SRS for MMC Fitness

By: May Tzadoky, Chan Shitrit and May Ovadia

Table of Contents

1. Introduction

1.1 Purpose

The purpose of this document is to specify the software requirement of the “MMC Fitness” project and to overview the main components and functionality that shall be available in the first version of the website.

1.2 Scope

The MMC Fitness website will provide a service to people who want to do sports and get nutrition menus from professionals.

The professionals will upload menus/training to the website and the subscribers will be able to use the content and thereby reach the nutrition and toning goals they have set for themselves. The website will consider physical disabilities such as tailoring specific training to subscribers according to ability and preference. It will also consider diet menus that meet a variety of restrictions such as (celiac, vegetarians, vegans, and all the preferences available in the system).

1.3 Glossary

Term	**Definition**
MMC	Abbreviation of May, May, Chen (the owners of this project)
MSC	Message Sequence Chart
DB	Database

2. Overall Description

2.1 Features

2.1.1 System Subscription

Enters his data into the system when he logs in for the first time. The data entry will contain detailed settings about the user such as:

- Weight
- Height
- Age
- Gender
- Dietary preferences
- Physical abilities

Using these details, the system will build the subscriber the most suitable menu and training board for him. That means building the menu and training will be individual to the user, so he will not get something generic but something he will be able to stick to and persevere with.

System Functionality:

- Change the details at any time - the subscriber will be able to update his personal details using a dedicated screen.
- Go to a custom menu - The subscriber will be able to go to the nutrition menu screen and choose the menu for himself from a variety of options that will be offered especially for him. That is, according to the number of calories in the menu/protein/preferences he defined in his personal details.
- Go to a customized training program - The subscriber will be able to go to a screen where he will be shown various fitness workouts, recommendations and explanations of each workout, and the option of marking if he met the target and did his workouts that week.
- Chat with a Nutritionist – any time the user will be able to start a chat with one of the nutritionists in the system, and he will receive answers about his nourishment questions.

- Chat with a Trainer – any time the user will be able to start a chat with one of the trainers in the system. He will be able to choose which specialty of sport the question will be related to, and the massage send to the chosen trainer. The user will receive answers about his workout questions.

2.1.2 Trainer type user

Enters his data into the system when he logs in for the first time. The data entry will contain detailed settings about the user such as:

- User id
- First + last name
- Specialty

System Functionality for Trainer user:

- Add Workout – the trainer would be able to add workouts to the workout collection, so the more workouts there will be, the system will be able to match more accurate the workout to the users
- Delete Workout - the trainer would be able to delete from the system workouts that he thinks are irrelevant.
- Edit Workouts - the trainer would be able to edit existing workouts, so it will be clearer for the users (such as add pictures, add exercise for different difficulty levels).
- Answer in chat – the trainer will answer Subscription users in chat, will provide them answer for questions and support them remotely.

2.1.3 Nutritionist type user

Enters his data into the system when he logs in for the first time. The data entry will contain detailed settings about the user such as:

- User id
- First + last name

System Functionality for Nutritionist user:

- Add Menu – the nutritionist would be able to add menu to the menus collection, so the more menus there will be, the system will be able to match more accurate the menu to the users
- Delete Menu - the nutritionist would be able to delete from the system menus that he thinks are irrelevant.
- Edit Menu - the nutritionist would be able to edit existing menus, so it will be clearer for the users (such as add pictures, add recommendation about specific products to use, change quantities, change products).
- Answer in chat – the nutritionist will answer Subscription users in chat, will provide them answer for questions and support them remotely.

2.2 General features

2.2.1 Reports

The user would be able to issue a report that contains his progress. In this report he will see information about the kilogram amount he lost (from his registration to the system). He will see his fat percentage compare to his starting point.

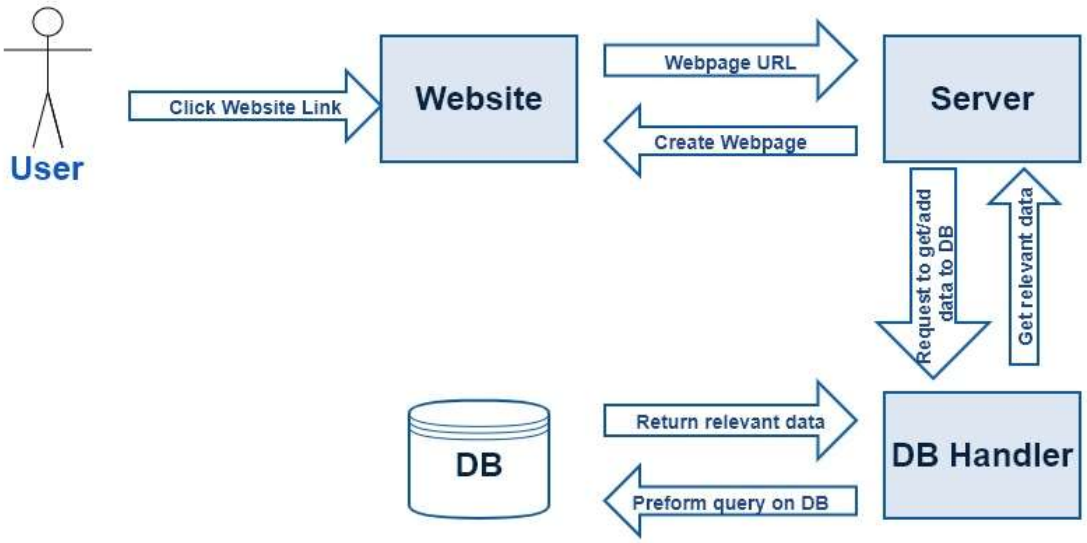
In addition, the data will be show with proper graph that will reflect his progress by week.

