



**ENGINEERING**  
Computing & Software

# ACME RUN

## *Requirements Standard Plan*

Aidan Froggatt, Monica Bazina-Grolinger, Kate Min

Version 3, 2023-12-06

# Table of Contents

<b>Control Information</b> .....	<b>2</b>
<b>(G) Goals</b> .....	<b>3</b>
<i>(G.1) Context and Overall Objectives</i> .....	<i>3</i>
<i>(G.2) Current situation</i> .....	<i>3</i>
<i>(G.3) Expected Benefits</i> .....	<i>4</i>
<i>(G.4) Functionality overview</i> .....	<i>5</i>
<i>(G.5) High-level usage scenarios</i> .....	<i>6</i>
<i>(G.6) Limitations and Exclusions</i> .....	<i>9</i>
<i>(G.7) Stakeholders and requirements sources</i> .....	<i>9</i>
Direct Stakeholders .....	9
Indirect Stakeholders .....	11
Requirements Sources .....	11
<b>(E) Environment</b> .....	<b>12</b>
<i>(E.1) Glossary</i> .....	<i>12</i>
(E.1.1) Vocabulary .....	12
(E.1.2) Domain Model .....	13
<i>(E.2) Components</i> .....	<i>13</i>
<i>(E.3) Constraints</i> .....	<i>14</i>
<i>(E.4) Assumptions</i> .....	<i>14</i>
<i>(E.5) Effects</i> .....	<i>14</i>
<i>(E.6) Invariants</i> .....	<i>15</i>
<b>(S) System</b> .....	<b>16</b>
<i>(S.1) Components</i> .....	<i>16</i>
<i>(S.2) Functionality</i> .....	<i>17</i>
(S.2.1) User Manager .....	17
(S.2.2) Trail Manager .....	17
(S.2.3) Run Manager .....	18
(S.2.4) Event Manager .....	18
(S.2.5) Non-Functional Requirements .....	18
<i>(S.3) Interfaces</i> .....	<i>18</i>
(S.3.1) APIs .....	19
(S.3.2) Wireframe Mockups .....	19
<i>(S.4) Detailed usage scenarios</i> .....	<i>21</i>
(S.4.1) Player starting a running session .....	21
(S.4.2) Player encountering virtual enemies during running session .....	23
(S.4.3) Player checking their stats and personalized training plan .....	24

(S.4.4) Local partner using app to update trail information .....	25
(S.4.5) Player submitting feedback .....	26
(S.5) <i>Prioritization</i> .....	27
<i>Priority Table for User Manager</i> .....	27
<i>Priority Table for Trail Manager</i> .....	28
<i>Priority Table for Run Manager</i> .....	28
<i>Priority Table for Event Manager</i> .....	29
<i>Priority Table for Non-Functional Requirements</i> .....	30
(S.6) <i>Verification and acceptance criteria</i> .....	30
<b>(P) Project</b> .....	<b>33</b>
(P.1) <i>Roles and personnel</i> .....	33
(P.2) <i>Imposed technical choices</i> .....	33
(P.3) <i>Schedule and milestones</i> .....	34
(P.4) <i>Tasks and deliverables</i> .....	35
(P.5) <i>Required technology elements</i> .....	35
(P.6) <i>Risk and mitigation analysis</i> .....	36
(P.7) <i>Requirements process and report</i> .....	37
<b>References</b> .....	<b>39</b>
<b>Appendices</b> .....	<b>40</b>
(Appendix A) <i>Security Considerations</i> .....	40
Security by Obscurity .....	40
Security Property Assessment .....	40
(Appendix B) <i>Window of Opportunity #1</i> .....	40
(Appendix C) <i>Window of Opportunity #2</i> .....	42

# Academic Integrity Disclaimer

We would like to acknowledge that as a dedicated students of McMaster University, we have thoroughly read and comprehended the [Academic Integrity Policy](#) published by the university. We are committed to upholding the principles of academic honesty and integrity in all aspects of our educational journey. We understand the importance of acknowledging the work and ideas of others, and we pledge to ensure that all our academic endeavors are conducted with the utmost originality and compliance with the university's policy.

**We affirm that the content presented in this document is entirely our own, and any external sources used have been appropriately cited and referenced.**

## Aidan Froggatt

As I submit my work, I, **Aidan Froggatt**, take full responsibility for the integrity of my work and promise to avoid any form of plagiarism, cheating, or dishonest behavior. This acknowledgment serves as a testament to my dedication to academic excellence and the fostering of a trustworthy academic community at McMaster University.

## Monica Bazina-Grolinger

As I submit my work, I, **Monica Bazina-Grolinger**, take full responsibility for the integrity of my work and promise to avoid any form of plagiarism, cheating, or dishonest behavior. This acknowledgment serves as a testament to my dedication to academic excellence and the fostering of a trustworthy academic community at McMaster University.

## Kate Min

As I submit my work, I, **Kate Min**, take full responsibility for the integrity of my work and promise to avoid any form of plagiarism, cheating, or dishonest behavior. This acknowledgment serves as a testament to my dedication to academic excellence and the fostering of a trustworthy academic community at McMaster University.

# Control Information

Version	Delivery		Feedback	
	<i>Deadline</i>	<i>Delivered</i>	<i>Received</i>	<i>Integrated</i>
V1	2023-10-15	2023-10-12	2023-10-24	2023-12-03
V2	2023-11-12	2023-11-12	2023-11-22	2023-12-05
V3	2023-12-06	2023-12-06		



# (G) Goals

## Control Information

Table 1. ACME RUN — Versioning Information — Goal Book

Section	Version	Lead	Delivered on	Reviewer	Approved on
<b>G.1</b>	1	AF	2023-09-20	MBG	2023-09-20
<b>G.2</b>	2	AF	2023-12-02	MBG	2023-12-02
<b>G.3</b>	2	KM	2023-12-03	AF	2023-12-04
<b>G.4</b>	2	AF	2023-12-02	MBG	2023-12-02
<b>G.5</b>	2	KM	2023-12-04	AF	2023-12-04
<b>G.6</b>	1	AF	2023-11-11	MBG	2023-11-11
<b>G.7</b>	2	MBG	2023-10-30	KM	2023-10-31

## (G.1) Context and Overall Objectives

In the era of health consciousness and escalating gym membership costs, ACME and HammerCorp Inc. recognize an unmet need for an engaging fitness solution in the North American market. The ACME RUN project is envisioned as a gamified mobile application, blending the thrill of gaming with physical activity. By incorporating curated safe running trails and integrating a heart rate monitor system, ACME RUN offers a personalized and dynamic workout experience for users. This initiative not only promotes physical well-being but also leverages local partnerships, such as the one with McMaster, to ensure safety and build community engagement. The overarching objective is to make fitness more accessible, enjoyable, and tailored to individual needs, while fostering community ties and ensuring user safety.

## (G.2) Current situation

ACME RUN is presently in its conceptual phase, aligning with the users' needs in the fitness market. The initiative is driven by a recognized gap in the market, emphasizing the desire for a gamified running application that can deliver personalized workouts at an affordable cost. Drawing inspiration from successful platforms such as "Zombies, Run!", ACME RUN seeks to stand out by integrating heart rate monitors and offering a tailor-made training experience. Exploration of partnerships with local authorities, particularly at McMaster University's campus, is underway to establish curated trails for safety, extending from the bayfront to the Dundas Valley Conservation Area. Despite this progress, there is currently no functional prototype, leaving the app's feasibility and user engagement untested. The emphasis is on addressing the users' current needs and motivations in the fitness domain, setting the foundation for the development of ACME RUN.

## (G.3) Expected Benefits

The launch of ACME RUN is expected to give rise to a new exercise routine for those interested in health and fitness. The main benefit is the increased interest in physical activity within the local community in which the app is deployed. Introducing dynamic "gamified" challenges to encourage physical exercise and quick decision-making will engage users, providing a fresh alternative to gym membership subscriptions and other competing apps. Additionally, by connecting a **Heart Rate Monitor** (HRM), the user can receive more personalized workouts and challenges, further increasing the user's satisfaction and experience. This ultimately aligns with the end goal to provide users with an entertaining outdoor fitness regimen catered to their individual needs.

It is anticipated that ACME RUN will improve the usage of natural parks, trails, and local shops by users. The introduction of these curated outdoor spaces can foster a sense of community amongst users, as well as a reinforced level of safety within said community. Trails near urban areas can present local shops as strategic **sheltering**, and such businesses can promote their brand, see an increase in their overall foot traffic, and potentially maximize their operational revenue through their participation in ACME RUN. Furthermore, recognizing a majority of trails are typically located in woody areas where local shops may not be found, the integration of local parks, trails, and other public spaces into the game system can provide a more efficient and consistent way to maintain a safe state of these spaces through user reports and logs. Within a fitness industry that is dominated by expensive personalized training programs, the launch of this app aims to redefine personalized workouts in a manner that is safe, affordable, and engaging.

The goal model diagram for this app is outlined below:

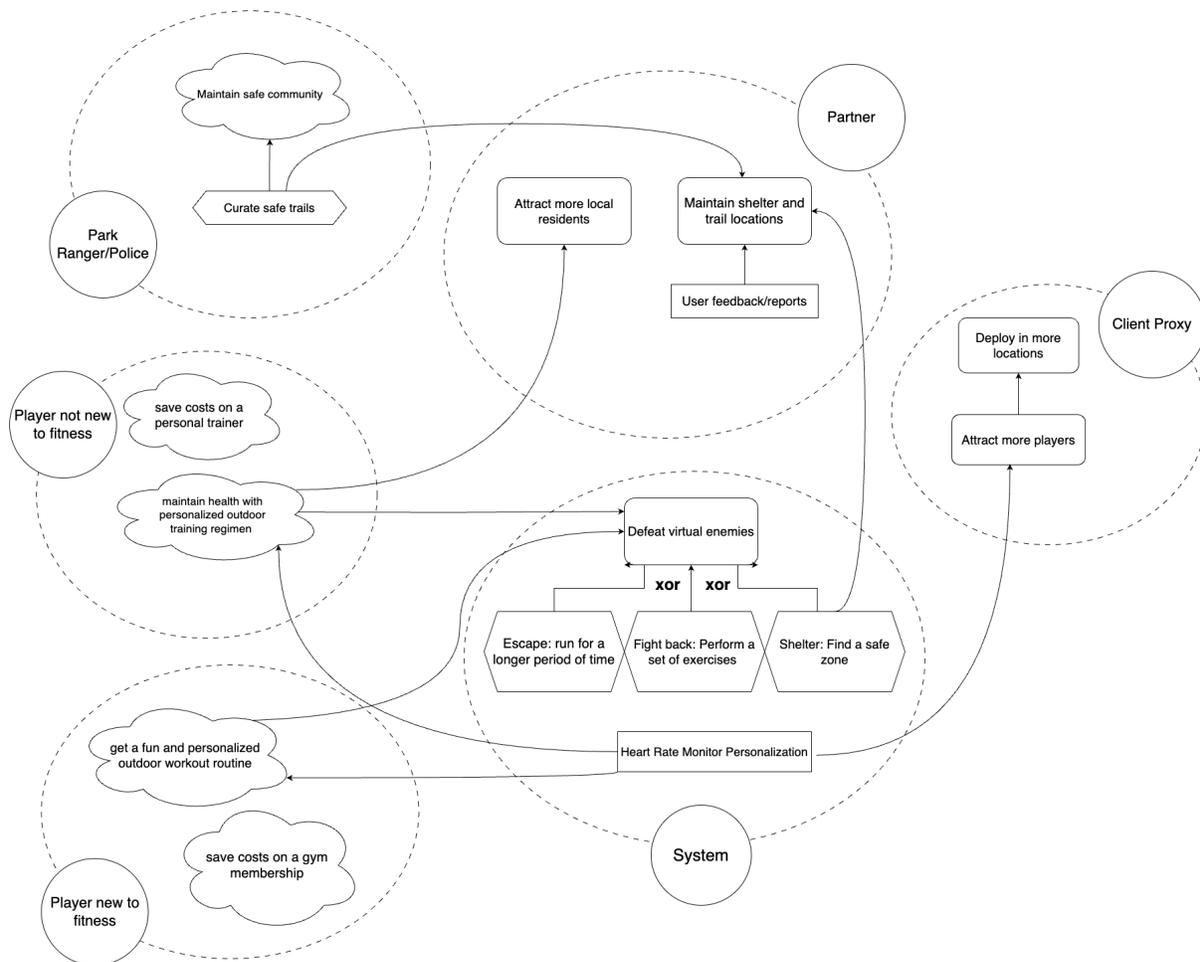


Figure 1. Goal model diagram for ACME RUN

## (G.4) Functionality overview

The high level functional requirements for the application based on client input can be defined as follows in no particular order:

- **Geographical Zone Integration:** The application must allow players to register and select a given geographical zone for running, with trails curated for safety in partnership with local authorities. This ensures players have a variety of paths to explore and use within the game.
- **HRM Connectivity:** The game must offer seamless integration with Heart Rate Monitors (HRM), collecting real-time heart rate information, and adapting the gameplay accordingly to align with the player's physical fitness levels and goals.
- **Gamified Running Events:** As players run, they should encounter virtual enemies and must decide on a course of action - sheltering, escaping, or fighting back. These events should be dynamically generated based on the player's profile and running statistics.
- **Profile Management & Badges:** Players should be able to access and manage their profiles, tracking their progress, achievements, and earning badges based on challenges within the app, ensuring continued engagement.

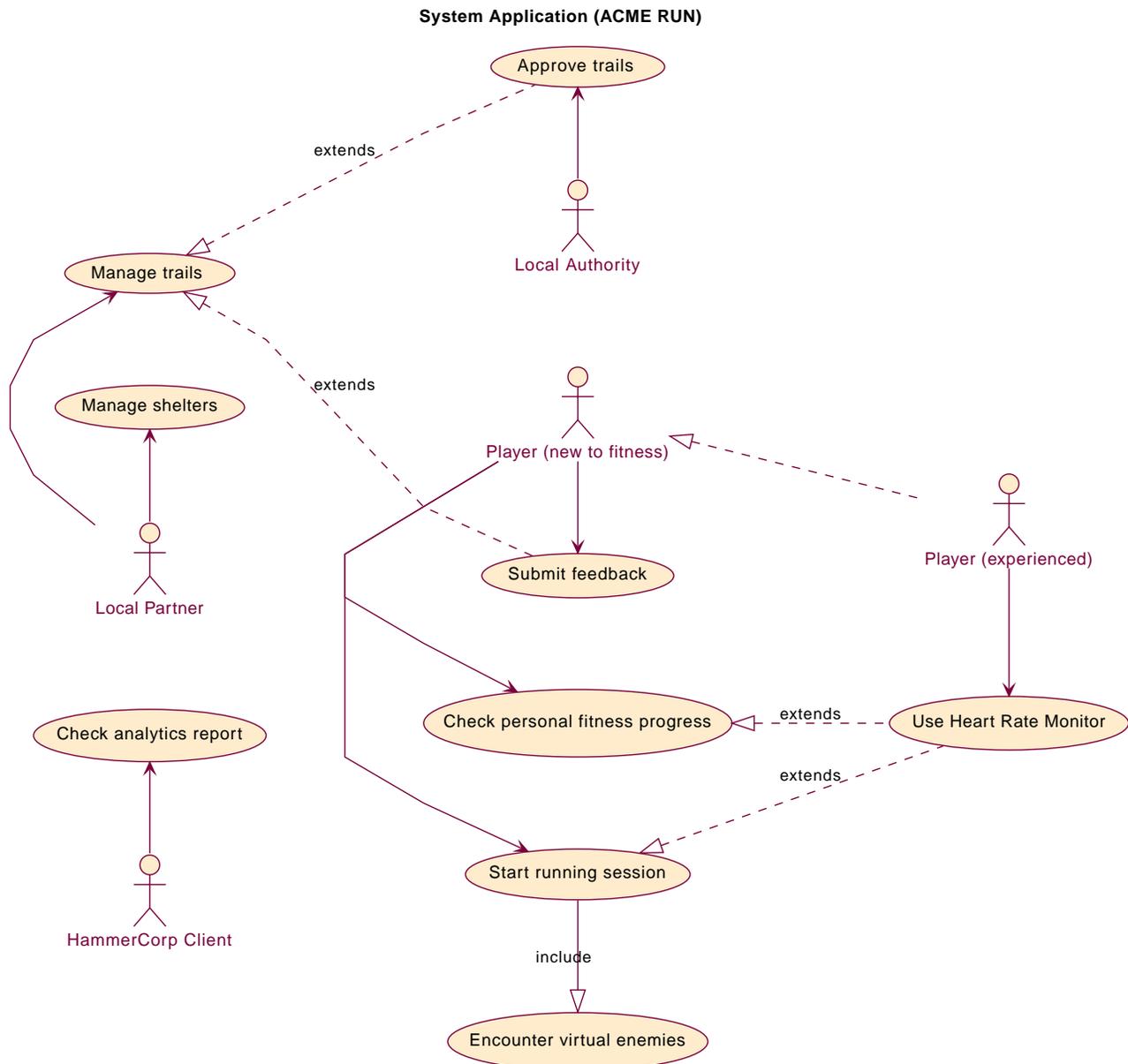
Among these functional requirements, the two of utmost importance are HRM Connectivity and Gamified Running Events. HRM Connectivity stands out because it personalizes the experience, ensuring that players can work out within safe limits while achieving their fitness objectives. Such personalization can be a significant USP (Unique Selling Proposition) in the market, offering a tangible health benefit to users. The Gamified Running Events, on the other hand, are the core of the application's engagement strategy. By making the running experience more interactive and challenging, it differentiates ACME RUN from standard running apps, potentially attracting and retaining more users, thereby directly impacting the application's success and revenue.

Some high level non-functional requirements for the application can be defined as follows in no particular order:

- **User Safety Assurance:** As the app involves physical activity, it's crucial that it operates without causing unnecessary distractions or disruptions, ensuring the player's safety at all times, especially when running in real-world scenarios.
- **Application Scalability:** Given the potential for deployment in various North American cities post the pilot phase, the application should be built to easily scale, accommodating a large number of users and integrating new geographic zones seamlessly.

The relevance of these non-functional requirements is paramount. User Safety Assurance is crucial not only from a moral standpoint but also from a business perspective. Any incidents caused due to app distractions could lead to legal implications and bad publicity, potentially harming the brand's image and trustworthiness. Application Scalability, meanwhile, directly affects business growth. As the app expands to new cities and acquires more users, the system's inability to scale can lead to downtimes, poor user experience, and, ultimately, user attrition. Thus, ensuring scalability is an investment in the app's future success.

## **(G.5) High-level usage scenarios**



*Figure 2. High Level use cases diagram*

The presented use case diagram illustrates an overview of the expected interactions between our direct stakeholders and the ACME RUN app. Below are the details for the key usage patterns:

- **UC1: Start a running session.**

1. Player opens the app to begin a new running session.
2. Player accesses nearby trails from list or from map.
3. Player browses trail options and filters through categorizations (i.e., level of difficulty, distance, trail length).
4. Player taps on a trail to review specific information and confirm selection
5. App prompts Player to connect their personal HRM device, if applicable.

6. Player taps start button to begin running session.

• **UC2: Encounter a virtual enemy.**

1. During the active session, the system continuously monitors player's location using location services.
2. The system pushes a notification when a virtual enemy appears along player's route.
3. Player can verbally say or tap the action event they want (i.e., fight back, escape, or seek shelter).
4. The system pushes another notification displaying the event the player must execute.
5. Player executes the specified event.
6. The system pushes another notification when the event is completed.
7. Player continues along route until next virtual enemy appears.

• **UC3: Check personal fitness progress and achievements.**

1. The system collects player's performance, decisions and workout metrics throughout the active running session.
2. Player completes route and taps "finish session" button in the app.
3. Player accesses session history to check summary of statistics.
4. The system displays a stats breakdown of the recent session.
5. The system displays new achievements/badges earned and prompts player to add achievements to profile or share to social media.
6. The system prompts player to set new fitness goals or personalization preferences based on session feedback.

• **UC4: Update information about a given trail.**

1. Local authority sends approved, safe trails to local partner by traditional means (e.g. e-mail).
2. Local partner opens the app and logs into admin portal.
3. Local partner navigates to reports menu within the app.
4. Local partner reviews list of feedback reports about trail conditions from players.
5. Local partner updates trail information within the app as needed.
6. The system pushes a notification to players when a trail's status is updated.

• **UC5: Check analytics report.**

1. The system collects data related to app performance and user engagement using various metrics.
2. HammerCorp client opens the app and logs into admin portal to check app performance.
3. Client navigates to analytics and insights menu within app.
4. The system generates an analytics report based on the collected data and displays a preview.
5. The system prompts the client to save the report or archive it.

