**Future work**

The future work contains four sides:

The function downloading is needed when we only have uploading and sharing currently. To implement this function, there are two typical ways. Firstly, we think about the standard URL download which is embedding an URL hyperlink in the web page and then downloading using a standard HTTP GET request. However, the drawback of this method is that the path of the file is completely exposed, which leads to a low security and privacy of the website. The second method is to submit the form, submitting the parameters to the server-side dynamic script by using POST request, and then the server-side script returns the output binary stream to the browser for download. This downloading method is not only available for the specific files on server but also for the data that dynamically generated by server. We will experiment with both methods and choose the one that works best.

Various privilege will be set. Now, the users can directly view and modify the file when the file creator add them, and they have the same privilege as the creator. In the future, we will divide permissions into read, write, and operate which contains deleting, downloading and sharing file. The owner of a file will have both these three permissions on the file initially. Besides, only the owner can edit the privilege of the file and he/she can let other users to only read, only operate or do nothing on the specific file as they want. To achieve this function, we will need to build the privilege form for each file, and the form should be automatically generated and deleted with the file.

At present our project can only support the operation on the file that browser can open directly like txt. We will improve the system to handle different types of files like excel and pdf.

Our project is based on local database without server to save time and money as the lease of the server is expensive, and the connection to the server is complicated, which involves licensing, installation, maintenance, support, and patching associated with the operating system. However, Serverless computing is not suitable for workloads with high computing performance requirements due to the limitations on resources, and there are many application components in the serverless architecture, so the system is also in high risk to be attacked. Thus, in the future work, we will try to connect our system to the server.