**Daily Meal Planner**

**Author (s): \_Zhongxing Qin\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: \_10/07\_\_\_\_\_\_\_\_\_\_**

**Version: \_\_\_\_\_\_\_\_\_\_\_**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE NAME:** | Recommend recipes based on available cooking time | | **USE CASE TYPE** |
| **USE CASE ID:** | 1 | | Business Requirements: **🞎** |
| **PRIORITY:** | medium | | System Analysis: 🗹 |
| **SOURCE:** | Team meeting | |  |
| **PRIMARY BUSINESS ACTOR** | planner | | |
| **PRIMARY SYSTEM ACTOR** | Planner UI | | |
| **OTHER PARTICIPATING ACTORS:** | None | | |
| **OTHER INTERESTED STAKEHOLDERS:** | None | | |
| **DESCRIPTION:** | The planner can input the amount of time they have available for cooking. Based on the planner's input, the system will recommend recipes that fit within that time range. | | |
| **PRE-CONDITION:** | The system must have a database of recipes that includes the estimated cooking time for each recipe. | | |
| **TRIGGER:** | The planner navigates to the recipe recommendation page and inputs their expected cooking time. | | |
| **TYPICAL COURSE** | Actor Action | System Response | |
| **OF EVENTS:** | **Step 1**: The planner navigates to the recipe recommendation page |  | |
|  |  | **Step 2**: The system displays a question asking the planner to enter the expected cooking time | |
|  | **Step 3**: The planner enters the expected cooking time and submits it to the system |  | |
|  |  | **Step 4:** The system searches the database for matching recipes based on the planner's input time and displays them to the planner. | |
|  |  |  | |
|  |  |  | |
| **ALTERNATE COURSES:** | If no recipes match the planner’s time constraint, the system will display a message: "No recipes found that match your time." | | |
|  | If the planner input is invalid or left empty, the system will display an error message asking the planner to enter valid time. | | |
|  |  | | |
| **CONCLUSION:** | The planner can see the list of recipes recommended by the system that match their available time. | | |
| **POST-CONDITION:** | The planner successfully views a list of recommended recipes. | | |
| **BUSINESS RULES** | * All recipes in the system must have associated preparation and cooking times. * The planner can only input a reasonable range of cooking time | | |
| **IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS** | * The system must have a recipe database with preparation and cooking times. | | |
| **ASSUMPTIONS:** | * The planner will input a reasonable time, and the system will have sufficient recipe data to make recommendations for planner. | | |
| **OPEN ISSUES:** | None | | |

