

Software Engineering CSC648/848



Team 02 / Section 01

Prathiba Ramesh - Team Lead
Tun-Ni Chiang - Frontend Lead
Jingxing Luo - Backend Lead
Aung Myat - Scrum Master
Christopher Ling - Frontend Support
Myat Kyaw - Github Master

"Milestone 2"

March 18, 2022


Revision History Table

Revision ID	Revision Date	Revised By

Table of Contents

Software Engineering CSC648/848	1
Short Description about iPlate Usage and it's working:	3
1. Data Definitions V2.....	4
2. Functional Requirements V2	5
3. High Level UI Mockups and Story Boards.....	6
4. High Level Architecture, Database Organization.....	11
DB Organization	11
Add/Delete/Search Architecture	11
Application Program Interface (API).....	12
5. High Level UML Diagram	12
UML Class Diagram	12
UML Sequence Diagram	13
6. Key Risks and Mitigation Plan	14
7. Project Management.....	15

Short Description about iPlate Usage and it's working:

Link to Milestone 01 Document	Link to iPlate Web Application
 CSC 648-848 M1 section 01 Team 02.p	http://34.127.94.100/

iPlate helps users to keep a track of their eating habits and lead a healthy lifestyle without putting much manual effort in calculation calories and tracking macronutrients.

Initially user would be asked to register as a new participant to the application with his personal details. And if he/she is a returning participant, they can login with their given credentials. Please note that the user's password is encrypted, which ensures security.

Once he is logged in to the application, he will be able to view the about page which has details about our mission and goals that triggered us to develop this application and also it talks about the food recording method that we will use.

If a user wishes to enter his/her food intake for the day, they can just click on the 'Record!' button which will take you to the MyPlate page. Here, the MyPlate shows different Food groups like Veggie, Fruit, Protein, Carbs in the form of a Pie chart. Once a food group is selected by the user, the different food choices within that group is listed, from which user can select their option for the day. Also, Quantity of their intake for that particular food item can be recorded. Initially before recording a food item, the user is supposed to select the date & meal for which there are going to log (this is a mandatory field).

If user wishes to check his history of food intake, that can be done by our history page, where an analysis diagram is shown based on all his previous food intakes. Also, suggestions if any for the future meals are provided here.

This is the basic working of the iPlate application. Our team is in the progress on enhancing it for better user interface and other exciting features. Stay tuned!!

Note: As part of Milestone 2, we have completed the signup/login page implementation. In order to test this, load the above URL in your browser,

- Click the "sign-up" button on top right corner. Enter username and password of your choice, also confirm the same password and hit the submit button.
- Once successfully signed up, enter the same username, password and login.
- Once you are logged in successfully, you will be redirected to the about page.
- Invalid scenarios- Try signing up with already existing username, Put different password and confirm password, Try logging in with wrong user and with wrong password. In all these cases, an alert with appropriate error message is popped up.

1. Data Definitions V2

Enhanced version of Data Definitions from Milestone 01 Document based on Instructor's Feedback.

Collection Name	Definition & Attributes	Usage
userDetails	Information about users on registration. Attributes- firstName, lastName, location, userName, email, password, platePreferences, bodyIndex	The user document is created when the user reaches the end of the registration. They can change the information such as password or body index in the setting (frontend form passed into the backend to rewrite the user document).
location	Dropdown which will state the list of countries, state and city (based on previous input). Attributes- city, state, country	This will enable the user to first select a country while registering in the app from a dropdown list, based on which the state, followed by city is populated in the respective DDL.
userLogin	login details of the user. Attributes- userName, password.	Login details for users are provided, which will inherit data from the above userRegister collection.
menu	The different types of food items that user can look at and choose from (on our page) Attributes- name, Calories, carbs, protein, fat. Note: The nutrition is measured by 100g only, so it depends on the frontend to pass an accurate measure to calculate the real input.	In a food collection, different food (measured in 100g by default) has its own document. (The food documents may be input manually via Robo3T, or there would be an admin account, which can grab this collection via backend feature, to add or to modify the food data).
foodHistory	food intake history for different users. (List of items from menu for an individual user) Attributes- userName, calories.	Enables users to view history or records of their previous food intake. Planned to view them in a grid using pagination.
advisorsList	Provides a list of advisors with whom users can connect in case of any health concerns. Attributes- advisorName, speciality, advisorEmail	Loads the list of health advisors and their specialization along with the user id, which will enable users to get in touch with them directly.
accountType	Type of an account which will allow users to perform operations based on access role. Attributes- accounttype, role, privilege.	The type of account describing what level of access a user has. Eg- owner, admin, moderator, user

