**Project Plan**

ViroTour Server Processing

University of Maryland Global Campus

Software Engineering 670

Spring Cohort 2023

Team B

Logo

Description automatically generated

**Document Control**

**Document History**

| Version | Issue Date | Changes |
| --- | --- | --- |
| 0.1 | 1/28/2023 | Initial draft. |
| 2.0 | 2/12/2023 | Revisions after milestone 1. |
| 3.0 | 3/25/2023 | Milestone 3 Revisions |

**Table of Contents**

[1 Project Information 3](#_Toc28841136)

[1.1 Purpose 4](#_Toc128989981)

[1.2 Statement of Need 4](#_Toc205167966)

[1.3 Vision Statement 4](#_Toc984762227)

[1.4 Stakeholders 4](#_Toc2085020649)

[1.5 Project Methodology 5](#_Toc922830428)

[1.6 Project Tools 5](#_Toc1296366951)

[2 Scope Management 5](#_Toc600457530)

[2.1 Scope 6](#_Toc127169371)

[2.2 Work Breakdown Structure (WBS) 6](#_Toc516101853)

[2.3 Deployment Plan 10](#_Toc699999997)

[2.4 Change Control Management 10](#_Toc697539946)

[2.4.1 Change Control Board (CCB) 10](#_Toc495927698)

[2.4.2 Internal Change Requests 11](#_Toc498353383)

[2.4.3 External Change Requests 11](#_Toc453907816)

[2.4.4 Change Request Process 11](#_Toc1645394946)

[2.5 Resources 12](#_Toc59794105)

[3 Time Management 13](#_Toc51204764)

[3.1 Schedule 14](#_Toc1823988588)

[3.2 Milestones 15](#_Toc1913385239)

[3.3 Dependencies 15](#_Toc1370332691)

[4 Cost Management 16](#_Toc121419391)

[5 Quality Management 16](#_Toc31807609)

[5.1 Code 17](#_Toc862425990)

[5.2 Documentation 17](#_Toc941531942)

[5.3 Software 17](#_Toc1461998280)

[6 Staffing Management 17](#_Toc2034333380)

[7 Communications Management 19](#_Toc165200110)

[7.1 Microsoft Teams 19](#_Toc1314820893)

[7.2 GitHub 20](#_Toc1158718525)

[7.3 Taiga 20](#_Toc1716115657)

[8 Risk Management 20](#_Toc1488905927)

[8.1 Risk Analysis 21](#_Toc731066387)

[8.2 Issue Management 21](#_Toc1509338996)

[8.3 Risk Matrix and Register 21](#_Toc1581057874)

[9 Appendix A – Acronyms and Definitions 23](#_Toc1303792502)

[10 Appendix B – Detailed Project Plan 23](#_Toc971313051)

[11 Appendix D – References 24](#_Toc879938823)

# Project Information

## Purpose

ViroTour is an application that will automatically generate virtual tours. The application will provide features to enable users to customize the virtual tours according to their needs.

The purpose of this Project Management Plan (PMP) is to provide information about ViroTour to our stakeholders. The intended audience of this document includes the client, project sponsor, leadership, and the project development team. This document is designed to communicate the procedures which the server processing project team will follow throughout the project lifecycle.

This PMP will document how the project team will create, track, and deliver tasks, as well as the approach and completion criteria for each deliverable. It will outline the schedule of delivery for every milestone in the project.

## Statement of Need

ViroTour is intended to provide tour creators with an application to create 360 virtual tours by uploading images of their desired tour location. ViroTour will automatically create 3D tours, place hotspots algorithmically, provide the ability to edit hotspots, scan images to extract text, and share snippets of the tour with others.

## Vision Statement

Our vision for the ViroTour application is to provide a tool for photographers to create virtual tours without added complexity. By delivering these capabilities in the ViroTour application, tour visitors can conduct self-guided tours, search text within the tour location, and take screen grabs of any part of the tour and share them.

## Stakeholders

Stakeholders are individuals or entities that have interest in the outcome of the project and may affect the requirements and schedule. The stakeholders for ViroTour are listed below:

|  |  |
| --- | --- |
| Stakeholder | Project Role |
| Dr. Mir Assadullah | Client/Professor |
| Roy Gordon | Project Mentor |
| Rob Wilson | DevSecOps Mentor |
| Ivelin Tchangalov | Project Manager (PM) |
| KC Harden | Product Owner (PO) |
| Alex Armel Wabo Tebu | Business Analyst (BA) |
| Hang Wang | Sr. Software Engineer (Dev Lead) |
| Nancy Lay | Test Manager (Test Lead) |
| Kelvin Huynh | Jr. Software Engineer (Dev) |
| Melika Shahani | Jr. Software Engineer (Dev) |
| Jean Pita Diomi Kazadi | Jr. Software Engineer (Dev) |
| Jah-wilson Teeba | Jr. Software Engineer (Dev) |
| Ronald Milligan | Jr. Software Engineer (Dev) |
| Ian Fischer | Jr. Software Engineer (Dev) |
| Team A | Application Development Team |

**Table 1.4 – Stakeholder information.**

## Project Methodology

This project will be executed using a modified Scrum Model. Due to the nature of Master’s courses, the project team, Professor/client, and mentors have varied availability. Therefore, Scrum ceremonies such as daily standups, sprint planning, sprint reviews, and retrospective cannot occur with regularity and full participation.

