

Multidimensional Data Visualization in iOS Using Google Cardboard

Software Requirements Specification

Version 2

8/2/2017

Submitted to

Dr. Razib Iqbal
Assistant Professor of Computer Science
Missouri State University

in partial fulfillment of the
CSC450 Course Project Deliverables

Summer 2017

Team Information

Team: 01

Members: Ryan Bagby, Chad Brewer,
Leonard Museau, Darren Williams

Contact Information:

Ryan Bagby – bagby777@live.missouristate.edu

Chad Brewer – chad5972@live.missouristate.edu

Leonard Museau –

Museau32613@live.missouristate.edu

Darren Williams –

Darren224@live.missouristate.edu

Revision History

Date	Description	Comments
6/25/17	Version 1	N/A
7/2/17	Version 2	Final revision

Document Approval

The following Software Requirements Specification has been accepted and approved by the following stakeholders:

Signature	Printed Name	Title	Date
	Chad Brewer Leonard Museau Ryan Bagby	Technical Writers	7/2/17
	Darren Williams	Interim Project Manager	7/2/17
	Dr. R. IQBAL	Instructor/Customer	7/2/17

Table of Contents

REVISION HISTORY	II
DOCUMENT APPROVAL	II
1. INTRODUCTION	7
1.1 PURPOSE	7
1.2 SCOPE	7
1.3 DEFINITIONS, ACRONYMS, AND ABBREVIATIONS	7
1.4 REFERENCES	8
1.5 OVERVIEW	8
2. GENERAL DESCRIPTION	8
2.1 PRODUCT PERSPECTIVE	8
2.2 PRODUCT FUNCTIONS	8
2.3 USER CHARACTERISTICS	9
2.4 GENERAL CONSTRAINTS	9
2.5 ASSUMPTIONS AND DEPENDENCIES	9
3. SPECIFIC REQUIREMENTS	9
3.1 EXTERNAL INTERFACE REQUIREMENTS	9
3.1.1 <i>User Interfaces</i>	9
3.1.2 <i>Hardware Interfaces</i>	12
3.1.3 <i>Software Interfaces</i>	12
3.1.4 <i>Communications Interfaces</i>	12
3.2 FUNCTIONAL REQUIREMENTS	12
3.2.1 <i><Functional Requirement or Feature #1></i>	12
3.2.2 <i><Functional Requirement or Feature #2></i>	13
3.3 USE CASES	13
3.3.1 <i>Use Case #1</i>	13
3.3.2 <i>Use Case #2</i>	14
3.3.3 <i>Use Case #3</i>	
3.3.4 <i>Use Case #4</i>	
3.5 NON-FUNCTIONAL REQUIREMENTS	17
3.5.1 <i>Performance</i>	17
3.5.2 <i>Reliability</i>	17
3.5.3 <i>Availability</i>	17
3.5.4 <i>Security</i>	17
3.5.5 <i>Maintainability</i>	18
3.5.6 <i>Portability</i>	18
3.6 INVERSE REQUIREMENTS	18
3.7 DESIGN CONSTRAINTS	18
3.8 LOGICAL DATABASE REQUIREMENTS	18
3.9 OTHER REQUIREMENTS	18
4. ANALYSIS MODELS	19
4.1 SEQUENCE DIAGRAMS	19
4.2 DATA FLOW DIAGRAMS (DFD)	23
4.3 STATE-TRANSITION DIAGRAMS (STD)	23
5. CHANGE MANAGEMENT PROCESS	24

A. APPENDICES	25
A.1 APPENDIX 1	25
A.2 APPENDIX 2	25

1. Introduction

The introduction to the Software Requirement Specification (SRS) document should provide an overview of the complete SRS document. While writing this document please remember that this document should contain all of the information needed by a software engineer to adequately design and implement the software product described by the requirements listed in this document. (Note: the following subsection annotations are largely taken from the IEEE Guide to SRS).

1.1 Purpose

The purpose of this SRS is to provide a detailed overview of our software product and its parameters and goals. This document describes our project's target audience and its user interface and hardware and software requirements. This software is open for the general public who would like to know more information of common birds in a certain location, given to us by the Computer Science department. This software allows users to identify bird data through virtual reality. The program is modified so the user can have a immersive experience while processing assets in the program.

1.2 Scope

By the end of the summer semester this project will have produced a completed GIS Summer Demo application. This product will allow the user to select data points in a map which represent southwestern Missouri. These data points will display, to the user, relevant information about this bird that is most common in the location based on data sent to the Computer Science department from the Audubon Society of Missouri. All of this information will be displayed to the user in an immersive environment using mobile virtual reality on both Android and the IOS platforms.

1.3 Definitions, Acronyms, and Abbreviations

Table 1 - Definitions

Term	Definition
User	A person who will be interaction with the application.
GIS	Geographic Information System
GPS	Global Positioning System
VR	Virtual Reality
Google Cardboard VR Headset	A simple VR headset that will work with the application to give the user a 3D experience.
iOS	Mobile operating system developed by Apple Inc.
Android	A mobile operating system developed by Google, designed for touchscreen mobile devices.
Unity Game Engine	A cross-platform game engine developed by Unity Technologies.
Asset	The application's models that the user will be able to interact with.

