Google Summer of Code

Proposal – Google Summer of Code 2023

Retroshare

Retroshare Web Interface (RSNewWebUI)

By Sumit Kumar Soni

About me

Name:	Sumit Kumar Soni
Email:	sumitkumarsoni123@gmail.com
Retroshare/IRC:	zelfroster
Links:	[Portfolio](#), [GitHub](#), [Twitter](#), [LinkedIn](#)
Resume:	[Resume](#)
Country/Region:	New Delhi, India
Timezone:	Asia/Kolkata (UTC+05:30)
Course:	Bachelor of Computer Applications
University:	Indira Gandhi National Open University

Why Retroshare? : Motivation

I was looking for internship from the beginning of this year and had heard about GSoC a lot from one of my friend [Mayank](#). Also, I have prior open source experience So, this was a perfect opportunity for me to work on something exciting.

I started looking for organisations, selected a few projects which I planned to work on but when I found about Retroshare Web Interface project, I felt like this was something I could do and, It will provide me opportunities to learn and grow my skills. Furthermore, I was able to spot a few issues and after giving sometime to understand the working of the project, I was able to fix those issues.

I got excited about this project as its ideology is well aligned with my field of interest. I will keep contributing to this project even after the GSoC period is over. The support I got from the mentor and fellow contributors was amazing so, choosing this project was natural.

Project Title

Implementation of remaining Web Interface of Retroshare.

Project Context

RetroShare provides a decentralized, encrypted connection with maximum security between nodes where they can chat, share files, mail, etc. Retroshare uses GXS (Generic eXchange System) that provides Asynchronous distribution, authentication, privacy, security of generic data. Retroshare is available on all platforms like Android, Linux, macOS and windows.

RetroShare is a C++ software program that comprises a headless lib called "libretroshare." This lib can be utilized in different contexts such as a headless server (retroshare-service), a standalone app with a user interface built using Qt, or an android client. Moreover, a web interface which has started being developed recently to enable users to control the headless server from their web browsers. The web interface communicates with libretroshare using a JSON API that is automatically generated and contains all necessary functions for sending and receiving data from the software.

Project Goal

The aim of my involvement in this year's GSoC program is to create the remaining interface of the Retroshare WebUI so that it has all of the necessary features which the main Retroshare app has and, users are able to fully use the web interface as more of a standalone app.

There is a list of features which have to be developed such as creating panels for software configuration, managing shared files, directories etc and also fix the existing ones such as Files, Mail, Forums, Channel section etc.

Mentor : @csoler

Pre-GSoC Involvements

Contributions to RSNewWebUI :

- #61 – fixed issues on Homepage :
  - fixed certificate textarea so that it is not behaving weird.
  - added cursor pointer on buttons for better feedback.
  - separated fields in add friend section and made it intuitive.
  - implemented one click select on certificate field.
  - centered popup cards respective to the viewport.

- #63 – improved sidebar ui closing issue #62 :
  - gave the sidebar a modern look.

- #66 – fixed some more issues suggested by defnax :
  - implemented a button to show more/less number of recipients when there is a large number of recipients.
  - fixed date was getting added to wrong field when no author.
  - made input password field autofocus on the login page.
  - implemented sender avatar in mail view and people section.

- #67 – created attachment section in the mail tab
  - all of the mails which have an attachment will be visible directly in this section.

Currently working on implementing more features in the mail tab as suggested by the members and mentors.

Some of my previous open source contributions in other projects :
- [#154](#) – revamped the whole readme section and added instructions for missing dependencies.
- [#12](#) – fixed docs for the build section.

To Do List : Milestones

Project Milestones :

A) Implement panels for software configuration.
  - Create new panels in the config section.
  - Fix existing panels in the config section.

B) Implement panels for configuring and visualizing own shared files.

C) Implement the display of forums.
  - Fix/Implement visualizing of messages in the posts.
  - Improve the Layout and usability of forums.
  - Fix the UI of Posts and comments.