With a modified Scrum Model, our project team can work with these constraints. We will hold weekly project team meetings to plan and review our work. The PM and PO will also attend meetings with the Project Mentor to ensure that we are on the correct path.

Although there are roles appointed for this project, the entire project team is expected to act as a cross-functional team to support all aspects of the project including project documentation, planning, execution, and testing.

## Project Tools

The tools used to manage the execution of ViroTour are listed below:

| Name | Description | Version |
| --- | --- | --- |
| Microsoft Teams | A messaging application the project team uses to collaborate, hold meetings, and share files. | N/A |
| SharePoint | A web-based Microsoft platform that allows the project team to manage and store documents. | N/A |
| Microsoft Word | A software-as-a-service (Saas) tool used for creating and editing word documents. | N/A |
| Microsoft Excel | A software-as-a-service (Saas) tool used for creating and editing excel documents. | N/A |
| Pencil | A GUI prototyping tool used to create mockups. | 3.1.0 |
| Taiga | A cloud-based agile project management tool that allows the project team to manage work. | N/A |
| GitHub | A software development internet tool used for version control using Git. | N/A |
| Windows Server | A database server to store information | WS 2022 |
| IDE (Android Studio) | An Integrated Development Environment for building software. | 2022.1.1 |

Table 1.6 – Project Tools

# Scope Management

## Scope

The goal of this project is to develop a virtual tour web and mobile application to be delivered by the Software Engineering Project Capstone class of Spring 2023. Two teams were created, Team A and Team B, and they will be responsible for application development and server-side development, respectively. This PMP is written for the server-side project. Thus, the requirements will only document the server-side requirements for this project.

**In Scope for Team B:**

* Database to store Images, Locations, Hotspots, and Text.
* Core Module: Image processing to enable tour navigation.
* Editing Module: Perform modifications on Images, Locations and Hotspots.
* Search Module: Perform text-based search.
* Backend support for any number of Virtual Tours.

**Out of Scope for Team B:**

Team A’s Responsibilities:

* User Interface foundational design.
* Ability to display a location.
* Ability to navigate from one location to another.
* Ensuring smooth transitions between locations.
* Ability to zoom and pan the view.
* Long press clip function.
* Displaying UI elements of the search results.
* Displaying UI elements of hotspot editing feature.
* Ability to view tour using VR viewer.
* Functional testing.

Other:

