Project Title

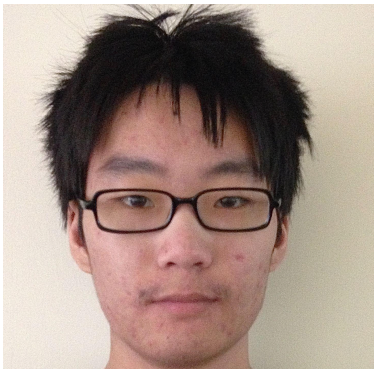
(This should be the same as in the progress report if there have not been changes in the last three weeks).

SnapHealth: Snap, Track, Thrive - the key to vitality simplified.

Student/Team Information

(This should be the same as in the progress report if there have not been changes in the last three weeks).

Team Name if any: **Team # on Canvas you have self-signed-up for: Group #29**	**Group 29**
Team member 1 (Team Lead) *(Ha, Marcus; 89529196; nguyeh3@uci.edu):*	
Team member 2 *(Last Name, name; student ID; UCI email, picture):* *(Aoun, Adam; 82395663, aouna@uci.edu)*	

Team member 3 (Nguyen, Andy; 92634585; andyqn1@uci.edu)	
Team member 4 (Lee, Jay; 61145573; jayl9@uci.edu)	

Project Description

Motivation

(Describe the problem you want to solve and why it is important. Min 100 words, Max 300 words).
(This should be the same as in progress report if there has not been changes in the last three weeks).

The general public typically glosses over how they eat, hydrate, and sleep because it is a natural occurrence in people's daily lives that seems to be second nature to us. This reason is why people rarely take into consideration how this may affect our health. Another reason why people are reluctant to prioritize these aspects in our lives is because of a lack of understanding of the significance of these issues as well as the lack of resources to help maintain a healthy balance of these factors. Acknowledging and prioritizing the impact of diet, sleep, and hydration is crucial for promoting overall health and one's well-being. These three factors play integral roles in supporting various physiological and psychological functions, influencing an individual's performance, mood, and long-term health outcomes. By proactively embracing these life choices, we can lead a healthier and prolonged life.

State of the Art / Current solution

(Describe how the problem is solved today (if it is). Min 60 words, Max 200 words).
(This should be the same as in the progress report if there have not been changes in the last three weeks).

Currently, addressing the issues of diet, hydration, and sleep often relies on one's own self-discipline and awareness of such issues. These three factors often do not come across people's minds because there is not enough reliable and convenient access to resources that provide reliable information and solutions pertaining to the issue. There are health professionals to give guidance, as well as wearable devices and health apps to monitor one's health patterns. However, these solutions rely heavily on an individual's motivation to seek these remedies. It can also be difficult for people to receive help on these issues because of a lack of such resources due to external issues. Therefore, there can be obvious improvement in creating a more accessible and personable resource catered toward addressing health in regard to diet, hydration, and sleep.

Project Goal and Approach

(Describe the goal of the project, design, and architecture you have implemented: List the data collection and storing strategies you have implemented. Define your system environment and how the context and personal model are determined. Define the ranking or utility function you use to identify the best item to recommend. Include an updated list of milestones as well. Max 500 words).

We have built a system that provides 3 main features for the user: Sleep Tracker, Hydration Tracker, and Food Recommendation. Food Recommendation recommends three meals per day for the user based on the user's body measurement, activity levels, diet type, and goals. Sleep Tracker gives the user sleep quality based on their age and their sleep habit while Hydration Tracker keeps track of their hydration of the day until they reach their goal based on age.

Personal Model Data Sources:

