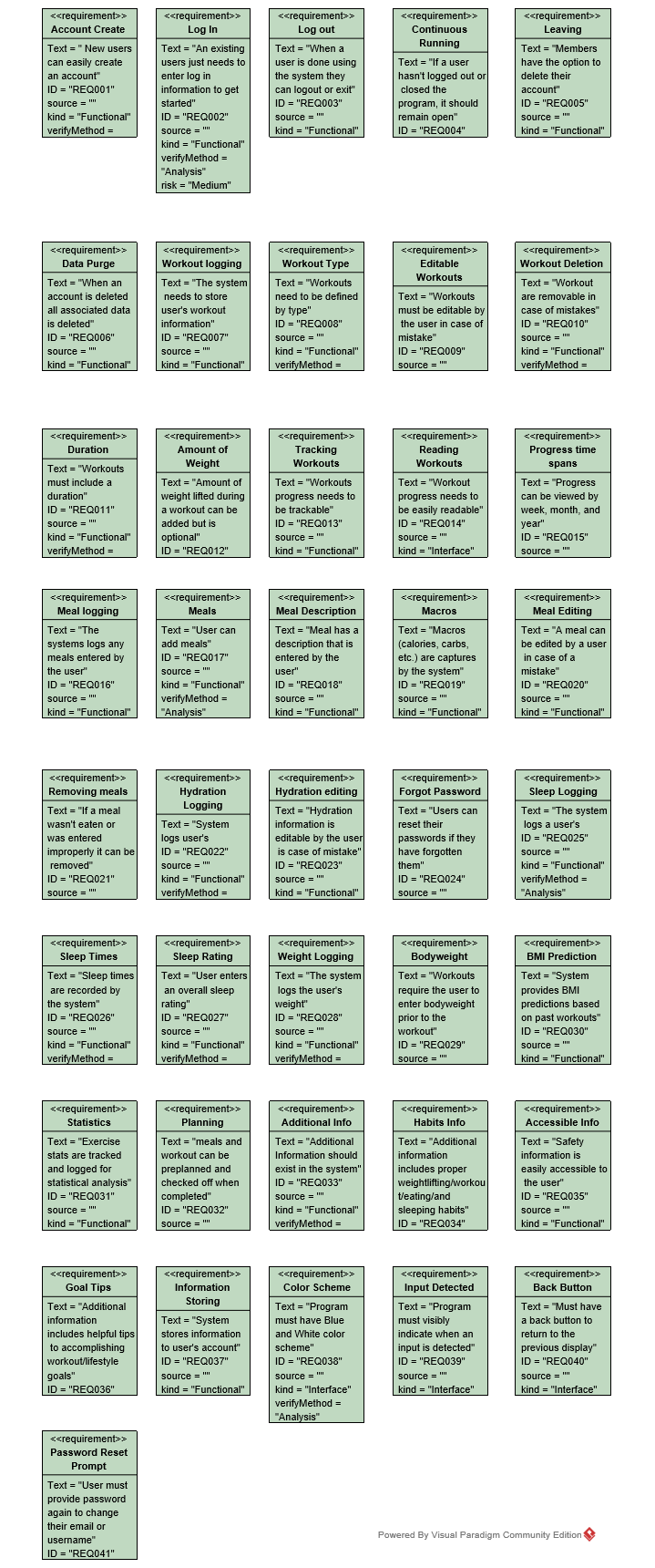
FitLife App Analysis

Requirements & Use Cases

**Use Case** UC-ACCT-01 Create Account

**ID:** UC-ACCT-01

**Scope:** User Account

**Level:** User Goal

**Stakeholders and Interests**

Customer

-Wants fast and easy account creation

-Expects to give Username, email, and password

Admin

-Expects user info to be stored for mass-viewing

**Pre-Conditions**

User is not logged in

User is on the home screen

**Post-Condition**

An account has been created

**Main Success Scenario**

1. User selects the "Create an account" option
2. System displays input boxes asking for email, username, and password
3. User inputs the required field
4. System temporarily stores the data
5. System sends an email to the user's email to verify the email
6. User follows the link in their email to verify their email
7. System creates a new account with the data the user entered
8. System sends the user back to the home screen

**Extensions**

3a. Entered username is already linked to a different account

1. System will display "Username already taken"

2. System will return to the start of step three

3b. Entered Password does not fulfill the requirement

1. System displays a message about what requirement was not fulfilled

2. System returns to the start of step 3

6a. User does not verify their account within 48 hours

1. Delete all temporarily stored data

6b. The email owner follows the link saying they did not sign up.

1. Delete all data temporarily stored.

**Use Case** UC-ACCT-02 Delete Account

**ID:** UC-ACCT-02

**Scope:** User Account

**Level:** User Goal

**Stakeholders and Interests**

Customer

-Wants an easy process

-Wants their data to be removed

**Pre-Conditions**

User is logged in

User is at the home screen

**Post-Condition**

User's account has been deleted

**Main Success Scenario**

1. User navigates to the account settings

2. System displays details about user account

3. User selects "Delete Account" option

4. System asks to verify the user's choice

5. User confirms their intent

6. System logs the user out

7. System deletes the user's account

**Extensions**

1-4a. Anytime user visits another page

1. Stop process

1-4b. Anytime user closes the application

1. Stop Process

5a. User does not wish to delete account

1. Return the system to it's state after step 2

**Use Case** UC-ACCT-03 Edit Account

**ID:** UC-ACCT-03

**Scope:** FitLife

**Level:** User Goal

**Stakeholders and Interests**

Customer

-Wants to be able to edit username, password, email, and avatar

**Pre-Conditions**

User is logged in

User is at home page

**Post-Condition**

User has edited desired detail

**Main Success Scenario**

1. User navigates to their account settings
2. System displays all account info
3. User selects desired setting to change
4. System handles the change
5. System stores the changed setting

**Extensions**

4a. User wishes to edit their email

1. User enters their password to validate identity

1a. User enters wrong password

1. System displays an error message and asks user to re-enter their password

2. System returns to step 4a-1

2. User enters the new email address

3. System removes the email's association with the account

3a. There are no more accounts associated with that email

1. System removes that email from the database

4. System associates the new email with the account

4a. The email is not in the database

1. The new email is added to the database

4b. User wishes to edit their password

1. System requests that the user enter their old password, and their new password twice

1a. Old password does not match current password

1. System displays an error message and returns to step 4b-1

1b. The entries of the new password do not match

1. System displays an error message and returns to step 4b-1

1c. The user's new password does not meet the password requirements

1. System displays an error message and returns to step 4b-1

1d. The user's new password is the same as the user's old password

1. System displays an error message and returns to step 4b-1

2. System sets the user's new password

4c. User wishes to edit their username

1. User enters their password to validate identity

1a. User enters wrong password

1. System displays an error message and asks user to re-enter their password

2. System returns to step 4c-1

2. User enters a the new username

2a. Entered username is already taken

1. System displays an error message and returns to step 4c-2

3. System removes the old username's association with the account

4. System associates the new username with the account

4d. User wishes to edit their avatar

1. System requests the user to upload a file

2. User uploads a file

2a. Image file is an incorrect ratio

1. System displays the image, with a box representing the appropriate image ratio

2. System crops the image

3. System deletes the account's previous avatar

4. System sets the account's image to the uploaded avatar

**Use Case** UC-ACCT-04 View Info

**ID:** UC-ACCT-04

**Scope:** FitLife

**Level:** User Goal

**Stakeholders and Interests**

Customer

-Wants to be able to easily view info about a specific subject

**Pre-Conditions**

User is logged in

User is at the home screen

**Post-Condition**

User is viewing the information about a health subject

**Main Success Scenario**

1. User selects the info screen from the home page
2. System displays the main page of the information section
3. User selects the topic he wishes to know more about
4. System displays information about the topic, as well as any subtopics that might exist
5. User Selects a sub topic to view
6. System displays information about the subtopic

**Extensions**

1. User wishes to stop viewing information
2. User may navigate to a different page
3. User may close the application
4. User may log out

**Use Case** UC-ACCT-05 Reset Password

**ID:** UC-ACCT-05

**Scope:** FitLife

**Level:** User Goal

**Stakeholders and Interests**

Customer

-Wants to be able to reset their password if forgotten

-Expects password reset to utilize their email

**Pre-Conditions**

User is not logged in

User is at the sign in screen

**Post-Condition**

User has reset their password

**Main Success Scenario**

1. User presses the "Forgot Password" button
2. System asks for the user's email or username
3. System creates a temporary password that can be used to sign in to the account
4. System sends a message to the associated account's email address that contains a temporary password
5. User signs in using their temporary password
6. System asks the user to input a new password twice
7. User inputs a new password twice
8. System changes the account's password to the new password
9. System removes the temp password

**Extensions**

2a. User enters a username or email that is not associated with an account

1. System sends an error message to the user and returns to the beginning of step 2

3-4a. User signs in using their old password

1. System deletes the temporary password so it can't be used

3a. Temporary password already associated with the account

1. Delete the old temporary password and replace it with the new one

5a. User has not signed in within 24 hours

1. System deletes the temporary password so it can't be used

7a. Input password does not match

1. Display an error message and return to step 7

7b. Passwords do not meet password requirements

1. Display an error message and return to step 7

7c. Passwords entered are the same as the account's current password

1. Display an error message and return to step 7

**Use Case** UC-ACCT-06: User wants to log-in `

**Scope:** FitLife

**Level:** user goal

**Primary Actor:** Registered User

**Stakeholders and Interests:**

- Registered User:

Wants accurate, fast entry, as registered user entered their correct account information with their corresponding password.

- Database:

Wants to receive correct login account information from user.

**Preconditions:**

User has registered an account and user information has identified.

**Success Guarantee (or Postconditions):**

User account information and password are correct. User can access into their account.

**Main Success Scenario (or Basic Flow):**

1. User open Its Fit application getting into the Log in page.
2. User entered their user name and corresponding password.
3. System recognized the user account and let user get into their main page.

**Extensions (or Alternative Flows):**

2a. User entered wrong account information of either username or password.

1. System signals error to the User, records the error, and enters a clean state.

2. System ask user to enter their log in information again.

**Use Case UC-PLAN-01:** Plan Workout

**Scope:** FitLife

**Primary Actor:** Registered User

**Stakeholders and Interests:**

- Registered User: Wants to have a simple and fast interaction with the system.

Wants to have the exact information stored in the system.

Wants to be able to plan a workout quickly.