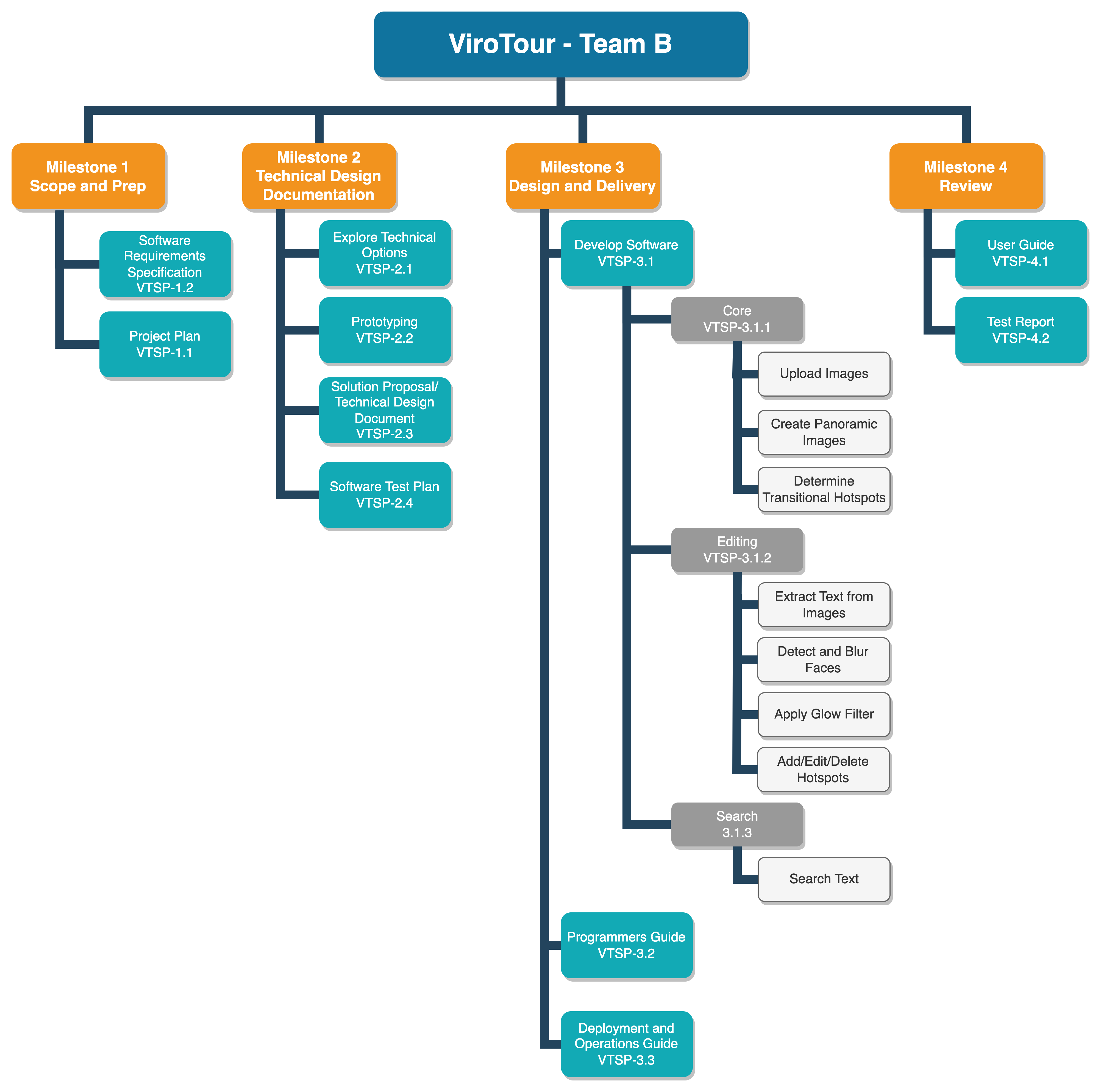
* User profile management.
* User authorization.
* Performance testing.

## Work Breakdown Structure (WBS)

This project has 4 major milestones each with added deliverables and the resubmittal of updated documents. For each milestone, the following can be expected.

* + Milestone 1 – Project Management Plan (PMP), as well as the Software Requirements (SRS) documents.
  + Milestone 2 – Technical Design Document (TDD), Software Test Plan, and updated SRS and PMP.
  + Milestone 3 – Application Demo, Deployment and Operations Guide, Programmers Guide (PG), and updated documents from prior milestones.
  + Milestone 4 – User Guide (UG), Test Report (TEST), application code, and updated documents from prior milestones.

Each milestone is concluded with a presentation to the professor. For more information related to each milestone, refer to the WBS below and/or the project schedule.



**Figure 2.2 - Work Breakdown Structure**

## Deployment Plan

The deployment plan will be established during Milestone 2. We will provide details about how to build and deploy the project through mobile and web platforms. A few of the steps we will need to take are outlined on Flutter’s online documentation (“Build and release,” n.d):

* 1. Signing the application with a digital signature.
  2. Creating an upload keystore.
  3. Shrinking the code.
  4. Reviewing the app’s manifest.
  5. Building the app for release.

## Change Control Management

In the event there is a change identified by the client, project team, or Team A, the team will utilize the following process to record, evaluate, and prioritize each request. In the event that a change request is accepted into the project teams' backlog, the new requirement will go through Team A’s change request process to evaluate whether the request affects their project requirements.

### Change Control Board (CCB)

The group of individuals responsible for reviewing, approving, and managing changes to the project are listed below. They are responsible for ensuring any changes made to the project requirements and/or system are properly evaluated and documented.

|  |  |  |
| --- | --- | --- |
| Project Role |  | Responsibility |
| Project Manager | Ivelin Tchangalov | -Coordinates the CCB  -Ensures the project schedule is updated to reflect any approved changes |
| Product Owner | KC Harden | -Manages the backlog and prioritizes changes  -Liaison between requestor and development team to ensure the change is delivered as requested |
| Development Team | Hang Wang  Nancy Lay  Kelvin Huynh  Melika Shahani  Jean Pita Diomi Kazadi  Jah-wilson Teeba  Ronald Milligan  Ian Fischer | -Responsible for assessing the impact of change including impact on project timeline  -Identify any risks associated with the change |
| Test Lead | Nancy Lay | -Responsible for assessing the impact of change on the testing strategy  -Ensures all tests are updated to cover new functionality |

**Table 2.4.1 - Change Control Board**

### Internal Change Requests

Change requests that derive from the project team will be collected and discussed internally and evaluated for viability. If the change request is deemed necessary and directly affects the client, the Project Manager and Product Owner will gain approval for the change from the client before prioritizing and executing the change.

