

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
THE UNIVERSITY OF TEXAS AT ARLINGTON**

**SYSTEM REQUIREMENTS SPECIFICATION
CSE 4316: SENIOR DESIGN I
SPRING 2024**



**SWIFT START
SPRINT O'CLOCK**

**GIN SANG
SHAHEEN NIJAMUDHEEN
KOSUKE SATAKE
LAUREN B EYUM
CESAR FRAYRE**

REVISION HISTORY

Revision	Date	Author(s)	Description
0.1	03.25.2024	SN, GS, LE, CF, KS	document creation
0.2	03.28.2024	SN, GS, LE, CF, KS	complete draft
0.3	03.29.2024	SN, GS, LE, CF, KS	draft revision
1.0	03.29.2024	SN, GS, LE, CF, KS	official release

CONTENTS

1	Product Concept	7
1.1	Purpose and Use	7
1.2	Intended Audience	7
2	Product Description	8
2.1	Features & Functions	8
2.2	External Inputs & Outputs	8
2.3	Product Interfaces	8
3	Customer Requirements	10
3.1	Randomized Start Time Feature	10
3.1.1	Description	10
3.1.2	Source	10
3.1.3	Constraints	10
3.1.4	Standards	10
3.1.5	Priority	10
3.2	Tracking the Distance of your Run	10
3.2.1	Description	10
3.2.2	Source	10
3.2.3	Constraints	10
3.2.4	Standards	10
3.2.5	Priority	10
3.3	Distance and Time Calculation	11
3.3.1	Description	11
3.3.2	Source	11
3.3.3	Constraints	11
3.3.4	Standards	11
3.3.5	Priority	11
3.4	Recording Run History in Calendar Interface	11
3.4.1	Description	11
3.4.2	Source	11
3.4.3	Constraints	11
3.4.4	Standards	11
3.4.5	Priority	11
3.5	Settings for Run Tracking	12
3.5.1	Description	12
3.5.2	Source	12
3.5.3	Constraints	12
3.5.4	Standards	12
3.5.5	Priority	12
3.6	Sign-In Feature Synced Across IOS & Android	12
3.6.1	Description	12
3.6.2	Source	12
3.6.3	Constraints	12
3.6.4	Standards	12
3.6.5	Priority	12

3.7	Data Analysis for Fitbit Integration	13
3.7.1	Description	13
3.7.2	Source	13
3.7.3	Constraints	13
3.7.4	Standards	13
3.7.5	Priority	13
3.8	Form Demonstration for Correct Running Forms	13
3.8.1	Description	13
3.8.2	Source	13
3.8.3	Constraints	13
3.8.4	Standards	13
3.8.5	Priority	14
3.9	Competition Feature for Distance-Based Challenges	14
3.9.1	Description	14
3.9.2	Source	14
3.9.3	Constraints	14
3.9.4	Standards	14
3.9.5	Priority	14
4	Packaging Requirements	15
4.1	User Account Creation and Data Synchronization	15
4.1.1	Description	15
4.1.2	Source	15
4.1.3	Constraints	15
4.1.4	Standards	15
4.1.5	Priority	15
5	Performance Requirements	16
5.1	Fast Startup and Responsive Operation	16
5.1.1	Description	16
5.1.2	Source	16
5.1.3	Constraints	16
5.1.4	Standards	16
5.1.5	Priority	16
6	Safety Requirements	17
6.1	Privacy and Data Security	17
6.1.1	Description	17
6.1.2	Source	17
6.1.3	Constraints	17
6.1.4	Standards	17
6.1.5	Priority	17
6.2	National Electric Code (NEC) wiring compliance	17
6.2.1	Description	17
6.2.2	Source	17
6.2.3	Constraints	17
6.2.4	Standards	17
6.2.5	Priority	17