CSV	A simple file format used to store tabular data. Stands for "Comma Separated Values."
-----	---

1.4 References

N/A

1.5 Overview

The Software Requirements Specification document contains the intended functions, constraints, specific requirements, functional requirements, non-functional requirements, and analysis models for the, "Multidimensional Data Visualization in IOS Using Google Cardboard" application, respectively.

2. General Description

As of now the application is usable on both Android and iOS devices. Either portable device can be placed inside of a Google Cardboard device and enable the user to experience the application. The user is also able to experience an immersive forest environment while wearing the Google Cardboard VR headset.

2.1 Product Perspective

This product is supposed to be an open source, under the Attribution-Noncommercial License. It's a VR system implementing gyroscopes and motion sensors for tracking head, hand, and body positions when exploring certain locations for birds. The system also provides small screens for stereoscopic displays and uses lightweight fast processors.

The following are main features that are included in the Birdboard:

- 1.) Multiplatform Support: Support for both Android and iOS
- 2.) Menu Screen: A screen that gives different options to explore program.
- 3.) Table View: The view of the map that allows users to choose a point of interest
- 4.) World View: The view of the area that was chosen from the Table View
- 5.) Instructions Page: A page that provides directions on how to use the program
- 6.) Credits Page: A page that provides credentials to everyone who worked on the software

2.2 Product Functions

The application will provide an immersive, 3D interactive diorama experience for the user. The user will be able to utilize VR technology to switch between menus and experiences in the mobile application. The user will be able to choose a GIS point and when the point is clicked, the user will be transported to the point in order to view various information about the birds that can be found in the area. While at the point, the user can switch between bird information in order to view other information about the birds that may appear in the area they have been virtually

transported to. The product will be able to work on devices that run on the iOS operating system in addition to Android devices.

2.3 User Characteristics

Users of the application should have no physical or health issues related to the viewing and/or interacting with a virtual environment.

Users must use an Android or iOS operating system, and should interact and explore the virtual forest environment to aid in gaining knowledge about the included birds.

2.4 General Constraints

A constraint that developers will face are financial constraints when using Unity Game Engine. Unity provides a way for developers to obtain already made assets from their asset store but not all assets are free to use. However, developers can develop their own assets to use but additional time will need to be added to the project.

2.5 Assumptions and Dependencies

There are no factors that will affect the requirements stated in the SRS.

3. Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

A first-time user of the application should see the icon of the application with “Start” below the icon. The user will also have a white dot in the middle of their screen. This dot will be how the user interacts with the application’s menus and assets. In order to begin, the user needs to move their dot to the “Start” icon and click on it, see Figure 1.

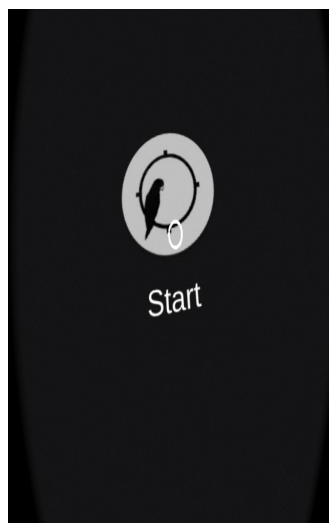


Figure 1 – Start Menu



Figure 2 – Main Menu



Figure 3 – Instructions

Multidimensional Data Visualization in iOS Using Google Cardboard

After starting, the user is presented with a menu with four different options. The options are “View Table”, “Instructions”, “Credits”, and “Exit.” If the user chooses the “Instructions” option, they are taken to another section that informs them on how to properly use the application, see Figure 3. If the user chooses the “Credits” option, they will be able to view information about who developed the application, see Figure 4. If the user chooses the “Exit” option, the application will shut down, see Figure 2.



Figure 4 – Credits

With the “View Table” option, the user is taken to an environment where a table is viewable. On the table, there are two menu options at the bottom called “Instructions” and “Back,” respectively. The “Instructions” option performs the same way as the option that is viewable in the previous menu. The “Back” option will return the user to the previous menu. There are several GIS pins located on a table that the user can interact with by focusing their dot on a pin and then by pushing the button located on the side of their VR device. Also on the table are blue and pink circles. If the user interacts with these circles, they can rotate the table to either the left or the right, see Figure 5.



Figure 5 – Table View

When the user selects a GIS pin, they are transported to the location of the pin and the bird information is displayed. The pin will be located in front of them with a forestry area surrounding them. Below the pin are two menu options, “Instructions” and “Back.” The “Instructions” option allows the user to view instructions on how to operate the application, as previously mentioned. The “Back” option will return the user to the overview of the table. If the user rotates their head or device, they are able to see forestry area and other points located on the map, see Figure 6.



Figure 6 – Bird Data View

3.1.2 Hardware Interfaces

Google Cardboard VR headset is recommended for the application to be viewed properly. The application is viewable without the VR headset but the purpose of the 3D technology will not be properly implemented. The application will be able to operate on both devices with the Android operating system and the iOS operating system.

3.1.3 Software Interfaces

The application will be built using the Unity Api. In it are several methods and classes that make physics and player input easy to use. Another Api the application must integrate is Google's GVR. This allows for virtual reality on a mobile device.