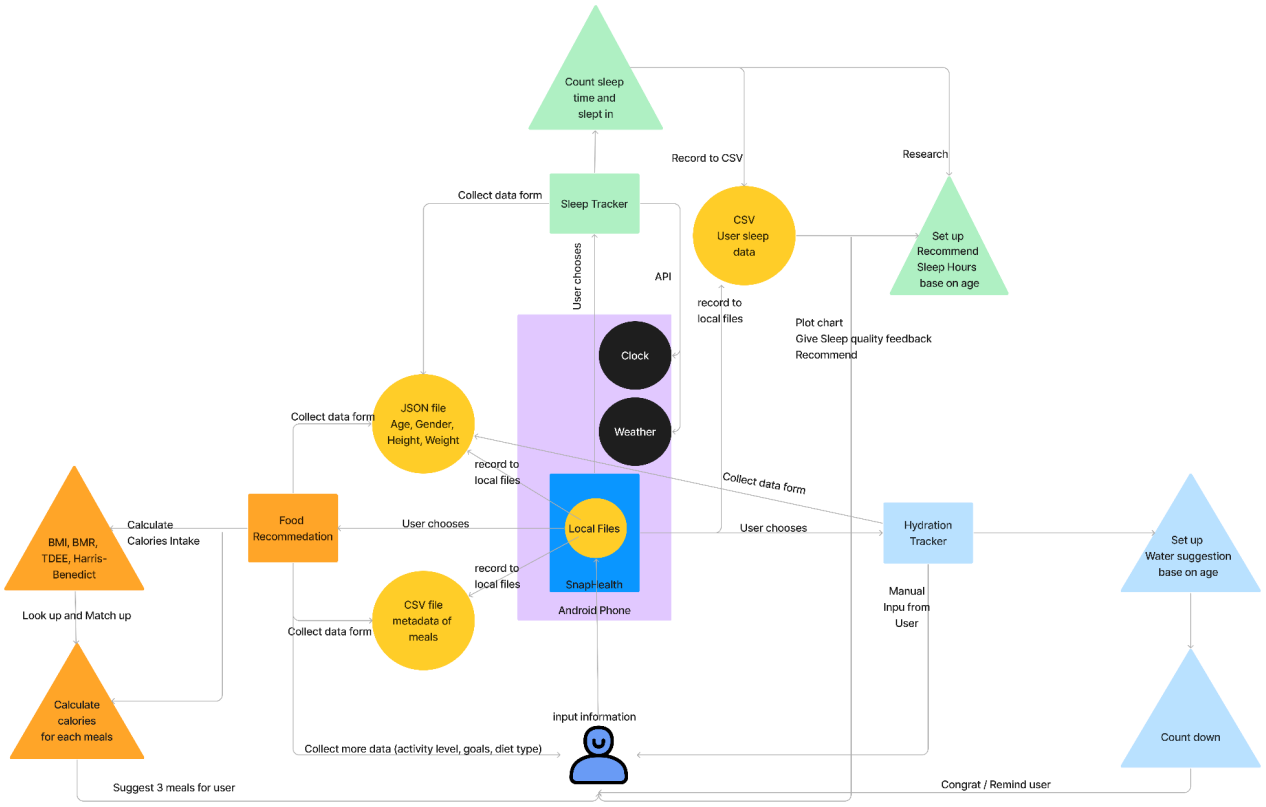
1. *Initial User's sign up (user profile)*
2. *User input based on FoodRecommendation page*
3. *Calories intake based on Algorithm*
4. *Sleep data based on Sleep Tracker page*
5. *Water intake based on user manual input*

Context Data Sources:

1. *Weather Information based on weather API*
2. *Time Information based on Clock API*
3. *Location from smartphone GPS.*
4. *Meals metadata from kaggle.com*

All information is stored in local files inside Device Explorer. The recommendation is presented to users in the mobile app. We cited all research sources in Readme.md

An updated high-level diagram of your project is as follows:



Implementation

(Describe the implementation details for your project based on each criterion (Max 300 words))

Frontend (Desing): Describe your frontend design and explain if it supports all functions you have implemented

For our frontend design, we have 6 total pages: Login, signup, homepage, meal recommendation, hydration, and sleep. In the login page, there are two text entry boxes, one for username and one for password. Then, there are two buttons, login and sign up. If the user has an account already they can enter their information and click login, otherwise, they can click sign up. If the user enters the wrong information for their username and password, an error message will pop up saying that the information was incorrect. Clicking sign up, takes the user to the sign up page, where the user will enter their information and create an account to log in. Afterwards, when the user logs in, the user will be sent to the home page, or the dashboard, where the user can choose what pages they would like to see, meal recommendation, hydration, or sleep. In the meal recommendation, the user will be prompted with their activity level, goal weight, and diet type. There is a calculate calories button afterward that takes into consideration their input. There is also a prepare your meals button that will take them to another page recommending the user meals. In the hydration, the user will have 4 buttons that keep track of how they drink their water which will then decrease the total water consumption goal number. If it reaches below 0, the user will receive a congratulatory message. For the sleep, there is a sleep toggle that enables a live timer. If the timer reaches a specific time, the sleep in timer starts running as well. There is also the temperature sleep recommendation. Clicking your sleep takes the user to another page showing how well they slept.

Frontend (UI): Explain how your product UI is feasible to the user (Usability)
(e.g. Asking the user to enter every meal four times per day is not feasible)

The login and signup process ensures easy access to the app. Once logged in, users are directed to a dashboard that clearly presents their options: meal recommendations, hydration tracking, and sleep monitoring. In the meal recommendation section, users are prompted to enter relevant information, and with the click of a button, they can calculate their calorie needs and receive meal suggestions. The hydration feature simplifies tracking water intake with clickable buttons that automatically adjust the user's daily goal, providing positive reinforcement upon completion. The sleep section offers an intuitive live timer for sleep tracking and temperature-based sleep recommendations, with a separate page to review sleep quality. By minimizing the need for repetitive data entry and providing clear, actionable insights, our UI enhances the user experience and supports the user's health and wellness goals.

