**Software Requirements Specification**

for

**KALAMAZOO NATURE CENTER TRAIL APP**

Prepared by

Wei Shun Tye

Devin James Ragotzy

Geoffrey James Fleenor

Ernest Au Yue Wei

Xin Yi Koay

*Western Michigan University*

12-05-2023

**Table of Contents**

**Abstract**

**Introduction**

Purpose

Intended Audience

Intended Use

Product Scope

Definition and Acronyms

**Overall Description**

User Needs

Assumptions and Dependencies

**Functional Requirements**

Functional Requirements

**Nonfunctional Requirements**

Performance requirements

Technical requirements

Usability requirements

Scalability requirements

**Other Requirements**

External Interface Requirements

System Features

Abstract

Kalamazoo Nature Center (KNC), which has been recognized by its peers as one of the top nature centers in the country, is home to over 14 miles of hiking trails winding through woods, wetlands, and prairies. There are numerous places/plots in KNC that have an interesting and impressive history besides being home to a variety of animals and hundreds of wildflowers and other plant life. To improve the visitor’s experience at KNC, we will design a software app via the senior capstone project at the department of Computer Science at WMU. As the first step towards establishing a reference model for software app that the seniors at WMU can further develop, I will design a series of artifacts that capture the important features such as software requirements, software architecture and test reports. In this report, a set of software requirements are identified, which are organized in tabular formats for better visualization.

Software requirements identified for the KNC app not only summarize the common and important features that should be implemented by a software app, but also lay down the ground for deriving a minimum set of safety properties that the app should satisfy. That is to say, the app should not be considered as complete. Based on the software requirements, software architecture will be given followed by the test reports where all important test cases are given.

Students may implement the software requirements in this document in their senior project in various programming environments. Importantly, all requirements presented in this document shall be considered as mandatory or exhaustive.

The rest of this document is organized as the following: in section 1 we first define the system features and requirements used throughout the KNC project; section 2 presents the non-functional requirements

Introduction

**Purpose**

* 1. The purpose of the app is to be used as a trail aid by the KNC. In general, the app would be used while walking on KNC trails to scan signage, as well as remotely walking through the trail remotely.

**Intended Audience**

* 1. The Intended audience are people who are members of the KNC and come to walk on KNC trails.

**Intended Use**

* 1. KNC employees: employees shall eventually have access to edit content in the app and make changes with trails.
  2. WMU students developers: The students have access to the entirety of the app, and shall follow the SRS by building an app in line with the goals of the KNC.

**Product Scope**

* 1. Business Goals:
     1. Build and finish creating the app by the end of Fall 2023 Semester.
  2. Objectives:
     1. Finish SRS by the end of the Spring 2023 semester.
     2. Start building app during the Spring 2023 semester, and continue throughout the summer and into the Fall of 2023

**Definitions and Acronyms**

* 1. KNC - Kalamazoo Nature Center
  2. React Native - UI software framework used to develop Android and iOS applications

1. Overall Description

**User Needs**

* 1. Trail history and conservation: Information about the history and conservation of trails and the surrounding areas.
     1. There shall be playable audio files for recordings of wildlife sounds or interviews providing insights into the trail’s history.
  2. Trail mapping and navigation: The ability to view maps of trails and to easily find and follow a desired route.
     1. There shall be a button that leads users to the trail maps.

