

# Immersive Experiences for Zoos and Aquariums

Design Document

sdmay21-02

## **Client**

True360

## **Advisor**

Dr. Alexander Stoytchev

## **Team Members**

Daniel Barnes - Meeting Scribe

Aryan Joshi - Report Manager

Austin Nebel - Chief Engineer

Samuel Shifflett - Meeting Facilitator

Noah Syens - Test Engineer

Thomas Powell - Chief Engineer (UI)

Sebastian Vang - Product Owner

[sdmay21-02@iastate.edu](mailto:sdmay21-02@iastate.edu)

<http://sdmay21-02.sd.ece.iastate.edu>

Revised: October 4th, 2020

# Executive Summary

This document provides the overall design and plan that the group *sdmay21-02* will be taking in order to complete the project for our client, *True360*. The document is split up into six different sections, and each section describes a specific part of the project. The sections are as follows:

1. Introduction - This section introduces the project, explaining what it is, why it is needed, and what is required of our group
2. Project Plan - This section describes how we will go about completing the project during our second semester of Senior Design
3. Design - This section illustrates how our project will be created, both on the software side, and the visual side
4. Testing - This section describes how our project will be tested to ensure correct functionality, both with automated and manual testing
5. Implementation - This section describes a preliminary implementation of what we will be creating during the second semester
6. Closing Material - This section ties all the previous sections together, and includes any miscellaneous information that did not fit in any other sections.

## Development Standards & Practices Used

Technology we will use in this project:

- Node.js
- Electron
- Express
- Mocha/Sinon/Chai
- SQL Database

Engineering Standards:

- Test Driven Development will be conducted
- Follow all best practices for security, ensuring that user data is secure and encrypted
- Code should be written in a way that it is easy to maintain and understand
- All code must be reviewed by at least 2 other developers

## Summary of Requirements

- Create a desktop application that allows users to upload WOW! Moments to various platforms by right clicking the videos
- Create a centralized web server that allows for communication between True360 and various zoos and aquariums.

## Applicable Courses from Iowa State University Curriculum

List all Iowa State University courses whose contents were applicable to your project.

- COM S 228

- COM S 252
- COM S 309
- COM S 319
- COM S 327
- COM S 329
- COM S 339
- COM S 363
- COM S 409
- CprE 230
- CprE 231

### New Skills/Knowledge acquired that was not taught in courses

List all new skills/knowledge that your team acquired which was not part of your Iowa State curriculum in order to complete this project (still in progress).

- Electron
- Registry Editing
- API Development
- Functional Testing
- Code Reviews
- Product Deployments
- Cybersecurity Practices

# Table of Contents

<b>1 Introduction</b>	<b>5</b>
1.1 Acknowledgement	5
1.2 Problem and Project Statement	5
1.3 Operational Environment	5
1.4 Requirements	6
1.5 Intended Users and Uses	7
1.6 Assumptions and Limitations	7
1.7 Expected End Product and Deliverables	7
<b>2 Project Plan</b>	<b>8</b>
2.1 Task Decomposition	8
2.2 Risks And Risk Management/Mitigation	10
2.3 Project Proposed Milestones, Metrics, and Evaluation Criteria	11
2.4 Project Timeline/Schedule	12
2.5 Project Tracking Procedures	12
2.6 Personnel Effort Requirements	13
2.7 Other Resource Requirements	16
2.8 Financial Requirements	16
<b>3 Design</b>	<b>17</b>
3.1 Previous Work And Literature	17
3.2 Design Thinking	17
3.3 Proposed Design	17
3.4 Technology Considerations	17
3.5 Design Analysis	17
3.6 Development Process	18
3.7 Design Plan	18
<b>4 Testing</b>	<b>18</b>
4.1 Unit Testing	18
4.2 Interface Testing	18
4.3 Acceptance Testing	19
4.4 Results	19
<b>5 Implementation</b>	<b>19</b>
<b>6 Closing Material</b>	<b>19</b>
6.1 Conclusion	19
6.2 References	19
6.3 Appendices	19

List of figures/tables/symbols/definitions (This should be the similar to the project plan)

