Homework 2: **Software Requirements Specification**
Due by 25 November 2025

Elcano

Project team: **Team 7**
Instructor: Kamila Ismayilova

Submitted in partial fulfillment of the requirements of the CSCI 3509: Intro to Software Engineering course project

Version date	Version information
23.12.2025	Initial draft
25.12.2025	Final Version

Team member	Contribution to the homework
Ilknur Huseynova	
Toghrul Pashazada	
Javidan Taghiyev	

Table of Contents

1. Introduction

Elcano is a fun mobile application developed using React Native that aims to help users maintain a healthy daily routine. The app encourages people to be more active by tracking physical activities like walking and step counts. By gamifying daily walking, users are rewarded with coins based on their activity levels. These virtual coins can then be exchanged for discounts at partner shops. To keep users motivated, the app includes features like monthly leaderboards, challenges, badges, and achievements.

Scope and Objectives

The scope of the Elcano project is the development of a mobile application that encourages people to take more steps daily and hence be more active. Users are rewarded in the form of coins (a virtual currency earned based on activity) and discounts from partner companies for the steps they take.

The application is being developed using React Native, which is a framework that allows the app to run on both Android and iOS devices. The app is intended to be used mostly outdoors and must be simple, accurate, and easy to use for both beginners and advanced users. The name Elcano comes from the traveler Juan Sebastian Elcano, who completed the first circumnavigation of the Earth, inspiring the app's aim for users to achieve and complete their "personal journeys".

The main objective of this project is to make people more active in a rewarding way by making it fun. By gamifying daily walking, users are encouraged to move around more and earn real-life benefits.

The project targets three main user groups:

1. Casual users who wish to easily track daily step counts.
2. Health-conscious end-users seeking to achieve specific activity targets.
3. Partner companies that want to promote their business by offering discounts and rewards.

The specific objectives include:

1. To track users' steps accurately using mobile sensors.
2. To reward users with coins or points based on their activity level.
3. To provide users with the advantage of exchanging their coins for partner company discounts.
4. To display user progress, achievements, and available rewards.
5. To promote a healthy lifestyle and stronger engagement between users and partner companies.

Main Features:

The application will possess the following main features:

Step Counter: To track the number of steps taken each day.

Coin System: Converts steps into coins that can be used for rewards.

User Profile: Stores personal data and progress.

Offers Page: Displays available discounts or offers by the partners.

Dashboard: user can see statistics of total taken steps, earned coins, and activity history.

Leaderboard: Ranks users based on steps or coins earned, encouraging competition.

Login/Registration: Allows users to sign up, log in, and manage their profile.

Notifications: Reminds or congratulates the user on reaching their goals.

Data Collection:
The app will collect some basic information from users to provide a better and more personalized experience. This includes their name, email, and age, which will be used mainly for registration and to personalize their profiles. The app will also gather step count data, which is essential for tracking activity and calculating how many coins or rewards a user earns. If the user allows, their location may be used to show nearby partner companies and local offers that they can benefit from. In addition, some device information will be collected to make sure the app runs smoothly on different smartphones and to improve performance. *All collected data will only be used to enhance the user experience and to keep the reward system fair, secure, and reliable.

Definitions

Here are technical terms and definitions relevant to the Elcano project

Term	Definition
Activity Tracker	User Interface, visual elements of app that users interact with, including buttons, menus, and screens.
API (Application Programming Interface)	A set of rules and tools that allows different software programs to communicate with each other, such as retrieving activity data from Google Fit or Apple HealthKit.
Apple App Store	The official app store for iOS devices, used to distribute and download applications.
Backend	The server-side part of the application handles data storage, processing, and business logic (For example, source code).
Coin	A virtual currency earned by users in the Elcano app based on their activity. Coins can be redeemed for rewards or discounts at partner shops.
Firebase	A cloud-based platform used for app backend services, including database storage, authentication, and analytics. The Elcano project targets using Firebase for the database and backend.
Gamification	The application of game design elements and mechanics, such as points, badges, and leaderboards, to non-game contexts to increase user engagement and motivation. The Elcano app uses gamification of daily walking to encourage users to move around more and earn benefits.
Git	A distributed version control software system that is capable of managing versions of source code or data.
GitHub	A web-based platform for version control and collaborative software development that uses the Git system. The Elcano team plans to use GitHub for managing and storing their code, version control, code management, and team collaboration.
Google Play Store	The official app store for Android devices, used to distribute and download applications.
Leaderboard	A ranked list showing users' performance, such as steps taken or challenges completed.
React Native	A framework for building cross-platform mobile apps using JavaScript, allowing apps to run on both Android and iOS devices. The Elcano app is being developed using React Native for cross-platform support.
System architecture	The high-level conceptual model that defines a system's structure, behavior, and the interaction of its components, including both hardware and software.