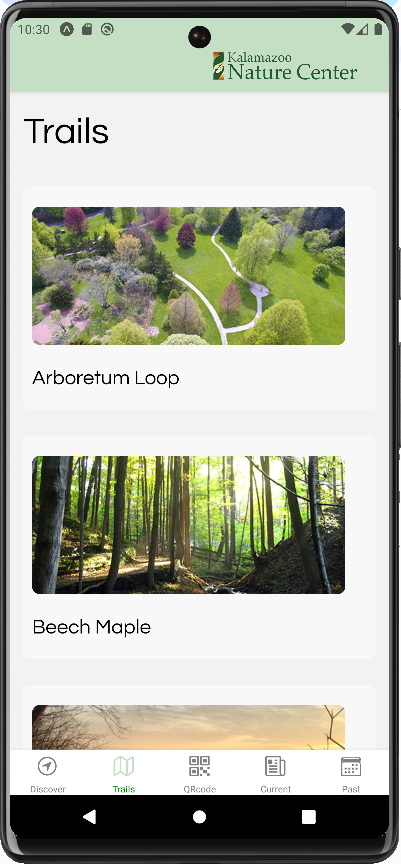
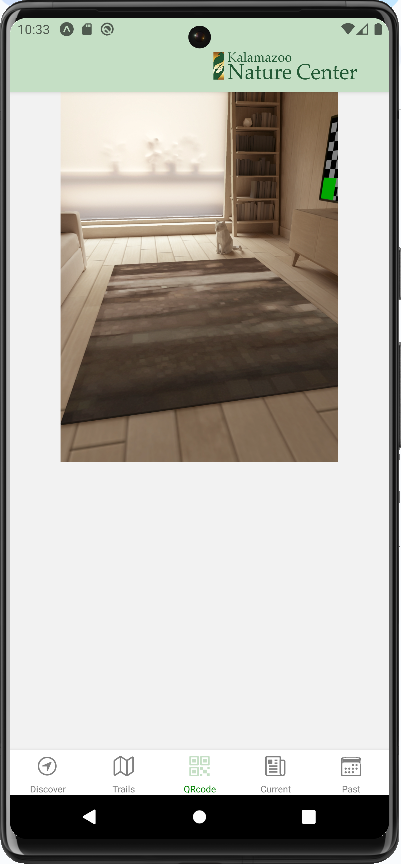
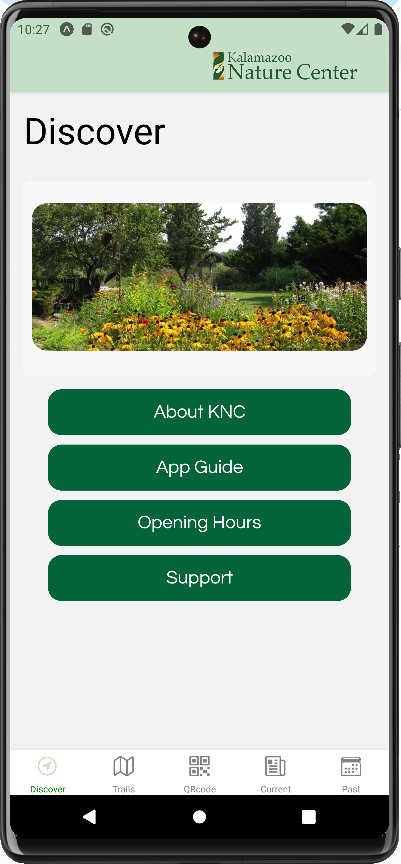
**Assumptions and Dependencies**

* 1. We are dependent on the React Native framework for the app to work on iOS and Android.

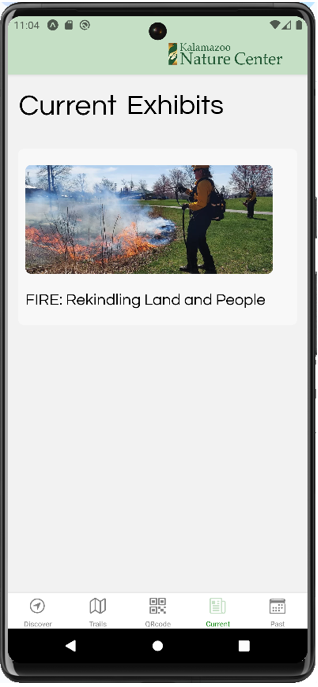
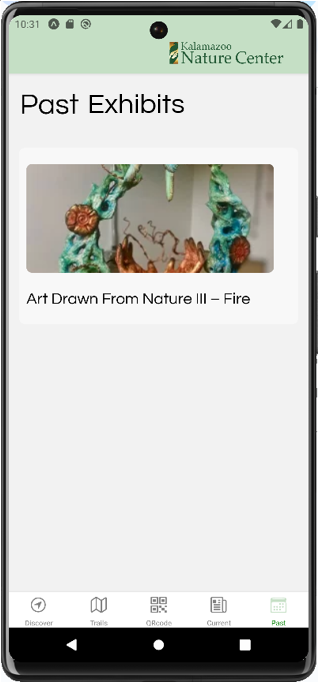
1. Functional Requirements

