

Nutritional Calculator

Iteration 2 | 02/23/2018

Hector Garza
Alexander Kermani
Songyang Li
Ryan Lira
Thiphavanh Phoutthivongsack
Parijat Singh

Contents

1. Organization/Responsibilities	3
2. List of Requirements	3
3. Iteration Plan	3
4. Domain Model	
a. Brainstorming	4
b. Diagram	4
5. Use Case Model	
a. High Level Use Case	5
b. Use Case Diagram	6
c. Extended Use Cases	7
d. Traceability Matrix	10
6. Sequence Diagrams	11
7. Class Diagram	

1. Organization/Responsibilities

System – A nutritional calculator to check daily calorie intake, manage a user's diet, and notify the user when they are failing/succeeding at reaching their calorie goals

Tasks	Reqs	Responsible	Iteration 1 2/23/2018	Iteration 2 3/23/2018
R3 Food Controller		Thiphavanh	X	
R3 Food Item		Songyang	X	
R3 Food Entry		Songyang	X	
R2 Goal Controller	R1	Ryan	X	
R2 Goal	R1	Alexander	X	
R2 Calculate	R1	Parijat	X	
User View		Hector		X
R2 User		Hector		X
R2 Database Manager	R2,R3	Thiphavanh		X
R2 Login View		Parijat		X
R2 Save/Load	R2,R3	Parijat		X
R2 Add/Delete Items	R3	Ryan		X

<i>Hector Garza</i>	Sequence diagram designer	Wednesday
<i>Alexander Kermani</i>	Organizer for meetings / Plan and requirements doc	Tuesday
<i>Songyang Li</i>	Plan and requirements doc / Coder	Wednesday
<i>Ryan Lira</i>	Class diagram designer / Coder	Tuesday
<i>Thiphavanh Phouthivongsack</i>	Sequence diagram designer / Class diagram designer	Tuesday/ Wednesday
<i>Parijat Singh</i>	Use case diagram designer / Coder	Tuesday/ Wednesday

2. List of Requirements for SUD

- R1: The system will allow the user to **create a new account**.
 - R1.1: The system will allow the user to **log into their account** and view their previously added entries and their goal
 - R1.2: The user will also be able to **log out of their account**.
 - R1.3: The system will allow the user to **save account details**, such as the previous items and goals of the account.
 - R1.4: The system will allow the user to **load the last account details** saved.

- R2: The system will allow users to **add a new goal**.
 - R2.1: Setting a new goal will **calculate the recommended caloric intake** (how many calories the user should aim for daily).
 - R2.2: The user can **edit their goal**, which will make the old goal deprecated.
- R3: The user will be able to **add a new food entry** to their food tracker.
 - R3.1: The user can **search for a food item** from the database to make a new entry they wish to add to the food tracker.
 - R3.2: The user can **add a new food item to the database** and that will be used to create a new food entry in the food tracker
 - R3.3: The user will be able to **delete food entries** from their food tracker.

