



Mission Support System: MSUI UI Redesign

About me

Name - Aravind R M (Github - [aravindm711](#))

University info:

- Name - [PSG College of Technology](#)
- Major - Software Engineering
- Current year - 4th year (8th Semester)
- Graduation - May 2022
- Degree - Master of Science (Integrated 5 years)

Time-Zone - Indian Standard Time (GMT + 5:30 hours)

Resume - [[LINK](#)]

Sub-org: [Mission Support System](#) (MSS)

Contributions to Mission Support System:

Title	Issue Number	Pull Request
Mscolab export/import ignores "filepicker_default": "fs" from mss_settings.json (merged)	#636	#716
Renamed get_project api endpoints with get_project_by_id for better readability (merged)	#586	#720
Mscolab chat upload/download ignores "filepicker_default": "fs" from mss_settings.json (merged)	#638	#718
Move performance settings from dialog window to overlay docking widget for tableview (merged)	#200	#725

Project Description:

A Brief description about MSS:

Mission Support System (MSS) is a tool that helps scientists in the field of atmospheric science to plan flight routes, in which parameters of the atmosphere, like temperature and aerosol particle characteristics, are measured. A research aircraft typically carries a comprehensive scientific payload, composed of data acquisition instruments used by different companies and research institutions. These instruments help scientists measure the relevant parameters for understanding the chemical processes in the atmosphere.

MSS is currently made up of two windows:

1. **MSUI** - The Main window of the application (Fig. 1), where the users can select multiple flight tracks to work on. There are three views which are, Top view, Side view and Table view, that are used to visualize various flight tracks from different perspectives and plan the routes. The maps shown in the views use another component of MSS called MSWMS (Mission Support Web Map Service), which serves generated visualizations of meteorological predictions, on demand. MSUI is developed using the PyQT5 framework.
2. **MSColab** - MSColab window (Fig. 2), has the same features of the MSUI window and in addition lets multiple users to collaborate. This tool also provides features like real-time syncing of files across systems, chat service, and managing user privileges, and much more. The MSColab server is built using a python backend framework called Flask.

