

UI-improvements

Python Software Foundation (Mission Support System)



About Me

- Name - Shubh Gaur
- IRC/Slack: Shubh Gaur (sratslla)
- GitHub - [sratslla](#)
- Email - sratslla@gmail.com
- University - Jaipur Engineering College & Research Centre
- Program - Bachelor of Technology in Computer Science
- Year - 3rd year
- Timezone - Indian Standard Time (GMT +5:30)

Background

As a junior pursuing a Bachelor of Technology Degree in Computer Science and Engineering, I have a passion for crafting and a deep interest in web development. Throughout my first year of college, I experimented with different development skills and discovered my love for creating unique UI designs and incorporating animation libraries to provide a calming user experience. Over the past year, I have gained expertise in various web development skills, including HTML, CSS, JavaScript, and the MERN stack. I have successfully built several full-stack web applications.

In addition to my web development skills, I have developed Python proficiency through my coursework and personal projects. As a tech enthusiast, I am constantly exploring new ways to improve my skills and expand my knowledge. I am excited about the opportunities that lie ahead in the field of technology, and I am eager to contribute to innovative projects that make a meaningful impact.

Logistics

- The GSoC timeline is in sync with my university's summer break (June end) and thus it will give me enough time to work on this project. I understand that GSoC is equivalent to a full-time program and hence I plan to devote at least 30-35 hours a week to this project.
- My university reopens in mid-August and even if some part of the timeline coincides there will be no exams in that period, allowing me to devote ample time to work on and complete this project in the stipulated time frame. I am excited to spend my summer working on this project!

Issues Resolved

S No.	Contribution	Issue	PR
1	pin werkzeug to >=2.2.3	#1687	#1688
2	Show results on the ctrl+f highlighter without clicking*	#1521	#1696
3	Dialog Box showing opposite order of Yes-No	#1076	#1711
4	Shortcut window acts like Search window if opened after Closing Search Window	#1733	#1734

Issues Reported

S No.	Issue	Link
1	Multiple Flightpath option resulting in an error	#1701
2	Shortcut window acts like Search window if opened after Closing Search Window	#1733

Project Information

- Organisation Name - Python Software Foundation
- Sub-Organisation Name - Mission Support System (MSS)
- Mentor - Reimar Bauer, Joern Ungermann
- Project Size - Large (350 hrs)
- Operating System - Ubuntu

Project Abstract

1. Modernize fs_filepicker:

- The fs_filepicker is a GUI used for selecting files in the msui_settings.json configuration file. Currently, the GUI only offers a basic set of navigation elements for selecting a file and returns the fs URL for accessing it. The goal of this project is to modernize the GUI using the Pyfilesystem module to make it more user-friendly and efficient.
- Pyfilesystem is a Python module used for accessing different types of file systems. It provides a common API for working with various file systems, such as local, remote, and cloud-based storage systems. The Pyfilesystem module is easy to use and has many features that can be used to improve the functionality of the fs_filepicker.

2. View Layout and Restoring::

- The mscolab project is a collaborative software platform that enables multiple users to work together on a shared operation in real time. The project is written in Python and uses the PyQt GUI library for the user interface. Currently, the PyQt Gui of the MSS client can handle different views by one flight path. However, once a new flight path is loaded and activated, all views change to this flight path. This is inconvenient for users who want to work on multiple flight paths simultaneously. Additionally, the view configuration consists of many windows with a

