**Software Requirements Specification (SRS)**

# *Little Wins – Meaningful Micro-Moments App*

**Team:** Junu Rahman, Theresa Hartmann, Arooj Shahzadi
**Version:** 1.0

## 1. Purpose, Goals & Background

Little Wins is designed to help users make meaningful use of short waiting periods instead of falling into passive doomscrolling. The app provides quick, positive micro-activities based on mood, cognitive needs, and available time. The goals of Little Wins are to help users use brief waiting moments productively and positively, to reduce unintentional doomscrolling, to offer playful, social or reflective micro-experiences, and to support well-being, focus, relaxation, and social connection. All in all, Little Wins is created to provide lightweight, enjoyable prompts that turn idle moments into “little wins”.

## 2. Functional Requirements

### 2.1 Must Requirements

**M1 – Account Management:** The system shall allow users to register, log in, and log out using email, a username, and a password.

**M2 – Mode Selection:** The system shall allow users to select one of the available modes: Mood Booster**,** Brain Booster**,** Relax & Reset**,** Kindness & Connection. Activities shown shall be based on the chosen mode. The system shall allow the user to select exactly one mode per session and shall store the selected mode for that session.

**M3 – Time Selection:** The system shall allow the user to select a duration (3, 5, 10, or 15 minutes). The system shall store the selected duration for that session.

**M4 – Activity Display:** Given a selected mode and duration, the system shall display exactly one activity for that session that matches the chosen mode. For the MVP, activities shall be stored in the system (e.g. static content or database entries) and selected according to the mode–activity mapping defined in Section 3.

**M5 – Activity Completion:** The system shall provide a control (e.g. a “Done” button) that allows the user to mark the current activity as completed.

**M6 – Well-Done Summary:** The system shall display a summary showing: selected mode and duration, type of activity completed, any photos or sensor results (if applicable), and a short motivational message.

**M7 – Session Count Storage:** The system shall store the number of completed sessions per calendar day for each user. A session is defined as: mode selection → time selection → display of exactly one activity → activity completion → summary.

## 2.2 Should Requirements

**S1 – Multiple Activities per Mode:** The system should provide more than one activity per mode so that users can receive different suggestions across multiple sessions.

**S2 – Statistics View:** The system should provide a statistics screen showing the last 7 days: the number of completed sessions per day and the number of sessions per mode. Optionally, the statistics screen may also show streaks of consecutive days with at least one completed session.

**S3 – Responsive Mobile UI:** The application should adapt its layout to different screen sizes so that it is usable on smartphones, tablets, and desktop browsers.

**S4 – Local Notifications:** The system should support local browser notifications to remind the user to start a session (e.g. inactivity reminders, after-work nudges), provided the user has granted notification permission.

**S5 – Extensible Activity Types:** The system should be designed so that new activity types (for example, activities using external APIs such as jokes or riddles) can be added without major changes to existing code.

**S6 – Extensible Sensor Use:** If device sensors (e.g. orientation, geolocation) are used, the system should isolate this logic in a way that allows additional sensor-based activities to be added later with limited impact on the rest of the system.

## 2.3 Could Requirements

**C1 – Joke Challenge (API):** The system may provide an optional “Joke Challenge” activity type that retrieves a joke from a configured external jokes API and displays it to the user.
If the user requests another joke, the system shall retrieve and display a new joke from the API. If the user confirms that a stranger laughed at the joke, the system shall allow the user to take a photo using the device camera and shall display this photo on the session summary screen.

**C2 – Riddle Challenge (API):** The system may provide an optional “Riddle Challenge” activity type that retrieves a riddle and its solution from a configured external riddles API and displays the riddle to the user. If the user requests another riddle, the system shall retrieve and display a new riddle from the API. If the user confirms that a stranger answered the riddle correctly, the system shall allow the user to take a photo using the device camera and shall display this photo on the session summary screen.

**C3 – GPS Shape Challenge:** The system may provide an optional “GPS Shape Challenge” activity type that records the user’s movement using GPS coordinates after the user has granted location permission. The system shall display the recorded GPS path on a map and shall calculate a shape accuracy score based on how closely the path matches the requested target shape (e.g. circle, square, heart). The system shall display both the GPS path and the calculated accuracy score on the session summary screen.

**C4 – Sensor Mini-Games:** The system may provide at least one optional sensor-based mini-game activity type that uses one or more supported device sensors (e.g. orientation sensor, gyroscope, accelerometer). For each sensor mini-game, the system shall evaluate the user’s performance (e.g. success/failure or a numeric score) based on the sensor data and shall display the result on the session summary screen.

### 2.4 Won’t Requirements

**W1 – No Two-Factor Authentication:** The system shall not include two-factor authentication. User authentication is limited to username/email and password login only.

**W3 – No Payment or Subscription Features:** The system shall not provide any payment, in-app purchase, or subscription functionality. All features of the app are free of charge.