6.3	RIA robotic manipulator safety standards	18
6.3.1	Description	18
6.3.2	Source	18
6.3.3	Constraints	18
6.3.4	Standards	18
6.3.5	Priority	18
6.4	Laboratory equipment lockout/tagout (LOTO) procedures	18
6.4.1	Description	18
6.4.2	Source	18
6.4.3	Constraints	18
6.4.4	Standards	18
6.4.5	Priority	18
7	Maintenance & Support Requirements	19
7.1	Closed Source	19
7.1.1	Description	19
7.1.2	Source	19
7.1.3	Constraints	19
7.1.4	Standards	19
7.1.5	Priority	19
7.2	Video User Manual	19
7.2.1	Description	19
7.2.2	Source	19
7.2.3	Constraints	19
7.2.4	Standards	19
7.2.5	Priority	20
8	Other Requirements	21
8.1	Functional Start	21
8.1.1	Description	21
8.1.2	Source	21
8.1.3	Constraints	21
8.1.4	Standards	21
8.1.5	Priority	21
9	Future Items	22
9.1	Competition Feature	22
9.1.1	Description	22
9.1.2	Source	22
9.1.3	Constraints	22
9.1.4	Standards	22
9.1.5	Priority	22

LIST OF FIGURES

1 High-Level Use Case Diagram 7

1 PRODUCT CONCEPT

This section describes the purpose, use, and intended audience for the Sprint O Clock app, which tracks the running distance using GPS and provides features that improve runners's race experience, especially those in sprints or interval training.

1.1 PURPOSE AND USE

Sprint O'Clock is a mobile app designed to accurately track running distance and time, particularly for sprints or interval training. Users can choose between two devices for tracking their runs: a phone or a smartwatch. When using a phone as the run device, the app provides various functionalities, including a customizable start timer, audio/feedback system, phone vibration, and GPS distance tracking. Pressing the play button on the app's home page initiates a countdown with "On your mark, Set, Go!" in a randomized time, enhancing auditory reaction. Once the distance is tracked, the data is logged into a calendar interface for future reference. On the other hand, selecting a smartwatch as the run device limits app functionality to syncing API data. The watch tracks the run, and the app logs the most updated run data into the calendar interface. The calendar interface offers a visual representation of the running journey, displaying past runs and running statistics over time. This feature helps users reflect on their progress and stay motivated to achieve their fitness goals.

1.2 INTENDED AUDIENCE

The intended audiences for the Sprint O Clock app includes runners and athletes who engage in sprint training or interval training. The app is suitable for both casual runners and serious athletes looking to improve their performance, auditory reaction time, and running form. It is designed to be user-friendly and accessible to anyone who wants to track their running distance and time more effectively.

