

Software Studio

Group 4



Faculty of Computer Science

NourishBoost

"Fuel Your Wellness Journey with
NourishBoost: Track, Thrive, Transform!"

- Ali Tekin (381778) / alitekin@student.agh.edu.pl
- Katsiaryna Yakubava (419179) / yakubava@student.agh.edu.pl
- Pravallika Mandavalli (419222) / mandavalli@student.agh.edu.pl
- Emanuel Borges (414067) / borges@student.agh.edu.pl

Professor: Paweł Skrzyński

Table of Contents

Project Description.....	3
Post Description – User Stories And Their Acceptance Criteria (24 Total).....	4

Project Description

NourishBoost is your comprehensive solution for maintaining a healthy lifestyle through mindful eating and nutrition tracking. In today's fast-paced world, keeping track of your daily food intake and ensuring you meet your nutritional goals can be challenging. That's where NourishBoost comes in.

Our app provides users with a user-friendly platform to easily track their meals' calories and macronutrients throughout the day or over extended periods. Whether you're striving to lose weight, gain muscle, or simply maintain a balanced diet, NourishBoost empowers you to take control of your nutrition.

With NourishBoost, setting and monitoring nutritional goals has never been easier. Whether you aim to increase your protein intake, reduce sugar consumption, or optimise your overall calorie intake, our app offers customisable goal-setting features tailored to your needs and preferences.

But NourishBoost is more than just a calorie-counting app; it's a holistic tool for promoting wellness and vitality. By providing valuable insights into your dietary habits and patterns, NourishBoost helps you make informed choices about the foods you consume, improving your overall health and well-being.

Whether you're a fitness enthusiast, a health-conscious individual, or someone simply looking to cultivate better eating habits, NourishBoost is here to support you on your journey to a healthier, happier you. Say goodbye to guesswork and hello to a nourished, revitalised lifestyle with NourishBoost.

Post Description – User Stories And Their Acceptance Criteria (24 Total)

Must-be user stories:

- 1) **As a user, I should be able to access the application without signing up for an account.**

Acceptance criteria:

- a) Upon starting the application for the first time, I should be immediately greeted with the main page and not bothered by the “log in” or “create account” menus.

- 2) **As a user, I must be able to register a food item.**

Acceptance criteria:

- a) On the main page, I should be greeted with a rectangular bottom-level navigation bar consisting of three buttons, the leftmost of which should be responsible for taking me to the food look-up page.
- b) Upon clicking on such a button, I should be taken to the food look-up page. Initially, there should be no food items. Therefore, I should be greeted by an empty page (possibly with a relevant message informing me that no items have been registered yet) with a lone “register item” button. This “register item” button should be on the top-right corner of the page, alongside the item search field, which should take over the entire rest of the available space.
- c) Upon clicking on the “register item” button, I should be taken to the food item registration page.
- d) On the food item registration page, I should be prompted to enter:
 - i) The name by which the food item should be identified.

Validation criteria:

- (1) It cannot be empty;

(2) It must be unique;

(3) It must not exceed 50 characters.

- ii) The item's nutritional value in the units of measurement (e.g., per 100g, per 100 ml, etc.) in which I intend to measure it afterwards.

Validation criteria:

(1) It cannot be empty;

(2) In the case of metric units, it should consist of numbers only;

(3) In the case of non-metric units (cups, tablespoons, etc.), it should consist of either integers or fractions.

- iii) The amount of kilocalories in the item per the chosen measurement standard.

Validation criteria:

(1) It cannot be empty;

(2) It should consist of numbers only.

- iv) The amount of each macronutrient (fat, carbohydrates, protein, and their relevant subdivisions) in the item per the chosen measurement standard.

- e) I should be presented with a "register item" button at the bottom of the food item registration page. This button should only be enabled once all the relevant information has been adequately filled out. By clicking on this button, the food item should be promptly added to my personal database.

- f) From this point onwards, I should be able to:

i) Find this item on the food-item-lookup table;

- ii) Add this item to a food entry for any specific meal (more on that functionality later).

3) As a user, I must be able to read all registered food items.