**W4 – No Social Media Integration:** The system shall not offer functionality to share activities, photos, results, or summaries on social media platforms. All results remain inside the application.

## 3. Definitions & Product Use

### 3.1. Definitions

**Mode:** A user-selected intention category that represents what the user currently needs (e.g. mood, focus, relaxation, social connection). The system provides the following modes: Mood Booster, Brain Booster, Relax & Reset, Kindness & Connection.

**Activity Type:** A reusable technical activity template (for example: a short reflection task, a joke-style activity, a riddle-style activity, a movement-based activity, or a sensor-based mini-game). An activity type can be assigned to one or more modes.

**Activity:** A concrete task that is shown to the user in a session. An activity is generated by combining a mode with an activity type and, optionally, a duration.

**Mode–Activity Mapping:** A configuration that assigns exactly one primary activity type to each mode for the MVP. The system uses this mapping to determine which activity type to display after the user selects a mode.

For the current version, the intended mapping is:

- Mood Booster → mood-lifting activity type
- Brain Booster → thinking / riddle-style activity type
- Relax & Reset → calm movement or breathing activity type
- Kindness & Connection → social or connection-oriented activity type

In the MVP, these activity types may be implemented as simple, internally stored activities. More advanced variants (e.g. using external joke/riddle APIs, GPS tracking or device sensors) are described as optional features in Section 2.3 (Could Requirements).

**Session:** One complete cycle starting with mode selection and time selection and ending with the summary screen. A session always includes exactly one activity: mode selection → time selection → display of one activity → activity completion → summary.

**Primary Activity Type (per mode):** The default activity type assigned to a mode via the mode–activity mapping. In the MVP, each mode has exactly one primary activity type that determines which activity is shown.

### 3.2. Product Use:

The app is designed for short idle moments such as waiting for public transport, standing in queues, waiting rooms, or brief downtime during the day. Users start and complete short sessions during these waiting periods.

## 4. Use Case Diagram and Descriptions

The following diagram shows the main interactions between the User and the *Little Wins* system. The User is outside the system boundary, while all use cases are inside the system boundary.

The use cases are organized into three groups to maintain clarity:

### 4.1 Core Session Use Cases (MUST)

- UC1 – Select Mode & Time
- UC2 – Display Activity
- UC3 – Complete Activity
- UC4 – Show Summary
- UC5 – View Statistics
- UC6 – Request Another Activity

### 4.2 Account Management Use Cases (MUST)

- UC8 – Register Account

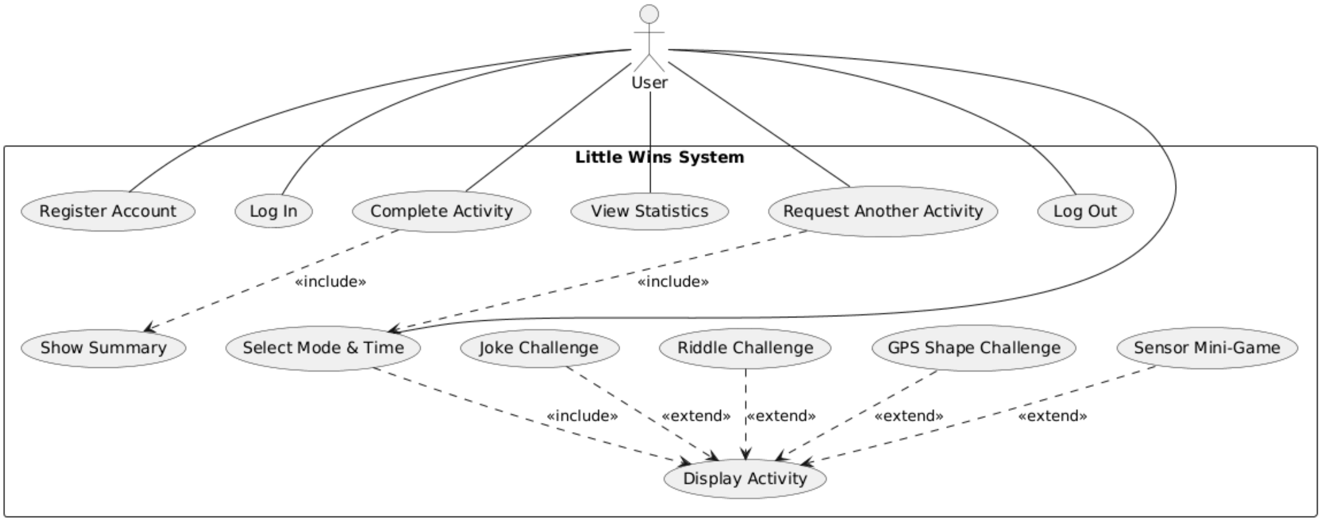
- UC9 – Log In
- UC10 – Log Out

## 4.3 Optional Activity Type Use Cases (COULD)

These represent optional advanced challenges that extend UC2.

