Software Requirements Specification

for

WakeUP: Alarm Clock Reinvented

Version 1.1 approved

Prepared by Rizwan Islam, Jooeun Jeon, Daniel Macias,
Darpan Patel, Yug Kalubhai Patel

CS 5337 - 01

Created Oct 10, 2024
Last Edited December 10, 2024

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Rizwan Islam, Jooeun Jeon, Daniel Macias, Darpan Patel, Yug Kalubhai Patel	10/10/2024	Initial Draft	1.0
Jooeun Jeon	12/10/2024	Addition to app functionality and UI	1.1

1. Introduction

1.1 Purpose

This document provides a comprehensive overview of the requirements for the **Wake UP!** mobile application. The primary goal is to develop an alarm application with interactive features like captchas, math problems, and trivia quizzes to enhance user engagement and productivity. This SRD outlines functional and non-functional requirements, ensuring a clear understanding of the application's behavior and purpose.

1.2 Intended Audience and Reading Suggestions

This document will be useful for the software development team, testing team, project managers, and the system's potential users. This document is best read in various ways depending on the role.

- Development Team: must understand product requirements and implementation
- Testing Team: must understand product purpose and requirements
- Project Manager: must understand product purpose and requirements
- Users: must understand product purpose and user interface paradigms

1.3 Product Scope

This mobile application Wake UP! holds the functions of traditional alarm applications and adds Google's reCAPTCHA puzzles to prevent oversleeping. The primary objective of Wake UP! is to provide users with a dependable and versatile alarm clock application that adapts to their lifestyle and improves their ability to wake up and stay organized.

1.4 Definitions, Acronyms, and Abbreviations

CAPTCHA: Completely Automated Public Turing test to tell Computers and Humans Apart.
Snooze: A feature to delay the alarm for a short period.
Timeout Period: A set duration during which the alarm cannot be dismissed.

1.5 References

1.5.1 Android Development Documentation

- Title: Android Developers Guide
- Author: Google LLC
- Version: Latest as of December 2024
- Date: Continuously updated
- Source/Location: https://developer.android.com/
- Description: Official documentation for Android app development, providing guidance on APIs, UI design, performance optimization, and platform-specific guidelines.

1.5.2 SQLite Database Documentation

- Title: SQLite Documentation
- Author: SQLite Consortium
- Version: 3.42.0
- Date: August 2024
- Source/Location: https://sqlite.org/
- Description: Reference for using SQLite for local data storage, including schema design and query optimization.

1.5.3 CAPTCHA Design Principles

- Title: CAPTCHA: Mechanisms and Implementation
- Author: Luis von Ahn et al.
- Version: 2023 Edition
- Date: February 2023
- Source/Location: https://captcha.net/
- Description: Detailed explanations of CAPTCHA systems and best practices for integration into software applications.

2. Overall Description

2.1 System Analysis

The alarm clock application is designed to enhance user productivity by providing a customizable and intuitive alarm management system. The system integrates with reCAPTCHA, to prevent oversleeping and ensure the user is fully awake, the application incorporates Google's reCAPTCHA feature. When users attempt to snooze the alarm, they are prompted to solve simple puzzles or challenges, such as identifying objects in images or solving CAPTCHA text. This ensures the user is alert before the alarm can be snoozed. Combining puzzle features and user-friendly customization, the alarm clock application aims to improve time management and ensure users can start their day efficiently and effectively.

2.2 Product Perspective

The application is independent of other products and caters specifically to users struggling with traditional alarms. While similar applications exist, **Wake UP!** differentiates itself from other alarm apps by integrating:

- Puzzles users are required to solve to turn the alarm off.
- Customizable difficulty levels.
- Purpose-driven templates for alarm configurations. These features are designed to improve usability and effectiveness compared to existing solutions.

2.3 Product Functions

2.3.1 Clock : Displays the current time.
2.3.2 Alarm: Sets alarm up like other alarm applications
2.3.2 Timer: Traditional timer function to count time in seconds.
2.3.3 Stopwatch: Traditional stopwatch function to keep track of time.
2.3.4 Settings: Changes the difficulty and alarm settings.
2.3.5.Tutorial Page: Gives brief explanations about the application and its functions
2.3.6 Help Page: Displays the information about the app and contact information for customer support

2.4 User Classes and Characteristics

Basic Users: Use traditional alarm features; minimal technical expertise required.

Advanced Users: Utilize interactive challenges and templates for enhanced productivity.

2.5 Operating Environment

The app will operate on Android devices with OS version 8.0 or later.

