Software Engineering CSC648/848

iPlate

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 4"
May 4, 2022

"Application URL"
http://34.127.94.100/

Revision History Table

Revision ID	Revision Date	Revised By

1. Product summary

Name of the application: iPlate

URL to our product: http://34.127.94.100/

Major functionalities:

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
User Login	The already registered users must be able to login and work on iPlate app efficiently	1
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 revisit their past meals (including searching history based on date, displaying the plate with food group's ratio based on their corresponding calorie)	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 mock-up for ratio representation	1

Our Mission and goal:

- ★ *"Track the foods you love and live healthy!!"*
- ★ "Learn about the foods you're eating and keep your calories within your daily budget"
- ★ "Log your meals and track all your macro and micronutrients"
- ★ "Be confident that the food you log has the correct nutrition data"

Unique Features:

- **No calorie counting:** iPlate uses the MyPlate method that is provided by the U.S. Department of Agriculture. Instead of focusing on calories, iPlate allows users to focus on the ratio of vegetables, carbs, protein, and fruit that they consume. By the amount that the user records in iPlate, we will calculate the ratio and display it with a plate design when they search in history. The plate is splitted into four areas that represent each food group, and by looking at the plate, users would

have an idea of which food groups they should intake more, and which food group they should intake less in order to get a balanced plate.

- **Prevent eating disorder:** Since iPlate is focusing on the ratio of each food group, not the calorie, it prevents the risk of developing an eating disorder. Eating disorders can start in many scenarios, one of the common cases is when people try to lose some weight and start to count their calorie intake for each meal. Counting calories is an effective way of losing weight, but it's not the best way of building our eating habits. Many people that tried calorie counting ended up messing up their relationship with foods, which they choose based on their calorie, but not what they truly enjoyed. Through the calorie restriction, some fall into the binge eating loop. iPlate focuses more on how users can enjoy their food while developing healthy eating habits.

- **Easy to manage, life-long eating habits:** By using iPlate, we hope users can enjoy the process of building healthy eating habits. As we mentioned earlier, iPlate focuses on the ratio not the calorie, which is easier for users to manage what they eat because they won't have to deal with the calculating process. It's more manageable and easier to access in users' daily life. After using iPlate, we also aim that users could transform this into a life-long eating habits where they won't require a food weighing scale, instead they can put their food on their plate according to iPlate's plate display.

2. QA testing

a) Unit Test

We have used the Jest to write our unit test cases and for the testing purpose on the whole. Unit test cases have been written for 2 major functionalities so far.

1. The post request for Signup is sent by passing mockup data of a valid username, password and confirm password (same as password above), the result is PASS with 100% coverage.
2. The post request for Login is sent by passing mockup data of a valid username, password, the result is PASS with 100% coverage.

Below is the screenshot for test coverage and test result,

```
PASS ./unit.test.js
  Test POST /api/login
    when user name and password is passed in correctly
      ✓ should respond with (154 ms)
  Test POST /api/signUp
    when user name, password and confirmPassword are provided
      ✓ should respond with (42 ms)
Test Suites: 1 passed, 1 total
Tests:       2 passed, 2 total
Snapshots:   0 total
Time:        3.208 s
Ran all test suites.
```

Note: The Unit test cases for remaining functionalities will be written in Milestone 5.

b) Integration Test

The template for integration testing was fetched from below link,

https://www.guru99.com/download-sample-test-case-template-with-explanation-of-important-fields.html

Please find below the attached excel sheet with Integration test cases and results,

CSC648-Section01-T
eam02-Integration T

The below functionalities are covered as part of Integration testing with a test result **PASS**.

User Register: Allows the user to register to iPlate

- TC_001 : Verify on entering valid user id and password, the customer can register with
- TC_002: Verify on entering different password on confirm password field
- TC_003: Verify on entering already existing user id/email

User Login: Allows the user to login to iPlate with created credentials.

- TC_004: Verify on entering valid user id and password, the customer can login
- TC_005: Verify on entering invalid user id
- TC_006: Verify on entering correct user id, but wrong password.
- TC_007: Verify Logout operation

Meal Record Function: Allows the user to record their meal intake for a particular date and type.

- TC_008: Verify the meal recording functionality by giving correct values in the dropdown and entering correct quantity of food intake for particular date.
- TC_009: Verify the meal recording functionality by not selecting a Meal type
- TC_010: Verify the meal recording functionality by not entering the quantity of intake

Meal History View: Allows the user to view their present and past recorded meal history

- TC_011: View the food history and the analysis chart based on calorie calculation for a valid date.
- TC_012: View the food history and the analysis chart based on calorie calculation for invalid date.

Homepage: Allows the user to know about iPlate, our mission and vision.

- TC_013: Verify the Homepage navigation properly

About Page: Allows users to get to know our team.

- TC_014: View the food history and the analysis chart based on calorie calculation for a valid date.

3. Code Review

a) Coding Style:

- The code is maintained in such a way that it is self-explanatory.
- The Naming convention in the code for functions and variables are well maintained as per the standard and is consistent throughout our project file.
- The code is properly indented for better understanding
- Each of the functions are well documented as to know what exactly it does.
- Comments are provided for each of the critical functions in the code.

b) Code Review:

Once the code has been developed by the frontend team, the backend team was assigned for the code review and vice versa.

Below are the Git Pull request urls that we fetched from GitHub for each code review requests and approvals,

- https://github.com/CSC-648-SFSU/csc648-spring22-01-Team02/pull/8
- https://github.com/CSC-648-SFSU/csc648-spring22-01-Team02/pull/7
- https://github.com/CSC-648-SFSU/csc648-spring22-01-Team02/pull/6
- https://github.com/CSC-648-SFSU/csc648-spring22-01-Team02/pull/5
- https://github.com/CSC-648-SFSU/csc648-spring22-01-Team02/pull/4
- https://github.com/CSC-648-SFSU/csc648-spring22-01-Team02/pull/3
- https://github.com/CSC-648-SFSU/csc648-spring22-01-Team02/pull/2
- https://github.com/CSC-648-SFSU/csc648-spring22-01-Team02/pull/1

4. Self-check: Adherence to original Non-functional specs

Below is the list of non-functional Specs that we mentioned earlier with their current status,

- (High priority) Application UI should be responsive so users can use it on their mobile or their laptop - **DONE**
- (High priority) User-friendly UI - **DONE**
- (High priority) Users' data must be protected - **ON TRACK**
- Database: keep the general information for each food group - **DONE** (but are maintained in frontend)