2.2.2 Periodic condition assessment

Every three months a personal meeting will be schedule for the Subscription user with trainer and nutritionist.

In this meeting the specialists will fits to user new menu in accordance with his progress in the last 3 months. Additionally, will be fit new training set according to his shape status.

3. System Architecture



4. Software Specific Requirements

4.1 Functional Requirements- MSC

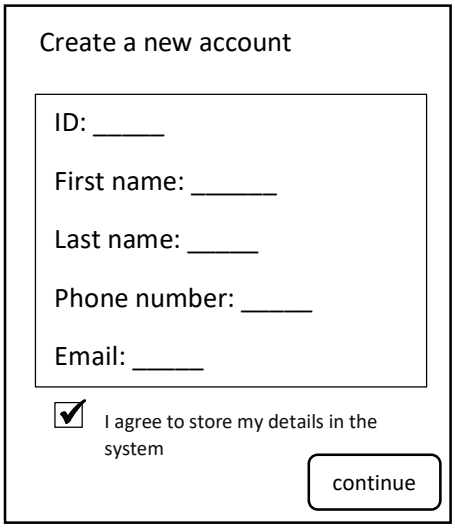
4.1.1 Login to the system

4.1.1.1 Input

Login – "Login" button was pressed in the welcome screen

4.1.1.2 Processing

1. The user presses "Login" button from the welcome screen
2. If this is the 1st time, the browser takes focus and asks the user to accept the permission required for the app to function and to fill in the following details: ID, first name, last name, phone number and email.
3. After logged in, the data will be stored on the DB, in a table called "User". If the operation failed, notify the user and stay in the welcome screen.
   "User" table – this table holds details about all users in the system, such as: ID, first name, last name, phone number, email.
4. Login data is also saved to the local DB, so user will not have to login again the next time he starts the app.



4.1.1.3 Outputs

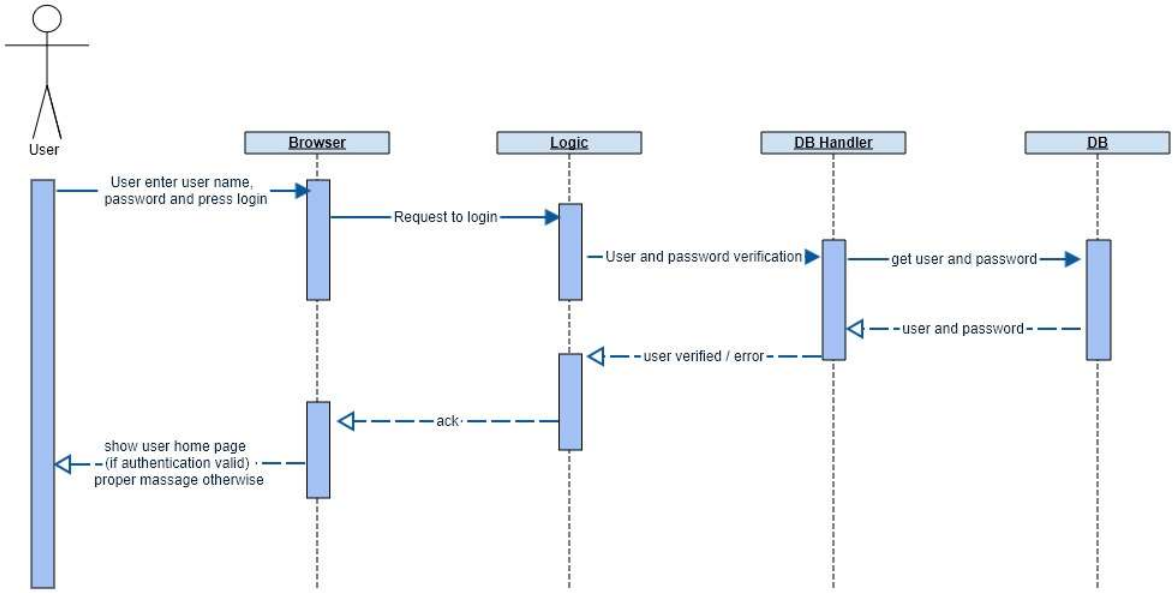
After successfully log-in - show the home page of the app.
If this was the 1st time a user logged in, show a welcome message.

After failed log-in - notify the user and stay in the welcome screen.