2.6 Design and Implementation Constraints

2.6.1 Regulatory policies: Adherence to Android security and privacy guidelines.
2.6.2 Compatibility: Supports Android Material Design principles.
2.6.3 Hardware limitations: Requires vibration and audio capabilities.

2.7 User Documentation

2.7.1 In-app Help button providing descriptions for each feature and tab.

2.8 Assumptions and Dependencies

2.8.1 Assume constant internet access for trivia question retrieval.
2.8.2 Assume user devices meet minimum hardware and OS requirements

2.9 Apportioning of Requirements

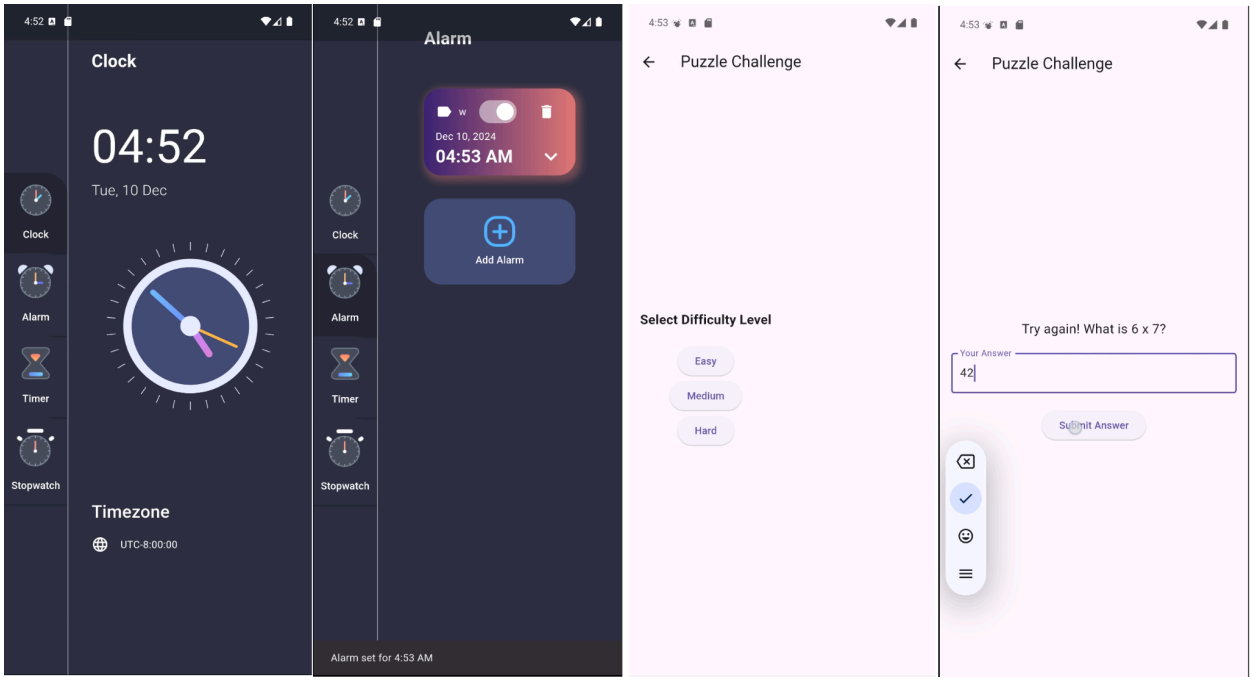
Future versions may include:

2.9.1 Additional puzzle types.

2.9.2 User login function for customized user experience.

3. External Interface Requirements

3.1 User Interfaces



3.1.1 Clock Screen

3.1.1.1 The system must display the current time with the mobile device's designated timezone.

3.1.2 Alarm Creation/Editing Screen

3.1.2.1 The system must include fields to set time, select challenge type, and configure difficulty.

3.1.2.2 The system must ensure large touch targets for accessibility.

3.1.3 Puzzle Challenge Screen

3.1.2.1 The system must dynamically display puzzles

3.1.2.2 The system must include clear instructions and large buttons for submissions.

3.1.4 Accessibility and Compliance

3.1.4.1 The system must support screen readers and provide high-contrast color options.

3.1.4.2 The system must ensure all touch elements meet a minimum size of 48x48 dp.

3.1.4.3 The system must include alt text for images and icons to support visually impaired users.

3.2 Hardware Interfaces

3.2.1 The application must support Android platforms
3.2.2 The application must operate on devices with touchscreen and speakers for sound.
3.2.3 The application requires a smartphone with network capabilities

3.3 Software Interfaces

3.3.1 The application must utilize local SQLite database for data storage
3.3.2 The application must utilize Android API for notification and alarms.