UI (User Interface)	The visual elements of the app that users interact with, including buttons, menus, and screens.
UX (User Experience)	The overall experience of a user while interacting with the app, focusing on ease of use, satisfaction, and efficiency.
Extreme Programming (XP)	An Agile method chosen by the Elcano team that focuses on communication and making constant small improvements through short iterations (around 1–2 weeks each).

2. Overall Description

Project Features

The major functions of the Elcano mobile application are designed to encourage users to maintain a healthy daily routine by rewarding their physical activity.
The functions can be organized into four main categories: User Access and Profile Management, Core Activity Tracking, Rewards and Gamification, and Progress and Information Display.

1. User Access and Profile Management

These functions manage the user's presence within the application and personalize their experience:

• Login/Registration: The app allows users to sign up, log in, and manage their access to the system.

• User Profile: This function stores personal data (including name, email, and age) and keeps track of the user's overall progress. Personal information is used mainly for registration and profile personalization.

2. Core Activity Tracking

These are the primary functions that measure and record user movement:

• Step Counter: This feature is responsible for tracking the number of steps taken each day.

• Accurate Activity Tracking: The app must track users' steps accurately using mobile sensors. Beyond steps, the app features tracking for activities such as walking, step counting, and biking distance tracking.

3. Rewards and Gamification System

These functions motivate users by converting physical activity into tangible benefits:

• Coin System: The app features a Coin System that converts steps into coins (a virtual currency) or points based on the user's activity level. Users earn these coins as they move, for instance, by completing a specific number of steps.

• Reward Exchange: Users have the advantage of exchanging their earned coins for partner company discounts.

• Offers Page: This feature displays available discounts or offers provided by the partner companies. If the user allows, their location may be used to show nearby partner companies and local offers they can benefit from.

• Motivation Features: The app uses gamification to encourage movement by including challenges, badges, and achievements to keep users motivated.

• Leaderboard: This feature ranks users based on criteria such as steps taken or coins earned, thereby encouraging competition among users.

4. Progress and Information Display

These functions ensure users are kept informed about their activity and rewards:

• Dashboard: The Dashboard allows users to see statistics related to their total steps taken, earned coins, and activity history.

• Progress and Achievements Display: The app is required to display user progress, accomplishments (achievements), and available rewards.

• Notifications: The system will send notifications to either remind users or congratulate them when they reach their set goals.

Target User Characteristics

The intended users of the Elcano application encompass a broad audience defined primarily by their activity levels and health goals, rather than specific demographics or educational levels.

General Characteristics of Intended Users

The Elcano app is designed for **anyone who wants to maintain a healthy daily routine**. The user base can be categorized into three main groups:

1. **Casual Users:** Individuals who wish to **track daily step counts easily**.
2. **Health-Conscious End-Users:** People **seeking to achieve specific activity targets** and motivated to be more active.
3. **Partner Companies:** Businesses that want to **promote their services by offering discounts and rewards** through the application.

Technical Expertise and Device Access: The app must be suitable for both **beginners and advanced users**. Users are assumed to possess a **smartphone running Android or iOS**, equipped with necessary sensors and GPS. They must also have an **Internet connection** for syncing data and leaderboards.

Personal Data Collection and Consent: Users are expected to **agree to share activity data** for tracking and rewarding purposes. The app collects basic information such as **name, email, and age**, which is used mainly for registration and profile personalization. If permitted by the user, their **location may be used to show nearby partner companies and local offers**.

Impact on Software Design (UI/UX)

The characteristics of the potential user base heavily influence the design requirements for Elcano:

1. Simplicity and Accessibility (UI/UX): Since the app targets casual users and must be effective for both beginners and advanced users, the software design must prioritize **simplicity, accuracy, and ease of use**. A straightforward UI (User Interface) is essential to ensure a positive UX (User Experience).

2. Mobility and Performance: The app is intended to be used **mostly outdoors** while tracking activities like walking and step counting. Therefore, the performance constraints require the app to track activities accurately **without overusing the battery**.

3. Cross-Platform Compatibility: The user base operates across two primary mobile platforms (Android and iOS). To accommodate this, the development must utilize **React Native**, which is specifically chosen for **cross-platform support**.

