360 Webcams for Zoos and Aquariums

Design Document

Revision: 1.0.0 – 2018.03.05

General Information

Team Number: sddec18-12

Client: *True 360* (Christopher James → chris906@iastate.edu)

Faculty Advisor: Dr. Henry Duwe (duwe@iastate.edu)

Team Members and Roles:

Name	Primary Role	Secondary Role(s)
Nathan Cool	Front-End Engineer	Project Manager, Webmaster
Zach Newton	Front-End Engineer	Scrum Master, QA
Ian Jamieson	Back-End Engineer	Graphics Lead
Alan Negrete	Back-End/Database Engineer	Scribe, QA
Tarek (TJ) Yacoub	Embedded Engineer	Communication Lead, QA
Hosam (Sam) Abdeltawab	Embedded Engineer	Software Architect

Team Email: sddec18-12@iastate.edu

Team Website: sddec18-12.sd.ece.iastate.edu

Table of Contents

List of Figures 5

List of Tables 5

List of Symbols 5

List of Definitions 5

1.01 - Acknowledgement 7

1.02 - Problem Statement 7

1.03 - Operating Environment 7

1.04 - Intended Users and Uses 8
- Zoo Marketing Directors 8
- Zoo Curators 8
- Zoo IT Departments 9
- General Public 9

1.05 - Assumptions and Limitations 9
- Assumptions 9
- Limitations 9

1.06 - Expected End Product and Other Deliverables 9
- 360° Live Stream Webcam System 9
- Administrative Portal Web Application 9
- Animal Health Monitoring Web Application 10
- Marketing Web Application 10

2.01 - Proposed Design 12
- How It Works 12
- 360° Webcams 12
- Storage Solution 12
- Admin Web Application 13
- Other 13

2.02 - Design Analysis 13
- Front End 13
- Back End 13
- Embedded 15

3.01 - Interface Specifications 17
- Firebase Authentication 17
- Socket 17

MTP Interface 17
REST API 17

3.02 - Hardware and Software 18
Hardware 18
Software 18

3.03 - Process 18
General 18
Front-End 18
Back End 18
Embedded 18
Flow Diagram 19

3.04 - Results 19

4.01 - Conclusion 21

4.02 - References 21

4.03 - Appendices 21

4.04 - Changelog 21

List of Figures

Figure 1: Proposed Design 12

Figure 2: Database Design 15

Figure 3: Interface Design 17

Figure 4: Process Flow Design 19

List of Tables

N/A – Stay tuned.

List of Symbols

N/A – Stay tuned.

List of Definitions

N/A – Stay tuned.

1 - Introduction

1.01 - Acknowledgement

We would like to acknowledge our client, Chris James, and our faculty advisor, Dr. Henry Duwe, for their contributions to our project. Chris has allocated approximately $20,000 and contributed multiple 360° webcams and other hardware components for our team to use during development. He has also dedicated significant time to meet with us every week as we continue to work out the details of the project. Dr. Duwe has contributed his time and technical advice to our team, as well as assisted in steering us in the right direction on multiple occasions.

1.02 - Problem Statement

360 Webcams for Zoos and Aquariums is the brainchild project of Christopher James, founder of the Ames, Iowa-based startup company, True 360. The purpose of the project is to provide zoos and aquariums with an easy-to-use 360° webcam system for educational, animal health, and business/marketing purposes.

As of now, there are three problems which True 360 hopes to solve with its 360° webcam project. The first problem is that zoos and aquariums do not have access to an easy-to-use 360° webcam solution for their exhibits. While there *are* zoos and aquariums that currently use webcams (even 360° webcams) for various purposes, no solution currently exists which would allow them to connect and control all of their webcams from a central (remote) location. The second problem is that zoos and aquariums are always looking for ways to boost their web and social media marketing techniques to both attract new visitors and improve public awareness of issues such as animal health and wildlife conservation. The third and final problem is that zookeepers, animal health professionals, and other staff members are currently required to visit each exhibit on a regular basis in order to monitor the animals. This process is time consuming, which means staff members have less time to focus on other responsibilities.