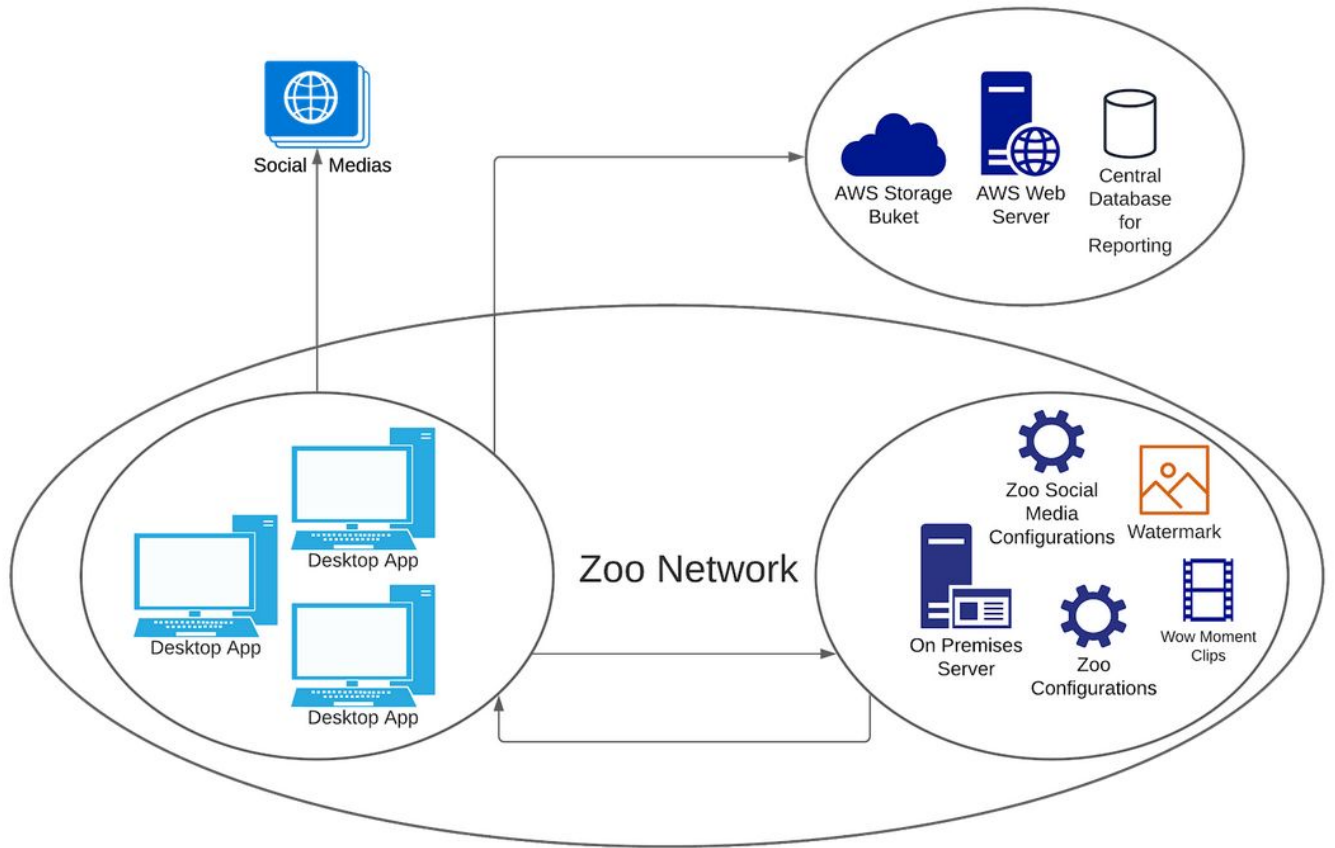


Figure 1. Overview of the product's architecture design plans.

# 1 Introduction

## 1.1 ACKNOWLEDGEMENT

We would like to thank the following individuals for assisting us in the completion of this document. Without their contributions, we would not have been able to complete it.

### True360:

- Chris James
- David Body
- Kuan-Chuen Wu
- Ryan Arndorfer
- Sari Lewinsky
- Wyman Martinek

### Iowa State University

- Akilesh Tyagi
- Alexander Stoytchev

## 1.2 PROBLEM AND PROJECT STATEMENT

### 1.2.1 Problem:

True360 aims to provide an immersive VR experience for customers going to the zoos and aquariums with published 360 video. These special moments are called WOW! Moments. The general problem here is that there is no way to publish these videos easily across multiple platforms on a user friendly desktop application. Additionally, there is no easy way to display the True360 license through a watermark on the WOW! moments without having to utilize editing softwares that the staff may not know how to use. Finally, there is also no ability to keep a record of the uploads. The watermark needs to be added as to preserve the rights to the footage. For example, uploading to Youtube does not add a watermark to their videos and True360 requires all of their WOW! moments to have a watermark placed on the videos. There also needs to be a method of keeping track of the media being published to social media webpages. Otherwise True360 may lose traces of their own content.