4. Gamification and Motivation: The design must effectively promote a healthy lifestyle by incorporating **gamification** elements. This means the UI needs to be structured to prominently **display user progress, achievements, and available rewards**, and feature a **Leaderboard** to encourage competition.

5. Data Privacy and Personalization: The app collects user age and basic data for **personalization**. The design must incorporate clear mechanisms for users to consent to sharing activity and location data, as this data is crucial for calculating coins and showing local offers.

Constraints

The developer's options for building the Elcano application are limited by several non-functional requirements related to budget, schedule, performance, and required technology.

1. Project Schedule and Budget

• Schedule Constraint: The entire project development process must be completed within the semester timeline.

• Budget Constraint: The project has no external budget. This means developers must restrict their tool selection to only free or educational tools.

2. Platform and Required Technology

The choice of core technology and operating environment is restricted to ensure compatibility and efficiency:

• Mandatory Cross-Platform Support: The application must be a mobile app that is capable of running on both Android and iOS devices.

• Required Development Tools: Developers are specifically targeting the use of React Native to achieve cross-platform support.

• Mandatory Backend System: The application must utilize Firebase for the database and backend services.

• Hardware and Connectivity Requirements: The app must rely on smartphones that possess necessary sensors and GPS capabilities. An Internet connection is also required for syncing data and updating leaderboards.

• API Access: Developers are limited by the requirement to gain access to device sensors or APIs to accurately track activity data.

3. Performance and User Experience (UX) Limitations

The operational environment imposes critical performance constraints:

• Battery Life: The application must be able to track activities without overusing the battery. Since the app is intended to be used mostly outdoors, battery efficiency is crucial.

• Simplicity and Accuracy: The design must ensure the app is simple, accurate, and easy to use for both beginners and advanced users. This strictly limits the complexity of the user interface (UI) and user experience (UX) design.

• Data Usage: The app requires users to agree to share activity data for tracking and reward purposes, placing a constraint on data collection practices and necessitating clear user consent.

Assumptions and Dependencies

Assumptions

The following assumptions must hold true for the project to proceed as planned:

• Device Ownership: Users are assumed to have a smartphone running Android or iOS.

• Data Sharing Consent: It is assumed that users agree to share activity data for tracking and rewarding purposes. This data, which includes name, email, and age (for personalization), and step counts, will only be used to enhance the user experience and maintain a fair, secure, and reliable reward system.

• Simulated Partnerships: It is assumed that partnerships for the coin reward systems (gamification) may initially be simulated in the prototype.

Constraints

The development of Elcano is restricted by several non-functional requirements:

• Schedule: The project development must be completed within the semester timeline.

• Budget: The project has no external budget, so developers must restrict their use to only free or educational tools.

• Platform: The application must be a Mobile app supporting both Android or iOS.

• Resources: Developers must have access to device sensors or APIs (Application Programming Interfaces).

• Performance: The app must be designed to track activities without overusing the battery.

• Software and Tools: The project is targeting to use React Native for cross-platform support and Firebase for the database and backend.

Hardware/Platform Requirements (Dependencies)

The application depends on the following hardware and connectivity features for functionality:

• Required Hardware: Users must have Smartphones in Android or iOS with sensors and GPS.
• Connectivity: An Internet connection may be required for syncing data and leaderboards.
• Distribution: The application relies on the Mobile platform for downloading from AppStore or Google Play.

3. Specific Requirements

Functional Requirements

The functional requirements for the Elcano application define the fundamental actions the software must perform to track user activity, manage the virtual reward system, and display progress. These requirements are grouped by business area or user story.

1. User Access and Profile Management

- The system shall allow users to sign up, log in, and manage their profile.
- The system shall store personal data including name, email, and age.
- The system shall use personal data mainly for registration and to personalize user profiles.
- The system shall store the user's progress for future reference and continuous tracking.

2. Core Activity Tracking and Data Collection

- The system shall track users' steps accurately using mobile sensors.
- The system shall function as a Step Counter to track the number of steps taken each day.
- The system shall gather step count data for tracking activity and calculating earned rewards.
- The system shall track physical activities such as walking, step tracking, and biking distance tracking.

3. Rewards and Gamification System

- The system shall, based on the user's activity level, reward users with coins or points.
- The system shall include a Coin System that converts steps into coins.
- The system shall provide users with the option to exchange their coins for partner company discounts.
- The system shall display available discounts or offers provided by the partners on an Offers Page.
- The system shall use the user's location (if the user allows) to show nearby partner companies and local offers.
- The system shall include features like challenges, badges, and achievements to keep users motivated.