4.1.1.4 Exception

- DB access errors (write / read)
- Local DB access errors (write / read)

4.1.1.5 MSC



4.1.2 Add Menu by Nutritionist

4.1.2.1 Input

Add menu – "Add menu" button was pressed in the home page

4.1.2.2 Processing

"Add menu" button

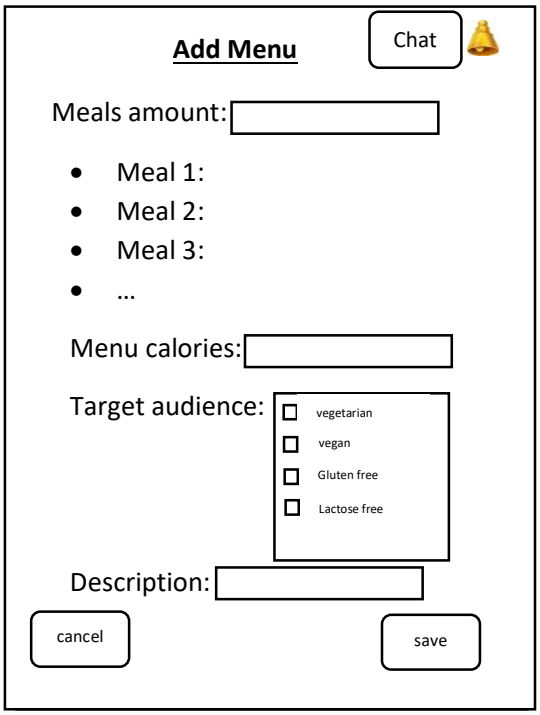
1. The nutritionist presses "Add menu" button from the home page
2. The nutritionist is taken to the "Menu page" and is asked to fill in fields for the new menu. The fields are:
    - How many meals in the menu.
    - According to number of meals will be open text boxes to add meals.
    - How many calories
    - Target audience: who can use this menu (like allergies and food preferences)
    - Description: free text the nutritionist can describe the menu, and add comments.

"Save" button

1. If all the fields are filled, the data will be stored on the DB, in a table called "Menu". If the operation failed, notify the user and stay in the "Add menu" screen.
   If not all the fields are filled, notify the user and stay in the "Add menu" screen.
   "Menu" table – this table holds details about all menus in the system, such as: description, fat percentage, popularity, calories.

"Cancel" button

1. The browser asks the user to accept the cancel of the data he filled out.
2. The nutritionist is taken to the home page.



4.1.2.3 Output

"Save" button

After successfully save- show the home page of the app.
and show a message "Saved successfully"

After failed save - notify the user and stay in the menu page.

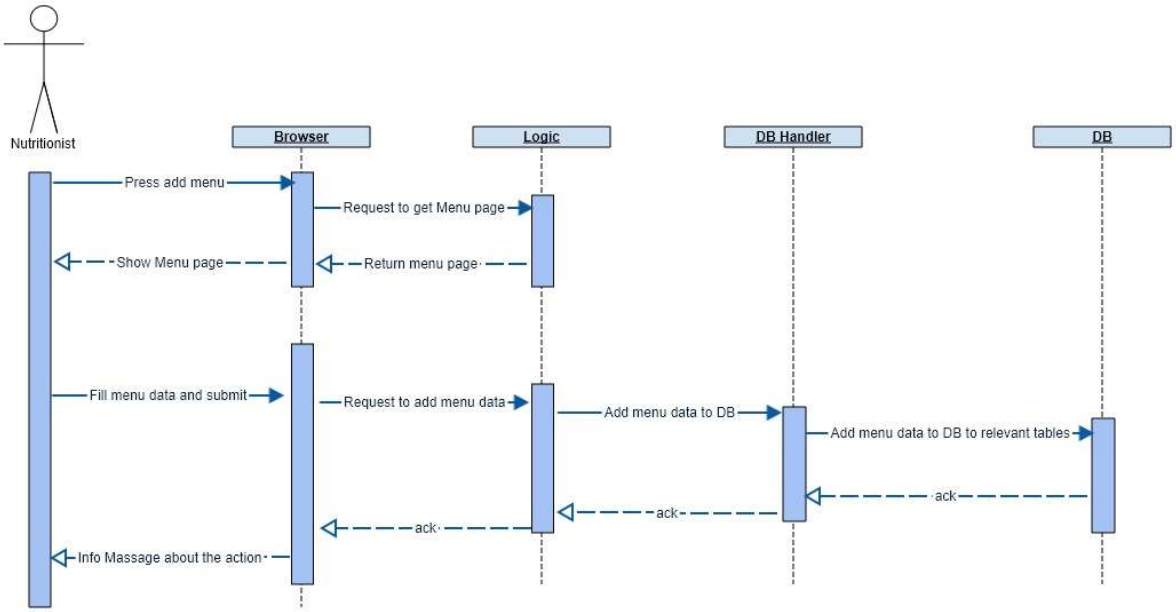
"Cancel" button

After cancel - notify the user and show the home page of the app.

