

Software Requirement Specification

| | |
|----------------------|---|
| Group members | Zachary James Benjamin Guadron Chris K. Appiah Jason Arias Benjamin Barnett Michael Carrington |
|----------------------|---|

1. Project Description

Our project is a complete fitness solution that can handle all your fitness needs. Main features are calorie tracking, Exercise tracking, and an AI assistant that provides advice based on your individual needs. The home page of the app links to the various features. It will run out of your web browser and store your information locally on your machine.

2. Functional Requirements

| | |
|-------------|--|
| FR01 | The software must enable the user to make accounts |
| FR02 | The software must enable the user to track calories |
| FR03 | The software must enable the user to track exercises |
| FR04 | The software must enable the user to get advice from an AI chat bot |
| FR05 | The software must enable the user to count steps using location |
| FR06 | The software must enable the user to create diet plans |
| FR07 | The software must enable the user to create workout plans |
| FR08 | The software must enable the user to get progress reports |
| FR09 | The software must enable the user to log water intake |
| FR10 | The software must enable the user to track their sleep patterns |
| FR11 | The software must enable the user to set daily reminders for water, meals, or sleep |
| FR12 | The software must enable the user to track their nutrient intake (vitamins/minerals) |
| FR13 | The software must enable the user to change language or regional settings |

3. Non-Functional Requirements

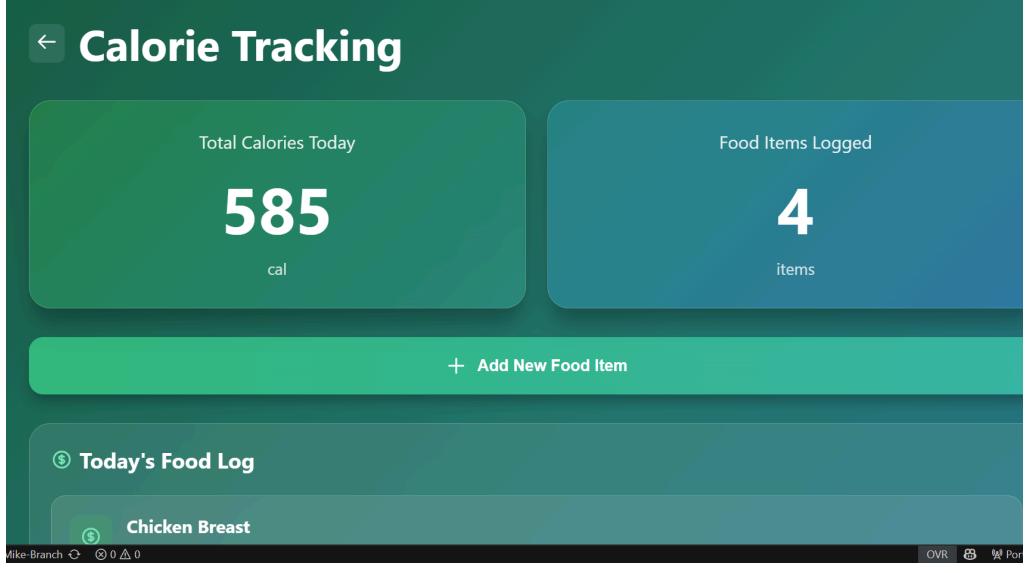
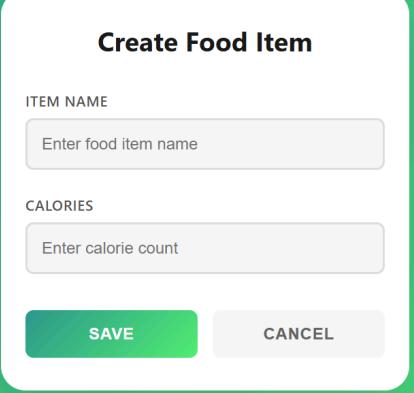
| | |
|--------------|---|
| NFR01 | New users must be able to complete the account registration and log their first workout within 5 minutes of starting. |
| NFR02 | The software must work on google chrome 141.0.7390.56 or later |
| NFR03 | The software must pass the siteimprove accessibility checker |
| NFR04 | The software must run locally on the users device except for AI |
| NFR05 | ≥ 99.9% monthly service availability for the ai. Monitor with status checks. |

