Software Requirements Specification for ERP Ranger Mobile Application Developed for EPI-USE ERP by The Tenacious Technicians

de Villiers, Charles
u16056559

Kirsten, Eric
u16020431

Nel, Johan
u16354029

Ross, Justin
u17080526

Schwikkard, Dylan
u16120206

18 July 2019

1 Introduction

1.1 Vision

To provide ERP with efficient and effective system, making for a better and easier experience as a ranger to better help safeguard the park's wildlife.

1.2 Objectives

- To provide rangers with a mobile application to simplify their shifts and incentivise them to do more shifts.
  - To allow rangers to start a patrol when they arrive at the park.
  - To monitor rangers while on patrol.
  - To keep track of ranger's patrol history.
  - To allow for reporting and feedback of relevant information.
  - To incentivise rangers to go on more patrols and follow certain routes.
- To provide a web-portal for park management and administration to make various management tasks easier
  - To analyze patrol patterns.
  - To manage users in a database
  - To manage notifications
  - To respond to incidents and keep track of reported incidents.
  - To add and remove rewards from the online store.

1.3 Business Need for project

The current system in place is not optimized and not well coordinated. There is no easy way for rangers to log their current shifts or book future shifts. Currently a Whatsapp group is used to monitor shifts, report and respond to incidents, as such the group quickly becomes cluttered and inefficient to use.

Hence, we intend to provide rangers with these features and more in order to better track shifts and ranger patrols. This makes coordination between rangers and park staff much simpler and provides the rangers with a safer and monitored way to go on their various patrols around the park. Our system will also provide incentive for rangers in the form of rewards from the ERP store (merchandise, etc.), to perform more patrols.

1.4 Scope

The ERP ranger mobile application is a utility for rangers to check in once they arrive to start their shift. The app also allows for rangers to book shifts in advance and log current shifts. Whilst out on patrol, various markers can be passed and marked which will reward the ranger with points. Rangers will also be able to report on events through the app, which, based on the severity of the incidident can notifiy administrative staff with an sms and/or email, the settings of which can be configured.

Administrators or management may use the web portal to add and manage rewards and ranger users. They will also be able to monitor specific ranger's patrol and incident reporting history.

The software requires Internet connection to retrieve and edit information in a remote database as well as a GPS connection to track the rangers as they patrol.

2 Domain Model

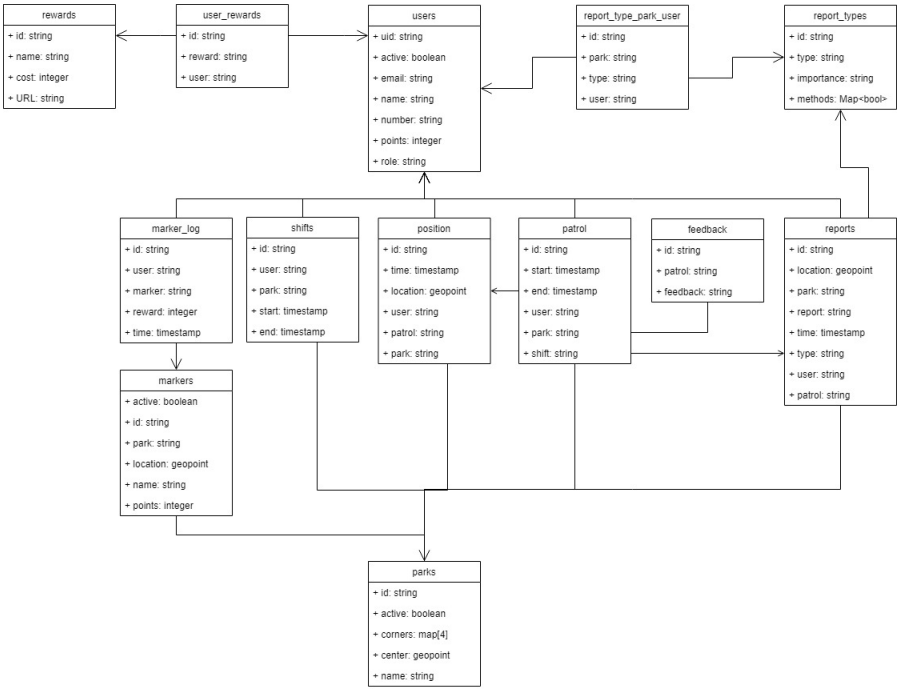

Figure 1. The domain model of the ERP Rangers application.

