Software Project Management Plan

Venom

Fall 2023

Joshua Zhu (Project Manager)

Anthony Arocha (Assistant Project Manager)

Nathaniel Wallace (Software Engineer)

Juan Trujillo (Software Engineer)

Armando Magallon (Software Engineer)

Marissa Bonilla (Software Engineer)

Mario Perez (Software Engineer)

1. Overview
    - 1.1 Project summary
        - 1.1.1 Purpose, scope, and objectives
        - 1.1.2 Assumptions and constraints
        - 1.1.3 Project deliverables
        - 1.1.4 Schedule and budget summary
    - 1.2 Evolution of the plan
2. References
3. Definitions
4. Project organization
    - 4.1 External interfaces
    - 4.2 Internal structure
    - 4.3 Roles and responsibilities
5. Managerial process plans
    - 5.1 Start-up plan
        - 5.1.1 Estimation plan
        - 5.1.2 Staffing plan
        - 5.1.3 Resource acquisition plan
        - 5.1.4 Project staff training plan
    - 5.2 Work plan
        - 5.2.1 Work activities
        - 5.2.2 Schedule allocation
        - 5.2.3 Resource allocation
        - 5.2.4 Budget allocation
    - 5.3 Control plan
        - 5.3.1 Requirements control plan
        - 5.3.2 Schedule control plan
        - 5.3.3 Budget control plan
        - 5.3.4 Quality control plan
        - 5.3.5 Reporting plan
        - 5.3.6 Metrics collection plan
    - 5.4 Risk management plan
    - 5.5. Closeout plan
6. Technical process plans
    - 6.1 Process model
    - 6.2 Methods, tools, and techniques
    - 6.3 Infrastructure plan
    - 6.4 Product acceptance plan
7. Supporting process plans
    - 7.1 Configuration management plan
    - 7.2 Verification and validation plan
    - 7.3 Documentation plan
    - 7.4 Quality assurance plan
    - 7.5 Reviews and audits
    - 7.6 Problem resolution plan
    - 7.7 Subcontractor management plan
    - 7.8 Process improvement plan

1. Overview

1.1. Project Summary

The Software Project Management Plan lays out the details of the management plan that will be followed in order to properly create the Venom Game Hosting website. This document will include the development cycle, organization, the specific role that everyone plays and the testing protocols with the project timeline for all items.

1.1.1. Purpose, Scope, and objectives

The Software Project Management Plan (SPMP) will provide all the details pertaining to the development plan and cycle for the entire project. It will assign each member their roles and responsibilities throughout the project, as well as how they will be completing their assigned tasks. It will also create projected deadlines and an overall timeline for the entire project. It will follow the specifications stated in the most current version of our SRS that has been signed by the client. The first prototype will mainly consist of the UI design. The user should be able to navigate through the different pages present on the web application. The website itself will have a logo created and dummy games filled out for the space. The purpose for creating this game hosting website is to give epoelple a space to have fun and play games without having to download an other applications. This website will also help aspiring game developers test out ideas and get feedback to the games they create and upload.

1.1.2. Assumptions and Constraints

Here is a list of all the assumptions and constraints

- Team members will attend all meetings (in class and out)
- Team members will meet all given deadlines
- Team members will follow requirements set by the SRS
- Team members will have to learn Firebase, Cloud Firestore and Git repository if they do not already know how to use these.
- We will assume the users will be competent enough to navigate their computer to access the website

1.1.3. Project Deliverables

- Working website
- SRS, SPMP, Documented Source Code, Software Architecture, Maintenance Manual

1.1.4.Schedule and Budget Summary

There is no budget provided to the team for this project, but the team will work as efficiently as they can as if there were one.

1.2. Evolution of the Plan

The entire team met with the client on September 5th, 2023 to discuss the expectations and functionality of the project. From there, the team met and further discussed the design, features, and implementation of the project. We then split the team into front end and back end, and each team discussed how they were going to work everything out.