4. Use Case Specification

<< Select three functional requirements and describe them in detail using use cases.>>

| | |
|---------------------------|--|
| UC01 Name: | Calorie Tracking |
| Description: | The user can add, edit, and remove food items from a list of the days total calories |
| Actor: | user |
| Entry condition: | The user selects Calorie Tracking from the home page |
| Basic path: | <ol style="list-style-type: none"> 1. The user is presented the calorie tracking page with the following options [PRO01] <ol style="list-style-type: none"> a. New Food Item b. Edit Food Item c. Delete Food Item d. Back Arrow 2. The user selects New Food Item [A01] [A02] [A04] 3. The user is presented the Food Item creation page with the following [PRO02] <ol style="list-style-type: none"> a. Item Name (editable) b. Calories (editable) c. Options <ul style="list-style-type: none"> i. Save ii. Cancel [A03] 4. The user fills out name and calories and selects Save 5. System verifies valid input [BR01] [BR02] [E01] 6. The user returns to the calorie tracking page 7. Use case concluded |
| Alternative paths: | <p>[A01] The user edits a food item</p> <ol style="list-style-type: none"> 1. The user selects Edit Food Item 2. The user is presented the food item creation page with the values populated [PRO02] <ol style="list-style-type: none"> a. Item Name (editable) b. Calories (editable) c. Options <ul style="list-style-type: none"> i. Save ii. Cancel [A03] |

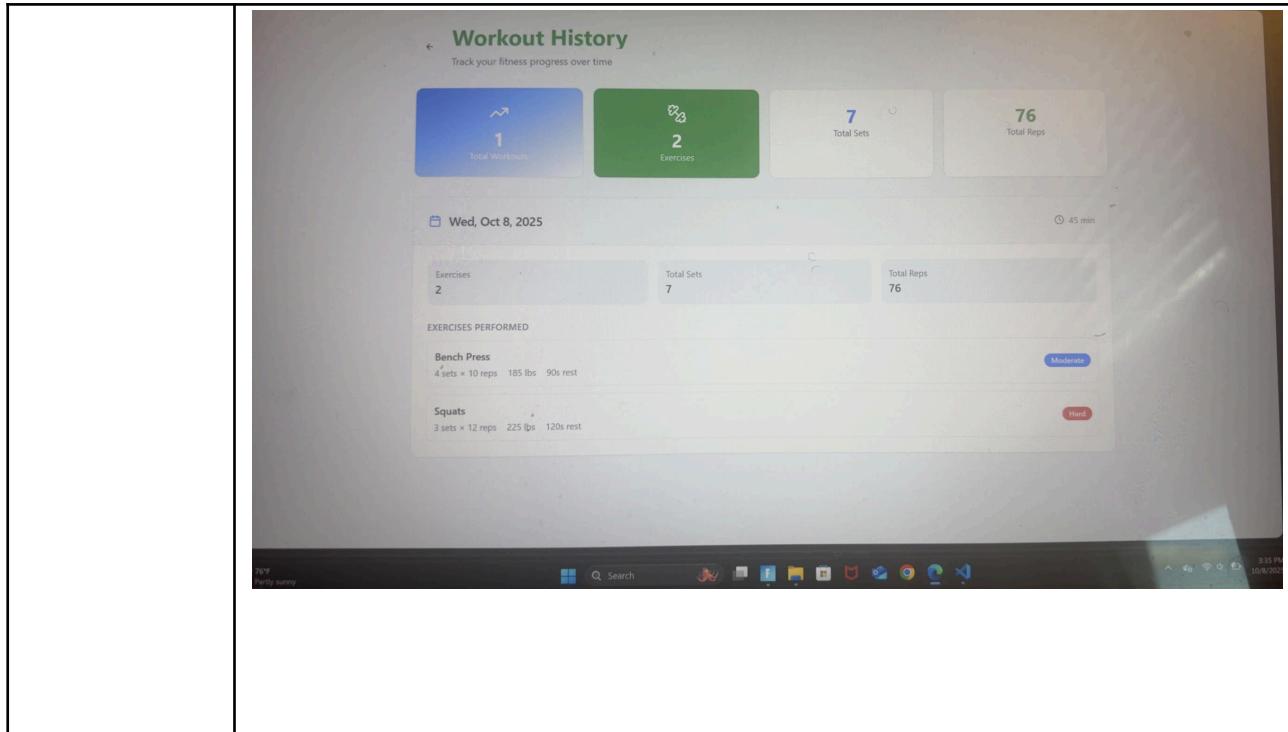
| | <p>3. The user updates name or calories and selects Save</p> <p>4. System verifies valid input [BR01] [BR02] [E01]</p> <p>5. The user returns to the calorie tracking page</p> <p>[A02] The user delete a food item</p> <ol style="list-style-type: none"> 1. The user selects Delete 2. The user is presented with the updated calorie tracking page missing the deleted food item <p>[A03] The user cancels modifying/adding a food item</p> <ol style="list-style-type: none"> 1. The user selects Cancel 2. The user returns to the calorie tracking page with nothing changed <p>[A03] The user returns to the home page</p> <ol style="list-style-type: none"> 1. The user selects the back arrow 2. The user returns to home page where they can select different features to use | | | | | | | | | | | | |
|-------------------------|---|---------|------|--------|------|------|--------|----|---|----------|--------------|---------|---|
| Exception paths: | <p>[E01] Invalid Input</p> <ol style="list-style-type: none"> 1. The user inputs invalid item calories/name 2. The user is prompted to reenter their item calories/name depending on what was invalid | | | | | | | | | | | | |
| Business Rules: | <p>[BR01] All fields are required</p> <p>[BR02] All calories must be positive</p> | | | | | | | | | | | | |
| Data description | <table border="1"> <thead> <tr> <th>Name</th><th>Type</th><th>Length</th><th>Mask</th></tr> </thead> <tbody> <tr> <td>Name</td><td>String</td><td>50</td><td>-</td></tr> <tr> <td>Calories</td><td>Unsigned Int</td><td>4 bytes</td><td>-</td></tr> </tbody> </table> | Name | Type | Length | Mask | Name | String | 50 | - | Calories | Unsigned Int | 4 bytes | - |
| Name | Type | Length | Mask | | | | | | | | | | |
| Name | String | 50 | - | | | | | | | | | | |
| Calories | Unsigned Int | 4 bytes | - | | | | | | | | | | |