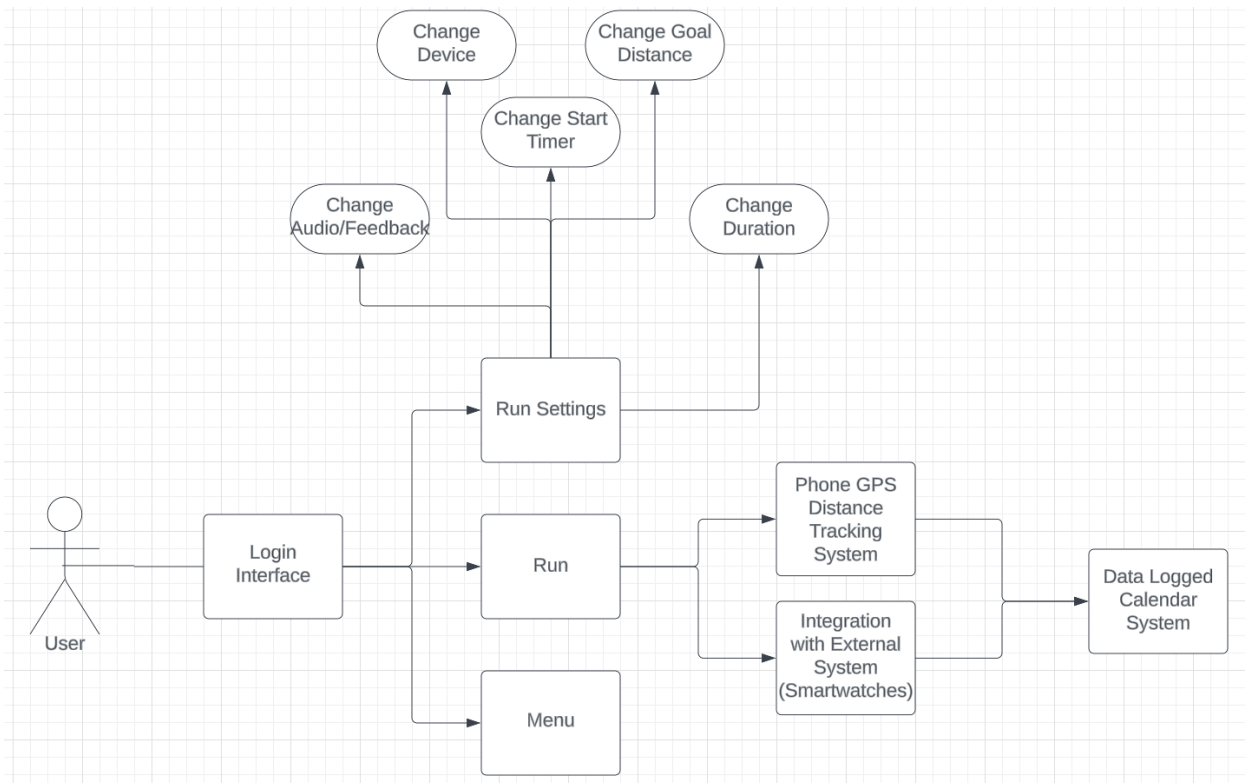


Figure 1: High-Level Use Case Diagram

2 PRODUCT DESCRIPTION

Sprint O’Clock is an advanced mobile app designed for sprinters and track field athletes. It helps them track and manage their training sessions and races in a new and improved way. The app is easy to use and offers a range of powerful features to enhance the sprinter’s experience. It provides detailed insights and analysis to help improve performance.

2.1 FEATURES & FUNCTIONS

Sprint O’Clock is a mobile app built using React Native, which enables cross compatibility to both IOS and Android phones. It will provide a sign in feature, which you can you to create your own username, password, and profile picture.

Sprint O’Clock provides a set of features that facilitate the Track Field race experience. Features such as competition feature, random start timer, phone vibration, audio/feedback, GPS distance tracking, Smartwatch Activity API Data, and Data-Logged Calendar Interface are all helpful for runners who are looking forward to practicing effectively anywhere and anytime.

2.2 EXTERNAL INPUTS & OUTPUTS

The Sprint O’Clock app integrates with an external smartwatch, syncing its activity API data to the mobile app, in addition to tracking the the running distance using GPS Data, Time Data, and Distance Data.

Data Element	Description	Use
GPS Data	Location data provided by GPS satellites	Used to track the user’s running route and distance
Time Data	Timestamps of when the run starts, and ends, as well as pause times.	Used to calculate the total time taken for the run
Distance Data	Distance covered during the run	Used to calculate the total distance covered.
Smartwatch Activity API Data	Data from the smartwatch’s activity tracking API, including steps, heart rate, and other activity metrics	Used to supplement the app’s tracking data and provide additional insights into the user’s activity

The mobile app expects to receive GPS Data, Time Data, and Distance Data from its internal tracking functionality, as well as Smartwatch Activity API Data from the external smartwatch. The app processes this data to provide users with comprehensive insights into their running activities, including route tracking, distance covered, time taken, and other relevant metrics.

2.3 PRODUCT INTERFACES

The app interface includes the following pages: Login interface where users can create an account using their username, email address, and password, or log in using their existing credentials. The Run Page, which is the main page displayed after logging in, features run settings for phone vibration, feedback audio, device choice (phone or smartwatch), and duration and distance settings, along with a "Start Run" button. It also displays device details, including distance, time limit, and start timer information, as well as the user’s profile pic and profile name in the top right corner. The Run Settings Page allows users to configure various settings related to their running experience, such as toggling phone vibration and feedback audio, choosing between using their phone or a smartwatch to track their run, and setting duration and distance preferences. The Start Timer Settings Page offers various timer functionalities, including adjustment time for extra preparation, a timed preparation period with "On your mark..." and "Set..." phases, and a random start timer that adds an element of unpredictability to the start time. After

the countdown concludes, the screen displays a "Go", the audio plays a gun sound, and the GPS distance tracker for the run starts simultaneously, simulating real-world scenarios and adding excitement to the start of the run.