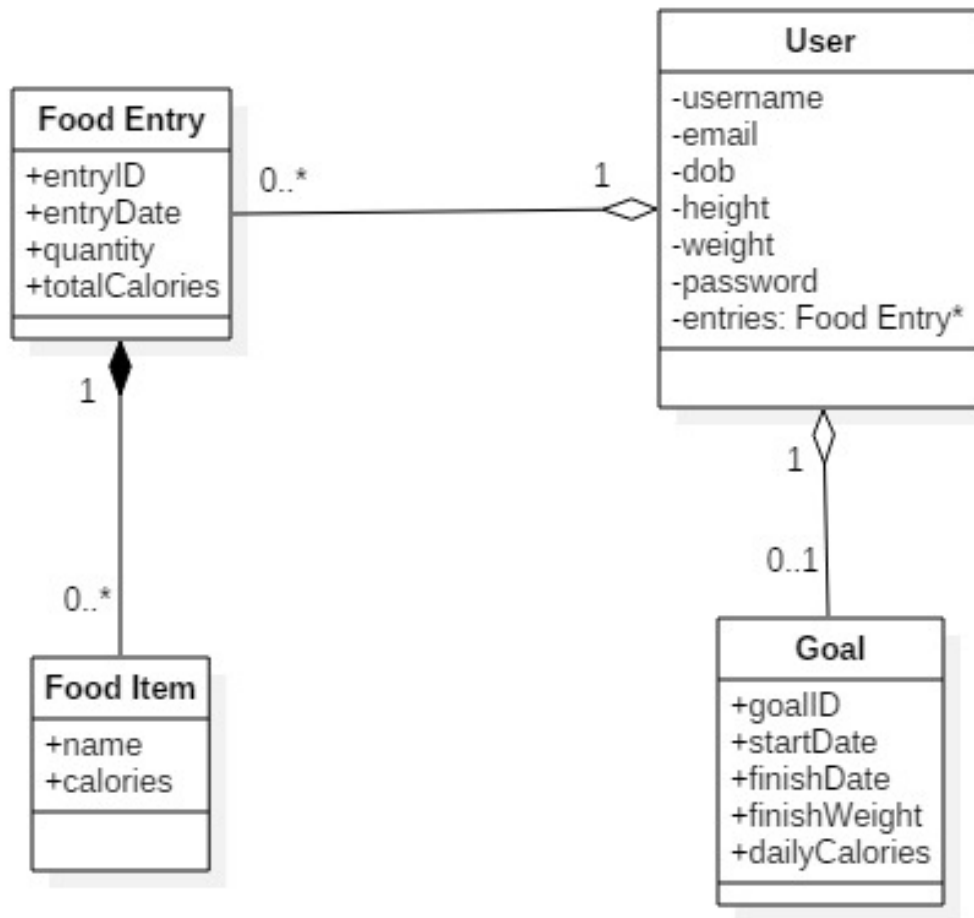
3. Iteration Plan

- 1st iteration (February 23): we will allow the user to add items and calculate the daily nutritional requirements to reach their goals.
- 2nd iteration (March 23): we will add a database that users can add items from, instantly including all of the nutritional data from it.
- 3rd iteration (April 15): we will add an interactive GUI to the system, as well as automated testing of program features

4. Domain Model - Brainstorming

- Food
 - Food Entries - When did they eat it/how much
 - Food Items – What food is it
- Goal
- User – to login/save data in database

5. Domain Model – Classification



6. Use Case Model – High Level Use Case

- Use Case 1: **Sign Up**

TUCBW User selects the “Sign Up” option on the login view
 TUCEW User seeing confirmation/decline of registration

- Use Case 2: **Log In**

TUCBW User selects the “Log in” option on the login view
 TUCEW User seeing the website main page or an error if the login was unsuccessful

- Use Case 3: **Log Out**

TUCBW User selects the “Log Out” option

TUCEW User sees the login view

- Use Case 4: **Save Data**

TUCBW User selects the “Save” option

TUCEW User can see confirmation of their data being saved

- Use Case 5: **Load Data**

TUCBW User selects the “Load” option

TUCEW User sees a confirmation that their previously saved data has been loaded

- Use Case 6: **Add Goal**

TUCBW User selects the “Add Goal” option

TUCEW User can see their goal and daily caloric intake recommendation

- Use Case 7: **Edit Goal**

TUCBW User selects the “Edit Goal” option

TUCEW User can see their updated goal and new daily caloric intake

- Use Case 8: **Add Food Entry**

TUCBW User selects the “Add Food Entry” option

TUCEW User can see a new entry in the food tracker with an item name, quantity, and calories