**Precondition:** User is registered and has successfully logged in. User has added a workout.

**Postcondition:** Workout information is saved. Information is updated. Database is updated.

**Main Success Scenario:**

1. Registered user arrives at home screen

2. User chooses an existing workout

3. System displays information of the selected workout

4. User selects a workout

5. User clicks "Plan Workout" button

6. User repeats steps 2-4 until completed

7. System displays a series of choices for the user to enter, including:

the Duration of workout, the Date and Time of workout, and the

additional Note for the workout

8. User selects a category and enters information

9. Information is updated and stored in database

10. User is able to review workout before logout

11. User logs out

**Extensions:**

\*a. At any time, Systems fails:

1. User restarts the system

2. User logs in

3. User repeats the process of planning a workout

8.a User selected the wrong choice before saving

1. User clicks "cancel" to go to the previous page

2. User repeats the process

3. User clicks "Save" to save the information

4. Information is updated and stored in the database

2-9.a User decides to change information after saving

1. User clicks "Plan Workout" button

2. System displays a series of choices for the user to enter, including:

the Duration of workout, the Date and Time of workout, and the

additional Note for the workout

3. User selects and enters desired information

4. User clicks "Save" to save

5. System stores information

6. Information is updated and stored into the database

**Use Case UC-PLAN-02:** Plan Meal

**Scope:** FitLife

**Primary Actor:** Registered User

**Stakeholders and Interests:**

* Registered User: Wants to have a simple and fast interaction with the system. Wants to have the exact information stored in the system. Wants to be able to plan a meal quickly and easily.

**Precondition:** User is registered and has successfully logged in.

**Postcondition:** Meal is added to specified date selected. Information is updated. Database updated.

**Main Success Scenario:**

1. Registered user arrives at home screen.

2. User selects Nutrition on home screen.

3. User chooses a date from the list.

4. System displays info about meal such as protein, carbohydrates,

natural fats, and calories, as well as how much water.

5. User enters information about their meal.

6. User selects to add meal to that day.

7. Information is updated and stored in database.

8. User repeats steps 2-7 until finished.

9. User logs out and exits the program.

**Extensions:**

\*a. System fails at any given time:

1. User restarts system

2. User logs in

3. User repeats process of planning their meal.

\*b. User navigates to another page during planning:

1. End process.

2. No new information saved.

3.a. User selects wrong date:

1. User simply clicks on their preferred date.

2. Re-Enter meal information.

8.a. User wants to delete a meal:

1. User selects a date.

2. User chooses meal from the entered meals on that day.

3. User selects to delete on the meal they would like removed.

4. Meal is removed from that days meals.

**Use Case UC-Plan-03:** Plan Sleep

**Scope:** FitLife application

**Level:** User Goal

**Primary Actor:** Registered User

**Stakeholders and Interests:**

- Registered User: Wants a simple interaction with the system. Wants to future sleep.

Wants to prefill the time, date, and duration for easier completion later.

**Precondition:** User is registered and has successfully logged in.

**Postcondition:** Future sleep is logged and waiting for user to complete

**Main Success Scenario:**

1. Registered User arrives at home screen

2. User selects "Plan" from the side menu

3. System displays the options for planning

4. User selects "Sleep" from the planning menu

5. System prompts user from time, duration, and date

6. User enters require information and clicks the "Submit" button

7. System takes User to the Plan Sleep page

Repeats steps 5-7 until all desired sleep has been planned

8. User logs out and exits program

**Extensions**

\*a. at anytime User navigates to a different page

1. End Process

\*b. at anytime the system fails

1. User restarts the System, logs in, and starts Scenario over again

7a. User wants to delete planned sleep

1. User selects the "Delete Entry" option

2. The system displays the log of planned sleeps

3. The user selects the sleep(s) to be deleted

4. The system asks the user if he/she wants to delete the sleep(s) for good

5. The user replies yes and is taken back to the Sleep Plan page

**Use Case UC-REPORT-01:** Add Workout

**Scope:** FitLife application

**Primary Actor:** Registered User

**Stakeholders and Interests:**

- Registered User: Wants to have a simple and fast interaction with the system.

Wants to add the correct information.

Wants to be able to add a workout quickly.

**Precondition:** User is registered and has successfully logged in.

**Postcondition:** Workout information is saved. Information is updated. Database is updated.

**Main Success Scenario:**

* 1. Registered user arrives at home screen
  2. User selects "Add Workout" from the side menu
  3. System displays a selection of different kinds of workout, and prompts for user's choice
  4. User selects a workout
  5. User clicks "Save" button
  6. User repeats steps 2-5 until completed
  7. Systems captures the information
  8. Information is updated and stored in database
  9. User is able to plan and review workout before logout
  10. User logs out

**Extensions:**

\*a. At any time, Systems fails:

1. User restarts the system

2. User logs in

3. User repeats the process of adding a workout

2-5.a User entered undesired workout

1. User clicks "Delete Workout" button

2. System displays a confirmation message to reassure the action

3. User clicks "Yes" to delete the selected workout, or "Cancel" to cancel the action

4. System stores information

5. Information is updated and stored into the database

2-5.b User decided to change workout

1. User clicks "Delete Workout" button

2. System displays a confirmation message to reassure the action

3. User clicks "Yes" to delete the selected workout, or "Cancel" to cancel the action

4. System stores information

**Use Case UC-REPORT-02:** Add Sleep

**Scope:** FitLife application

**Level:** User Goal

**Primary Actor:** Registered User

**Stakeholders and Interests:**

- Registered User: Wants a simple interaction with the system. Wants to accurately record sleep quality and time.

Wants to enter information quickly.

**Precondition:** User is registered and has successfully logged in.

**Postcondition:** Sleep information is saved. Statistics are updated.

**Main Success Scenario:**

1. Registered User arrives at home screen

2. User selects "Add Sleep" from the side menu

3. System prompts user for required sleep information

4. User enters quality and duration of sleep

User repeats steps 2-4 until completed

5. System captures the information

6. Database stores the information

7. Statistic Library generates and updates statistic

8. User reviews statistics and logs out when complete

**Extensions**

\*a. at anytime User navigates to a different page

1. End Process

\*b. at anytime the system fails

1. User restarts the System, logs in, and starts Scenario over again

8a. User wants to remove entry

1. User selects the "Delete Entry" option

2. The system displays the log of sleeps

3. The user selects the sleep(s) to be deleted

4. The system asks the user if he/she wants to delete the sleep(s) for good

5. The user replies yes and is taken back to the Sleep page

**Use Case: UC-REPORT-03:** Add Meal.

**Scope:** FitLife Application.

**Level:** User Goal

**Primary Actor:** Registered User

**Stakeholders and Interests:**

-Registered User: Wants a simple interaction with the system.

Wants to accurately be able to enter meal information.

Wants to enter information quickly.

**Precondition:** User is registered and successfully logged in.

**Postcondition:** Meal information is saved.

**Main Success Scenario:**

1. Registered user arrives at home screen.

2. User selects Nutrition from home screen.

3. System displays Nutrition screen.

4. User selects date.

5. System displays form for user to fill out information about meal

with criteria such as protein, carbohydrates, natural fats, and others.

6. User fills out criteria with meal information.

7. User selects to add meal to selected date.

8. System captures information.

9. Information stored in database.

10. User logs out and exits application.

**Extensions:**

\*a. User tries to enter meal without selecting date:

1. System prompts user to choose a date.

2. User selects date.

3. User re-enters meal information.

4. User adds meal to that date.

\*b. At any time the system fails:

1. User restarts the System, logs in, and starts Scenario over again.

\*c. User navigates to a different page before adding meal:

1. All unsaved info is lost.

2. User must navigate back and re-enter information.

8.a. User wants to remove entered information:

1. User selects date.

2. System displays all previously entered meals on that date.

3. User chooses delete on the meal to be deleted.

4. Meal is deleted and removed from display.

5. Meal removed from database.

6. Database updated.

**Use Case UC-REPORT-04:** Add Hydration

**Scope:** FitLife application

**Primary Actor:** Registered User

**Stakeholders and Interests:**

- Registered User: Wants to have a simple and fast interaction with the system.

Wants to add the correct information.

Wants to be able to add a hydration quickly.

**Precondition:** User is registered and has successfully logged in.

**Postcondition:** Hydration information is saved. Information is updated. Database is updated.

**Main Success Scenario:**

1. Registered user arrives at home screen
2. User selects "Add Hydration" from the side menu
3. System displays a selection of different types of beverage, and prompts for user input
4. User select the type of beverage and enter the amount
5. User clicks "Save" button
6. User repeats steps 2-4 until completed
7. Systems captures the information
8. Information is updated and stored in database
9. User reviews information and logout

**Extensions:**

\*a. At any time, Systems fails:

1. User restarts the system

2. User logs in

3. User repeats the process of adding a hydration

\*b. User entered incorrected information

1. User clicks "Edit Hydration" button
2. User enters desired information
3. User saves

\*c. User decided to change hydration

1. User clicks "Delete hydration" button
2. System displays a confirmation message to reassure the action
3. User clicks "Yes" to delete the selected hydration, or "Cancel" to cancel the action
4. System stores information
5. Information is updated and stored into the database

**Use Case UC-REVIEW-01:** Review Workout

**Scope:** FitLife application

**Level:** User Goal

**Primary Actor:** Registered User

**Stakeholders and Interests:**

- Registered User: Wants a simple interaction with the system.

Wants to view added workouts.

Wants to be able to review a workout.

Wants to be able to filter by week, month, and year.

Wants to be able to view the total workouts within a week, a month, or a year.

**Precondition:** User is registered and has successfully logged in.

User has added at least one workout

**Postcondition:** workout information and trends are displayed to User

**Main Success Scenario:**

1. Registered User arrives at home screen

2. User selects "Review Workout" from the side menu

3. System displays the trend options

4. User selects "Workout" from the trends menu

5. System displays workout trends in graphical format

6. User reviews workout

**Extensions:**

\*a. at anytime User navigates to a different page

1. End Process

\*b. at anytime the system fails

1. User restarts the System

2. user logs in, and starts Scenario over again

6.a Weekly Trends

1. User selects "Weekly" option

2. System displays the modified graph of trends for the current week

3. User reviews the weekly trends

6.b Monthly Trends

1. User selects "Monthly" option

2. System displays the modified graph of trends for the current month

3. User reviews the "Monthly" trends

6.c Yearly Trends

1. User selects "Yearly" option

2. System displays the modified graph of trends for the current year

3. User reviews the "Yearly" trends

**Use Case UC-REVIEW-02:** Review Sleep

**Scope:** FitLife application

**Level:** User Goal

**Primary Actor:** Registered User

**Stakeholders and Interests:**

- Registered User:

Wants a simple interaction with the system. Wants to view previous sleep logs. Wants to be able to review a sleep log. Wants to view aggregate data in graph format. Wants to be able to filter by week, month, and year.

**Precondition:** User is registered and has successfully logged in.

**Postcondition:** Sleep logs and trends are displayed to User

**Main Success Scenario:**

1. Registered User arrives at home screen

2. User selects "Review Trends" from the side menu

3. System displays the trend options

4. User selects "Sleep" from the trends menu

5. System displays Sleep trends in graphical format

6. User reviews trends

**Extensions**

\*a. at anytime User navigates to a different page

1. End Process

\*b. at anytime the system fails

1. User restarts the System, logs in, and starts Scenario over again

2-6a. User selects a specific sleep log item

1. System displays the stats for the specified log item

2. User reviews sleep information

6.a Weekly Trends

1. User selects "Weekly" option

2. System displays the modified graph of trends for the current week

3. User reviews the weekly trends

6.b Monthly Trends

1. User selects "Monthly" option

2. System displays the modified graph of trends for the current month

3. User reviews the "Monthly" trends

6.c Yearly Trends

1. User selects "Yearly" option

2. System displays the modified graph of trends for the current year

3. User reviews the "Yearly" trends

**Use Case UC-REVIEW-03:** Review Diet

**Scope:** FitLife Application

**Level:** User Goal

**Primary Actor:** Registered User

**Stakeholders and Interests:**

- Registered User: Wants a simple interaction with the system.