| | |
|------------|--|
| Prototype: | <p>[PRO01]</p>  <p>Calorie Tracking</p> <p>Total Calories Today 585 cal</p> <p>Food Items Logged 4 items</p> <p>+ Add New Food Item</p> <p>Today's Food Log</p> <p>Chicken Breast</p> <p>Mike-Branch ○ ⊖ 0 △ 0 OVR 🔍 Port</p> <p>[PRO02]</p>  <p>Create Food Item</p> <p>ITEM NAME Enter food item name</p> <p>CALORIES Enter calorie count</p> <p>SAVE CANCEL</p> |
|------------|--|

| | |
|---------------------------|--|
| UC02 Name: | Exercise Tracking |
| Description: | The user can log their current workout split this includes their exercises, number of reps/sets, weight, rest time, and difficulty of each set. |
| Actor: | User |
| Entry condition: | The user will select the Exercise Tracking button from the Home page of the application. |
| Basic path: | <ol style="list-style-type: none"> 1. The user is presented with the exercise tracking page with the following options [PRO01] <ol style="list-style-type: none"> a. Add New Exercise b. Edit Exercise c. Delete Exercise d. Back Arrow 2. The user selects Add New Exercise [A01] [A02] [A03] 3. The user is presented with the New Exercise Entry page containing the following fields [PRO02] <ol style="list-style-type: none"> a. Exercise Name (editable) b. Number of Sets (editable) c. Reps per Set (editable) d. Weight Used (optional, editable) e. Rest Time (editable, in seconds) f. Difficulty (dropdown: Easy, Moderate, Hard) g. Options h. Save i. Cancel [A04] 4. The user fills in all fields and selects Save 5. The system validates inputs [BR01] [BR02] [E01] 6. The new exercise is added to the user's daily log 7. The user returns to the Exercise Tracking page 8. Use case concluded |
| Alternative paths: | <p>[A01] The user edits an existing exercise</p> <ol style="list-style-type: none"> 1. The user selects Edit Exercise 2. The user is shown a list of previously logged exercises 3. The user selects one to edit 4. The Exercise Entry Page opens with pre-filled values 5. The user modifies the details (e.g., sets, reps, weight) |

