

# Abstract

Implementation of the following features for p5-web-editor-

1. Find/Replace.
2. Limitation on upload of files to S3.
3. Refactor server-side code using generators to avoid callback hell.
4. Add library reference to the editor.
5. Social media signup/login.

## Find/Replace -

To implement this, find and replace would have to be implemented within Editor component.

Options provided in the Find/Replace dialog would be

1. Find
2. Replace
3. Replace all
4. Other options
  - Case sensitive/insensitive
  - Regex
  - Only in selection(find replace only in selection)

The design of the modal would be similar to what is seen in the Processing desktop editor.

## Limitation of file upload to S3 -

If there is no limit on the file upload, many malicious users would upload their content to p5 buckets. To implement this, the following changes will be made to database-

1. "size" parameter will be added to file schema. This will store the size of the file
2. A "totalSize" parameter will be added to project schema. This will store the size of all the files in the project.

These parameters will be updated each time a file is uploaded in a project. Thus using these a limit will be imposed on the user for maximum size of file uploads. Because of the size parameter inside file schema, we can even present the user with a view in which all the file's size are shown and the user can delete the files to free space.

## Refactor server side code using generators -

Generators are used to make async code look serial. As more functionality is being added to the controllers more callback hell are being introduced.

To this "co" module will be used along with generators and the functions would have to be rewritten to return Promises.

## Add library reference to the editor -

For the users that are using the editor, it would be a huge help if a reference box would pop-up when they use a method with the documentation of the method. This is a big task to do completely, hence would be done in phases. First phase would be to just give a link to the reference of a method when a user uses it. Second phase would be to show the documentation in a hover box.

This feature will also lay the foundation for auto-complete/suggest.

## Social media signup/login -

Though I have added google singup/singin, a lot is still to be done for a seamless user experience like

- Fix github signup/login issue.
- Handling the case when the user decides to change his email.
- Verifying if the email is correct before signup(not the format, but if the email is valid using node email-check).
- Navigating to respective social media platform when the email entered in the username/password form is linked only to a social account.

These feature would mostly require tweaking passport and the user.controller.js code.

## Expected Results

By the end of fellowship, all the features would be working and well documented in the p5-web-editor. I would be maintaining these features in the future as well. The individual results of the features are already mentioned.

## Impact of Project

As p5-web-editor is the next big thing from The Processing Foundation, addition of these features will make the experience very smooth for the users of the editor. The upload limit feature will ensure that no malicious user is exploiting the free service offered by the organization and the code refactoring task would make the editor code much more readable and easy to contribute to. The contributors would be easily able to figure out the logic of the code and would not have to waste hours in debugging the callbacks.

## My Profile

I have been contributing to Processing for about 3 years now. I am quite familiar with the technology that p5-web-editor uses. MongoDB, Express, Redis, React, S3, GCP, Elasticsearch are the technologies/framework I work daily on. My open-source contribution can be found below -

1. The Processing Foundation - <https://goo.gl/PyJmCh>
2. NetLogo - <https://goo.gl/KVcZ8B>
3. P5-web-editor - <https://goo.gl/19rZuN>

My LinkedIn profile is available at <https://goo.gl/JzyTDh>