Acceptance criteria:

- a) Upon clicking the leftmost button of the bottom-level navigation bar, I should be taken to the food-item lookup menu.
- b) On the food-item lookup menu, I should see a list of all the items I have registered so far by their names. Each item on the list should consist of a rectangle inside of which I could find the food item's name and whose width comprises the entire width of the screen.
- c) On the top of the screen, just below the search bar and the "register item" button, I should find a drop-down menu with selectable sorting options. The sorting options should be as follows:
 - i) Registration date (ascending order);
 - ii) Registration date (descending order);
 - iii) Alphabetic order (ascending);
 - iv) Alphabetic order (descending);
 - v) Most consumed to least consumed;
 - vi) Least consumed to most consumed.
- d) Once I click on a specific food item, I should be taken to its nutritional information page, which should display:
 - i) All the information entered during the item's registration. The display format should be the very same as that of the form used to first enter the item's information; the only difference is that now such form should be static, and not editable;
 - ii) An "edit item" button (preferably on the bottom of the page, alongside the "delete item" button);
 - iii) A "delete item" button (preferably on the bottom of the page, alongside the "edit item" button).

4) As a user, I must be able to edit any food item on the item-lookup table.

Acceptance criteria:

- a) Once I have clicked on the “edit item” button, I should be redirected to the same form that was used to first enter all the item’s information, with the following particularities:
 - i) All of the fields should be pre-filled with the item’s current properties to allow for the editing of the information.
 - ii) Instead of a single “register item” button, I should have two buttons:
 - (1) One that should take its place: an “edit item” button;
 - (2) Another that should allow me to cancel the editing of the item without making any changes to the database: a “cancel” button.
- b) Once the item has been edited, the change should be promptly written to the database, altering the item on the item-lookup table.
- c) However, previous food entries registered by me (more on that functionality later) should NOT BE ALTERED.

5) As a user, I must be able to delete any food item from the item-lookup table.

Acceptance criteria:

- a) I should be prompted for confirmation once I have clicked on the “delete item” button (see must-be user stories: number 3, point iii).
- b) Once I confirm that I wish to delete the item, it should be promptly deleted from the item-lookup table.
- c) However, previous food entries registered by me (more on that functionality later) should NOT BE ALTERED.

6) As a user, I must be able to register a food entry in my journal so that I can keep track of my daily nutritional intake.

Acceptance criteria:

- a) The middle button of the bottom-level navigator should be pre-selected when the app is started and should take me to the application's main page.
- b) On the main page, I should be presented with a list of my expected daily meals, each of which should be presented as a rectangle on the screen, inside of which its name should be visible. The rectangles should all be stacked vertically. On the right extremity of each daily meal's list item, their subtotal caloric intake should be displayed). Initially (before I issue any specific configuration), I should be presented with the following meals:
 - i) Breakfast;
 - ii) Morning snack;
 - iii) Lunch;
 - iv) Afternoon snack;
 - v) Dinner;
- c) Upon selecting any of these meals, it should expand to display:
 - i) A list of all the food items consumed (together with their respective amount of calories) on that meal;
 - ii) An option to add another food entry to that list.
- d) Upon selecting to add another food entry to that list, I should be taken to the food-lookup table, wherein all my previously registered food items should be displayed as available options. I should then be prompted to choose one.
- e) Upon selecting a food item from the food-lookup table, I should be redirected to a page on which I would be presented with the selected item's nutritional information and prompted to enter the amount of that food that I took or desire to take during its assigned meal. The food amount should be given in the measurement chosen as the standard during the item's registration.
- f) Upon entering the consumed amount, the nutritional values (quantities of macronutrients and kilocalories) should be updated in real-time.

- g) On the same page, fixed to the bottom of the screen, I should find an “add entry” button that should add the specified quantity of that specific item to my food diary. Such a button should only be enabled once the relevant field (in this case, quantity) has been adequately filled.
- h) Upon adding an item to my food diary, I should be redirected to the main screen. The list representing the meal to which food has just been added should already be expanded. All the values relating to the daily nutritional and caloric intake should be updated and displayed (more on this functionality later).
- i) Each page of the food journal (that is, everything consumed during each of the daily meals) should be dated. By default (i.e., when opening the application), the main page should be dated to the current date, which should be updated automatically and synchronised with my phone’s date and time settings.
- j) Finally, on the very top of the main page, the subtotal of calories and the amounts of each macronutrients consumed throughout all the daily meals should be presented.

7) As a user, I must be able to view all previous food entries, regardless of when they were registered.

