**COMP 490/L, 491/L. Senior Design Project I and Lab (2/1)**

**Fall 2024 - Spring 2025**

**Software Requirements Specification (SRS)**

**For**

**FitGenius**

Team Number: 3

Team Members:

| Role | Name |
| --- | --- |
| Team Lead | N/A |
| Team Member | Sergio Garcia |
| Team Member | Isai Flores |
| Team Member | Emmanuel Diaz |
| Team Member | Arpa Hakopian |
| Team Member | Mirna Farahat |

**Contents**

[**1.0 Introduction 3**](#_cdi2dyylovxj)

[1.1 Scope 3](#_byvr3v1hkn3g)

[1.2 System Architecture Diagram 5](#_efg9t7acd9e4)

[1.3 Definitions, Acronyms, and Abbreviations 7](#_lsm4n1irihxm)

[1.4 Stakeholders 7](#_tkdt69fy6ju7)

[1.5 Assumptions and Dependencies 8](#_dpvwpcdkpbp3)

[**2.0 Functional Requirements 9**](#_837llf4vnjwp)

[2.1 Use Case Descriptions 9](#_4bkbrl2p34wg)

[2.2 Functional Requirement 1 9](#_oumg4r3d4iip)

[2.3 Functional Requirement 2 12](#_xygr8wvs3984)

[**3.0 Non-Functional Requirements 13**](#_ydsu2oi1bn7s)

[3.1 Performance Requirements 13](#_32dz3jgm53ib)

[3.2 Security Requirements 14](#_hdw1z3vs6xms)

[3.3 Usability Requirements 14](#_bu10se9rismr)

[**4.0 Review and Approval 15**](#_jjf3kgn7nen5)

[4.1 Reviewers 15](#_v4nmxff26fnj)

# 1.0 Introduction

FitGenius is an iOS fitness and nutrition application designed to help users achieve their health and wellness goals through personalized plans, activity tracking, and community engagement. Leveraging artificial intelligence (AI), the app provides tailored workout routines, meal suggestions, and injury prevention tips, ensuring users receive guidance that adapts to their unique needs and progress. By integrating gamification, FitGenius creates an interactive and motivational platform that makes fitness enjoyable and sustainable.

The app addresses the challenges of maintaining a healthy lifestyle in today’s fast-paced world. Unlike generic fitness solutions, FitGenius offers personalized recommendations based on user data, preferences, and goals. Features like fitness tracking, AI-driven plans, and detailed progress analytics empower users to take control of their health. Additionally, gamification elements such as rewards, badges, and avatar customization keep users motivated.

This Software Requirements Specification (SRS) document outlines the functional and nonfunctional requirements for the development of FitGenius. It serves as a guide for developers, designers, and stakeholders to ensure the system meets user needs and business objectives. By combining advanced technology with user-centric design, FitGenius aims to revolutionize the way individuals approach fitness and nutrition, making it more accessible, engaging, and effective.

## 1.1 Scope

The scope of **FitGenius** encompasses the development of an iOS application that provides users with a comprehensive platform for fitness and nutrition management, personalization, gamification, and social interaction. The system will include the following key features:

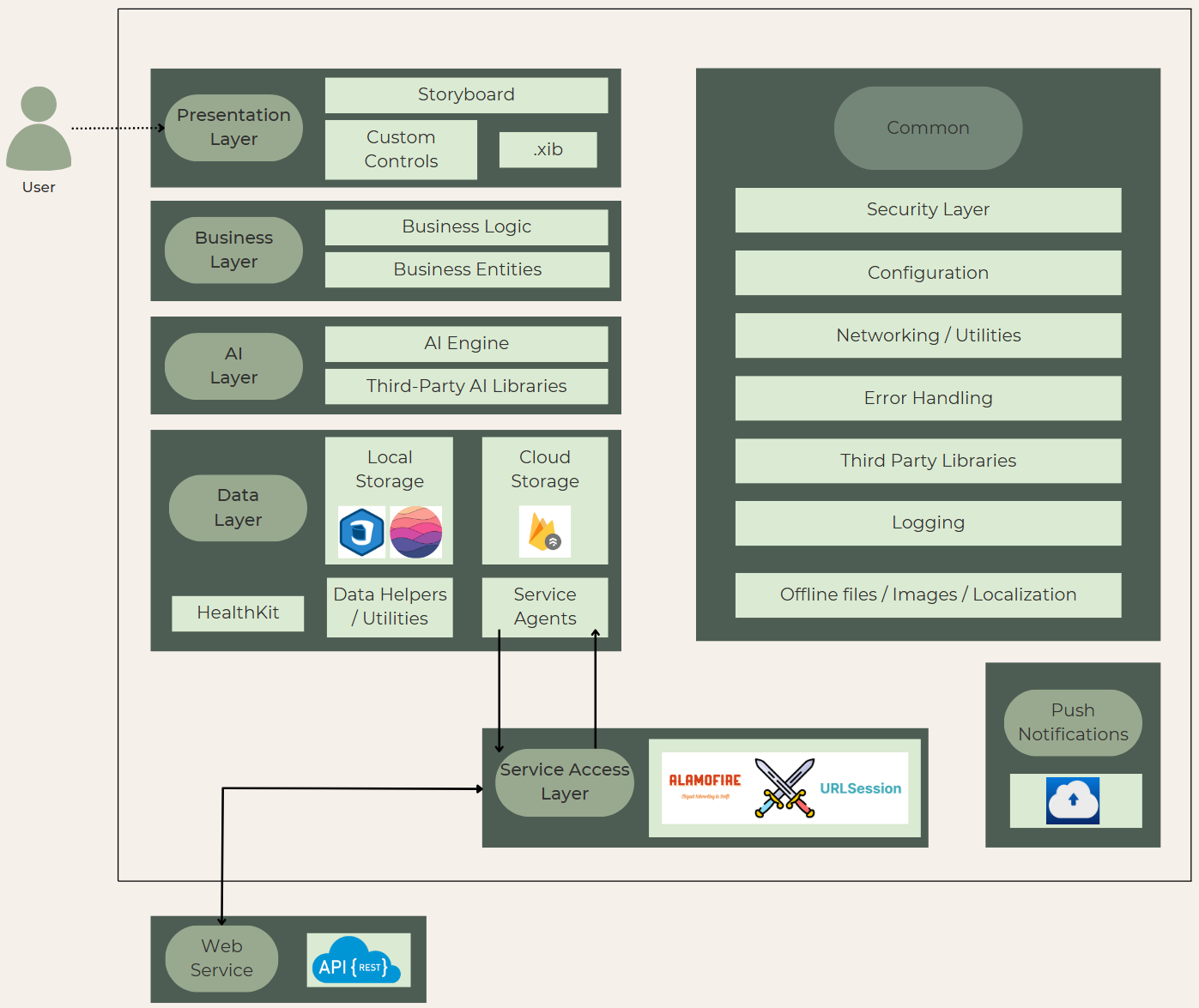
1. **Fitness Tracking**
   * Users can log workouts, steps, and calorie expenditure.
   * The app will track dietary habits and provide insights into nutritional intake.
2. **Personalization**
   * AI-driven customized workout and meal suggestions based on user data, preferences, and goals.
   * Recommendations for exercises, meal ideas, and healthy habits tailored to individual needs.
3. **Gamification**
   * Users can earn rewards and badges for achieving milestones.
   * Avatar customization allows users to visually reflect their fitness progress.
4. **Data Integration**
   * Detailed analytics on fitness progress, enabling users to monitor achievements and areas for improvement.