- UC11 – Joke Challenge (extends UC2)
- UC12 – Riddle Challenge (extends UC2)
- UC13 – GPS Shape Challenge (extends UC2)
- UC14 – Sensor Mini-Game (extends UC2)

## 4.4 Use Case Diagram



## 4.5. Use Case Descriptions

### UC1 – Select Mode & Time

**Actor:** User
**Description:** The user selects a mode and a duration to begin a new session.
**Stimulus:** The user chooses a mode and duration on the home screen.
**Response:** The system loads and displays an activity based on the selected mode (UC2).
**Comment:** This use case always triggers UC2 via an include relationship.

### UC2 – Display Activity

**Actor:** User
**Description:** The system displays one activity that matches the chosen mode and duration.
**Stimulus:** The system receives a completed mode/time selection from UC1.
**Response:** The activity appears on the screen. Optional extended activities may be

triggered (UC11–UC14).
**Comment:** Extended use cases (UC6, UC11–UC14) provide optional variations.

### UC3 – Complete Activity

**Actor:** User
**Description:** The user marks the shown activity as completed.
**Stimulus:** The user selects the “Done” button.
**Response:** The system records the completion and displays the summary (UC4).
**Comment:** This use case always leads to UC4 via an include relationship.

### UC4 – Show Summary

**Actor:** User
**Description:** The system shows a summary of the completed session.
**Stimulus:** The user completes an activity (UC3).
**Response:** The summary screen displays mode, duration, activity type, and any optional media or sensor results.
**Comment:** Summary content may vary depending on activity type.

### UC5 – View Statistics

**Actor:** User
**Description:** The user views their historical usage statistics.
**Stimulus:** The user selects “Statistics” from the menu.
**Response:** The system displays session counts, mode distribution, and optional streaks.
**Comment:** Only available for logged-in users.

### UC6 – Request Another Activity

**Actor:** User
**Description:** The user requests another activity instead of continuing with the current flow.
**Stimulus - two variants:**

- **Skip current activity:** The user decides not to perform the suggested activity and requests a different one.
- **Start a new activity:** After completing an activity and viewing the summary, the user chooses to begin another activity. A new session is created.

**Response:** The system selects and displays a new activity that matches the chosen mode. If the previous activity was completed, a *new session* is created.

**Comment:** Only one use case is needed, as both situations follow the same system logic of providing a new activity.

**Optional Activity Types (Could Requirements)**

*(All extend UC2)*

**UC11 – Joke Challenge (extends UC2)**

**Actor:** User
**Description:** Displays a joke-based activity retrieved from an external jokes API.
**Stimulus:** The user selects the Joke Challenge variant while viewing an activity.
**Response:**

- The system retrieves and displays a joke.
- “Next Joke” retrieves a new one.
- If the user confirms a stranger laughed, the system enables photo capture.
- The photo is shown in the summary.
  **Comment:** Requires API availability and camera permission.

**UC12 – Riddle Challenge (extends UC2)**

**Actor:** User
**Description:** Displays a riddle-based activity retrieved from a riddles API.
**Stimulus:** The user selects the Riddle Challenge variant while viewing an activity.
**Response:**

- The system retrieves and displays a riddle.
- “Next Riddle” retrieves a new one.
- If a stranger solves the riddle, the system allows a photo.
- The photo appears in summary.
  **Comment:** Optional; requires API and camera capabilities.

**UC13 – GPS Shape Challenge (extends UC2)**

**Actor:** User
**Description:** Lets the user draw a shape using GPS movement tracking.
**Stimulus:** The user selects the GPS Shape Challenge variant and grants location access.
**Response:**

- The system records GPS movement.
- Displays the drawn shape on a map.
- Calculates an accuracy score.
- Adds map + score to summary.

**Comment:** Requires GPS/location permission.

**UC14 – Sensor Mini-Game (extends UC2)**

**Actor:** User
**Description:** Runs a short activity using available device sensors (orientation, gyroscope, accelerometer).

**Stimulus:** The user selects the Sensor Mini-Game variant.
**Response:**

- The system detects device sensors.
- Presents a small sensor-based challenge.
- Evaluates the user's performance.
- Adds score/result to summary.

**Comment:** Works only if device sensors are available.

## 6. Non-Functional Requirements

**N1 – Performance**: The system shall ensure that the initial screen loads within 2 seconds when accessed over a 10 Mbit/s internet connection.

**N2 – Interaction Efficiency**: The system shall allow the user to start an activity within a maximum of 3 clicks or taps from the home screen.

**N3 – Compatibility**: The system shall function correctly on the latest stable versions of the following web browsers: Google Chrome, Mozilla Firefox, Safari, and Microsoft Edge.

**N4 – Scalability:** The system shall support at least 100 concurrent users while maintaining a server response time of less than 2 seconds for all core interactions.

**N5 – Accessibility**: The system shall comply with the WCAG 2.1 AA accessibility standard, including requirements for color contrast, keyboard navigability, and compatibility with screen readers.