Wants to view diet history.

Wants to be able to review diet.

Wants to be able to filter by day, month, and year.

**Precondition:** User is registered and has successfully logged in.

**Postcondition**: User is able to view diet statistics and trends.

**Main Success Scenario:**

1. Registered user arrives at home screen.

2. User selects "Review Nutrition" from home screen.

3. System displays calendar.

4. User selects a specific day, week, month, or year from the calendar.

5. Diet trends and information are displayed based on selected timeframe.

6. User reviews diet.

7. User repeats steps 4-6 until finished.

Extensions:

\*a. At any time, system fails:

1. User restarts system.

2. User logs in.

3. User begins scenario again.

\*b. At any time user navigates to new page:

1. End process.

4.a. User selects a specific day:

1. Diet trends and information adjusted for that selected day.

2. User reviews trends and information for that day.

4.b. User selects a specific week:

1. Diet trends and information adjusted for that selected week.

2. User reviews trends and information for that week.

4.c. User selects a specific month:

1. Diet trends and information adjusted for that selected month.

2. User reviews trends and information for that month.

4.d. User selects a specific year:

1. Diet trends and information adjusted for that selected year.

2. User reviews trends and information for that year.

**Use Case UC-REVIEW-04**: Review Hydration

**Scope:** FitLife application

**Primary Actor:** Registered User

**Stakeholders and Interests:**

- Registered User: Wants a simple interaction with the system.

Wants to be able to view hydration history.

Wants to be able to review hydration history.

Wants to be able to filter by day, week, month, and year.

**Precondition:** User is registered and has successfully logged in.

**Postcondition:** Hydration information and trends are displayed to user.

**Main Success Scenario:**

1. Registered user arrives at home screen.

2. User selects "Review Nutrition" from home screen.

3. System displays calendar.

4. User selects a specific day, week, month, or year from the calendar.

5. Hydration and diet trends and information are displayed based on selected timeframe.

6. User reviews hydration and diet.

7. User repeats steps 4-6 until finished.

**Extensions:**

\*a. At any time, system fails:

1. User restarts system.

2. User logs in.

3. User begins scenario again.

\*b. At any time user navigates to new page:

1. End process.

4.a. User selects a specific day:

1. Hydration and diet trends and information adjusted for that selected day.

2. User reviews trends and information for that day.

4.b. User selects a specific week:

1. Hydration and diet trends and information adjusted for that selected week.

2. User reviews trends and information for that week.

4.c. User selects a specific month:

1. Hydration and diet trends and information adjusted for that selected month.

2. User reviews trends and information for that month.

4.d. User selects a specific year:

1. Hydration and diet trends and information adjusted for that selected year.

1. User reviews trends and information for that year.

**Use Case UC-Review-05:** Review BMI Trends

**Scope:** FitLife Application

**Level:** user goal

**Primary Actor:** Registered User

**Stakeholders and Interests:**

- Registered User:

Wants to have a simple and fast interaction with the system, able to keep track on

the BMI trends, calculate and update the current BMI.

- Database:

Wants to receive correct (reasonable) information of weight and height from user.

**Preconditions:**

User has logged into their account and user information has identified.

**Success Guarantee (or Postconditions):**

User can update their current BMI and get the BMI trend.

**Main Success Scenario (or Basic Flow):**

1. User select BMI Trends from the user main page.
2. User entered their height and weight into the System.
3. System calculate the user’s current BMI and update the BMI trends.
4. System shown the BMI trends to User.

**Extensions (or Alternative Flows):**

\*a. At anytime User navigates to a different page

1. End Process

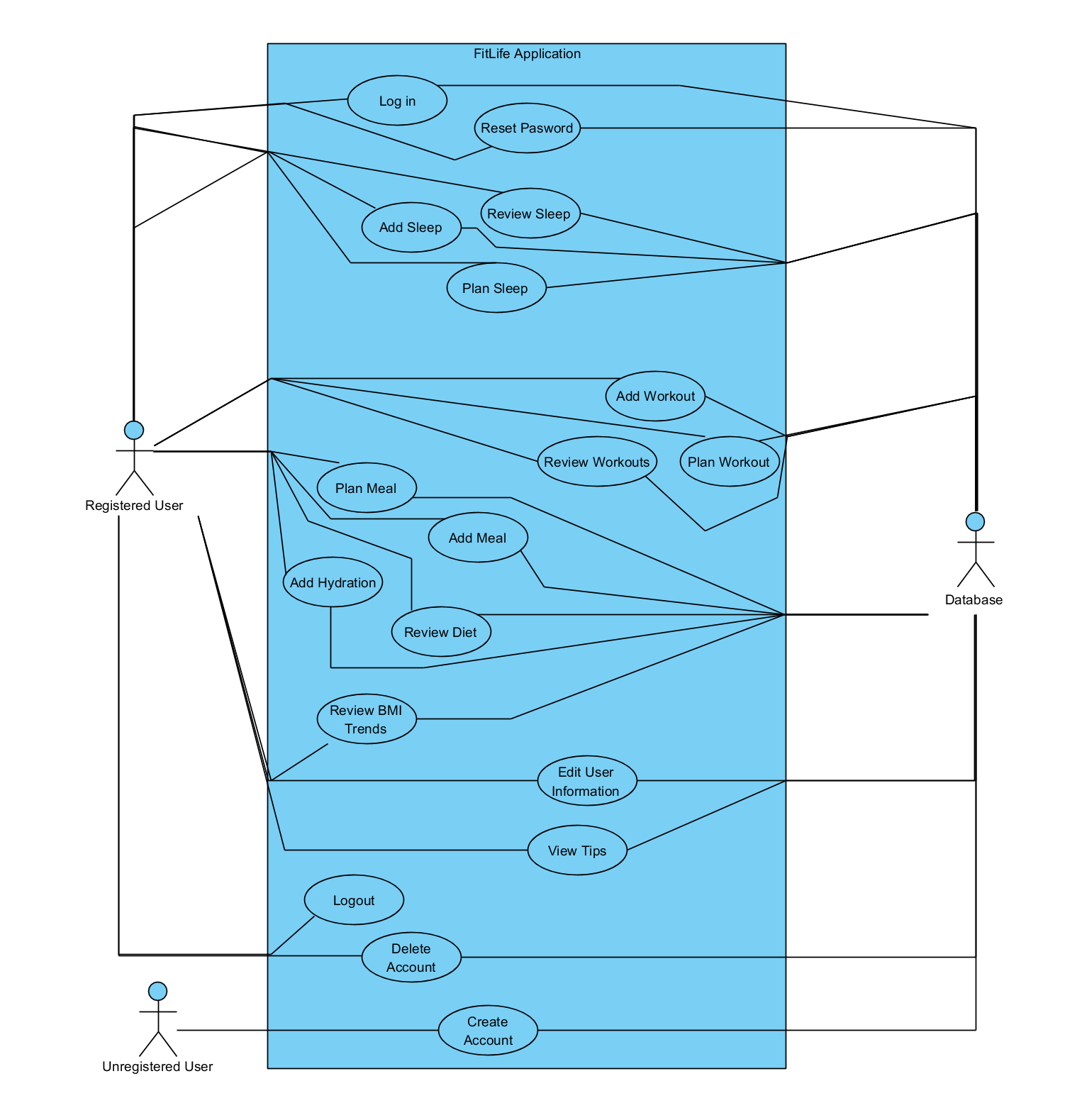
\*b. Anytime user closes the application

1. End Process

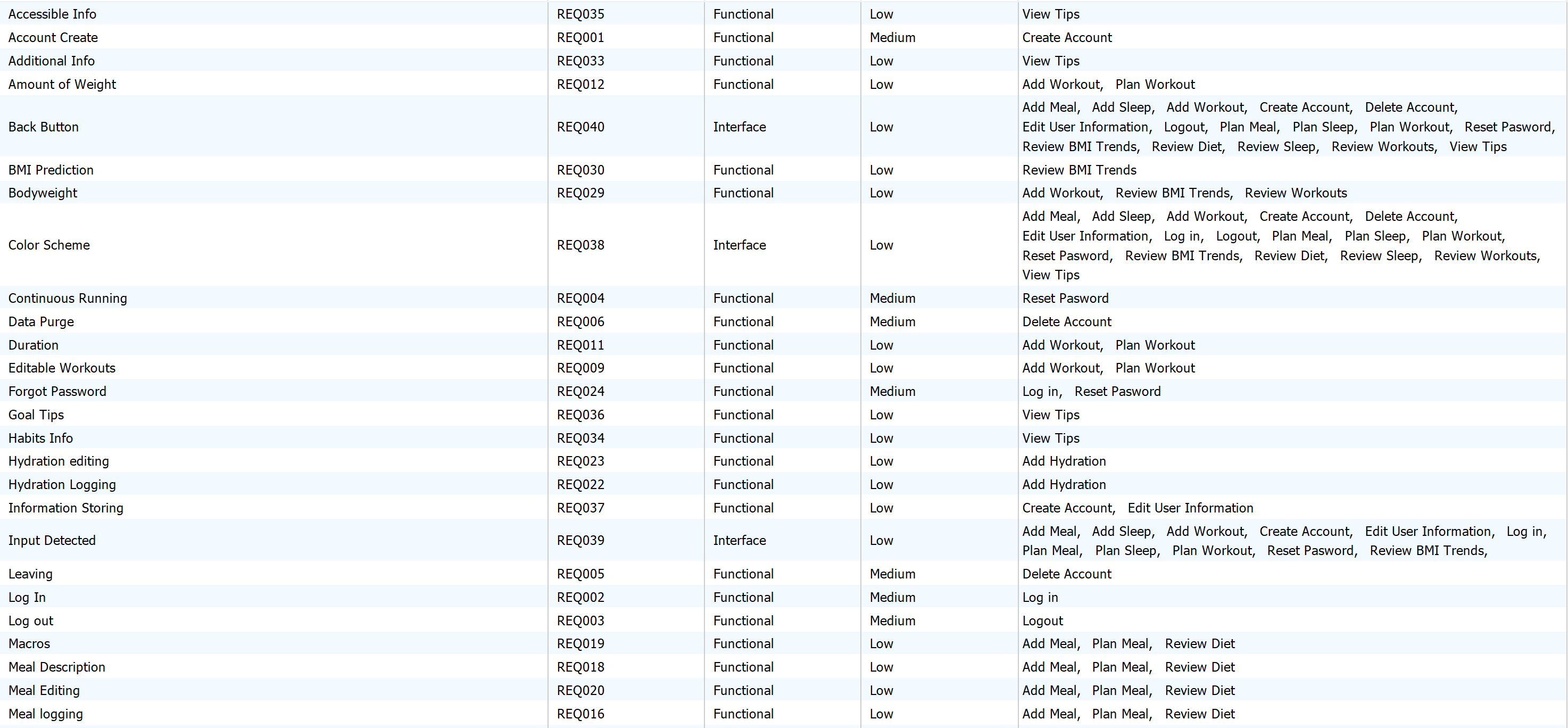
2a. User entered unreasonable input into System.