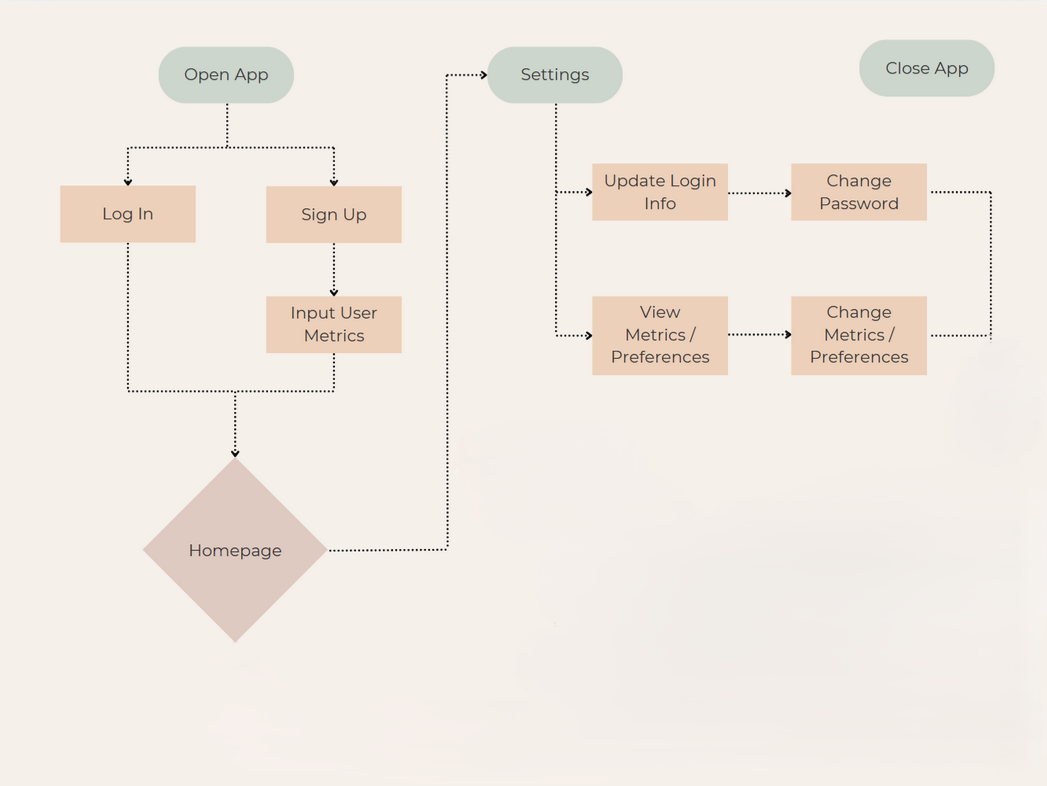
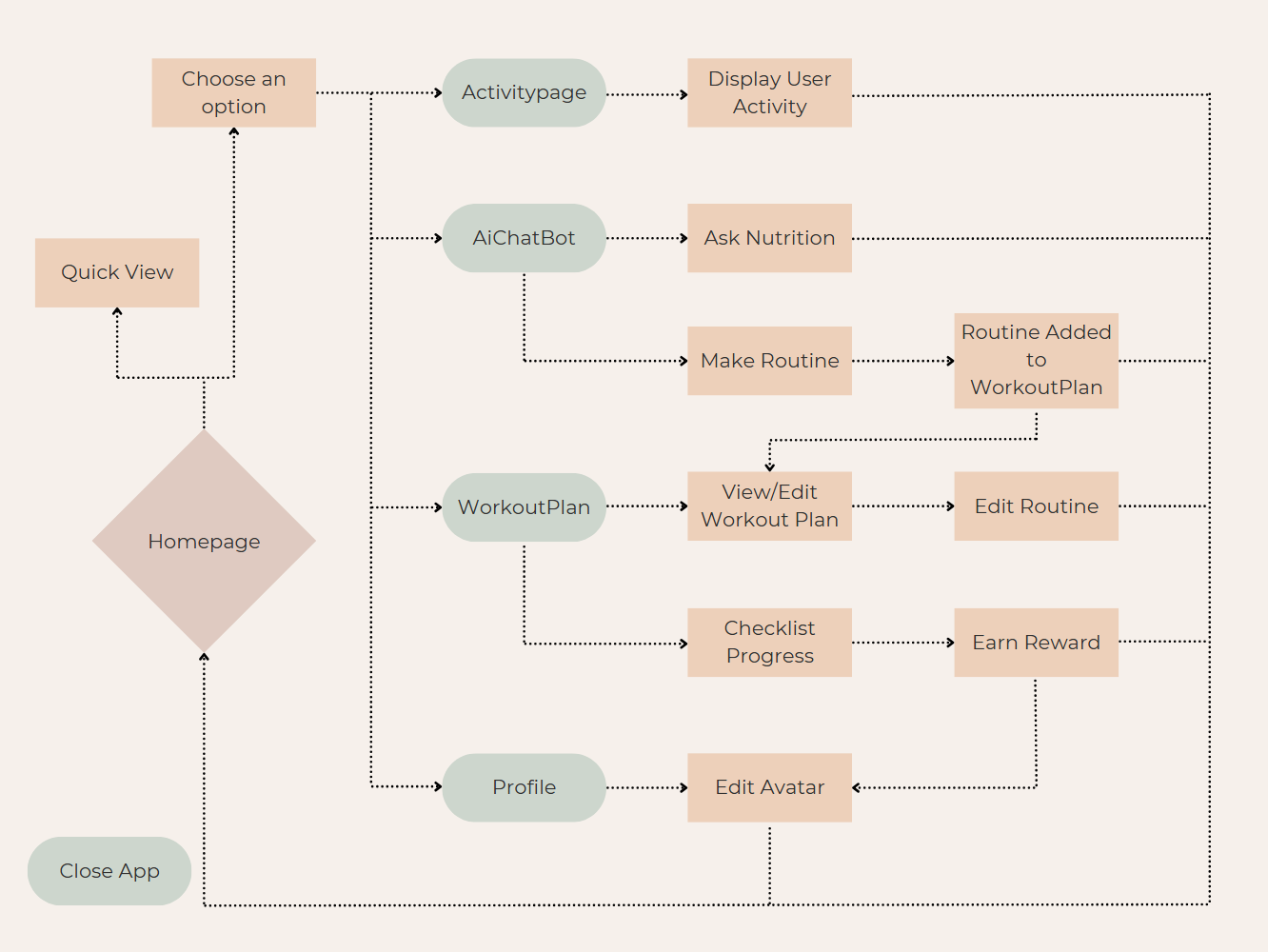
#### **Boundaries of the System**