- Use Case 9: **Delete Food Entry**

TUCBW User selects the “Delete Food Entry” option

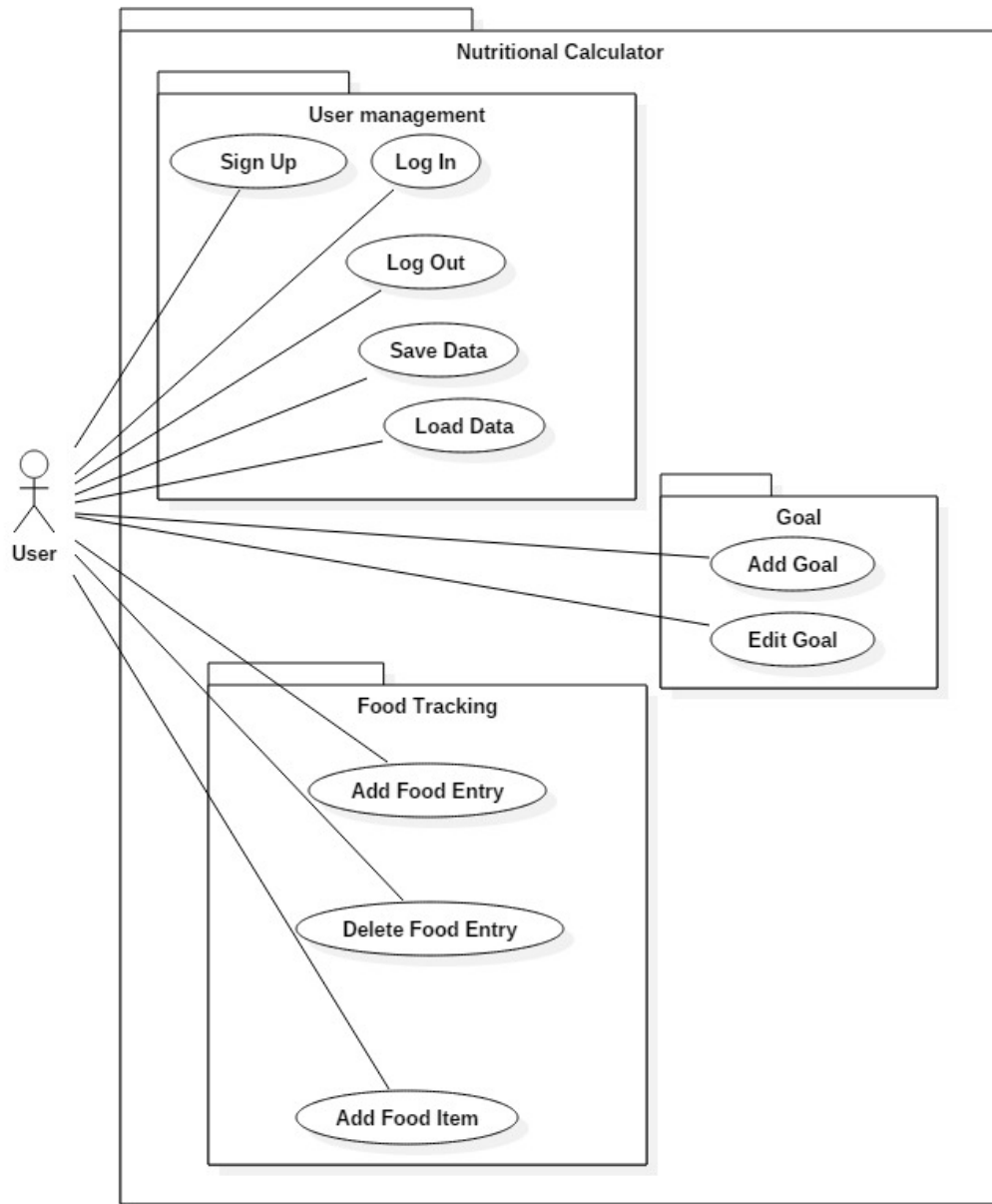
TUCEW User can see an updated entry in the food tracker (name, quantity, and calories)

- Use Case 10: **Add Food Item**

TUCBW User selects the “Add Food Item” option

TUCEW User sees confirmation that the food item has been added to the database

7. Use Case Model – Diagram



8. Use Case Model – Extended Use Case

Use Case 1: **Sign up**

	0. User sees the Login view.
1. TUCBW User selects the “Sign Up” option on the login view	2. The login view displays the fields for username and password.
3. User inputs their username into the fields password, and selects the confirm option	4. the login view passes the user’s information to the system to either:

	<ul style="list-style-type: none"> a. create the user's account in the database and notify the user of the success b. notify the user an account with that username already exists
5. TUCEW User seeing confirmation/decline of registration	

Use Case 2: Log In

	0. User sees the Login view
1. TUCBW User selects the "Log in" option on the login view	2. The login view displays the fields for username and password.
3. User inputs their username/password into their respective fields and clicks "Log In" option	2. The account details are checked against the accounts in the database, and the website GUI displays either: <ul style="list-style-type: none"> a. The main website b. Incorrect login info error
3. TUCEW User sees either: <ul style="list-style-type: none"> a. The website main page. b. "Wrong username/password" if login was unsuccessful 	

Use Case 3: Log Out

	0. User sees the user/main view.
1. TUCBW User selects the "Log Out" button	2. The website GUI prompts if the user is sure if they want to log out
3. The user clicks the "Yes" option	4. The system logs out the user and displays the login page
5. TUCEW User sees the login view	