### External Change Requests

External requests can derive from the client and/or Team A and will follow the change request process defined in section 2.4.3. The requirement will be discussed with the team for viability and if accepted, the request will be properly groomed and prioritized. In the event the request conflicts with the project requirements, the Project Manager and/or Product Owner will discuss further with the requestor.

### Change Request Process

|  |  |
| --- | --- |
| Step | Description |
| Create Taiga Task | A team member determine what needs to be change and create Taiga task |
| Complete Task and Create Code Review | Team member completes a CR Form and sends the completed form to the Change Manager |
| Authorize | Approval to move forward with incorporating the suggested change into the project/product |
| Implement | If approved, make the necessary adjustments to carry out the requested change and communicate CR status to the submitter and other stakeholders |

**Table 2.4.4 - Change Request Process**

Change request form and change management log

|  |  |
| --- | --- |
| Element | Description |
| Date | The date the CR was created |
| Code Review# | Assigned by the Change Manager |
| Title | A brief description of the change request |
| Description | Description of the desired change, the impact, or benefits of a change should also be described |
| Submitter | Name of the person completing the CR Form and who can answer questions regarding the suggested change |
| Product | The product that the suggested change is for |
| Version | The product version that the suggested change is for |
| Priority | A code that provides a recommended categorization of the urgency of the requested change (High, Medium, Low) |

**Table 2.4.4 - Change Request Form and Log**

Evaluating and authorizing change request

|  |  |
| --- | --- |
| Priority | Description |
| High | The change is needed as soon as possible because of potentially damaging service impact. |
| Medium | The change will solve irritating problems or repair missing functionality. This change can be scheduled. |
| Low | The change will lead to improvements, changes in workflow, or configuration. This change can be scheduled. |

**Table 2.4.4 - Change Request Evaluation and Authorization**

Change requests are evaluated and assigned one or more of the following change types

|  |  |
| --- | --- |
| Type | Description |
| Scope | Change affecting scope |
| Time | Change affecting time |
| Cost | Change affecting cost |
| Resources | Change affecting resources |
| Deliverables | Change affecting deliverables |
| Product | Change affecting product |
| Quality | Change affecting quality |

**Table 2.4.4 - Change Request Evaluation Criteria**

Change request are evaluated and assigned on of the following status types

|  |  |
| --- | --- |
| Status | Description |
| Open | Entered/Open but not yet approved or assigned |
| Work in Progress | CR approved, assigned, and work is progressing |
| In Review | CR work is completed and in final review prior to testing |
| Testing | CR work has been reviewed and is being tested |
| Close | CR work is complete, has passed all tests, and updates have been released. |

**Table 2.4.4 - Change Request Status Types**

## Resources

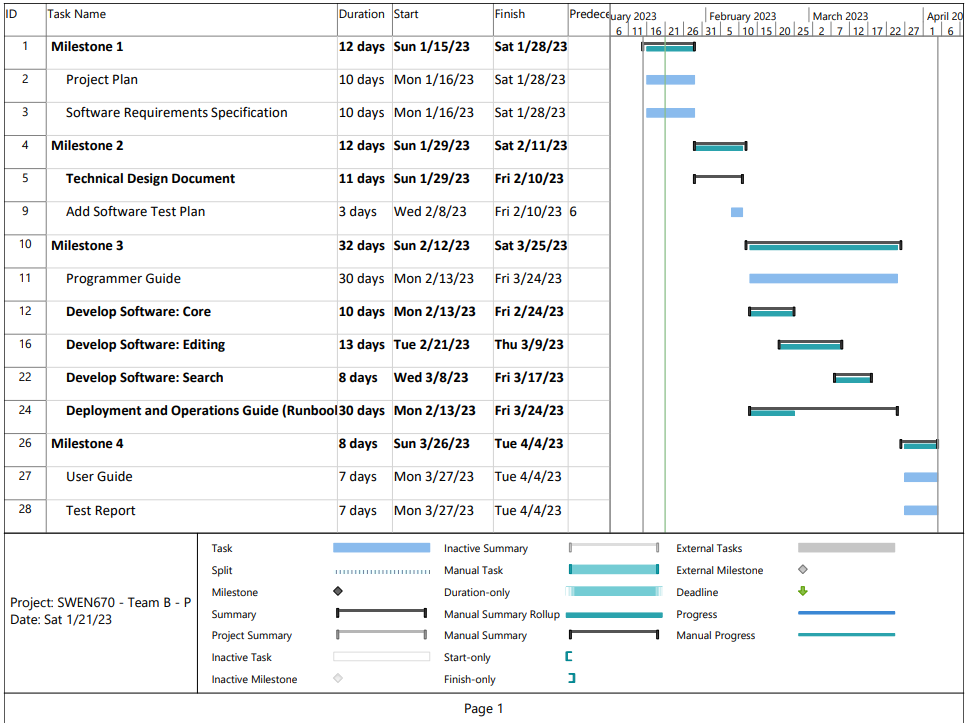
The following table identifies the resources responsible for the tasks outlined in the project schedule.