Acceptance criteria:

- a) I should be able to find an unobtrusive calendar icon at the top right of the application's main page.
- b) I should be presented with a calendar view after clicking on such an icon.
- c) Each day should be displayed as a square.
- d) In case there were any food entries on a day inside its square, there should be:
 - i) On the upper part, the number of the day;
 - ii) On the lower part, the amount of kilocalories consumed is listed.
- e) Upon clicking on a given day’s square, I should be redirected to its daily food journal page (which is the same as the main page, albeit referring to a different date).

- f) On the top-left corner of the application's main page, I should be informed of the date to which the currently displayed food journal page belongs.

8) As a user, I must be able to edit a previously registered food entry.

Acceptance criteria:

- a) Upon expanding a meal's list of entries, I should be able to select each individual entry.
- b) Upon selecting each individual entry, it should expand to reveal two choices: "edit" and "delete."
- c) Upon selecting "edit," I should be redirected to the same screen on which I first entered the selected item's consumed amount (with its value pre-filled) and be given the possibility to alter it.
- d) At the bottom of the page, instead of a button prompting me to add this entry, the button should convey the information that I am editing instead. It should only be enabled once the relevant field (quantity) has been adequately filled.
- e) Upon editing the entry, I should be redirected to the main screen. The list representing the meal whose entry has just been edited should be already expanded. All the values relating to the daily nutritional and caloric intake should be updated and on display (more on this functionality later).

9) As a user, I must be able to delete a previously registered food entry.

Acceptance criteria:

- a) Upon expanding a meal's list of entries, I should be able to select each individual entry.
- b) Upon selecting each individual entry, it should expand to reveal two choices: "edit" and "delete."
- c) Upon selecting "delete," I should be prompted for confirmation.
- d) Once I have confirmed my desire to delete the food entry, the relevant entry should be promptly deleted.
- e) After the deletion, all the values relating to the daily nutritional and caloric intake should be updated and displayed in real-time (more on this functionality in the next user story).

10) As a user, I should be clearly informed of my daily intake of calories and macronutrients.

Acceptance criteria:

- a) On the very top of the application's main page (that is, on each day's food journal), above the list of expected daily meals, I should be informed of the number of calories and each macronutrient consumed over the entire day.

11) As a user, I must be able to export all my local data to a file.

Acceptance criteria:

- a) After swiping up on the bottom-level navigation menu, I should be presented with two more options: export and import data.
- b) Upon selecting “export data,” a file should be generated with all my data so far in a format that I could then import back into the app if needed.
- c) I should then be displayed my cellphone’s default “share” options, and the exported data should be the contents of such sharing, i.e., the file in which SQLite is currently storing all of its data (most probably either a *.db* or a *.sqlite* file – the specifics of this implementation can only be truly determined during development, and the exact file type is not at all important. What is important is that, whatever the exported file type, it should be accepted during importation).
- d) Upon clicking anywhere on the application, such menu extensions (i.e., the “export” and “import” options) should retract.

12) As a user, I must be able to import data from a file.

Acceptance criteria:

- a) Upon selecting "import data," I should be warned that importing data from a file will permanently delete all my data and advised to export it prior to continuing with the importation process. I should then be given the choice to continue or to go back and export my data first.
- b) Upon choosing to continue, I should be prompted to select a file from my device that contains a journal's data.
- c) Upon selecting a file, the app should check whether it is a valid file or not. A valid file will most probably be of either the *.db* or *.sqlite* file extension (most probably either a *.db* or a *.sqlite* file – the specifics of this implementation can only be truly determined during development and the exact file type is not at all important. What is important is that, whatever the exported file type, it should be accepted during importation).
- d) If it is not, I should be informed of it by a relevant warning message and prompted to try again.
- e) If it is, I should be informed that the importing operation was successful and once again warned that proceeding with the importation will permanently delete all my data. I should then be given the option to either continue the importation process or cancel it altogether.
- f) If I choose to continue, all my data should be overwritten by the imported data, and I should be redirected to the application's main page on today's date.
- g) If I choose to cancel the importing operation, clicking anywhere on the screen should cause the newly expanded section of the bottom-level page navigator (containing the "export data" and "import data" options) to collapse.

Should-be user stories:

- 1) **As a user, I should be able to register weight entries on the app to keep track of my weight over time**

Acceptance criteria:

- a) The rightmost button of the bottom-level navigator should take me to the weight-tracking section of the app.
- b) The app's weight tracking section should be divided into two equally sized sections. The upper section should have an input field where I can enter my current weight. This input field should start empty for each new entry.