3 CUSTOMER REQUIREMENTS

"Swift Start" is a mobile application tailored for sprinters, facilitating efficient management of their starting times through agile and scrum methodologies. The following customer requirements are essential features and functionalities of the product to meet the needs of the intended audience:

3.1 RANDOMIZED START TIME FEATURE

3.1.1 DESCRIPTION

Swift Start shall offer a randomized start time functionality accompanied by audible alerts or vibration from smartphones. This feature will be initiated from the Home Page by tapping the "Start Button".

3.1.2 SOURCE

Randomized Start Time Feature must be created according to customers' needs.

3.1.3 CONSTRAINTS

Randomized Start Time Feature must provide a user-friendly interface for getting the randomized start time.

3.1.4 STANDARDS

The randomized start time feature must adhere to best practices for user experience design to ensure intuitive and user-friendly operation.

3.1.5 PRIORITY

- High

3.2 TRACKING THE DISTANCE OF YOUR RUN

3.2.1 DESCRIPTION

This feature shall allow users to monitor their running distance in real-time using GPS technology. Accessed from the "Run" page, users start tracking with a tap, seeing their distance covered as they run. They can pause or stop tracking as needed. Afterward, users can review their total distance for the session, aiding in goal setting and fitness monitoring.

3.2.2 SOURCE

Randomized Start Time Feature must be created according to customers' needs.

3.2.3 CONSTRAINTS

1. Battery Usage: Ensure minimal battery drain from GPS usage. 2. GPS Accuracy: Account for varying GPS signal strengths. 3. Device Compatibility: Optimize for diverse smartphone models. 4. Privacy Compliance: Adhere to stringent data protection regulations. 5. Network Reliance: Address challenges in areas with poor connectivity. 6. User Safety: Provide in-app safety guidance for runners.

3.2.4 STANDARDS

A comprehensive privacy policy must be developed and made available to users. This policy should address the collection, storage, and use of user data, including any data captured or processed during the operation of the randomized start time feature. Compliance with applicable data protection regulations, such as the General Data Protection Regulation (GDPR) in the European Union or the California Consumer Privacy Act (CCPA) in the United States, should be ensured.

3.2.5 PRIORITY

- High

3.3 DISTANCE AND TIME CALCULATION

3.3.1 DESCRIPTION

This feature uses GPS to measure and display real-time distance and elapsed time during runs. It continuously updates the distance covered and records the duration from the start of the session. Users can easily monitor their progress on the Home page. Afterward, they can review their total distance and time for insights into their performance.

3.3.2 SOURCE

Randomized Start Time Feature must be created according to customers' needs.

3.3.3 CONSTRAINTS

Detailed description of applicable constraints...

3.3.4 STANDARDS

A comprehensive privacy policy must be developed and made available to users. This policy should address the collection, storage, and use of user data, including any data captured or processed during the operation of the randomized start time feature. Compliance with applicable data protection regulations, such as the General Data Protection Regulation (GDPR) in the European Union or the California Consumer Privacy Act (CCPA) in the United States, should be ensured

3.3.5 PRIORITY

- High

3.4 RECORDING RUN HISTORY IN CALENDAR INTERFACE

3.4.1 DESCRIPTION

This feature logs completed runs in a calendar interface, providing users with a visual history of their activities. Each run, including date, distance, and duration, is automatically added to the calendar. Users can easily navigate, search, and analyze their past runs to track progress and set future goals effectively.

3.4.2 SOURCE

Randomized Start Time Feature must be created according to customers' needs.

3.4.3 CONSTRAINTS

1. Storage Capacity: Manage database storage efficiently to accommodate past run records. 2. Data Integrity: Ensure robust error handling to prevent data corruption or loss. 3. Calendar Compatibility: Ensure seamless integration across different calendar platforms. 4. Privacy Protection: Implement security measures to safeguard user data. 5. User Control: Provide options for users to manage their run history data. 6. Performance Optimization: Optimize calendar rendering for smooth performance on all devices.

3.4.4 STANDARDS

Data Security: Ensure secure storage and encryption of user data. Calendar Integration: Follow compatibility standards for seamless integration. Privacy Compliance: Adhere to privacy regulations and implement privacy-by-design. Data Retention: Establish clear policies for data retention and deletion.