1.System signals error to the User, records the error, and enters a clean state.

2. System ask user to enter their information again.

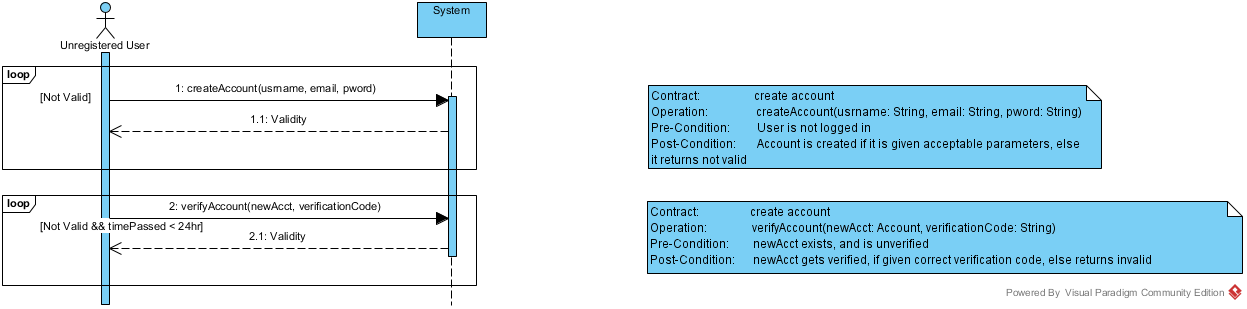
Use Case Diagram

Traceability Matrix

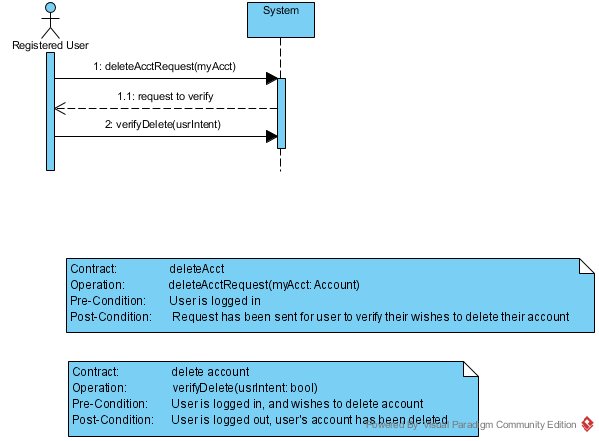


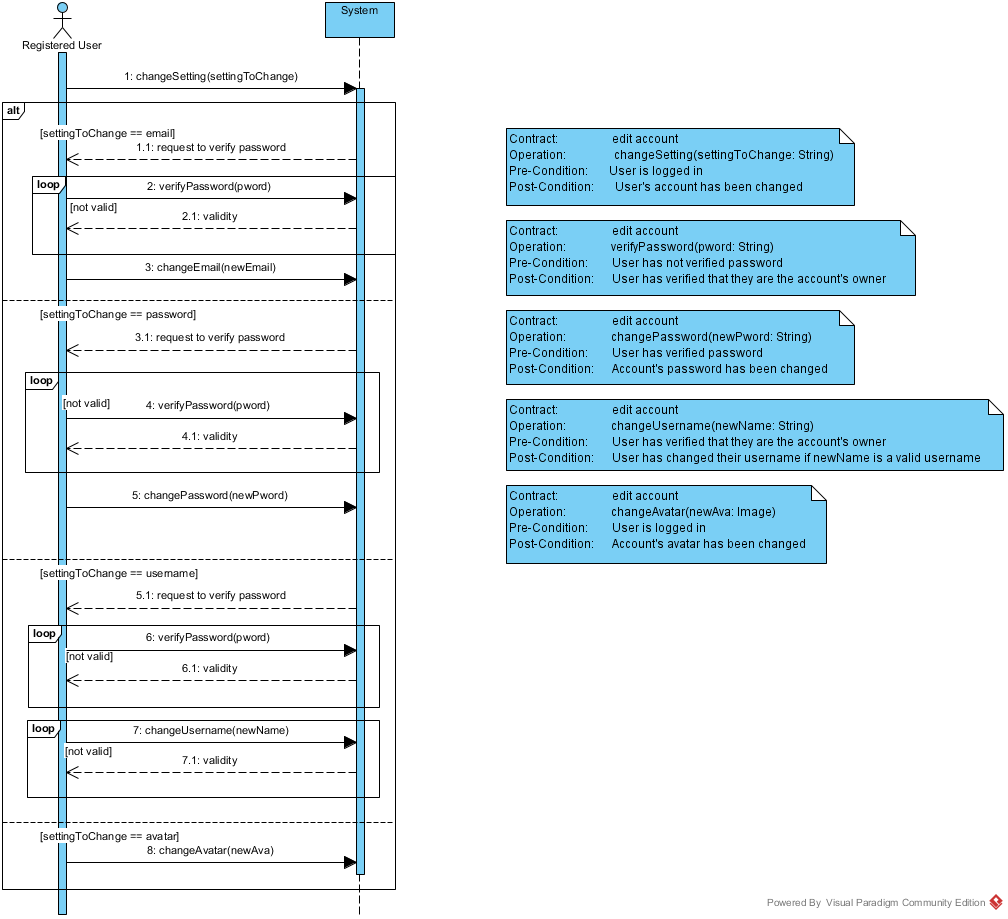
System Sequence Diagrams & Operation Contracts