### 1.2.2 Project Statement:

To solve this problem, we will be creating a cross-platform desktop application that can be opened by right clicking a video, which will allow the user to publish their video through the Desktop Application. This improves the quality of usage for the user due to the ease of video publishing and ensures the best possible quality of the video being uploaded to their respective social media platform. It also ensures that the videos follow the True360 video licenses by adding the watermark to the video. We will have direct control over the publishing process so we can keep records of the videos uploaded through an API we will be developing.

## 1.3 OPERATIONAL ENVIRONMENT

We will develop a desktop application running on either Windows 10 or MacOS Catalina in a business environment. The on premises server will be handled by the zoo in a controlled environment with heat dissipation and dust filtering.

## 1.4 REQUIREMENTS

### General:

- Automated tests will be run on the systems before it is deployed to production.
- All features and code within the systems will be documented with comments within the code.
- All code must be approved by at least 2 other developers before being merged into a codebase.
- All UI designs will be approved by True360 before they are submitted to production.

### Cloud System:

- Create a cloud based Express web server that is able to receive HTTP POST requests from the desktop app that contain video publication data.
- Create a cloud based SQL database that the cloud based Express web server can store information from the HTTP Post requests messages.
- Contains a */publish* POST endpoint that adds a published video to the SQL database
- Create a cloud based process that runs daily to retrieve views, likes, and shares from all the videos that were published through the True360 Desktop Application across multiple social media platforms.
- Contains a */support* POST endpoint that receives a name, description, and application version number, and then formats and sends the information in an email to [support@true360.com](mailto:support@true360.com)
- Contains a */requestEdit* POST endpoint that receives a video ID/Title, and associated zoo, and formats and sends the information in an email to [editing@true360.com](mailto:editing@true360.com)

### Desktop App:

- Allows for a right click “Open with True360” to open the application
- Desktop Application can run on Windows 10 and MacOS Catalina
- Allows users to publish a selected video to YouTube, Facebook, Vimeo, and Hootsuite in the optimal video format for each platform.
- Allows users to configure login credentials to multiple accounts per platform, and then select a specific account when publishing
- The Desktop Application will be built in Electron to offer cross compatibility on MacOS and Windows
- API keys and passwords for connected publishing platforms will be encrypted and hashed using the NIST standard and stored on the Network Share Server
- Desktop app will automatically update when the user launches it if update is available
- Desktop app will retrieve the watermark image from Network Share Server
- Desktop app will hit the */publish* endpoint on the cloud web server on successful video publish. It will supply the video URL
- Uninstallation of the Desktop Application should remove all the files associated with the Desktop Application on the user’s computer as well as the right click option from the popup menu on the OS but keep the corresponding information within the cloud database.
- Users that have the Desktop Application will have an option in the Application to send a support request to the cloud based web server that will notify the necessary individuals of the support request by sending a */support* POST request to the cloud web server.
- The Desktop Application will include a “Help” section to aid users in understanding how to properly use it.
- Adds Metadata for copyright, publication info, and tags to videos before publishing
- Users should be able to submit “project” requests to True360 through the cloud based web server at */requestEdit* that notifies the editing team certain videos they would like True360 edit.

## 1.5 Intended Users and Uses

To attain an end product that performs well and provides the best user experience, we must make sure that we fulfill the uses and needs of the users specified below.

### Intended Users:

- Zoo/Aquarium Management
- Zoo/Aquarium Staff
- True360 Administration

### Intended Uses:

- Management of camera footage by the zoos/aquarium staff.
- Uploads 360 captured camera footage of animals (WOW! moments) to specified media platforms.
- Extendability to add additional social media platforms.
- Documentation of the video publication data.
- Automatically watermark uploaded footage.
- Designate videos for True360 to directly edit.

## 1.6 ASSUMPTIONS AND LIMITATIONS

### Assumptions:

- True360 app will be cross platform.
- Users will have either the latest versions of Windows 10 or MacOS Catalina installed.
- True360 app will be extensible to support other social media and VR kiosks.
- True360 app will be easy and seamless for the user experience.
- Detected WOW Moments! will already be stored on the zoo's local server.
- End users will know how to navigate to their directory to find the videos.
- End users have existing social media accounts to publish videos to.
- Zoos will have enough internet bandwidth to successfully upload 8k resolution videos.
- Watermarks will be added to each video that is published with the True360 app.
- The end product will eventually be deployed across multiple zoos in the US.