3 User characteristics

- Ranger
  - A general ranger will be using the app primarily. This will be for various tasks such as creating or view his/her shifts, reporting things they've seen in the park, as well as check off a marker.Rangers will also have low level access to the web portal as well to view their shifts as well as their current points as well as a store for them to redeem these points
- Administrator
  - This user has high level access to the web portal and are able to view all user shifts, reports and where these reports took place. The administrator is also able manage the park markers. In addition to this, the user will be able to add or remove items from the web store.

4 Functional Requirements

Use Cases

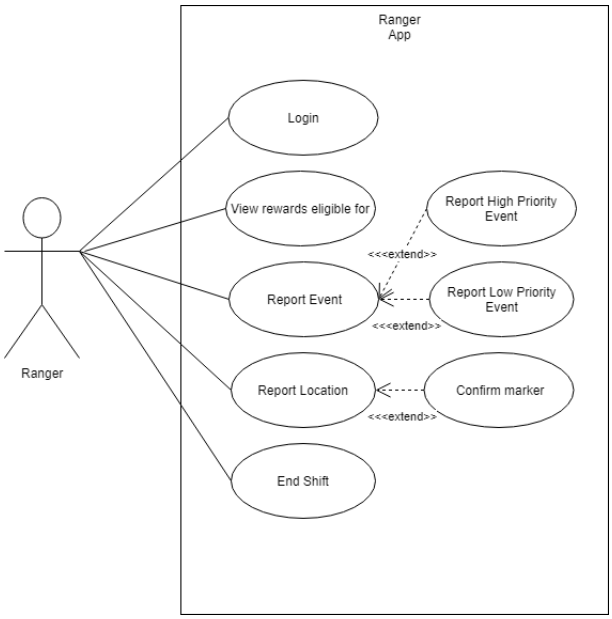

Figure 2.1: The use case regarding the front end of the ERP Ranger Mobile Application.

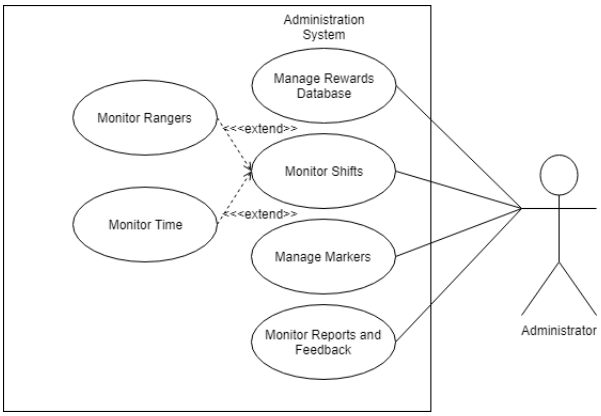

Figure 2.2: The use case regarding the administrative web portal for the ERP Ranger Mobile Application

Requirements

R1 The system shall only allow registered rangers to login.

R2 The system shall allow rangers to view and manage their patrol shifts.

R3 The system shall only allow rangers to start their shifts once it detects that they are inside a specific geofenced area.

R4 The system shall report a ranger's position at specific intervals whilst they are on patrol.

R5 The system shall allow rangers to see the location of markers on a map.

R6 The system shall allow rangers to earn points by checking off a marker that they pass.

R7 The system shall allow rangers to report any events or anomalies they encounter during their patrol.

R8 The system shall allow a ranger to log feedback upon completion of their shift.

R9 The system shall allow a ranger to redeem points in return for merchandise.

R10 The system shall allow an administrator to register new rangers.

R11 The system shall allow an administrator to add and remove markers.