With the 360 Webcams for Zoos and Aquariums project, our team—in collaboration with True 360—aims to provide solutions to these problems. By developing a system which will allow zoo and aquarium staff members to connect and remotely control multiple 360° webcams, the necessity of physically interacting with each webcam will be removed. The system will allow for webcams to be installed in both above-ground (indoor and outdoor) and underwater exhibits and controlled via a central web application. Conveniently, this application will help us solve the aforementioned marketing/awareness and animal health problems. Our system will enable zoos and aquariums to constantly capture 360° footage of animals in their exhibits, which can then be used for educational live streams, creating promotional content for their websites and/or social media accounts, monitoring animal health and behavior, and archival purposes.

1.03 - Operating Environment

The major physical components of the system include one or more 360° webcams and any central computing hardware necessary for connecting/controlling the webcams and managing the captured footage. In order for the system to maintain a normal operational state, there are many environmental factors which must be taken into account.

Zoos and aquariums will place webcams in a variety of locations, including both above-ground (indoor *and* outdoor) and underwater exhibits. Webcams placed above-ground and outdoors must be able to operate while exposed to the local weather. Webcams placed above-ground and indoors must be able to operate while in glass-covered exhibits or in areas where visitors' devices may tax the network. Exposing electronics to underwater conditions inevitably poses many potential problems which will have to be addressed (e.g., cable exposure, underwater webcam temperature regulation).

Regardless of where webcams are placed, there are many universal environmental factors which must be considered in order to maintain a normal operational state. The webcams must have constant access to ample power, a stable and strong (wired or wireless) Internet connection, and a case to physically protect the webcam unit. Fortunately, our client is in the process of fabricating a case to fit the webcam model used in the system.

One of the most important environmental factors which must be considered is the presence of animals. The webcam unit must be durable enough to withstand the impact and pressure of strikes, bumps, bites, and other animal interactions. Any cables (e.g., power, Ethernet, USB) connected to the webcams must be installed in such a way as to remain out of the sight/reach of the animals.

In addition to the webcams, we must take into consideration the computing/storage hardware and the software (web application) components and their respective environments. Any computers (desktops, servers) used must be stored in a secure indoor and temperature-regulated environment in order to avoid issues such as overheating and unauthorized access. The web application must be developed with digital security in mind.

Overall, these environmental factors represent our basic understanding of and assumptions about how zoos and aquariums will use the system. They do not encapsulate every possible factor, as there will inevitably be some variance between facilities/users.

1.04 - Intended Users and Uses

There are three end users we need to account for when developing the system: zoo marketing directors, zoo curators, and zoo IT departments.

Zoo Marketing Directors

Zoo marketing directors will be able to embed or publish (to YouTube) 360° live streams of animal exhibits. Marketing teams will receive a weekly portfolio of pictures and timelapses of when the animals were most active.

Zoo Curators

Zoo curators will be able to monitor animals without the need to go to the desired exhibit. Moreover, curators will be able to go back in time—since streams will be digitally archived—to learn more about animals' everyday patterns.

Zoo IT Departments

Zoo IT departments will be able to remotely control the webcams, as well as manage live streams.

General Public

The 360° live streams of the zoo enclosures will be able to reach people in the general public that would otherwise not be able to experience the zoo. This will increase animal education and conservation awareness to a group of people who were excluded from this level of information previously.

1.05 - Assumptions and Limitations

Assumptions

- Our product will not be responsible for managing payments or transactions.
- Our product will be responsible for providing:
    - Webcam management
    - User management (multiple user types)

Limitations

- Using manufacturer-provided SDKs/APIs for managing the webcam, setting up live streams, etc.
- Webcams may be limited to connection via WiFi or USB.
- Having broadcasting software (OBS) as a middleman limits our ability to control webcams.
- Server storage cost/space and Internet bandwidth at zoos and aquariums may be bottlenecks for collecting high-resolution footage (1080p, 4K, etc.).

1.06 - Expected End Product and Other Deliverables

360° Live Stream Webcam System

- Estimated Delivery Date: May 2018 (Version 1), December 2018 (Version 2)
- Product Description (OLD): Our team will develop software for a webcam which can stream 360° video. This webcam will be the first of two primary products that we will deliver. The webcam will use an existing 360° webcam in tandem with the software we develop. The webcam will be used to stream live activities occurring in the exhibits.
- Product Description: Our team will select hardware/software components appropriate for developing 360° webcam software. These components will be selected from amongst existing industry products (webcams, servers, protocols, database frameworks, front-end frameworks, etc.).