3.4.5 PRIORITY

- High

3.5 SETTINGS FOR RUN TRACKING

3.5.1 DESCRIPTION

This feature allows users to customize their run tracking experience. Users can enable phone vibration for progress updates and configure feedback intervals for lap notifications. Additionally, users can choose distance units (e.g., kilometers or miles) for displaying run distance.

3.5.2 SOURCE

Randomized Start Time Feature must be created according to customers' needs.

3.5.3 CONSTRAINTS

Device Compatibility: Ensure compatibility across various smartphones. Battery Efficiency: Optimize features to minimize battery consumption. User Preferences: Offer customizable options for vibration and feedback. Accessibility: Adhere to accessibility standards for all users. Privacy Compliance: Securely manage user data in accordance with regulations.

3.5.4 STANDARDS

User Experience: Design interface for intuitive customization. Vibration Patterns: Follow industry standards for clear feedback. Feedback Intervals: Balance timely updates with user experience. Distance Unit Selection: Maintain consistency in displaying distance. Security Measures: Implement encryption to protect user settings.

3.5.5 PRIORITY

- High

3.6 SIGN-IN FEATURE SYNCED ACROSS IOS & ANDROID

3.6.1 DESCRIPTION

This feature allows users to sign in to the app using their credentials, ensuring seamless synchronization of data and preferences across both iOS and Android devices. Users can access their accounts from any compatible device, maintaining consistency in their experience regardless of the platform used.

3.6.2 SOURCE

Randomized Start Time Feature must be created according to customers' needs.

3.6.3 CONSTRAINTS

Compatibility: Ensure compatibility across iOS and Android versions. Synchronization: Implement robust data synchronization mechanisms. Security: Adhere to stringent security protocols for user data protection. Consistency: Maintain consistency in user experience across platforms. Account Management: Enable seamless account management across devices.

3.6.4 STANDARDS

OAuth Protocol: Use OAuth for secure user authorization. Authentication Standards: Follow OpenID Connect for reliable sign-in. Encryption: Employ TLS for data encryption during transmission. Interoperability: Adhere to standards for smooth communication between platforms. Accessibility Compliance: Design interfaces in compliance with accessibility standards.

3.6.5 PRIORITY

- Low

3.7 DATA ANALYSIS FOR FITBIT INTEGRATION

3.7.1 DESCRIPTION

This feature integrates Fitbit data into the app for analysis. Users sync their Fitbit accounts to view fitness metrics like steps, distance, calories, and sleep patterns. Data is displayed in a dashboard with graphs and stats, helping users track progress and set goals. The integration provides a comprehensive view of fitness activities, enhancing user understanding and motivation.

3.7.2 SOURCE

Randomized Start Time Feature must be created according to customers' needs.

3.7.3 CONSTRAINTS

API Compliance: Adhere to Fitbit's API guidelines to avoid usage limitations. Data Privacy: Ensure compliance with privacy regulations and implement robust security measures. Compatibility Testing: Validate functionality across various Fitbit devices and versions. User Consent: Obtain explicit user consent for accessing and using Fitbit data.

3.7.4 STANDARDS

API Adherence: Follow Fitbit API documentation for reliable data synchronization. Data Security: Implement encryption and security measures for data protection. Privacy Compliance: Comply with GDPR and HIPAA regulations for data privacy. Interoperability: Ensure seamless integration with Fitbit devices. Usability Optimization: Design user-friendly interfaces for presenting Fitbit data.

3.7.5 PRIORITY

- Moderate

3.8 FORM DEMONSTRATION FOR CORRECT RUNNING FORMS

3.8.1 DESCRIPTION

This feature offers instructional content on proper running form. Users access videos, animations, or written guides covering posture, stride, arm movement, and breathing techniques. Interactive elements may include quizzes for self-assessment. By promoting correct form, the feature aims to enhance running performance and reduce the risk of injury for users of all levels.

3.8.2 SOURCE

Randomized Start Time Feature must be created according to customers' needs.

3.8.3 CONSTRAINTS

Content Accuracy: Ensure instructional accuracy for reliable guidance. Accessibility: Make content accessible to users with diverse needs. User Engagement: Maintain engagement with interactive elements. Device Compatibility: Optimize content for various devices. Continuous Improvement: Regularly update content based on feedback and research.