Use Case 4: Save Data

	0. User sees the user/main view.
--	----------------------------------

1. TUCBW User selects the “Save” option	2. The database manager updates and saves the user’s information, and the website GUI displays a “Data Saved” prompt.
3. TUCEW User can see confirmation of their data being saved	

Use Case 5: Load Data

	0. User sees the user/main view.
1. TUCBW User selects the “Load” button	2. The user view pulls the user’s information from the database and displays the previously saved items and goals.
3. TUCEW User sees a confirmation that their previously saved data has been loaded	

Use Case 6: Add Goal

	0. User sees the user/main view.
1. TUCBW User selects the “Add goal” option.	2. The user view shows input fields for the goal calorie count and goal duration
3. User enters the calorie count and the time they wish to achieve it	4. The user view displays that a goal has been set and displays the recommended daily caloric intake
5. TUCEW User can see their goal and daily caloric intake recommendation	

Use Case 7: Edit Goal

	0. User sees the user/main view.
1. TUCBW User selects the “Edit Goal” option	2. The user view shows input boxes for the goal calorie count and goal duration
3. User enters the calorie count and the time they wish to achieve it.	4. The user view displays that a new goal has been set and displays the new recommended daily caloric intake
5. TUCEW User can see their updated goal and new daily caloric intake recommendation	

Use Case 8: Add Food Entry

	0. User sees the user/main view.
--	----------------------------------

1. TUCBW User selects the “Add Food Entry” option	2. The user view displays all available food items and a field for the user to select one
3. User enters the food item they wish to add to the calculator	4. The user view prompts the user for the quantity of the item to add
6. User inputs the quantity into the field	7. The user view displays options to either confirm the entry addition or decline the entry addition.
5. User either: a. Confirms the addition of the entry b. Declines the addition of the entry	6. The system adds the entry to the database and updates the food tracker table with the new entry
7. TUCEW User can see a new entry in the food tracker with an item name, quantity, and calories	

Use Case 9: **Delete Food Entry**

	0. User sees the user/main view
1. TUCBW User selects the “Delete Food Entry” option	2. The user view displays a list of food entries they have entered and a field for the user to select one
3. User enters the food entry the wish to delete from the calculator	4. The user view displays options to either confirm the entry deletion or decline the entry deletion.
5. User either: a. Confirms the deletion of the entry b. Declines the deletion of the entry	6. The system deletes the entry to the database and updates the food tracker table
7. TUCEW User can see their food tracker without the entry they wish deleted	

Use Case 10: **Add Food Item**

	0. User sees the user/main view
1. TUCBW User selects the “Add new food item” option	2. The user view displays input fields for food item name and calories
3. The user types the name and calories into the fields	4. The System adds a new food item it to the database
5. TUCEW User sees confirmation that the food item has been added to the database	

--	--

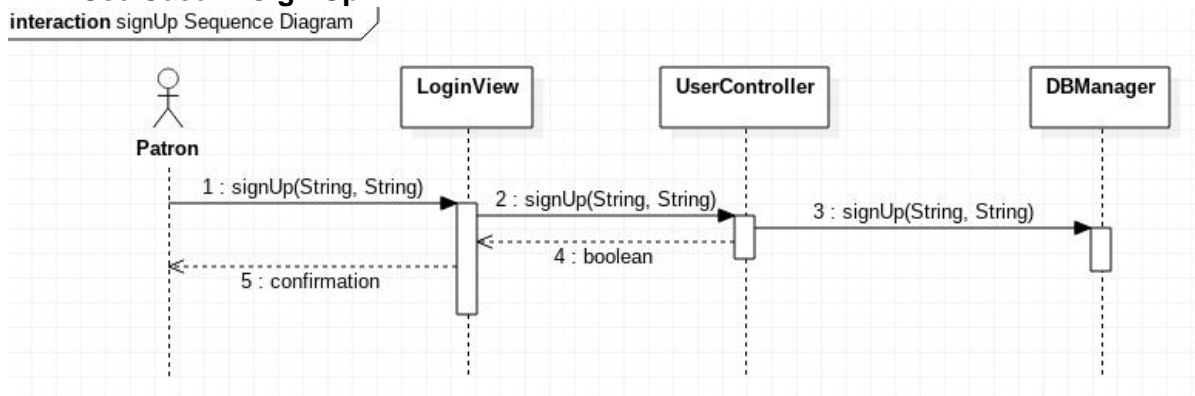
9. Use Case Model – Traceability Matrix

	Weight	UC1	UC2	UC3	UC4	UC5	UC6	UC7	UC8	UC9	UC10
R1	3	X	X	X	X	X					
R2	1						X	X			
R3	2								X	X	X
	Score	3	3	3	3	3	1	1	2	2	2