Data Collection: Describe the data collected from users and devices.
(We expect you to use minimum of four data containing at least one static data (user input) and one live data(e.g. device, external API, external webcrawler data, etc.)

1. User input their body measurement (Age, height, weight, gender) (static)
2. User input their specific information (weight goals, activity level, diet type) (static)
3. Weather API (live data)
4. Clock API
5. User sleep habit data

Backend (Users Database): Describe whether your application supports multiple users by making use of a database and maintains historical information (not reset/restart).

The application supports multiple users by saving user information in a json file on the device called “user_data.json”. Once a user signs up, the user can log back in with their username and password, even if the device is restarted. The recommendation results for the user will be correct each time because the user’s body measurement data is saved.

Backend (Use of Online Services): If you use an API to retrieve data, explain how the API works.

The application uses the OpenWeather API in the sleep tracker to recommend whether a user should sleep or not based on the outside temperature. The OpenWeather API uses the user’s city in the request call, and the response is live data of the current temperature for the city in celsius. The live temperature is retrieved each time the user opens the sleep tracker page.

Backend (Database): Describe database schema

For saving user information in a json file, a unique username is mapped to the user’s data, which is stored in a list. The list contains the user’s first name at index 0, last name at index 1, password at index 2, gender at index 3, age at index 4, height at index 5, and weight at index 6. For saving sleep data for the sleep analysis, a unique csv file is created for the current user, called “sleep_data_{user’s username}.csv”, when they start and stop the sleep timer. Each time the timer starts and stops, a new line of begin time, end time, sleep time (ms), and oversleep time (ms) is written to the csv file.

Recommendation algorithm (Key modules): List and describe the three main modules of your recommendation algorithm: Personal model + Context + Recommendation = Personal contextual Recommender System.

1. Personal Model:
    - For the sleep tracker: these modules gather information about sleep habits by duration of sleep.Also collect data about user's age to set up recommended sleep hours for different users. Furthermore, the module gathers the user's sleep environment by location and temperature for the best quality of sleep.
    - For the hydration tracker: this module tracks the user's water intake by manual inputting. It sets up the hydration goals based on the user's age and gender and monitors user progress until they reach their goals.
    - For the food recommendation: this module gathers user information such as body measurement, activity level, weight goals, and dietary preference to personalize their nutrition profile. Also calculate the calories intake they need based on BMI, BMR, TDEE, and Harris-Bennedict equations and tailor meals that they need.

2. Context:
    - For the sleep tracker: the context for this module is real-time weather, which can influence their sleep quality. This module considers that the user sleeps once a day for accuracy
    - For the hydration tracker: contextual factors such as time will affect the system, if th time pass 30 minutes but the system not detect user drink water, system pop up reminder (not yet implemented)
    - For food recommendation: contextual factors such as activity level will affect the calories intake. Ingredients, or allergy might cause a bad reaction (not yet implemented).

3. Recommendation system:
    - For sleep tracker: based on collected data and contextual information above, the system provides users with a chart of their sleep habits and recommend users to improve their sleep quality. The recommendation such as cool down the room when the weather is hot, heat up the room when the weather is cold, and give feedback base on their current sleep habit in a week such as (good quality, bad quality or unstable sleep habit)
    - For hydration tracker: based on collected data and contextual information above, the system remind user to drink water if they are not meet their goals and encourage user when they finish their goals for a day. The recommendation to ensure hydration level are maintained.
    - For food recommendation: base on collected data and contextual information above, the system personalize user nutrition profiles and suggest 3 meals for the following day. The recommendation include 3 meals with their macros nutrition (protein, carbs, fat), and the total calories. The system also warns user to research for the food's ingredient for bad reaction caused by allergy.

Recommendation algorithm (Utility function): Describe the utility function (scoring algorithm) that you have implemented to make a proper recommendation (Recommendation results must return no more than 15-20% of possible data queried).

Food Recommendation:
- Collect user data (body measurement, activity level, body goal, diet type) to calculate calories intake using BMI, BMR, Harris-Benedict
- Assign score to each meal in metadata based on calorie intake and diet type then recommend up to 3 meals per day.

Sleep Tracker:
- Collect user data (age) and assign recommended sleep hours.
- Calculate sleep duration based on sleep start and sleep end and calculate average sleep duration and compare to sleep hour recommended with error +- 1 hours to give feed back about their sleep quality.

