks, assign them to other users and remove them when done.

CHALLENGES

A few of the challenges during the development that we anticipate are as follows:

1. Extent of integrations – Since most pieces of the project lifecycle depend on third party tools like Google Docs, Github etc, the integration with all these varied platforms would be a problem. A wide variety of tools exist that could be possibly integrated.
2. Authorization – Since a large number of integrations are to be done, ensuring the user's authentication and authorization for use of these third party tools is also a possible challenge.
3. Ensuring the compatibility of the application on devices of varying display dimensions is also a potential cause for concern.

LICENSING

We would like to use the EULA license for our product.

Permitted uses: The license for the product will be exclusive for one user.

Restrictions: Reverse Engineering of the product is not allowed.

Intellectual property rights: The ownership of the product belongs to our group and will strictly be closed source.

Other Conditions:

1. The product cannot be used for any illegal activity.
2. The ownership of the product cannot be transferred at any point of time.
3. The product cannot be rented or subleased at any point of time by the user.

SCHEDULING OF TASKS

Task Name	Week Number (Academic calendar based)
No login features (UI)	5
User specific features (UI)	6
Project specific features(UI)	7
Database setup	8
Backend development	9,10
Integration and Testing	11,12,13