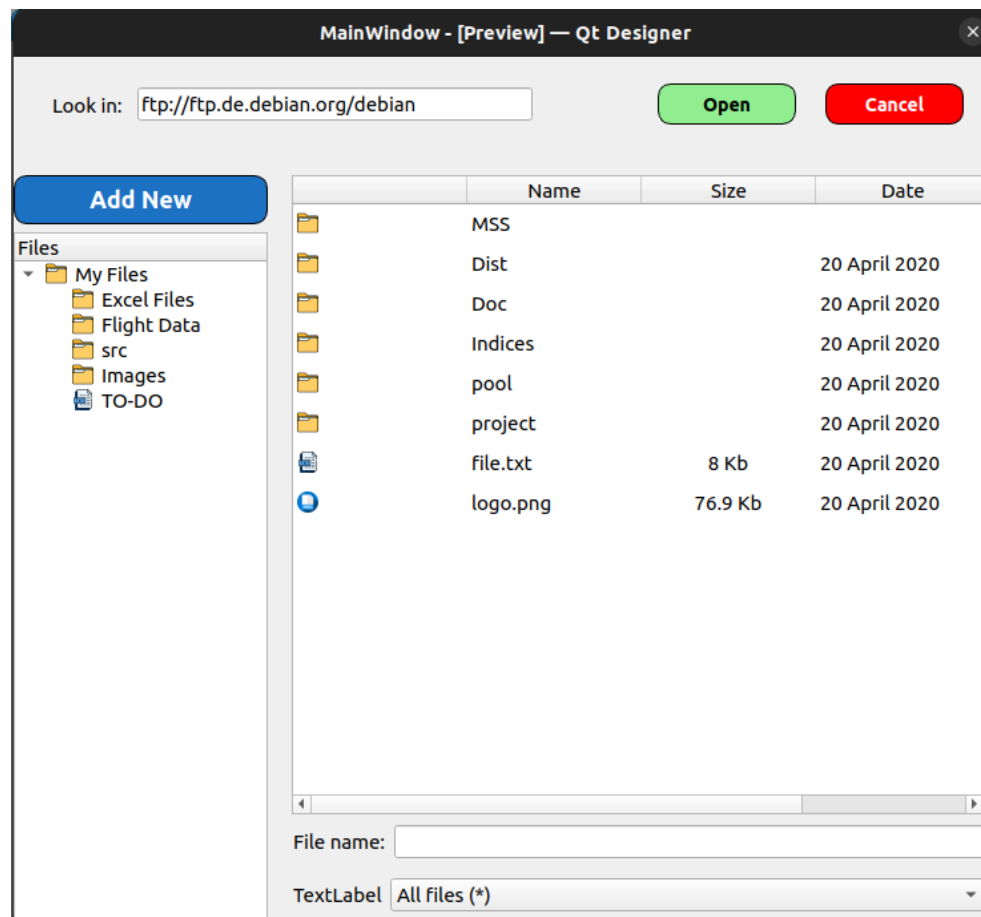
complex set of configuration options that are tedious to re-create after shutting down the application.

- This proposal aims to improve the mscolab user interface by adding the ability to store and restore the view configuration of multiple windows for the user. Additionally, the proposal seeks to add a layout option for any participant on the same flight path, as well as the ability for the creator of an operation to set the layout of all participants.

Detailed Objectives

1. Modernize fs_filepicker:

- Upgrade the fs_filepicker to have a modern look and feel. This will make it more user-friendly and attractive.

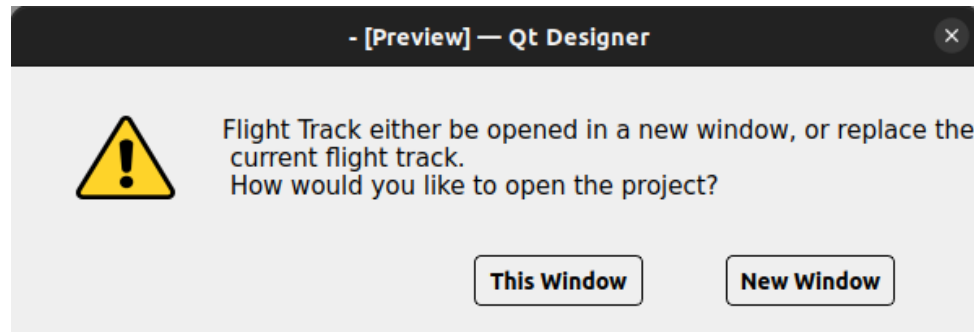


- Implement non-essential methods of Pyfilesystem to enable more advanced functionality. Pyfilesystem has many non-essential methods that can be used to improve the performance and functionality of the fs_filepicker. For example, the copydir method can be used to copy directories between file systems, and the copy method can be used to copy files.
- Implement a directory view as a tree. Pyfilesystem's common API allows for the implementation of a directory view as a tree where folders or files can point to different storage locations. This feature will make it easier for users to navigate through the file system and select files.
- Write tests to cover new source lines. It is important to ensure that the modernized GUI works as expected and does not introduce any new issues. Writing tests to cover new source lines will ensure that the code works correctly and prevent any potential issues that may arise in the future.
- Provide documentation that explains how to use the new features and functions.