3.4 Communications Interfaces

3.4.1 The application requires a stable network connection to use reCAPTCHA
3.4.2 The application must have secure HTTP protocols for trivia API interactions.

4. Requirements Specification

4.1 Functional Requirements

4.1.1 Application Requirements

- 4.1.1.1 The application shall support Android devices
- 4.1.1.2 The application shall include an clock to show the current time
- 4.1.1.3 The application shall include an alarm interface where users can add or delete alarms.
  - 4.1.1.3.1 The alarm interface shall include simple buttons to turn alarms on or off
- 4.1.1.4 The application shall include a timer interface with basic timer functionalities.
- 4.1.1.5 The application shall include a stopwatch interface with basic stopwatch functionalities.
- 4.1.1.6 The application shall include a setting page to set puzzle difficulty for the alarm.
  - 4.1.1.6.1 The application should include a button to turn puzzles off
  - 4.1.1.6.1 The application should include multiple settings for the difficulty
- 4.1.1.7 The application should include a help page for tutorial and to contact developers of the application
- 4.1.1.8 The application shall provide a navigation to the home page

4.2 External Interface Requirements

4.2.1 User Interfaces

- 4.2.1.1 Clock
  - 4.2.1.1.1 The application must provide users without accounts a way to sign up
  - 4.2.1.1.2 The application must include input boxes for username and password
  - 4.2.1.1.3 The application must display a confirmation message once an account is formed
- 4.2.1.2 Credential Verification
  - 4.2.1.2.1 The application must verify the username and password of the user
  - 4.2.1.2.2 The application must input boxes for username and password
  - 4.2.1.2.3 The application must display all functionalities

4.2.2 Software Interfaces

- 4.2.2.1 Damoov Datahub & AWS
  - 4.2.2.1.1 The application must interface with Damoov DataHub and AWS to store user information
  - 4.2.2.1.2 The application must query according to the data accessed
  - 4.2.2.1.3 The application must output results relevant to the application

4.3 Logical Database Requirements

4.3.1 Types of information used

4.3.1.1 Application data

4.3.1.1.1 Device type

4.3.1.1.2 Android Version

4.3.1.1.3 Application Version

4.3.1.2 User Data

4.3.1.2.1 Alarm Data

4.3.1.2.2 User Preference

4.3.2 Frequency of use

4.3.2.1 Application data

4.3.2.1.1 The application must access device name and android version to check whether application is compatible with the device on start up

4.3.2.1.2 The application must access the application version to ensure the application is up to date on start up.

4.3.2.2 User Data

4.3.2.2.1 The application must access user alarm data from the database to display all current alarms.

4.3.2.2.2 The application must access user preference to check the difficulty of the puzzle.

4.3.2.2.3 The application must access user preference to check whether the quiz function is turned on or off.

4.3.3 Accessing capabilities

4.3.3.1 Application Data

4.3.3.1.1 The application may access application data exclusively via methods specifically crafted to address the situations outlined in 4.3.2.1

4.3.3.2 User Data

4.3.3.2.1 The application may access user data exclusively via methods specifically crafted to address the situations outlined in 4.3.2.2

4.4 Design Constraints

4.4.1 Internet connectivity

4.4.1.1 The application must be run with reasonably stable internet access, as slow and inconsistent internet may cause a lag in services

4.4.2 Mobile access

4.4.2.1 The application must have access to an Android device to run

4.4.2.2 The Android device must include touch screen and speaker for user interactions

5. Other Nonfunctional Requirements

5.1 Performance Requirements

5.1.1 The system shall support data storage and access for a single user.
5.1.2 The system shall take no more than three seconds for loading the data.
5.1.3 The system shall trigger alarm notifications within 1 second of the set time.

5.2 Safety Requirements

5.2.1 The system shall prevent alarms from interfering with emergency calls.
5.1.2 The system shall have a method to shut the alarm off in case of emergence.

5.3 Security Requirements

5.1.1 The system shall encrypt the setting and user data in local storage
5.1.2 The system shall use secure protocols to communicate with reCAPTCHA API.

5.4 Software Quality Attributes

5.4.1 The system must be accessible on Android devices
    5.4.1.2 The system must be responsive on all devices with version over 8.0
5.4.3 The system must accurately ring the alarm at the correct time.