**Daily Meal Planner**

**Author (s): \_\_\_\_\_Maheswari\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_10/7/2024\_\_\_\_\_\_\_\_**

**Version: \_\_\_\_\_\_\_\_\_\_\_**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE NAME:** | **Calories for Each Food Item in Meal Plan** | | **USE CASE TYPE** |
| **USE CASE ID:** | **2** | | **Business Requirements: 🞏** |
| **PRIORITY:** | **High** | | **System Analysis: 🗹** |
| **SOURCE:** | **Team discussion** | |  |
| **PRIMARY BUSINESS ACTOR** | **Planner (meal planner )** | | |
| **PRIMARY SYSTEM ACTOR** | **Meal Planner System** | | |
| **OTHER PARTICIPATING ACTORS:** | * **Database server** * **API providing nutritional data** | | |
| **OTHER INTERESTED STAKEHOLDERS:** | * **Health-conscious users** * **Nutritionists** | | |
| **DESCRIPTION:** | **The planner selects or creates a meal plan. For each food item in the meal plan, the system retrieves and displays the corresponding calorie count. This information helps users monitor and manage their caloric intake.** | | |
| **PRE-CONDITION:** | * **The meal plan should already have food items added to it.** | | |
| **TRIGGER:** | **The planner opens a meal plan and selects the option to view calories for each food item.** | | |
| **TYPICAL COURSE** | **Actor Action** | **System Response** | |
| **OF EVENTS:** | **Step 1. The planner selects a meal plan from their profile and clicks on "Show Calories."** | **Step 2. The system retrieves the meal plan from the database.** | |
|  |  | **Step 3. The system displays a list of food items in the selected meal plan.** | |
|  |  | **Step 4. The system fetches calorie data for each food item from the database.** | |
|  |  | **Step 5. The system displays the calorie count next to each food item in the meal plan.** | |
|  |  | **Step 6. The system calculates the total calorie count for the entire meal plan by summing all food items' calories and displaying it alongside individual item calories.** | |
|  |  |  | |
|  |  |  | |
|  |  |  | |
| **ALTERNATE COURSES:** | * **If the system cannot retrieve calorie data, it shows an error message, allowing planner to retry.** | | |
|  | * **If the food item does not have associated calorie data, the system displays "Data Not Available" instead of a calorie count.** | | |
|  |  | | |
|  |  | | |
|  |  | | |
|  |  | | |
|  |  | | |
|  |  | | |
| **CONCLUSION:** | **The planner successfully views the calorie information for all food items in their meal plan.** | | |
| **POST-CONDITION:** | **The system remains available for further interaction. The planner can proceed to modify their meal plan or view other details.** | | |
| **BUSINESS RULES** | * **The system must ensure the accuracy of the calorie data retrieved.** * **The system should update any changes to calorie data from the source as needed.** | | |
| **IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS** | * **The system must be able to handle API or database calls to retrieve calorie information in real time.** * **The feature must be compatible with various types of meal plans, including those with custom food items.** | | |
| **ASSUMPTIONS:** | * **Calorie data for all food items is available and accurate in the source database or API.** | | |
| **OPEN ISSUES:** | * **Handling the display of calorie information for foods without readily available data.** | | |

**Daily Meal Planner**

**Author (s): \_\_\_\_\_\_Sarayu Pacca\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_10/7/24\_\_\_\_\_\_\_**

**Version: \_\_\_\_\_1.0\_\_\_\_\_\_**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE NAME:** | Custom meal plan option | | **USE CASE TYPE** |
| **USE CASE ID:** | 3 | | Business Requirements: **🞏** |
| **PRIORITY:** | High | | System Analysis: 🗹 |
| **SOURCE:** | Team Discussion | |  |
| **PRIMARY BUSINESS ACTOR** | Planner | | |
| **PRIMARY SYSTEM ACTOR** | Meal Planner Application (UI and Backend) | | |
| **OTHER PARTICIPATING ACTORS:** | * Database, Recipe API | | |
| **OTHER INTERESTED STAKEHOLDERS:** | * N/A | | |
| **DESCRIPTION:** | This use case describes how a planner can select a "Custom Plan" option when generating a meal plan and specify the number of days for their meal plan. The system then generates a meal plan based on the selected number of days and planner preferences. | | |
| **PRE-CONDITION:** | The "Custom Plan" option is displayed alongside other meal plan options (e.g., Weekly, Daily). | | |
| **TRIGGER:** | The planner selects the "Custom Plan" option from the meal planning interface. | | |
| **TYPICAL COURSE** | Actor Action | System Response | |
| **OF EVENTS:** | **Step 1**: The planner selects the "Custom Plan" option. | **Step 2:** The system displays an input mechanism to allow the planner to specify the number of days. | |
|  | **Step 3:** The planner inputs the number of days. | **Step 4:** The system verifies the input and retrieves meal options based on the planner’s preferences and constraints. | |
|  | **Step 5:** The planner confirms the meal plan preferences. | **Step 6:** The system generates a custom meal plan based on the specified number of days and planner preferences. | |
|  | **Step 7:** The planner views the generated meal plan. | **Step 8:** The system displays the custom meal plan to the planner. | |
|  |  |  | |
|  |  |  | |
|  |  |  | |
|  |  |  | |
| **ALTERNATE COURSES:** | If the planner fails to input a valid number of days, the system prompts the planner to correct the input. | | |
|  | If the system fails to retrieve meal options, it notifies the planner and suggests retrying. | | |
|  |  | | |
|  |  | | |
|  |  | | |
|  |  | | |
|  |  | | |
|  |  | | |
| **CONCLUSION:** | The planner successfully creates a custom meal plan based on their specific preferences and the number of days they selected. | | |
| **POST-CONDITION:** | The custom meal plan is stored in the system and accessible to the planner.  The planner can proceed to view, edit, or save the plan. | | |
| **BUSINESS RULES** | * The meal plan must adhere to the planner’s dietary preferences and restrictions. | | |
| **IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS** | * The meal plan data is saved in the cloud database. | | |
| **ASSUMPTIONS:** | * Planner will enter a valid number of days when specifying their custom meal plan (e.g., no negative numbers, unrealistic large numbers) | | |
| **OPEN ISSUES:** | Finalize how the planner will input the number of days (dropdown or text input). | | |