4.1.2.4 Exception

- DB access errors (write / read)
- Local DB access errors (write / read)

4.1.2.5 MSC



4.1.3 Add Workout by Trainer

4.1.3.1 Input

Add Workout – "Add workout " button was pressed in the home page

4.1.3.2 Processing

"Add workout " button

1. The Trainer presses "Add workout" button from the home page
2. The Trainer is taken to the "Workouts page" and is asked to fill in fields for the new menu. The fields are:

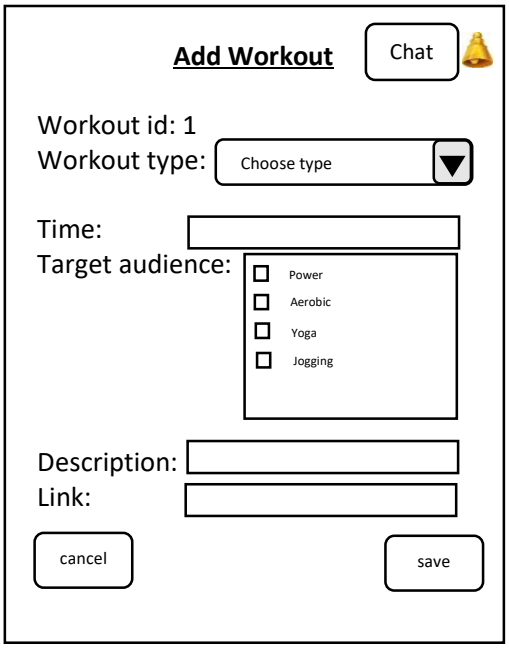
- Workout id: will be giver automatically by the system.
- Workout type: choosing the workout type from closed list.
- Description: free text the trainer will be able to describe the workout details.
- Target audience: the users' parameters that can do the workout.
- Time: how long this workout is.
- Link: option to add link to a video that will present the workout.

"Save" button

If all the fields are filled, the data will be stored on the DB, in a table called "Workout". If the operation failed, notify the Trainer and stay in the "Add workout" screen.
If not all the fields are filled, notify the Trainer and stay in the "Add workout" screen.
"Workout" table – this table holds details about all workouts in the system, such as: description, type, focus, popularity, timestamp, link.

"Cancel" button

The browser asks the Trainer to accept the cancel of the data he filled out.
The trainer is taken to the home page.



4.1.3.3 Output

"Save" button

After successfully save- show the home page of the app.
and show a message "Saved successfully"

After failed save - notify the user and stay in the workout page.

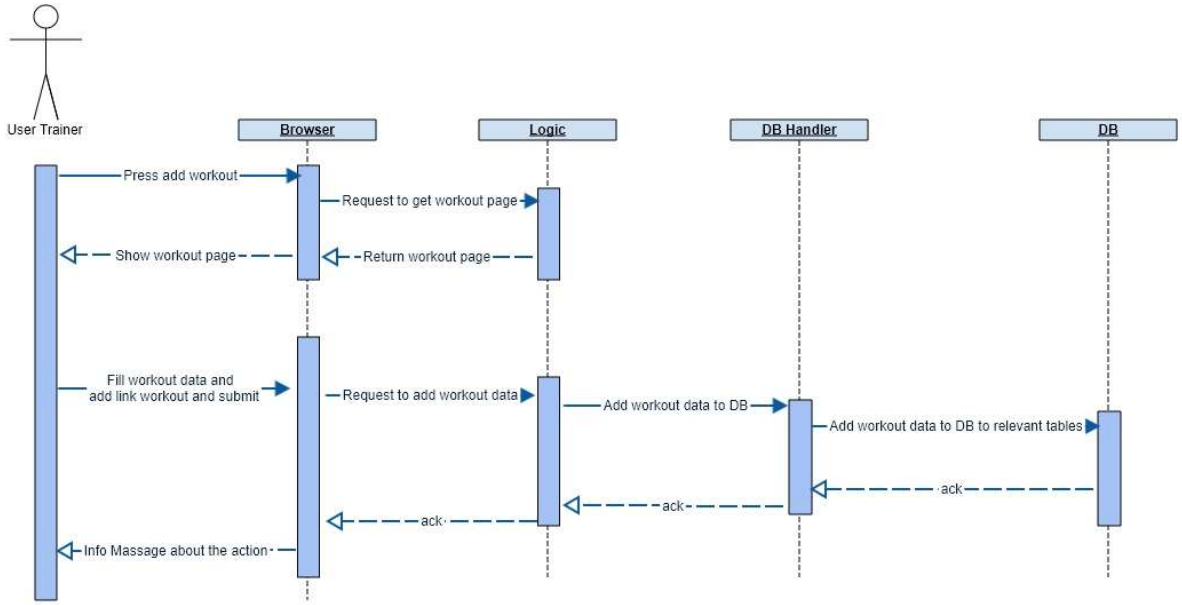
"Cancel" button

After cancel - notify the trainer and show the home page of the app.

