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 havean interesting and impressive history besides beinghome to a variety of animals and hundreds of wildflowers and other plant life. To improve the customer'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 designa series of artifacts thatcapture theimportant features such as software requirements, software architecture and test reports. In this report, a set of software requirementsare identified, which are organized in tabular formats for better visualization.

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

Students may implementthe softwarerequirements in this document in their senior project in various programming environments. Importantly, allrequirements 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 requirementsused throughout the KNC project; section 2 presents the non-functional requirements

1. **Introduction**
    - **1.1. Purpose**
        - 1.1.1. 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.
    - **1.2. Intended Audience**
        - 1.2.1. The Intended audience are people who are members of the KNC and come to walk on KNC trails.
    - **1.3. Intended Use**
        - 1.3.1. KNC employees: employees will eventually have access to edit content in the app and make changes with trails.
        - 1.3.2. WMU students developers: The students have access to the entirety of the app, and will follow the SRS by building an app in line with the goals of the KNC.
    - **1.4. Product Scope**
        - 1.4.1. Business Goals:
            - 1.4.1.1. Build and finish creating the app by the end of Fall 2023 Semester.
        - 1.4.2. Objectives:
            - 1.4.2.1. Finish SRS by the end of the Spring 2023 semester.

1.4.2.2. Start building app during the Spring 2023 semester, and continue throughout the summer and into the Fall of 2023

1.5. Definitions and Acronyms

1.5.1. KNC - Kalamazoo Nature Center

1.5.2. React Native - UI software framework used to develop Android and iOS applications

2. Overall Description

2.1. User Needs

2.1.1. Trail history and conservation: Information about the history and conservation of trails and the surrounding areas.

2.1.2. Trail mapping and navigation: The ability to view maps of trails and to easily find and follow a desired route.

2.1.3. Social features: The ability to connect with other users, share trail experiences, and find new friends to explore with.

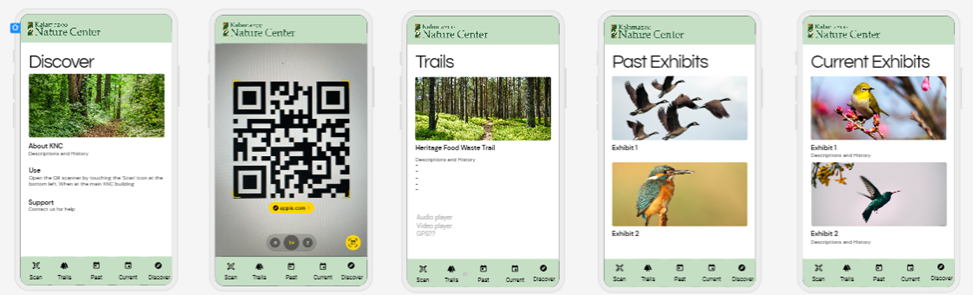
2.2. Assumptions and Dependencies

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

3. System Features and Requirements

3.1. Functional Requirements

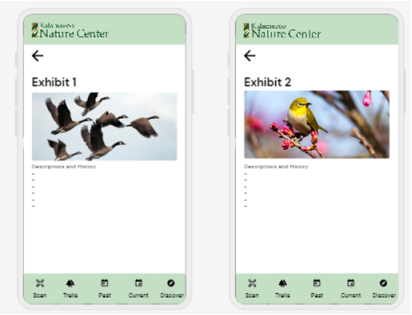
3.1.1. The App shall have five main screens: Discover, Scan, Trails, Past Exhibits and Current Exhibits



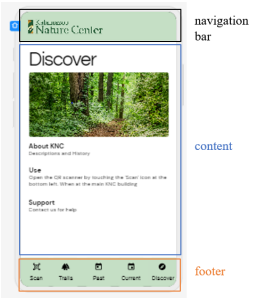
3.1.2. The Past Exhibits and Current Exhibits screens shall list out each exhibit with a picture and title.

3.1.2.1. When each exhibit picture or title is clicked, users shall be redirected to a specific page with descriptions and history of that exhibit.