* The app will be developed exclusively for iOS devices.
* It will not include medical diagnosis or treatment features.
* Direct integration with third-party fitness equipment (e.g., treadmills, bikes) is out of scope.
* The app will rely on user-provided data for personalization and will not include advanced medical or biometric analysis.

By focusing on these features and boundaries, **FitGenius** aims to deliver a user-friendly, engaging, and effective fitness and nutrition solution tailored to the needs of its target audience.

## 1.2 System Architecture Diagram



**Flow Chart**

## 1.3 Definitions, Acronyms, and Abbreviations

* **AI**: Artificial Intelligence
* **iOS**: Operating system for Apple mobile devices
* **Gamification**: The use of game-like elements (e.g., rewards, challenges) in non-game contexts to increase engagement
* **Avatar**: A customizable digital representation of the user within the app
* **SRS**: Software Requirements Specification

## 1.4 Stakeholders

## The success of **FitGenius** depends on the collaboration and contributions of various stakeholders. Below is a list of identified stakeholders, their roles, and their interests in the project:

Primary Stakeholders

1. Users
   * **Role**: End-users of the **FitGenius** application.
   * **Interests**:
     + Access to a user-friendly platform for tracking fitness and nutrition goals.
     + Personalized workout and meal suggestions tailored to their preferences and progress.
     + Gamification features (e.g., rewards, badges, avatar customization) to stay motivated.
2. Development Team
   * **Role**: The team responsible for designing, developing, and maintaining the **FitGenius** application.
   * **Interests**:
     + Delivering a high-quality, functional, and scalable application that meets user needs.
     + Learning and implementing new technologies to enhance the app’s features.
     + Ensuring the app is maintainable and adaptable for future updates.