Validation criteria:

- i) It cannot be empty;
 - ii) It should consist of numbers only.
-
- c) On the right side of the input field (preferably inside the input itself), there should be an option to toggle the chosen measurement units for the current weighting between kilograms and pounds.
 - d) A button should be on either the right side of the input field or beneath it to effectuate the weight-in. By clicking on this button, the data concerning this weight entry should be written to my weight diary (that is, the local database).
 - e) Each weight entry should be assigned to a specific date.

2) As a user, I should be able to view all my previous weight entries.

Acceptance criteria:

- a) On the top-right corner of the weight tracking section of the app, I should be able to find a calendar icon similar to the one on the main page. The way they should work should be absolutely the same, with the only difference being that the weight at that date should be displayed (instead of the caloric intake) on each square representing a day.
- b) On the top-left corner of the weight tracking section of the app, I should be informed of the date to which the currently displayed weight refers.
- c) The bottom part of the weight tracking section of the app should consist of a graph of nodes connected by ridges on which all previous weight entries are displayed. The X-axis of this graph should be the date of the weighting-in; the Y-axis should be the weight measured at the time.
- d) I should be able to scroll the graph to the left (up to my first weight-in) and to the right (up to the current date) to view my weight fluctuations over time.

3) As a user, I should be able to edit any weight entry on the app, regardless of when it was registered.

Acceptance criteria:

- a) In the weight-tracking section of the application, for each already registered weight entry (to which I could arrive through the calendar navigation), I should be able to overwrite the weight in the weight-in input field, which I should find pre-filled with the entry's original value.
- b) Upon clicking on the weight-in button, the weight value for that day should be updated in my journal (that is, in the local database).

- 4) **As a user, I should be able to delete any weight entry on the app regardless of when it was registered.**

Acceptance criteria:

- a) On the weight-tracking section of the application, for each already-registered weight entry (to which I could arrive through means of the calendar navigation), I should be presented with the option to delete the weight entry.
- b) Upon selecting this option, I should be prompted for confirmation.
- c) Upon confirming my desire to delete the weight entry, it should be deleted from my journal (that is, from the local database), and the top part of the weight-tracking section of the app (referring to that day) should now exhibit an empty weight input field.

- 5) **As a user, I should be able to register custom recipes as food items and be provided with a nutritional calculator when preparing them.**

Acceptance criteria:

- a) Before registering an item, I should be asked which type of item it is; the choices would be:
 - i) Ready-to-eat or ingredient;
 - ii) Recipe.
- b) In the case of either ready-to-eat meals or ingredients, registering would be done as previously mentioned (see must-be user stories, number 6).
- c) However, in the case of recipes, I should be taken to the dedicated recipe preparation screen, which would work as follows:
 - i) I should be prompted for the recipe's name.
 - ii) At first, I should be presented with an empty list of ingredients and a single "add ingredient" button.
 - iii) Upon clicking on the "add ingredient" button, I should be taken to the food-lookup table.
 - iv) If my ingredient is not on the food-lookup table, I should be able to register it (that is, be redirected to the food-item register screen) and add it to my recipe (that is, come back to the recipe preparation

screen) without losing the already input data for it.

- v) I should be able to see the list of ingredients added so far at each step of the way. Furthermore, I should be able to:

- (1) Edit the quantities for a given ingredient;

- (2) Remove an ingredient from the recipe altogether.

- vi) I should be presented with a "register recipe" button at the bottom of the recipe registration page. This button should only be enabled once all the relevant information has been adequately filled out. By clicking this button, the recipe should be promptly added to my food-lookup list.
- vii) From now on, instead of having to add every ingredient of a meal to my food diary, I should be able to simply add the quantity of the recipe consumed and be informed of the amounts of macronutrients and calories it amounts to.

6) As a user, I should be able to see all registered custom recipes.

Acceptance criteria:

- a) The food-lookup list should have two sections, both selectable by clicking on their dedicated buttons at the very top of the list.
- b) The default section would be the regular food items section and would work as explained above (see must-be user stories, number 3), displaying all items except for the custom recipes.
- c) The recipes section would be dedicated entirely to custom recipes and would work in the exact same way, except it would display custom recipes exclusively.
- d) Identically to regular items, once I click on a specific recipe, I should be taken to its nutritional information page, which should display:
 - i) All the information entered during the item's registration;
 - ii) An "edit recipe" button;
 - iii) A "delete recipe" button.
- e) All the information entered during the item's registration;

7) As a user, I should be able to edit any registered custom recipe.