| | |
|-------------------------|---|
| | <p>6. The user selects Save</p> <p>7. System validates updated information [BR01] [BR02]</p> <p>8. The user returns to the Exercise Tracking page with the updated entry</p> <p>[A02] The user deletes an exercise</p> <ol style="list-style-type: none"> 1. The user selects Delete Exercise 2. A list of current exercises is displayed 3. The user selects one to delete 4. The system prompts a confirmation message (Y/N) 5. The user confirms deletion 6. The system removes the selected exercise from the log 7. The user is returned to the updated Exercise Tracking page <p>[A03] The user returns to the home page</p> <ol style="list-style-type: none"> 1. The user selects the Back Arrow 2. The user returns to the Home Page <p>[A04] The user cancels adding or editing an exercise</p> <ol style="list-style-type: none"> 1. The user selects Cancel 2. The system discards all changes and returns to the Exercise Tracking page |
| Exception paths: | <p>[E01]</p> <ol style="list-style-type: none"> 1. The user leaves a required field empty or enters invalid data 2. The system highlights the invalid field(s) and displays an error message <p>[E02]</p> <ol style="list-style-type: none"> 1. The system fails to save exercise data due to a network or storage error 2. The user is prompted to retry or save offline until the connection is restored |
| Business Rules: | <p>[BR01] All required fields (Exercise Name, Sets, Reps) must be filled</p> <p>[BR02] Sets, Reps, and Rest Time must be positive numbers</p> <p>[BR03] Only verified users can track workouts or access premium exercise templates</p> |

| Data description | <table border="1"> <thead> <tr> <th>Name</th><th>Type</th><th>Length</th><th>Mask</th></tr> </thead> <tbody> <tr> <td>Sets</td><td>String</td><td>50</td><td>-</td></tr> <tr> <td>Reps</td><td>Int</td><td>4 bytes</td><td>-</td></tr> <tr> <td>Weight</td><td>Float</td><td>4 bytes</td><td>-</td></tr> <tr> <td>Rest Time</td><td>Int</td><td>4 bytes</td><td>-</td></tr> <tr> <td>Difficulty</td><td>Enum</td><td>1</td><td>Easy/Moderate/Hard</td></tr> </tbody> </table> | Name | Type | Length | Mask | Sets | String | 50 | - | Reps | Int | 4 bytes | - | Weight | Float | 4 bytes | - | Rest Time | Int | 4 bytes | - | Difficulty | Enum | 1 | Easy/Moderate/Hard |
|-------------------------|---|---------|--------------------|--------|------|------|--------|----|---|------|-----|---------|---|--------|-------|---------|---|-----------|-----|---------|---|------------|------|---|--------------------|
| Name | Type | Length | Mask | | | | | | | | | | | | | | | | | | | | | | |
| Sets | String | 50 | - | | | | | | | | | | | | | | | | | | | | | | |
| Reps | Int | 4 bytes | - | | | | | | | | | | | | | | | | | | | | | | |
| Weight | Float | 4 bytes | - | | | | | | | | | | | | | | | | | | | | | | |
| Rest Time | Int | 4 bytes | - | | | | | | | | | | | | | | | | | | | | | | |
| Difficulty | Enum | 1 | Easy/Moderate/Hard | | | | | | | | | | | | | | | | | | | | | | |
| Prototype: | <p>[PRO01] :</p> <p>The screenshot shows a mobile application interface titled "Exercise Tracking". At the top, it says "4 exercises logged" and has a button "+ Add Exercise". Below are four cards, each representing an exercise:</p> <ul style="list-style-type: none"> Barbell Squat: Hard. Sets x Reps: 5 x 5, Weight: 225 kg, Rest Time: 180s. Barbell Bench Press: Moderate. Sets x Reps: 4 x 8, Weight: 185 kg, Rest Time: 120s. Conventional Deadlift: Hard. Sets x Reps: 3 x 5, Weight: 315 kg, Rest Time: 240s. Overhead Press: Easy. Sets x Reps: 3 x 10, Weight: 95 kg, Rest Time: 90s. | | | | | | | | | | | | | | | | | | | | | | | | |



