Florida International University
School of Computing and Information Sciences

Software Engineering Focus

Final Deliverable

Project Title: Learning with Augmented Reality

Team Members: Jeffrey Perez, Luis Perera

Product Owner(s): Dr. Shahin Vassigh

Mentor(s): Dr. Francisco Ortega, Mohsen Taheri

Instructor: Dr. Masoud Sadjadi

Abstract

This document presents the information necessary to gain a good understanding of how the Learning with Augmented Reality project was managed. The project aims to improve the learning process of students of the School of Architecture. The first steps to accomplish this were to identify the problem to be solved and the requirements of the system. Through scrum development, the application was developed in 7 sprints of 2 weeks each. A new set of features were added to the app to improve user experience and visual stimulation in order for students to have a good learning experience. To make it accessible, two applications under the name SKOPE-AR were deployed for Android and iOS smartphones.

Table of Contents

33

INTRODUCTION

The Learning with Augmented Reality project emerged from the desire of enhance learning using technology. With the integration of Building Information Modeling (BIM), Augmented Reality (AR), visual simulations, and interactive lessons, this project aims to support interdisciplinary learning for building sciences.

The new functionalities that needed to be added to the project required from the application to have different behaviors depending on the user interaction. Towards the end of the project, it was important to create and upload to the cloud a user compliance SKOPE-AR application for mobile phones to make it accessible for everyone.

Current System

The current application displays a 3D model of the SIPA building while capturing real time image with the rear camera of the device. To make easier the user navigation through the app and understandable, new panels and description of most important parts of the model were implemented. The application also gives the option of stop and resume capturing video with the camera or stop the getting data from the gyroscope so the model stays in place. A movie player was implemented for some elements of the model for the students to easily assimilate the information. To finalize, the application can be found in Play Store for Android phones and iOS for iPhone.

Purpose of New System

SKOPE-AR provides a model with information of the SIPA building to help the Architecture Department to teach core building science concepts. The application enhances learning by showing descriptions and videos of specific design and construction aspects of the building.

USER STORIES

The following section provides the detailed user stories that were implemented in this iteration of the Learning with Augmented Reality project. These user stories served as the basis for the implementation of the project's features. This section also shows the user stories that are to be considered for future development.

Implemented User Stories

User Story #198 - Understanding Unity and Project
User Story #199 - Understanding Unity and Project
User Story #184 - SKOPE AR Bug fixes
User Story #185 - SKOPE AR create and upload build to website
User Story #186 - SKOPE AR Update labeling and annotation diagram
User Story #187 - SKOPE AR User Interface Navigation
User Story #188 - SKOPE AR State Machine for UI
User Story #189 - SKOPE AR Setting up cameras
User Story #190 - SKOPE AR Upload for Android
User Story #191 - SKOPE AR Panel and Image Description
User Story #192 - Upload SKOPE AR for iOS
User Story #193 - SKOPE AR Panel Description Update
User Story #194 - SKOPE AR Fix Panel
User Story #195 - SKOPE VR Movie Player Implementation
User Story #196 - SKOPE VR Improve Exterior Diagrams
User Story #197 - SKOPE AR Implement Movie Player

User Story #197 - Understanding Unity and Project

Description:

- As a developer, I must install and understand the necessary software to begin assigning tasks.

Acceptance Criteria:

1. Install Unity
2. Learn Unity
3. Become Familiar with the SKOPE AR and SKOPE VR projects

User Story #198 - Understanding Unity and Project

Description: Understanding Unity and Project

- As a developer, I must install and understand the necessary software to begin assigning tasks.

Acceptance Criteria:

1. Install Unity
2. Learn Unity
3. Become familiar with SKOPE AR and SKOPE VR projects

User Story #184 - SKOPE AR Bug fixes

Description:

- As a developer, I want to fix all the bugs related to all interfaces, diagrams and animations so that the end user will have a better experience when using the software

Acceptance Criteria:

1. The system needs to run error free.
2. Diagrams must display properly.
3. Ensure the animations work properly.

User Story #185 - SKOPE VR create and upload build to SKOPE website

Description:

- As a developer, I want to create and upload a successful build of the software to the SKOPE website so that it can be accessible on Internet.