3.1.4 Communications Interfaces

There isn't any outside communication to be done. The application software package contains all of the necessary code to run the application, even without being inside the Google Card Board VR headset.

3.2 Functional Requirements

3.2.1 Enable the Application to Work on IOS

3.2.1.1 Introduction

We will enable the application to work on IOS devices by using the Unity Game Engine to port the application from Android to IOS.

3.2.1.2 Inputs

The application that currently runs on Android devices.

3.2.1.3 Processing

Removal of outdated dependencies.

3.2.1.4 Outputs

Multidimensional Data Visualization in iOS Using Google Cardboard

An application that runs on IOS devices.

3.2.1.5 Error Handling

3.2.2 Create an Immersive Experience for the User, Using Virtual Reality Technology

3.2.2.1 Introduction

The application already has a basic immersive experience, letting the user view bird information by clicking various points. We will add to that experience by adding structures (Trees, rocks, streams) by utilizing Unity Game Engines library of assets.

3.2.2.2 Inputs

The application's basic Virtual Reality experience

3.2.2.3 Processing

Addition of the assets from Unity.

3.2.2.4 Outputs

A more immersive experience giving the user a feeling of being in a forest setting.

3.2.2.5 Error Handling

3.3 Use Cases

3.3.1 Use Case #1

Use Case Name: View Immersive Forest in Table Scene

Context: Become immersed in a forest environment in the Table Scene view

Actor(s): User

Precondition(s): None

Main scenario steps:

1. The user inserts their phone into Google Cardboard.
2. The user initiates the application.
3. The user uses Gaze Click to select "Start" and view the first menu.
4. The user uses Gaze Click to view the GIS Table from the beginning menu.
5. The user is immersed in a forest environment while viewing the GIS table.

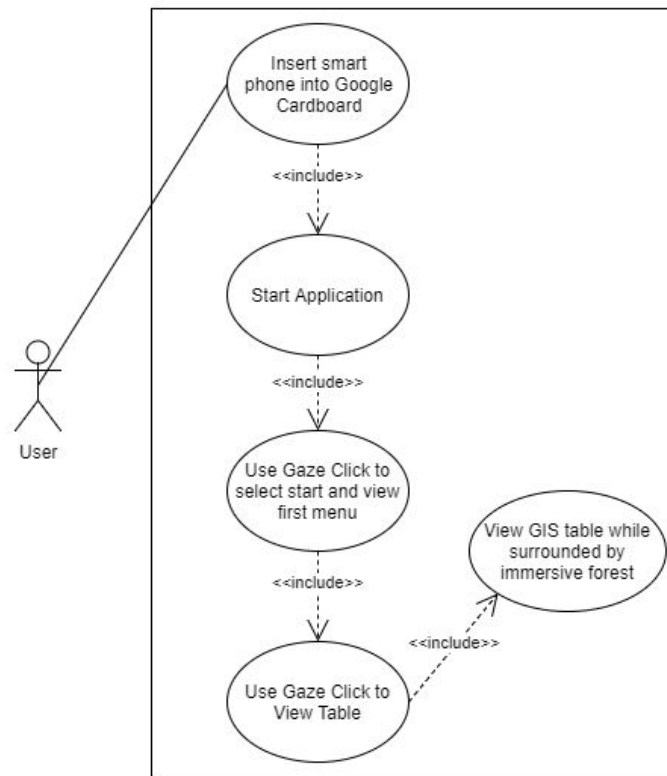
Alternative(s): None

Exception(s): None

Post Condition(s): View GIS table in an immersive forest environment.

Multidimensional Data Visualization in iOS Using Google Cardboard

Use Case Diagram: View GIS Table in Immersive Forest



3.3.2 Use Case #2

Use Case Name: View Updated Immersive World Scene

Context: Become immersed in a forest environment in the World Scene.

Actor(s): User

Precondition(s): The application is on and on the menu screen.

Main Scenario Step(s):

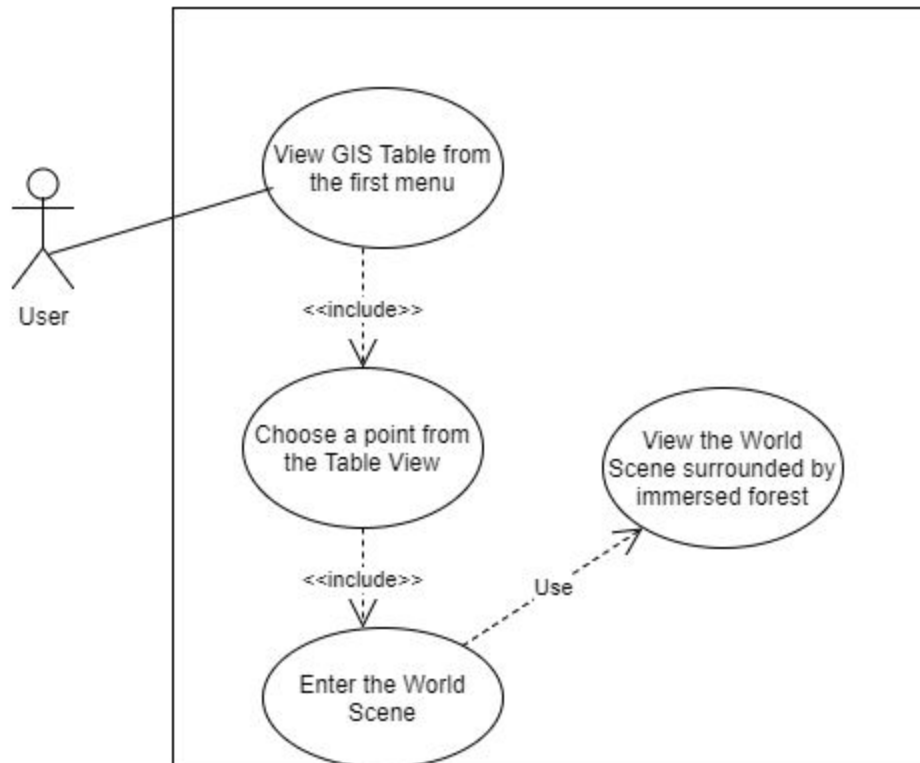
1. The user uses Gaze Click to view the GIS Table from the first menu.
2. The user uses Gaze Click to choose a point in the Table Menu.
3. The user enters the World Scene from the chosen point.
4. The user is immersed in a forest environment while viewing the World.

