

**Sprint 3 Retrospective Review**

Team 18: Chunao Liu, Anurag Shah, Jenna Zhang, Yierpan Abuduwaili, Michelle He, Jingyuan Yang

**What Went Well?**

We managed to complete all user stories in our initial design document, and implement continuous delivery. Our identification and visualization of program errors has ended up being a very smooth process that completes much faster than the maximum delay set in our design document. The front end did not have to partake in overcomplicated processes to achieve a good looking and high performance application. It has been a great project and working experience and all of us have learned a lot from this project, and we feel confident in our ability to work in such a team in the future.

* User Story #1

As a user, I would like to be able to update and delete my account.

| # | Description | Estimated Time | Owner |
| --- | --- | --- | --- |
| 1 | Implement setAvatar functionality on the backend and frontend | 5 hrs | Jenna |
| 2 | Further implement backend apis for changing account info/ | 3 hrs | Jenna |
| 3 | Implement delete account functionality | 3 hrs | Jenna |
| 4 | Test and debug the new features (Possibly using unit test). | 4 hrs | Jenna |
| # | Description | Estimated Time | Owner |

**Completed:** Users will be able to update their account information (username and email). The backend will check if the new username and email address are already in use and give a warning if they are. Users are able to delete their account, and if the account is deleted, all the teams where that user is a team leader, and all images associated with those teams, will be removed as well.

* User Story #2

As a user, I would like to be able to create/delete a team and add/remove a member if I am the team leader.

| # | Description | Estimated Time | Owner |
| --- | --- | --- | --- |
| 1 | Implement setAvatar functionality on the backend and frontend | 5 hrs | Jenna |
| 2 | Further implement backend apis for changing account info/ | 3 hrs | Jenna |
| 3 | Implement delete account functionality | 3 hrs | Jenna |
| 4 | Test and debug the new features (Possibly using unit test). | 4 hrs | Jenna |
| # | Description | Estimated Time | Owner |

**Completed:** Users will be able to create a new team and become the team leader of that team. If a user wants to delete a team, they have to be the team leader, otherwise, an error message will be displayed and prevent them from deleting the team. The user will also be able to add or remove a team member, but only if they are the team leader.

* User Story #3

As a developer, I would like to prompt users to re-login when the account is inactive for a week.

| # | Description | Estimated Time | Owner |
| --- | --- | --- | --- |
| 1 | Set an expiration time for the login session on the frontend | 2 hrs | Jenna |
| 2 | Set an expiration time for login token on the backend | 2 hrs | Jenna |

**Completed:** If the user is inactive for a week, the system will log the user out automatically, so when the user reenters the app, the app will ask the user to login again and the user is able to choose to login or stay logged out.

* User Story #4 (Runtime Result Display)

As a user, I would like to see the detailed runtime error message or runtime output of the code written in the text editor.

| # | Description | Estimated Time | Owner |
| --- | --- | --- | --- |
| 1 | Write a backend API that puts the typed code received from the frontend into the compiler and gets the runtime results | 3 hrs | Michelle |
| 2 | The backend sends a response including the runtime result (error or output) and the frontend will pop up a window notifying whether the code executed with or without error | 2 hrs | Michelle |
| 3 | Display the runtime error message or the runtime output in a good format, with line management and scrolling | 3 hrs | Michelle |
| 4 | TextEditorPage receives the ocr\_text\_output and displays it as editable | 2 hrs | Michelle |
| 5 | Debugging and testing functionality | 5 hrs | Michelle |

**Completed:** The backend compiles the code and sends a response with the runtime result, either error or output. The frontend displays the result with correct format and color. Clicking “Edit” will redirect the ocr text output to TextEditorPage so the user can edit the code, TextEditorPage will receive the ocr\_text\_output (the output from OCR sent by the backend), so that the user can edit this and fix any Syntax errors of their own, or errors from OCR.

* User Story #5 (Save Typed Code)

As a user, I would like to save the typed code into a specific group and name the code.

| # | Description | Estimated Time | Owner |
| --- | --- | --- | --- |
| 1 | Add the save and discard buttons on the text editor page | 2 hrs | Michelle |
| 2 | Create a popup window that asks user to choose which group to save the typed code | 4 hrs | Michelle |
| 3 | Create a popup window that asks user to name the given code | 3 hrs | Michelle |
| 4 | Include the group ID number and the name in the request to the backend and send request | 3 hrs | Michelle |
| 5 | Display a saving success message to user after receiving response from the backend | 2 hrs | Michelle |
| 6 | Debugging and testing functionality | 5 hrs | Michelle |

**Completed:** After the runtime result is displayed, save and discard buttons will be shown. When the discard button is pressed, the runtime result will be discarded. When the save button is pressed, windows that let the user select the group and rename the file will be displayed. After confirming the group selected and the file name, the file will be saved to the user’s library.