Acceptance Criteria:

1. Create a build of the SKOPE VR project
2. Make available a Google Drive link of the project in the SKOPE website

User Story #186 - SKOPE VR Update labeling and annotation of diagrams and animations

Description:

- As a user, I want to see the updated version of all the labeling, annotation diagrams and animations so that I can have a better understanding.

Acceptance Criteria:

1. All labels, diagrams and animation have up-to-date text and non-empty.

User Story #187 - SKOPE AR User Interface Navigation

Description:

- As a user, I want to navigate through the menus of the application so that I can select the viewing options that will be displayed in the interface.

Acceptance Criteria:

1. The user can navigate through the UI and select the option he/she wants.
2. The correct panel will be displayed depending on what the user touches.
3. The previous panel is closed and the next one is opened.

User Story #188 - SKOPE AR State Machine for UI

Description:

- As a user, I want to have the right options I can do on certain interface so that I can see the details of the structures based on the option I chose.

Acceptance Criteria:

1. State Machine created to navigate through different options (states) displayed on the interface.
2. Change from one state to the other depending on what option the user selected.

User Story #189 - SKOPE AR Setting up Cameras

Description:

- As a user, I want to be able to see the building model with the cellphone/tablet camera as background so that I can have a good and fast learning experience.

Acceptance Criteria:

1. Set up an interface camera with all panels.
2. Set up an AR camera to display the building model
3. Set up a background camera to display the content the device is capturing in real time.

User Story #190 - SKOPE AR Upload for Android

Description:

- As a user, I want to have access to the SKOPE AR app on Google Play Store so that I

can install it on my phone and try it.

Acceptance Criteria:

1. Create a stable build of the SKOPE AR app.
2. Correct some errors in the code and in the UI.
3. Set up Google Play account, fill up all the information and upload the app.

User Story #191 - SKOPE AR Panel and Image Description

Description:

- As a user, I want to see a description of a model or image displayed when I choose an option so that I have a better understanding of what I'm watching.

Acceptance Criteria:

1. A file is created to pull all the information from it.
2. Store the name and the description (pulled from the file) in a dictionary, where name is the key.
3. Set the description text based on the current state of the state machine and based on the panel or image displayed.

User Story #192 - SKOPE AR Upload for iOS

Description:

- As a user, I want to have access to the SKOPE AR app on the Apple Store so that I can install it on my phone and try it.

Acceptance Criteria:

1. Create a stable build of the SKOPE AR app.
2. Set up Apple Store Developer Console.
3. Upload application to the Apple Store.

User Story #193 - SKOPE AR Panel Description Update

Description:

- As a user, I want to be able to read and see the information about a particular component of the building in a fancier way so that I can learn faster with things I like.

Acceptance Criteria:

1. The information must be displayed when the user clicks new buttons created handled by the new class IconStateAR created.
2. The information needs to be accurate at all times.

User Story #194 - SKOPE VR Movie Player

Description:

- As a user, I want to be able to click on specific icons in the game world to watch a short video clip of a particular component so that I can see what's inside that part of the building.

Acceptance Criteria:

1. When the user clicks the icon the movie player should play a video.
2. The video being played must be the correct video.
3. All three videos need to be added to the world.
4. Video should stop when the panel is toggled off and start again when the panel is toggle on.

User Story #195 - SKOPE AR Fix Panel

Description:

- As a user, I should be able to smoothly transition between panels**.**

Acceptance Criteria:

1. All the buttons on the panel must be linked to the correct panel.
2. The user should have no issue transitioning between panels through the buttons.

User Story #196 - SKOPE VR Improve Exterior Diagrams

Description:

- As a user, I should have an easy time reading and understanding the diagrams displayed in the world.

Acceptance Criteria:

1. Diagram needs to be correct and in the right position.
2. Clicking on the icon should only toggle one diagram.

User Story #199 - SKOPE AR Movie Player

Description:

- As a user, I want to be able to click on expand icons and in the window, that opens touch the movie button to watch an informational video.

