

Functional Requirements:

1.User Account Management

- USR_01: User can register with an email,password and other personal information
- USR_02:User can login with email and password
- USR_03:Users can update their profile and account information(password,weight,height,etc.)
- USR_04:Users can delete their account.
- USR_05:User can upload a profile picture

2.User Nutrition Tracking

- NTR_01:Users can log their daily calorie intake and set a daily calorie goal.
- NTR_02:Users can track their macronutrient intake(protein,carbs,fats)
- NTR_03:Users can log their daily hydration intake.
- NTR_04:Users can log their sleep, and get suggested recommend hours.

3.User Workout Management

- WRK_01:Users can browse and filter pre-made workouts.
- WRK_02:Users can create,save and edit custom workouts for themselves.
- WRK_03:Users can log and track workout sessions.
- WRK_04:Users can choose if they want to be notified for workout sessions.
- WRK_05:Users can favorite workouts for quick access.

4.Badge and Gamification System

- BDG_01:User can earn badges on predefined criteria(e.g., workout streaks,achieving goals)

5.User Progress Tracking

- PRG_01:Users can track workouts streaks.
- PRG_02:Users can view progress charts(e.g.,weight tracking,PRs).
- PGR_03:Users can update personal goals(e.g., target weight) and track progress towards them.
- PRG_04:Users can export their progress data(workout history,PRs) as a CSV file.

6.Notification System

- NOS_01:Users can enable or disable workout reminders and motivational notifications.
- NOS_02:Users can receive notifications about new pre-made workouts.
- NOS_03:Users can receive weekly summary about their progress.

7.User Experience Level and Goals

- UXP_01:Users will be introduced by a questionnaire when creating their account gathering essential information about them and their fitness health goals.
- UXP_02:Users can choose their preferred units(metric,imperial).
- UXP_03:Users can select the app's theme(light/dark mode).

8.Admin Management

- ADM_01:Super Admins can add new admins.
- ADM_02:Super admins can change permissions of admin(e.g.,creating workouts,sending notifications etc).
- ADM_03:Admins can send notifications to users

- ADM_04:Admins can create new workouts with a name,level,times per week,exercises etc.
- ADM_05:Admins can update or delete existing workouts
- ADM_06:Admins can view a list of all users.
- ADM_07:Admins can delete user accounts.
- ADM_08:Super admins can promote users to admin.
- ADM_09:Admins can add FAQs.
- ADM_10:Super admins has access do to everything admin does

10.Settings Configuration(Admin)

- SET_ADM_01:Admins can set up automated notifications.

11.Search and Filter(Admin)

- SRCH_ADM_01:Admins can search for users by email or name and filter them by experience level, goals etc.

12.Monetization

- MON_01:User can pay a subscription to get access to extra features.

Non Functional Requirements

1.Performance:

- The app must load pages within 2-3 seconds under normal load.
- Performing Lazy Loading to ensure smooth experience.

2.Security:

- Data access is only possible with security strategies(JWT,OAuth).
- User passwords must be stored as hashed values using a secure hashing algorithm (e.g., bcrypt).
- The app must use HTTP for all data transmissions.
- The app must implement role-based access control to restrict admin actions based on permissions.

3.Scalability:

- The app should scale horizontally to accommodate an increasing number of users.
- The database should support efficient querying for large datasets (e.g., nutrition logs, workout history).

4 Compatibility:

- Support Chrome, Firefox, Safari, and mobile browsers.

5.Maintainability:

- The codebase should follow a modular architecture for easy updates and maintenance.
- The app should include detailed documentation for developers to understand the ERD and system design.
- The app should use a version control system to track changes.

6.Data integrity.

- The app should enforce data validation (e.g., email format, non-negative values for weight, height, calories).
- The app should ensure referential integrity between entities.

7 Error Handling:

- Display user-friendly error messages with resolution steps.

8.Localization:

- Support date/time formats and units (metric/imperial).

9.Third-Party Integration:

- Validate external API responses to prevent data corruption.