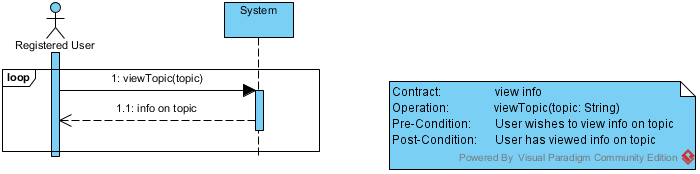
**Create Account**



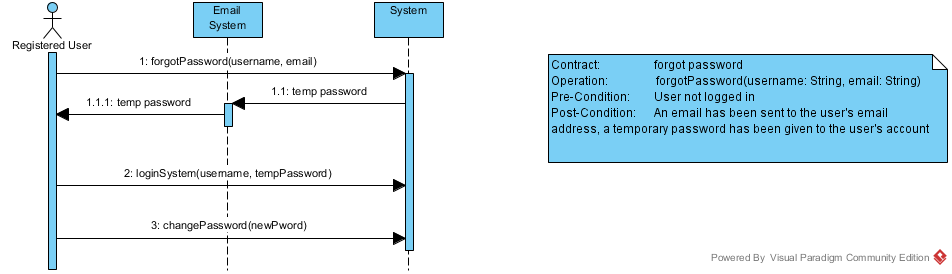
Delete Account



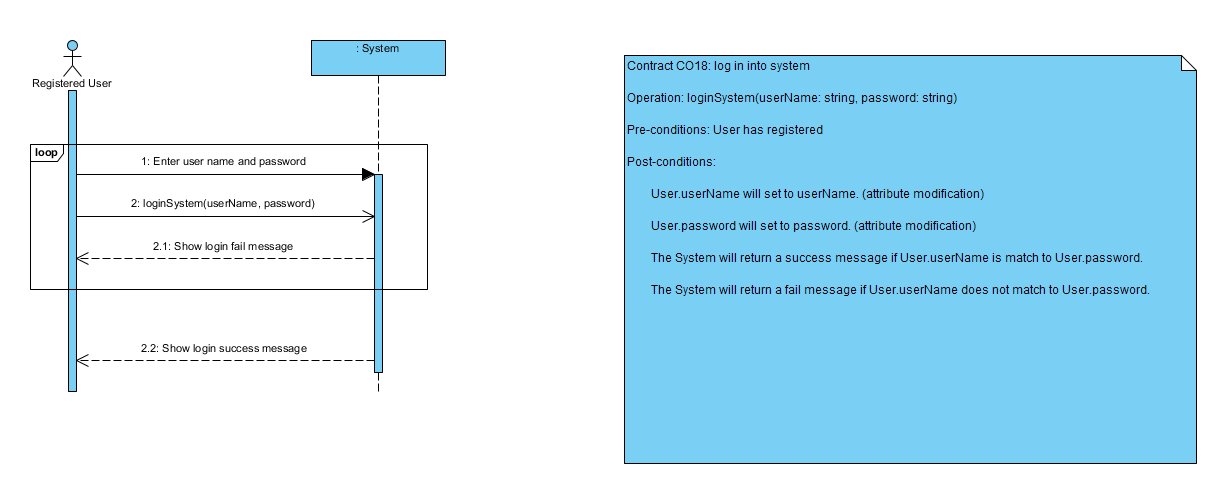
**Edit Account**

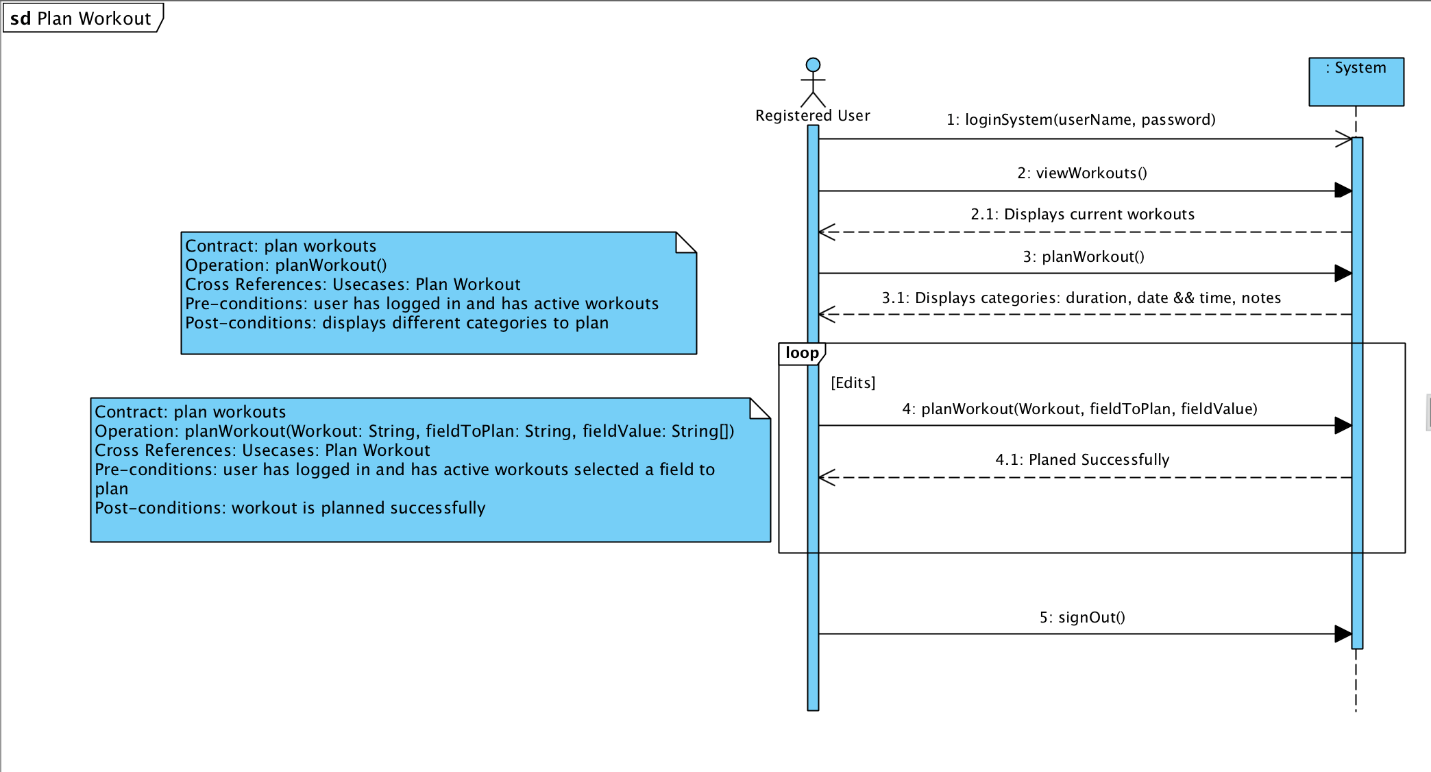
**View Info**

**Reset Password**

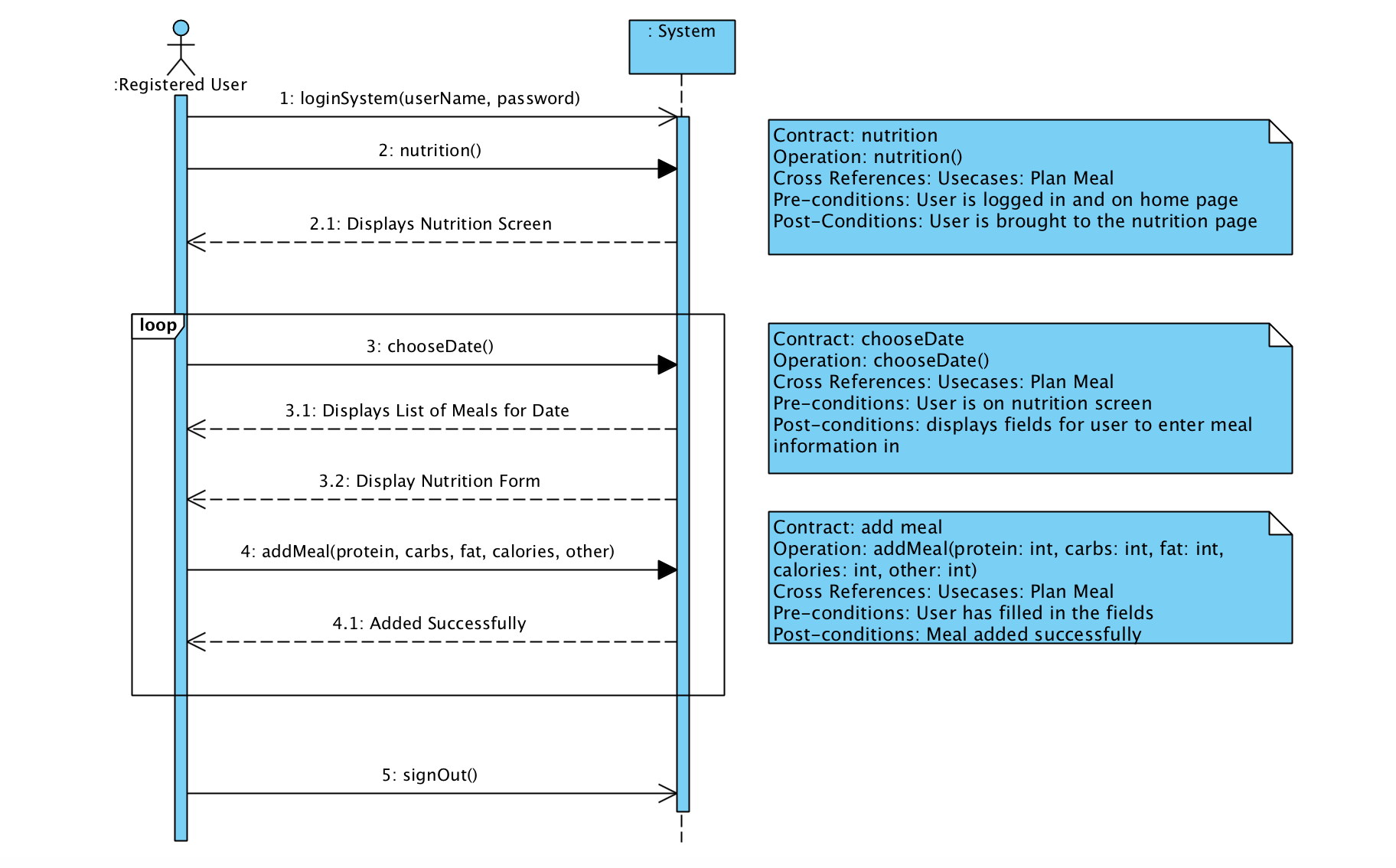


**Log In**

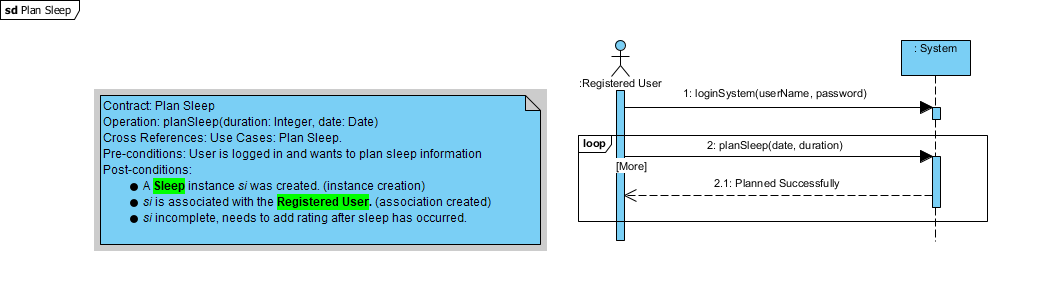