## (G.6) Limitations and Exclusions

Below is a list of limitations that the system will not do:

- System will not be responsible for monitoring or controlling the actual physical safety of players during their running sessions. It does not have the capability to intervene in real-world emergencies or incidents, and players are solely responsible for their well-being while using the application.
- System will not guarantee compatibility with all models of Heart Rate Monitors (HRM). The system's adaptation to heart rate data relies on the functionality and accuracy of the connected HRM, and any issues arising from HRM compatibility or performance are beyond the scope of the system's responsibility.
- System will not integrate with third-party fitness apps or platforms for data sharing or collaborative features.

## (G.7) Stakeholders and requirements sources

Table 2. Stakeholders for ACME RUN

Stakeholder		Persona	Category
Players	New to fitness	Charlie	Direct
	Not new to fitness	Blake	Direct
HammerCorp Inc.	HammerCorp Inc. - Representative	Avery	Direct
Local authorities	Parks and Recreation Department - Park Ranger/Police	Jordan	Direct
McMaster University	McMaster University - IT Department	Ezra	Direct
HRM manufacturer			Indirect
Partnering shops			Indirect

### Direct Stakeholders

#### Players

The Players stakeholder in the ACME RUN project is the end user. They register in a specific geographical region, use the application while running, including its features sheltering, escaping, and fighting virtual enemies. They are the entity the application is ultimately designed for. These individuals heavily impact the overarching objective to make fitness more accessible and enjoyable. They are crucial to achieving this goal

#### Charlie (new to fitness)

Charlie is a 28-year-old high school teacher in Ancaster, ON. Charlie has one young child with their

partner. The whole family enjoys spending time together being active. Activities the family likes to do include: biking, soccer, and basketball. Charlie is searching for fun exercise activities they can do as well as their family.

### **Blake (not new to fitness)**

Blake is 21-year-old software engineering student at McMaster University. They live in the Westdale area (approximately a 5-minute walk to the school). During their free time, Blake goes for a run around the Westdale area or uses a treadmill at the Crunch fitness near their house, where they work part time. Their job is as a personal trainer that specializes in training long distance runners.

### **HammerCorp Inc.**

The Client stakeholder, HammerCorp Inc., has commissioned ACME to create this application. They have recognized an unmet need for an engaging fitness solution in the North American market. They have outlined what they envisioned for this application, summarized in G.1. HammerCorp Inc. has a vested interest in the success of this project.

### **Avery (HammerCorp Inc. Representative)**

When HammerCorp Inc. initially started conducting business, Avery purchased shares in the company. Shortly after, they went through the necessary steps to become a representative. Now, one of Avery's responsibilities is to relay requirements information to partners.

### **McMaster University**

The McMaster stakeholder will locally operate this game. They are responsible for defining the trails and overseeing shelter locations. Therefore, the ACME RUN project leverages local partners (McMaster University) to optimize the success of the application. The pilot partner is McMaster University, and they are essential in testing the app's features in a real-world setting. Having the application run locally could create some community ties. Thus, this stakeholder is integral to the overall objective of the application.

### **Ezra (McMaster IT Department)**

Ezra is a long-time member of the McMaster IT Department. They were hired for this position after completing their undergraduate degree at 22 years old and are now 31 years old. Their daily responsibilities include overseeing and maintaining McMaster's technological infrastructure. Therefore, Ezra is very familiar with the technological infrastructure at McMaster, knowing what the University can and cannot support.

### **Local Authorities**

The Local Authorities stakeholder is responsible for planning, approving, and maintaining running trails as well as ensuring player safety. The agency that governs trails in the partner's location is the authority that affects the application the most (in Hamilton, this is the Parks and Recreation Department). Two main individuals in this agency that will be involved are the Park Rangers and Trail Supervisors. As stated in G.1, a key feature in our overarching objective is to ensure player safety. Since the local authorities ensure safety, they are integral to the success of this application.

## Jordan (police officer)

Jordan is a recent graduate from Fanshawe College's Police Foundations program, located in London, ON. They have recently secured a position within the Hamilton Parks and Recreation Department as a Park Ranger. Jordan's main responsibilities include ensuring the safety and maintenance of Hamilton's parks and wildlife areas. Therefore, their daily work activities include enforcing rules, regulations, and laws at the Hamilton parks and trails.

## Indirect Stakeholders

- **HRM manufacturer:** Heart Rate Monitor manufacturers produce devices that are used during physical activity to measure the user's heart rate. This application supports connecting an HRM. Since the application supports integrating an HRM, ACME RUN needs to comply with constraints set by the manufacturer. This is integral to tailoring the application to the user.
- **Partnering shops:** Partnering shops are stores that collaborate with the application to provide shelter to users. These shops serve as shelter spots in ACME RUN; therefore, they are affected by the game's success. If the game achieves great success, the partnering shops will draw more people to their businesses, thus increasing the number of potential customers.

## Requirements Sources

- **Existing documents (maps):** Specifically maps on a digital application, are digital displays of geographical areas. This is need in ACME RUN to develop curated running trails and integrating them into the application. Thus the ACME RUN project may use some location-based services, and it will need to adhere to the requirements of this service. This will aid in support our overarching objective as outlined in G.1.
- **Similar system (Zombies, Run!):** The ACME RUN game is inspired by an existing game, "Zombies, Run!". "Zombies, Run!" is a mobile fitness game where users have to run to survive a zombie apocalypse. As stated in G.2, ACME RUN will take inspiration from the existing platform "Zombies, Run!". [https://en.wikipedia.org/wiki/Zombies,\\_Run!](https://en.wikipedia.org/wiki/Zombies,_Run!)
- **Existing documents (exercise research):** Since the ACME RUN game determines events based on the user's heart rate, ACME will be utilizing research resources from the following sources:

Laursen, Paul B., and David G. Jenkins. "The scientific basis for high-intensity interval training." *Sports medicine* 32.1 (2002): 53-73

Aubert, André E., Bert Seps, and Frank Beckers. "Heart rate variability in athletes." *Sports medicine* 33 (2003): 889-919.

# (E) Environment

## Control Information

Table 3. ACME RUN — Versioning Information — Environment Book

Section	Version	Lead	Delivered	Reviewer	Approved
<b>E.1</b>	4	MBG	2023-11-22	KM	2023-11-26
<b>E.2</b>	1	MBG	2023-10-24	KM	2023-10-27
<b>E.3</b>	2	KM	2023-12-04	AF	2023-12-04
<b>E.4</b>	2	AF	2023-12-02	MBG	2023-12-02
<b>E.5</b>	2	KM	2023-12-03	AF	2023-12-04
<b>E.6</b>	1	AF	2023-10-02	MBG	2023-10-10

## (E.1) Glossary

### (E.1.1) Vocabulary

#### Run Session

In the context of this document, a "run session" refers to details and available options accessible to a player while using the application during a run.

#### Run Event

In the context of this document, a "run event" refers to one of the options available to users to stay alive in the game. These options include sheltering, escaping, and fighting back.

#### Sheltering

In the context of this document, "sheltering" refers to a safe zone available on a trail where players can rest. Partnering shops would be an example of a possible safe zone. This is always an option unless the player activates "hardcore" mode.

#### Partnering Shops

Partnering shops are local establishments that provide users with shelter. Examples of partnering shops include cafes, grocery stores, and retail stores.

#### Escaping

In the context of this document, "escaping" refers to evading an attacker. To do so, the user must run faster for a duration of time determined by their previous statistics.

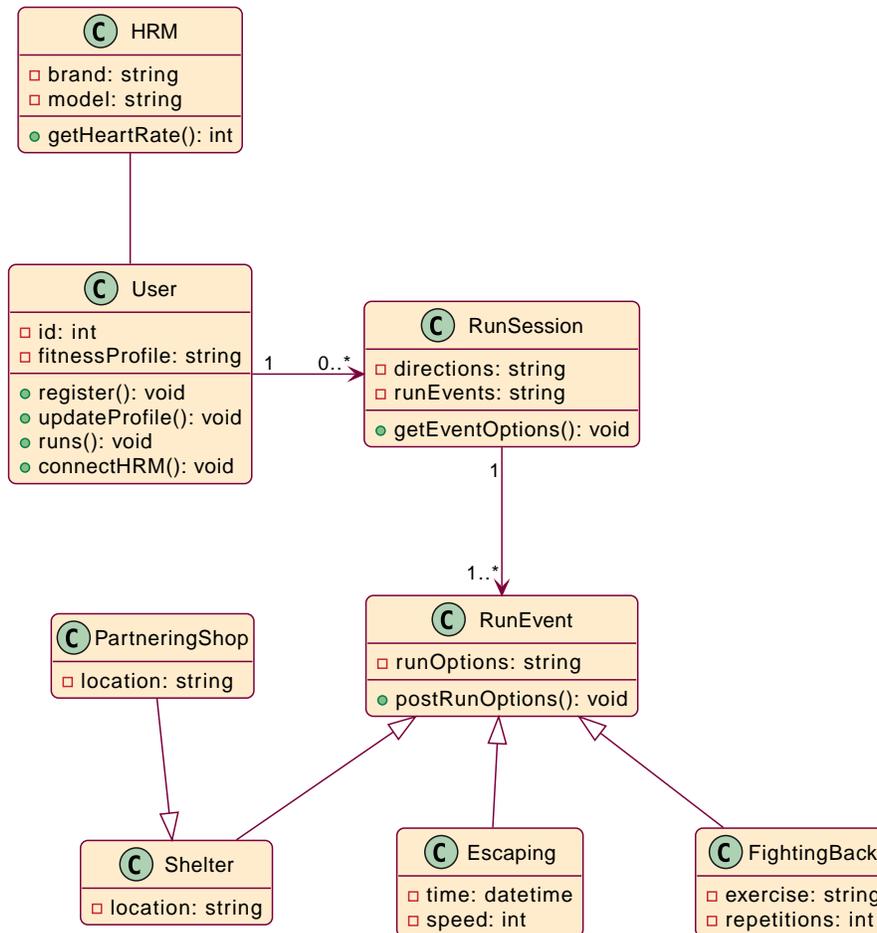
## Fighting Back

In the context of this document, "fighting Back" refers to a player's ability to fight off attackers by completing a short set of body-weight exercises (e.g. push ups).

## Heart Rate Monitor (HRM)

HRM is the abbreviation for Heart Rate Monitor. These are wearable devices that measure and display a user's heart rate or pulse. Popular HRMs include FitBit and the Apple Watch.