Acceptance Criteria:

1. When the user touches, the movie button a video should be played in full screen with controls.
2. The user should be returned to SKOPE AR after the film is done or the user tries to go back.
3. The two educational video clips should be added to their corresponding expand icon.

Pending User Stories

User Story #200 - Point B implementation

Description:

- As a user, I want to be able to select a new location in which I can see the SIPA building from another angle.

Acceptance Criteria:

4. Set a new camera in Unity.
5. Modify the UI to show the new point
6. Update the BetaState script.

Project Plan

This section describes the planning that went into the realization of this project. This project incorporated the agile development techniques and as such required the sprints to be planned. These sprint plannings are detailed in the section. This section also describes the components, both software and hardware, chosen for this project.

Hardware and Software Resources

The following is a list of all hardware and software resources that were used in this project:

Hardware:

- Asus Laptop with Windows 10
- HP Laptop with Windows 10
- iMac OS X
- Samsung Phone
- IPhone SE

Software:

- Unity 3D 5.5.1
- Microsoft Visual Studio 14
- XCode 8
- Mingle
- Google Drive
- GitHub

Sprints Plan

Sprint 1

- User Stories **#197** and **#198**
- Install and learn Unity
- Import SKOPE VR/AR Project to Unity
- Understand the SKOPE VR and SKOPE AR project

Sprint 2

- User Stories **#184**, **#185**, and **#186**
- Fix bugs in the SKOPE VR system.
- Use Unity to build the project and upload it to the SKOPE site.

Sprint 3

- User Stories **#187**, **#188**, and **#189**
- Setup User Interface for the user to navigate through panels.

- Create a State Machine to handle the panels to be displayed.
- Setup the camera for the Augmented Reality aspect.

Sprint 4

- User Stories **#190** and **#191**
- Build the project to upload SKOPE AR to the Google Play Store.
- Link the images and description to their corresponding panel.

Sprint 5

- User Stories **#192** and **#193**
- Create a stable build to upload to the Apple Store.
- Fix the panel description but for the SKOPE AR project.

Sprint 6

- User Stories **#194**, **#195**, **#196**, and **#199**
- Added an icon and in world movie player for the SKOPE VR project.
- Correct an issue with the description of the SKOPE AR panels.
- Added a movie player to the SKOPE AR project.

SYSTEM DESIGN

This section contains information on the design decisions that went into this project. The architecture patterns are outlined and explained. The entire system is shown in a package diagram and the subsystems are explained. Finally, the design patterns used in the project are discussed.

Architectural Patterns

The architectural pattern used in this project was the Three-Tier Architecture. With this pattern, we can keep the system organized and well defined. In the top tier, can be found the Presentation Tier, which is the one that handles all graphics and modeling. A game object named "GameManager" and containing most important scripts was set up in Unity, making this the

Application Tier, the one that handles the logic of the system. Finally, the Data Tier was defined, and it uses the file system of the device to load all the descriptions to the panels.

System and Subsystem Decomposition

This section contains information on the minimal class diagram which can be found in Appendix A. Listed below is a brief explanation of the functionality of each subsystem.

- **IARState**: interface that will establish the behavior of the states **MainStateAR**, **AlphaState**, **BetaState** and **IconStateAR**.
- **StatePatternAR**: controller that will keep track on which state the application is. It is also in charge of transitioning from one state to the other.
- **MainStateAR**: this is the state that will be running when the application start. Here you can only go back or select any location.
- **AlphaState**: state in which the application will be when Point A is selected. From here you can go back to **MainStateAR** or go into **IconStateAR**.
- **BetaState**: state in which the application will be when Point B is selected. From here you can go back to **MainStateAR** or go into **IconStateAR**.
- **IconStateAR**: state that will be loaded when you click on an icon in the model. In this state, you can either go back to the previous state (**AlphaState** or **BetaState**), click on Diagram button to show image and description or play a video. The description is loaded using the Description class.
- **Description**: class in the Data Tier that oversees loading all the descriptions from a text file to a dictionary.
- **GyroCamera**: modified class that controls the AR camera by reading data from the gyroscope of the device and applying the rotation needed to move the model in the application.
- **BGCamScript**: modified class that captures video using the rear camera and displays it in the application as a background.
- **MovieTexturePlayer**: class used to play the videos when the user clicks the Movie button in the **IconStateAR**.