### Limitations:

- There is no budget for developing the True360 app.
- The end product will not support mobile OS.
- Most of the development team has little to no experience developing Electron apps.
- True360 app is limited to the Electron framework.
- True360 app is not guaranteed to work with older builds of supported OS's.
- The end product will only be used by zoos in the US and that are contracted with True360.

## 1.7 EXPECTED END PRODUCT AND DELIVERABLES

A desktop-based commercial application that will streamline the zoo employee experience for publishing WOW! Moments to various social media platforms. The application will be installed via an ease-of-use downloadable installer which will deploy the software onto the employee's machines, and it will be designed to work on both Windows 10 and Mac OS Catalina. Once installed, the user will be able to access



WOW! Moments in a shared drive on their local network which is connected to the True360 server located at the zoo.

Once in the server's WOW! Moment directory, the user will be able to right-click on any mp4 files generated by the True360 system and have the option to open the True360 Desktop Application to publish that video. If the user decides to publish a video, a True360 watermark and several mp4 metadata fields will automatically be attached to it and the video will be published to the platform of choice. From this context menu the user will also be able to send True360 personalized support requests as well as configure their social media account configurations..

There will only be one initial configuration aspect of the application which will be to link the app with the zoo's social media accounts. This configuration will live on the True360 server at the zoo and all computers that have the True360 software installed will access this file. This will allow for the configuration process to be efficient for the end-user. The application will then receive automated updates at any given time to both improve the user experience and to provide security fixes. This software will be delivered by the end of April, 2021.

## 2 Project Plan

### 2.1 TASK DECOMPOSITION

#### Project Goal:

True360 has tasked us with creating a desktop application for zoos that are partnered with True360. The Desktop Application will allow for employees at the zoo to publish videos that are "WOW" moments to the zoo's corresponding social media platforms and also maintain a database of publishing data that lives within True360's AWS cloud account.

#### Elements of the Overall System:

##### Desktop Application:

The Desktop Application will be the tool that employees at the zoo will use to publish their videos to their corresponding social media platforms.

##### AWS Cloud Account:

The AWS Cloud Account will be the centralized area where the publishing data will be stored so that True360 can monitor usage of the Desktop Application.

##### On Premises Server:

The On Premises Server holds the video files that can be published as well as configuration files that will link their social media accounts to their Desktop Application.

#### Tasks:

##### Source Control Setup:

- Create Repositories for Desktop Application Code and Cloud Based Scripts

- Create Branching Strategy for Each Repository

#### Cloud Environment Setup:

- Setup Virtual Machine what will Run the Cloud Based Web Server
- Setup SQL Database that is able to be Linked to Cloud Based Web Server
- Create Recurring Process to Retrieve Interaction Data with Videos Published through True360 Desktop Application

#### Desktop Development and Testing Environment Setup:

- Setup Testing Environments for Mac OS Catalina and Windows 10 Operating Systems
- Build Basic Electron App that Each Team Member can Test on their Machine
- Organize Code for Electron App in Source Control Repository into Front-End and Back-End Categories

#### Operating System Registry Development and Testing Setup:

- Develop Process for Testing Right Click Feature on Mac OS Catalina and Windows 10

#### Right Click Feature Development:

- Develop Basic Right Click Functionality on MacOS Catalina and Windows 10
- Integrate Basic Electron Application to Open with Right Click Feature

#### Backend Desktop Application Development:

- Develop Backend of Desktop Application to Send HTTP Post Requests to Localhost Server
- Change Basic HTTP Post Requests to Send Requests to Cloud Based Web Server

#### Cloud Web Server Development:

- Develop Web Server to Receive HTTP Post Requests with Multiple Endpoints
- Develop Process to Ingest Basic Data Into Cloud Based SQL Database

#### Testing of Desktop Application Linked with Cloud Web Server and Database:

- Create Action Button in Desktop Application to Send Basic HTTP Post Request to Cloud Based Web Server
- Test Ingesting of Basic HTTP Post Request to Cloud Based SQL Database from Cloud Based Web Server

#### Social Media API Integration Development:

- Develop Basic Publishing of Video to Youtube, Facebook and Vimeo Using Each Platform's API
- Develop HTTP Post Requests to Send Responses of Successful/Failed Publishing to Cloud Based Web Server