**Plan Workout**

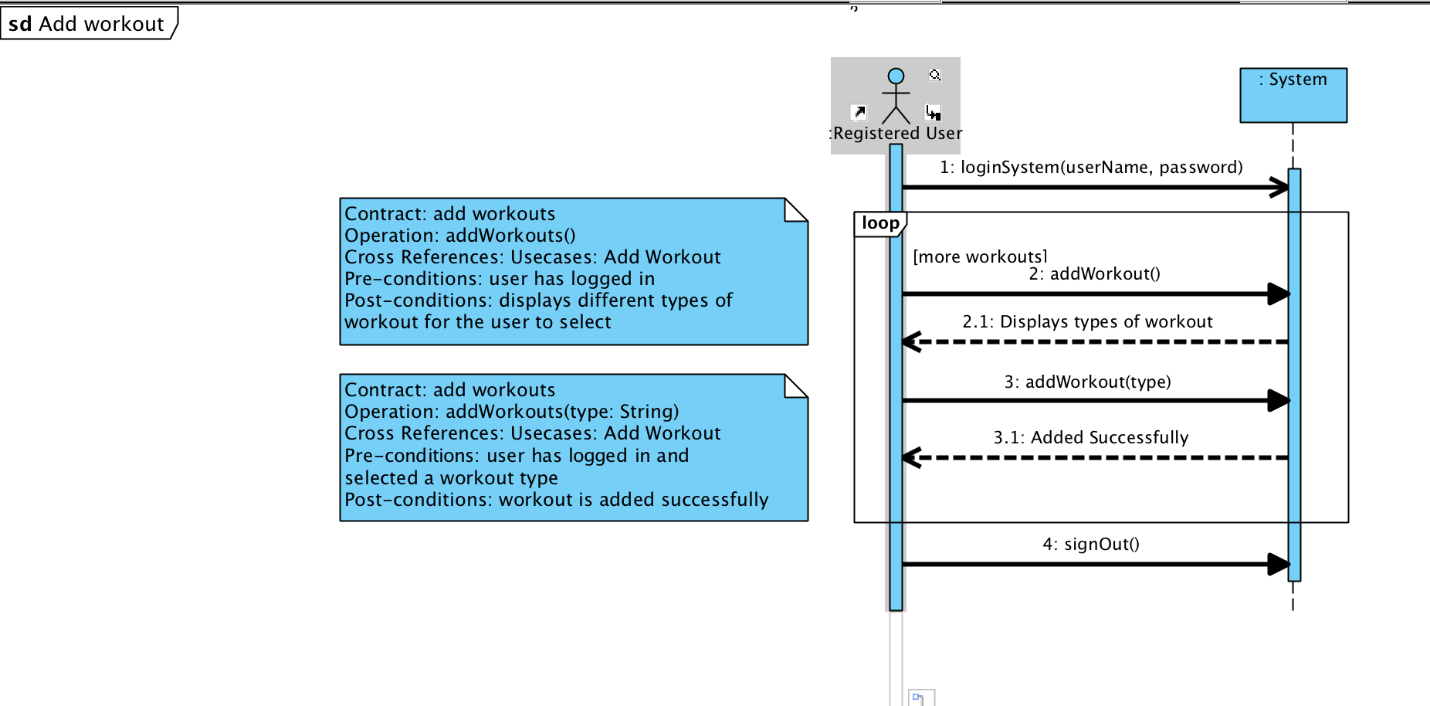
**Plan Meal**



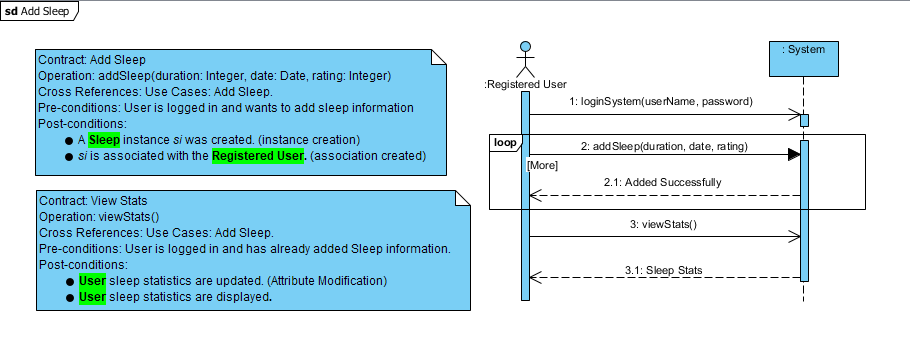
**Plan Sleep**

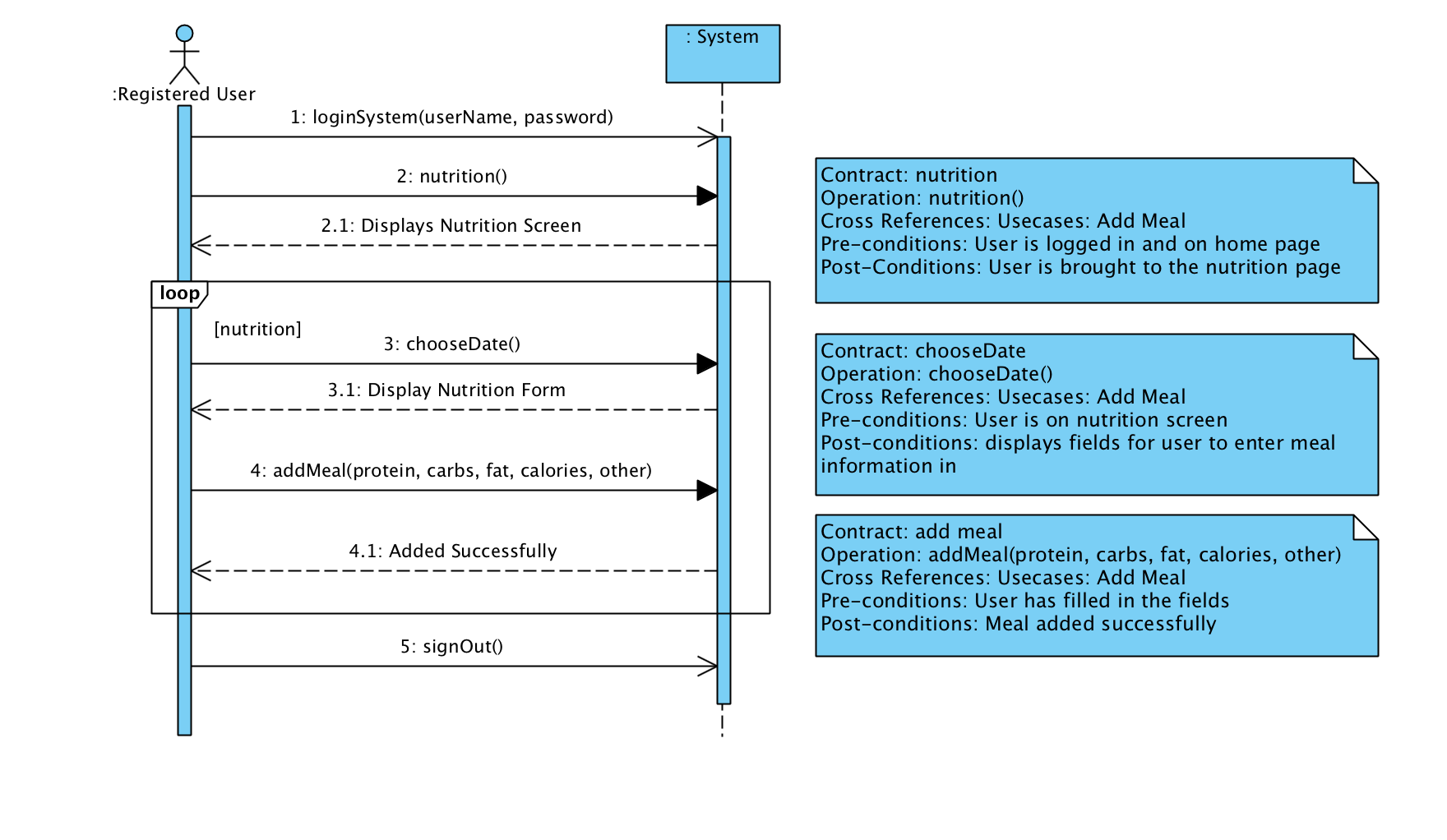


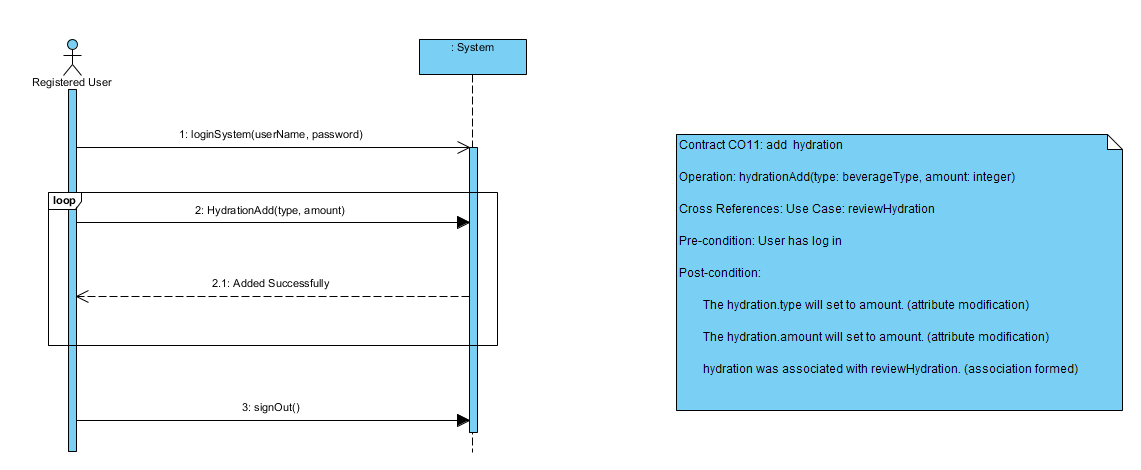
**Add Workout**



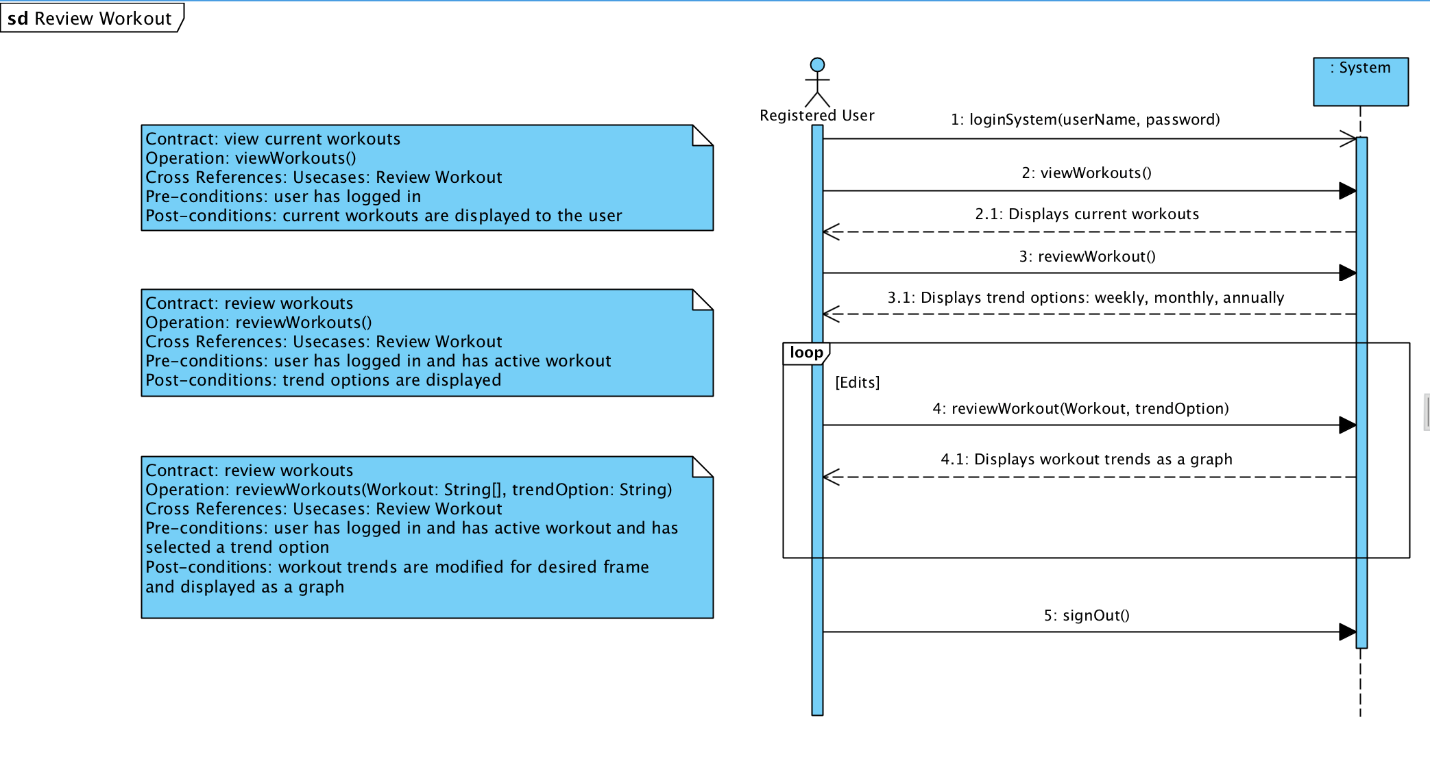
**Add Sleep**



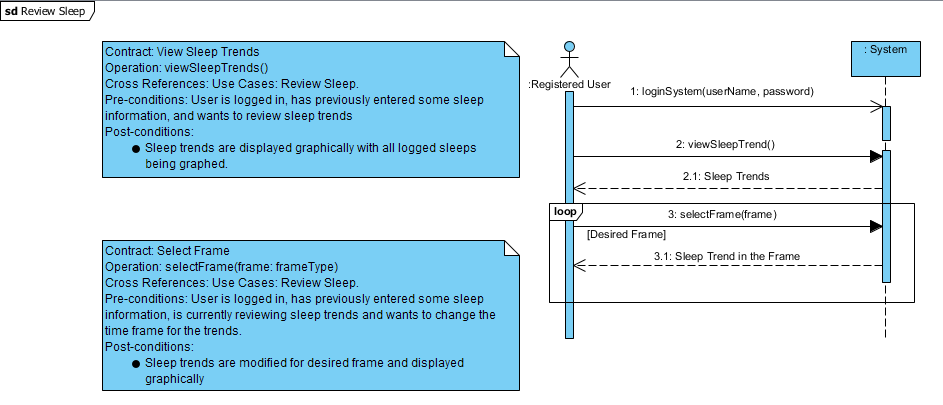