* User Story #6

As a user, I would like to save the image result to a certain group library or discard the image.

| # | Description | Estimated Time | Owner |
| --- | --- | --- | --- |
| 1 | After receiving the result back from the backend, logged in users should be able to save the picture to a certain group selected by the user or simply discard. There should be a pop up window with the list of groups that the user has, so the user can select one and save. | 4 hrs | Yierpan |
| 2 | After selecting the group, there should be another window prompting the user for a name, to rename the picture that is being saved. After selecting a group and renaming the picture, if the picture is saved successfully, show a success message. | 3 hrs | Yierpan |
| 3 | If the user is not logged in, save the picture to the local gallery or discard, if the picture is saved successfully, show a success message. | 4 hrs | Yierpan |
| 4 | Test and debug | 3 hrs | Yierpan |

**Completed:** After the result is received from the backend, if a logged in user presses the “save” button, they will see a list of group\_id (groups) to save the image to (including their personal group). After the user selects a group, there is another pop up window which asks the user to rename the image. If a user that is not logged in presses the “save” button, the result will be saved to the local device gallery. The user can discard the result and keep using the application.

* User Story #7

As a user, I would like to see the runtime result of the image code and have a pop up window for the errors.

| # | Description | Estimated Time | Owner |
| --- | --- | --- | --- |
| 1 | Have pop up windows around the red curly lines in the result image, the correct pop up window with the correct “error message” should pop whenever the red curly line is pressed, using the y-coordinates, the same error message popup should show for the same red curly line. | 10 hrs | Yierpan |
| 2 | Have a separate “terminal” window below the result image that shows the output of the program. | 4 hrs | Yierpan |
| 3 | Edit function redirects ocr\_text\_output to TextEditorPage | 2 hrs | Yierpan |
| 4 | Test and debug | 4 hrs | Yierpan |

**Completed:** A pop up appears whenever the user clicks the image code. There will be another pop up shown as the small “terminal” window that displays the return value or the output of the program. Users can click the image again to see the result and error. Clicking “Edit” will redirect the ocr text output to TextEditorPage so the user can edit the code.

* + - User Story #8 (Library page)

As a user, I would like to access all the images and code pieces shared among my teams.

| # | Description | Estimated Time | Owner |
| --- | --- | --- | --- |
| 1 | display a list of images / code pieces for each group | 2 hrs | Jingyuan |
| 2 | Add button for delete files and hook up back end | 2 hrs | Jingyuan |
| 3 | Each group has an extendable link to the group file | 4 hrs | Jingyuan |
| 4 | Implement a page for displaying a specific image and its runtime errors / output. | 3 hrs | Jingyuan |
| 5 | Implement a page for displaying a specific piece of typed code and its runtime errors / output. | 3 hrs | Jingyuan |
| 6 | testing and debug | 4 hrs | Jingyuan |

**Completed:** A list of images is displayed for each group when the corresponding group number is clicked. Pressing the delete button will delete the corresponding image on the server and automatically update the image list. The original image, after image and the code are displayed in the extended link for each file.

* User Story #9 (OCR Preprocessing)

As a developer, I would like to have an image preprocessing suite that can de-skew and remove lines/spots

| # | Description | Estimated Time | Owner |
| --- | --- | --- | --- |
| 1 | Binarize the image | 3 hrs | Anurag |
| 2 | Rotate (Deskew) the image using Projection-Profile method | 2 hrs | Anurag |
| 3 | Remove noise from the image | 4 hrs | Anurag |
| 4 | Thin and Skeletonize input (Handwritten only) | 6 hrs | Anurag |

**Completed:** Images are binarized to a black background. The thresholding is carried out using the adaptive thresholding method, with different thresholds for back background and white background images. Images are rotated with the same method as post-processing. Only handwritten images are denoised, and data is skeletonized with a kernel of 2.

* User Story #10 (OCR for Handwritten data)

As a developer, I would like to have an image preprocessing suite that can de-skew and remove lines/spots

| # | Description | Estimated Time | Owner |
| --- | --- | --- | --- |
| 1 | Implement handwritten vs typeform detection | 10 hrs | Anurag |
| 2 | Transform input text to be work with existing OCR | 6 hrs | Anurag |

**Completed:** Images are segmented with a basic segmentation algorithm, and individual letters are recognized as handwritten or typeform for the image to be classified as such. Handwritten images produce textual output with over 80% accuracy.