2. Functional Requirements V2

Priority Level and code:

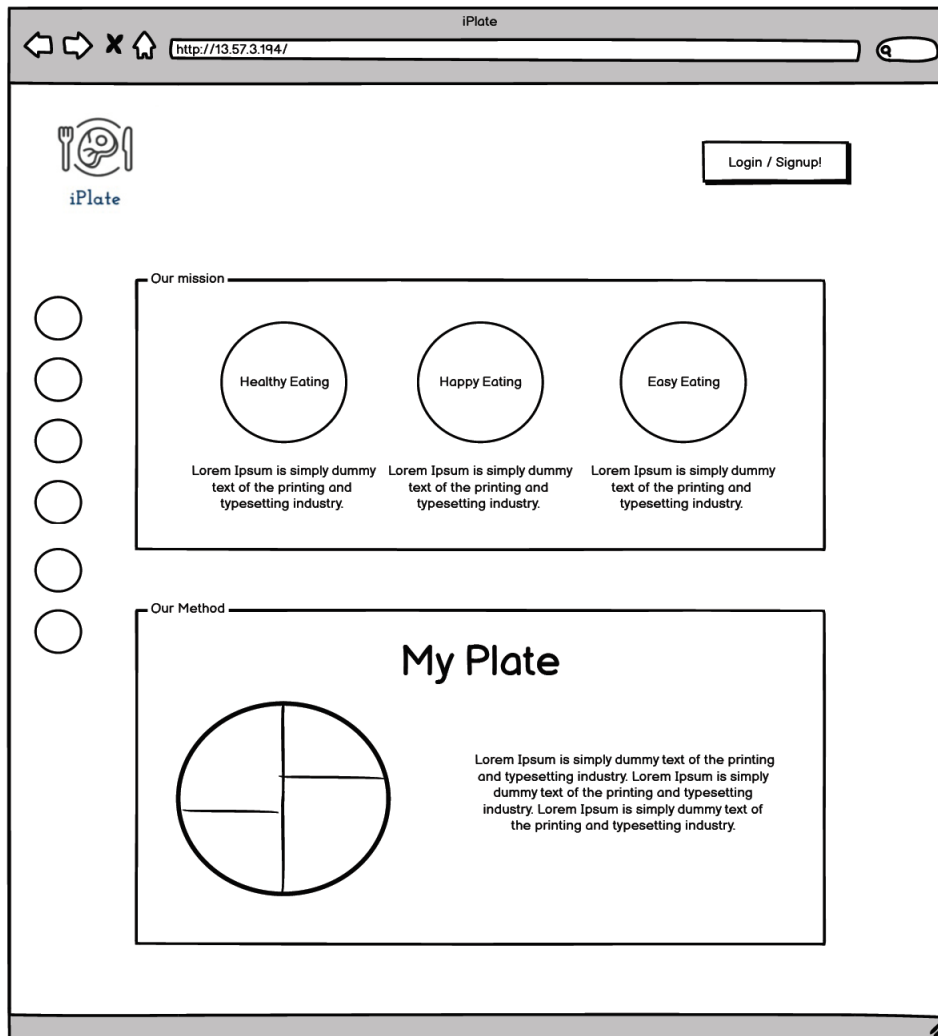
- 1- Must Have
- 2- Desired
- 3- Opportunistic