**Add Meal**

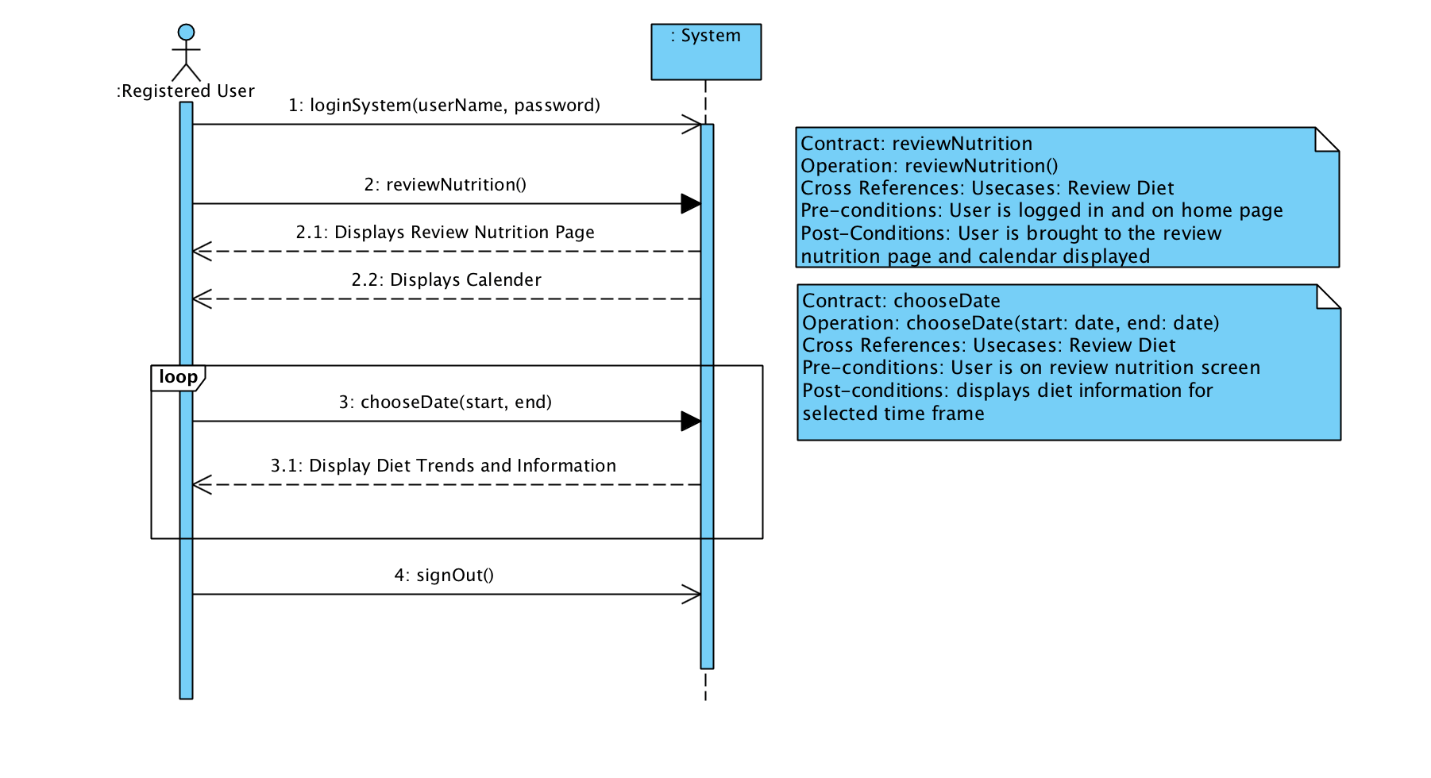
**Add Hydration**

**Review Workout**

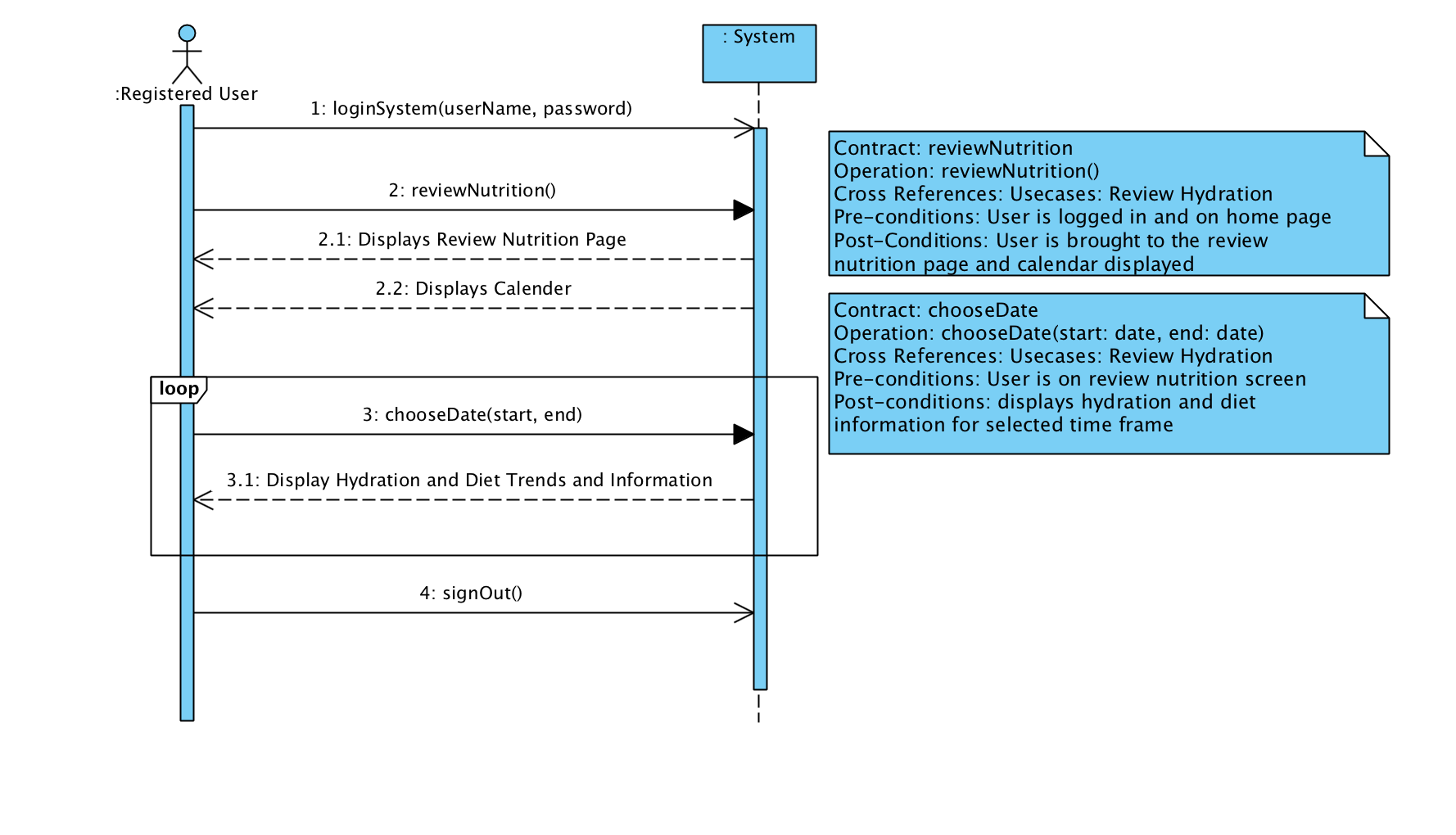


**Review Sleep**

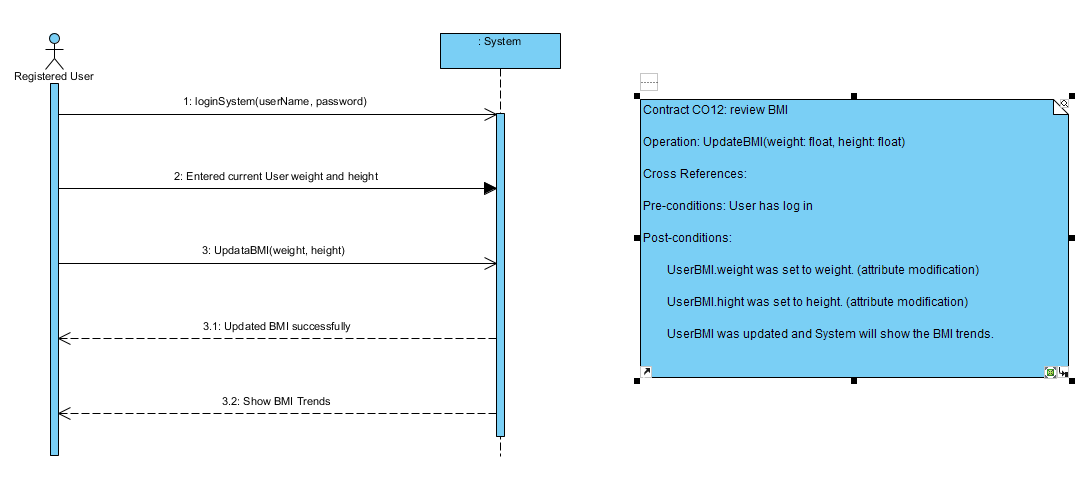


**Review Diet**

**Review Hydration**



**Review BMI**



Wireframes

**Create Account**



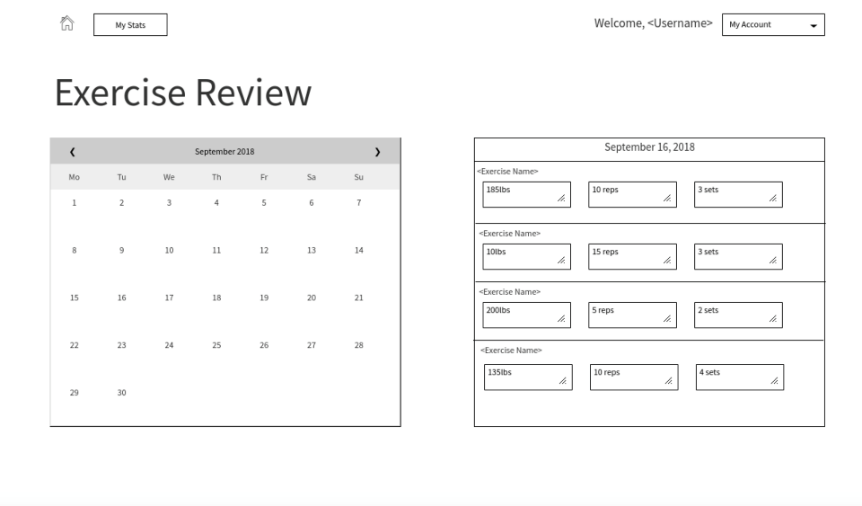
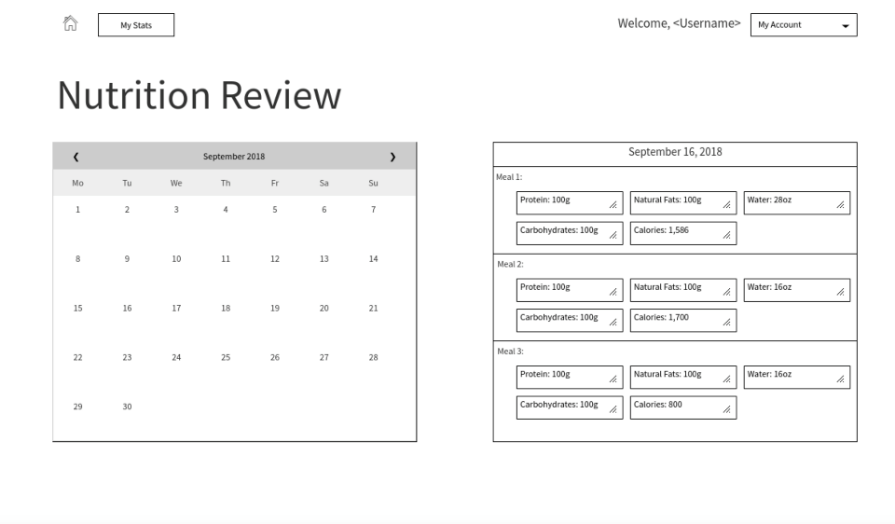