10. Use Case Model – Sequence Diagrams

- Use Case 1: **Sign Up**

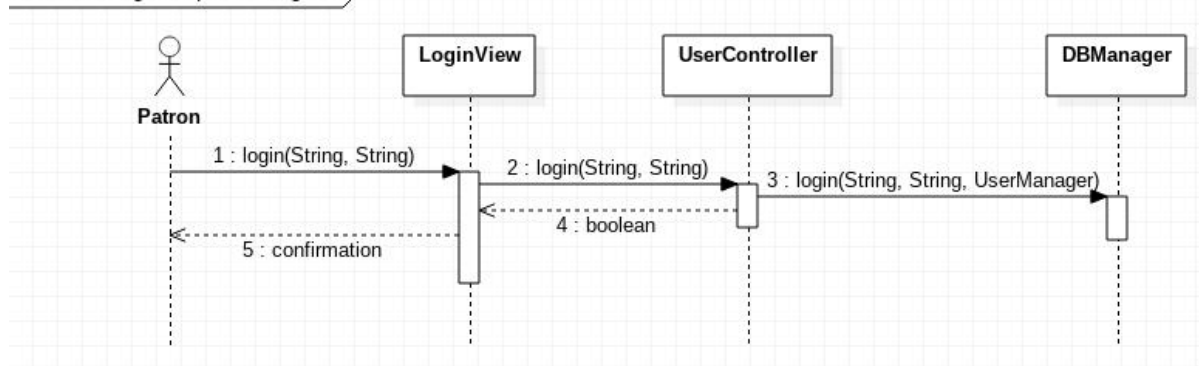
interaction signUp Sequence Diagram



- Use Case 2: **Log In**

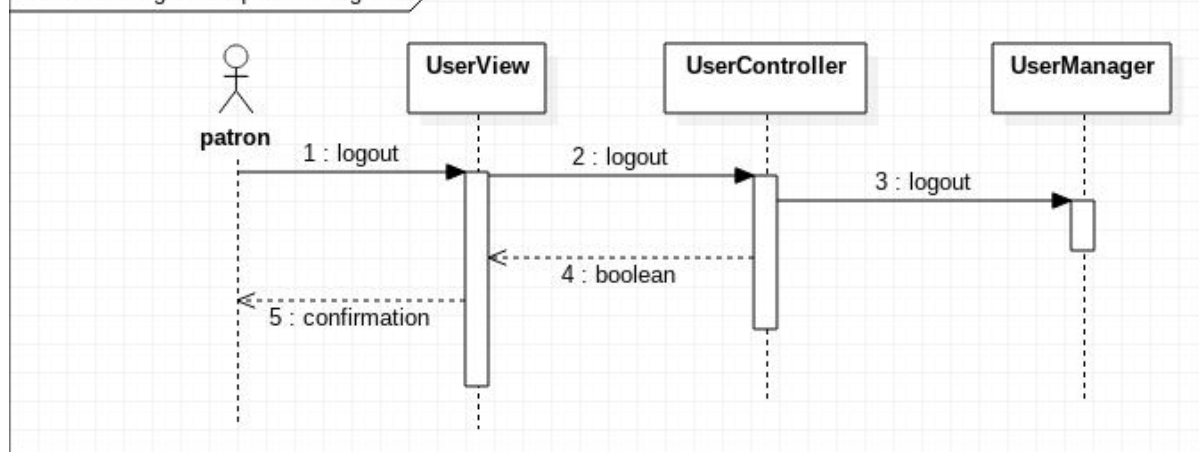
Nutritional Calculator Iteration 2

interaction login Sequence Diagram