Hydration tracker:

- Collect user data (age, gender) and assign recommended water intake per day.
- Monitor hydration

In Overall:

- Scoring is used in food recommendation, sleep tracker, and hydration tracker combined, weight score based on goal, adjust recommendation inside a bound and make sure not exceed 15-20% possible data queries

Hardware Required

(The updated list of devices/sensors you are using to collect data for the project)

Component/part	Quantity
Android Phone	1
Android Tablet	1

Project Tasks / Features completed

(Describe the main tasks that have been assigned and completed. Max 250 words).

Task Completed	Team Member
Login Page - Check username, password - UI	- Andy Nguyen, Jay Lee - Andy Nguyen
Sign up Page - Record user information - UI	- Andy Nguyen, Jay Lee - Andy Nguyen
Sleep Page - Sleep and slept in time counting - Record sleep data - Weather API - Plot chart - Recommendation	- Marcus Ha, Andy Nguyen - Marcus Ha, Jay Lee - Andy Nguyen - Marcus Ha - Marcus Ha
Homepage	Marcus Ha, Andy Nguyen
Hydration Page - Water Count	- Marcus Ha, Andy Nguyen, Jay Lee

- Notification - UI - Clock API	- Marcus Ha - Marcus Ha, Andy Nguyen - Marcus Ha
Food Recommendation - Calories Count - User-specific information - Manipulate CSV and retrieve meals - Extra information for each meal - UI	 - Marcus Ha - Marcus Ha, Jay Lee - Marcus Ha - Marcus Ha - Marcus Ha
Other Backend	- Jay Lee

Challenges/roadblocks

(Describe the technical challenges that you have faced and whether you were able to solve them. Max 300 words).

One challenge we faced was our inexperience in our technical background for backend. We were struggling on figuring out how to implement the backend of our application, so that users can save and store their profile information. Eventually, we were able to resolve the issue through self-learning and decided to do a locally based database and store it through json. This way users could have their information saved and stored regardless of resetting the application.

Tasks / Features not completed.*(Describe the main tasks/features that your team was not able to complete. Max 10 tasks).*

Task Not Completed	**Team Member**	**Reason**
Using GoogleFIT API	Jay Lee	Was not able to retrieve information from OAuth playground. Asked TA in Office Hours and told us it may be a Google account issue.
Cannot deal with users who sleep many times with short duration each time per day	Marcus Ha	We consider it is a complicated task and might take us a lot of time to handle
Cannot deal with users with any ingredient allergy	Marcus Ha	The data from kaggle did not provide enough information about ingredient

Not yet implement sleep hour recommendation on Slept in (assign 10 seconds on demo)	Marcus Ha, Jay Lee, Andy Nguyen, Adam	If we set up 8 hours or more, it will be unverifiable

Learning outcomes / Week points / Future work

(Mention what things you learned by implementing this project. List at least two points of your project that have room for improvement. These points can be additions to the existing project setup (components) or improvements to the current implementation. Max 200 words).

From this project, we were able to gain a better understanding of how the developmental process works in creating a mobile app. We also learned how to build effective search systems as well as recommendation systems that are personalized for the user. One thing that our project has room for improvement in is the backend. We utilized a local database through json, but we believe that we could transfer over to a cloud based database for better scalability, accessibility, and overall performance. Another improvement is to integrate more live data within the app. We could connect to other devices such as a smartwatch, and provide better contextualized data for the application. Adding more live data would provide an even more personalized experience for the user.

Project Outcome / Deliverables

Video Presentation

Please include a link to a final video presentation showing the setup, the components and functionality of your recommendation system (10 min max). Please follow the guidelines given in lecture for your presentation. All team members should present a part of the project.

CS125 Presentation + Demo.mp4

Video Demo

Please include a link to a video demo (5 min. max) to demonstrate how your recommendation system works. The approach and other details should not be included here. This demo should focus on showing what your system does, not how it does and what is 'under the hood'.

Use a very brief intro to what the system does (say 30 seconds) and then show important functionality. Don't waste time on basic things like how to login or registering -- show what the recommendation system does and how the information is presented. Show all interactions with users:

1. *for getting data and info as needed*
2. *for recommendation/search*
3. *for presenting alerts (if any)*

In case not all modules/features are working, then use your imagination on how to show creatively what is working and how it would look if every module/feature were working.

demosystem.mp4

Source Code

Please include a link to the source code of your project. A link to a repository (like GitHub) is preferred. We will look for specific commits/contributions/branches/forks by each member to evaluate each student's work.

https://github.com/marcusha4299/SnapHealth/tree/master