2. References

Github

Firebasef

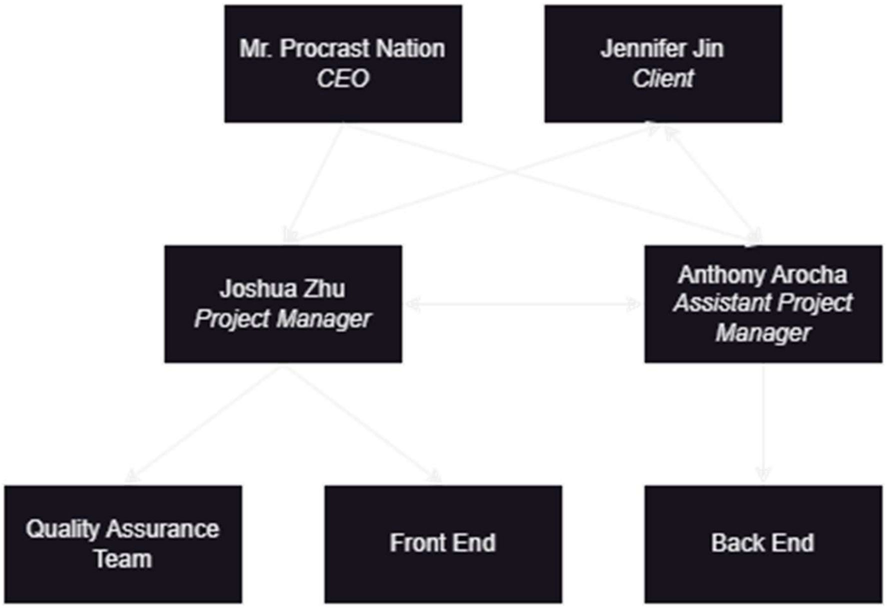
3. Definitions

Algorithm: A finite sequence of well-defined, computer-implementable instructions, typically to solve a class of problems or to perform a computation.

Cloud Firestore: A flexible, scalable database for mobile, web, and server development from Firebase and Google Cloud Platform.

Computer: A machine that can be instructed to carry out sequences of arithmetic or logical operations automatically via computer programming. These include desktops and laptops.
CSS: Cascading Style Sheets is a style sheet language used for describing the presentation of a document written in a markup language such as HTML.
Firebase: A platform developed by Google for creating mobile and web applications.
Homepage: The primary page of the web application.
HTML: Hypertext Markup Language is the standard markup language for documents designed to be displayed in a web browser.
HTTP: Hypertext Transfer Protocol. A session layer protocol for distributed, collaborative, hypermedia information systems.
Internet Browser: A software application for accessing information on the World Wide Web.
Javascript: A programming language that conforms to the ECMAScript specification.
Real-time Web Database: A database system that uses real-time processing to handle workloads whose state is constantly changing.
SRS: This document, the Software Requirements Specification (SRS), that illustrates the functional and non-functional requirements of the 4-Wheel Drive Used Car Dealership web application.
UID: Unique Identifier used by Firebase Authentication. Each user account created is given a UID.
User: Someone who uses the web application.
User Credentials: A user’s username and password used to sign into their account.
Web Application: An application software that runs on a web server. Web applications are accessed by the user through a web browser with an active internet connection. The term will also be used synonymously with website.

4. Project Organization
   4.1. External interfaces



Jennifer Jin (Client) – Communicates directly with the Project and Assistant Project managers for the requirements of the web application. She also gives feedback on the prototypes that will be presented and answer any questions that either manager bring up

Joshua Zhu (Project Manager) – Is the main line of communication between the client and the software engineering team. He will also make contact with the quality assurance team after they run their tests and discuss what needs to be changed. He is in charge of all the paperwork and documentation for the team. He also monitors the progress of the front end of the website, putting his input for the page.