|  |  |
| --- | --- |
| Task | Resource |
| Project Plan | Project Manager  Product Owner  Software Engineer |
| Software Requirements Specification | Project Manager  Software Engineer |
| Technical Design Document | Software Engineer |
| Software Test Plan | Test Manager |
| Programmer Guide | Software Engineer  DevSecOps Mentor |
| Software Development | Software Engineer |
| Deployment and Operations Guide | Software Engineer  DevSecOps Mentor |
| User Guide | Product Owner  Software Engineer |
| Test Report | Test Manager |
| Software Fixes | Software Engineer |
| Project Delivery | Project Manager  Product Owner |

**Figure 2.5 – Project Resources**

# Time Management

## Schedule

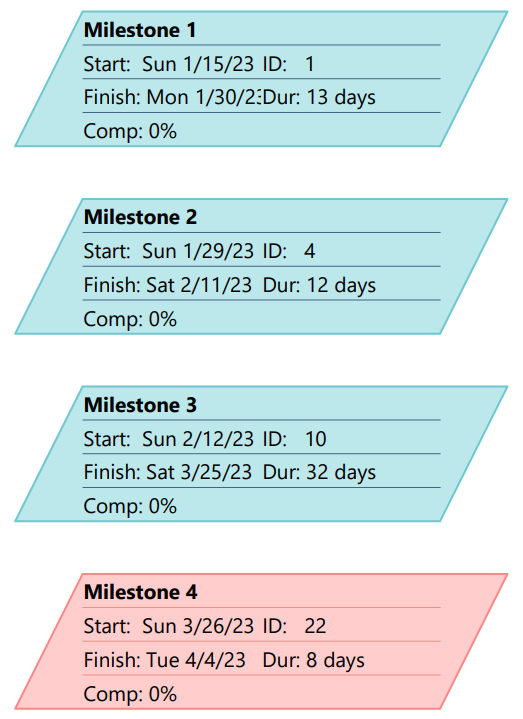


**Figure 3.1 – Project Schedule**

**Please refer to “Appendix B – Detailed Project Schedule” for more details.**

## Milestones

Milestones contain the main deliverables of the project.



**Figure 3.2 – Milestone Overview**

A milestone is “a major event in the project” and represents the completion of a set of activities. The milestones for the project are shown in table 3.2.

|  |  |
| --- | --- |
| **Description** | **Delivery Date** |
| **Milestone 1** | |
| Project Plan | Week 3 (1/28) |
| Software Requirements Specification (SRS) | Week 3 (1/28) |
| Presentation 1 | Week 3 (1/28) |
| **Milestone 2** | |
| Technical Design Document | Week 5 (2/11) |
| Programmer Guide | Week 5 (2/11) |
| Add Software Test Plan to the Project Plan | Week 5 (2/11) |
| **Milestone 3** | |
| Deployment and Operations Guide (Runbook) | Week 10 (3/25) |
| Deploy version 1 | Week 10 (3/25) |
| **Milestone 4** | |
| Test Report | Week 12 (4/4) |
| User Guide | Week 12 (4/4) |
| Programmer Guide | Week 12 (4/4) |
| Deploy version 2 | Week 12 (4/4) |
| Presentation 2 | Week 12 (4/4) |

Table 3.2 – Project Milestones

## Dependencies

|  |  |  |
| --- | --- | --- |
| **Activity** | **Depends on** | **Dependency** |
| Project plan, software Requirements Specification, Presentation 1 | Team assembly, requirement analysis, Business case | Start to start |
| Technical Design Document, User guide, Programmer Guide, Deploy Prototype | Project plan, Software Requirements Specification | Finish to start |
| Deployment and Operations Guide, Software Test Plan, Deploy 1 | Technical Design Document, User guide, Programmer Guide, User Prototype | Finish to start |
| Test Report, User Guide, Programmer Guide, Deploy 2, Presentation 2 | Deployment and Operations Guide, Deploy 1, Test plan | Finish to finish |

**Table 3.3 - Project Dependencies**

# Cost Management

The ViroTour Project has no realized cost to the Customer throughout the project. The cost analysis of the salaries listed below were retrieved in January 2023 from salary data provided publicly by Zip Recruiter. Team B members will work on the project for 11 consecutive weeks, an average of 1-2 hours Monday through Friday.