2. View Layout and Restoring:

- To implement the view layout and restoring feature, the mscolab UI will need to be modified to allow for the creation and saving of different view configurations. This feature will require the addition of new UI elements such as buttons and dialogues to facilitate the saving and loading of view configurations.
- To save the window's size and location, one option is to make use of a Python module such as pyautogui. With this module, we can leverage its getWindowsWithTitle() method to retrieve a list of all the currently opened windows on the system, and subsequently, we can iterate through this list to extract the relevant location and size information and save it to a configuration file.

- Once a user imports a new flight path, a pop-up dialogue box can appear with two options. The first option is to continue on the same window, which means the new flight track will be added to the Flight Tracks list on the main MSUI window, as the application currently works. The second option is to open the flight track on another MSUI window, which will save the user from having to manually start MSS multiple times.



- The feature will also require the creation of a new data model to store and manage the view configurations. This data model will need to be designed to allow for the storage of multiple view configurations per user, and for the sharing of view configurations between users.
- To enable the saving of user-configured data to the server, the msui-core.conf can be stored on the server, along with information about which views are opened and which server was selected. The data can be sent to the server when the user logs out or when it is explicitly sent.
- The feature to enable sharing of saved view configurations with other users can be implemented by adding a dialogue option that allows the user to select the view configuration they want to share and with whom they want to share it. This would require a socketio event implementation for the user to receive the configurations, which would prompt the user to accept now, accept later or reject the shared configurations through a UI feature. In addition, it could be useful to include an option to save the configuration locally for future use. It is possible that the chat feature in mscolab could be used as a platform to implement the UI for sharing

saved view configurations, or alternatively, a similar approach to sharing access to an operation could be used.

- The creator of an operation should have the ability to set the layout of all participants to ensure that everyone is viewing the project in the same way. This feature would require a "force" socketIO event on the server side, and an event on the client side that loads the shared configuration. It may also be useful to include an option for the user to disagree with the shared configuration in the `msui_settings.conf` file. The default setting for this option could be a nudge to agree, as maximum collaboration with the operation's owner is typically desired.

Weekly Timeline

Timeline	Work
Pre GSoC	<ul style="list-style-type: none"> → Contribute to the project by working on issues to further improve my understanding of the codebase. → Interact with the mentors and other contributors.
Community Bonding Period (May 4- May 28)	<ul style="list-style-type: none"> → Bonding with the community actively. → Discussing and refining the project idea with the help of the community and the mentors.
1st Week and 2nd Week (May 29- June 11)	<ul style="list-style-type: none"> → Create the new modern UI for <code>fs_filepicker</code>. → Integrating non-essential methods of <code>Pyfilesystem</code>.

3rd Week and 4th Week (June 12- June 25)	<ul style="list-style-type: none">→ Implement a tree-based directory view.→ Write tests for the new feature and improve them where they seem to be lacking.
5th Week and 6th Week (June 26- July 9)	<ul style="list-style-type: none">→ Implement the basic UI components for the view layout and restoring feature.→ Design and implement the database schema for storing user-configured data on the server.
7th Week and 8th Week (July 10- July 23)	<ul style="list-style-type: none">→ Implement the saving and loading of view configurations.→ Test the saving and loading of view configurations to ensure that they are working correctly.
(July 10- July 14) Mid-term evaluation	<ul style="list-style-type: none">→ Submitting mid-term evaluations.
9th Week and 10th Week (July 24 - August 6)	<ul style="list-style-type: none">→ Implement the sharing of view configurations between users.→ Create an option for the creator of an operation to set the layout of all participants.
11th Week and 12th Week (August 7- August 20)	<ul style="list-style-type: none">→ Add the ability to export view configurations as templates that can be shared with others.

	<ul style="list-style-type: none">→ Test and evaluate the system for its ability to store and restore view configurations, and to allow participants to set their view layout.
Final Week (August 21 - August 28)	<ul style="list-style-type: none">→ Buffer week.→ Fix bugs if there are any.→ Improve tests and documentation where they seem to be lacking.
Post GSoC	<ul style="list-style-type: none">→ Continue contributing to the project by fixing issues and adding new enhancements.

Eligibility

- Yes, I'm eligible for Google Summer of Code 2023

Are you applying for other Projects?

- No, I am applying only for this project.

Communication

- You can reach me via Slack, email, or phone calls.
- I will provide frequent updates to my mentor(s) regarding the project's advancement and any obstacles I may face.
- Weekly blog posts will be written by me to document my project's progress.