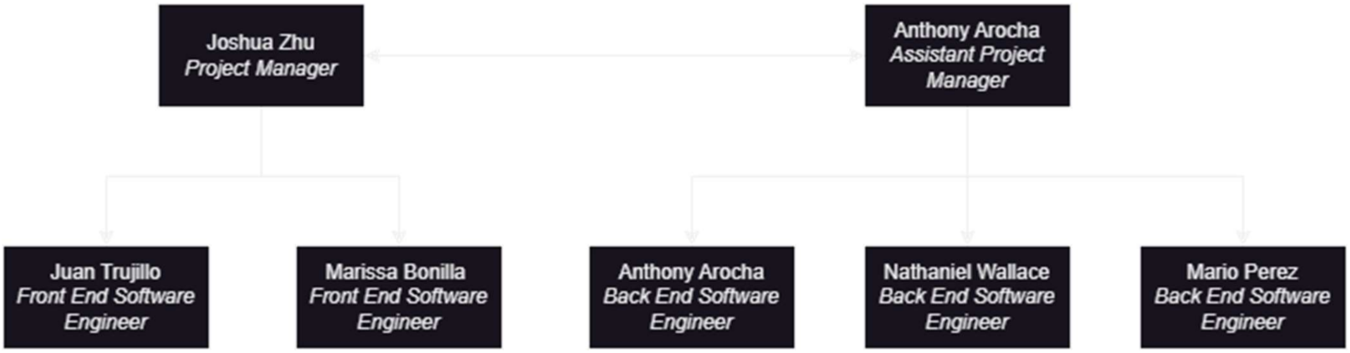
Anthony Arocha (Assistant Project Manager) – Monitors the progress of the back end and the overall goal of the project. In the event that the project manager is absent, the APM will take over the responsibility of the Project Manager.

Front End Team – Is responsible for how the website looks and the layout of it.

Back End Team – Is responsible for the database and the features of the website

Quality Assurance Team – Evaluates the prototype provided and tests for user error and/or worst case scenarios.

4.2. Internal Structure


Joshua Zhu
Project Manager
Anthony Arocha
Assistant Project Manager
Juan Trujillo
Front End Software Engineer
Marissa Bonilla
Front End Software Engineer
Anthony Arocha
Back End Software Engineer
Nathaniel Wallace
Back End Software Engineer
Mario Perez
Back End Software Engineer

Although the project manage is directly responsible for the front end engineers and the assistant project manager is responsible for the back end engineers, both managers work together to help move the project along at a steady pace. Both managers communicate in order to make sure that the back end has what the front end needs and vice versa.

4.3. Roles and Responsibilities

Joshua Zhu (Project Manager) – Communicate with the team what the client wants. Responsible for all the documentation of the project

Anthony Arocha (Assistant Project Manager) – works with the project manager and the client to complete the website.

Juan Trujillo (Front End Software Engineer) – Focuses on Sign up, login and about us page

Marissa Bonilla (Front End Software Engineer) – Focuses on the homepage, individual game page and logo

5. Managerial Process Plans

5.1. Start-up Plan

5.1.1. Estimation Plan

- Discuss with the client about the specifications on the web application

- Figure out the hardware necessary for engineers to create application up to specification
- Understand and set deadlines for project
- Set individual goals for prototypes 1 and 2
- Break each prototype down to individual targets, create deadlines for those targets and assign engineers for those targets.

5.1.2. Staffing Plan

The project staff was peer selected and the professor/client received a filled out form of the make up of the team in the first week of class. Since the first week, we have had 2 more members join the team.

5.1.3. Resource Acquisition Plan

All necessary software for this project is free for all members. Engineers just need to download their own IDE's, the Git repository is online and accessible by all team members.

5.1.4. Project Staff Training Plan

All team members have previous knowledge on the software language, however some members needed some time to learn how to access, use and modify the Git repository. We also took some time to learn about the database we plan on using.

5.2. Work Plan

5.2.1. Work Activities

- Prototype 1: The first prototype will be a function website with the ability to create an account, play games and change the user's account settings. The front end team will help develop the website, it's layout and the overall looks of the application, while the back end team will create the necessary database and work on how to link the two parts together.
- Prototype 2: The second prototype will make sure that the games in the database will be properly integrated onto the website, making sure that the developer uploading games and the communication between themselves and the administrators goes smoothly. The entire team will work on this as well as any other bugs that the QA finds.
- Documentation: All members will make sure that the code they write is documented and understandable.

