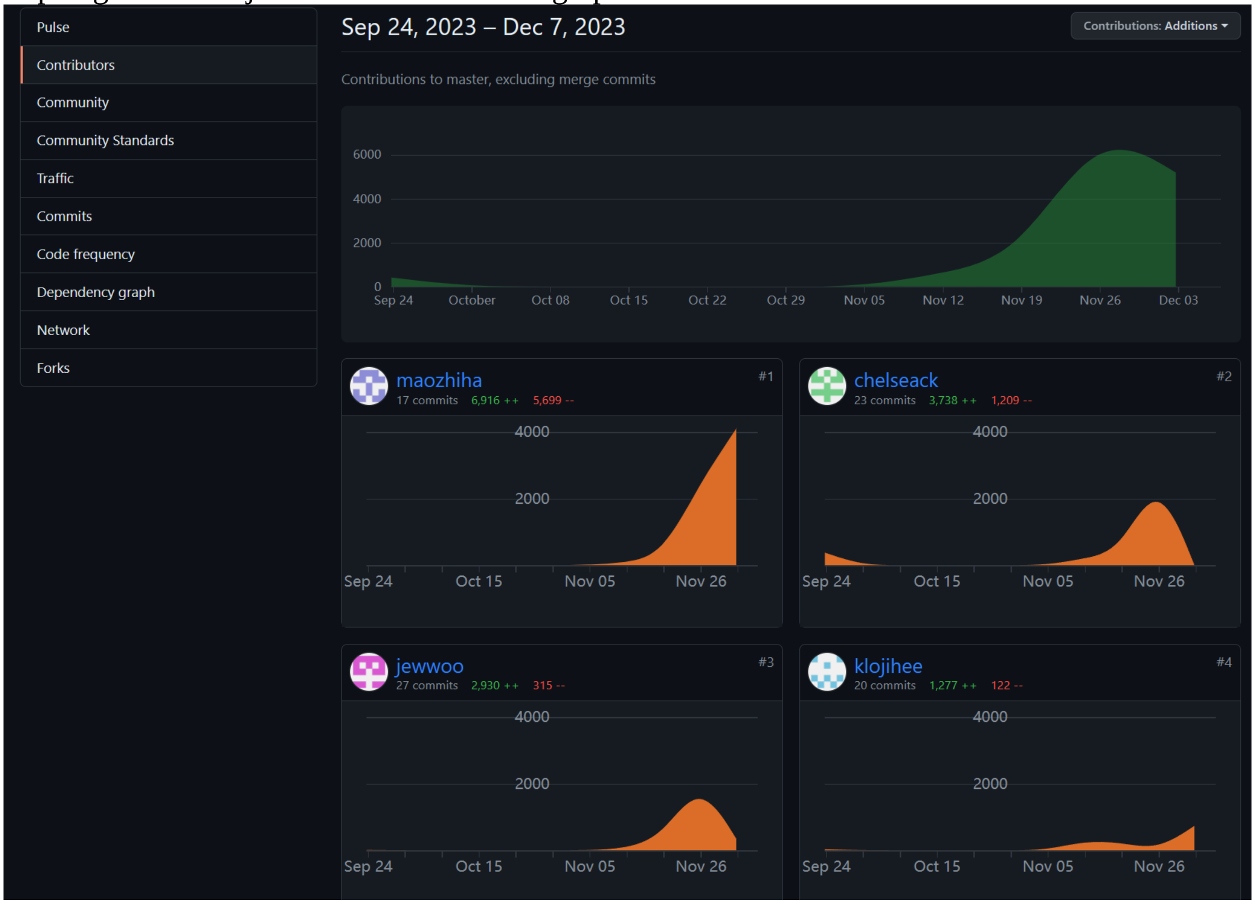
CSC207 Individual Contribution

GitHub username: maozhiha
Link to GitHub repository: https://github.com/jewwoo/Fantastic-Four

---

Git Contributions:

- A link to your repository's GitHub contributors graph OR a screenshot of the contributors graph
- https://github.com/jewwoo/Fantastic-Four/graphs/contributors



Sep 24, 2023 – Dec 7, 2023

Contributions to master, excluding merge commits

maozhiha #1
17 commits 6,916 ++ 5,699 --

chelseack #2
23 commits 3,738 ++ 1,209 --

jewwoo #3
27 commits 2,930 ++ 315 --

klojihee #4
20 commits 1,277 ++ 122 --

-

- Explanation of why your git contributions are different from the average contributions of your team members. **(Note: either significantly higher or significantly lower needs to be convincingly addressed here or we can’t give you full credit for this git contributions component of the project)**

I have significantly higher lines of contribution than my teammates (maozhiha). I think this is mainly due to three reasons:

1. I took the responsibility to do the initial repository set up. I set up the codebase as a Maven project and borrow the code architecture we saw in the week5ca (Clean Architecture) to set up our code base. With this, team member can start working on implementing their own use cases.
2. I implemented most of the Junit Test to test our code base, mainly due to other team member not familiar with using JUnit in Intellij IDEA, writing tests resulted my line of code contribution to be higher than my team mates.
3. I implemented most of UseCaseFactory code in the code base, other team members weren’t really sure on how show we use ViewManagerModel to switch view as well as initialize the views and use cases in the Main.java, I took that task and went ahead to do it for them, thus resulted that I have higher code contribution than others.