3.8.4 STANDARDS

Content Accuracy: Adhere to established standards for instructional accuracy. Accessibility: Design content following WCAG guidelines. User Engagement: Utilize effective UX design principles. Multimedia Integration: Integrate multimedia using industry-standard formats. Quality Assurance: Implement QA processes for consistent quality.

3.8.5 PRIORITY

- Low

3.9 COMPETITION FEATURE FOR DISTANCE-BASED CHALLENGES

3.9.1 DESCRIPTION

This feature allows users to participate in distance-based challenges and competitions. Users can create or join challenges, setting goals for covering specific distances within a set time frame. They track their progress in real-time, comparing their performance with others on a leaderboard. Social sharing options promote community engagement and motivation.

3.9.2 SOURCE

Randomized Start Time Feature must be created according to customers' needs.

3.9.3 CONSTRAINTS

Scalability: Ensure platform performance with a large user base. Fairness: Prevent cheating and ensure fair competition. Privacy: Respect user privacy and comply with regulations. Engagement: Maintain user interest with diverse challenges. Compatibility: Ensure functionality across various devices.

3.9.4 STANDARDS

Fair Play: Establish clear rules for fair competition. Security: Protect user data with industry-standard encryption. Accessibility: Design for accessibility following WCAG guidelines. Interoperability: Ensure compatibility with external devices. User Experience: Prioritize intuitive design for user satisfaction.

3.9.5 PRIORITY

- Low

4 PACKAGING REQUIREMENTS

Our Sprint O'Clock app, designed to enhance productivity and fitness routines with a touch of unpredictability, is set to revolutionize training and work sessions. Available for free on both the App Store and Google Play Store, this app is crafted for seamless user installation and accessibility, ensuring a broad spectrum of users can benefit from its unique features. With a focus on cross-platform availability and minimal device requirements, we're committed to delivering a user-friendly and engaging experience to anyone looking to transform their sprints into something extraordinary. This section will include any constraints, standards, and priorities that apply to the packaging requirements.

4.1 USER ACCOUNT CREATION AND DATA SYNCHRONIZATION

4.1.1 DESCRIPTION

This requirement aims to enable users to create personal accounts in the app for saving their preferences, workout history, and progress, with seamless data synchronization across devices. Users will register via email, ensuring their data is securely stored and accessible. The app will operate consistently across iOS and Android, utilizing React Native for a uniform interface. This ensures all features, like timer customization and notifications, work flawlessly on any device, enhancing user engagement by maintaining progress and preferences regardless of the device.

4.1.2 SOURCE

This requirement is derived from the core objective of maximizing the app's accessibility and user base, catering to both major mobile operating systems.

4.1.3 CONSTRAINTS

- **Development Environment:** The development team must be proficient in React Native and its ecosystem, including external libraries compatible with both iOS and Android.
- **Performance Optimization:** Achieving similar performance benchmarks on both platforms, considering their distinct hardware and operating system characteristics.
- **Testing Coverage:** Ensuring comprehensive testing on both platforms, requiring access to a range of devices to account for different screen sizes, resolutions, and OS versions.

4.1.4 STANDARDS

- **React Native Documentation:** Adherence to best practices and guidelines as outlined in the official React Native documentation.
- **Platform-Specific Guidelines:** Compliance with Apple's Human Interface Guidelines and Google's Material Design for iOS and Android app development, respectively.

4.1.5 PRIORITY

Moderate.

5 PERFORMANCE REQUIREMENTS

Our Sprint O’Clock app stands out by ensuring quick start-up, seamless activity transitions, and optimized battery life for everyday use. It’s built to work fast and efficiently across different phones, minimizing setup and loading times. With swift responsiveness and low latency, it keeps you moving without delays, making every workout or productivity sprint more effective. Practical and user-friendly, it’s designed to fit into your busy life effortlessly.

5.1 FAST STARTUP AND RESPONSIVE OPERATION

5.1.1 DESCRIPTION

This requirement specifies that the app must launch within 2 seconds of user initiation and ensure operations such as timer start, stop, and setting adjustments are responsive, with actions taking no more than 1 second to register. The goal is to provide a seamless user experience, minimizing wait times and enhancing the app’s usability during workout sessions or productivity sprints.