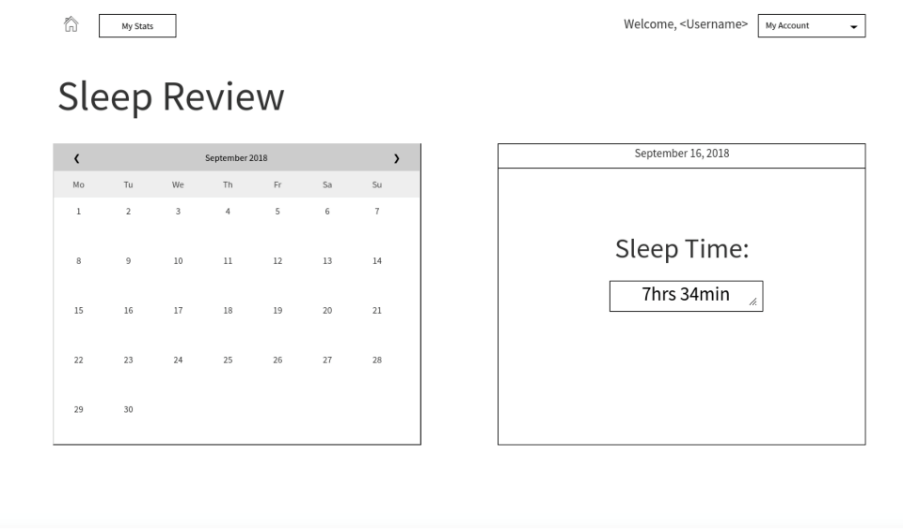
**Log In**



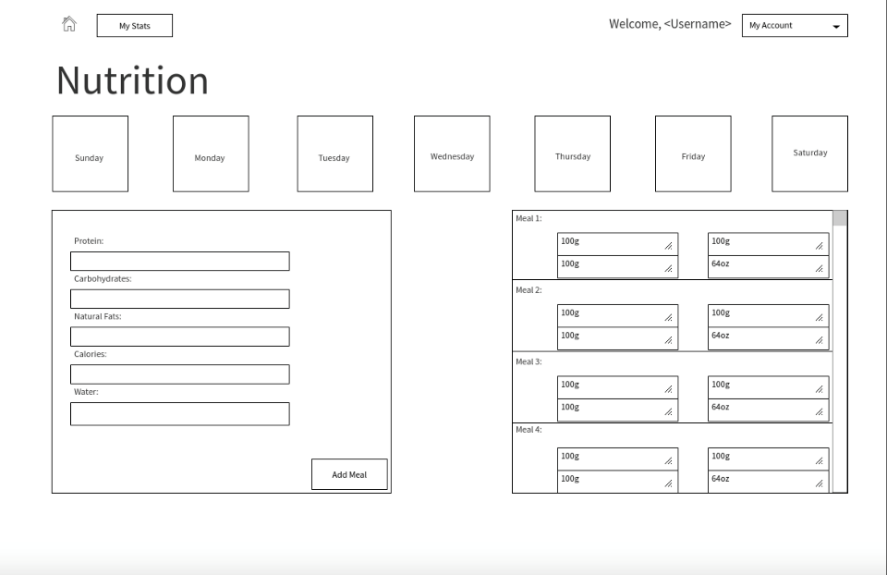
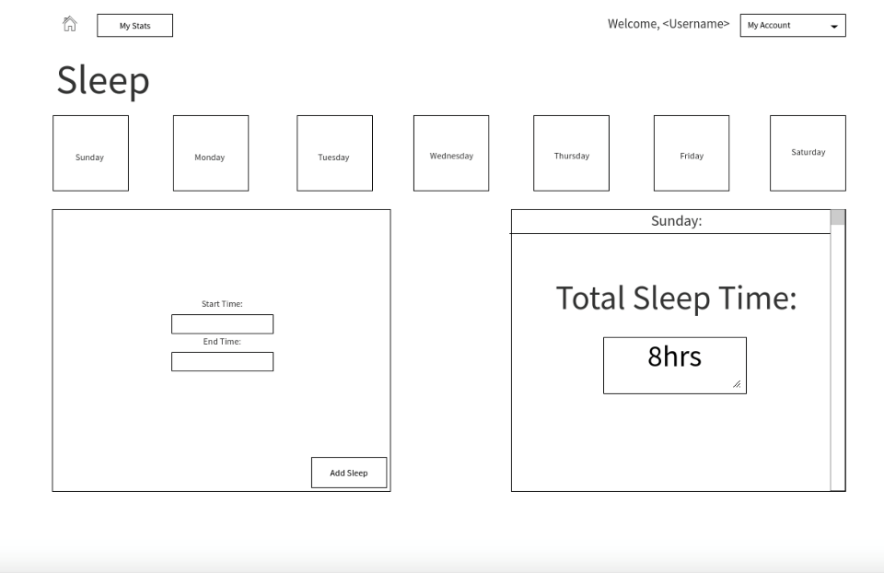
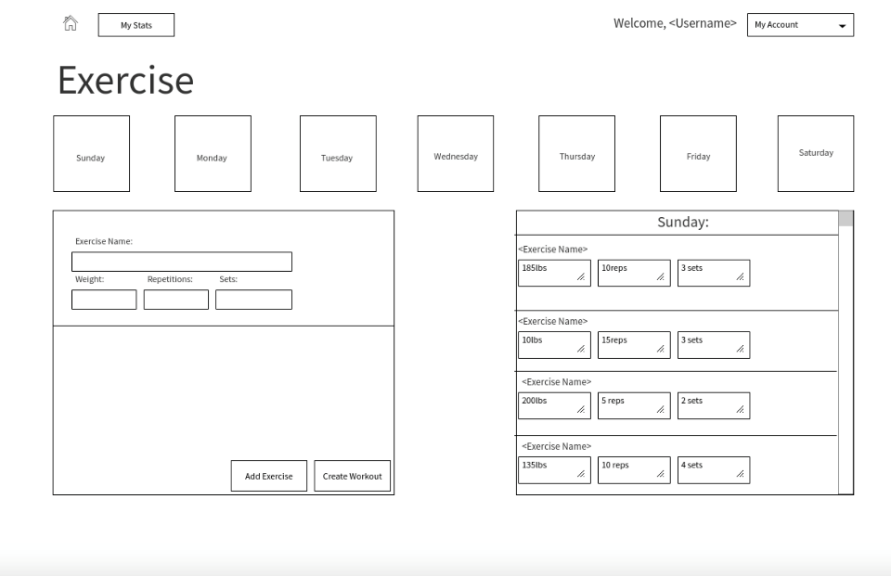
**Account Information**

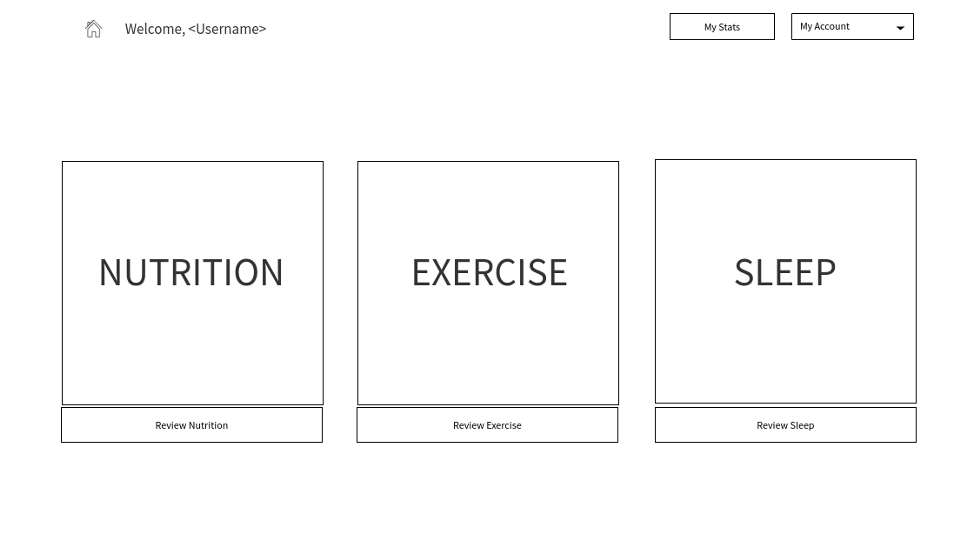


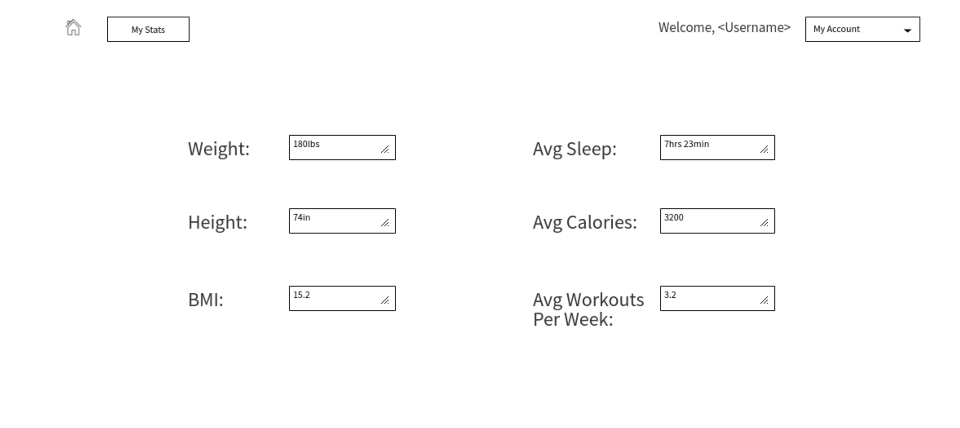
**Review**

**Logging/Planning**



**Home**

**Stats**

**Domain Model**