Function	Sub Function - Description	Priority
Create an Account	New users are required to create account with their personal details like username/email and password	1
	If user prefer to have more accurate plan, then they can choose to do more questions which is the advanced mode (age, weight, height, BMI)	3
Record their daily meals	The recording (main) page would have the plate split into five areas (fruits, vegetables, grains, protein, and dairy). Users are able to record their food based on the food history that they had.	1
	If users prefer, they can record the food with their desired unit. If not, we would use the recommended portion which is published by the U.S. Department of Agriculture.	1
Edit/manage their past or future meals	Users are able to change or pre-plan their future meals. For example, if the users want to have more calories or less calories in the future, users can pre-plan their future meals by choosing	2
	Users are able to revisit their past meals	1
Analysis diagram	Help the users to keep aware of the ratio of the food that they ate. Ratio is calculated by calories defined in DB multiplied by Quantity from user - refer UI mockup for ratio representation	2
	iPlate will also provide the calorie & macro-nutrient option, which will allow user to see their calorie intake and nutrients report (this feature is set to off in default)	2
Diet suggestions	Based on the analysis diagram, iPlate will give suggestions on which food group the user should consume more.	2
	If users choose to turn on advanced mode, a detailed suggestions like the exact quantity they have to consume, etc are given. If not, would be a general suggestion.	3
Water Tracker	Allow user to track how much water they drink a day. The amount of water that users intake will be user inputs. For example, users can input how many ounces of water they have each day	3
Change the plate style	Users are able to customize their plate. For example, color of the plate, reordering of food group, etc	3
HealthAdvisors connect	List the advisors and their specialization along with their contact information such as email so that users can directly click on the link and get in touch with them in case of any health concerns.	2

3. High Level UI Mockups and Story Boards

About Page

This is the page where we share our mission and goals to the user. We will also talk about the food recording method that we will use.



Login Page

- This is the page where users can log in to their account. If the user is in our database, then the page will redirect to the food recording page. If not, will redirect to the registration page.
- req.body will have username and password

The image is a hand-drawn mockup of a web browser window. The browser's title bar says "iPlate". The address bar shows the URL "http://13.57.3.194/". The page content includes the "iPlate" logo (a fork and plate icon) in the top left corner. On the left side of the page, there is a vertical stack of seven empty circles. In the center of the page, there is a large rectangular box containing the text "Welcome back!". Below this text are two input fields: "Username/email" and "Password". Below the "Password" field is a "Login" button. Below the "Login" button is the word "or", followed by a "Signup" button. The browser window has a standard navigation bar with back, forward, and home icons.

Registration Page

- This is the page where users can register for their account.
- req.body will have username, password, confirm password, (optional) height, weight, age

iPlate

http://13.57.3.194/

iPlate

Hello There!

Username/email

Password

Confirm Password

Optional

Your Height

Your Weight

Your Age

Signup!