| | |
|---------------------------|---|
| UC03 Name: | AI Gym Assistant |
| Description: | An integrated AI assistant that recommends personalized workouts, diet plans, and wellness tips, while adapting to user goals and activity data in real time. |
| Actor: | User |
| Entry condition: | User will press Gym Partner AI button from home section |
| Basic path: | <p>1. The user selects the Gym Partner AI button from the home section.</p> <p>2. The system presents the AI interface with the following sequence:</p> <ul style="list-style-type: none"> ● User Input → AI Processing → AI Response → User Views Plan → User Logs Action → Database Saves Progress → AI Learns for Next Time. <p>3. The system confirms successful update and displays feedback message.</p> <p>3. Use case concluded.</p> |
| Alternative paths: | <p>[A01] Quick Plan (minimal input):</p> <ol style="list-style-type: none"> 1. User taps Quick Start. 2. System uses default profile + goal to generate a 7-day starter plan. <p>[A02] Regenerate/Refine:</p> <ol style="list-style-type: none"> 3. User requests changes (e.g., “Swap squats for leg press,” “No seafood”). 4. System regenerates only affected parts and updates plan version. <p>[A03] Q&A Mode:</p> |

| | <ol style="list-style-type: none"> 1. User asks a question ("How many calories should I eat on rest days?"). 2. System responds with a short explanation and the exact number based on targets, with a link to Apply to Plan. | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------------------------|---|--------------|-----------|--------|------|--|------------|--------|--------------|-----------|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|
| Exception paths: | <ol style="list-style-type: none"> 4. [E01] Missing critical data: System asks for height/weight/age or equipment list before generating. 5. [E02] AI timeout/error: System shows error and offers Retry or Fallback template plan. 6. [E03] Network offline: System queues the request and notifies when generated; offers viewing last saved plan. 7. [E04] Safety flag: If user mentions injury/medical risk, system halts and shows safety disclaimer with Seek professional advice. | | | | | | | | | | | | | | | | | | | | | | | | | |
| Business Rules: | <p>[BR01] .Plans must respect allergies/restrictions and never suggest contraindicated exercises.</p> <p>[BR02] .Calorie targets within $TDEE \pm 25\%$; protein 1.6–2.2 g/kg unless user overrides.</p> <p>[BR03] .No medical advice; show disclaimer on first use and in safety cases.</p> | | | | | | | | | | | | | | | | | | | | | | | | | |
| Data description | <table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Length</th> <th>Mask</th> <th></th> </tr> </thead> <tbody> <tr> <td>User Input</td> <td>String</td> <td>Max 25k char</td> <td>Free text</td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table> | Name | Type | Length | Mask | | User Input | String | Max 25k char | Free text | | | | | | | | | | | | | | | | |
| Name | Type | Length | Mask | | | | | | | | | | | | | | | | | | | | | | | |
| User Input | String | Max 25k char | Free text | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Prototype: | [PRO01] | | | | | | | | | | | | | | | | | | | | | | | | | |

The screenshot shows a mobile application interface. At the top, a message says "Perfect! How old are you?" with a blue circular button containing the number "20". Below this, another message asks "Almost done! How would you describe your activity level?" with four options: "Sedentary", "Lightly Active", "Moderately Active", and "Very Active". A blue button labeled "Very Active" is highlighted. The next message says "Your personalized plan is ready!" and includes a "Daily Targets" section with placeholder data: "Calories: NaN kcal", "Protein: NaN", "Carbs: NaN", and "Fats: NaN". It also states "I've created a custom meal and workout plan for you!". Below this are three buttons: "Show Meal Plan", "Show Workout Plan", and "Ask a Question". At the bottom, there is a text input field with "Type a message..." placeholder text and a blue send button with a white arrow icon. A small note at the bottom right says "Powered by AI".

[PRO02]

