

**Attendees (8):**

Jacquelyn Co  
Tiana Canja  
Aaron Chao  
Dustin Lin  
Hana Kim  
Nhan Pham  
Sasha Kubichka  
Sanat Bhandarkar

**Objective:**

In this meeting we will present our pitch as a team, present questions to our TA, listen to feedback, and prepare for the next upcoming deadlines.

**Agenda:**

- Go over anything that has occurred since the last meeting
- Present project pitch as a team
- Feedback/Questions

**Information Covered:**

- Good pitch, good structure
- Questions we asked -
  - What was expected from the system diagram?
    - Extension of system diagram from labs
    - Maps out the sequence of events when the user is going through the application
    - System diagrams/flow charts aren't very different from each other
  - What is the purpose of tagging -
    - For possible searching - explore grouping in case there is time to implement search
    - There are 2 types of collection: predefined and custom
- Feedback/Questions from Sanat -
  - Work on a series of flowchart to map out usability and events from user perspective - same as lab
  - Section after problem statement - was about personas and not stories
    - User stories focuses more on functionality and features from user perspective
  - Wireframes are low fidelity - are there any high fidelity wireframes on figma?
    - Located on slack from Monday's post
    - Not an issue or problem
  - What is your vision for the UI? Minimalist route or customizability -
    - Talked about having a seasons theme
    - No current central theme or style
    - Focus on user customizability based on the different time zone

- Need to have a baseline - default setting on UI and provide customizations
- If we had a high fidelity wireframe of default setting, this wouldn't have been a question
  - Need to construct them at some point - they must be concrete
- Do you have concerns about the login page?
  - Not sure if we needed it at first
  - We won't be using cloud based storage, more so storing on users local machine
  - Using a pin number rather than password
- What is the function of the pin or password?
  - If the data is being loaded from the users machine then pin/password may not be necessary
- How you are treating login as a feature - doesn't seem like a strongly included feature. You want data stored locally, and are abandoning cloud storage.
- What was Powell's view on storing data locally
  - We asked how to store data, and he gave us options, decided locally was a good idea
- If it's stored locally - there is no need for encryption
  - We can focus more on other features since this isn't needed
  - We reduced the load of backend significantly
  - Explore customizability, integrating camera or audio API
- If we change to cloud tell TA immediately
- I like the track of features
- Search feature is complicated - scope it out so it is simple to access data
- Do you have a generic search bar or are you tagging the data and searching specific tags?
  - No concrete decision had been made yet
  - You shouldn't have a generic search bar, it's a huge overhead and won't be intuitive
- You are focused on making is desktop based - is this a conscious decision to not make it responsive or is it due to time
  - We thought it needs to be strictly a web app not necessarily a time constraint
  - Web apps are responsive in some way - think about it towards the end
  - Try to explore responsive UI even at a low level
- Why is nested bullets not a core feature
  - We should include as optional feature rather than a no go
  - They should be a core feature - they don't need to be infinite
- Core features in order of priority:
  - Login
  - Nestability for bullets
  - Index page for navigation
  - Collections - group data based on similarity
  - Logging for daily or future logs

- Add on features:
  - Security - having a pin feature
  - Personalization / user customization (push it and make it stand out)
  - Video and audio capture
- What about exporting data?
  - Export into text file in a presentable way rather than just seeing a bunch of data
  - Csv or excel would be logical (using JSON data)

Extra Notes:

- **What is our vision for our app is and what do we want to do?**
- **We need crud application & our application needs to persist in some sort of way**

**Core features to include:**

- logging for domain specific items (tasks,events,notes)
- nestability to some extent
- index page/navigation
- collections (irrespective of whether or not we want to search) group data based on similarity, tags, or etc
- daily and future logs (monthly weekly logs)

**Add ons that would make sense to add**

- security related stuff (pin)
- user customizability (design personalization - push through this w our application especially prioritize this)
- video + audio capture (API's)
- build app that addresses minimum concerns and doing it well
- exporting data onto somewhere (export text file in a more presentable way -- csv or excel sheet) (this is optional)
- migration can be a very logical feature to include (especially for habit tracking - makes sense especially for the tracker feature to keep track of how well they're doing)

**Collections**

Collections are part of bullet journals by default and that's a core feature of a bullet journal (need to group data logically)

- few tags or categories for your data
- we can just group things into categories .. prioritize categories over search bar
  
- grouping data
- having a predefined set of categories
- assigning those tags
- organizing our logs

## **2 kinds of collections**

- predefined something you hardcode specific type of category  
ex: collection for ucsd coursework notes, household stuff, personal wellbeing stuff (predefined collections that we create that we can allow users to pick)
- and we can filter and search this data; explore having collections in our application & we want to filter and group that data or perform search for that data
- custom collection : user can create their own collection

## **Key applications of databasing on cloud**

- accessibility of data
- want their application to be accessible on any device across any device at any time

## **Basic structure of pipeline on github (eslint on pipeline, jsdocs, prettier) documentation/linting on your code**

- need to create an action on github and set that workflow running
- could be something as trivial as jsdocs and prettier
- can explore stuff like code quality or style enforcement
- code climate can include in pipeline rn don't need to worry about dependencies
- need to establish a pull request strategy (for example when somebody wants to commit code, how would you have that person systematically do that when they want to pull?) how to set up pull request and manual review strategy
- someone manually reviewing pull request & to look for potential conflicts.. then approving and then finally merging it (note this down for checkpoint 1)
- adding jest and cypress might be harder for ci/cd pipeline
- essential: linting, code climate, pull request strategy, jsdocs
- can't be broken pipeline
- anyone who commits code into repository will trigger the workflow

## **More Notes:**

- ci/cd pipeline and how we are going to be committing code
- know what additional functions we are doing
- come up with concrete design for both default and customizable pages
- come up with flowchart