#### Cloud Database Schema Design for Social Media Response Messages:

- Create Table Designs for Each Social Media's Response Message
- Create Specific Endpoints for Each Social Media Response

- Create Ingestion of Each Social Media Endpoint Into Cloud Based SQL Database

Mac OS Catalina and Windows 10 Installer Generation:

- Test Installer Generation for Mac OS Catalina and Windows 10
- Test Installation on Mac OS Catalina and Windows 10
- Test Uninstallation on Mac OS Catalina and Windows 10
- Test Desktop Functionality on Mac OS Catalina and Windows 10

Develop CI/CD Process for Updates to Desktop Application:

- Develop Process to Check Version of Desktop Application with Latest Version of Code Base in Source Control Repository
- Develop Process to Either Update Application Immediately or Delay Update

Develop Process for Building Social Media Configurations:

- Develop Process to Retrieve Social Media Configurations from Network Share Server
- Develop Process to Add Multiple Accounts for Each Social Media Platform

Develop Process for Parsing Social Media Configurations:

- Develop Process to Parse Social Media Configurations that Allows Users to Publish to Corresponding Social Media Platforms

Develop Process for Updating Social Media Configurations:

- Develop Process for User's to Update Their Social Media Configurations

Create UI That is Linked to the Backend Functionality of the Desktop Application:

- Implement Designs of UI
- Link Action Buttons to Backend Functionalities

Test Functionality of Entire Desktop Application and Cloud Based System Working Together:

- Test Functionality of Desktop Application Using UI for Each Social Media Platform
- Test Functionality of Cloud Based Web Server and Ingestion of Data Into Cloud Based SQL Server
- Test Functionality of Daily Process that Retrieves Interaction Data for Each Video Published Through True360 Desktop Application

## 2.2 RISKS AND RISK MANAGEMENT/MITIGATION

There is no budget for developing the True360 app: **Score 0.2**

The end product will not support mobile OS: **Score 0.1**

Most of the development team has little to no experience developing Electron apps: **Score 0.6**

Mitigation Plan: The Desktop Team will create a basic Electron application in the beginning to better understand how to move forward with Development during the rest of the project.

True360 app is limited to the Electron framework: **Score 0.4**

True360 app is not guaranteed to work with older builds of supported OS's: **Score 0.3**

The end product will only be used by zoos in the US and that are contracted with True360: **Score 0.2**

The Desktop Application will rely heavily on the strength of the internet connection at the zoos: **Score 0.4**

The Desktop Application will rely on the user's computer resources for video processing and social media uploads, which will require a computer with good processing speed. **Score: 0.4**

## 2.3 PROJECT PROPOSED MILESTONES, METRICS, AND EVALUATION CRITERIA

The milestones are derived from the tasks in section 2.1. Since we are building the system from the ground up, these are our milestones and evaluation criteria:

### Phase 1: Environment Setup with Managed Source Control Process

- Source Control Setup: The application will be maintained with a git repository and any merge requests will be approved.
- Cloud Environment Setup: Setting up a database that will be hosted on the cloud.
- Desktop Development and Testing Environment Setup: Setting up testing environments so that the application can be tested thoroughly across each developers system.
- Operating System Registry Development and Testing Setup: Developing a test for right clicking functions on Windows and MacOS.

### Phase 2: Basic Functionality Development

- Right Click Feature Development: End users will be able to right click the context menu and select options with our app.
- Backend Desktop Application Development: Develop backend to send HTTP requests to the local server.
- Cloud Web Server Development: Develop a cloud web server to receive HTTP post requests.
- Testing of Desktop Application Linked with Cloud Web Server and Database: Set up basic testing of sending data to the web server.

### Phase 3: Finalizing Advanced Functionality Development

- Social Media API Integration Development: Basic publishing of videos to various social media using their API.
- Cloud Database Schema Design for Social Media Response Messages: Creating specific endpoints for each social media response.
- Mac OS Catalina and Windows 10 Installer Generation: Testing installation and functionality on Windows/MacOS.
- Develop CI/CD Process for Updates to Desktop Application: Checking for application updates.
- Develop Process for Building Social Media Configurations: Developing a process to retrieve configurations and adding multiple accounts for each social media platform.
- Develop Process for Parsing Social Media Configurations: Developing a process to parse social media configuration.
- Develop Process for Updating Social Media Configurations: Developing a process to allow users to update social media configuration.