* User Story #11 (Furthur Backend API Implementation)

As a developer, I would like to have more APIs to support temporary Image saving, Typed code compilation, etc. A MySQL trigger should be implemented so that if a group is out of user, the group should be automatically deleted.

| # | Description | Estimated Time | Owner |
| --- | --- | --- | --- |
| 1 | Implement temporary Image upload and processing function | 3 hrs | Chunao |
| 2 | Implement the TextCode execution API | 4 hrs | Chunao |
| 3 | Implement the API for removing an Image from an group and update any correlated field change | 5 hrs | Chunao |
| 4 | Implement a trigger that if no user exist in a group, the group will be automatically deleted | 6 hrs | Chunao |

**Completed:** The back-end now fully supports text-compilation with only text as input. It also supports processing text and images without saving them into the database. In addition, the server also allows deletion of images, alongside correcting any data related to it. Django automatically deletes the foreign field if the original field is destroyed with parameter ON\_CASCADE.

* User Story #12 (Language Detection)

As a developer, I would like to know what language I am supposed to compile with. We will use existing tools to help distinguish languages

| # | Description | Estimated Time | Owner |
| --- | --- | --- | --- |
| 1 | Implement a function that can detect whether the language is Java, C, C++, etc. | 7 hrs | Anurag |
| 2 | Upon having the correct language identified, proceed to call the compiler wrapper and return the runtime result | 6 hrs | Anurag |
| 3 | Setting up the environment for various languages, such as Java | 5 hrs | Chunao |
| 4 | If one language is not supported by our OCR, we should not proceed to call the OCR handler and return only an error message | 2 hrs | Chunao |

**Completed:** A function can detect the language of a code sample. In case the code sample is misidentified due to ocr errors, C is taken as a default language. The compiler is able to create docker images and scripts to execute code in C, C++, Java, and C#. The regular backend (outside of containers) is able to compile these same languages for testing purposes. In case a user specifies a language not listed above, an error message is returned.

* User Story #13 (Multiple Language Support)

As a user, I would like to have my C/C++ code compiled & run.  
As a user, I would like to have my Java code compiled & run.  
As a user, I would like to have my C# code compiled & run.

| # | Description | Estimated Time | Owner |
| --- | --- | --- | --- |
| 1 | Implement language detection | 4 hrs | Anurag |
| 2 | Support Java code | 5 hrs | Anurag |
| 3 | Support C++ code | 4 hrs | Anurag |
| 4 | Support C# code | 5 hrs | Anurag |

**Completed:** Language detection is implemented using the Guesslang library which uses tensorflow. Java code is supported using the openjdk:11 container. C++ code is supported using the alpine-gxx container (alpine is used as a lightweight container to save resources with C and C++). C# code is supported using the mono container. Error messages are parsed to identify line numbers for the errors, to use in highlighting.

* User Story #14 (Jenkins & Apache setup)

As a developer, I would like to achieve continuous delivery via Jenkin

| # | Description | Estimated Time | Owner |
| --- | --- | --- | --- |
| 1 | Setup Jenkins server on localhost | 6 hrs | Chunao |
| 2 | Setup scripts and web hook to allow Jenkins to automatically build upon git push | 12 hrs | Chunao |
| 3 | Upon failed build, Jenkin should revert the change to the previous state | 4 hrs | Chunao |
| 4 | Testing to make sure Jenkins’ workspace is working | 5 hrs | Chunao |

**Completed:** The server currently has the Jenkins running in port 8000. The port was manually configured so that it won’t conflict with the production API server. Jenkins script and build script are written and are available on Github for Jenkins to fetch and execute. A Github webhook is set so that whenever something is pushed in the main branch, Jenkins will automatically build with the scripts written. Upon a successful build, Jenkins will push the server so that the server will be running in the most updated version; Upon a failed build, Jenkins will reject the push and roll back to the most recent successful build.

**What Didn’t Go Well?**

We managed to completely finish sprint objectives. The main thing that did not go well was that our backend server crashed before the review presentation. However, we dealt with it well, were able to quickly recover from the crash and give our presentations as we intended. This needs to be completely avoided for the next sprint.

**How should you improve?**

The main improvement we need, regarding what is left in the course, is to arrange for the final in person meeting, to create our final presentation and rehearse it.