### (E.1.2) Domain Model



## (E.2) Components

The system interacts with four external components:

- **Heart Rate Monitor (HRM) Devices:** Heart rate monitor devices, or HRM devices, are external devices that provide the system with information about the users heart rate data. This data is used in escape and fight events to keep the user's heart rate in the specific zone. Thus the HRM provides vital information for the system.
- **Maps API:** Provides location information about a region, particularly important for determining the

locations of trails and shelters. This information helps users have a clear understand of the route they are taking or possible routes they can take.

- **Local Authorities:** External entity that is responsible for ensuring user safety on trails. This information is vital for the applications guarantee of user safety.
- **Local Partners:** Local partners are responsible for the local operation of the application in a certain region. They possess knowledge of the region's trails and shops, which are used to define trails and shelters respectively within the application. Thus, local partners affect and are effected by the application.

## (E.3) Constraints

The following are constraints that are imposed on the application:

- **Localization of trails:** The availability and approval of usable trails is limited to those provided by the local partner, as system relies on collaboration with local partners and authorities to curate such trails.
- **Mobile support:** The system must work on standard mobile devices (Android, Apple) to support core functionality in regards to outdoor gameplay. It thereby follows that the system must adhere to app store policies.
- **Geographical limitations:** The system will operate only within the boundaries of the specific geographical region(s) in which it is deployed.

## (E.4) Assumptions

Some assumptions made can be found below:

1. **Local Partnership Availability:** It is assumed that local partners, such as McMaster, will be readily available and willing to cooperate in identifying safe trails and shelter locations. This partnership is essential to ensure the safety and effectiveness of the ACME RUN app in various locations.
2. **User's Honesty in Physical Responses:** Given that the application can provide options like "fight back" which involves users doing physical exercises like push-ups, we assume that users are honest and accurate in inputting or confirming their physical responses. Without a method of verifying this, the app's efficacy in training could be compromised if users bypass these challenges.

## (E.5) Effects

The following are some effects that the app may induce:

- **Increase in foot traffic near trails and local shops:** The integration of local trails and shops into the gameplay will encourage users to seek them out while the app is in use. This will result in an increase of the daily foot traffic of these spaces and promote the appeal of natural trails and local businesses. However, this can also unintentionally give rise to harmful environmental impacts, such as littering

and/or rapid deterioration of trail conditions.

- Increase in users' physical activity and overall health: As the objective of the app is to create personalized workout routines, successful deployment of the app will see more people being active within the community and contribute to their overall health and well-being.
- Increase in events within the community: This is possible through the app's inclusion of global challenge events to encourage consistent activity. This will result in more outdoor events related to the in-app challenges and further enhance both user and community engagement.

## (E.6) Invariants

Some Invariants of this system can be defined as follows:

- ACME RUN's trails will always be within predefined geographical zones that have been curated and approved by local authorities for runner safety.
- The maximum heart rate calculation for players will always start with the formula:  $220 - \text{age}$ , even though it may be adjusted later based on additional information.
- The ACME RUN application will always offer the three core player interactions in response to virtual enemies: Sheltering (unless the player is in "hardcore" mode), Escaping, and Fighting Back.

# (S) System

## Control Information

Table 4. ACME RUN — Versioning Information — System Book

Section	Version	Lead	Delivered	Reviewer	Approved
<b>S.1</b>	2	MBG	2023-11-26	KM	2023-11-28
<b>S.2</b>	2	MBG	2023-11-28	KM	2023-11-29
<b>S.3</b>	2	AF	2023-12-02	MBG	2023-12-02
<b>S.4</b>	2	KM	2023-12-04	AF	2023-12-04
<b>S.5</b>	1	MBG	2023-11-29	KM	2023-12-03
<b>S.6</b>	2	MBG	2023-11-29	KM	2023-11-29
	3	KM	2023-12-06	AF	2023-12-06

## (S.1) Components

The system relies on the following components:

- **User Manager:** Relevant for the user interaction with the system. The component in charge of managing the user database (Manage Users). Its goal is to allow the user to create and manage accounts in a seamless manner.
- **Trail Manager:** Relevant for users running experience. The component that manages the trail information database (Manage Trails). Its goal is to provide users access to a variety of running trails, that can be tailored to user preferences.
- **Run Manager:** Relevant for real-time tracking of runs and analytics. Its goal is to gather and process data during a users run. Essential for providing the user with insights on their progress and performance.
- **Event Manager:** Relevant for the overarching idea of the application. Its goal is to provide and manage events that help keep the user alive. This includes shelter, escape, and fight back information. Gives the user what options are available in addition to specifics (e.g. partnering shop locations). Enriching the user experience.

In addition to these components, the system will interact with two external components Maps API and a HRM. Maps API is used for trail directions, and HRM are external devices that will be used to record users heart rate (see E.2 for more details).

Users will use ACME RUN-App (a mobile application) to interact with the system.



distance. (F222)

3. **Search:** Trail manager shall allow users to search the trail database. (F223)
4. **Trail categorization:** Trail manager shall be able to filter/sort trails by difficulty. (F224)

### (S.2.3) Run Manager

#### Functional Requirements

1. **Run statistics:** Run manager shall collect run information including distance, duration, and time. (F231)
2. **Integrate Maps:** Run manager shall record and display the users location during a run in relation to a map of a specific trail. (F232)
3. **Integrate HRM:** Run manager shall collect and display heart rate information collected from a connected HRM. (F233)
4. **Provide event options:** Run manager shall notify users when they encounter a virtual enemies on a trail. (F234)

### (S.2.4) Event Manager

#### Functional Requirements

1. **Event options:** Event manager shall determine what event options are available to a user and at what time. (F241)
2. **Include shelter information:** Event manager shall store the name and location of partnering shops (shelter locations). (F242)
3. **Include escape information:** Event manager shall provide a period of time the user has to run faster to escape a virtual enemy. (F243)
4. **Include fight back information:** Event manager shall store body-weight exercise descriptions and user specific repetition numbers. (F244)

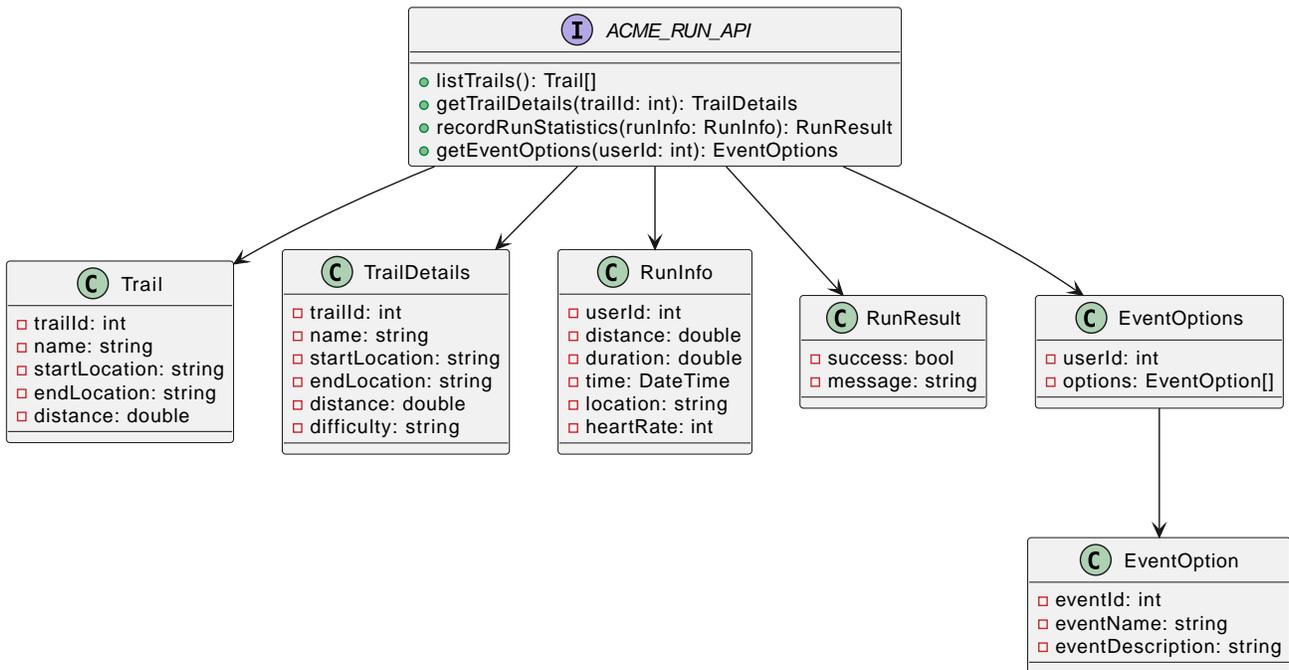
### (S.2.5) Non-Functional Requirements

1. **Search trail response:** The application shall provide minimal delay in response time to a user request. (NF251)
2. **Accurate trail data:** The application shall keep up-to-date information on trails. (NF252)

## (S.3) Interfaces

### (S.3.1) APIs

The ACME RUN system exposes several APIs to the external world to facilitate integration and interaction. The main API is the ACME\_RUN\_API which is the main external interface providing functionality. Manager involvement in the API is not required, however, they could potentially oversee access, security, and overall system functionality. The primary external interfaces are as follows:



- ACME\_RUN\_API is the main external interface providing functionality related to trails, runs, and events.
- The Trail and TrailDetails classes represent the structure of trail information.
- RunInfo encapsulates the details of a running session, while RunResult represents the outcome of recording run statistics.
- EventOptions and EventOption are used to retrieve available event options for a user.

### (S.3.2) Wireframe Mockups

#### (S.3.2.1) IUser Profile (User Interface)

The user interface for user profile is designed to provide users with quick access to upcoming challenges, options, and user information. It includes visual elements and interactions suitable for mobile devices.