**Daily Meal Planner**

**Author (s): Willy Date: 10/7/2024**

**Version: \_\_\_\_\_\_\_\_\_\_\_**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE NAME:** | Create a meal plan with a specified number of meals for each day and view the same number of meals presented for each day in the meal. | | **USE CASE TYPE** |
| **USE CASE ID:** | 4 | | Business Requirements: **🞏** |
| **PRIORITY:** | High | | System Analysis: 🗹 |
| **SOURCE:** | Team Discussions | |  |
| **PRIMARY BUSINESS ACTOR** | Planner | | |
| **PRIMARY SYSTEM ACTOR** | Meal Planner Program | | |
| **OTHER PARTICIPATING ACTORS:** | * N/A | | |
| **OTHER INTERESTED STAKEHOLDERS:** | * N/A | | |
| **DESCRIPTION:** | Implement a bug fix where the number of meals for the meal options reflect accurately on the meal plan. The planner should be able to see the desired number of meals they selected in the plan. | | |
| **PRE-CONDITION:** | The planner has given answers to all prompt questions. | | |
| **TRIGGER:** | The planner creates the meal plan. | | |
| **TYPICAL COURSE** | Actor Action | System Response | |
| **OF EVENTS:** |  | **Step 1**: The system redirects the planner to a display of the meal plan. | |
|  | **Step 2**: The planner views the first day of the meal plan with the correct number of meals displayed. |  | |
|  | **Step 3**: The planner navigates to pages for subsequent days in the meal plan, and sees the correct number of meals displayed. |  | |
| **ALTERNATE COURSES:** | If the system fails to display the correct number of meals for a day, the system will revert to the default error and display an incorrect number of meals instead (i.e. if the planner chose two meals for each day in the plan, the system may display three meals instead). | | |
|  | If the system fails to process the planner inputs, the system will ask the planner to create another meal plan. | | |
| **CONCLUSION:** | The planner can see the contents of all meals, with the number of meals being equal to the number of meals they specified in the prompts. | | |
| **POST-CONDITION:** | The planner can freely access this plan and view meal contents, with the plan designed to be in-line with the planner’s answers. | | |
| **BUSINESS RULES** | N/A | | |
| **IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS** | * Modify the files without changing much of the functionalities of the program. * Do not introduce new bugs. | | |
| **ASSUMPTIONS:** | Assume there are no underlying bugs related to the bug at hand, but be prepared to fix them if there are. | | |
| **OPEN ISSUES:** | The bug for the incorrect number of meals for a plan still need to be fixed. | | |