### Phase 4: Complete Functionality with UI Development and Testing

- Create UI That is Linked to the Backend Functionality of the Desktop Application: Create a UI that end users will interact with that is linked to the desktop application backend.
- Test Functionality of Entire Desktop Application and Cloud Based System Working Together: Testing functionality for the UI linked with multiple social media platforms, video data analytics retrieved and ingestion of data into the cloud.

## 2.4 PROJECT TIMELINE/SCHEDULE

The following timeline is how our team plans to complete this project. The different colors represent different phases of the project throughout its development lifecycle. Each task is assigned to either the Cloud Team or Desktop Team. Each task also shows the tentative start and end dates.

### True360 Desktop Application Development

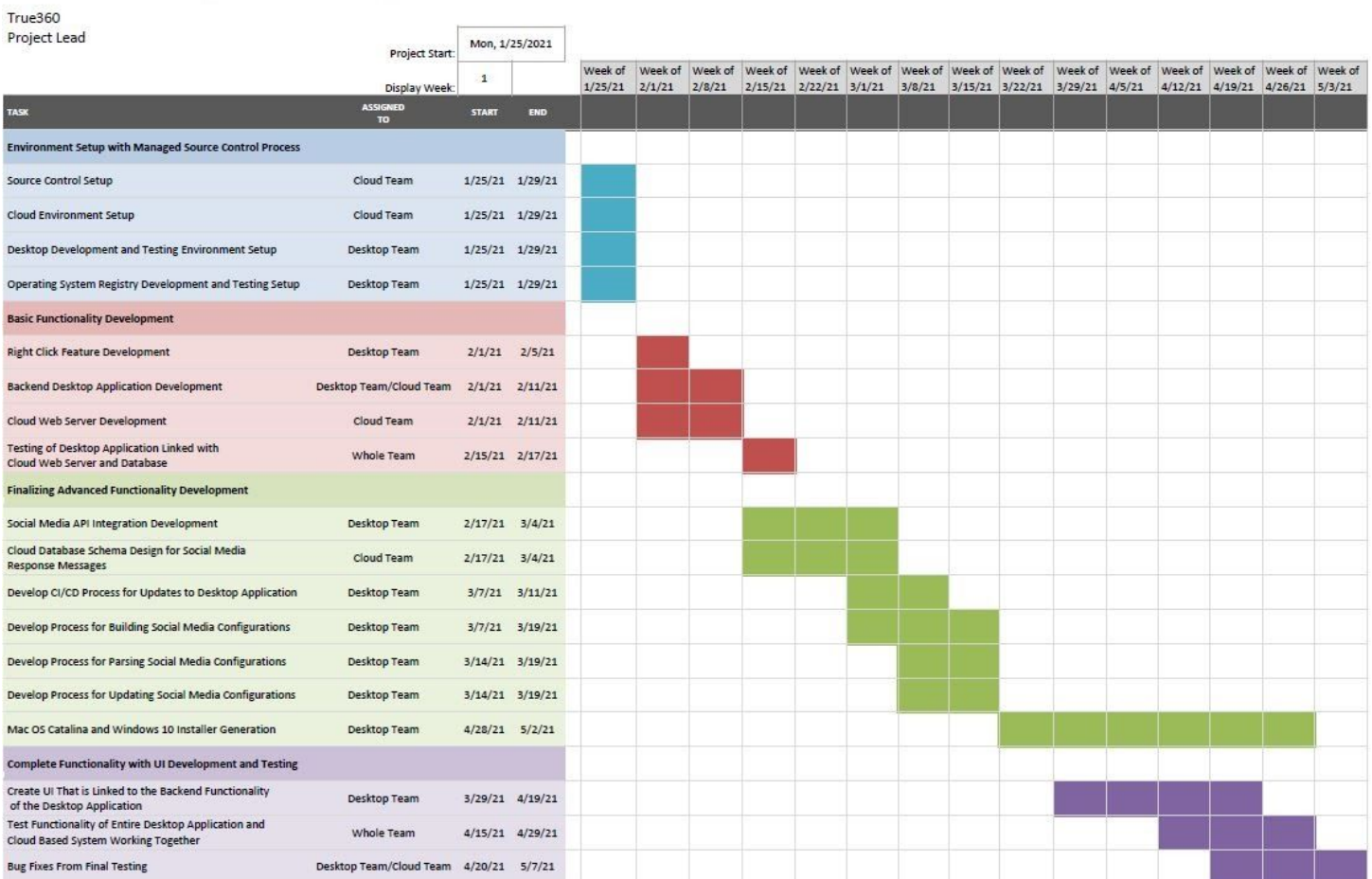


Figure 2. Gantt Chart of the Project Timeline

## 2.5 PROJECT TRACKING PROCEDURES

The 3 major tools that our group will use to track progress throughout the lifecycle of this project include Slack, Pivotal Tracker, and Gitlab.