Wireframe Mockup:

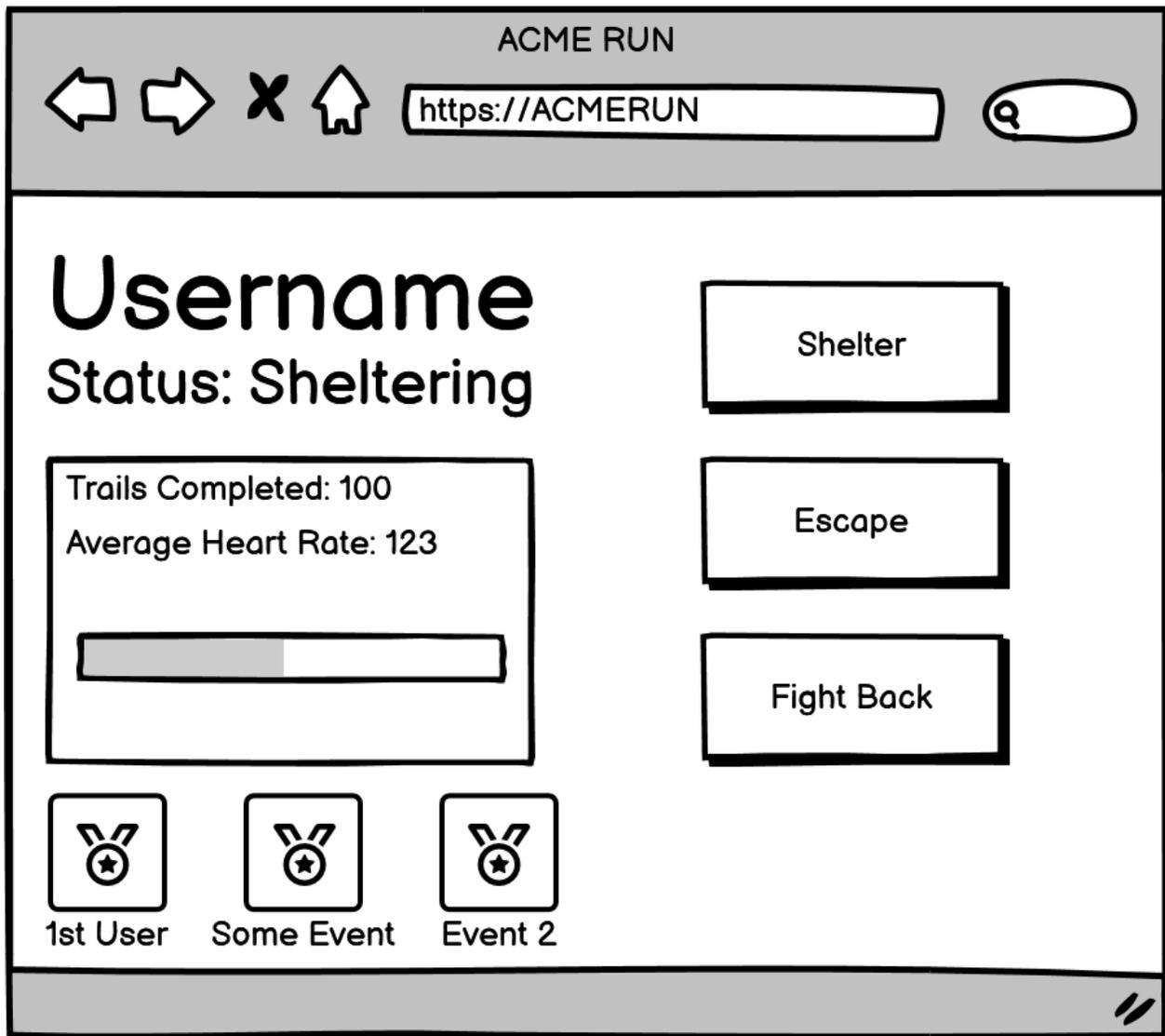


Figure 3. User Interface Wireframe Mockup

Notes:

- The UI displays upcoming events, user information, and status allowing users to see available options such as sheltering, escaping, or fighting back.
- Visual elements, such as icons and progress bars, enhance user experience.

### (S.3.2.2) Trail Details (User Interface)

The user interface for trail details aims to provide users with comprehensive information about a selected trail, including its difficulty level.

Wireframe Mockup:

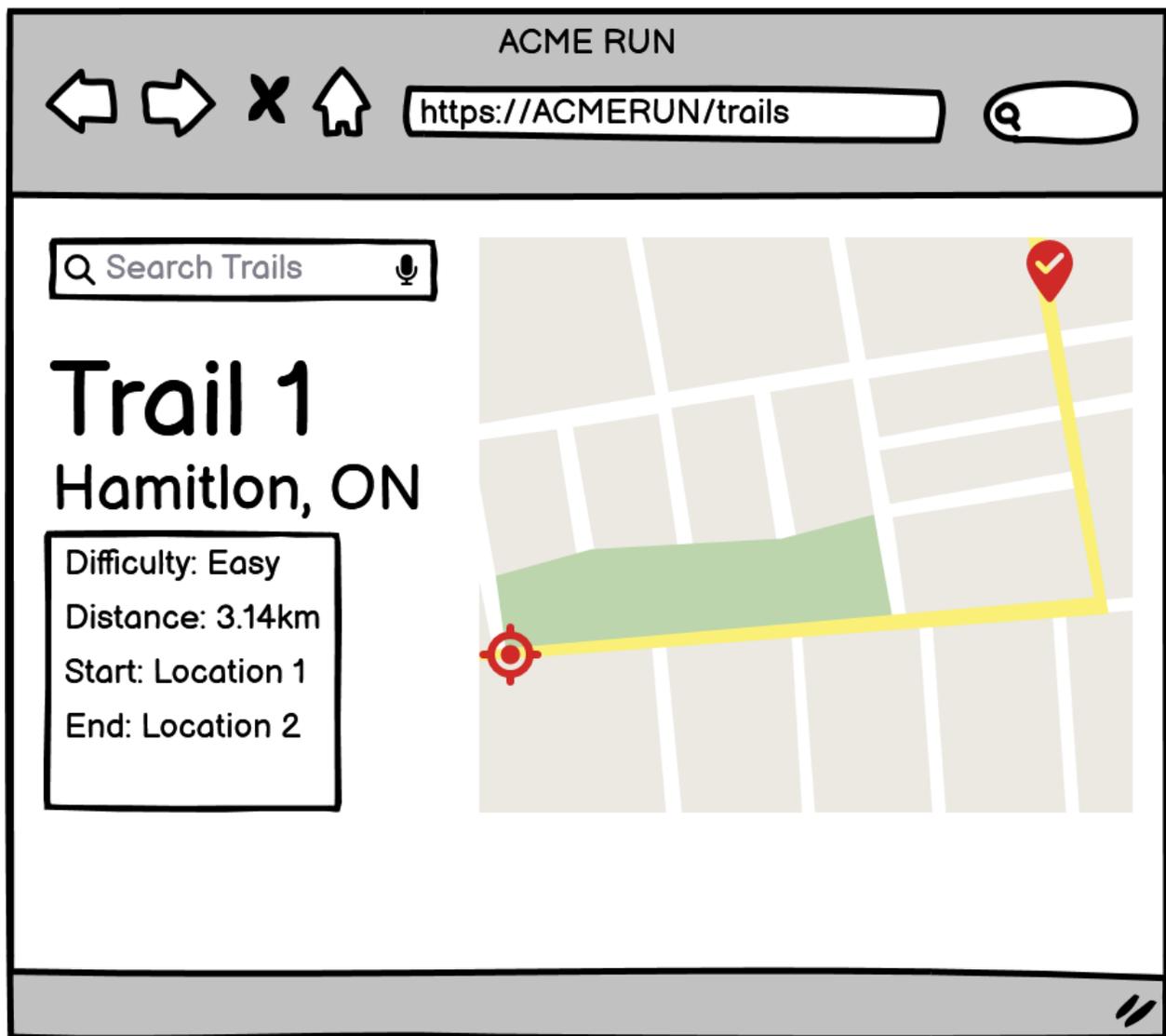


Figure 4. Trail Details Wireframe Mockup

Notes:

- The UI displays detailed information about a selected trail, including its name, start and end locations, distance, and difficulty level.
- Visual elements and clear organization contribute to a user-friendly experience.

## (S.4) Detailed usage scenarios

### (S.4.1) Player starting a running session

- **Use Case:** UC1
- **Primary Actor:** Player
- **Precondition:** Player has internet access and the app downloaded

- **Trigger:** Player wants to go on a run in a nearby trail
- **Main Success Scenario:**
  - 1. Player opens the ACME RUN app on their mobile device.
  - 2. App shows nearby available trails that Player can choose from.
  - 3. Player chooses a trail and chooses a difficulty mode (regular or "hardcore").
  - 4. Player connects HRM device.
  - 5. Player begins the running session.
  - 6. The app tracks player's statistics.
- **Secondary Scenarios:**
  - 1.1. Player registers for a new account.
  - 1.2. Player logs into existing account if not already logged in.
  - 2.1. Player activates specific filters to narrow down list of trails.
  - 3.1. Player goes back and chooses a different trail.
  - 4.1. Player continues without a personal HRM.
  - 4.2. Player purchases the in-app HRM device.
- **Success Postcondition:** Player successfully starts a running session on the trail of their choice.

This scenario is important as it is the fundamental interaction that is expected between the app and the users. Additionally, the client may achieve a profit goal through the establishment of an in-app purchase.

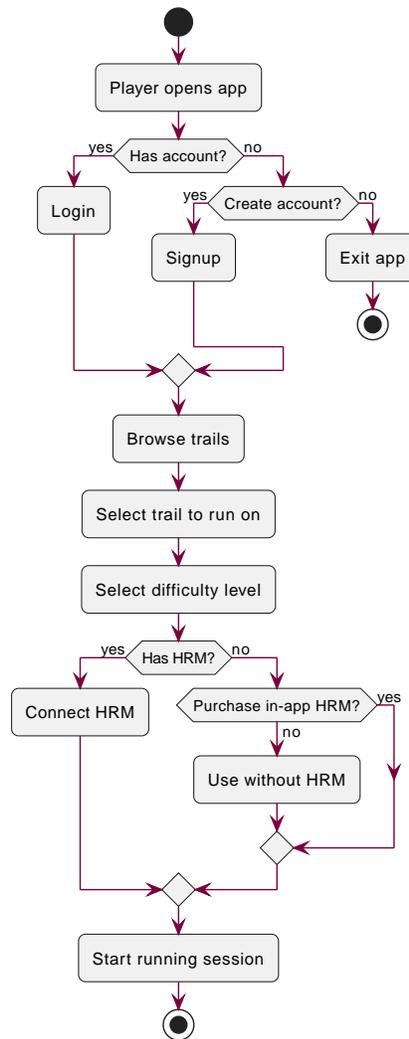


Figure 5. Activity diagram for 'Starting a running session'

The activity diagram helps illustrate the expected behaviour of the usage scenario and potential divergences when different decisions are made. This is particularly important as it shows the different options for the HRM feature which is critical to the personalization aspect of the app.