Additional Milestone :

D) Webui todos.
  - Implement Reply, Reply All, Forward feature in Mail View.
  - Implement the remaining feature in Channel and Boards Section.
  - Fixing the CSS structure and Layout of the project.
  - More options in the Config Section.

The additional milestones are the some important features which are much needed. So, I will try to implement some of them during the Community Bonding Period and some during the GSoC period.

Approach

Retroshare automatically generates the boiler code to deal with JSON API from comments using Doxygen. All the routes are present in the libretroshare/src/retroshare folder in the Retroshare repository. After checking the files inside that folder, We can understand how many routes are available, what are they used for, their return type as well as the arguments they require.

After knowing the information about the routes, I will do the API calls using mithril and output their return type on the Web page accordingly using HTML, CSS and mithril.js. However, some API calls have some require headers (such as turtleSearch for files) and to do that I will discuss with the mentors about it and will implement them afterwards.

I will try to keep the progress in check with the timeline proposed and will be in touch with the dev team. If I get stuck on a problem for a long time then I will ask the mentor for guidance and will learn and try to solve the problem.

Timeline

Duration	Milestone
April 4, 2023 – May 4, 2023	Application Review Period

April 4, 2023 – May 4, 2023	• Get more familiar with the codebase and develop a better understanding of the working behind it. • Learn the core concepts of Mithril.js such as Lifecycle Methods, Autoredraw system, vnodes, Components etc. • Building test components with Mithril.js to create a concrete understanding of it. • Solve existing bugs and issues in the project and improving/implementing mini features. • Discuss other project approaches with mentor and members/contributors.
May 4, 2023 – May 28, 2023	Community Bonding Period
May 5, 2023 – May 28, 2023	• Fix working and UI of Channel section and Boards Section. • Refactor overall Layout for easy creation of further components. • Fix CSS code structure and remove duplicate and obsolete code. • Deciding Designs for the sections with members or Making a specific Design for it in Figma.

May 29, 2023 – August 21, 2023	**Coding Period Start**
May 29, 2023 – June 11, 2023	• Implement Mail panel in Config section with custom tag creation feature. • Fix existing panels in Config section and add necessary missing features.
June 12, 2023 – June 26, 2023	• Implement Other panels which are specific and important to the Config section. • Resolving all bugs and issues that arrises while implementing these features.
June 27, 2023 – July 10, 2023	• Fix Compose mail popup in Mail section. • Implement Reply, Reply All and Forward Feature in mail.
July 10, 2023 – July 14, 2023	**Midterm Evaluation**
July 14, 2023 – July 28, 2023	• Implement remaining features in the Forums section and fix the existing bugs. • Make the UI of the forum similar to the main app version with better Layout of posts and comment view/reply features.

July 29, 2023 – August 21, 2023	• Implement panel for configuring and visualising own shared files with features for managing shared directories, user permissions etc. • Fix display of Boards and Channels section. • Fix all of the bugs and issues during working.
August 21, 2023 – August 28, 2023	Submit Code and Final Evaluation
September 5, 2023	Results Announced

More About me: Who am I?

I am Sumit Kumar Soni, a Frontend developer who loves linux, design and pushing boundaries. I love creating websites with not just functional and responsive ones but also beautiful and interactive ones which last a long impression on the users. I love the ideology of open source and try to contribute as It enables me to gain real world experience and exposure as well as helps in making the world a better place. I am currently exploring opportunities where I get to learn, thrive, gain experience and build interesting stuff.

Experience :
- I worked as a Frontend Intern at InditechIT where I learnt the fundamentals of web development. And created mini projects and built a website for a business.

- I have [won](#) Hackathons at Major League Hacking.

- I have also volunteered at Devsnest where I was a Frontend Mentor and took doubt sessions on basics of Web development. I also solved doubts of student regarding challenges on Devsnest platform.