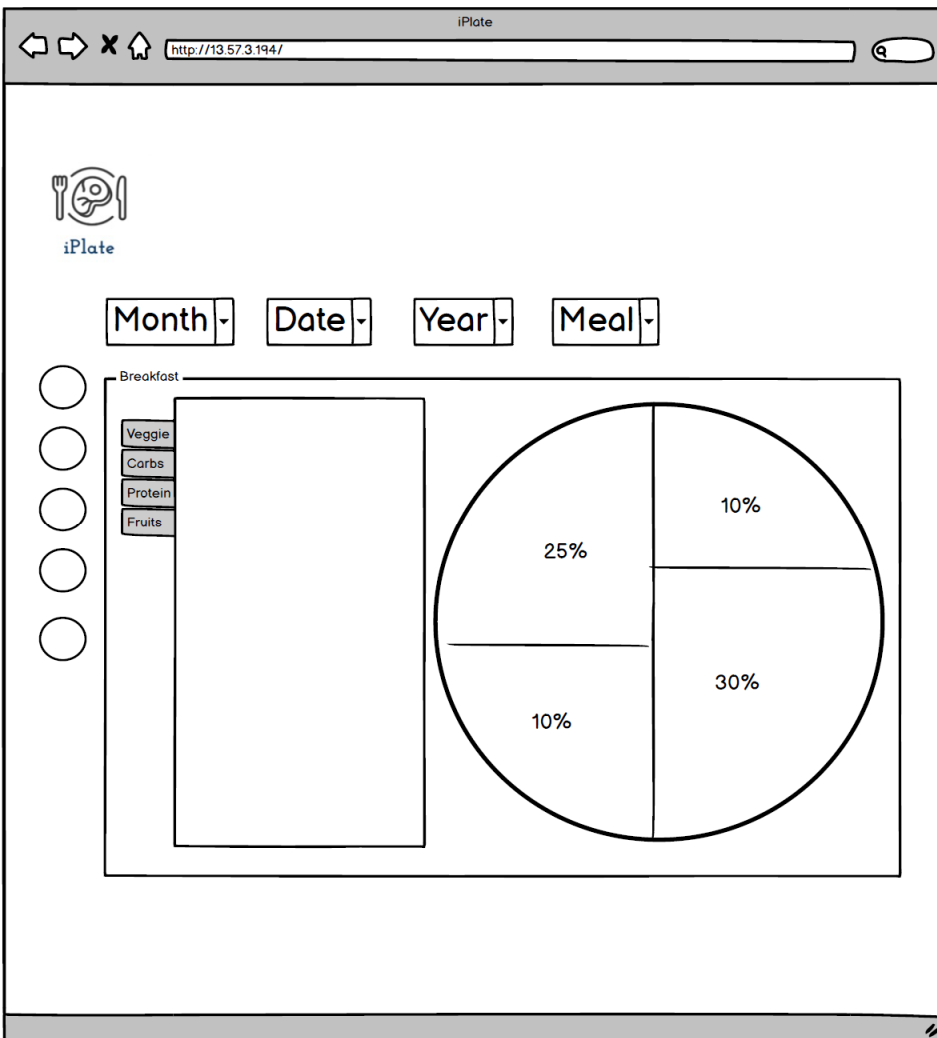
Food Recording Page

- This is the page where the user records their meal using the MyPlate method. Once the user clicks on one of the food groups, the right box will show the food options, and the user can choose their food and record the amount.
- req.body will have month, date, year, meal, food group, food name, food quantity

The screenshot shows a web browser window titled "iPlate" with the URL "http://13.57.3.194/". The page features the iPlate logo (a fork and knife forming a plate) and the text "iPlate". Below the logo is the heading "My Plate for" followed by four dropdown menus: "Month", "Date", "Year", and "Meal". A horizontal line separates this header from the main content area. The main content area contains a large circle divided into four quadrants labeled "Veggie", "Fruit", "Protein", and "Carbs". To the left of the circle is a vertical column of six empty circles. To the right of the circle is a box titled "Veggie Group" containing a list of food items: "Broccoli", "Spinach", "Bok Choi", and "Lettuce".

History Page

- This is the history page where the user can search their records by passing their desired date.
- req.body will have month, date, year, meal



4. High Level Architecture, Database Organization

DB Organization

Table Name	Attribute Name	Data Type
userDetails	userName	String
	userAddress	String
	userId	String
	password	String
	BMI	Integer
	gender	String
	dailyCalories	Float
	userOption	Boolean
Calories	caloriesId	Integer
	foodName	String
	foodType	String
	caloriesCount	Integer
FoodDetails	foodName	String
	quantity	Integer
	foodType	String
FoodType	Id	Integer
	Type	String
UserCalories	loggedCalories	Integer
	Date	string
	time	string

Add/Delete/Search Architecture

DB's add operations:

- Add new user info (regular users as well as admins)
- User adds food history (food, calories, type)
- Admin adds new food (type, quantity, calories, etc)

DB's delete operations:

- Admin deletes a food in DB
- User can disable (delete account) and related data with it (CASCADE delete)
- User can delete food history

DB's search operations:

- Admin and user can search a food in DB
- Admin can look up for users (account, calories history)
- User can look for a specific calorie intake based on date

Application Program Interface (API)

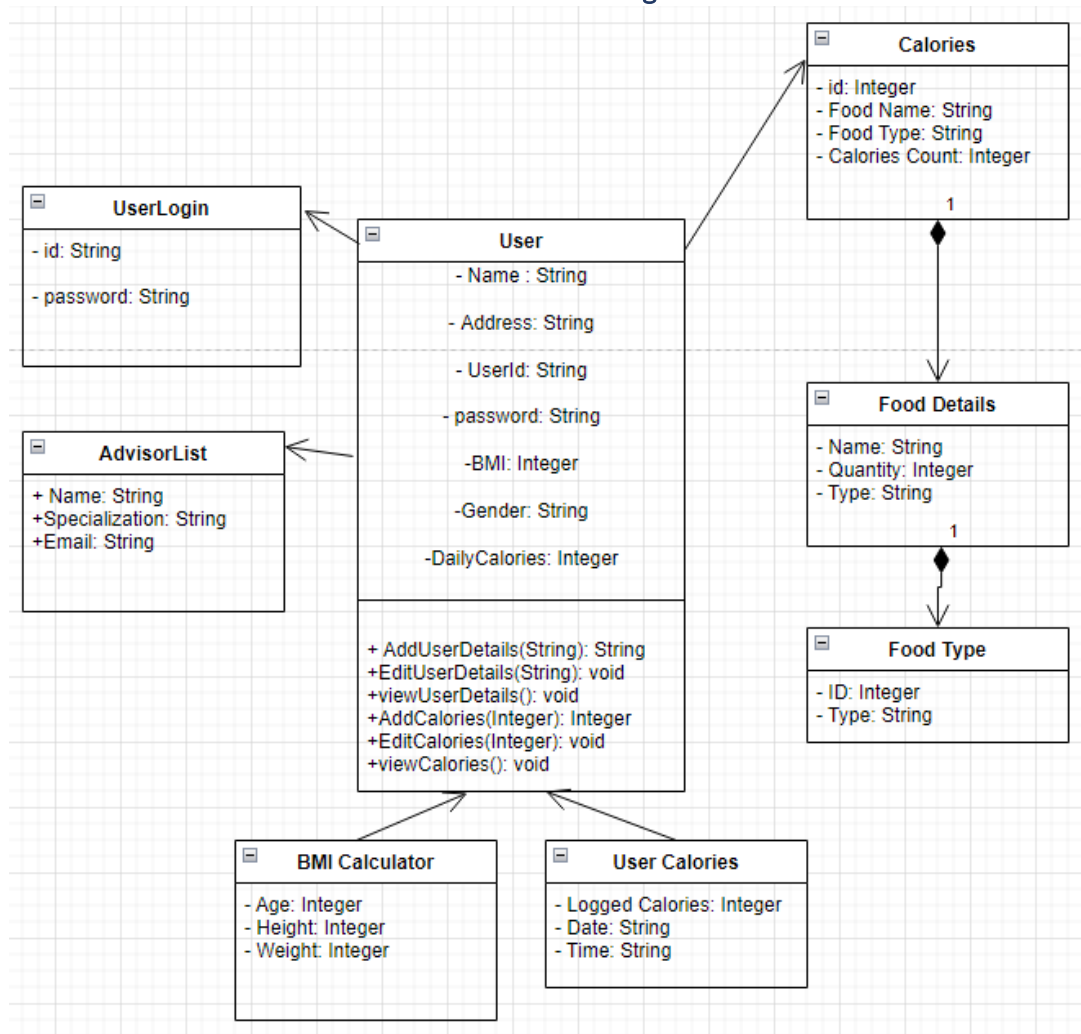
The team has decided to use a third-party API platform for fetching the list of food items and their nutritional information. For this we have come up with few data collections as below from internet sources,