4.1.3.4 Exception

- DB access errors (write / read)
- Local DB access errors (write / read)

4.1.3.5 MSC



4.1.4 Show Workout Planning (for Subscription user)

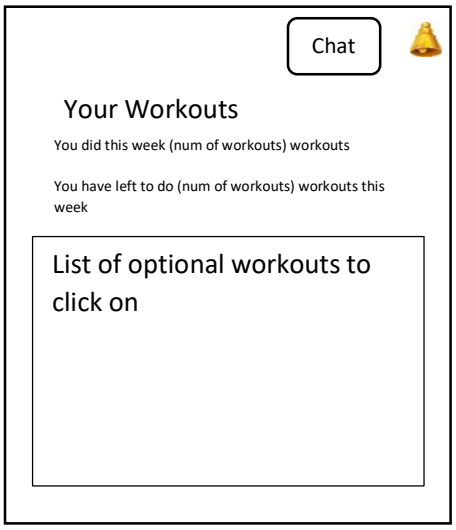
4.1.4.1 Input

Show workout planning – “Show workout planning” button was pressed in the home page

4.1.4.2 Processing

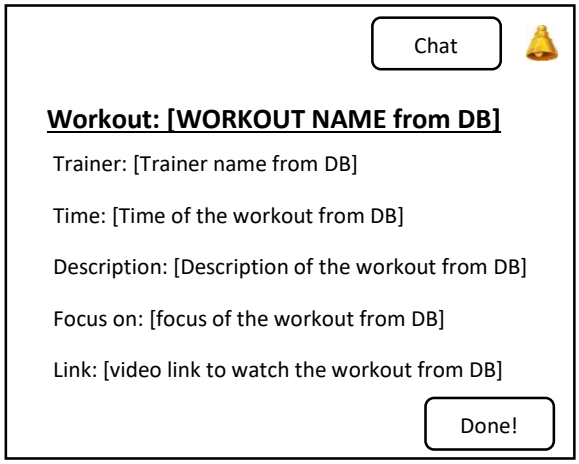
<u>"Show workout planning" button</u>

- The subscriber user presses "Show workout planning" button from the home page
- New page is opened to user and it show the user this data:



- Each link of an optional workout contains a description, duration of the workout, type of workout, instructional video, instructional photos, for user convenience

- After user choose specific workout, the Logic component ask from “DB Handler” to gets the data of the workout from DB, from table “workout”, than the data: name, createdBy, timestamp, description, focus and link return to “Logic” that create new page that show the chosen workout information:



4.1.4.3 Output

"Workout completed"

After each completed workout, the "Training completed" will be marked next to it.

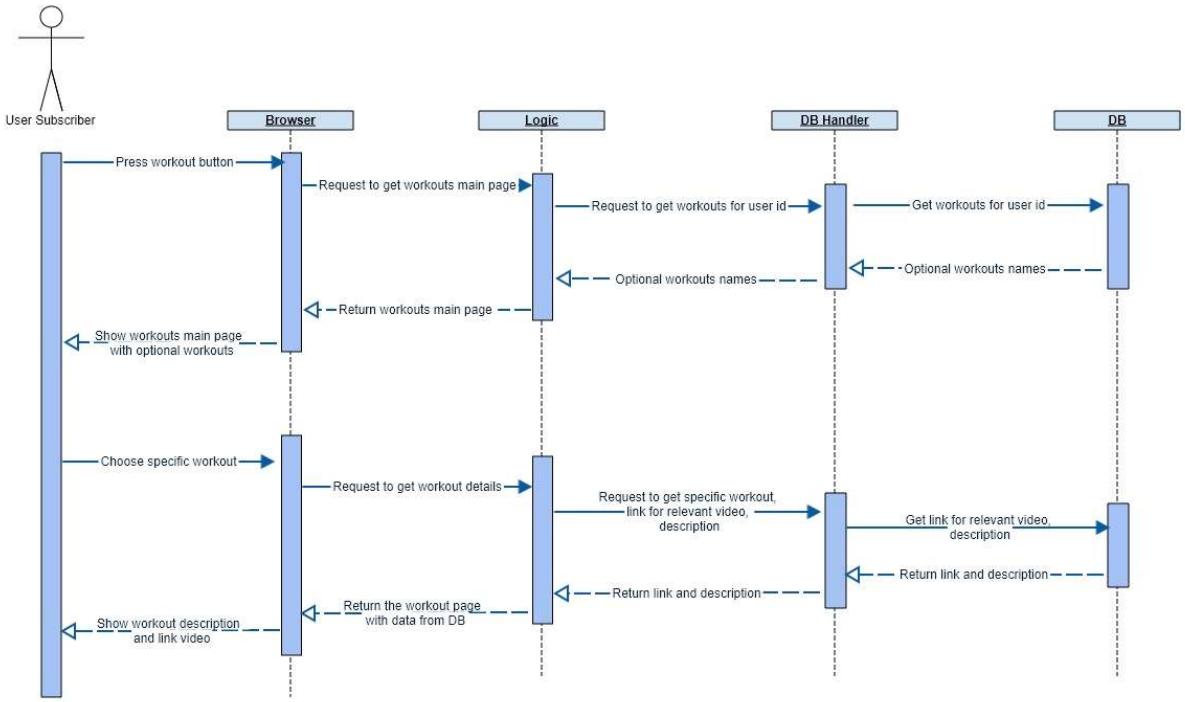
"To do"