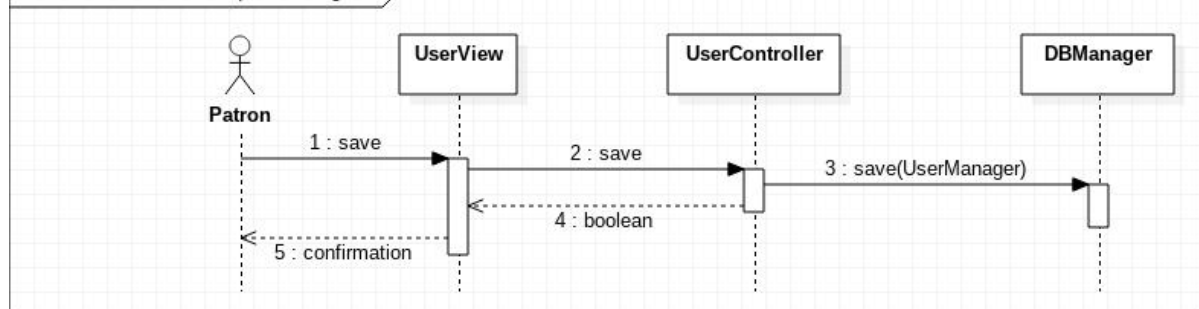
- Use Case 3: **Log Out**

interaction logout Sequence Diagram



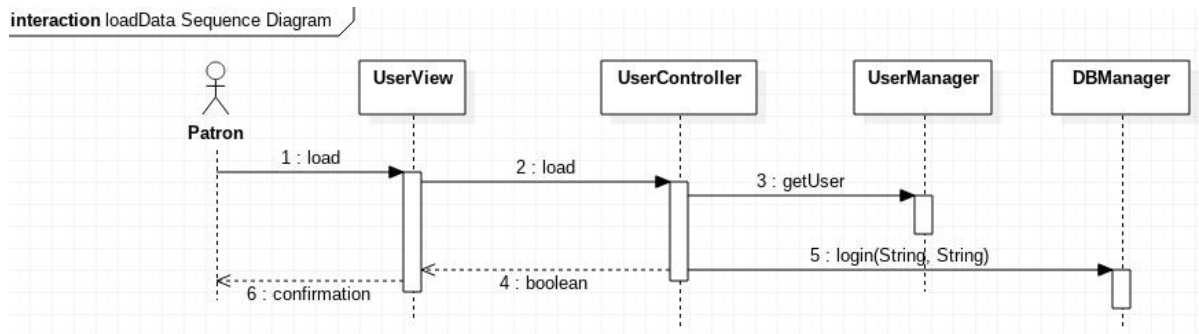
- Use Case 4: **Save Data**

interaction saveData Sequence Diagram



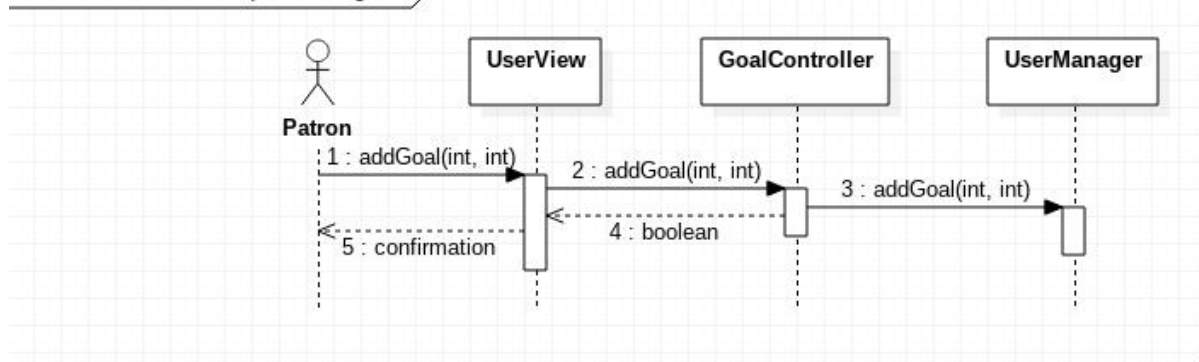
- Use Case 5: **Load Data**

Nutritional Calculator Iteration 2