- <https://platform.fatsecret.com/api/?gclid=Cj0KCQjwz7uRBhDRARIsAFqjulmoVXbq8A2S8V4cxT9YC N26Dx6jjuCRI2TnviwCPQt2qxv qwMwu5UaAru9EALw wcB>
- <https://spoonacular.com/food-api/pricing>
- <https://fdc.nal.usda.gov/api-guide.html>

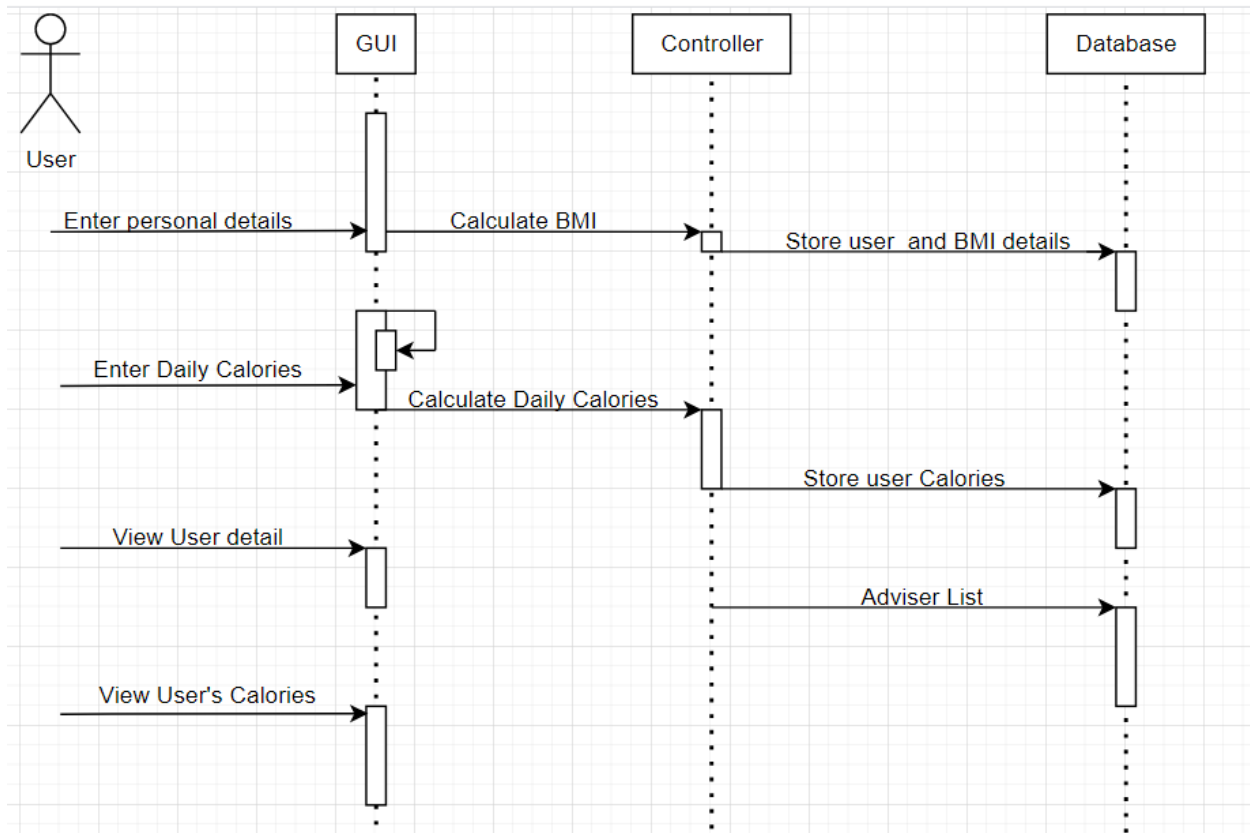
We are still in the phase of analyzing which API best suits our application's motive and yet to finalize.