### (S.4.2) Player encountering virtual enemies during running session

- **Use Case:** UC2
- **Primary Actor:** Player
- **Precondition:** Player is currently in an active running session and on the chosen trail.
- **Trigger:** A virtual enemy appears along the trail.
- **Main Success Scenario:**
  - 1. The system pushes a notification about a virtual enemy appearing.
  - 2. Player tells the system on the chosen action event (fight back, escape, shelter).

- 3. Player executes the action event. If "fight back" was chosen, player executes a set of exercises. If "escape" was chosen, player runs faster for an extended period of time. If "shelter" was chosen, player finds the designated nearby shelter.
- 4. The system records the Player's decision and HRM information during execution.
- 5. Player continues along the route until the next enemy appears.

- **Secondary Scenarios:**

- 2.1. Player does not choose an event. Notification times out and event resets.
- 3.1-4.1. Player does not execute the event after selecting one. System records the behaviour and event resets.
- 3.2. Player wants to change the event to execute. Go back.
- 4.1. System does not record Player's data due to no connected HRM.
- 5.1. Player diverges from the current trail. System updates location.

- **Success Postcondition:** Player successfully engages in game strategy and improve personalization of training plan.

This scenario is important as it is the core gameplay that is expected to be carried out by the player during an active session. The addition of events occurring throughout the player's session intends to maintain engagement and adapt to the player's decision-making behaviours. This scenario highlights the different directions a user may take in regards to the events.

### (S.4.3) Player checking their stats and personalized training plan

- **Use Case:** UC3
- **Primary Actor:** Player
- **Precondition:** Player is logged into their account and has previously completed a session.
- **Trigger:** Player wants to check their fitness progress, goals, and achievements.
- **Main Success Scenario:**
  - 1. Player completes a running session and reopens ACME RUN.
  - 2. The app shows a breakdown of the player's statistics and decisions made during the session.
  - 3. The app shows achievements that were earned, if any, and prompts player to share them on profile.
  - 4. Player can set new fitness goals or adjust personalized training plan.
- **Secondary Scenario:**
  - 1.1. Player opens the ACME RUN app to check historical stats.
  - 3.1. Player skips sharing achievements on profile.
  - 4.1. Player does not want to change anything.

- **Success Postcondition:** Player gain insight into their fitness progress and the personalizations made for them, as well as be able to set new goals or amend existing ones.

This scenario is important as the goal of the app is to provide personalized workout routines for a given user. It covers how the player will be able to view their progress and customize the preferences that the app suggests based on their session data.

#### (S.4.4) Local partner using app to update trail information

- **Use Case:** UC4
- **Primary Actor:** Local partner
- **Secondary Actors:** Local authority, player
- **Precondition:** Local authority has approved trails and local partner has the app downloaded.
- **Trigger:** Local partner needs to update a trail or shelter with latest information
- **Main Success Scenario:**
  - 1. Local partner has approved trails from local authority.
  - 2. Local partner launches ACME Run app.
  - 3. Local partner navigates to the reports section of the app.
  - 4. Local partner reviews feedback reports about trail conditions.
  - 5. Local partner navigates to the trail database section of the app.
  - 6. Local partner updates a trail information. The app displays a confirmation of changes.
  - 7. Local partner confirms and submits update.
- **Secondary Scenario:**
  - 2.1. Local partner receives notification from app about feedback reports and launches the app.
  - 4.1. Local partner receives no feedback reports. Skip.
  - 6.1. Local partner updates the wrong trail/shelter. Redo update.
  - 6.2. Local partner updates the trail/shelter with the wrong information. Redo update.
- **Success Postcondition:** The app is up-to-date with correct information about trail and shelter conditions.

This scenario is important as it is the key method for keeping trail conditions safe and up-to-date for players. It demonstrates how the local partner may easily update (add, remove, modify) trails, as well as show the need for user feedback to successfully complete their task.

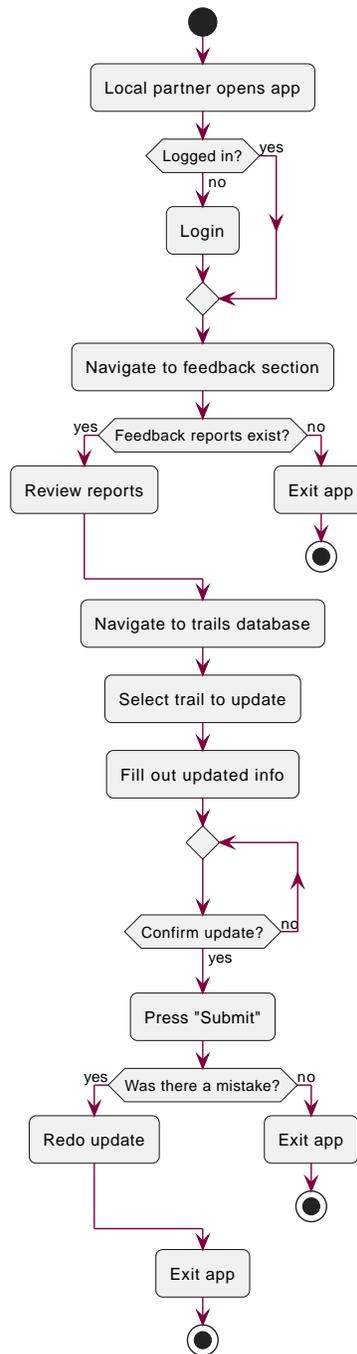


Figure 6. Activity diagram for 'Updating trail information'

The activity diagram for this usage scenario breaks down the outcomes of the different decisions that can be made by the local partner when managing the available trails and shelters for players to use. In addition, it helps to visualize the usability of feedback reports.

### (S.4.5) Player submitting feedback

- Use Case: UC5
- Primary Actor: Player

- **Secondary Actors:** Local partner, client
- **Precondition:** Player is logged in on the app and has internet access. Local partner and client have the app downloaded.
- **Trigger:** Player wants to provide feedback on trail conditions.
- **Main Success Scenario:**
  - 1. Player is in an active session and experiences undesirable trail conditions.
  - 2. Player opens ACME RUN app and navigates to feedback section on app.
  - 3. Player selects the trail that they want to provide feedback on.
  - 4. The app displays a feedback form in which Player can report specific trail conditions.
  - 5. Player submits feedback form.
  - 6. The app stores Player’s feedback and transmits it to the reports section for local partner.
- **Secondary Scenario:**
  - 1.1. Player is not in an active session
  - 3.2. Player does not select a trail. Exit.
  - 4.2. Player does not report specific conditions. Form is saved for later.
  - 5.2. Player does not or forgets to press submit. Form is saved for later.
- **Success Postcondition:** Player successfully submits feedback for local partner to review.

This scenario is important as the local partner and client will rely on user feedback reports to update trail conditions in real-time, improve app functionality, and review user engagement. The client can use the feedback reports to gauge usage activity and generate an analytics report as mentioned in UC6. This also ties in with (S.4.4) in which local partners can rely on user feedback to update availability of safe trails.

## (S.5) Prioritization

### Priority Table for User Manager

Requirement ID	Description	Classification	Reasoning
[F211]	Register service	Must have	Essential to identify a user and personalize the application experience.
[F212]	Sign in	Must have	Important for secure access to user specific information.

Requirement ID	Description	Classification	Reasoning
[F213]	Sign out	Should have	Important for user security, however individuals usually remain signed in after registering.
[F214]	Reset password	Should have	Necessary for account security and user access, however once they sign up they are unlikely to log out.

## Priority Table for Trail Manager

Requirement ID	Description	Classification	Reasoning
[F221]	Trail database	Must have	Essential for the applications goal, trail data needs to be stored so it is able to be displayed to the user.
[F222]	Trail information	Must have	Essential for the applications goal, trail data needs to be stored so the user can choose a trail.
[F223]	Search	Should have	Improves usability and user experience but not essential to functionality or the goal of the application.
[F224]	Trail categorization	Could have	Adds value to user experience but not essential to functionality or the goal of the application.

## Priority Table for Run Manager

Requirement ID	Description	Classification	Reasoning
[F231]	Run statistics	Must have	Vital data to be used for the fitness aspect of the game.
[F232]	Integrate maps	Should have	Enhance the user experience by allowing them to see their real-time position on the trail map. However, the user can be provided with a map without real-time updates, making this requirement not essential for the core functionality of the application.
[F233]	Integrate HRM	Should have	Not all users will want to connect a HRM, thus this can be a feature that is added later on.
[F234]	Provide event options	Must have	Part of the applications core functionality.

## Priority Table for Event Manager

Requirement ID	Description	Classification	Reasoning
[F241]	Event options	Must have	Essential to the applications core functionality of providing run events to the user.
[F242]	Include shelter information	Should have	Since a shelter location is not restricted to only being partnering shops this is not essential for a first release (could use other sheltering locations such as park benches).

Requirement ID	Description	Classification	Reasoning
[F243]	Include escape information	Must have	Essential to the applications core functionality of offering the user an escape event.
[F244]	Include fight back information	Must have	Essential to the applications core functionality of offering the user an escape event.

## Priority Table for Non-Functional Requirements

Requirement ID	Description	Classification	Reasoning
[NF251]	Search trail response	Should have	Important to user satisfaction but users can accept some delay.
[NF252]	Accurate trail data	Should have	Required to provide the user with reliable trail information.

Any requirements in regards to security concerns were given the highest prioritization. These requirements were selected since they deal with sensitive information such as personal information, health data obtained from HRMs, and location tracking. If the application is not secure users will not use the application thus these requirements are fundamental to the apps success and user confidence.

To determine prioritization, the functionalities needed for a minimal viable product were considered first and given the classification of 'Must have'. These would include requirements essential to the apps operation and legal obligations. Next considered were requirements that would enhance user experience and engagement. These requirements were given the 'Should have' classification. Than features that offer additional value but are not essential were given the 'Could have' classification. This ensure the product has a smooth release.

## (S.6) Verification and acceptance criteria

To validate **functional** requirements we will use a series of tests conducted during the development process. Unit testing will be conducted to ensure that each component works correctly in isolation. Then integration testing to ensure that components work together seamlessly. System testing will be conducted as well to test the system as a whole. We will have a focus on acceptance testing that is supplemented by the above tests. Before, development work we will include early validation with low, medium, and high fidelity prototypes we will present to relevant stakeholders. In addition before a release the release product will be presented to relevant stakeholders.

Unit testing will have 85% coverage per component, integration testing at a minimum will have a 95% pass rate, system testing will produce no major bugs and 3% maximum minor bugs. Acceptance testing should have 90% stakeholder satisfaction, 90% of key stakeholders should be presented prototypes, and 90% of stakeholders should approve the product before release.