Alternative(s): None

Exception(s): None

Post Condition(s): View World Scene in a forest environment.

Use Case Diagram: View World Scene with Updated Forest



3.3.3 Use Case #3

Use Case Name: Toggle Background Music On/Off

Context: Enable music to add to the immersive environment.

Actor(s): User

Precondition(s): None.

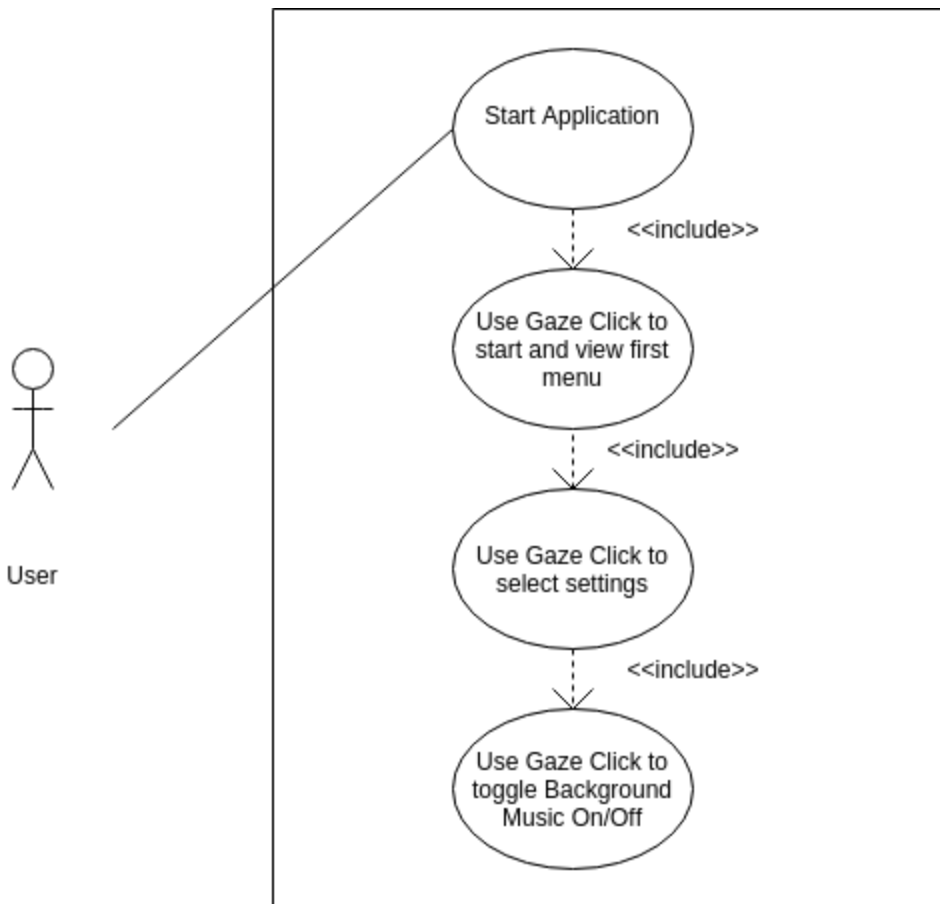
Main Scenario Step(s):

1. The user starts the application.
2. The user uses Gaze Click to view the first menu.
3. The user uses Gaze Click to choose the settings option.
4. The user uses Gaze Click to choose to toggle the Background Music On/Off.

Alternative(s): None

Exception(s): None

Post Condition(s): Background Music is On or Off depending previous setting.