5.2.2. Schedule allocation

5.3. Control Plan

5.3.1. Requirements Control Plan

Each team member of the project is required to attend the meetings in the classroom as well as our online meetings on Discord. If they are unable to attend any of the meetings, they are to let the managers know. Managers or fellow team members who did attend will give a short summary of what happened during those meetings and if any new information was give, relay them over. Additionally, each team member is required to document their code according to the SRS and meet the given deadlines. If something requires more time, then team members should let the their respective manager know and we can assess the workload and add more time or add an additional team member to that assignment if needed.

5.3.2. Schedule Control Plan

As stated before, the team is required to attend both meetings, in the classroom and on Discord, in order to finish the project in a timely manner. Managers will make sure that each members is completing their responsibilities on time.

5.3.3. Budget Control Plan

This project has no budget allocated for it, however the project will still be treated as if there is one so that we work as efficiently and effectively as possible.

5.3.4. Quality Control Plan

The management and engineering teams will both be performing quality checks on all of the software produced. These tests will be done during the meetings on Discord. Additionally, third party Quality Assurance teams will conduct tests on the web application down the line to insure that nothing gets missed.

5.3.5. Reporting Plan

The management team will be responsible for contacting the CEO for meetings as well as staying in contact with the client, Professor Jin, to insure that everything is going accordingly.

5.3.6. Metrics Collection Plan

The management team will make sure that each developer is completing their tasks and that the project is on track through meetings and progress reports.

5.4. Risk Management Plan

Development

- The team will meet regularly to make sure that the production is not stopped at any point in time.
- The scheduled deadlines will be followed by everyone on the team to ensure that the project is not delayed.
- Each team member will stay up to date with their individual tasks, and inform the management team if any issues do arise so that a plan can be made on how to work the situation out.

Project Failure

- If the technology does not exist, or the current system is unable to run the web application, the managing team will discuss with the client to make sure that something else can be done so that the client is still satisfied.

5.5. Close-out Plan

The team will submit the entirety of the project in the Github repository, with the maintenance manual and will present the final revision of the application on the day of the final.

6. Technical Process Plans

6.1. Process Model

Scrum model

6.2. Methods, Tools and Techniques

Method: Incremental Development Model
Tools: Github repository, Firebase server
Techniques: regular meetings with team and client

6.3. Infrastructure Plan

As long as the server that the website is hosted on is still up, then this web application will be able to stay online.

6.4. Product Acceptance Plan

The client will be kept in the loop throughout the entirety of the project development. Not only will we be doing internal quality control tests, but outside Quality Assurance teams will also test the application to make sure that the prototypes are working properly.

7. Supporting Process Plans

7.1. Configuration Management Plan

We are using Github for the configuration and will hold all of the changes made to the code. This will allow for a smoother configuration throughout the entire project.

7.2. Verification and Validation

Periodic testing of the web application will be done throughout the project. Any bugs or error will be reported, documented, and promptly fixed.

7.3. Documentation Plan

The management team will be in charge of all the documentation for this project.

7.4. Quality Assurance Plan

The quality assurance team will make sure that the application is working condition and report any bugs or errors.

7.5. Reviews and Audits

During development and testing phases, every member will be testing their own work and reporting any errors that show up. Any design flaws or bugs will be report and documented for an immediate fix.

7.6. Problem Resolution Plan

Each member on the engineering team will keep their respective managers up to date on any issues that arise. The management team will discuss on how to resolve these issues without impacting the deadlines too much. If a change in deadline or reallocation of resources is needed, they will make those calls.

7.7. Subcontractor Management Plan

There are no subcontractors for this project.

7.8. Process Improvement Plan

In order to improve the process and development of the software, it is crucial to write a good documentation of the source code that they will be provided. There are features that cannot be implemented in the given time at the moment, and therefore have been pushed back to the next stage of development. Writing quality code now, documenting it and creating a good maintenance manual will help the future development team to make all the changes that they think are necessary. We also made sure that the client was kept up to date on these features, and so he will also be able to help the future development team figure out what to do next.