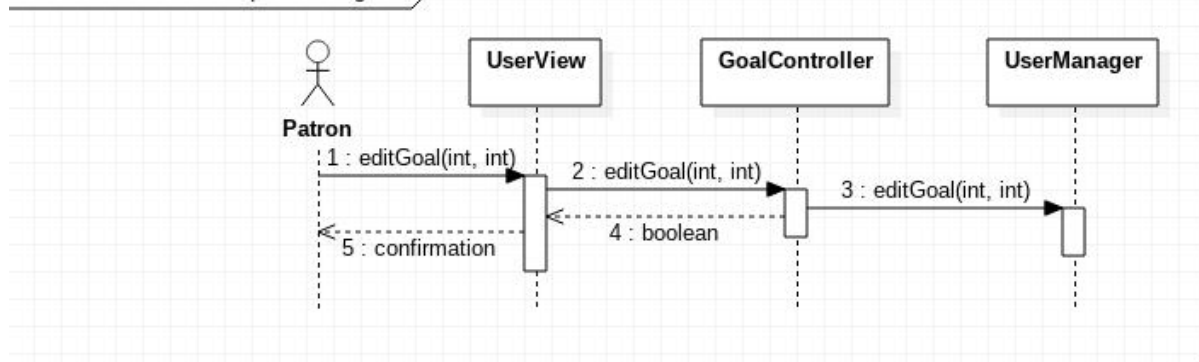
- Use Case 6: **Add Goal**

interaction addGoal Sequence Diagram



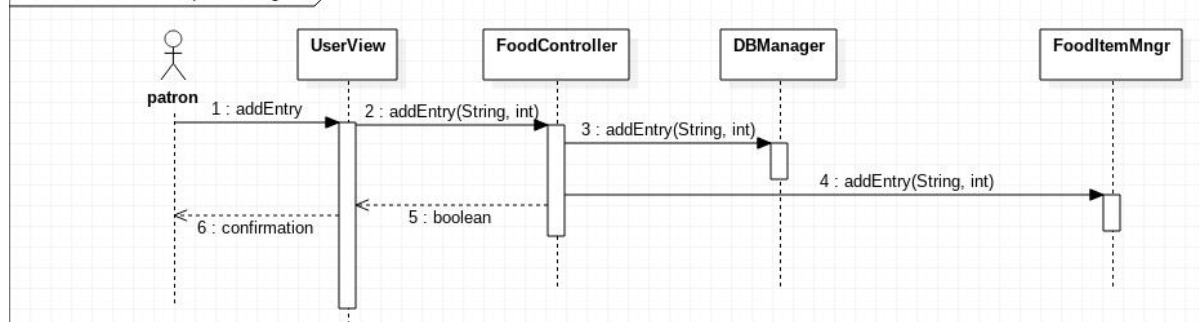
- Use Case 7: **Edit Goal**

interaction editGoal Sequence Diagram

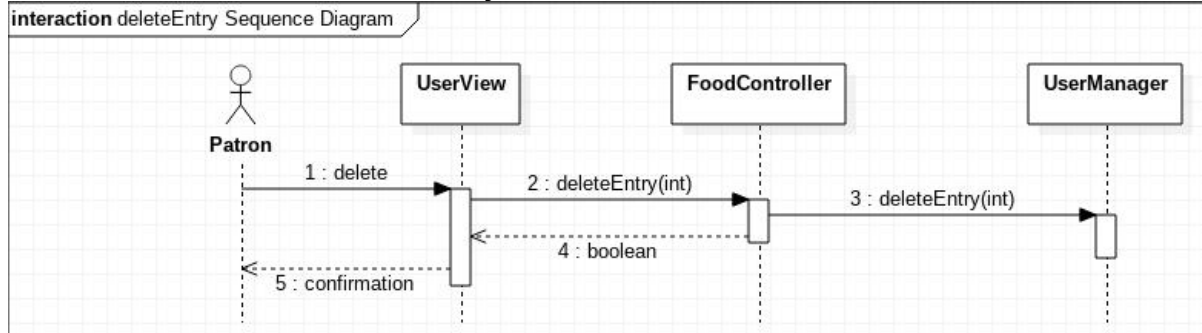


- Use Case 8: **Add Food Entry**