5.5 Business Rules

5.5.1 The system must encrypt the user data safely
5.5.2 The system must comply to the California Consumer Privacy Act Compliance

6. Legal and Ethical Considerations

6.1 California Consumer Privacy Act Compliance

6.1.1 Data Transparency

6.1.1.1 The system must inform users about what data is collected and how it is used.

6.1.1.2 The system must not collect sensitive data without explicit user consent.

6.1.2 User Rights

6.1.2.1 The system must allow users to view and delete their data upon request.

6.1.2.2 The system must provide an option to opt-out of data sharing.

6.1.2.3 The system must ensure no reduced functionality for users exercising their privacy rights.

6.1.3 Data Security

6.1.3.1 The system must encrypt stored data using AES standards.

6.1.3.2 The system must use HTTPS for secure communication with external services.

6.1.4 Compliance

6.1.4.1 The system must comply with current CCPA regulations.

6.1.4.2 The system must notify users of any changes in privacy policies.

6.2 Data Privacy

6.1.1 Data Collection

6.2.1.1 The system must collect only essential data necessary for core functionalities, such as alarm settings and preferences.

6.2.1.2 The system must not collect personally identifiable information (PII) unless explicitly required and agreed to by the user.

6.1.2 Data Storage

6.2.2.1 The system must store all data locally on the user's device by default.

6.2.2.2 The system must encrypt stored data to prevent unauthorized access.

6.1.3 Data Access and Management

6.2.3.1 The system must provide users with access to view their stored data.

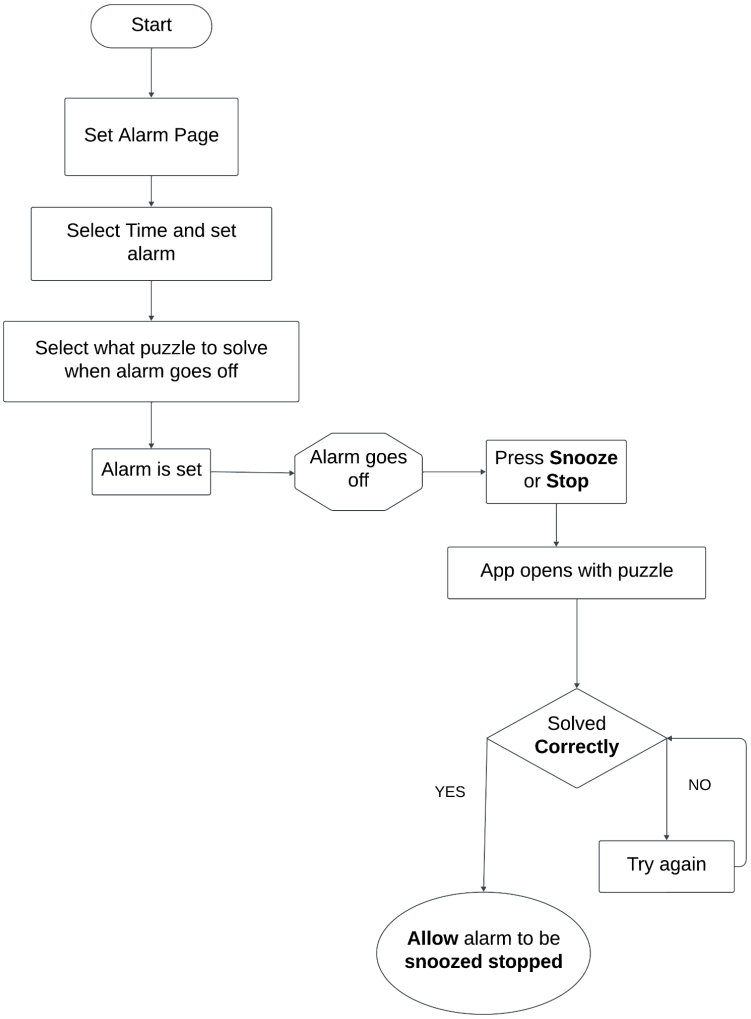
6.2.3.2 The system must allow users to delete their stored data permanently.

6.1.4 Third-Party Data Handling

6.2.4.1 The system must limit data sharing with third-party services to non-sensitive information required for external features (e.g., trivia questions).

6.2.4.2 The system must notify users of any third-party data sharing and provide an option to opt out.

Appendix A: Analysis Models



Appendix B: To Be Determined List

B.1 Feature Enhancements

B.1.1 Integration with wearable devices for additional alarm triggers (e.g., vibration-based alarms).

B.1.2 Addition of more challenge types, such as memory games or logic puzzles.

B.1.3 Support for custom user-generated challenges.

B.2 User Preferences

B.2.1 Implementation of multiple alarm tones with user-uploaded options.

B.2.2 User-defined themes and visual customization of the app interface.

B.3 Localization

B.3.1 Addition of multilingual support for broader accessibility.

B.3.2 Localization of questions and puzzle content.