**Functional Requirements**

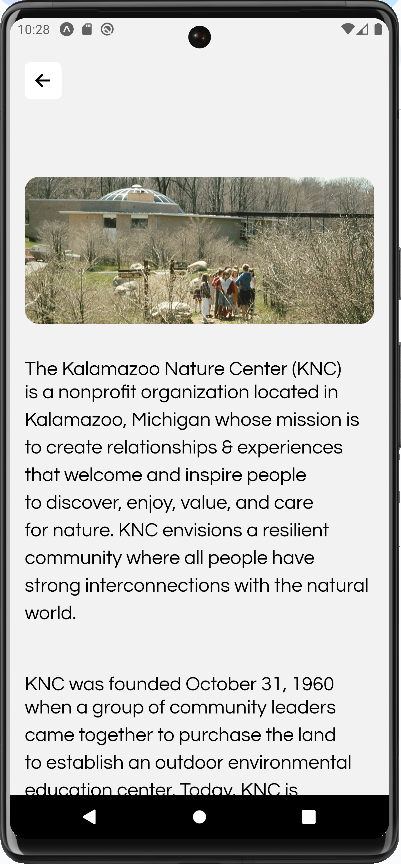
* 1. The App shall have five main screens: Discover, QRcode, Trails, Current Exhibits and Past Exhibits
     1. Each screen shall have a header with the KNC logo.
     2. Each screen shall have a footer that navigates users to the five main screens.
     3. Each image shall be cached in the phone when the app starts.



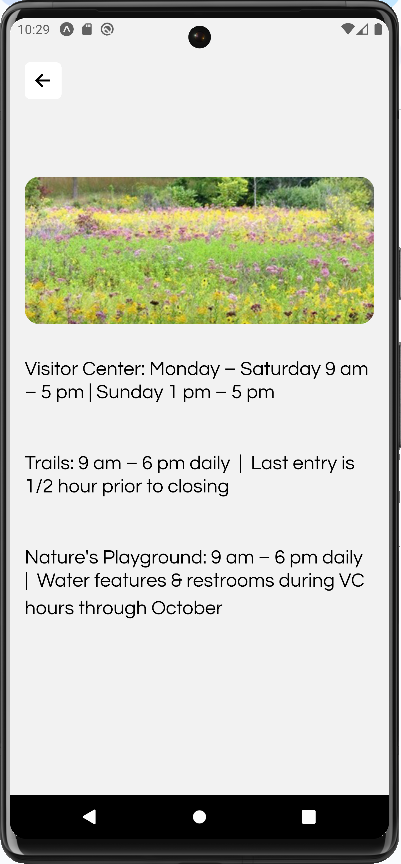




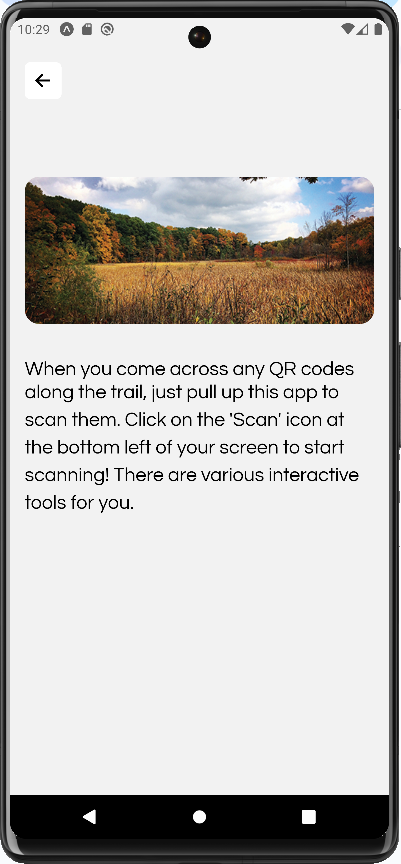
* 1. The Discover screen shall display a picture and 4 buttons, which are “About KNC”, “App Guide”, “Opening Hours”, and “Support”.
  2. The About KNC screen shall display an image along with paragraphs of description on the mission and origin of KNC.
     1. The page shall have a back button to navigate to the previous Discover page.



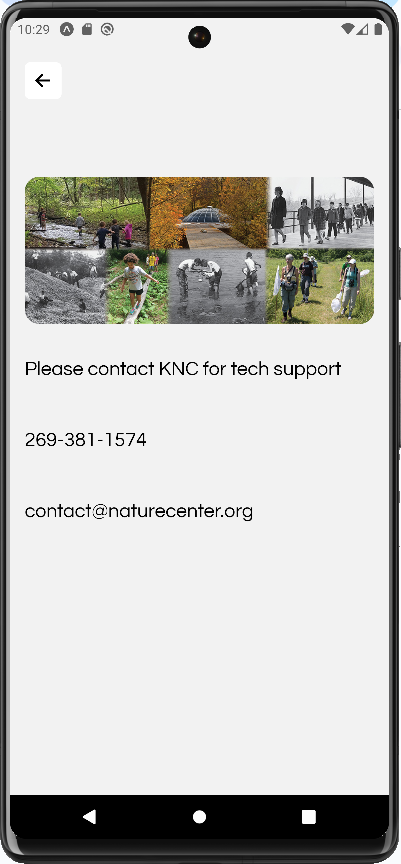
* 1. The Opening Hours screen shall display an image along with hours for the visitor center, trails, and nature’s playground.
     1. The page shall have a back button to navigate to the previous Discover page.