Administrative Portal Web Application

- Estimated Delivery Date: May 2018 (Version 1), December 2018 (Version 2)
- Product Description: Our team will develop a web application which will allow zoo and aquarium staff members to connect and remotely control the various webcams installed throughout the animal exhibits. As a web application, staff members will be able to access it from a variety of desktop, tablet, and mobile platforms.

Animal Health Monitoring Web Application

- Estimated Delivery Date: December 2018 (Version 1)
- Product Description: Our team will develop a web application which will allow zoo and aquarium staff members to monitor the health and well-being of animals. It will utilize machine learning to observe normal animal behavior in order to alert staff members of abnormalities.

Marketing Web Application

- Estimated Delivery Date: December 2018 (Version 1)
- Product Description: Our team will develop a web application which will allow sponsors to integrate company logos and product advertisements into live streams. The web application will also allow zoo and aquarium marketing staff to extract various pictures and timelapses for marketing purposes.

2 - Specifications and Analysis

2.01 - Proposed Design

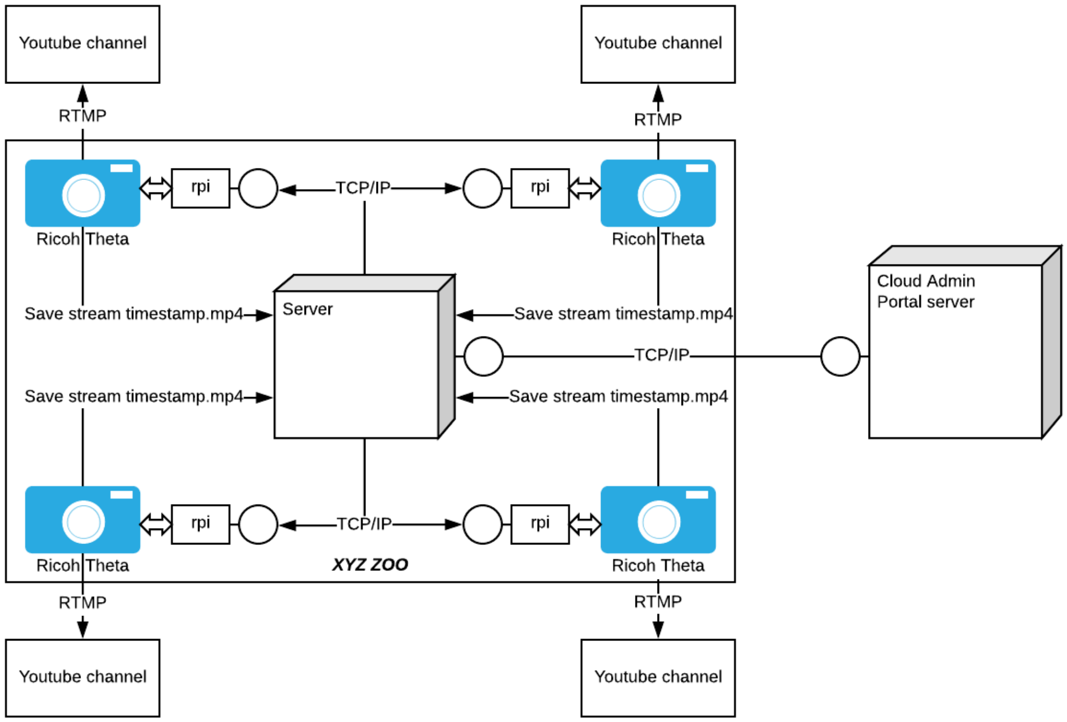

Figure 1: Proposed Design

How It Works

Each Ricoh Theta is behind an interface provided by the Raspberry Pi. The Raspberry Pi is able to receive commands from the local server and in turn sends commands to the Ricoh Theta. The Raspberry Pi will run the Ffmpeg library (transcoding software) which streams videos to youtube as well as archive to a given path. The server in the zoo will handle communication coming from the web application for managing webcams or accessing archives.

360° Webcams

The Ricoh Theta is compacted live streaming webcam that provides good accessibility for developer to develop software around it. The webcam is provides two was to communicate with it: MTP protocol through a usb cable, REST api protocol through WIFI. The Ricoh Theta is able to stream and take still pictures in different resolutions that the developer can change through the provided commands.