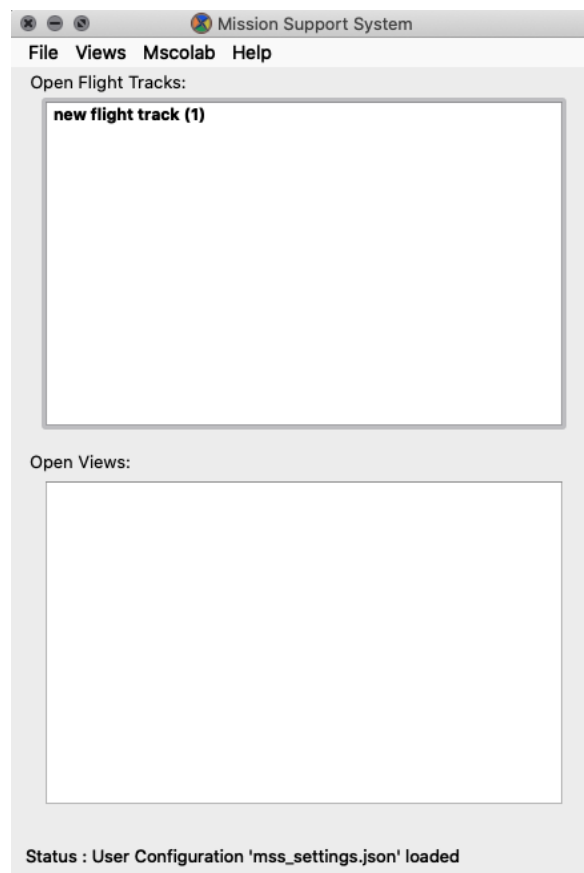


Fig.1. Current state of MSUI window

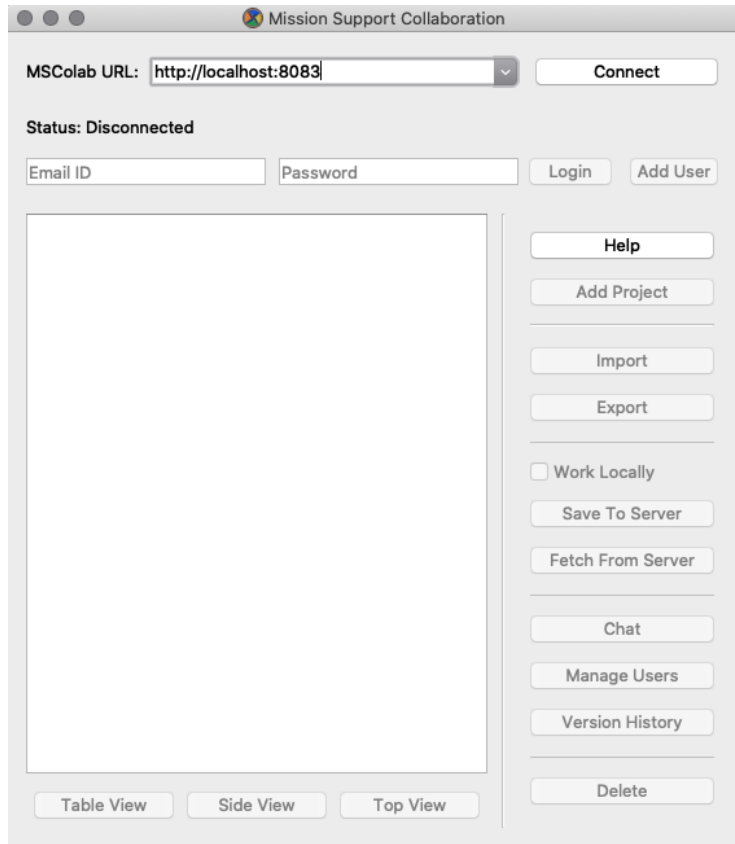


Fig.2. Current state of MSColab window

Current Issues in MSS:

1. **Two separate windows:** There are currently two separate windows, one for working locally (MSUI) and the other for working collaboratively (MSColab). Since both the windows offer the same set of functionalities to the user, having to switch between windows while working on a project becomes cumbersome.
2. **Configuration Editor:** The current window for modifying the mss settings json file contains a plain-text editor, which requires the user to have prior knowledge about the json syntax. This approach to modifying the settings file does not suit for all kinds of users.

My Proposal:

1. Two separate windows:

The goal of this idea primarily is to merge the main MSUI and the MSColab window into a single window offering the same set of functionalities in a more convenient and intuitive way. I propose a new design for the UI, which is explained in detail below.

The new design consists of three states,

- **Logged out state** - When the user wants to work locally.
 - Menu Bar contains the following menus:
 - File - “Open”, “Save”, “Save As”, “Import [CSV]”, “Export [CSV]”, “Quit”
 - Config - “Edit Configuration”, “Change from defaults”
 - Help - “Online help”, “About MSS”