**DAILY MEAL PLANNER**

**Author (s): \_\_\_\_\_ Mrinal Raj Lakkimsetty Date: \_\_\_\_10/07/24\_\_\_\_\_\_\_**

**Version: \_\_\_\_\_\_\_\_\_\_\_**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE NAME:** | Switch Out Food in Meal Plan | | **USE CASE TYPE** |
| **USE CASE ID:** | 5 | | Business Requirements: **🞏** |
| **PRIORITY:** | High | | System Analysis: 🗹 |
| **SOURCE:** | Team brainstorming, user feedback | |  |
| **PRIMARY BUSINESS ACTOR** | Meal Planner | | |
| **PRIMARY SYSTEM ACTOR** | Meal Planner System | | |
| **OTHER PARTICIPATING ACTORS:** | * Database System | | |
| **OTHER INTERESTED STAKEHOLDERS:** | * Dieticians, Nutrition Experts | | |
| **DESCRIPTION:** | This use case enables the user to replace a specified food item in an already created meal plan with another food item from the database or suggested list, modifying the nutritional values and calorie count appropriately. | | |
| **PRE-CONDITION:** | User has a generated meal plan. | | |
| **TRIGGER:** | User selects the option to modify the meal plan. | | |
| **TYPICAL COURSE** | Actor Action | System Response | |
| **OF EVENTS:** | **Step 1**: Planner clicks on the “Switch Food” button next to a food item in the meal plan. |  | |
|  |  | **Step 2:** System displays a list of alternative food items with similar nutritional value. | |
|  | **Step 3:** Planner selects an alternative food item from the list. |  | |
|  |  | **Step 4:** System updates the meal plan with the new food item, recalculates nutritional information, and displays the updated meal plan. | |
|  |  |  | |
|  |  |  | |
|  |  |  | |
|  |  |  | |
| **ALTERNATE COURSES:** | Planner decides not to change the food item and closes the selection window. | | |
|  | If the system fails to procure an alternative food, the system will give an error message and prompt the planner to choose again. | | |
|  |  | | |
| **CONCLUSION:** | The meal plan is updated successfully with the user's desired changes. | | |
| **POST-CONDITION:** | The meal plan reflects the new food item, and all related data (calories, nutrients) are updated. | | |
| **BUSINESS RULES** | * Only foods from the same category can be swapped to maintain nutritional balance. | | |
| **IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS** | * Integration with the food database API to fetch real-time data and updates; responsive design to handle UI changes dynamically. | | |
| **ASSUMPTIONS:** | * Food items are readily available in the database; the system can handle multiple requests simultaneously. | | |
| **OPEN ISSUES:** | Determining which food items can be considered equivalent substitutes nutritionally. | | |