4. Progress and Information Display

- The system shall display user progress, achievements, and available rewards.
- The system shall include a Dashboard where the user can see statistics of total steps taken, earned coins, and activity history.
- The system shall include a Leaderboard that ranks users based on steps or coins earned, encouraging competition.
- The system shall send Notifications to either remind or congratulate the user upon reaching their set goals.

Nonfunctional Requirements

1. Usability and User Experience (UX)

- The system shall be simple, accurate, and easy to use for both beginners and advanced users.
- The system shall be designed with an intuitive UI to enhance usability for all users.
- The system shall focus on overall user experience, providing high satisfaction and efficiency.

- The system shall be optimized for use outdoors, where users are likely to engage with the app.

2. Performance and Efficiency

- The system shall track users' steps accurately using mobile sensors without draining battery excessively.
- The system shall be optimized to run smoothly across different smartphone models.
- The system shall ensure database queries and syncing operations are fast and efficient.

3. Operational and Platform Constraints

- The system shall be a mobile app and shall run on both Android and iOS devices.
- The system shall be developed using React Native for cross-platform support.
- The system shall utilize Firebase for the database and backend services.
- The system shall require access to device sensors or APIs to track activity data.
- The system may require an internet connection for syncing data and updating leaderboards.
- The system shall be deployable to mobile platforms and available for download from the App Store or Google Play.

4. Security, Data Integrity, and Privacy

- The system shall ensure that the reward system is fair, secure, and reliable.
- The system shall only use collected data (name, email, age, step count) to enhance the user experience and improve functionality.
- The system shall operate under the assumption that users consent to share their activity data for tracking and rewarding purposes.
- The system shall securely store user data, including step counts, achievements, and reward history.

5. Project and Budget Constraints

- The system's development must be completed within the semester timeline.
- The system's development shall utilize only free or educational tools, as there is no external budget.
- The development process shall follow the principles of Extreme Programming (XP), with short iterations (1–2 weeks each) to fix issues early and improve the app step by step.

4. Analysis Models (Use Cases)

External Actor Descriptions

Human Actors

User	The primary actor who uses the Elcano application to track steps, earn coins, view progress, and redeem offers. Users include casual users and health-conscious individuals.
Partner Company Representative	A business representative who registers their company in the system to provide discounts or offers (can be part of future phases).

Computer/System Actors

Mobile Sensors (Accelerometer / Pedometer / GPS)	Provides raw step-count and activity data to the system.
Firebase Backend	Stores user profiles, activity data, coins, achievements, and reward history.
Notification Service	Sends reminders, motivational messages, and goal-achievement notifications.

Use Case Descriptions

No	User story name	Description
1	User Registration & Login	The system must allow individuals to create accounts, authenticate securely, and store their personal data to enable personalized access.
2	Step Tracking	The system must continuously collect step and movement data from mobile sensors to monitor user activity throughout the day.
3	Coin Earning	The system must convert tracked steps into a virtual currency (coins) according to a predefined formula and update the user's balance.
4	View Dashboard	The system must provide a centralized dashboard that displays total steps, daily activity, earned coins, and historical data.
5	Redeem Offers	The system must display available partner offers and allow users to exchange coins for discounts or reward codes.
6	Leaderboard Participation	The system must generate a ranked leaderboard based on user steps or coins to promote competitive engagement.
7	Achievements & Challenges	The system must award badges, achievements, and challenge completions based on reaching specific activity thresholds.
8	Receive Notifications	The system must send notifications related to goals, achievements, reminders, or inactivity alerts to enhance engagement.

Use Case Diagram

The system is designed to track user activity, convert it into rewards, and provide gamified engagement through challenges, notifications, and a leaderboard. The primary actor is the User, who interacts with the system through a mobile application. Secondary actors include the Firebase Backend, Mobile Sensors, and the Notification Service.
The main functionalities captured in the use cases include:

1. **Account Creation and Login (UC-01):** Allows users to create accounts or log in to access personalized data stored in Firebase.
2. **Automatic Step Tracking (UC-02):** Collects step data from device sensors and stores it in Firebase for activity monitoring.
3. **Coin Conversion (UC-03):** Converts steps into coins automatically for use in the reward system.
4. **Offer Browsing and Reward Redemption (UC-04):** Enables users to browse available offers and redeem coins for rewards.
5. **Viewing the Leaderboard (UC-05):** Displays ranked user activity performance, fostering competition and motivation.
6. **Completing Challenges and Unlocking Achievements (UC-06):** Tracks progress toward challenges, awards badges, coins, and achievements.
7. **Sending Goal Reminders and Achievement Notifications (UC-07):** Provides push notifications to remind users of goals and congratulate achievements.