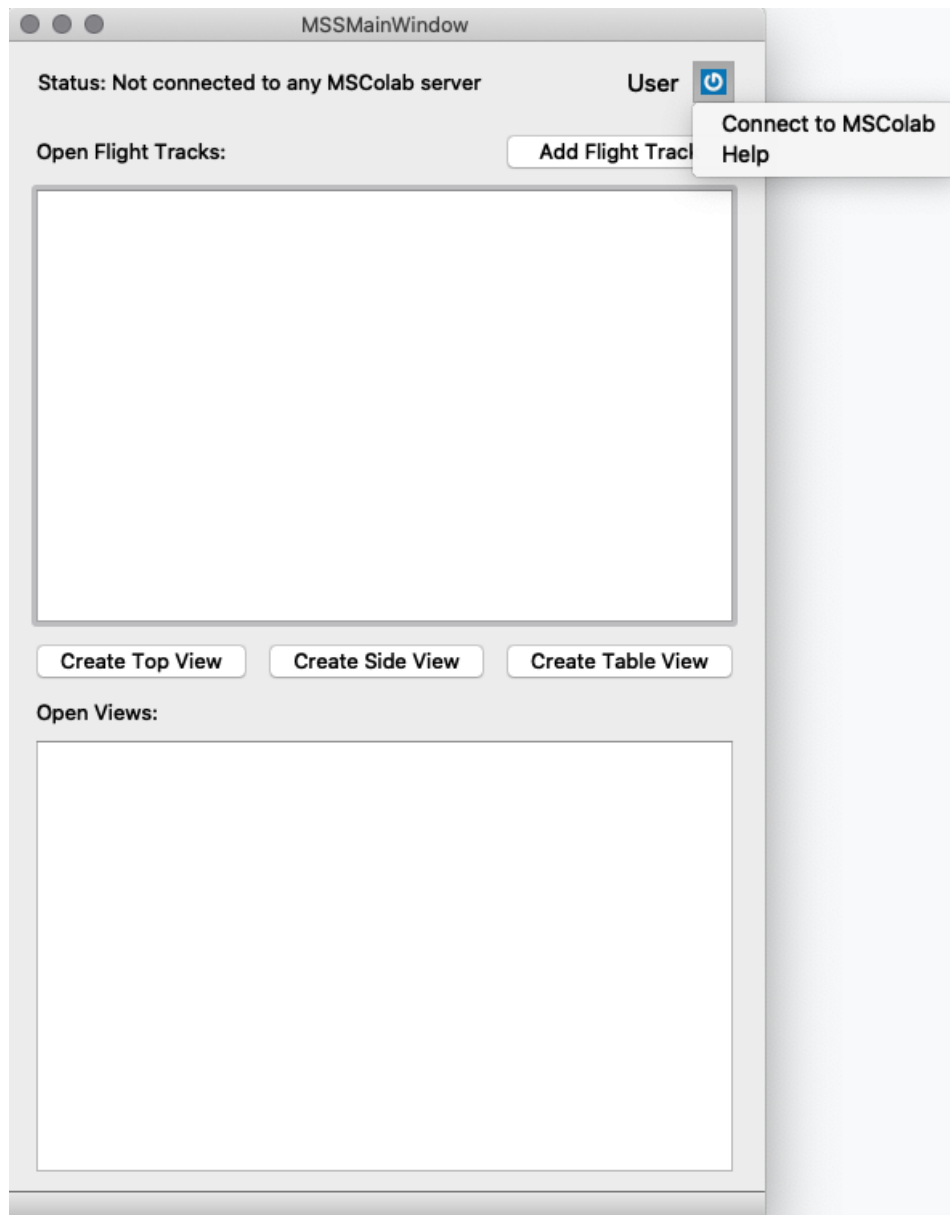


Fig 3. Logged out state

- **Connect to MSColab state** - When the user wants to connect to an MSColab server. After connecting to a server by providing an URL, the user can either login or add a new user.

The screenshot shows a window titled "MSColabConnectWindow" with the heading "Connect to MSColab Server". Below the heading is a text input field labeled "MSColab URL :" followed by a "Connect" button. A horizontal line separates this section from the "Login Details:" section. The "Login Details:" section contains two empty text input fields, a "Login" button, and a link "Click here if new user" next to an "Add user" button.

Fig 4. Connect to MSColab state - User Login

The screenshot shows a window titled "MSColabConnectWindow" with the heading "Connect to MSColab Server". Below the heading is a text input field labeled "MSColab URL :" followed by a "Connect" button. A horizontal line separates this section from the "New User Details" section. The "New User Details" section contains four labeled text input fields: "Username:", "Email:", "Password:", and "Confirm Password:". At the bottom of this section are "Cancel" and "OK" buttons.

Fig 5. Connect to MSColab state - New user registration

- **Logged in state** - The user is logged in to an MSColab server. All the MSColab options are displayed in this state.