3.3.4 Use Case #4

Use Case Name: Menu Interaction

Context: The user is able to traverse the app using the main menu

Actor(s): User

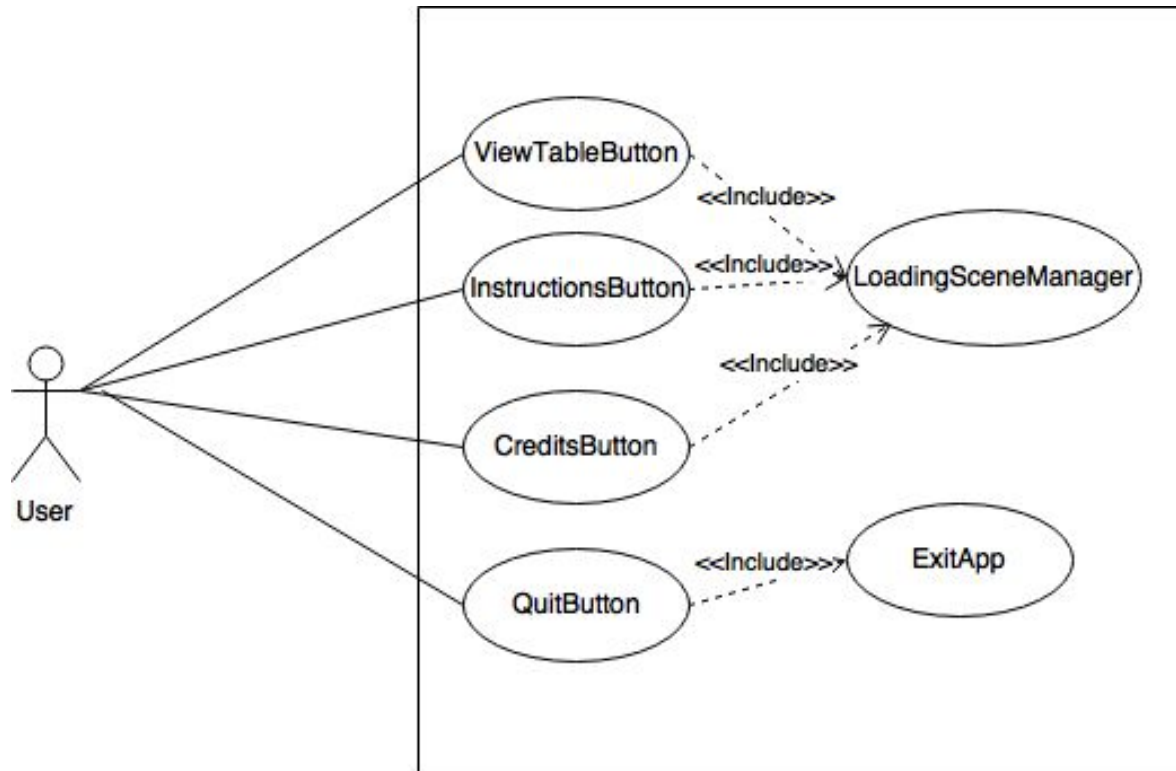
Precondition(s): User has clicked start on the splash screen.

Main Scenario Step(s):

1. The user starts the application.
2. The user uses Gaze Click to select a scene.
3. Scene is loaded into the background

Alternative(s): None

Exception(s): The User selects the Quit option



3.5 Non-Functional Requirements

3.5.1 Performance

The application will perform the same on iOS as it does on Android while wearing a Google Cardboard VR headset.

3.5.2 Reliability

The application must operate with the same features, reliably, across both Android and iOS platforms.

3.5.3 Availability

The implementation of the added features and assets will be available on both types of portable devices, iOS and Android.

3.5.4 Security

No security is necessary.

3.5.5 Maintainability

The code for the application must be written well with documentation and will be easily understood by other developers so that they may add new features or take away features.

3.5.6 Portability

The application will be able to be ported to either iOS or Android devices..

3.6 Inverse Requirements

N/A

3.7 Design Constraints

The previous use of Gaze Click will not be able to be implemented due to licensing issues and will need to be purchased to continue its use. There are no other limitations.

3.8 Logical Database Requirements

The original developers utilized CSV files to contain the images of the bird data.