Deployment Diagram

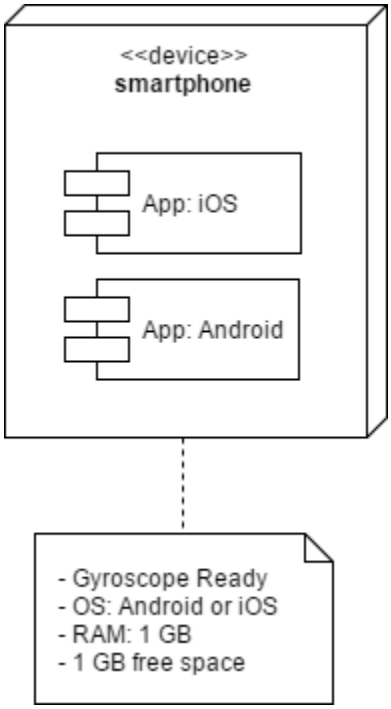

Figure 1. Deployment Diagram

Design Patterns

There was only one design pattern used in the application. Because we only wanted to have only one instance of the Description class, we implemented the Singleton Pattern here. Every time, before calling the method to set the description, a new instance of the class was created and, therefore, the application would have to load the descriptions to the dictionary again. With the Singleton Pattern, only one instance is created, resulting in loading the data from the file system only once and improving performance.

SYSTEM VALIDATION

The system was validated using unit testing and system testing on rainy days and sunny days.

For unit testing was used the library NSubstitute.dll to create instances of the elements we need, which is like create mocks and stubs but what this library does is create a substitute instance of the class we need. For system testing we created test cases and tested the application on both iOS and Android devices.

Test Case ID	UT_DESC_001
Purpose	Verify that the dictionary is populated with the right data
Preconditions	File with description and name of the panel must exist
Input	Address of the description file
Expect Output	Dictionary is populated with the data

Test Case ID	UT_DESC_002
Purpose	Verify that the description is properly set to the panel
Preconditions	User is in the IconState and dictionary is populated
Input	User touches the Diagram button
Expect Output	Description of the panel is properly set according its name

GLOSSARY

Android: A mobile operating system developed by Google.

iOS: A mobile operating system created and developed by Apple Inc., exclusively for its hardware.

Unity / Unity 3D: A cross-platform game engine developed by Unity Technologies and used to develop video games for PC, consoles, mobile devices and websites.

State Machine: An abstract machine that can be in exactly one of a finite number of states at any given time and you can transition from one state to another in the event of some external input.

Augmented Reality (AR): A live direct or indirect view of a physical, real-world environment whose elements are augmented (or supplemented) by computer-generated sensory input such as sound, video, graphics or GPS data.

Virtual Reality (VR): A computer technology used for headsets to generate realistic images, sounds, and other sensations that replicate real environment or create an imaginary setting.

Google Play Store: A digital distribution service operated and developed by Google.

App Store: A digital distribution platform, developed and maintained by Apple Inc., for mobile apps on its iOS operating system.