Any workout that has not yet been completed will be marked with "To do"

4.1.4.4 Exceptions

- DB access errors (write / read)
- Local DB access errors (write / read)

4.1.4.5 MSC



4.1.5 Show Menu (for Subscription user)

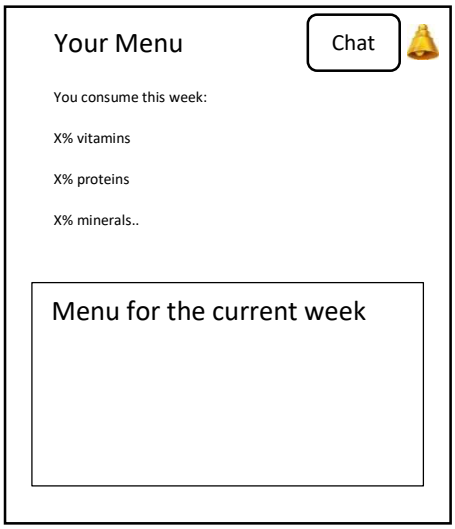
4.1.5.1 Input

Show menu – “Show menu” button was pressed in the home page

4.1.5.2 Processing

"Show menu" button

- The subscriber user presses "Show menu" button from the home page
- The data about the menu is taken from table “subscriber_to_menu” there the menu id is taken, and the menu details is taken from table “menu”.
  "Subscriber_to_menu" table – this table holds the menus of the user and which menu is active.
- The Logic component gets the details and create with the information from DB this table:



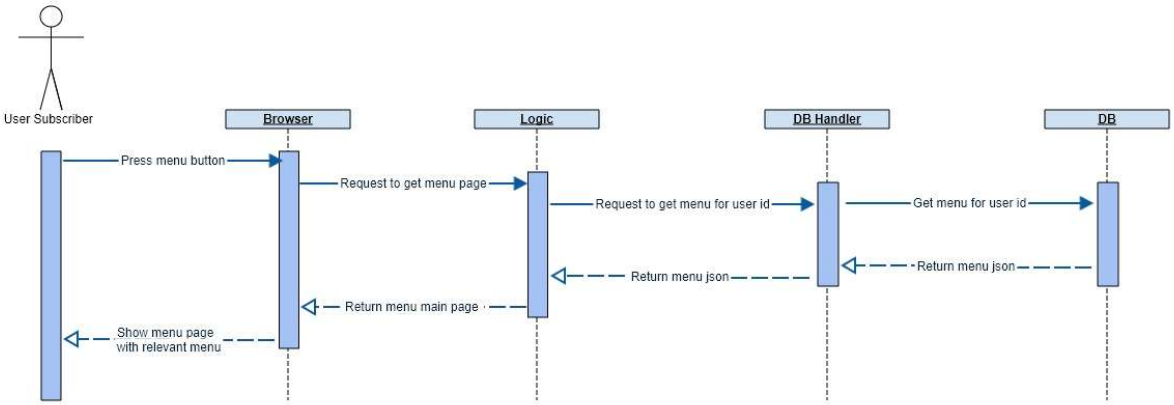
4.1.5.3 Output

Current status of the nutritional values consumed in the current week and menu for the rest of the week.

4.1.5.4 Exceptions

- DB access errors (write / read)
- Local DB access errors (write / read)

4.1.5.5 MSC



4.1.6 Chat with Expert

4.1.6.1 Description

Every page the user will be able to start a chat with a nutritionist,
Every page will be button “Chat with expert”, when the user presses this button, he redirect to different page there he will be able to choose “nutritionist or Trainer”.

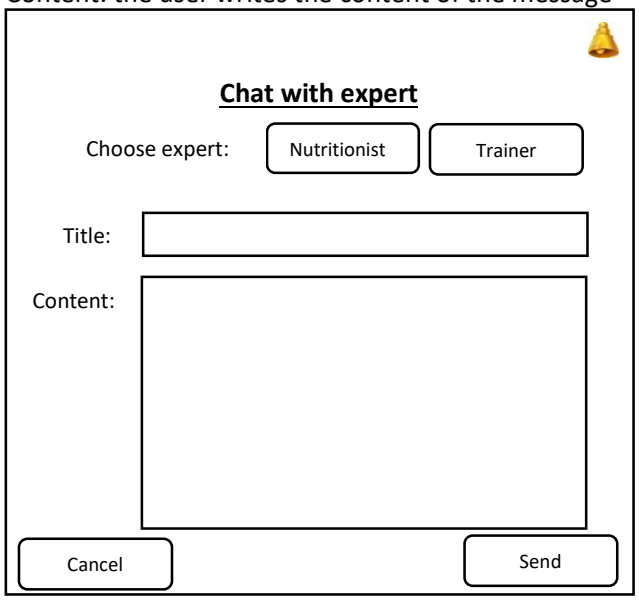
4.1.6.2 Input

All pages have button of “chat”.
Also, all pages have a bell , when user have unread message, he will see the bell with mark.
After user press the button different page with two buttons opened, and the chat options where

he inserts the content of the chat.