For the **non-functional** requirements:

- **Trail user request:** A user request on the trail database (for example, a search for a specific trail) should have a quick response time.
  - Best achievable response time is within 0.1 second
  - Worst acceptable response time is 2 seconds
  - Target response time is 0.5 seconds
  - Covered requirements: [NF251]
- **Up-to-date trail information:** Trail information should be regularly updated, user feedback should reflect that they perceive the trail information as accurate and current. 90% of user sample group should not report come across inaccurate trail information.
  - Best achievable 100% of users
  - Worst acceptable 80% of users
  - Target 90% of users
  - Covered requirements: [NF252]

One of the most important usage scenarios for the system is [S.4.2] Player encountering virtual enemies. This is the core of the app's gameplay. To validate this scenario, we propose the following five tests in Gherkin Language:

**Feature:** Player encounters virtual enemies during an active running session.

- **Scenario:** Player gets a notification about virtual enemy appearance (S.6.1)
  - **Given:** The player is in an active running session
  - **When:** A virtual enemy appears on the player's route
  - **Then:** The player should receive a notification about it
- **Scenario:** Player chooses an action event (S.6.2)
  - **Given:** The player has encountered a virtual enemy
  - **And:** The system has provided action event options to the player
  - **Then:** The player should select one of the action events
- **Scenario:** Player executes an action event (S.6.3)
  - **Given:** The player has encountered a virtual enemy
  - **And:** The player has selected an action event

- When: The system provides the event information to execute
- Then: The player should execute the event
- Scenario: Player's location changes during active running session (S.6.4)
  - Given: The player is in an active session
  - When: The player moves or diverges from current location
  - Then: The system should update the player's location
- Scenario: System collecting player's HRM data during active running session (S.6.5)
  - Given: The player has connected an HRM device
  - When: The player selects and executes an action event
  - Then: The system should record and measure the player's HRM data

The traceability matrix below summarizes these tests and demonstrates how they link to the requirements.

Table 5. Traceability matrix for scenario validation

Req. ID	Req. Description	Test	Test Status
[F232]	Record and display location information	S.6.4	Not executed
[F233]	Collect and display heart rate information	S.6.5	Not executed
[F234]	Notify user of virtual enemy	S.6.1	Not executed
[F241]	Determine available event options to user	S.6.2	Not executed
[F242]	Provide escape, shelter, fight back information	S.6.3	Not executed
[F243]			Not executed
[F244]			Not executed

# (P) Project

## Control Information

Table 6. ACME RUN — Versioning Information — Project Book

Section	Version	Lead	Delivered	Reviewer	Approved
P.1	1	MBG	2023-12-03	KM	2023-12-04
P.2	1	KM	2023-11-30	AF	2023-12-03
P.3	1	AF	2023-11-25	MBG	2023-11-28
P.4	1	AF	2023-11-28	MBG	2023-11-30
P.5	1	KM	2023-12-04	AF	2023-12-04
P.6	3	KM	2023-12-05	AF	2023-12-06
P.7	2	MBG	2023-10-31	KM	2023-10-31

## (P.1) Roles and personnel

- **Project Manager:** Oversees the applications development. Facilitates integration between different roles/teams and component integration. Involves a lot of planning such as quarterly planning and sprint planning while ensuring the project aligns with its intended goal.
- **Full Stack Developer:** Responsible for server-side and client-side operations of the application. This includes database management, integration with external hardware, user interface, and user experience.
- **Fitness Specialist:** Provides knowledge in exercise science. This will be used when creating the physical challenges in the applications such as fighting back and escaping. This role ensures that the exercises maximize the positive outcome for the user based on their fitness profile.
- **Security Specialist:** Utilized to ensure the security of the application. This role would secure user data, ensure compliance with security regulations, and protect against security threats.
- **Legal Specialist:** Ensure the application fulfills legal obligations. Including adhering to intellectual property rights, patent law, and user agreements. Vital for navigating legal complexities.

## (P.2) Imposed technical choices

From the client end, there are no restrictions that bind the project to specific technical elements or parameters. However, the following should be considered:

- Cross-platform (mobile) for user application. Game engines like Unity may be used for cross-platform app development.

- Database(s) for storing trail and user data is essential. This may necessitate the use of SQL in backend development.

## (P.3) Schedule and milestones

**Sprint 1 (Weeks 1-4): Establishing Foundations** - Epic 1: Set up the basic infrastructure for ACME RUN, including user authentication, database schema, and initial app structure. - Epic 2: Implement the user profile system, allowing players to register and input basic information. - Epic 3: Develop the geo-fencing mechanism to define the initial running trails within McMaster's selected pilot area. - Epic 4: Integrate basic functionalities for sheltering, escaping, and fighting back based on predefined scenarios.

**Justification for Sprint 1 Minimal Viable Product (MVP):** During the first sprint, our focus is on laying the groundwork for ACME RUN. By establishing the fundamental infrastructure and implementing basic user functionalities, we aim to create a foundation upon which subsequent features can be built. This sprint won't deliver the full game experience but sets the stage for subsequent sprints to add more sophisticated game mechanics and enhance user engagement. The MVP at the end of Sprint 1 will allow users to register, access their profiles, and interact with basic gameplay elements (sheltering, escaping, and fighting back) on the defined running trails.

**Sprint 2 (Weeks 5-8): Gameplay Mechanics Enhancement** - Epic 5: Expand the options for sheltering, integrating additional safe zones on the trails. - Epic 6: Implement dynamic escaping mechanics, considering user statistics and previous runs. - Epic 7: Introduce the fighting back feature, incorporating a set of body-weight exercises as a response to virtual enemies. - Epic 8: Enable user decision-making and interaction with the app during gameplay scenarios.

**Sprint 3 (Weeks 9-12): Heart Rate Monitor Integration and Fitness Customization** - Epic 9: Integrate Heart Rate Monitor (HRM) support for collecting user heart rate data. - Epic 10: Develop adaptive workout events based on HRM data, aligning with user fitness goals. - Epic 11: Implement badge and challenge systems, allowing users to track their achievements. - Epic 12: Refine user profiles to include additional fitness-related information.

**Sprint 4 (Weeks 13-16): Final Polishing and Deployment Preparation** - Epic 13: Conduct extensive testing to ensure the app's stability, performance, and security. - Epic 14: Implement localization features for potential deployment in other North American cities. - Epic 15: Finalize administrative tools for local operations, including trail definition and shelter administration. - Epic 16: Prepare documentation for deployment, including user guides and operational manuals.

**Justification for Sprint 4:** In the final sprint, our focus will be on refining the app, conducting thorough testing, and preparing for deployment beyond McMaster's pilot phase. This includes localization features for potential expansion, ensuring administrative tools are robust, and providing comprehensive documentation for a seamless handover to local partners. The end of Sprint 4 marks the completion of the ACME RUN project, ready for launch in Fall 2024.

## (P.4) Tasks and deliverables

### Sprint Backlog for Sprint 1: Establishing Foundations

1. User Authentication Setup
  - Business Value: Enables secure user registration and login.
  - Technical Estimate: 5 story points.
2. Database Schema Implementation
  - Business Value: Establishes the foundation for storing user data.
  - Technical Estimate: 8 story points.
3. Basic App Structure Development
  - Business Value: Provides the structural framework for future feature integration.
  - Technical Estimate: 3 story points.
4. User Profile System Implementation
  - Business Value: Allows players to register and input basic information.
  - Technical Estimate: 5 story points.
5. Geo-fencing Mechanism for Running Trails
  - Business Value: Defines initial running trails within the pilot area.
  - Technical Estimate: 8 story points.
6. Basic Functionalities for Sheltering, Escaping, and Fighting Back
  - Business Value: Introduces rudimentary gameplay mechanics.
  - Technical Estimate: 10 story points.

Note: Story points are a relative measure of complexity and effort, with a higher number indicating greater complexity.

Total Story Points for Sprint 1: 39

Justification for Sprint 1 Backlog: The selected stories for Sprint 1 prioritize foundational aspects of ACME RUN, focusing on user authentication, database setup, and the initial app structure. These tasks collectively establish the infrastructure needed for subsequent sprints to build upon. The distribution of story points reflects the estimated complexity of each task, ensuring a balanced workload for the development team during the sprint.

## (P.5) Required technology elements

The following elements will be required for the project:

- **Compliance with external Maps API:** This is critical to the application in order to render maps and enable geo-fencing/location services.
- **Compliance with external HRM devices:** The connection of an HRM device is decided by the user, but this element is still required in the event of connection. This is critical in order to collect the connected user's real-time heart rate data.

These elements introduce a shared security concern related to data collection and privacy. Personal information, including health statistics and real-time location, will be collected to enhance gameplay. This information is sensitive data and improper handling can compromise user privacy.

## (P.6) Risk and mitigation analysis

Some risks in relation to the functional requirements and their respective mitigation strategies have been identified below:

- **HRM compatibility:** A major aspect of this app is the ability for users to connect their personal HRM device in order to log activity and customize their workouts. There may be issues in making sure a variety of HRM devices are compatible with the app if close collaboration with HRM manufacturers becomes difficult or not feasible. This may be mitigated by thoroughly testing a multitude of commonly-used HRM devices in parallel with the development of the app. Another similar strategy would be to develop a custom compatible HRM device that users will have the option to purchase. This can ensure that users have access to the personalized feature of connecting an HRM device without compatibility issues.
- **Difficulties with geo-fencing and real-time location tracking:** As the app will be geo-fenced and maintained by local partners, difficulties may arise in implementing safe and accurate localized gameplay. This may occur if geographical data is not accurate due to changes in the physical environment, or it becomes difficult to have the application maintained by local partners. One mitigation strategy may be to ensure approved trails and shelters are kept up-to-date by including user feedback or self-reports. By allowing local partners to access and respond to these reports, they can promptly update current conditions as necessary. This would allow for quick fixes in any discrepancies between the app's stored data and real-time conditions.