The **use case diagram** connects the User with each primary functionality, while backend systems like Firebase and Notification Services act as supporting actors. The diagram illustrates both **main flows** for successful operation and **alternate flows** for scenarios such as permission denial, sensor failure, network unavailability, or insufficient resources.

Use Case 1

Use Case Number:	UC-01
Use Case Name:	**Account Creation and Login**
Actor(s):	User, Firebase Backend
Description:	The user creates an account or logs into an existing account to access personalized data.
Trigger:	External — User opens the app and selects "Sign Up" or "Log In".

Pre-condition(s):	• The system is running and connected to Firebase. • The user has a valid internet connection.
Scenario Flow:	Main (Success) Flow 1. User opens the application. 2. System displays "Login" and "Register" options. 3. User selects "Register". 4. System requests personal data (name, email, age, password). • If required information is missing → System shows error, return to Step 4. 5. System creates a user account in Firebase Authentication. 6. System stores user profile in Firebase Database. 7. System displays confirmation of successful registration. 8. User logs in using email and password. 9. System verifies credentials. • If invalid → System displays error, return to Step 8. 10. System grants access to the Home Dashboard.
Alternate Flows:	Alternate Flows AF-01: User chooses Login instead of Register 1. User selects "Login". 2. User enters credentials. 3. System validates them. 4. System grants access to the Dashboard. AF-02: User cancels registration 1. After Step 3, user cancels registration. 2. System returns to the welcome screen.
Post Condition:	User account is created OR user is authenticated successfully. Profile data is stored in Firebase.

Use Case 2

Use Case Number:	**UC-02**
Use Case Name:	**Automatic Step Tracking**
Actor(s):	User, Mobile Sensors, Firebase Database
Description:	The system collects step data from device sensors and stores it for activity tracking.
Trigger:	Temporal — Tracking begins when the user starts the app and continues in the background.
Pre-condition(s):	• User is logged in. • System has permission to access motion sensors.
Scenario Flow:	Main (Success) Flow 1. System initializes step tracking module. 2. System requests permission to access motion sensors. • If denied → go to Alternate Flow 1. 3. Sensor streams step data to the system. 4. System calculates total steps for the day. 5. System stores step count in Firebase. 6. Dashboard updates with the new step count.
Alternate Flows:	Alternate Flows AF-01: Permission denied 1. User denies sensor access. 2. System displays message "Step tracking unavailable without permission." 3. System disables tracking features. AF-02: Sensor unavailable 1. Device sensors fail or are not present. 2. System displays error message and stops tracking.

Post Condition:	Step data is recorded and stored if tracking is functional.

Use Case 3

Use Case Number:	**UC-03**
Use Case Name:	**Coin Conversion**
Actor(s):	User, Firebase Database
Description:	The system converts recorded steps into coins according to preset rules.
Trigger:	Temporal — Conversion occurs automatically whenever new step data is saved.
Pre-condition(s):	• Valid step data exists in the system. • User is logged in.
Scenario Flow:	Main (Success) Flow 1. System initializes step tracking module. 2. System requests permission to access motion sensors. a. If denied → go to Alternate Flow 1. 3. Sensor streams step data to the system. 4. System calculates total steps for the day. 5. System stores step count in Firebase. 6. Dashboard updates with the new step count.
Alternate Flows:	Alternate Flows AF-01: Permission denied 1. User denies sensor access. 2. System displays message "Step tracking unavailable without permission." 3. System disables tracking features. AF-02: Sensor unavailable 1. Device sensors fail or are not present. 2. System displays error message and stops tracking.
Post Condition:	Coin balance is updated and stored accurately.

Use Case 4

Use Case Number:	**UC-04**
Use Case Name:	**Offer Browsing and Reward Redemption**
Actor(s):	User, Firebase
Description:	The system displays available offers and allows users to redeem coins for discounts.
Trigger:	External — User selects the "Offers" page.
Pre-condition(s):	• User is logged in. • User has a positive coin balance.
Scenario Flow:	Main (Success) Flow 1. User opens the Offers page. 2. System retrieves list of partner offers. 3. System displays offer details (reward, coin cost, partner info). 4. User selects an offer to redeem. 5. System checks if user has enough coins. • If insufficient → Alternate Flow #1. 6. System deducts required coins from the user balance. 7. System generates or displays the reward/discount code. 8. System logs redemption in user history.