The cost analysis/predictions are for the total life span of the project, which is 11 weeks. The table covers weekly work hours, hourly pay per professional, and a weekly pay analysis. Team B can provide a cost analysis report on every milestone completion by the project manager or assigned member.

|  |  |  |  |
| --- | --- | --- | --- |
| **Role** | **Hourly Rate** | **Weekly Cost** | **Total Cost** |
| Product Owner (L5) | $95.00 | $950.00 | $10,450.00 |
| Product Manager (L5) | $90.00 | $900.00 | $9,900.00 |
| Senior Software Lead (L4) | $75.00 | $750.00 | $8,250.00 |
| Test Manager (L4) | $75.00 | $750.00 | $8,250.00 |
| Junior Software Developer (L3) | $65.00 | $650.00 | $7,150.00 |
| Junior Software Developer (L3) | $65.00 | $650.00 | $7,150.00 |
| Junior Software Developer (L3) | $65.00 | $650.00 | $7,150.00 |
| Junior Software Developer (L3) | $65.00 | $650.00 | $7,150.00 |
| Junior Software Developer (L3) | $65.00 | $650.00 | $7,150.00 |
| Junior Software Developer (L3) | $65.00 | $650.00 | $7,150.00 |
| Business Analyst (L3) | $65.00 | $650.00 | $7,150.00 |
| **Total** | **$790.00** | **$7,900.00** | **$86,900.00** |

**Table 4.1 – Project Cost**

# Quality Management

Quality management is the act of overseeing all activities and tasks needed to maintain a desired level of excellence. Establishing good quality posture ensures that the software being produced meets the requirements established by the customer. As software is developed through an iterative process, quality metrics can be collected to help gauge the level of quality in key areas. Quality will be managed in the areas regarding code, documentation and software.

## Code

Code quality will be managed by defining a code review process. We will review proposed code changes and perform smoke tests to validate them before merging them to our stable branches in git. Code reviews and testing at the development stage of the software development life cycle, reduces the number of bugs and defects found when software is released.

## Documentation

Documentation serves as a communication aide to help clarify requirements, designs, models, etc. It’s important to continuously review, update and maintain all documentation to ensure accuracy. In this project, we will manage documentation quality by focusing on the 3 C ‘s of documentation which are:

* Compliance – each member of the project will need to perform every task the same way,
* Consistency – things that belong to the same entity should look the same,
* Completeness – lack of it will affect or destabilize the result of the project.

## Software

Software quality will be managed by establishing processes, resources, and procedures for testing throughout the SDLC. We will develop a test plan according to our use cases to ensure the validity of our system.

# Staffing Management

Team B will be composed of the Project Manager, Product Owner, Senior Dev Lead, Test Manager, Business Analyst, and Junior Dev’s. Team members will be cross-functional. Figure 6.1 displays the RACI matrix for this project. Table 6.1 shows the responsibilities of each of the team members for this project.

Table, treemap chart

Description automatically generated

Text

Description automatically generated with medium confidence**Figure 6.1 - RACI Matrix**

|  |  |  |
| --- | --- | --- |
| **Resource** | **Role** | **Responsibilities** |
| Dr. Mir Assadullah | Client | * Provides business context, expertise, and guidance to the project manager and the team. * Acts as the link between the project, the business, and strategic level decision-making groups. |
| Roy Gordon  Robert Wilson | Project Mentor | * Provide guidance and advisory for any impediments, issues, and improvements. |
| Ivelin Tchangalov | Project Manager  Scrum Master | * Provide project guidance for the project team. * Foster communication and collaboration within the team. * Ensure project remains on schedule and on budget and bring to attention any risks/issues that may impact the team. * Contributes to documentation and implementation (project plan, SRS, etc.). |
| KC Harden | Product Owner | * Manage and prioritize the project backlog. * Serve as a liaison between product and development. * Evaluate project progress. * Contributes to documentation and implementation (project plan, SRS, etc.). |
| Hang Wang | Software Engineer Senior | * Leads development of coding and deployment. * Researches and proposes architecture. * Provides technical leadership. Supports Junior Engineers. * Contributes to documentation and implementation (project plan, SRS, etc.). |
| Nancy Lay | Test Manager | * Develop test strategy and test implementation plan. * Similar responsibilities as the Software Engineer Senior. |
| Alex armel Wabo tebu | Business Analyst | * Manages requirements. * Serves as a liaison to the Product Owner. * Contributes to documentation and implementation (project plan, SRS, etc.). |
| Kelvin Huynh  Melika Shahani  Jean Pita Diomi Kazadi  Jah-wilson Teeba  Ronald Milligan  Ian Fischer | Software Engineer Junior | * Contributes to coding, deployment, and testing. * Contributes to documentation and implementation (project plan, SRS, etc.). |

**Table 6.1 - Resources and Responsibilities**

# Communications Management

The communication management plan for Team B anticipates frequent communication among all stakeholders. The plan also identifies specific communication channels to ensure effective project coordination, with the goal of maintaining a clear and consistent flow of information throughout the duration of the project. Additionally, the plan establishes protocols for addressing and resolving any communication-related issues that may arise, to minimize disruptions and maintain momentum on the project.