In order to communicate effectively, our group will use Slack to communicate. This will allow us to determine when to schedule team meetings, advisor meetings, and meetings with True360 to keep them

informed. This will also allow for quick, informal communication to troubleshoot and resolve any problems, instead of waiting until the next meeting to bring the issue up.

In order to stay on track with our project timeline, we will be using Pivotal Tracker, a tracking tool similar to Trello. This will allow us to create stories to work on and keep track of what needs to be done and what is done. In addition, we will be able to brainstorm ideas as well as assign and prioritize certain tasks.

In order to keep the group’s software uniform and up to date, we will be using Gitlab. This will allow us easy access to update, manage, and receive the latest revisions of the project’s code. Additionally, it will allow for the group to resolve merge conflicts and allow for any changes in the code to be approved by other developers.

## 2.6 PERSONNEL EFFORT REQUIREMENTS

In order to keep track of effort levels for a task, we will be assigning each task with a number of hours to complete the task. These hours are generally assumed to be for the task overall, rather than per team member. Depending on the task, this means that if a task can be completed by one person in 2 hours, but everyone in the group has to do it, it will be listed as 14 (2 hours \* 7 team members) hours. These numbers also include the hours put into creating automated tests for the feature, as necessary.

Each member of the team is expected to complete approximately 5-7 hours of work per week.

Table A

Task	Effort Level (hours)	Explanation
Source Control Setup	14	Source Control is a simple process, and we all have experience with this, so we have assigned it 2 hours per member.
Cloud Environment Setup	10	The cloud environment needs to be deployed in a place that is accessible, but is otherwise a simple process that will take approximately 3 hours for 3 members.
Desktop Development and Testing Environment Setup	12	The desktop environment is similar to the cloud environment setup, but slightly longer since it includes the testing environment.
Operating System Registry Development and Testing Setup	12	Editing the registry can be a difficult task, but it is still simple in scope, thus it is given a 12.

Right Click Feature Development	6	Once it is known how to edit the registry, adding a registry key for a right click feature becomes a simple task.
Backend Desktop Application Development	60	The bulk of development for the desktop application is done within this task, and since it contains a large number of features, it will take 60 total hours.
Cloud Web Server Development	60	Similar to the backend desktop application, the cloud web server will also take a similar number of hours, since it is large in scope.
Testing of Desktop Application Linked with Cloud Web Server and Database	8	Since most of the testing is done when different functions are created, this task will have the majority of work already finished, with it needing minor additions.
Social Media API Integration Development	20	This task is largely dependent on the number of platforms, but it is estimated that it will take 5 hours for each of the 4 platforms we will support.
Cloud Database Schema Design for Social Media Response Messages	8	The database we will be using for social media will be simple, but it still requires thought. This should take 2 members about 4 hours.
Mac OS Catalina and Windows 10 Installer Generation	10	Electron supports creating installers, but we will need to edit this slightly by including editing of the registry keys, which will take slightly longer than the custom out of the box Electron installers.
Develop CI/CD Process for Updates to Desktop Application	10	Setting up the CI/CD will need to be done for both the desktop application and cloud server, but it is a straightforward process for anyone that has created a CI/CD pipeline before.

Develop Process for Building Social Media Configurations	20	This task is slightly more complex than other tasks because it requires taking into account many different other tasks.
Develop Process for Parsing Social Media Configurations	6	Assuming the configurations are made with a standard notation such as JSON, parsing is simple. It is assigned a few more hours to account for any errors that come up with the config associated with our code.
Develop Process for Updating Social Media Configurations	10	Once we know how to parse a configuration, we will be able to update it, but we will need to figure out what information needs to be updated.
Create UI That is Linked to the Backend Functionality of the Desktop Application	30	The UI requires more care since it is a product that will be given by a company. This requires it to be modern, and have significant research done into ensuring everything is working correctly.
Test Functionality of Entire Desktop Application and Cloud Based System Working Together	15	This task assumes that most functions are already tested via automated testing, but it requires more hours to ensure that we have covered all edge cases, and put the product through QA.