5.1.2 SOURCE

User feedback and competitive benchmarking have highlighted the importance of quick startup times and responsive app operations as key factors for user satisfaction and retention.

5.1.3 CONSTRAINTS

- **Device Variability:** Performance must be consistent across various devices with differing hardware capabilities.
- **Battery Efficiency:** Enhancing responsiveness should not come at the cost of significantly increased battery consumption.
- **Background Processes:** The app must manage its background processes efficiently to avoid slowing down the device’s overall performance.

5.1.4 STANDARDS

- **ISO/IEC 25010:** This standard for software quality requirements and evaluation (SQuaRE) covers aspects of usability and performance efficiency that are relevant to this requirement.
- **Mobile Application Performance Metrics:** Guidelines on responsiveness and battery usage efficiency for mobile applications will also be adhered to.

5.1.5 PRIORITY

Moderate.

6 SAFETY REQUIREMENTS

Because Sprint O' Clock is a software application, privacy and data security are of the utmost concern. Therefore the safety requirements relevant to this application will be software based.

6.1 PRIVACY AND DATA SECURITY

6.1.1 DESCRIPTION

Proper authentication and authorization mechanisms should be implemented to prevent unauthorized access to user data. User data, including personal health data from Fitbit integration, must be encrypted both during transmission and storage to protect against unauthorized access.

6.1.2 SOURCE

Ensure compliance with relevant data protection regulations such as GDPR or CCPA to safeguard user privacy.

6.1.3 CONSTRAINTS

Ensuring data privacy and security often requires intricate technical solutions, which may be challenging to implement or maintain, especially for smaller development teams or less experienced developers. Limited time and budget constraints may hinder the implementation of comprehensive data privacy and security measures, potentially leading to compromises in the thoroughness and effectiveness of security protocols and compliance efforts.

6.1.4 STANDARDS

Developers should collect only the minimum amount of data necessary for the app's functionality, avoiding the storage of unnecessary sensitive information. This approach not only mitigates privacy risks but also reduces the impact in the event of unauthorized access or data breaches.

6.1.5 PRIORITY

Critical

6.2 NATIONAL ELECTRIC CODE (NEC) WIRING COMPLIANCE

6.2.1 DESCRIPTION

Any electrical wiring must be completed in compliance with all requirements specified in the National Electric Code. This includes wire runs, insulation, grounding, enclosures, over-current protection, and all other specifications.

6.2.2 SOURCE

CSE Senior Design laboratory policy

6.2.3 CONSTRAINTS

High voltage power sources, as defined in NFPA 70, will be avoided as much as possible in order to minimize potential hazards.

6.2.4 STANDARDS

NFPA 70

6.2.5 PRIORITY

Critical

6.3 RIA ROBOTIC MANIPULATOR SAFETY STANDARDS

6.3.1 DESCRIPTION

Robotic manipulators, if used, will either be housed in a compliant lockout cell with all required safety interlocks, or certified as a "collaborative" unit from the manufacturer.

6.3.2 SOURCE

CSE Senior Design laboratory policy

6.3.3 CONSTRAINTS

Collaborative robotic manipulators will be preferred over non-collaborative units in order to minimize potential hazards. Sourcing and use of any required safety interlock mechanisms will be the responsibility of the engineering team.

6.3.4 STANDARDS

ANSI/RIA R15.06-2012 American National Standard for Industrial Robots and Robot Systems, RIA TR15.606-2016 Collaborative Robots

6.3.5 PRIORITY

Critical

6.4 LABORATORY EQUIPMENT LOCKOUT/TAGOUT (LOTO) PROCEDURES

6.4.1 DESCRIPTION

Any fabrication equipment provided used in the development of the project shall be used in accordance with OSHA standard LOTO procedures. Locks and tags are installed on all equipment items that present use hazards, and ONLY the course instructor or designated teaching assistants may remove a lock. All locks will be immediately replaced once the equipment is no longer in use.

6.4.2 SOURCE

CSE Senior Design laboratory policy

6.4.3 CONSTRAINTS

Equipment usage, due to lock removal policies, will be limited to availability of the course instructor and designated teaching assistants.