5. High Level UML Diagram

UML Class Diagram



UML Sequence Diagram



Summary: First time users will spend a short amount of time going through filling in their personal data which includes the optional height and weight. If height and weight are filled we calculate BMI in the backend and send all the data to the user's specific database. After logging into our website users are able to record and store their daily calories and store their information into their user account's history where they're able to access/edit/view their details.

6. Key Risks and Mitigation Plan

- **Time to learn.** It may take time for a team to study a new programming language, piece of software. We are pushing ourselves hard to engage and learn new concepts in a scheduled manner and the knowledge is shared with the team during weekly meetings.
- **Poor team cohesion.** Inner conflicts can cause a team to put spokes in each other's wheels. In order to avoid that we are making sure that transparency is maintained. Contribution by each team member is fully transparent, so as to track and adjust if anything goes wrong. Also, this way the team collaboration is achieved at the maximum.
- **Security Risk.** This is the most common risk factor as the users might be sensitive with their personal information. We are trying our best to mitigate this risk by encrypting the passwords and we are enabling admin access to the owner of application, who will be solely responsible for any addition/deletion of data.
- **Time management risk** – This is again a most common issue that might occur while working in a group. But our team is very much organized in this aspect. We set up deadline for each of the tasks that an individual has to do and since this is done, the team is sparing some bandwidth each day to accomplish their target.

7. Project Management

The Milestone 2 had two sub tasks to be accomplished. One, being the ***SW architecture design*** and the other is ***Implementation of Vertical SW Prototype***.

As soon as the Milestone was assigned, we had set the deadline for each of the sub-tasks as 03/11 and 03/17 respectively.

For Sub task 1 (SW architecture design), below are the procedure we followed:

- We created google doc containing all the topics that needs to be covered as part of this task
- The team was split in pairs and was assigned with each topic.
For example,
 - The Data Definitions and Functional requirement (V2) was taken care by Aung
 - UI Mockups and Storyboards was taken care by Tunni (Front end lead as suggested by instructor)
 - High level Architecture, Database Organization done by Myat with the help of Jingxing
 - High Level UML Diagrams was taken care by Prathiba and Christopher
 - Key risks were identified by Aung and mitigation plan for those were discussed.
- If team had any discrepancy with their topics, they were discussed transparently with the team via Discord.
- The team met on 03/09 before the deadline for sub task 1 to review the team's work on each topic and each team member actively gave feedback to their peers.
- The final draft version was formatted and reviewed by the entire team before submission

For Sub task 2 (Implementation of Vertical SW Prototype), below are the procedure followed.

- This task was started on completion of sub-task 1, where we decided that the login page to our application would act as homepage for Milestone 2.
- The GitHub master (Myat) created a development branch out of our Master branch in GitHub, in order for the team to collaboratively work.
- The backend lead (Jingxing) created the Database schema for userDetails with login information for the user using Robo3T tool.
- The frontend lead (Tunni) created the UI design for the login page using CodeSandbox and Christopher support was very much appreciated in this. (She made sure that the design is responsive- using bootstrap)
- In order to have a CI-CD pipeline, Scrum master (Aung) and Team lead (Prathiba) made sure that the deployment window is clear for the application to be hosted and tested thoroughly.
- The team met on 03/15 to share a brief knowledge about their understanding upon working on the technology so far and have integrated between the frontend and backend successfully.
- The implementations were then tested with multiple test cases, and after they all passed, the code was merged to master branch and deployed.

Note: The MoM (Minutes of meeting) was shared within the team by the scrum master after each team meeting and the transparency of each team member's work was maintained throughout the Milestone 2