Table A. Tasks and Corresponding Effort Levels

2.7 OTHER RESOURCE REQUIREMENTS

Other resources that are required to complete this project include:

- Access to an AWS Web Server
- Access to an AWS Storage Bucket
- Access to a Central Database
- Designated Social Media Accounts for Testing Purposes
- Testing VMs for Mac OS and Windows 10
- Online Material and Information for Programming and Development Purposes

2.8 FINANCIAL REQUIREMENTS

No financial resources will be required to conduct the project.



Note: Sections 3-6 are not required for design document version 1

Updates to be implemented to section 1 & 2 (these were determined changes after v1 was due):

- Change “watermark” to stamp/logo since watermark means something different

## 3 Design

### 3.1 PREVIOUS WORK AND LITERATURE

Include relevant background/literature review for the project

- If similar products exist in the market, describe what has already been done
- If you are following previous work, cite that and discuss the **advantages/shortcomings**
- Note that while you are not expected to “compete” with other existing products / research groups, you should be able to differentiate your project from what is available

Detail any similar products or research done on this topic previously. Please cite your sources and include them in your references. All figures must be captioned and referenced in your text.

### 3.2 DESIGN THINKING

Detail any design thinking driven design “define” aspects that shape your design. Enumerate some of the other design choices that came up in your design thinking “ideate” phase.

### 3.3 PROPOSED DESIGN

Include any/all possible methods of approach to solving the problem:

- Discuss what you have done so far – what have you tried/implemented/tested?
- Some discussion of how this design satisfies the **functional and non-functional requirements** of the project.
- If any **standards** are relevant to your project (e.g. IEEE standards, NIST standards) discuss the applicability of those standards here
- This design description should be in **sufficient detail** that another team of engineers can look through it and implement it.

### 3.4 TECHNOLOGY CONSIDERATIONS

Highlight the strengths, weakness, and trade-offs made in technology available.

Discuss possible solutions and design alternatives

### 3.5 DESIGN ANALYSIS

- Did your proposed design from 3.3 work? Why or why not?
- What are your observations, thoughts, and ideas to modify or iterate over the design?

### 3.6 DEVELOPMENT PROCESS

Discuss what development process you are following with a rationale for it – Waterfall, TDD, Agile. Note that this is not necessarily only for software projects. Development processes are applicable for all design projects.

### 3.7 DESIGN PLAN

Describe a design plan with respect to use-cases within the context of requirements, modules in your design (dependency/concurrency of modules through a module diagram, interfaces, architectural overview), module constraints tied to requirements.

## 4 Testing

Testing is an **extremely** important component of most projects, whether it involves a circuit, a process, or software.

1. Define the needed types of tests (unit testing for modules, integrity testing for interfaces, user-study or acceptance testing for functional and non-functional requirements).
2. Define/identify the individual items/units and interfaces to be tested.
3. Define, design, and develop the actual test cases.
4. Determine the anticipated test results for each test case
5. Perform the actual tests.
6. Evaluate the actual test results.
7. Make the necessary changes to the product being tested
8. Perform any necessary retesting
9. Document the entire testing process and its results

Include Functional and Non-Functional Testing, Modeling and Simulations, challenges you have determined.

### 4.1 UNIT TESTING

– Discuss any hardware/software units being tested in isolation

### 4.2 INTERFACE TESTING

– Discuss how the composition of two or more units (interfaces) are to be tested. Enumerate all the relevant interfaces in your design.

### 4.3 ACCEPTANCE TESTING

How will you demonstrate that the design requirements, both functional and non-functional are being met? How would you involve your client in the acceptance testing?

### 4.4 RESULTS

- List and explain any and all results obtained so far during the testing phase

- Include failures and successes
- Explain what you learned and how you are planning to change the design iteratively as you progress with your project
- If you are including figures, please include captions and cite it in the text

## 5 Implementation

Describe any (preliminary) implementation plan for the next semester for your proposed design in 3.3.

## 6 Closing Material

### 6.1 CONCLUSION

Summarize the work you have done so far. Briefly re-iterate your goals. Then, re-iterate the best plan of action (or solution) to achieving your goals and indicate why this surpasses all other possible solutions tested.

### 6.2 REFERENCES

List technical references and related work / market survey references. Do professional citation style (ex. IEEE).

### 6.3 APPENDICES

Any additional information that would be helpful to the evaluation of your design document.

If you have any large graphs, tables, or similar data that does not directly pertain to the problem but helps support it, include it here. This would also be a good area to include hardware/software manuals used. May include CAD files, circuit schematics, layout etc., PCB testing issues etc., Software bugs etc.