6.4.4 STANDARDS

Occupational Safety and Health Standards 1910.147 - The control of hazardous energy (lockout/tagout).

6.4.5 PRIORITY

Critical

7 MAINTENANCE & SUPPORT REQUIREMENTS

Maintenance and support requirements for the Sprint O' Clock app are essential to ensure its ongoing functionality and usability post-delivery. As developers, we must consider what is necessary for effective product maintenance. These requirements include considerations for access to technical documentation and specific tools needed for maintenance.

7.1 CLOSED SOURCE

7.1.1 DESCRIPTION

Sprint O' Clock's source code will remain closed source upon final delivery. This means that the source code will not be made publicly available, limiting access to internal development teams only.

7.1.2 SOURCE

The requirement was determined by the development team of Sprint O' Clock.

7.1.3 CONSTRAINTS

Maintaining closed-source code limits external contributions to the app's development, ensuring that changes align with the product's goals but may restrict innovation from the broader developer community.

7.1.4 STANDARDS

Closed-source projects adhere to internal coding standards and documentation practices to maintain code integrity and facilitate internal collaboration.

7.1.5 PRIORITY

This requirement has a high priority to safeguard proprietary code and intellectual property, ensuring that the app's development remains under the control of the internal team.

7.2 VIDEO USER MANUAL

7.2.1 DESCRIPTION

The Sprint O' Clock app shall provide a comprehensive video user manual as a support resource for users. This video manual will offer visual demonstrations and walk-through of key app features, guiding users through setup, usage, and troubleshooting processes.

7.2.2 SOURCE

The requirement was determined by the development team of Sprint O' Clock.

7.2.3 CONSTRAINTS

Creating a video user manual requires significant time and resources for scripting, filming, editing, and distribution. Additionally, ensuring the video content remains up-to-date with app updates and changes poses ongoing maintenance challenges.

7.2.4 STANDARDS

The video user manual must adhere to high production standards, including clear audio, visual clarity, and concise yet informative content. It should align with the app's branding and design guidelines for consistency.

7.2.5 PRIORITY

This requirement has a moderate priority as it enhances user experience and reduces support inquiries by providing users with accessible, visual guidance. However, its implementation may require careful planning and coordination to balance resource allocation with other development priorities.

8 OTHER REQUIREMENTS

In order for this app to be complete, the app needs to be able to have a randomized start time feature and functional to the point that it has a start button on the run page that tracks the distance. This app will be need to be configured in React Native/Java Script, we are still figuring out how to implement this part. The app is leaning more towards android since it does have better implementation and the watches are compatible.

8.1 FUNCTIONAL START

8.1.1 DESCRIPTION

The app shall have a functional starting phase that will have a randomized start time feature and then the distance should be covered by the runner.

8.1.2 SOURCE

Functional start will follow through the runners pace from start to finish by the watch.

8.1.3 CONSTRAINTS

Distance might not be 100 percent accurate as there is not computer vision.

8.1.4 STANDARDS

The application should meet industry standards for app development.

8.1.5 PRIORITY

High

9 FUTURE ITEMS

Since this is a mobile app we will consider creating a competition feature, where users can request a competition with others for a selected distance. The most notable requirements for this feature are listed below:

1. Customer/Communication Requirements:
 - Sprint Planning, Backlog Management, Task Boards
2. Package Requirements:
 - Software Delivery
3. Performance Requirements:
 - Server, Live Action for Competition Feature
4. Maintenance and Support Requirements:
 - Server Support/User Account
5. Other Requirements:
 - Server/Watch compatibility
6. Future Items:
 - App competition and User Log in

9.1 COMPETITION FEATURE

9.1.1 DESCRIPTION

The app shall have a competition feature, where users can request a competition for a selected distance. This probes the user to not only keep track of their own runs, but also strive the user to compete with other runners.

9.1.2 SOURCE

Server Implementation

9.1.3 CONSTRAINTS

This feature may require to add a lot of other features which will dictate our time to create this app.

9.1.4 STANDARDS

This feature should adhere to the same designs that other apps usually do competitions with other players, like a mini game.

9.1.5 PRIORITY

Low