Use case that you were primarily responsible for:

- Links to 1 (or more) Pull Request(s) related to the implementation of this use case, which contain a significant amount of code written by you.
  https://github.com/jewwoo/Fantastic-Four/pull/26

I was mainly responsible for User Comment Use Case. To summarize, this use case is for logged in user to leave a comment on the recipe. To implement this Use case, I first implemented a CommentView using Swing Components in our code base. The view contains a TextPane that display existing comments for a recipe. The view also have a Editor Box that current user can use it to leave comments for the recipe. There is also a `Leave Comment` JButton in the view.

When user click the `Leave Comment` button, the ActionListener will trigger the CommentController I wrote, the CommentController will then prepare necessary input data (In this case, user name and comment detail), and pass it to CommentUseCaseInteractor. In my use case interactor, I used a DAO(Data Access Object) to persist the comments into storage. For simplicity, this time I choose to store comments in a CSV file. I encountered a issue that comment string might cotain comma (,) in the string, if we store it plain, it might conflict with the data format of CSV file since CSV also uses comma as separator. With the help from only resources, I realized that I can encode comment into base64 string and store it in the CSV file, when reading the comment in the DAO, I just need to use base64 functions provided by Java to decode it.

Besides the interactor logic, I implemented the CommentPresenter and CommentViewMode, when user finishes writing comments, the View's TextPane will be updated to display the comments.

To summarize, to achieve the comment use case, I did:

1. Implement a Comment View and initialize it Main.java
2. Implement a Comment Entity
3. Implement Comment Use Case using Clean Architecture
   a) CommentController
   b) CommentViewModel
   c) CommentInteractor (CommentInputBoundary)
   d) CommentPresentor(CommmpentOutputBoundary)
   e) CommentFileDAO
4. For CommentFileDAO, implement logic to read and write Comment in CSV file

Another code contribution that you made:

- Links to 1 (or more) Pull Request(s) demonstrating another significant code contribution which you made to the project.
  https://github.com/jewwoo/Fantastic-Four/pull/31 (Implementing Unit Test)
  https://github.com/jewwoo/Fantastic-Four/pull/25/ (Add Search View for Search Recipe Use Case)

- A short paragraph describing this code contribution.

For PR#25, I implement a Search View using Swing Components. Initially, my Colleague was using JFrame to design a search recipe view, such JFrame is not compatible with our CardLayout JPanel and we are not able to achieve view switching using my colleague's JFrame Search View. I refactored her code to implement same visual components in JPanel, so it can be added to our ViewMangerModel

For PR#31, I implemented Unit Test for our use cases. My teammates were not quite familiar with using JUnit to write Unit test, thus I created this MR to implement Unit Test for all of our use cases. I then use IntelliJ's run with coverage feature to get a code coverage report and included it in our presentation slides.

Two examples of code reviews that you performed:

- Link to one of your code reviews.
  - https://github.com/jewwoo/Fantastic-Four/pull/27/files/6a7c0c22d2d4c9bb6e6bd1af38b7b21e2529a491
- Brief explanation of this code review.
  In this Pull Request, I reviewed my team mate's save recipe use case. I noticed that she did not use UseCaseFactory to create SaveRecipeUseCase but rather created Controller and Input/OutputBoundary in the Main.java and passed it into SaveRecipeView's constructor. I think this is not a good design and thus commented in the PR to advise using factory pattern to create SavedRecipeView.

- Link to another one of your code reviews.
  - https://github.com/jewwoo/Fantastic-Four/pull/22
- Brief explanation of this code review.
  In this Pull Request, I reviewed my team mate's view design for search recipe view. I noticed that she is using ` javax.swing.SwingUtilities.invokeLater` to load a JFrame in the Main.java. Doing so will only load the Search View (written in a JFram) that she wrote and we are not able to perform view switch. I advised her that we need to change this behaviour in the coming commits and did not approve this PR.

Other Group Contributions:

- a short paragraph describing how you contributed to the design of your project.
  When design the project, team member wasn't quite sure what API to use and how to use the search recipe API, I gave my thoughts on the API usage and provided with an example to call REST API in Java using OkHttp and parse the response into Java class. Other than that, I helped build the basic code architecture using clean architecture so team members know where to implement the detail for the use case.

- a short paragraph describing how you contributed to any other non-coding and non-design aspects of the project.
  I helped making the presentation slides in the final presentation. I worked on gathering code coverage report and added it into the slides.

Attendance / Communication / Peer Evaluations:

- You don’t have to submit anything for this part, but you may optionally include an explanation if you had previously unexplained absences during the term or want to draw attention to any other factors impacting your participation in the group project.

- **Reminder to submit your peer evaluations — failure to do so will impact this component of your individual contribution grade.**