APPENDIX

Appendix A - UML Diagrams

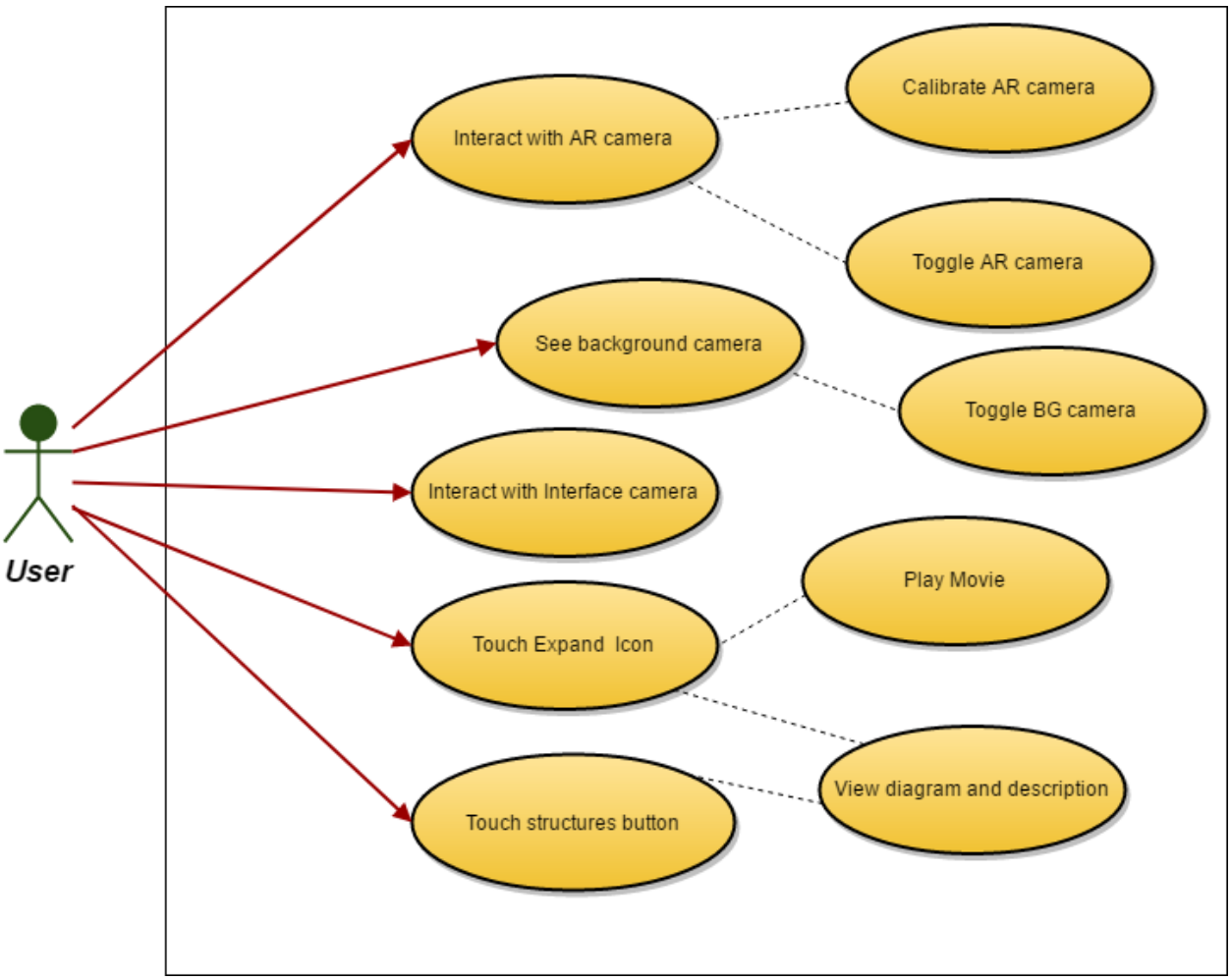

Figure 2. Use Case Diagram

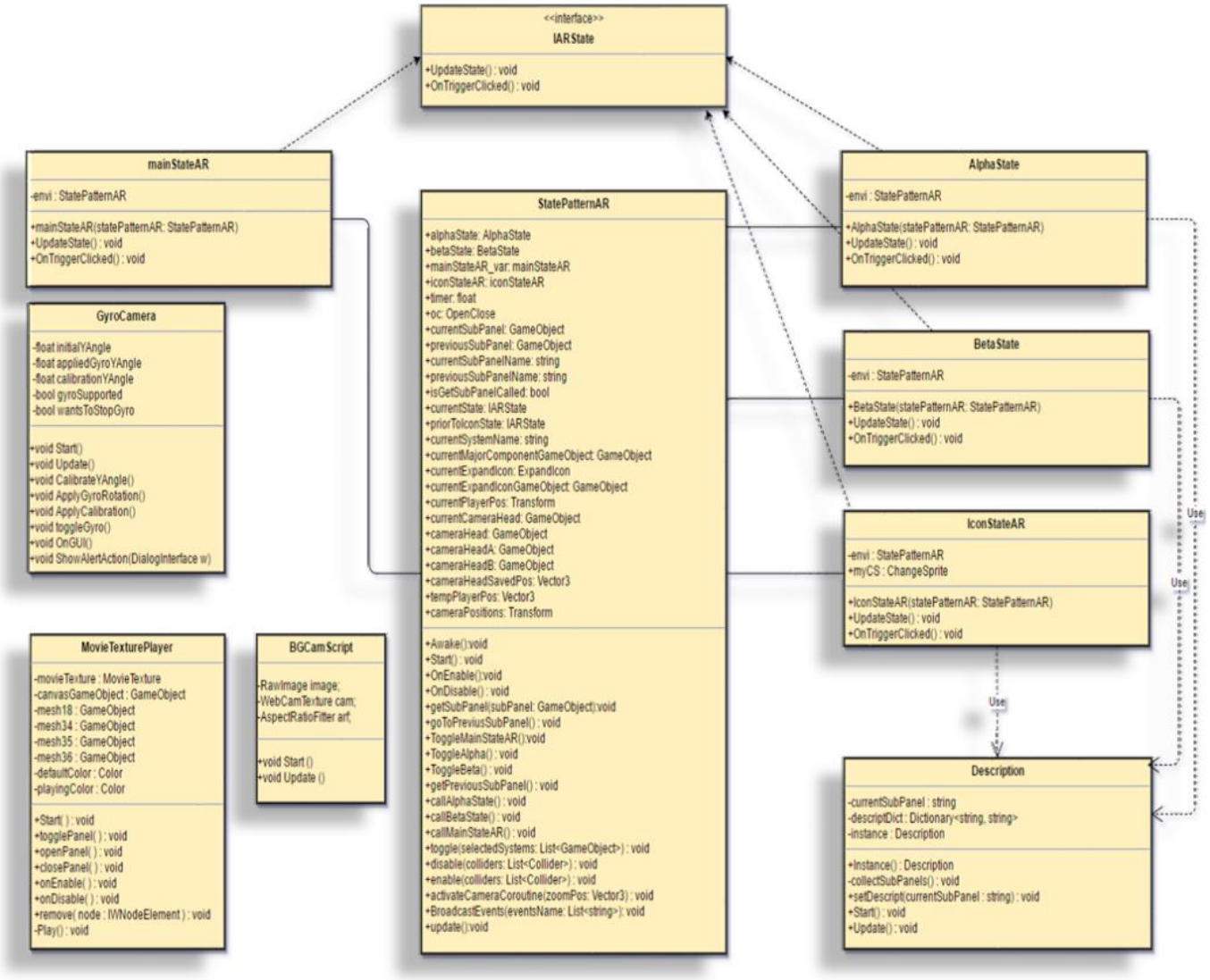

Figure 3. Minimal Class Diagram

Appendix B - User Interface Design

Figure 4. Entry to SKOPE-AR

Figure 5. Instructions how to align camera

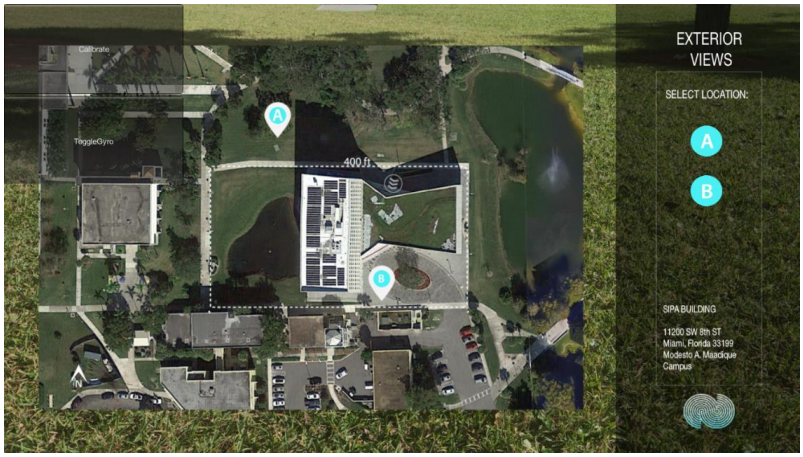

Figure 6. Select your location A or B

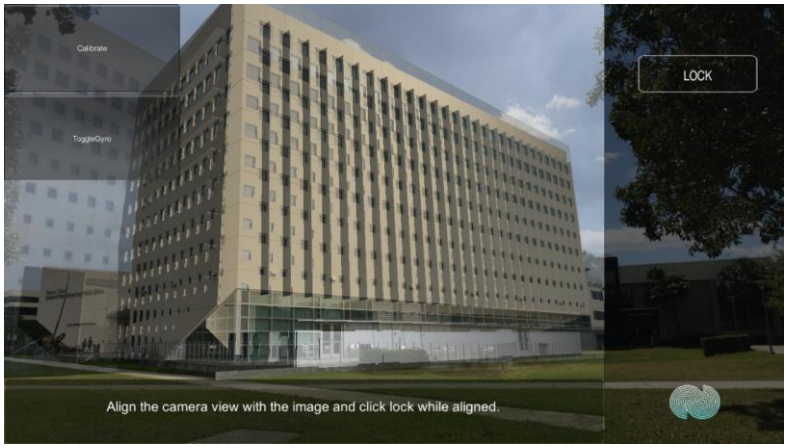

Figure 7. Align and lock camera

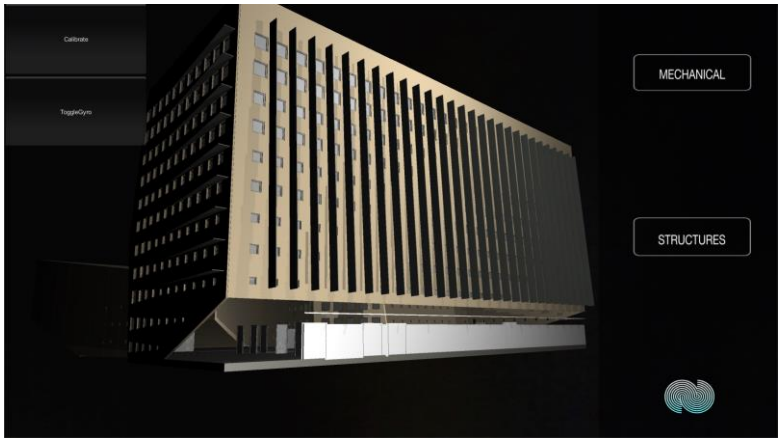

Figure 8. Select Mechanical or Structure

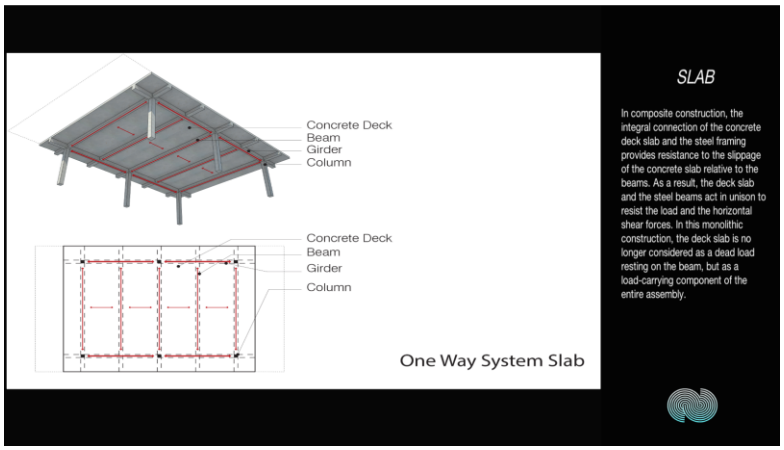

Figure 9. Sub-Panel of the structure, SLAB

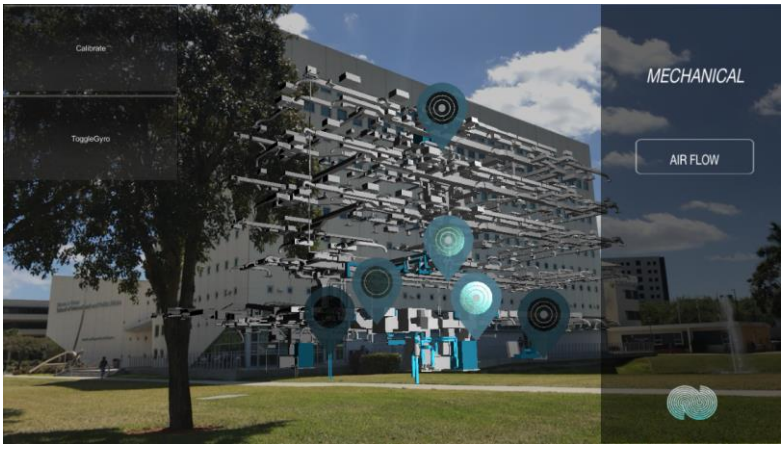

Figure 10. After user touches mechanical

Figure 11. After user touches the icon

Appendix C - Sprint Review Reports

Sprint 1, 2

02/10/2017

Attendees: Jeffrey Perez, Luis Perera
Start time: 5:00 PM
End time: 5:30 PM

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners: All.