4.1.6.3 Processing

- choose the expert: two buttons are enables to choose, “Nutritionist” of “Trainer”, only when one of the selected the other option in the screen is enable.
- Title: the user writes the title of the message - free text
- Content: the user writes the content of the message – free text



4.1.6.4 Output

"Send" button

After finish writing the message the user presses the button, if some fields are not full, error message will pop up to notify the user to fill all fields.
Then return to the same page.

If all fields are full, message “success to send message” will pop up and user will return to same page that he was before starting the chat.

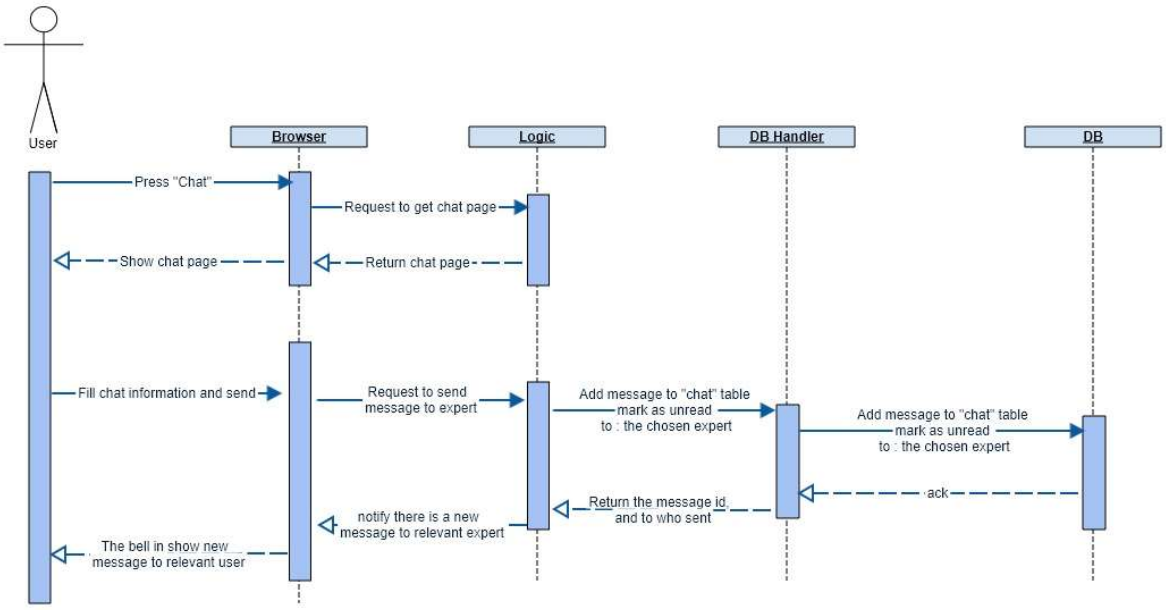
"Cancel" button

After cancel - notify the user and return to same page that he was before starting the chat.

4.1.6.5 Exceptions

- DB access errors (write / read)
- Local DB access errors (write / read)