Storage Solution

Zoo will have to host a server that acts as a public facing machine that the web application (Admin Portal) will communicate with. The second important purpose of this server is to archive 72 hours of previous live stream from each webcam. Consequently, zoo staff will be able to go back at most 72 hours to address any issues related to animal exhibits.

Admin Web Application

Since different zoos will be using the same admin portal, the web application will be hosted on the cloud. Users will be associated with a zoo, therefore, when a user logs in the Admin Portal they will be able to manage (depending on the user permission) the system related to the zoo he/she works in. Finally, users will be able to request previous live stream if needed.

Other

The Ricoh Theta is able to stream in 4K resolution. Additionally, the Raspberry Pi handles synchronous streaming and archiving by using the FFmpeg library. The hardware is not publicly accessible. The server in the zoo is the only public facing machine; therefore, the hardware in the zoo is secure. The web application will use Firebase Authentication to prevent people from messing around with zoo's stream.

2.02 - Design Analysis

Front End

- Details: Our system's front-end component will consist of a web application. This web app will allow zoo staff members to interact with their cameras, stream archives, etc. Ideally, the web app will be easily accessible via any Internet-connected device, whether it be a computer, a tablet, or a phone.

- Status: The front end is still being investigated, but it appears that there are existing frameworks which are compatible with our backend and embedded solutions.

- Going Forward: We are continuing to build the framework for the True 360 website and identify the minimal viable product.

- Dependencies: We are currently considering using React and Redux.

Back End

Node JS Server

- Details: We have developed our first prototype on the back end using Node JS. We have competed setting up the project, and the API is fully functional. The project has been broken down into a MVC structure. We have also implemented an ORM design to ensure modularity of the project going forward and to allow us to expand on our API functionality quickly and efficiently. Currently we have set up the migrations, models, seeders and controllers based on our Database Diagram below.

- Status: So far, it appears that this approach will work well. We will continue expanding on this prototype and analyze it to make sure we start doing a solid best practices approach design, instead of focusing on just making it work.

- Going Forward: We will begin expanding on the API endpoints, allowing our React front end to easily modify & request data for Zoos, Users, Exhibits and Cameras. Another next piece we want to focus on is the authentication aspect, we will use Firebase Authentication on the front end and add a middleware on the back end using the Firebase Admin package to ensure our API is secure from unauthenticated users. Lastly, we will start generating documentation for our API. This will include details for each controller, such as the routes, request types, request parameters and some expected response examples.

- Dependencies: Our primary depency is Sequelize, which is what we use for our ORM design. We'll have to dig deeper and test this library, to ensure it will fulfill our future requirements as well. If it can't, we would then be able to quickly look for an alternate solution, while it's still early in the project. We are also using Faker for seeding our database during development. Along with the other popular libraries that work with Node, such as express.

Database

- Details: Our initial prototype is using MySQL for the database. Currently, our design only includes the core aspects of the application. We begin expanding as the project progresses. This is something that we plan to do going forward. For now, we want to get the core functionality of the application and ensure there is a good communication between our software and hardware components. From there, we will expand on the React side of things, along with the necessary additions to the back end and database.

- Diagram:

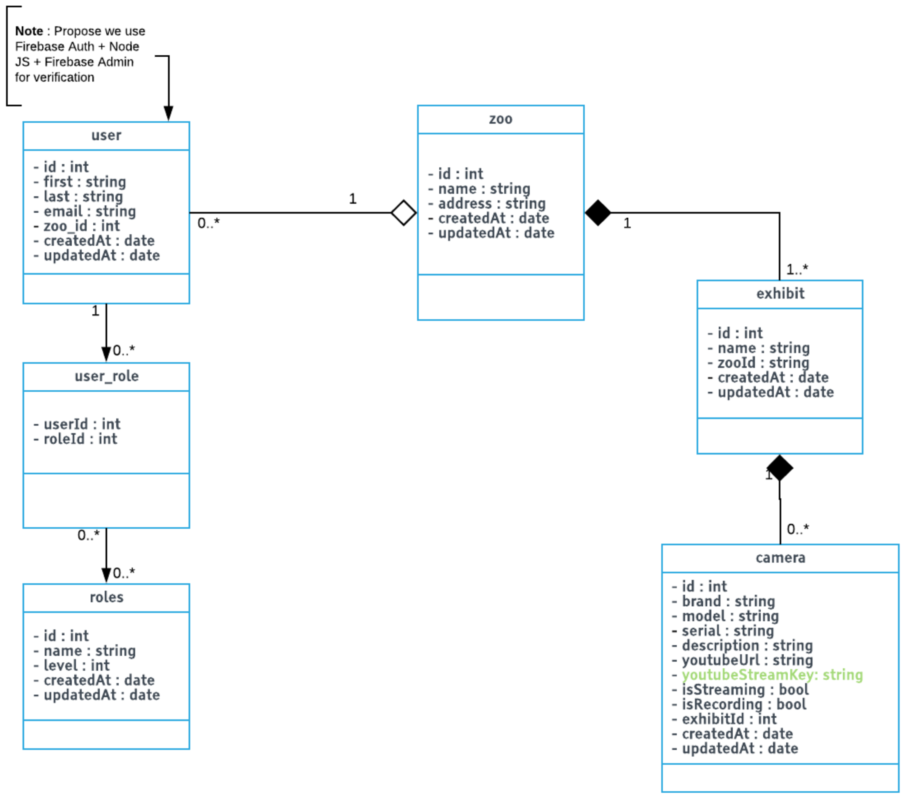

Figure 2: Database Design

Embedded

- Details: As of the time of writing this document, the embedded portion of this project is not compartmentalized. The Java application that is being developed currently is implemented as one component that does the job of both the Raspberry Pi and the local server.

- Status: The Java application currently handles YouTube authentication; creates a stream configuration if one has not been created yet; get the RTMP endpoint from YouTube; publish the stream to the retrieved RTMP, as well as archive to a specified storage path.

- Going Forward: There are still some more steps that the application needs to implement to make the streaming workflow fully automatic. Moreover, the application should be split to server code and client code where client is the Raspberry Pi's in the zoo.

- Dependencies: The application depends on the YouTube SDK, and the FFmpeg Java library developed by Andrew Brampton.