R12 The system shall allow an administrator to monitor rangers in the park.

R13 The system shall allow an administrator to add and manage rewards.

R14 The system shall allow an administrator to monitor reports and feedback.

R14 The system shall allow an administrator to to receive SMS and email notifications based on report(R7) severity .

Subsystems

- U1: Login Subsystem
- U2: Shift Subsystem
- U3: Report Event Subsystem
- U4: Report Location Subsystem
- U5: Feedback Subsystem
- U6: Marker Subsystem
- U8: Admin Portal Subsystem

5 Quality Requirements

This section provide a definition of the quality for the ERP Ranger System. The definition is provided using quality factors and their associated quality metrics.

5.1 Quality Factors

5.1.1 Mobile Application

- Performance - the speed at which the system manages to complete functions.
- Reliability - the probability that the system will successfully perform its functions.
- Security - how secure the system is to outside interference.
- Integratability - how easily the service integrates with the target environment.
- Usability - the ease by which the client can use the system.
- Availability - how often the service is accessible to the users.

5.1.2 Admin Portal

- Performance - the speed at which the system manages to complete functions.
- Reliability - the probability that the system will successfully perform its functions.
- Security - how secure the system is to outside interference.
- Integratability - how easily the service integrates with the target environment.
- Usability - the ease by which the client can use the system.
- Availability - how often the service is accessible to the users.

5.1.3 Server

- Performance - the speed at which the system manages to complete functions.
- Reliability - the probability that the system will successfully perform its functions.
- Security - how secure the system is to outside interference.
- Integratability - how easily the service integrates with the target environment.
- Usability - the ease by which the client can use the system.
- Scalability - how easily the service can be scaled to accommodate a larger environment.

5.2 Quality Metrics

5.2.1 Mobile Application

- Performance:
  - Measurement Method: The average time to perform a function.
  - Frequency: Weekly.
  - Target Performance: Less than 1 minute.
- Reliability:
  - Measurement Method: The percentage of failed cases out of attempted cases, where cases are the attempt to perform a function.
  - Frequency: Weekly.
  - Target Reliability: Less than 1%.
- Security:
  - Measurement Method: The number of standard security check failures.
  - Frequency: Weekly.
  - Target Security: 0.
- Integratability:
  - Measurement Method: Monitored through the use of a survey.
  - Frequency: Monthly.
  - Target Usability: Good reports from majority of the users.
- Usability:
  - Measurement Method: Monitored through the use of a survey.
  - Frequency: Monthly.
  - Target Usability: Good reports from majority of the users.
- Availability:
  - Measurement Method: Percentage of time the service is not available.
  - Frequency: Weekly.
  - Target Availability: Less than 25%.

5.2.2 Admin Portal

- Performance:
  - Measurement Method: The average time to perform a function.
  - Frequency: Weekly.
  - Target Performance: Less than 1 minute.
- Reliability:
  - Measurement Method: The percentage of failed cases out of attempted cases, where cases are the attempt to perform a function.
  - Frequency: Weekly.
  - Target Reliability: Less than 1%.
- Security:
  - Measurement Method: The number of standard security check failures.
  - Frequency: Weekly.

  - Target Security: 0.

- Integratability:
  - Measurement Method: Monitored through the use of a survey.
  - Frequency: Monthly.
  - Target Usability: Good reports from majority of the users.

- Usability:
  - Measurement Method: Monitored through the use of a survey.
  - Frequency: Monthly.
  - Target Usability: Good reports from majority of the users.

- Availability:
  - Measurement Method: Percentage of time the service is not available.
  - Frequency: Weekly.
  - Target Availability: Less than 25%.

5.2.3 Server

- Performance:
  - Measurement Method: The average time to perform a function.
  - Frequency: Weekly.
  - Target Performance: Less than 1 minute.

- Reliability:
  - Measurement Method: The percentage of failed cases out of attempted cases, where cases are the attempt to perform a function.
  - Frequency: Weekly.
  - Target Reliability: Less than 1%.

- Security:
  - Measurement Method: The number of standard security check failures.
  - Frequency: Weekly.
  - Target Security: 0.