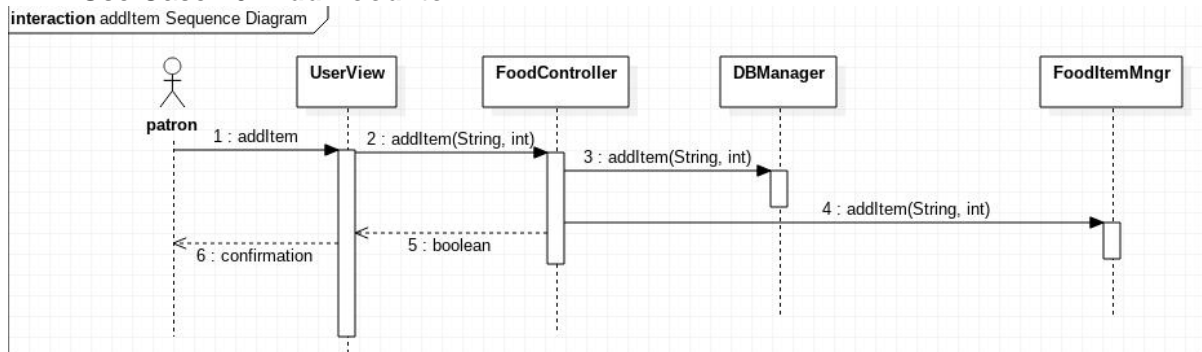
interaction addItem Sequence Diagram



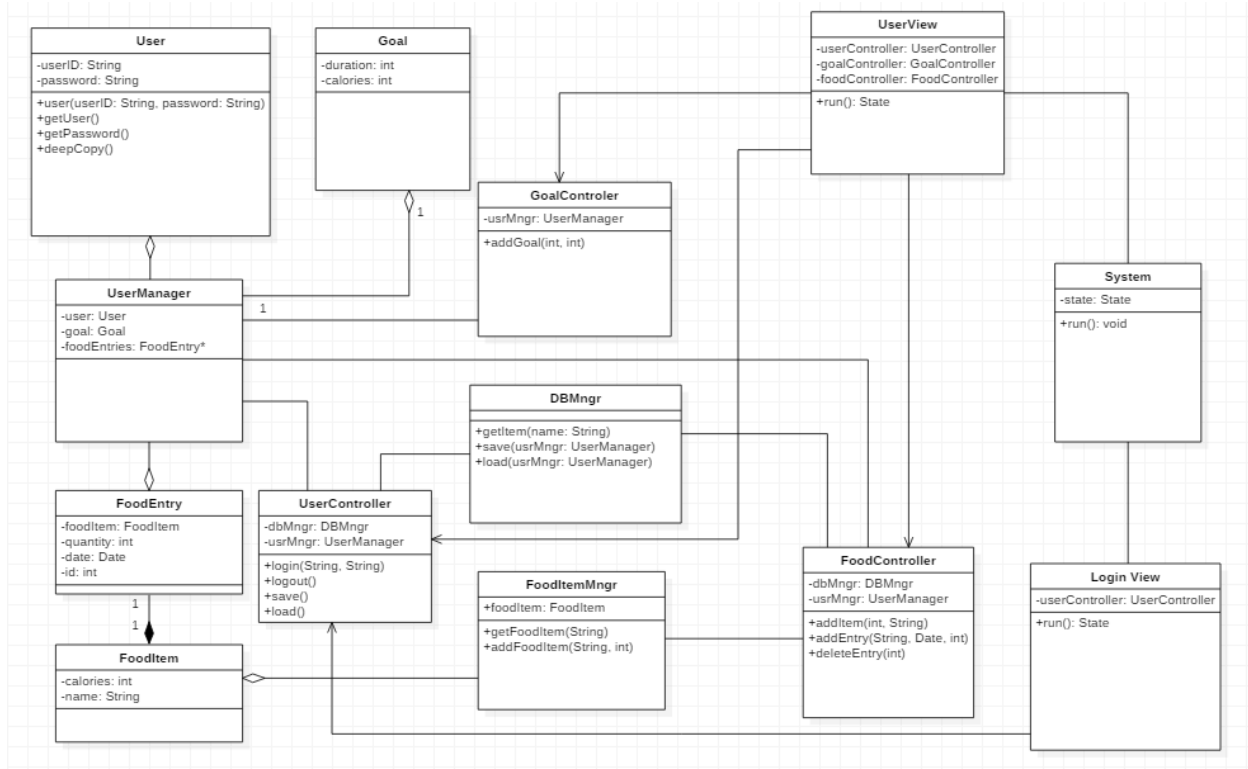
- Use Case 9: Delete Food Entry



- Use Case 10: Add Food Item



11. Design Class Diagram



11. Code

Located at public repository:

<https://github.com/3354Group8/Nutritional-Calculator-Code>