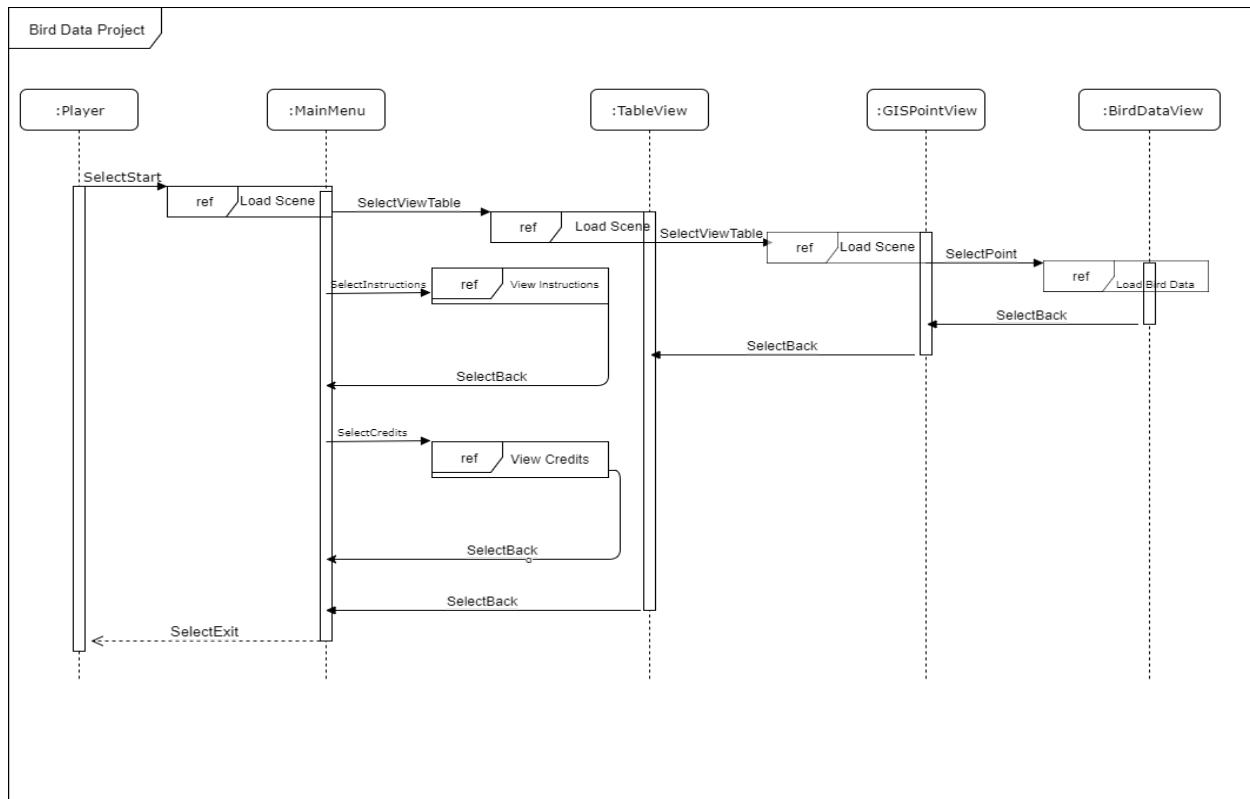
3.9 Other Requirements

There are no other requirements at this time.

4. Analysis Models

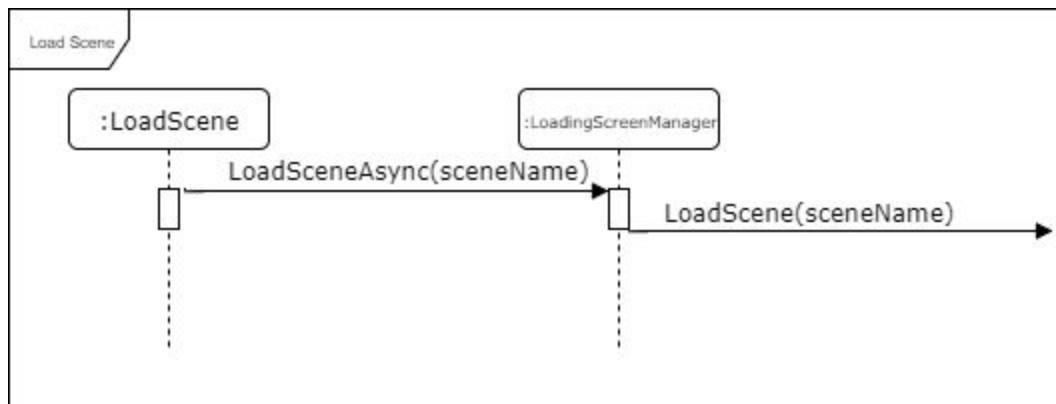
4.1 Sequence Diagrams

Bird Project Sequence Diagram



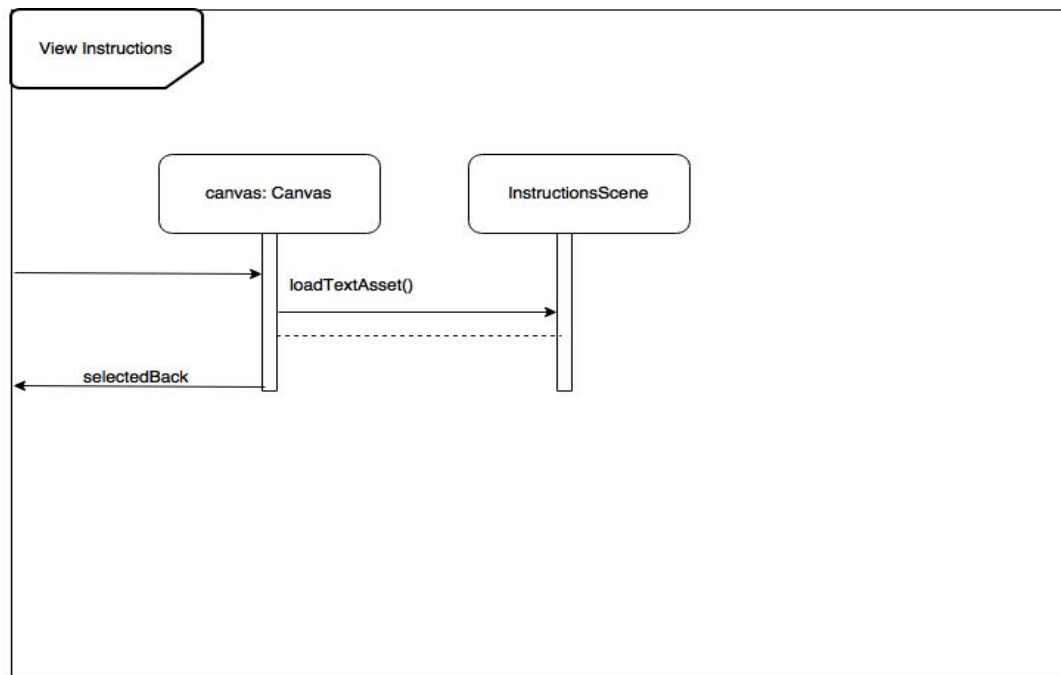
Description: The Bird Project Sequence Diagram shows the overall sequence of when the user interacts with the application. The user uses their Google Cardboard's choose option, such as click or press, to select start which brings up the main menu. At the main menu, the user may click again to either select Table View, Credits, Instructions, or Exit. When the user selects Instructions, they are brought to a scene where they are able to view the instructions on how to use the application. If they choose Credits, they are brought to a scene that allows them to view information about the developers of the application. They can select the Back option in order to go back to the main menu from the instructions or the Credits scenes. When the user selects the Table View, they are then brought to a scene that allows them to view the map with the GIS points on it. From the Table View, the user may select the Back option to return to the main menu or they may select a GIS point to go to the GIS point view. Once the user is in the GIS point view, they may select the point to view the bird data or they can select the back option to return to the Table View.