4.1.6.6 MSC



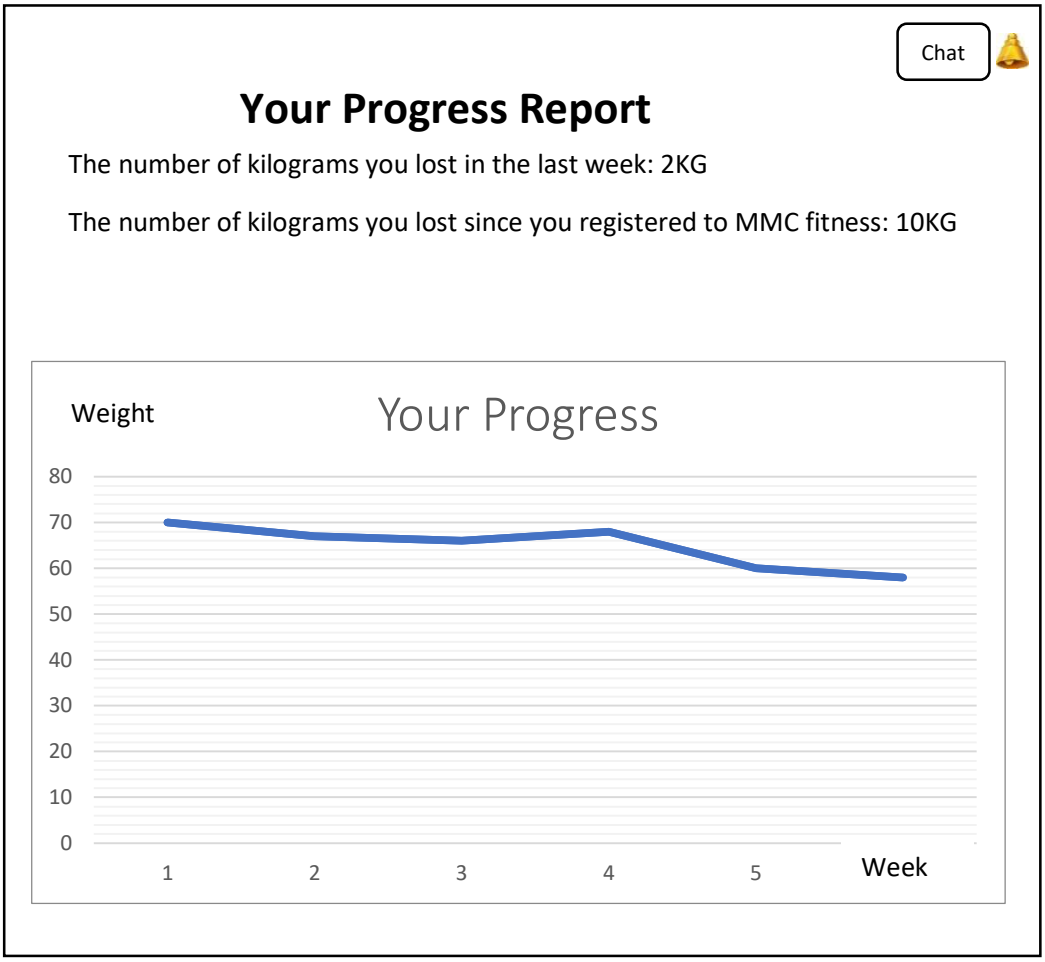
4.1.7 Generate Report

4.1.7.1 Input

Generate report- “generate report” button was pressed from the home page

4.1.7.2 Processing

- The subscriber user pressed “Generate report” button from home page
- A request for report was sent to the Logic component
- The Logic component got the data from the “system events” table in the DB. This table should contain the weight and fat percentage of the user with the timestamp.
- On each login, when the user enters his current weight, the “system events” table will be updated with this information and the timestamp
- The Logic component should generate a progress report and return new page with this report:



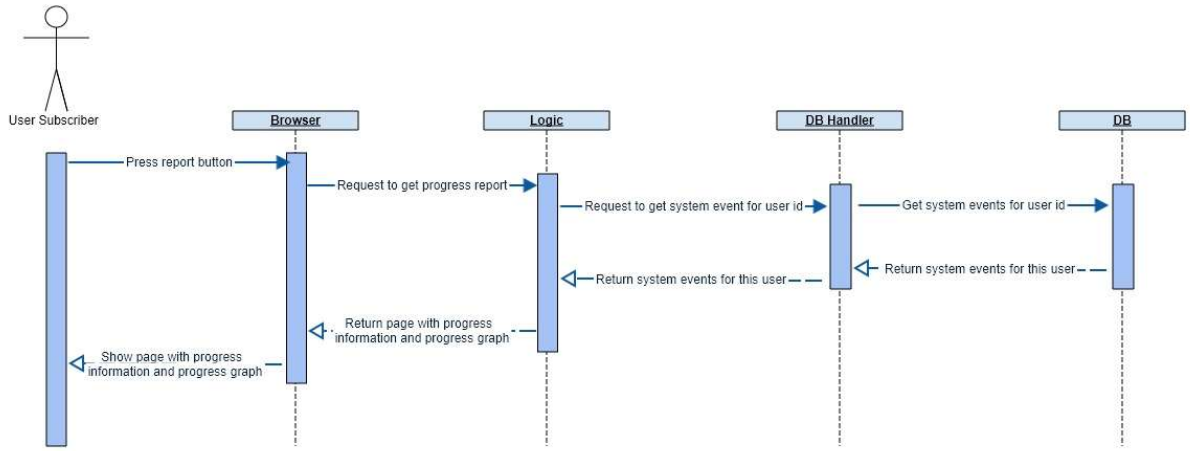
4.1.7.3 Output

A report that will display the progress of the subscriber. The report should generate a graph that will reflect the user weight per each week since he registered to the MMC fitness system.

4.1.7.4 Exceptions

- DB access errors (write / read)
- Local DB access errors (write / read)

4.1.7.5 MSC



Every three months a personal meeting will be schedule for the Subscription user with trainer and nutritionist.

In this meeting the specialists will fits to user new menu in accordance with his progress in the last 3 months. Additionally, will be fit new training set according to his shape status.

4.1.8 Periodic condition assessment

4.1.8.1 Input

The user logged in to the system

4.1.8.2 Processing

- After three months since the user was registered to the system the Logic component will generate new menu and training set and update 2 tables in the DB:
    1. Subscriber_to_menu – this table holds the menus of the user and which menu is active.
    2. Subscriber_to_workout – this table holds the workouts of the user and which workout is active.
- The user will get a notification in the home page that new menu and workouts were generated according to his progress

4.1.8.3 Output

A notification for the user that informing that the menu and training set have been updated for him by certified trainer and nutritionist.

5. Logical Database Requirements