## 1.5 Assumptions and Dependencies

**Assumptions:**

* It is assumed that all users have access to a stable internet connection for data synchronization and real-time features.
* Users will grant necessary permissions (e.g., access to the Health app data) during the onboarding process.
* The user data provided (such as health metrics and contact details) is accurate and up to date.
* The target audience primarily uses iOS devices that support at least iOS 17.6.

**Dependencies:**

* **Apple HealthKit Integration:** FitGenius relies on Apple's HealthKit to retrieve user health data, making it dependent on the Health app's API and proper permission management.
* **iOS Ecosystem:** The app's functionality is built exclusively for iOS; any changes to the iOS platform or updates (minimum required version: iOS 17.6) could impact the application’s performance or compatibility.
* **Backend Services:** Dependencies on cloud-based services for user authentication, data storage, and AI processing must be reliable and secure.
* **Third-Party Libraries:** The application uses third-party libraries for features like AI recommendations, which must be kept up to date to avoid security vulnerabilities.

# 2.0 Functional Requirements

This section describes the specific functionality that the software must provide. It details how the system will behave and respond to different inputs or scenarios. In addition to the detailed use cases provided below, the system will also incorporate several supporting features that ensure a seamless user experience.

## 2.1 Use Case Descriptions

In addition to the detailed descriptions provided for User Registration and Authentication, Profile Management, Tracking, AI Chat Box, and Build Avatar, each use case follows a similar structure:

* **Actors:** Identify primary and secondary actors (e.g., End Users, System, Third-Party Services).
* **Preconditions:** Conditions that must be met before the use case can be initiated (e.g., valid account creation, granted permissions).
* **Postconditions:** The state of the system after successful completion of the use case (e.g., updated user profile, synchronized health data).
* **Basic Flow:** Step-by-step actions taken by the user and system responses.
* **Alternative Flows:** Variations in the process, including error handling (e.g., invalid login credentials, missing data from the Health app).

## 2.2 Functional Requirement 1

**1. User Registration and Authentication**

The system must allow users to register and authenticate their accounts securely.

Inputs

* User-provided email and password during registration
* Login credentials (email and password)
* Data retrieved from the Health app
* Additional user-provided health information (if not available in the Health app)

Outputs

* Confirmation message upon successful registration
* Error messages for incorrect login attempts
* Retrieved health data displayed for user verification
* Prompts requesting missing health information

Behavior

* The app must allow users to register with a valid email.
* It must verify credentials upon login and display an error message for incorrect information.
* The app must retrieve user health data from the Health app if permission is granted.
* If required data is missing from the Health app, the system must prompt users to manually input the missing information.

**2. Profile Management**

Users must be able to manage their profile details, including contact information, health data, and goals.

Inputs

* User-provided new email
* User’s previous password for authentication
* Updated health information
* Updated personal goals

Outputs

* Confirmation message upon successful changes
* Error messages for incorrect authentication
* Updated user profile details

Behavior

* The app must allow users to change their email.
* Before changing contact information or password, users must verify their previous password.
* Users must be able to update collected health information at any time.
* Users must be able to update their fitness goals.

3. **Tracking**

Users must be able to track their workouts and activity history.

Inputs

* User-logged workouts
* User-logged meals (including nutritional values)
* Data retrieved from the Health app
* User-defined tracking preferences

Outputs

* Timeline of logged workouts
* Summary of Health app activity
* Configured tracking settings

Behavior

* The app must allow users to log workouts manually.
* The system must maintain a timeline of logged activity, including dates.
* Users must be able to view their tracked activity from the Health app.
* Users must be able to configure tracking preferences.