Load Scene Sequence Diagram:



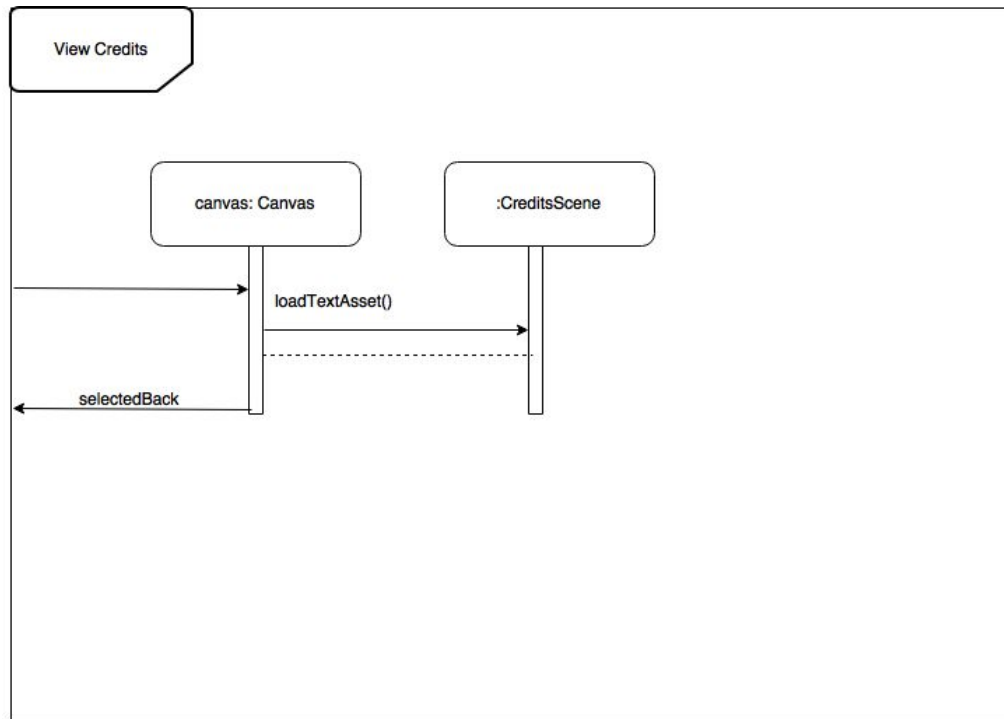
Description: The Load Scene is used in almost all scene transitions within the app. Its job is to make sure the user's screen isn't frozen while loading between scenes in the app.

View Instructions Sequence Diagram:



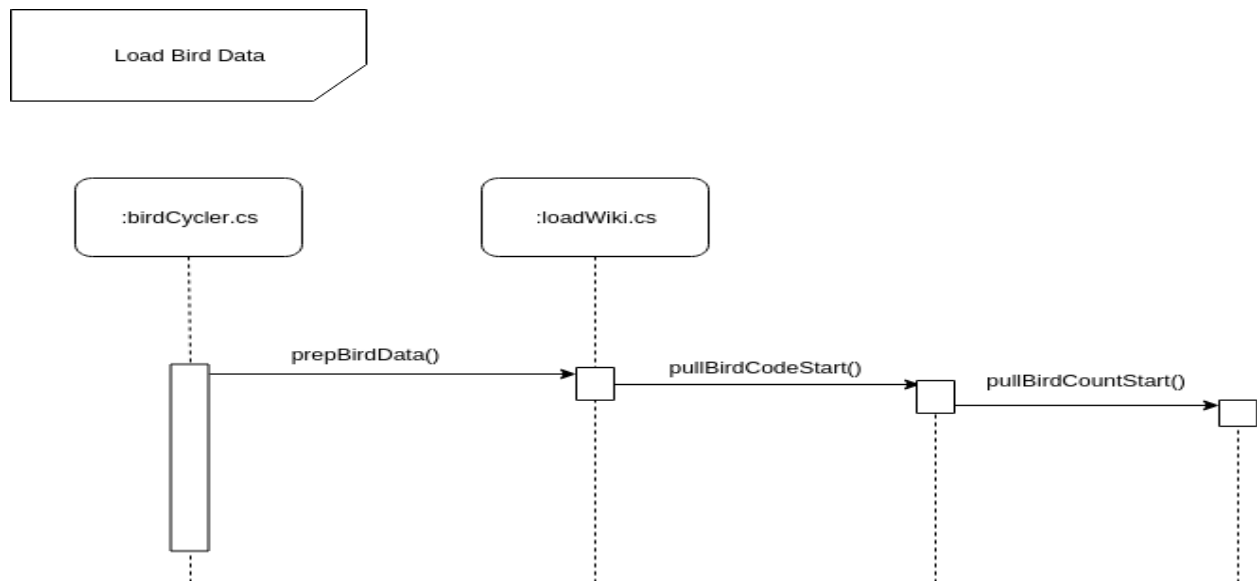
Description: The Canvas, which displays instructions for the user, will load the texts to be displayed on the screen after transitioning to the Instruction scene.