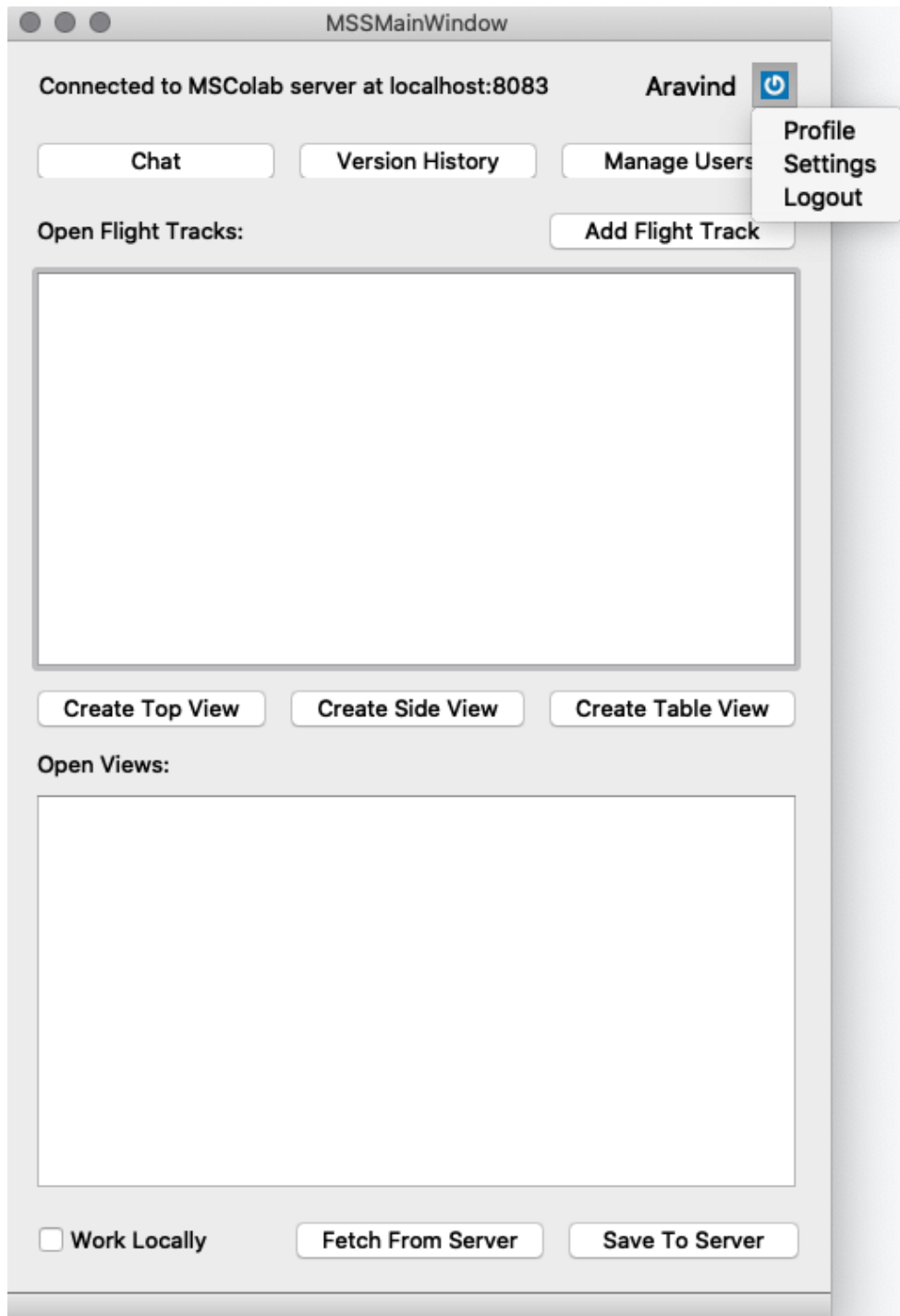


Fig 6. Logged in state

New features:

- Profile page to display information related to the users. This window also includes other options like setting gravatar profile image and delete account.

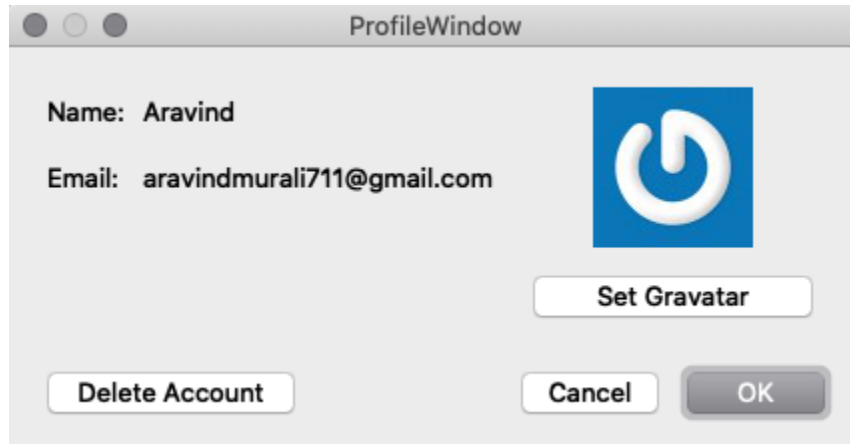


Fig 7. User profile page

- Right-clicking on a flight track from the "Open Flight Tracks" pane would display a drop-down for renaming, closing the flight track.

2. Configuration Editor:

I propose to design a new editor interface which will provide an intuitive approach to editing json files.