3 - Testing and Implementation

3.01 - Interface Specifications

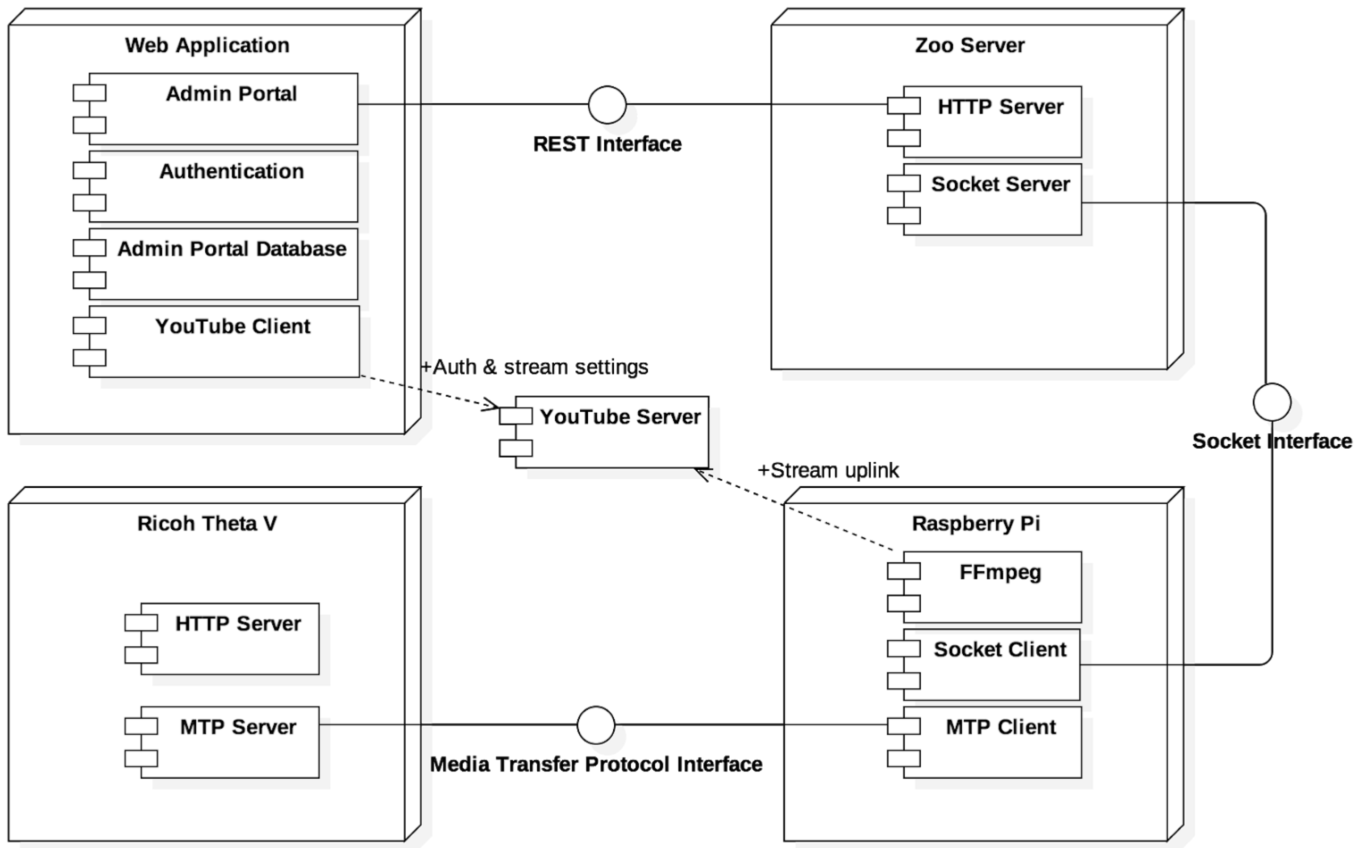

Figure 3: Interface Design

Firebase Authentication

Our web application will need to communicate with an authentication service in order to provide security for user login purposes. We are using the Firebase Authentication service, which will provide our web application with a straightforward JavaScript API interface through which to store credentials.

Socket

We are using Socket communications to interface between the zoo server and the Raspberry Pi.

MTP Interface

The MTP (Media Transfer Protocol) interface is the protocol used by the Ricoh Theta V to interface through a USB cable.

REST API

The REST API will be used to interface between the Admin Portal and the zoo server.

3.02 - Hardware and Software

Hardware

- Ricoh Theta V Camera: The Ricoh Theta V is being used for minimal viable product and test any new ideas we think of along the way.
- Raspberry Pi 3: The Raspberry Pi is being used to store and stream the video from the camera.
- Zoo server: The zoo server will be used as the an archive, and a public facing API to control all the components in the zoo (rpi's and Ricoh Theta V's).

Software

- OBS Studio: OBS Studio is considered to be the "gold standard" for streaming video to Youtube or Twitch. We used OBS Studio to test the cameras and their abilities.
- NOOBS: We used NOOBS for the Raspberry Pi operating system.
- NodeJS: We used, and are still using, NodeJS for the server side technology.
- Cameras' SDK & APIs: Many of the cameras that we've worked with have SDKs or APIs available for development purposes. We've been using the ones available to us for the testing phase of the development process.
- MySQL Database: We are using a provided Iowa State MySQL Database to help test some of our data management technology.

3.03 - Process

General

We plan on using the GitLab continuous integration and continuous development tools for testing builds of the entire project. Any time changes are committed to a branch, our testing server or virtual machine will start a build. Any pull requests will also be required to have a successful build before being merged into the master branch.

Front-End

The front-end testing process will involve the use of a build server. Whenever code changes are pushed, the server will allow us to build the project and run various types of tests to validate the new features. We will focus on both code-based and interaction-based tests.

Back End

The testing we do for the back end of our system will be fairly simple. Our focus at the moment is to ensure that we have a solid foundation set up so that the other teams are able to continue development.

Embedded

Our hardware programming will be tested in a few different ways, but at the moment, we are just doing verification and validation testing. This has allowed us to properly ensure that the product that we are building is meeting expectations for the project.