Acceptance criteria:

- a) Once I have clicked on the “edit recipe” button, I should be redirected to the same form that was first used to enter all the items that comprise the recipe, with the following particularities:
 - i) All of the fields should be pre-filled with the recipe’s current properties to allow for editing.
 - ii) Additionally, the recipe’s item list should also be present, and I should be able to perform the very same operations that I could during its registration. Those are:
 - (1) Add an ingredient to the recipe;
 - (2) Edit quantity for a given ingredient;

Validation criteria:

- (a) It cannot be empty;
 - (b) In the case of metric units, it should consist of numbers only;
 - (c) In the case of non-metric units (cups, tablespoons, etc.), it should consist of either integers or fractions.
- (3) Remove an ingredient from the recipe altogether.
- iii) Instead of a single “register recipe” button, I should have two buttons:
 - (1) Firstly, it should take its place: an “edit recipe” button;
 - (2) Second, it should allow me to cancel the editing of the recipe without making any changes to the database: is a “cancel” button.
- b) Once the item has been edited, the change should be promptly written to the database, altering the item on the item-lookup table.
- c) However, previous food entries registered by me (more on that functionality later) should NOT BE ALTERED.

8) As a user, I should be able to see all registered custom recipes.

Acceptance criteria:

- a) I should be prompted for confirmation once I have clicked on the “delete recipe” button.
 - b) Once I have confirmed that I indeed wish to delete the recipe, the recipe should be promptly deleted from the item-lookup table.
 - c) However, previous food entries I registered should NOT BE ALTERED.
-

Could-be user stories:

1) As a user, I can register a custom expected daily meal.

Acceptance criteria:

- a) On the main page, below all expected daily meals, I could find an additional “add custom daily meal” item.
- b) By clicking on it, I would be prompted to write the name of my custom daily meal on a pop-up box.

Validation criteria:

- i) It cannot be empty;
 - ii) It cannot exceed 20 characters;
 - iii) It should be unique.
- c) By providing the name for my custom daily meal and selecting “add custom meal”, another item would be added to the list of expected daily meals (see must-be user stories, number 6 letter b).

2) As a user, I could be able to reorganise the display order of the expected daily meals.

Acceptance criteria:

- a) By pressing and holding on to an expected daily meal item, I could go into “reorganisation mode”, during which:
 - i) All expected daily meal lists collapse;
 - ii) All expected daily meal lists become movable.
- b) By moving the expected daily meal lists around, I would be able to reorder them.
- c) All alterations to the expected daily meal order should live on to the next day but should NOT cause changes to any previously filled food journal pages.
- d) Clicks anywhere on the page (except for the edit and trash icons on each expected meal component; more on that soon) would deactivate reorganisation mode.

3) As a user, I could be able to edit an expected daily meal.

Acceptance criteria:

- a) Once in “reorganisation mode”, the caloric intake summaries on the right-hand-side extremity of each expected daily meal component could be replaced by two icons:
 - i) A pencil icon;
 - ii) A trash can icon.
- b) Upon clicking on the pencil icon, I would be prompted to edit the expected daily meal’s name.
- c) All alterations to the expected daily meal names should continue into the next day but should NOT affect any previously filled food journal pages.
- d) Clicks anywhere on the page (except for the edit and trash icons of each expected meal component) would deactivate reorganisation mode.

4) As a user, I could be able to delete an expected daily meal.

Acceptance criteria:

- a) Upon clicking the trash can icon, I could be prompted to confirm whether I want to delete the expected daily meal.
- b) If there are currently registered food entries under it, upon confirming my choice, I would be warned that all currently registered food entries are going to be permanently deleted (ideally, such entries could be shown) and prompted for my confirmation once again.
- c) Upon confirming my choice, the expected daily meal section and all the food entries registered under it would be permanently deleted.
- d) Clicks anywhere on the page (except for the edit and trash icons of each expected meal component) would deactivate reorganisation mode.

5) As a user, I can be informed of my daily macronutrient intake through a pizza graph.

Acceptance criteria:

- a) On the top of the main page, instead of displaying the macronutrients intake by numbers only, a pizza graph could be employed to show me the proportions of:
 - i) Protein;
 - ii) Carbohydrates;
 - iii) And fat

That comprises my caloric intake for the day.

Tech Stack:

The application is going to be built with React Native (JavaScript).

The database is going to be implemented using SQLite.