## 

## 

## 

## 2.3 Functional Requirement 2

**1. AI Chat Box**

The system must provide AI-powered workout and meal recommendations while ensuring user safety.

Inputs

* User queries for workout plans
* User queries for meal ideas
* User selections for AI-generated plans
* User modifications to suggested plans

Outputs

* Suggested workout plans
* Suggested meals
* Warnings for dangerous or unhealthy requests
* Updated workout schedules/logs

Behavior

* Users must be able to request specific workout plans from the AI.
* Users must be able to request meal ideas from the AI.
* The AI must intercept and block dangerous or unhealthy fitness/nutrition requests.
* Users must be given the option to adopt suggested workout plans.
* The system must allow users to transfer AI-recommended workout plans into their in-app schedule or log upon request.
* Users must be able to manually adjust AI-suggested plans.

**2. Build Avatar**

Users must be able to create and customize a personal avatar.

Inputs

* User selections for avatar customization
* User milestone achievements

Outputs

* Customized avatar
* Unlocked customization options highlighted in gold

Behavior

* The app must allow users to create a custom avatar.
* The system must guide users through a selection process for basic customization options.
* Users must be able to edit their avatar at any time.
* The app must unlock additional customization options as users reach milestones.
* When users unlock a new accessory in the creator, it is highlighted in gold to draw attention to the new customization option.

# 3.0 Non-Functional Requirements

Non-functional requirements define how the system performs rather than detailing specific behaviors. These requirements are critical for ensuring the application is efficient, secure, and user-friendly.

## 3.1 Performance Requirements

**Response Time:** The application must load any view or screen within 2 seconds on supported devices.

**Throughput:** The system should support simultaneous use by up to 10,000 concurrent users without significant performance degradation.

**Scalability:** The backend infrastructure must scale horizontally to accommodate a growing user base, ensuring consistent performance across varying load conditions.

**Resource Efficiency:** Optimize battery consumption and memory usage to ensure the app is lightweight and efficient, especially on older iOS devices that meet the minimum requirement.

## 

## 3.2 Security Requirements

**Data Encryption:** All user data, both in transit and at rest, must be encrypted using industry-standard protocols (e.g., TLS for data in transit, AES for data at rest).

**Authentication:** Utilize multi-factor authentication (MFA) where possible, including integration with Face ID/Touch ID for secure login.

**Access Control:** Implement role-based access control (RBAC) to ensure that only authorized users can access or modify sensitive data.

**Privacy Compliance:** Ensure that the app adheres to relevant data privacy regulations (e.g., GDPR, CCPA) by providing users with clear consent options and data usage policies.

**Vulnerability Management:** Regularly update the application and third-party libraries to patch known security vulnerabilities. Conduct periodic security audits and penetration tests.

## 3.3 Usability Requirements

**User Interface:** The app should have an intuitive and visually appealing interface that adheres to Apple’s Human Interface Guidelines, ensuring consistency and ease of use.

**Learnability:** The onboarding process should be straightforward, with tooltips and tutorials to assist new users in understanding key functionalities.

**Customization:** Allow users to personalize their experience through adjustable settings for display themes, and tracking preferences.

**Feedback Mechanism:** Provide clear and immediate feedback for user actions (e.g., registration confirmation, error messages) to facilitate a smooth interaction experience.

# 

# 4.0 Review and Approval

This section outlines the approval of the Software Requirements Specification (SRS) by internal stakeholders, including the development team and project testers. Approval confirms that the documented requirements align with the intended functionality of the FitGenius app. Feedback from team members and test users will be considered and incorporated into future updates of this document.

## 4.1 Reviewers

| **Name** | **Role** | **Approval Status** |
| --- | --- | --- |
| Mirna Farahat | Documentation Lead / Tester | Approved |
| Emanuel Diaz | Developer / Database Engineer | Approved |
| Arpa Hakopian | Health Integration Lead | Approved |
| Sergio Garcia | AI & Chatbot Engineer | Approved |
| Isai Flores | Avatar Creator | Approved |