

Functional Requirements

1. User Registration and Authentication

Description

Enables users to create accounts, log in, reset passwords, and manage profiles securely.

Actors

- User
- Authentication Service

Functional Requirements

1.1 Users must be able to sign up using an email/password or third-party authentication services (e.g., Google, Apple).

1.2 The system must enforce secure password policies, including minimum character length and special character inclusion.

1.3 Users must be able to reset passwords securely via email.

1.4 Two-factor authentication (2FA) must be available as an option.

1.5 All user credentials must be encrypted before storage.

1.6 The system must log authentication attempts and notify users of suspicious activity.

2. Dashboard

Description

Displays a summary of user progress, goals, and relevant statistics.

Actors

- User
- System

Functional Requirements

2.1 The dashboard must display daily goals, achievements, and activity history.

2.2 Users must be able to set and modify personal goals (e.g., completion time, progress tracking).

2.3 Users must be able to view AI-powered personalized suggestions for improvements.

2.4 The system must integrate with third-party apps for additional data visualization.

2.5 Dashboard updates must occur in real-time to reflect the latest user activity.

3. Exercise Content

Description

Provides categorized exercise routines with instructional multimedia content.

Actors

- User
- Exercise Content Manager

Functional Requirements

3.1 Exercises must be categorized by difficulty level.

3.2 Each exercise must include text instructions, timers, and video demonstrations.

3.3 Users must be able to create custom workout plans.

3.4 AI-powered recommendations must be available to users.

3.5 Users must be able to save favorite exercises for quick access.

4. Gamification

Description

Enhances user engagement through rewards and challenges.

Actors

- User
- Gamification Engine

Functional Requirements

4.1 Users must earn points and badges for completing exercises and milestones.

4.2 Leaderboards and daily challenges must be available to foster motivation.

4.3 Progress bars and streak tracking should be implemented for user retention.

4.4 New challenges must unlock as users progress to different levels.

4.5 Users must be able to share achievements on social media.

5. Progress Tracking

Description

Monitors user activity and provides analytical insights.

Actors

- User
- Progress Tracker

Functional Requirements

5.1 Users must be able to log workouts, calories burned, and exercise streaks.

5.2 Displays graphical progress reports for easy visualization.

5.3 AI-generated insights should be available for workout optimizations.

5.4 Users must be able to download progress reports for offline review.

6. Exercise Reminders

Description

Notifies users of scheduled exercises based on customizable settings.

Actors

- User
- Notification Service

Functional Requirements

6.1 Users must be able to set daily or weekly reminders.

6.2 Notifications must support geolocation-based reminders.

6.3 The system must send adaptive reminders based on inactivity.

6.4 Users must be able to customize notification preferences.

7. Social Features

Description

Enhances user interaction and community engagement.

Actors

- User
- Social Media Integration

Functional Requirements

7.1 Users must be able to share achievements and connect with friends.

7.2 Include privacy settings for sharing content.

7.3 Users should be able to compete with friends via leaderboards.

7.4 Users must be able to send and receive friend requests within the app.

8. AI-Powered Coaching

Description

Provides AI-generated coaching insights based on user performance.

Actors

- User
- AI Coach

Functional Requirements

8.1 Users must be able to receive AI-driven fitness insights.

8.2 AI-generated recommendations should be customizable.

8.3 The AI system should track workout trends and suggest improvements.

8.4 AI must support voice-based coaching for real-time feedback.

9. Music Integration

Description

Syncs workouts with music streaming services.

Actors

- User
- Music Service

Functional Requirements

- **9.1** Users must be able to connect their workouts with Spotify, Apple Music, and YouTube Music.
 - **9.2** Music tempo must adjust dynamically based on workout intensity.
 - **9.3** The system should allow users to create personalized workout playlists.
 - **9.4** The system should support offline music playback for workouts.
-

10. Data Security and Privacy

Description

Ensures secure data storage and user privacy.

Actors

- User
- System

Functional Requirements

- **10.1** All sensitive user data must be encrypted.
- **10.2** The system must comply with GDPR and other relevant data protection regulations.
- **10.3** Users must have control over their privacy settings and data sharing.
- **10.4** Secure authentication tokens must be implemented for user sessions.
- **10.5** Users must be able to delete their accounts and associated data permanently.

11. Performance and Reliability

Description

Ensures high performance and system scalability.

Functional Requirements

11.1 The system must load content within 2-3 seconds under normal conditions.

11.2 During peak times, 90% of requests must respond within 5 seconds.

11.3 The application must support at least 10,000 concurrent users without major performance degradation.

11.4 The system should auto-scale based on demand to ensure smooth performance.

12. Integration with External APIs

Description

Facilitates third-party application integrations.

Functional Requirements

12.1 The system must integrate with fitness tracking apps (e.g., Apple Health, Fitbit, Google Fit).

12.2 The application should support API-based connections to weather services for workout recommendations.

12.3 Data exchange between external APIs must be secure and optimized for efficiency.

13. Cross-Platform Support

Description

Ensures accessibility across multiple devices.

Functional Requirements

13.1 The application must be accessible via Android and iOS devices.

13.2 Users should be able to sync their workout data across multiple devices.

13.3 A web-based dashboard should be available for users who prefer desktop access.

14. Offline Mode

Description

Allows functionality without an active internet connection.

Functional Requirements

14.1 Users must be able to access their past workouts offline.

14.2 The system should allow offline viewing of downloaded exercise content.

14.3 AI-powered recommendations should work with locally stored data when offline.

14.4 Users must be able to log workouts offline, with automatic syncing once online

Non-Functional Requirements

1. Performance

Description: Ensures the application performs optimally under varying conditions.

1.1.1 The system must load content within 2-3 seconds under normal conditions.

1.1.2 During peak times, 90% of requests must respond within 5 seconds.

Dependencies:

1.2.1 Scalable server infrastructure.

1.2.2 Efficient database queries.

Verification Method:

1.3.1 Conduct load testing using tools like LoadRunner.

1.3.2 Measure response times under simulated peak conditions.

2. Scalability

Description: Supports growth in user base and data volume.

2.1.1 The system must handle up to 40,000 active users per month.

2.1.2 Data storage must accommodate 50 KB per user.

Dependencies:

2.2.1 Scalable cloud storage solutions.

Verification Method:

2.3.1 Simulate high user loads and measure system performance.

2.3.2 Verify storage scaling during tests.

1. **Availability:**

- High availability with minimal downtime for updates.

2. **Security:**

- Secure storage of user data and encryption of sensitive information.
- Implement two-factor authentication (2FA).

3. **Usability:**

- Simple, intuitive interface with multi-language support.

4. **Cross-Platform Support:**

- Available on both iOS and Android with responsive design.

5. **Data Privacy:**

- Comply with data protection laws (e.g., GDPR, CCPA).
- Users control their privacy settings.

6. **Accessibility:**

- Support for users with disabilities (e.g., voice commands, screen reader compatibility).

7. **Interoperability:**

- Integrate with fitness trackers (e.g., Fitbit, Apple Health) and music services.

8. **Maintainability:**

- Modular code for easy updates and bug fixes.

11. **Analytics:**

- Collect user behavior data to improve the app experience