- User Story 184
- User Story 185
- User Story 186

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.
No user story was rejected.

Sprint 3

2/24/2017

Attendees: Jeffrey Perez, Luis Perera
Start time: 5:00 PM
End time: 5:30 PM

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners: All.

- User Story 187
- User Story 188
- User Story 189

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

- No user story was rejected.

Sprint 4

3/10/2017

Attendees: Jeffrey Perez, Luis Perera
Start time: 5:00 PM
End time: 5:30 PM

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners: All.

- User Story 190
- User Story 191

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

- No user story was rejected.

Sprint 5

03/24/17

Attendees: Jeffrey Perez, Luis Perera
Start time: 5:00 PM
End time: 5:30 PM

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners:

- User Story 192
- User Story 193

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

- No user story was rejected.

Sprint 6

04/07/17

Attendees: Jeffrey Perez, Luis Perera
Start time: 5:00 PM
End time: 5:30 PM

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners:

- User Story 194
- User Story 195
- User Story 196
- User Story 199

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

- No user story was rejected.

Appendix D - User Manuals, Installation/Maintenance Document, Shortcomings/Wish list Document and other documents

Requirements:

This application is meant to be used on smartphones. Although it works on tablets, the behavior of some features is not the expected, resulting in a poor user experience. The required operating system for the app to work properly on iOS is 6.0 or later, and on Android is 4.4 or later. The application requires gyroscope on the smartphone, as well as 1GB of storage and RAM.

Installation and Usage:

To start using SKOPE-AR, it first needs to be installed on the desired device. These can be accomplished by going to the App Store (iOS) or Play Store (Android), searching for "SKOPE-AR" and downloading it. The application can be used wherever the user wants, but for a better experience, it is recommended to start it next to the SIPA building. The points are shown in a map provided by the application and can be selected when the app starts. Once the point is selected, the user can lock the model to overlay the real building captured by the device camera. From this point on, the user will be able to select some parts of the building and see the description of them as well as an explanatory video.

Wish list

The following features would be added in a feature release:

- Add more point views to the application.

REFERENCES

http://skope.fiu.edu/
https://unity3d.com/
https://developer.apple.com/xcode
http://nsubstitute.github.io/help/getting-started/