3.1.2.2. The page shall have a back button to navigate to the previous exhibits page.



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



3.1.3.1. The navigation bar shall have a KNC logo.
3.1.3.2. The content shall contain information, except the Scan screen.
3.1.3.3. The footer shall have five icons that allows user to visit the corresponding screens: Scan, Trails, Past, Current, Discover

3.1.4. The App shall use a phone scanner to read a main QR code located near KNC Visitor Center.
3.1.4.1. The QR code shall lead users to download KNC App in the app store
3.1.4.2. The App shall download all content needed for functioning upon downloads
3.1.5. The App shall be able to read multiple QR codes along points of the trail (Point A, Point B etc.)
3.1.5.1. 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.)
3.1.5.2. The App shall check validity of QR code
3.1.5.2.1. If valid the App shall open the page
3.1.5.2.2. If invalid the App shall display error page
3.1.6. The App shall be able to work without WIFI/cell reception
3.1.6.1. The App shall call and play media files stored in a database
3.1.7. The App shall have navigable pages
3.1.7.1. Based on the QR Code, the app shall open a page but the user needs to be free to navigate

3.1.8. The App shall keep old exhibits in a usable format (JPEG / txt / link / mp3 / mp4)
3.1.8.1. Archive past/current exhibits so users can access remotely
3.1.8.1.1. The past/current exhibits screens shall have pictures and titles of all exhibits
3.1.8.1.2. The past exhibits screen shall provide links to past exhibits
3.1.8.1.3. The current exhibits screen shall provide links to current exhibits
3.1.8.2. The App shall display audio/video links for user

3.1.9. The app shall have video interviews from youtube embedded on the content page for the foodways trail
3.1.9.1. This video will enlarge when clicked and take up the entirety of the user's screen.
3.1.10. The App shall have an audio player page
3.1.10.1. This audio player will be a pop up
3.1.10.2. The audio player shall have a play/pause and stop button
3.1.10.3. The audio player shall have a played progress bar

3.2. External Interface Requirements
3.2.1. The App shall interface with QR codes on outdoor signs
3.2.2. The App shall communicate with some kind of backend
3.2.2.1. The video and audio shall have to be downloaded
3.2.3. The App navigation shall be obvious to users

3.3. System Features
3.3.1. The app must function with periodic internet connection/cell reception
3.3.2. The team shall provide documentation for maintenance and installation
3.3.2.1. A document to install the development environment
3.3.2.2. A document that explains the code layout
3.3.2.3. A document explaining how to upload to the Google Play Store and Apple App Store

3.4. Nonfunctional Requirements
3.4.1. Performance requirements
3.4.1.1. The app shall run at 60 fps to give an impression of smoothness
3.4.1.2. The frame rate shall be constant and not drop
3.4.1.3. The component load time shall be less than 10 seconds
3.4.1.4. The app shall not take more than 3 seconds to load the initial screen
3.4.1.5. The app size shall be less than 500 MB

3.4.2. Technical requirements

3.4.2.1. The app shall run on both iOS and Android platforms
3.4.2.2. The app shall be able to work in different network conditions, such as Wi-Fi or cellular data, and handle network failures gracefully.
3.4.2.3. The app shall have minimum and recommended memory and storage requirements, such as RAM and internal storage.
3.4.2.4. The app shall be optimized for different screen sizes and resolutions to ensure a good user experience.

3.4.3. Usability requirements

3.4.3.1. The app shall respond quickly to user input by providing feedback in less than 1 second to provide a smooth user experience.
3.4.3.2. The app shall maintain consistency in its design, language, and behavior to reduce confusion for users
3.4.3.3. The app shall provide clear and concise feedback to users, such as confirming a successful action or indicating an error.
3.4.3.4. The app shall minimize the risk of user errors and provide clear instructions for resolving any issues that may arise.
3.4.3.5. The app shall offer personalized experiences, such as customizing the display or recommendations based on user preferences.

3.4.4. Scalability requirements

3.4.4.1. The app shall be able to handle an increasing number of users and user accounts (1000+ users) without impacting performance or security.
3.4.4.2. The app shall be efficient in its use of system resources, such as memory and CPU, to prevent crashes or slowdowns.
3.4.4.3. The app shall be able to distribute workloads evenly across multiple servers or resources to ensure reliability and prevent bottlenecks.
3.4.4.4. 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