Alternate Flows:	Alternate Flows AF-01: Insufficient coins 1. System displays "Not enough coins to redeem this reward." 2. System returns to Offers page. AF-02: Partner service unavailable 1. System cannot load offers due to partner server issues. 2. System displays "Offers temporarily unavailable."
Post Condition:	Coins deducted, redemption stored, reward presented to user.

Use Case 5

Use Case Number:	**UC-05**
Use Case Name:	**Viewing the Leaderboard**
Actor(s):	User, Firebase Database
Description:	The system displays a ranked leaderboard showing users' activity performance.
Trigger:	External — User navigates to the Leaderboard page.
Pre-condition(s):	• User is logged in. • Leaderboard data exists.
Scenario Flow:	Main (Success) Flow 1. User selects "Leaderboard". 2. System retrieves user rankings from Firebase. 3. System sorts users by selected metric (steps or coins). 4. System displays the ranked list. 5. User scrolls or filters results if options are available.
Alternate Flows:	Alternate Flows AF-01: No leaderboard data available 1. System shows message "Leaderboard not available yet." 2. System returns user to Dashboard. AF-02: Network connection lost 1. System displays "Cannot load leaderboard. Check your connection."
Post Condition:	Leaderboard view is displayed or user receives an appropriate message.

Use Case 6

Use Case Number:	**UC-06**
Use Case Name:	**Completing Challenges and Unlocking Achievements**
Actor(s):	User, Firebase Database
Description:	The system tracks user activity to determine when challenges are completed or achievements are unlocked, then awards badges and rewards.
Trigger:	Temporal/External — Triggered automatically when the user reaches required activity thresholds, or when the user opens the Challenges page.
Pre-condition(s):	• User is logged in. • Challenge/achievement rules are stored in Firebase. • Step tracking is functional.
Scenario Flow:	Main (Success) Flow 1. User opens the Challenges/Achievements page, or system detects progress automatically. 2. System retrieves the list of active challenges and logged user activity. 3. System compares user progress against challenge requirements (e.g., 10,000 steps/day).

	4. When a challenge requirement is met, the system marks it as completed. 5. System awards associated badges, coins, or achievements. 6. System updates user profile with newly earned badges. 7. System notifies the user of the completed challenge or unlocked achievement. 8. User views updated achievements on their profile.
Alternate Flows:	Alternate Flows AF-01: Challenge not yet completed 1. System detects insufficient progress. 2. System displays remaining requirements (e.g., "2,500 steps remaining"). 3. User may continue activity. AF-02: Database error 1. System fails to retrieve challenge definitions. 2. System displays: "Challenges unavailable at the moment." AF-03: User denies achievement notifications 1. System attempts to notify, but permission is denied. 2. System silently logs the achievement without alerting the user.
Post Condition:	• Achievements and challenge completions are stored in Firebase. • User profile displays updated badges and earned rewards.

Use Case 7

Use Case Number:	**UC-07**
Use Case Name:	**Sending Goal Reminders and Achievement Notifications**
Actor(s):	User, Notification Service, Firebase Database
Description:	The system sends push notifications to remind users about daily step goals and congratulate them after achieving milestones.
Trigger:	Temporal — At scheduled times or upon reaching activity milestones.
Pre-condition(s):	User is logged in. User has allowed notification permissions. Goal thresholds (e.g., "5,000 steps/day") exist in the database.
Scenario Flow:	Main (Success) Flow 1. System retrieves user's daily activity data. 2. System checks whether the user is close to achieving a daily or weekly goal. 3. If threshold is nearly reached, system prepares a reminder notification. 4. Notification service sends the reminder to the user's device. 5. When user reaches a goal, system prepares an achievement notification. 6. Notification service delivers the achievement message ("Congratulations! You reached 10,000 steps today!"). 7. User views the notification and returns to the app.
Alternate Flows:	Alternate Flows AF-01: Notification permissions disabled 1. System attempts to send a notification. 2. Device blocks it. 3. System logs a silent reminder internally (no user alert). AF-02: Network unavailable 1. Notification cannot be delivered. 2. System queues the notification for later. AF-03: User clears daily goal settings 1. User disables goal reminders. 2. System stops sending scheduled notifications.
Post Condition:	Notification logs stored in the system. User receives reminders and achievement alerts (if allowed).

5. References

No external references were used in the preparation of this document. All information is based on the project requirements and internal team discussions.