The following are potential threats to the project:

- **Malicious user attempts to gain access:** The system may have vulnerabilities related to data encryption (such as outdated protection/encryption models), there is a possibility that a malicious user would exploit these vulnerabilities and attempt to access private data. Two example scenarios in which this can occur are: compromising a user's device or installing malware onto the servers, which can compromise the confidentiality, integrity, and availability of the project. One mitigation mechanism that can be proposed to combat the risk of this threat is the routine validation and/or update of the system's firewalls and encryption software to maintain robustness and defenses against such attacks.
- **Unauthorized access to external components:** The system is dependent on the utilization of several external components such as APIs. This dependency exposes the system to vulnerabilities related to

the third-party services. For example, the system would be impacted if a service disruption or some form of cyber-attack occurs on the Maps API, potentially compromising confidentiality, integrity, and availability (e.g., user's location/health data is stolen, maps can't be rendered). A mitigation mechanism to avoid this potential threat would be to use authentication keys/tokens or a gateway to reduce the possibility of an unauthorized attempt, or consider implementation of internal APIs to reduce reliance on external components.

## (P.7) Requirements process and report

To elicit information for each stakeholder we will be relying on interviews. For each stakeholder we will define the type of interview and the most important question.

- New to fitness (**Charlie**)
  - Type of interview: **Closed**. Since Charlie is inexperienced in fitness, and the information we want to obtain is exclusively fitness related, we would want to have a more focused interview.
  - Question: *What five features in a fitness app would entice you to download the application?*. This would tell us some fitness needs of someone who is new to fitness. We could then use this to better shape the application to meet user needs.
- Not new to fitness (**Blake**)
  - Type of interview: **Open**. Compared to Charlie, Blake has more experience in fitness. Blake may have insights that have not yet been explored. Thus, we would like to gather as much information as possible.
  - Question: *"If you were to download a fitness app, what are features you would look for?"*. This would be a great opening question that allows us to go into specifics with Blake on what a player that isn't new to fitness would look for. Might result in a discovery of new feature that have not been explored.
- Parks and Recreation Department - Park Ranger/Police (**Jordan**)
  - Type of interview: **Closed**. From this individual, we want to gather information about trail safety. Specifically, we are seeking details about the procedure for patrolling and responding to an incident on a trail. Therefore, we intend to keep the questions restricted.
  - Question: *When you come across an incident or an incident is reported, how is this logged and what are your next steps?*. This will allow us to discover their logging method as well as what is done after an incident. These procedures will affect requirements for ACME RUN. Specifically important to ACME RUN's overarching goal of safety.
- HammerCorp Inc. - Representative (**Avery**)
  - Type of interview: **Open**. We want to have an open discussion on the expected requirements and deliverables for the project. This will enable us to gather as much information as possible regarding what is expected.
  - Question: *What is the goal of this application?*. This is intended to be an opening question to prompt further discussion. It seeks to uncover the overarching goal of the application, which is

essential to aligning the development with the overall mission. Based on the response, we can then ask more specific questions.

- McMaster University - IT Department (**Ezra**)
  - Type of interview: **Closed**. For this interview, we aim to obtain a specific set of information. We are interested in learning about the technological infrastructure the partner has in place, how they are maintaining it, and how this infrastructure can support the ACME RUN. application.
  - Question: **What is required for an application to be locally operated by your facility?**. The most important information we aim to obtain is in regards to the infrastructure requirements. Therefore, this is a pointed question to extract information about what specifically our app must to adhere to in order for McMaster University to operate it locally.

# References

- [1] Bertrand Meyer. *Handbook of Requirements and Business Analysis*. Springer. 2022.
- [2] Ian Sommerville and Peter Sawyer. *Requirements Engineering: A good Practice Guide*. Wiley. 1997.

# Appendices

## (Appendix A) Security Considerations

### Security by Obscurity

- The "Security by Obscurity" approach is a method where the security of a system is based on the secrecy of its design or implementation. In the context of the ACME RUN project, this approach would not be effective. The reason is that the application's security is not solely dependent on the secrecy of its design or implementation. The application's security is also dependent on the security of the data it handles, which includes user profiles, running data, and potentially sensitive health information. If this data is compromised, the obscurity of the system's design or implementation would not protect it. Furthermore, the application's security also depends on the security of the devices it runs on, which are typically owned by the users. If these devices are compromised, the obscurity of the system's design or implementation would also not protect it.
- If the "Security by Obscurity" approach were to be implemented, it would be violated in a situation where a user's device is compromised. For example, if a user's device is infected with malware, the malware could potentially gain access to the data stored on the device, including the data related to the ACME RUN application. This would violate the obscurity of the system's design or implementation, as the security of the system is not solely dependent on the secrecy of its design or implementation.

### Security Property Assessment

We will deploy standard encryption protocols for stored and in transit data. For example a symmetric encryption scheme can be used for the data that is being stored and Secure Socket Layer/Transport Layer Security (SSL/TLS) can be used for data that is being transported. In addition only select individuals are able to access this information . When these individuals attempt to access this information there will be several layers of authentication. Regular security audits will also aid in confidentiality.

Availability when talking about the application would mean that the application is available when users need it. Several techniques can be deployed to keep the system available to users. This would include load balancing, regular maintenance, regular updated, and redundancy in critical components. Load balancing would aid the application in high traffic situations. Regular maintenance and updating are crucial for discovering vulnerabilities and compatibility. Redundancy in critical components would prevent the system from having a single point of failure. Furthermore, having a disaster recovery plan and regular backups will aid in the event of an outage and help with near continuous availability.

## (Appendix B) Window of Opportunity #1

Below is a table summarizing the changes made to the document related to the feedback provided from the window of opportunity #1.

Section	Feedback	Changes made
G1	Good explanation of the overall objective!	N/A
G2	The current situation is meant to reflect more the current situation of the users rather than of the application. Think of it as the 'why are we building this'.	Rewrote the current situation to reflect the current situation of the users and why we are building this application.
G7	The categorizing the users into those that are new to fitness and those who are more comfortable with it is a good idea! However, for the 'clients' its not enough to say 'clients', every project has a client(s), you need to be specific about who those are for this. Although its identified in the paragraph, you need to be explicit about it in your summary table and paragraph title. It might be a good idea to remove 'proxy' from the table since its a little confusing on if theyre a direct client or if theyre a proxy for the actual client. The same goes for the 'indirect client' — its not a good idea to lump all clients under one name since they will each have a different need or interest, so be explicit (just say McMaster University). What is HRM? For the requirements sources, it was stated 'up to three', choose the 3 that are the most important from your list moving forward.	Specified HammerCorp Inc. as the client. Changed 'proxy' to 'representative'. Specified McMaster University as the indirect client. HRM is defined in the glossary. Choose the three most important requirements.
G3	Good summary of goals! Only thing to improve upon is the explanation for the local shops (refer to E1 here for the reader to understand what you mean by sheltering). One thing to consider is that if a lot of these trails are in woody areas, its unlikely that they will have easy access to these local shops.	Added a reference to shelters from E.1. to provide context on sheltering, and provide insight into consideration for trails located in more isolated areas where local shops are not accessible.
G4	In general, these are good requirements. The geo-fencing and local operations is a little unclear — they could be moved to NFRs as geolocation is a property of the system rather than a feature of it.	Removed geo-fencing and local operations from the functional requirements. Since there is a maximum of 2 non-functional requirements geo-fencing was not included in the non-functional requirements.

Section	Feedback	Changes made
E1	Good glossary terms. One area of improvement might be to just give examples of what partnering shops might be (ex. are they general sports shops? or any other kind of shops?). Good way of clarifying exactly where the body weight exercises will be required in the application.	Included an extra sentence to the end of the partnering shops definition: 'Examples of partnering shops include cafes, grocery stores, and retail stores.'
E5	These are good as well. The only thing I would ask is are there any potential negative effects? These all seem to be repeated from the Expected Benefits section and so thinking either about more benefits or even harmful effects would strengthen this section.	Considered potential negative effects related to the first bullet point.
E6	Good invariants!	N/A
P6	Good risks and mitigation tactics. To add on to the geo-fencing issue, what will you do with the reports?	Added clarification about user reports.
P7	Questions and justifications are good! Again, just revise the 'client' classification.	Changed the client and partner interviews based on what was changed in G7. Specified HammerCorp Inc. is the client and McMaster University is the partner.

## (Appendix C) Window of Opportunity #2

Below is a table summarizing the changes made to the document related to the feedback provided from the window of opportunity #2.

Section	Feedback	Changes made
G5	Generally the use cases are good in their structure but the steps should be very coarse (ie. should be more specific). Many of the scenarios seem very general and very vague.	I interpreted the meaning of "coarse-grained steps" as in the steps should just be simple/broad/roughly described (i.e. high-level). However, I also agree that the use cases were a bit too ambiguous and thus refined steps to include more detail and be overall more specific to the project. Additionally, "Manage shelters" was removed as a use case because it felt redundant with the "Manage trails" use case upon review.
G6	Good limitations!	N/A

Section	Feedback	Changes made
E3	The listed points don't exactly sound like constraints on the system. Constraints on a system essentially are meant to limit the applications scope. The listed items sound like general ideas of the system.	Revised first constraint to focus on how the integration of local partners limits the system. Replaced a constraint to focus on geo-fencing boundaries instead. The third constraint was revised to describe the limitations on the scope more clearly; it should still be considered a constraint because an outdoor-based game cannot be played on a non-mobile device.
E4	Good assumptions but the second one about the heart rate monitor, that seems to be something that's dependent on what the requirements are.	Reordered the assumptions and removed the heart rate monitor assumption.
E2	Good job here!	N/A
E1	Generally good work here but some of the attributes of the class are software implementations rather than actual business logic (ex. password, email, username) because authentication details are not part of the core business concepts.	Removed password, email, and username from the diagram.
S1	For the descriptions here, please make sure they are a little more informative of what the relevance and goal of the component is. Good component diagram!	Add more detail to each component including stating the relevance and goal explicitly.
S3	Please provide a description for the API! Is there only one API? How are the managers involved in the API?	Updated the description for the API, including more detail. Also added a description for the managers interaction with the API.
S2	The instructions might've been misinterpreted, it was asking for two NFRs for the general system and not for each component. Additionally, ensure that requirements can be verifiable! Good mockups otherwise!	Selected the two best NFRs, and reviewed requirements.
S6	Really good work for identifying the fit criterion with measurable points! The section for the functional requirements could be improved by including just a few measurements as well (ex. is it 80% unit test coverage?). Without numbers, it is almost impossible to verify NFRs!	For the functional requirements section I added measurements for all types of tests mentioned.

Section	Feedback	Changes made
S4	Confused as to why some of the use cases scenario encapsulate multiple use cases? A scenario typically considers one use case at a time. Additionally, make sure to include the list of secondary actors where appropriate! Aside from that, good job on the activity diagrams!	Added secondary actors where needed, modified certain scenarios such that they only encapsulate a single use case.