## Microsoft Teams

MS Teams environment will be utilized as the primary communication channel to achieve effective and centralized information exchange among the team members. The different communication channels should be used according to their purpose and project-related tags should be utilized to quickly connect with the relevant team members.

**MS Teams Channels:**

* *General:* For all communication related to the project between the development tram
* *Joint Collaboration:* For all communication related to the project between members of Team A and Team B.
* *Team A:* Acts as the communication and working directory for Team A.
* *Team B:* Acts as the communication and working directory for Team B.

## GitHub

GitHub will serve as the main tool for the team to coordinate their work on issue tracking and application development. Team members will keep track of backlogs, assign tasks to themselves based on the priority of current milestones, and update the status of tasks to communicate progress. This approach will enhance the team's ability to coordinate their work and give each member a clearer picture of the project's progress.

## Taiga

Taiga is an open-source project management platform that allows teams to collaborate on software development projects. It is designed to be simple and easy to use, and it provides a variety of features such as task management, issue tracking, backlog management, and sprint planning. Taiga is based on the agile development methodology, and it supports Scrum, Kanban, and other agile methodologies.

The team will use Taiga to effectively manage project progress by assigning tasks to each developer, tracking specific development issues and monitoring the overall progress of the project. Taiga will provide a central location for the team to keep track of project milestones, deadlines, and deliverables. It will also provide a way for team members to communicate and collaborate on tasks, as well as to document and track the progress of each task. Furthermore, Taiga will allow the team to use agile methodologies such as Scrum and Kanban to organize and prioritize their work and make adjustments as needed to ensure the project stays on track.

# Risk Management

All projects s no matter the size or complexity are vulnerable to risks, but the importance is how to mitigate them for the outcome to be successful. In our case we will analyse some of the risk that could occur during all the step of our project. The framework presented below is a risk management process.

Diagram

Description automatically generated with medium confidence

**Figure 8.1 – Risk Management Process**

* Risk identification: early in the process in other to arrange it and prioritize.
* Risk assessment: to located and find the appropriate solution.
* Risk control: develop and appropriate response to minimize his effect on the project.
* Risk review control: to make sure that the solution is efficient and some case if there is no way to eradicate at least mitigate the risk and make sure that the control is effective.

## Risk Analysis

All project risks will be identified and documented. The risk management process will be utilized to identify risks on your project. Industry checklists will be utilized as well, in order to help offset inexperience in the risk identification process or help make certain all the risks have been identified. As with all the other project documents, we will file the list of risks in the project repository.

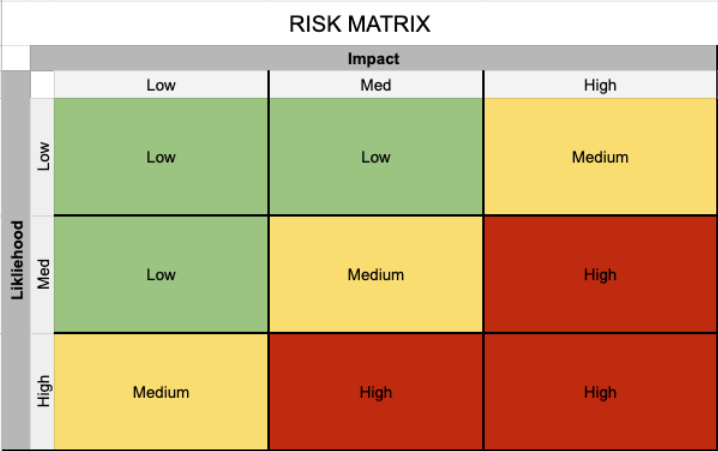
## Issue Management

Most of the issues that we have in Team B are:

* Trying to understand the viewpoint of the shareholder.
* Trying to establish a proper communication with the shareholder.
* Understanding the large and complex system requirements.
* Proper meeting time is not easy to set.

## Risk Matrix and Register

A risk matrix is used to identify the severity of impact of risks that can be encountered throughout the project. By identifying risks early and creating a mitigation plan, we can prioritize high severity risks that pose the greatest threat to delivering a successful project within budget and on time.