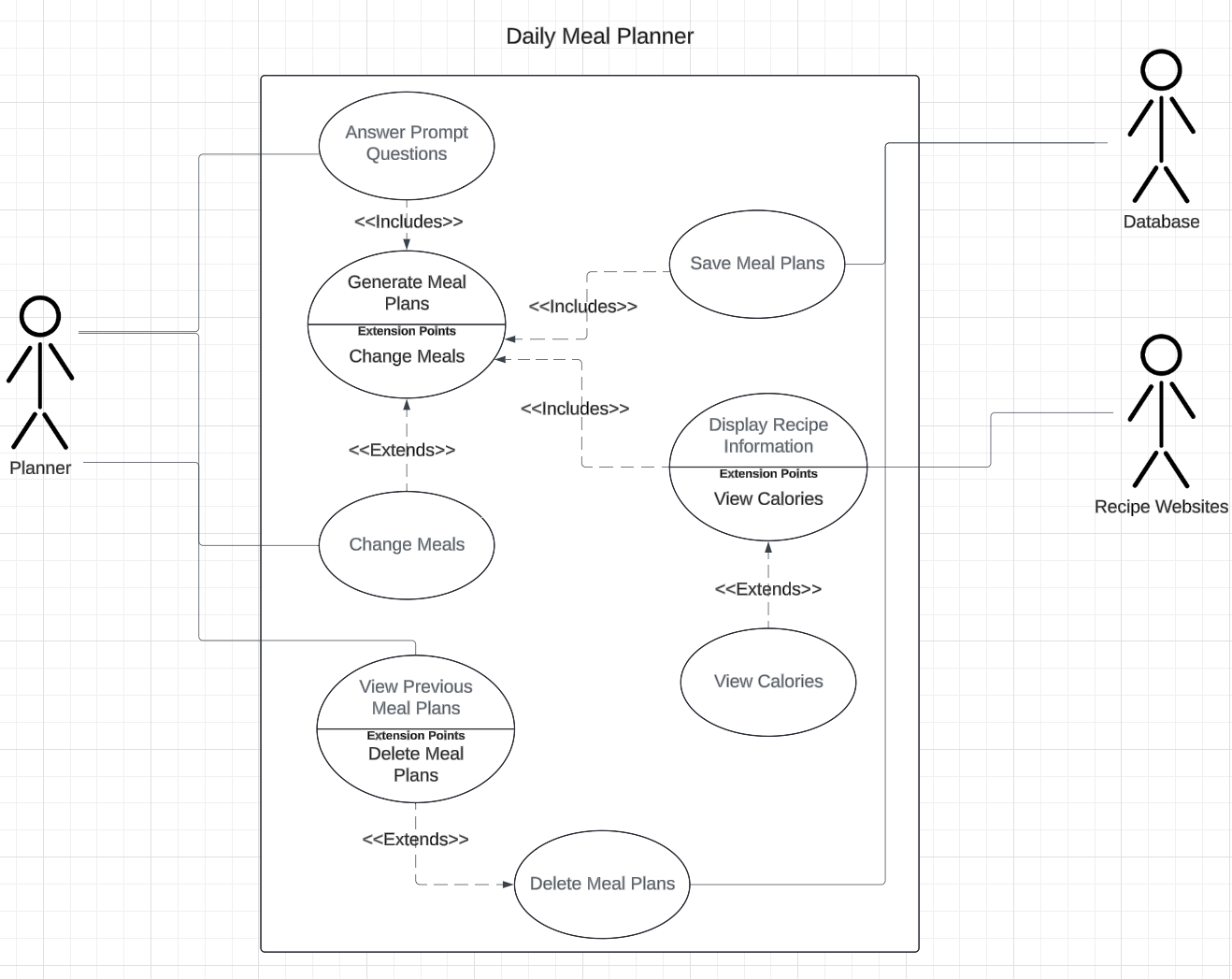
**Daily Meal Planner**

**Author (s): Willy Date: 10/7/2024**

**Version: \_\_\_\_\_\_\_\_\_\_\_**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE NAME:** | View past meal plans in a history tab | | **USE CASE TYPE** |
| **USE CASE ID:** | 6 | | Business Requirements: **🞏** |
| **PRIORITY:** | Not high | | System Analysis: 🗹 |
| **SOURCE:** | Team Discussion | |  |
| **PRIMARY BUSINESS ACTOR** | Planner | | |
| **PRIMARY SYSTEM ACTOR** | Database | | |
| **OTHER PARTICIPATING ACTORS:** | * Meal Planner Program | | |
| **OTHER INTERESTED STAKEHOLDERS:** | * N/A | | |
| **DESCRIPTION:** | The planner will automatically save generated meal plans to a cloud database. Then, the user can click on a history tab to view past meal plans, and the meal planer will be able to display past plans from the database. | | |
| **PRE-CONDITION:** | The planner has already generated at least one meal plan.  The database has the ability to automatically save meal plans. | | |
| **TRIGGER:** | The user navigates to the history tab on the webpage | | |
| **TYPICAL COURSE** | Actor Action | System Response | |
| **OF EVENTS:** | **Step 1**: The user navigates to the history tab. |  | |
|  |  | **Step 2**: The meal planner will fetch the database data. | |
|  |  | **Step 3**: The system will display the fetched data on the history tab. | |
|  | **Step 4**: The user selects a meal plan to view its details. |  | |
|  |  |  | |
|  |  |  | |
|  |  |  | |
|  |  |  | |
| **ALTERNATE COURSES:** | If the history tab fails to display recent meal plans, then the recent meal plans will not show up on the history page. | | |
|  | If there is a data fetching error, then an error message will be displayed stating there was a fetching error on the page. | | |
|  | If there is a database error, then (if possible) an error message stating there is a database error will be displayed. Otherwise, no error message will be displayed. | | |
| **CONCLUSION:** | The planner can see the contents of a specific meal plan | | |
| **POST-CONDITION:** | The planner has access to all previously generated meal plans. The planner can click on each plan to see specific details such as dates and food for each day. | | |
| **BUSINESS RULES** | * Planners cannot directly access the database | | |
| **IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS** | * Use of a cloud database to store data. * History data will be displayed in real time | | |
| **ASSUMPTIONS:** | * Cloud storage will be free and with no-cost | | |
| **OPEN ISSUES:** | N/A | | |

**Updated Use Case Diagram:**



Update Notes:

* Added an extended “View Calories” function to “Display Recipe Information”
* Added a relationship between the Planner and “Change Meals” function
* Changed “Delete Meal Plans” to become an extended function for “View Previous Meal Plans”