- Current Window

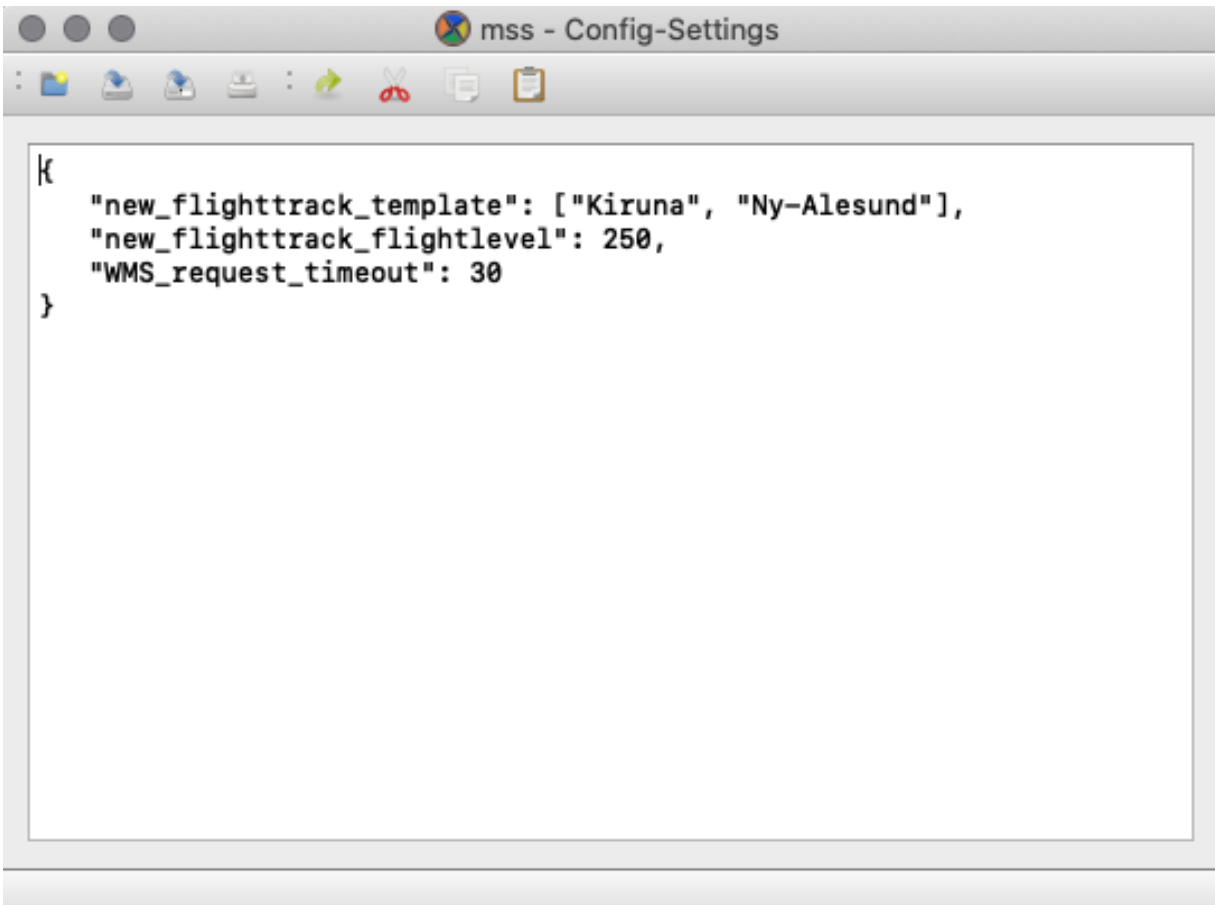


Fig 8. Current config editor window

- Proposed Window

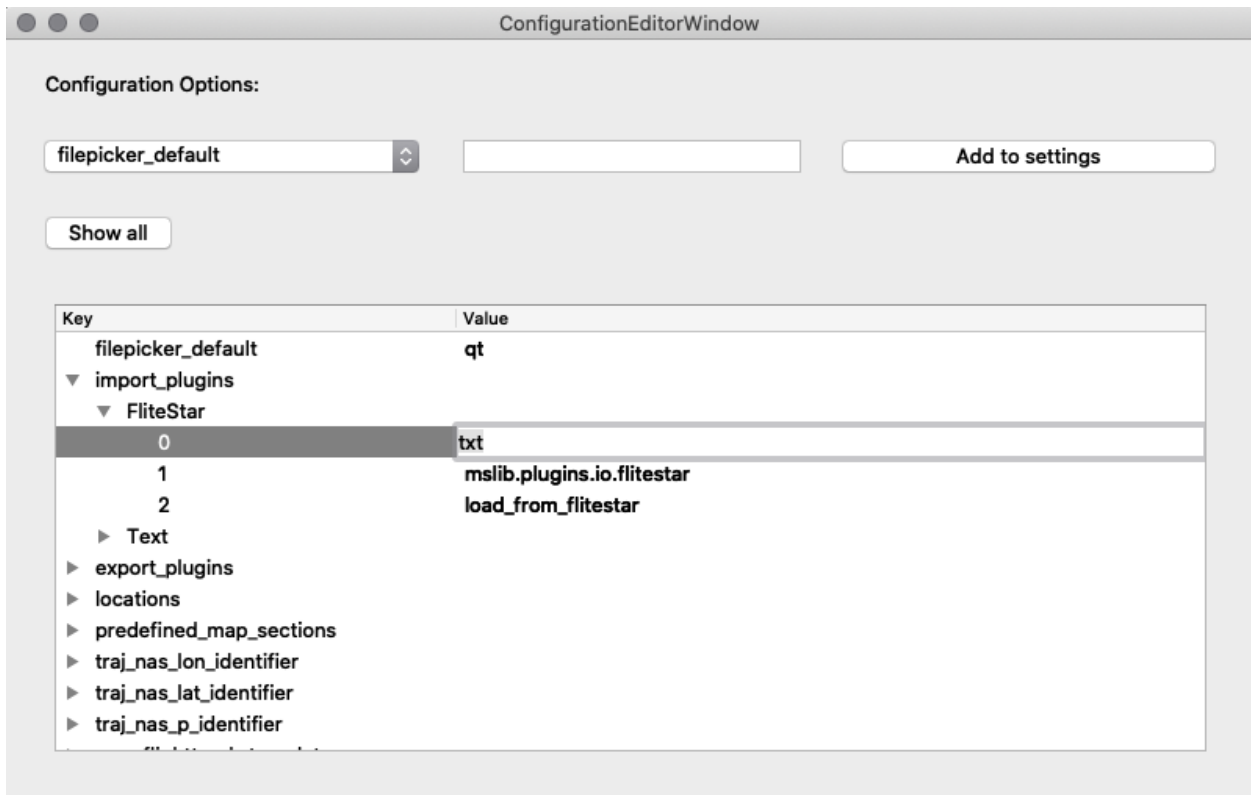


Fig 9. Proposed config editor window

- This feature would be implemented using the [qt-json-view](#) library.
 - Also solves issue [#634](#).
- New features:
 - Combobox to display all the available configuration options.
 - Option to display description of the selected key.
 - Options to filter, sort.
 - Load settings from another json file.
 - Save only the changes that have been made from the default settings.

Project Timeline:

Timeline	Work
May 17	Student Projects Announced
May 17 - June 7	Community Bonding <ol style="list-style-type: none"> 1. Deep-dive into the code base. 2. Set up the development environment. 3. Finalize the design.
June 7 - June 11 (Week 1)	<ol style="list-style-type: none"> 1. Redesigning MSUI window with appropriate layouts and widgets. 2. Examine the differences in feature implementation between MSUI and MSColab.
June 14 - July 2 (Week 2 - 4)	Implementation of the new design <ol style="list-style-type: none"> 1. Refactoring underlying logic for proper functionality of the new UI. 2. Migrating MSColab features to the new window. 3. Development of the user login page. 4. Implementation of the new user profile page.