Flow Diagram

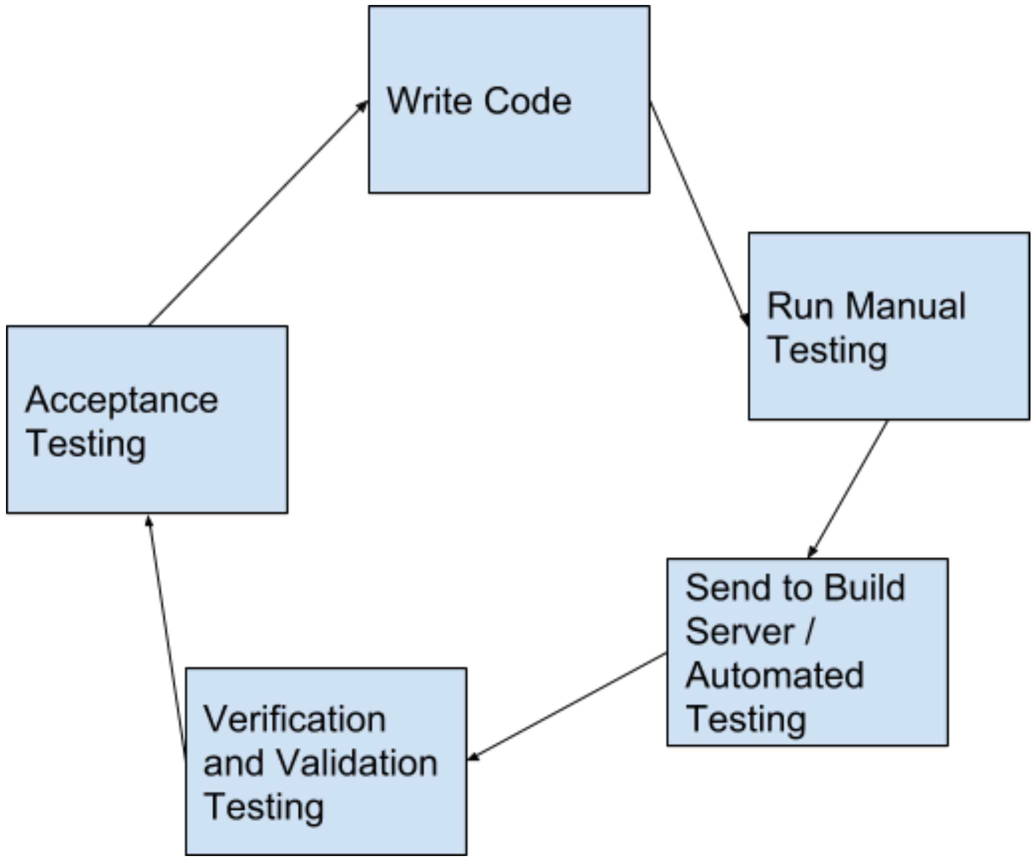

Figure 4: Process Flow Diagram

3.04 - Results

Throughout the duration of the semester so far, we've had a few key opportunities to test what we've been working on and the ideas we've come up with. As discussed above, we tried quite a few different types of cameras and technologies. This has allowed us to get a better understanding of the market, what's available, and what we can develop to fill the need.

In testing our camera options, we've come to the conclusion that the Ricoh Theta is the best camera for the time being. This has allowed us to begin development with this camera and make progress towards a minimal viable product. That being said, we're still working towards better testing methods which will allow us verify the quality of the product we're producing. The next few months will be focused on building the product out and developing testing environments for the technology.

4 - Closing Materials

4.01 - Conclusion

In collaboration with True 360, our team aims to provide zoos and aquariums with an easy-to-use 360° webcam system for educational, animal health, and business/marketing purposes. Webcam setups which are currently used by zoos and aquariums provide limited functionality respect to interconnectivity, footage quality, and software features. Our solution will remove the need for physical interaction with each webcam and provide a wide array of software features including centralized webcam control, 360° footage capture/archiving and live streaming, animal health monitoring, and curation of marketing content.

4.02 - References

Note: *When academic references are used to assist us in designing and developing our project, we will transition to an official format (MLA or APA) in order to properly cite said references. As of now, this subsection contains bookmarks for the sites we have referred to.*

Description	Link
True 360 (client website)	https://true360.weebly.com/
Ricoh Theta V product site	https://theta360.com/en/about/theta/v.html
Ricoh Theta V developer site	https://developers.theta360.com/en/
Open Broadcast Software (OBS)	https://obsproject.com/
FFmpeg	https://www.ffmpeg.org/
FFmpeg Java	https://github.com/bramp/ffmpeg-cli-wrapper
YouTube SDK	https://developers.google.com/youtube/
San Diego Zoo	http://sdzsafaripark.org

4.03 - Appendices

N/A – Stay tuned.

4.04 - Changelog

Note: *Changes are listed in descending order (newest on top).*

- **2018.03.05 (Version 1.0.0)** – First draft submitted