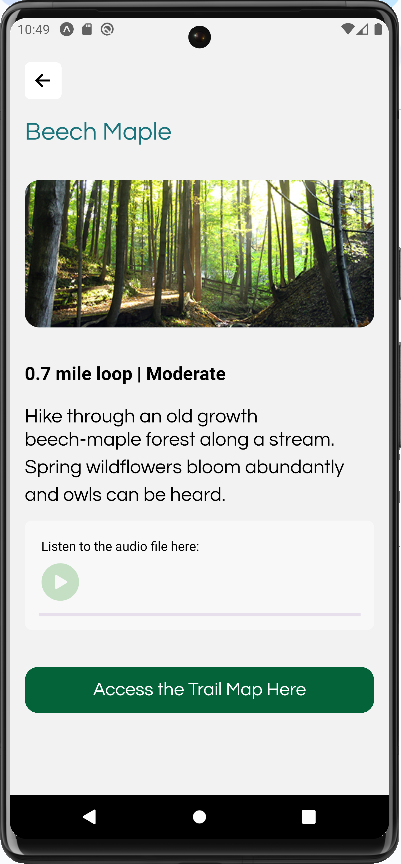
* 1. The App Guide screen shall display an image along with a brief description on how to use the app.
     1. The page shall have a back button to navigate to the previous Discover page.



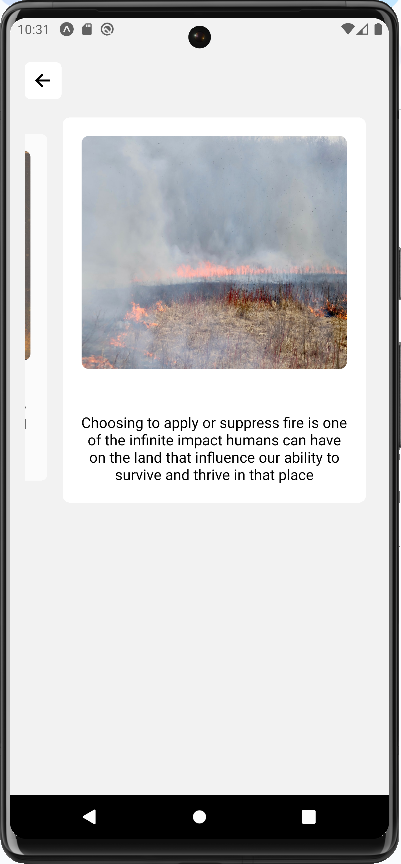
* 1. The Support screen shall display an image along with phone number and email for tech support.
     1. The page shall have a back button to navigate to the previous Discover page.



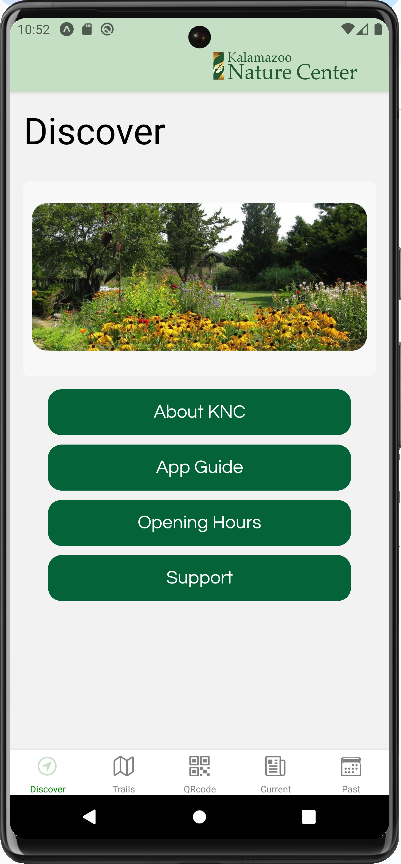
* 1. The Trail page shall display each individual trail name and picture.
     1. Each individual Trail screens shall display a title, image, loop length, difficulty, descriptions of the trail, an audio button and a trail map button.
     2. Data shall be loaded from Firebase when the screen is navigated.
     3. When users scan a QR code on their trail, the app shall redirect users to the specific trail page. The audio player in the corresponding page shall be playable.
     4. The page shall have an Access Trail Map button to navigate to the trail map.
     5. The Trail Map page shall display the Trail Map.
        1. The page shall have a back button to navigate to the previous Trail page.