July 5 - July 11 (Week 5 - 6)	<ol style="list-style-type: none"> 1. Refactoring of tests for the new design. 2. Updating documentation in online resources.
July 12 - July 16	<ul style="list-style-type: none"> ● Phase I Evaluation
July 19 - July 30 (Week 7 - 8)	<ol style="list-style-type: none"> 1. Implementation of PyQt Json Editor 2. Adding new features
August 2 - August 6 (Week 9)	<ol style="list-style-type: none"> 1. Refactoring and adding tests for the new configuration editor 2. Document code
August 9 - August 13 (Week 10)	<ol style="list-style-type: none"> 1. Examining and fixing bugs. 2. Further improvements, if any. 3. Extending test scope.
August 16 - August 23	<ol style="list-style-type: none"> 1. Submission of work, project summaries 2. Final evaluations of mentors
August 23 - August 30	<ul style="list-style-type: none"> ● Mentors Submit Final Student Evaluations
August 31	<ul style="list-style-type: none"> ● Final Results Announced

Communication:

- I'll be available through slack/mail during the regular working hours of the day.

Commitments:

- If selected for GSoC, I'll be dedicating my time entirely for this project as I have not applied for any other organization.
- Despite my university reopening on June 16th, I'll be working full-time and also keep up with the project timeline.