Team Number 13

Team Members: Parveen Kaur, Jowaki Merani, Jui Nagarkar, Eduardo Sanchez

Project Name: Mindly

Project Synopsis:

Helps users maintain an online journal, track their moods, stress relieving activities, calming music for meditation and provide mental health resources as needed.

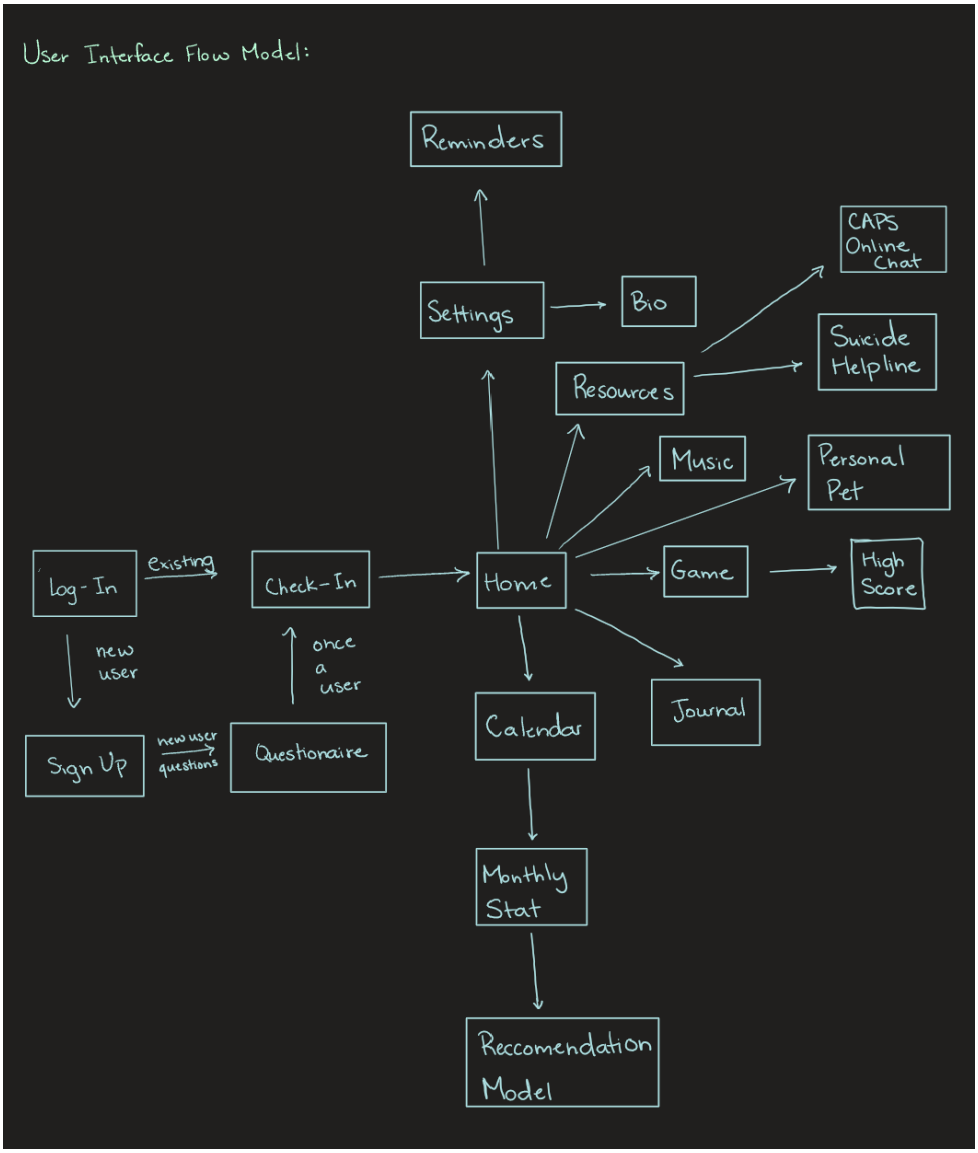
Architecture (or Design):

In our software, we will be creating a mobile application with the use of Django, Python, HTML, CSS, mongoDB for the iOS mobile operating system.

User Interface Model:

Based on the User Interface Model above, once the application is opened, the first screen the users are taken to is a login screen where they are asked to input their email along with their password. The username and password are cross-checked with the database mongoDB. If the user does not exist, they are given the option to sign up and are asked to fill in a questionnaire on personal information that includes their Date of Birth, gender and some questions on their personality type. Next they are taken to a daily check-in page that asks them about their mood during the day. On completing that the user can view their home screen which will have multiple clickable icons for them to choose from. These icons include a Calendar where they could access their journal entries from previous dates and includes the monthly statistics page which will reflect the check in status for the current month. Based on this it will give the user a recommendation model to further improve their mental health for the following month. Next, there will be a Journal icon which will allow users to input information about their day. The Music section will give the user a set of calming music to choose from. This music could be used to relax, meditate or study and would be provided by the School Of Music at KU. Next, the Resource Page will allow users to access support services such as the Counselling and

Psycological Services (CAPS) online chat as well as provide the contact number for the suicide help line. The Settings will comprise two sub categories of daily reminders and a bio. The bio creates a personalized section which gives information on the user's personality preferences such as introvert or extrovert. Daily reminders serve as a notification to the user's phone as a reminder to complete the daily check-in. This will encourage them to use the app daily to improve the recommendation model in the app.