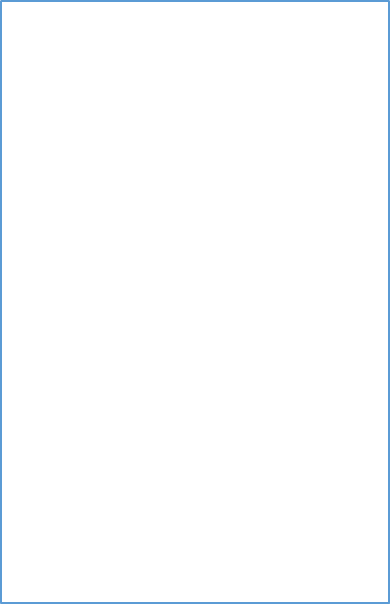
* 1. The Past Exhibits and Current Exhibits screens shall display each exhibit with an image and title.
     1. Data shall be loaded from Firebase when the screen is navigated.
     2. When each exhibit picture or title is clicked, users shall be redirected to a carousel of images. The images would be highlights from the exhibit, with a description of what the picture is showing.
     3. The page shall have an Access Story Map button to navigate to the story map in a browser.
     4. The page shall have a back button to navigate to the previous exhibits page.



* 1. Each screen shall be divided into three sections: navigation bar, content, and footer.



Navigation Bar



Scrollable Area with content



Footer with 5 page links

* + 1. The navigation bar shall have a KNC logo.
    2. The content shall contain information, except the QRcode screen.
    3. The footer shall have five icons that allow users to visit the corresponding screens: Discover, Trails, QRcode, Current, and Past.
  1. The App shall use a phone scanner to read a main QR code located near KNC Visitor Center.
     1. The App’s QR code scanner shall request access to the phones camera
     2. The App shall download all content needed for functioning upon downloads
  2. The App shall be able to read multiple QR codes along points of the trail (Point A, Point B etc.)
     1. The App shall pull up specific pages based on each QR code, with text/audio/visual files that users can play along the trail (used to be a corn harvesting land, beavers habitat, etc.).
     2. The App shall check validity of QR code.
        1. If valid the App shall open the page.
        2. If invalid the App shall display an error page.
  3. The App shall be able to work without WIFI/cell reception.
     1. The App shall call and play media files stored in a database.
  4. The App shall have navigable pages.
     1. Based on the QR Code, the app shall open a page but the user needs to be free to navigate.
  5. The App shall keep old exhibits in a usable format (JPEG / txt / link / mp3 / mp4).
     1. Archive past/current exhibits so users can access remotely.
        1. The past/current exhibits screens shall have pictures and titles of all exhibits.
        2. The past exhibits screen shall provide links to past exhibits.
        3. The current exhibits screen shall provide links to current exhibits.
  6. The App shall have an audio player.
     1. The audio player shall have a play/pause button.
     2. The audio player shall have a played progress bar.

1. Nonfunctional Requirements

**Performance requirements**

* 1. The app shall run at 60 fps to give an impression of smoothness.
  2. The frame rate shall be constant and not drop.
  3. The component load time shall be less than 10 seconds.
  4. The app shall not take more than 3 seconds to load the initial screen.
  5. The app size shall be less than 500 MB.

**Technical requirements**

* 1. The app shall run on both iOS and Android platforms.
  2. The app shall be able to work in different network conditions, such as Wi-Fi or cellular data, and handle network failures gracefully.
  3. The app shall be optimized for different screen sizes and resolutions to ensure a good user experience.

**Usability requirements**

* 1. The app shall respond quickly to user input by providing feedback in less than 1 second to provide a smooth user experience.
  2. The app shall maintain consistency in its design, language, and behavior to reduce confusion for users.
  3. The app shall provide clear and concise feedback to users, such as confirming a successful action or indicating an error.
  4. The app shall minimize the risk of user errors and provide clear instructions for resolving any issues that may arise.

**Scalability requirements**

* 1. The app shall be able to handle an increasing number of users and user accounts (1000+ users) without impacting performance or security.
  2. The app shall be efficient in its use of system resources, such as memory and CPU, to prevent crashes or slowdowns.
  3. The app shall be able to distribute workloads evenly across multiple servers or resources to ensure reliability and prevent bottlenecks.
  4. The app shall be able to deliver fast and responsive user experience even when the number of users or the amount of data processed increases.

1. Other Requirements

**External Interface Requirements**

* 1. The App shall interface with QR codes on outdoor signs.
  2. The App navigation shall be obvious to users.

**System Features**

* 1. The app must function with periodic internet connection/cell reception.
  2. The team shall provide documentation for maintenance and installation.