**Figure 8.3 - Risk Matrix**

The project team will identify, assess, control, and monitor risks throughout the project lifecycle and the risk register will be continuously updated. Risks identified are captured in the risk register below.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Date Raised** | **Risk Description** | **Likelihood** | **Impact** | **Severity** | **Owner** | **Mitigation Plan** |
| 1 | 01/28/2023 | Selected tools & libraries do not integrate well together. | Med | High | High | Project Manager | PM will do early evaluation of potential solutions to ensure it doesn’t block us. |
| 2 | 01/28/2023 | Text extraction from images don’t work well on panoramic images. | Low | Med | Low | Project Manager | Engineering team will test solutions early on to identify what works best. |
| 3 | 01/28/2023 | Inability to find ideal tour location to test/demonstrate applications full capabilities. | Low | Med | Low | Project Manager | Use a less appropriate location. |
| 4 | 01/28/2023 | Team is not well-balanced (resource constraints). | Med | High | High | Project Manager | Balance resources with Team A. |
| 5 | 02/08/2023 | Image Analysis solution – Algorithm does not output accurate data | Med | High | High | Software Lead | Pivot to hybrid custom/Matterport option defined in TDD Appendix C |
| 6 | 02/10/2023 | Image processing speed does not meet performance requirement | Med | Med | Med | Software Lead | Implement GPU |

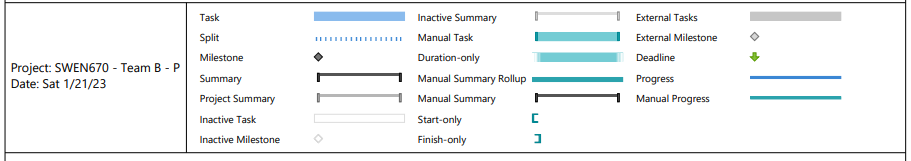
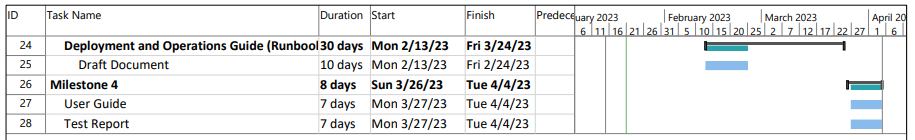
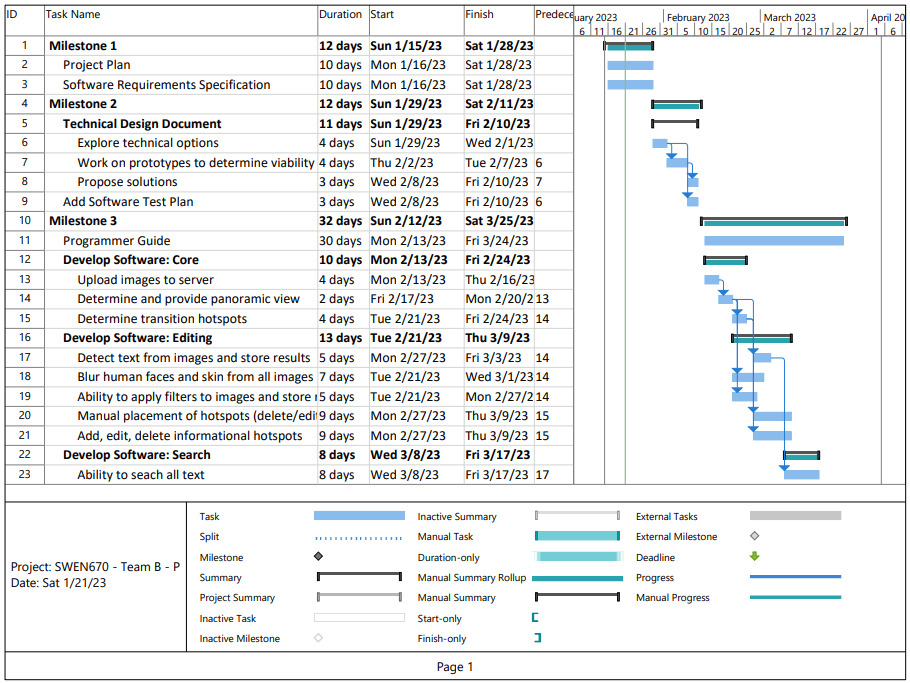
**Table 8.3 - Risk Register**

Appendix A – Acronyms and Definitions

| Term | Meaning |
| --- | --- |
| PMP | Project Management Plan |
| BA | Business Analysts |
| SAAS | Software-As-A-Service |
| GUI | Graphical user interface |
| WBS | Work Breakdown Structure |
| VR | Virtual Reality |
| CR | Code Review |
| GPU | Graphics Processing Unit |

**Table 9 – Acronyms and Definitions**

Appendix B – Detailed Project Plan

**Figure 10 - Project Schedule**

Appendix D – References

*Build and release an Android app*. (n.d.). Retrieved January 23, 2023, from <https://docs.flutter.dev/deployment/android>