- Integratability:
  - Measurement Method: Monitored through the use of a survey.
  - Frequency: Monthly.
  - Target Usability: Good reports from majority of the users.

- Usability:
  - Measurement Method: Monitored through the use of a survey.
  - Frequency: Monthly.
  - Target Usability: Good reports from majority of the users.

- Scalability:
  - Measurement Method: Average amount of time and cost required to expand the
  - Frequency: Weekly.
  - Target Cost: Nothing.
  - Target Time: Within a day.

6 Trace-ability matrix

	U1	U2	U3	U4	U5	U6	U7	U8
R 1	X							
R 2		X						
R 3		X			X			
R 4			X					
R 5					X			
R 6							X	
R 7					X		X	
R 8				X	X			
R 9						X		
R 10							X	
R 11								X
R 12							X	X
R 13					X			X
R 14			X					X
R 15				X		X		X

7 Deployment Model

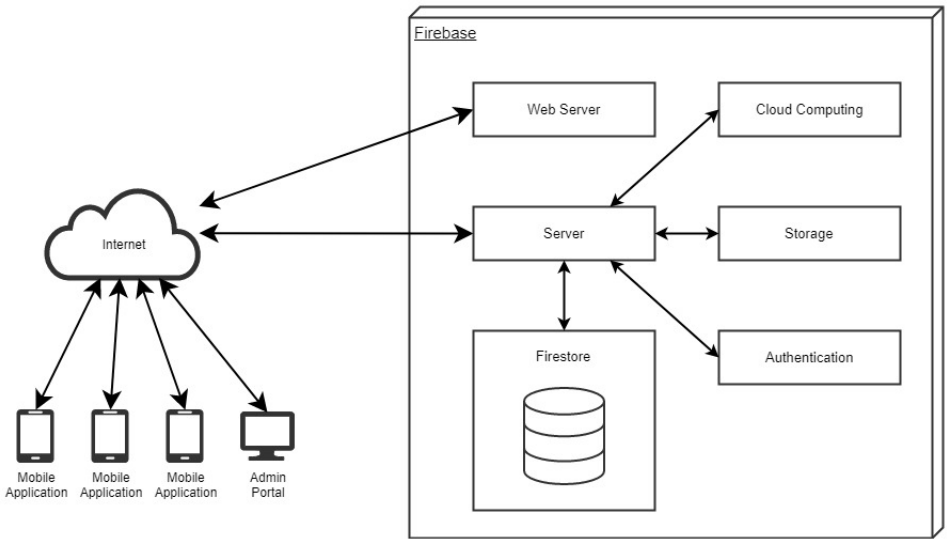

Figure 4.1 The deployment model of the ERP Rangers application.

8 Architectural design

The design objectives for this system is to incentivise rangers to patrol more regularly and also patrol certain paths by providing them with points that can be redeemed on the ERP online store. It also aims to give administrators an easy platform from which to manage and monitor the rangers and the markers in the park.

Our system mainly uses a Client Server Architecture. The reason for this is because many subsystems of our firebase server provide services to the rangers mobile application. The application communicates with an external authentication service to verify the user that logs in. After this it will fetch the location pins for the "markers" that can be visited during that shift. If an event occurs a report can be sent to the web administrator portal (on the firebase server) which will then notify the administrators, or the parks external security service.

A device running the mobile application does at no point share data directly with another device other than the web administrators portal. Hence it functions as a client to our firebase server.

9 Technology Decisions

9.1 Server side

We decided on Google's firebase technology for our server side computing needs as it saves on development time, it provides many features in the form of databases, cloud computing, cloud storage and authentication that we can use and it also has great reliability. It also easily integrates with various other platforms, languages and technologies.

9.2 Ranger Mobile Application

We decided on the flutter framework because it allowed us to develop the application for both Android and iOS users.

9.3 Other

Because we need to monitor that rangers are actually at the park before beginning their shifts we will be using low energy bluetooth beacons to geofence the area where the rangers will report in before starting their shifts.