View Credits Sequence Diagram:



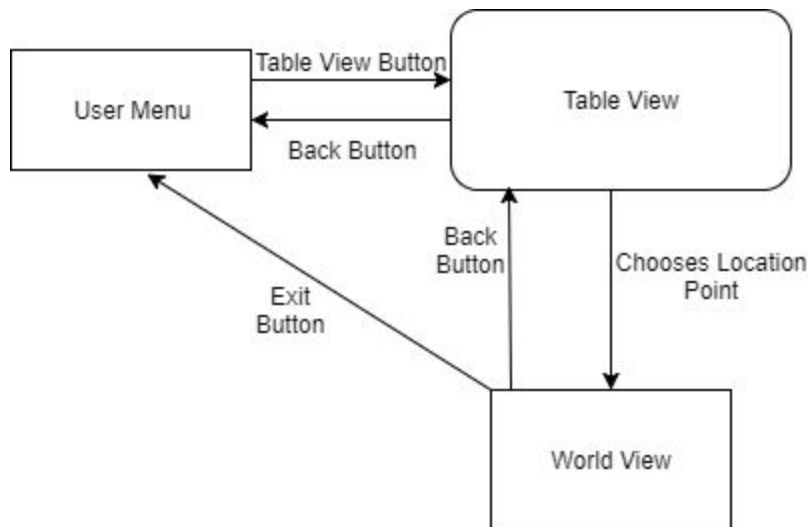
Description: The Canvas, which displays the credits the user, will load the texts to be displayed on the screen after transitioning to the credits scene.

Load Bird Data Sequence Diagram:



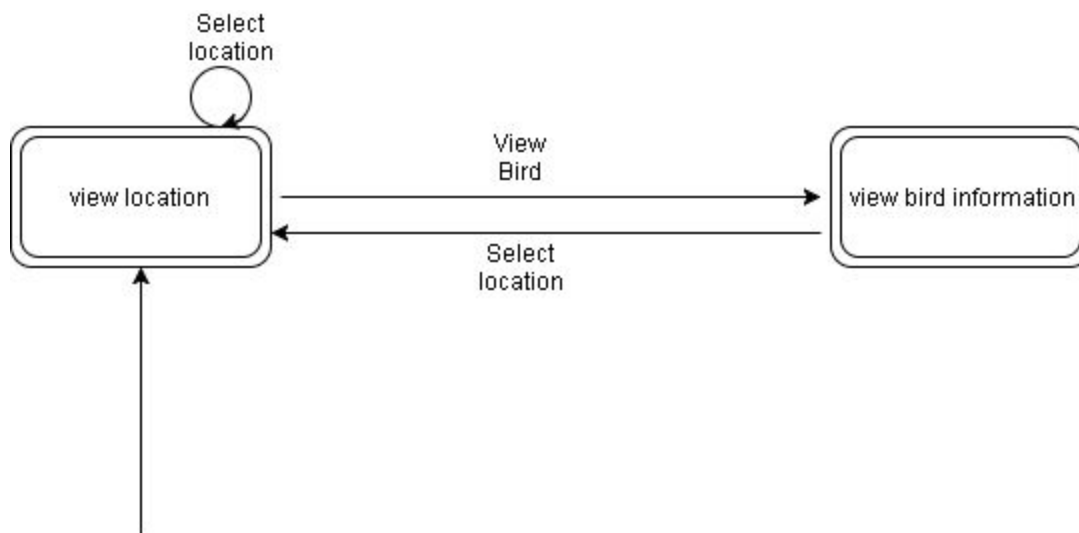
Description: After the user has loaded the World Scene the BirdCycler class will call a method which grabs data from the LoadWiki class and display this information on a canvas to the user.

4.3 Data Flow Diagrams (DFD)



Description: This is a graphical representation of the “flow” of data through this information system. This shows the general overview of our system without going into great detail.

4.2 State-Transition Diagrams (STD)



Description: View location and view bird information are both accepting states which can transition to the other via selecting location and viewing a bird respectively.

5. Change Management Process

All changes to this document can be initiated by any team member. The Scrum Master (Darren Williams) will need to approve any appending changes brought forward by a team member. Finally, the Technical Writer (Chad Brewer) will implement these changes into the SRS document with one final review from the Scrum Master.