January 14th, 2025, 9:00 am to 10:30 am
Present: Jackson Rawes, Owen Fleming, Ryan Ulgiati, Ares Miller, Nathan Hawrylak, Connor Dodds
Absent: Brendan Park (TA)

- Decided to contact TA to arrange a meeting with them to discuss our project on Thursday the 16th at 6:00 pm.
- Discussed how text and images will be pulled from written notes with snipping tool-like functionality.
- Developed requirements list for the functionalities of the application including photo submission, file type conversion, "snipping" files, multi-uploading, separating images, commenting, note creation, liking notes, plaintext conversion, image compression, a show more button, and keeping load times reasonable.
- Designated requirements as front-end or back-end.
- Discussed logistics of compression using gzip. Jackson will investigate this further.
- Decided to convert image to png python code into javascript and discussed edge case of the of the file having "." in the file name. To overcome this edge case, start reading file names from the back.
- Converted requirements into user stories for the report and covered areas missed in initial requirement run through.
- Created a Jira project to hold user stories and separate them into sprints.
- Split the requirements between members for implementation.
- Decided to wait for the meeting with the TA to finish the backlog.

End of meeting

January 17th, 2025, 1140: am to 12:00 pm
Present: Jackson Rawes, Owen Fleming, Ryan Ulgiati, Ares Miller, Nathan Hawrylak, Connor Dodds, Brendan Park (TA)
Absent: None

- TA went over the table of contents expectations (it's on the example release planning document we were given, it's intended to help whoever is reading the document jump between sections.)
- Reviewed our user stories/requirements with the TA, mostly to gauge roughly what is expected in the release planning. (Needs more structure/more req./stories, product backlog etc. Be as extensive as possible. Followed by sprint backlogs. While it won't be fully structured, he (Nasser) is looking that we have sprint 1 planned out and then a rough draft for the next few sprints planned out. JIRA/Github screenshots, group contributions, and so on.)
  - Template is a rough idea, but if we follow close to it we should be on the right track.
- Stuff from meeting minutes into release planning/other reports (we kind of did for the first meeting)

- Brendan mainly says we want our:
  - User Stories
  - Tasks related to user stories (start with as much as we can think of)
  - Tasks in the product backlog, that will be distributed into the sprints
    - I.e just give a "global zoomed out view" about what we're trying to do.
- Brendan talked about the importance of even adding tentative tasks, to show we were talking about other possible features
  - Also gave an example of breaking down stories: "As a user, I want to be able to upload batches of notes in bulk so that I can save time."
    - Need to be able to upload files
    - Need to be able to upload more than one file at a time
    - Need to have some sort of association between the user and uploads, etc.
- Discussed the importance of Database Management tasks
  - Not necessarily a user story, but we can add it from a user's POV.
    - I.e "As a user, I want my scans to load quickly, so that …"
- Splitting user stories into tasks could result in overlapping tasks.
- Elaborated on how we were to share our user stories/tasks and how to build them. (Brendan taught us how to branch stories into tasks on JIRA. Thanks, Brendan!)
- Discussed the breakdown of tasks for adding more user stories/expanding upon user stories.
- Release Planning Document overview for task breakdown.

End of meeting

January 21st, 2025, 9:00 am to 10:00 pm
Present: Jackson Rawes, Owen Fleming, Ryan Ulgiati, Ares Miller, Nathan Hawrylak, Connor Dodds
Absent: None

- Discussed the current breakdown of tasks in the sprint backlog/sprint planning for evaluation.
- We discussed the possibility of adding more features, but we are still tentative.
- Talked about which features may be more important, and thus should be implemented first within the first sprint.
- Breakdown of which developers should start on which tasks.
  - A bit more in specifics, such as what tools we think will work best, who has experience with those tools (such as NodeJS, Postgres, etc.)
- Rearranging of some tasks to some later sprints, as they are non-functional/lower priority tasks.