We would also like to implement a game that would help relieve the anxiety. The game would be a tapping game which would allow users to start a timer and see how many taps they are able to do in a certain amount of time. Users will also be given a highscore as an incentive to try and beat their previous score. Finally, we will have a Personal Pet for the user. If the user consistently checks in everyday, the pet maintains its happy state otherwise the pet becomes sad. If we have more time to go further with implementing other features for the pet, we are planning to include the pet within the

calendar where based on whatever the user inputs for each day for their check in, the pet will reflect the same mood and might consider in the future to add pet accessories option to the application.

Technology Stack:

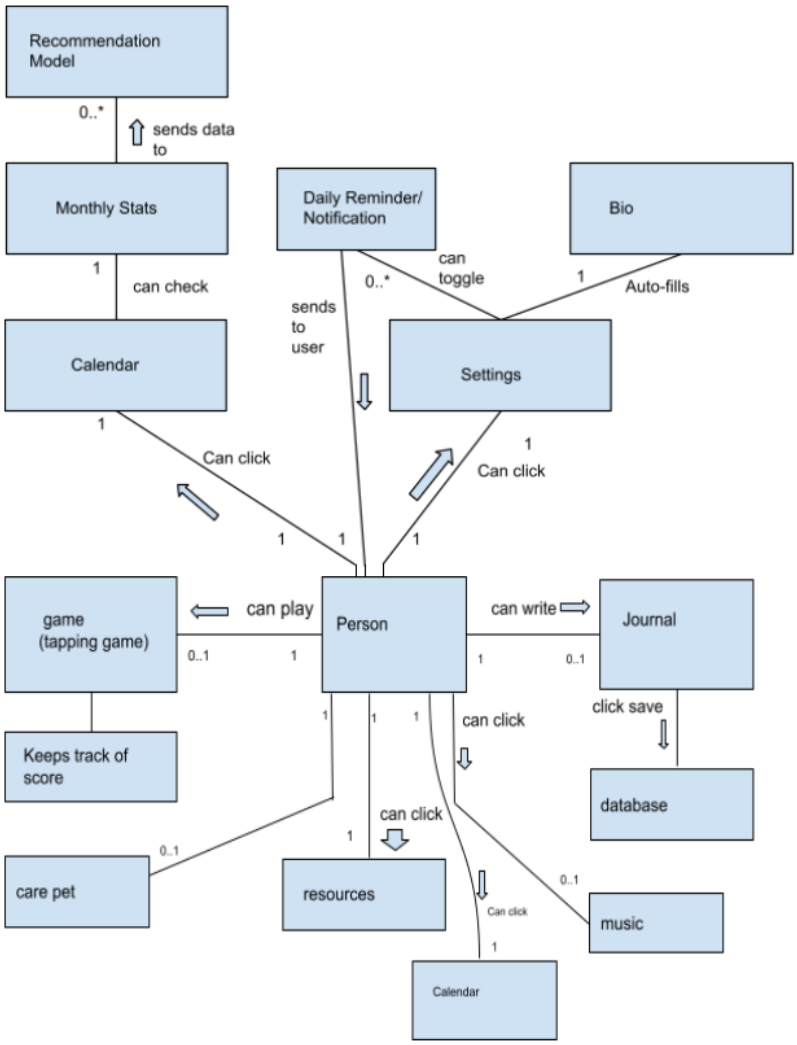
As per our technology stack shown above, our plan is to use python, mongoDB, django, HTML and CSS. We want to use python and mongoDB when implementing the backend portion of our project. This would include communicating with the database and implementing the required queries and implementing the tapping game. The backend includes using mongoDB as the database which will help ensure the user has a previous account in order to move to the next phase of the application. MongoDB is where the data for the journal entries and the daily check-ins would be stored. The data would be retrieved from the database to help create the monthly graph. If the user does not exist, the questionnaire they fill in will be saved in the mongoDB database. This information should be retrieved from the database if the user chooses to view, remove or update their personal data. We chose python as our main backend language. It is easy to understand and has a lot of useful libraries that help visualize data. This would be useful in displaying the

graphs and analyzing journal entries for trigger words so appropriate resources can be provided.

For the frontend portion of the application, we plan to use Django, HTML and CSS. Django is an open-source, Python-based web framework. We plan to use Django to ensure that we can add animations and ensure that the application we create is easy for the user to use. Django will also be important as we plan to make a tapping game for the users. HTML and CSS will be supporting frameworks that we are considering for the implementation of our application. We plan to build this application specifically for the IOS platform. Django supports iOS making the implementation much easier. Since the iPhone is one of the most popular choices among students we choose the IOS platform. Additionally, we think that IOS is a more reliable and user friendly system. Building applications on IOS is also more cost effective.

Domain Model:

The Domain model visualizes the dependencies between the various applications the user can choose. On a high level, the main app is centered around the user's home screen . The user can choose to make instances of the options provided to them. This encourages a stress free environment to explore the application however the user wishes. There are pieces of software that require user



input for the model to work as intended. For example, the recommendation model could have zero or more instances of recommendations depending on how frequent the user uses the app, the more frequent the user visits the app and interacts with other parts, the better the recommendations are given to the user. The care pet is not forced on to the user. If the user decides not to activate the pet that part of the application would go unused. The mentality is shared along with the game designed and music choices to relax the user but is never forced on the user. The music is selected from an established library and pulls inspiration from many public lofi music streams to encourage a calm mentality.