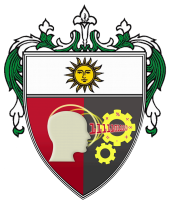

University of Santo Tomas

College of Information and Computing Sciences

España, Manila

Software Engineering I

Software Project Management Plan
for
Datu Shop Inventory Management System

Client: Datu Shop

Prepared By:
Abrigo, Nathanael Chris O.
Daliuag, Ronan Manuel D.
Dela Paz, Angelo Daniel A.
Entrata, Joshua Kyle K.
Lorenzo, Lex Zedrick M.
Padrejuan, Shaun Kristoffer C.
Salvador, Fabian III R.
Sebastian, Nigel Haim N.

3CSC

Submitted To
Asst. Prof. Janette E. Sideño

Table of Contents

1. Introduction

1.1. Project Overview

This project aims to design and develop an inventory management system for Datu Shop. Datu Shop is a Micro-Small-Medium Enterprise (MSME) located in Quezon City, focusing on procuring and selling various motorcycle parts and servicing said parts to their customers. The software system will have to perform the following functions to assist the company in solving its current issues:

1. Handle the inventory of the motorcycle supplies supplied to and from the shop for easy access and recording.
2. Display a dashboard with analytics regarding shop sales, high-selling products, and low-stock products.

Included in this document are several essential aspects to be considered and addressed during the development period of the software system, which include the following: critical deadlines of the development process, the structure of the organization in charge of the development, and their responsibilities, the resources to be used during the development process, and the deliverables to be submitted to the client.

1.2. Project Deliverables

In fulfillment of the project, the following requirements and deliverables in the tables below will be completed and submitted to the client and professor in charge of the group.

Table 1.1 Documentation Table

Deliverable	Date	Description
Software Project Management Plan (SPMP)		Provides critical information related to the project, including project overview, process model used, organizational responsibilities, risk management strategies, and project schedule.
Software Requirements		Describe the system's requirements,

Specifications (SRS)	**First Semester**	scope, and features to be delivered to the client.
Software Design Document (SDD)		Illustrates the software system's overall design, including its architecture, scope, and features.
Software Test Plan (STP)	**Second Semester**	Lists the cases in which the developed system will be tested and the method in which the cases will be tested.
Software Test Document		Includes a derivation of the STP but includes the test cases used.

Table 1.2 Software Deliverables Table

Deliverable	Date	Description
Datu Shop Inventory Management System Prototype	**First Semester**	Includes a working prototype of the system to see the various parts of the system to be developed in action.
Datu Shop Inventory Management System Working Version One	**Second Semester**	Includes the complete and fully functional version of the software system, which will be delivered to the client for use.

2. Project Organization

2.1. Process Model

The process model decided by the developers to be used for this project is the Agile Development Methodology. The Agile Development Methodology is a collection of beliefs and concepts that strongly emphasize self-organizing teams, collaboration, and adaptation to change. It is the understanding that the same approaches and the capacity to adapt to change and uncertainty will only sometimes work. The Agile Methodology has five phases during each iteration: Requirements Gathering, Planning and Designing, Development, Testing, and Evaluation.

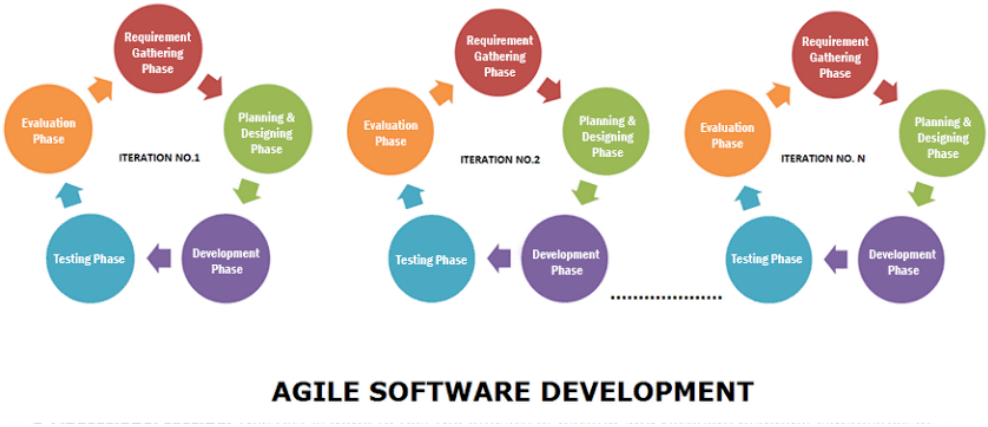

Figure 2.1: Agile Development Methodology Model

Listed below are the phases of each iteration of the Agile Development Methodology life cycle:

1. **Requirements Gathering**
    - This phase focuses on gathering the requirements needed to complete the project. The problems to be addressed and possible solutions will be identified during this phase. This also includes vital information regarding development, such as the project's scope, features, and timetable.
2. **Planning and Designing**
    - Upon identifying the significant requirements of the project, the overall design of the project will be tackled in this phase. This phase of the project includes the conception of the Software Design Document (SDD), which illustrates the

planned flow of the software itself. The SDD will consist of the system architecture, mockups of the software itself, and use cases of the system to be developed.

3. **Development**
    - The development phase is the most extended phase for each iteration during the Agile Software Development life cycle. This part of the cycle includes the actual development of the software using the design created in the planning and design phase. All aspects of the software, such as the front-end and back-end parts of the system will be developed in this phase.
4. **Testing**
    - This cycle phase involves testing the developed system for bugs and possible sources of errors. The quality assurance officers will spearhead this phase to ensure the code is clean, creating test cases for the system for possible bugs. The development team will use the Software Test Plan as a reference during this phase of the iteration.
5. **Evaluation**
    - The last phase of each iteration focuses on identifying possible room for system improvement. The development team and the client will examine for bugs in the system, possible new features, and quality-of-life changes to be implemented in the next cycle iteration.

There are numerous advantages to applying the Agile Development Methodology in this project. First, the priority of this approach is to fulfill the client's needs by promptly and consistently delivering high-quality outputs, which means that despite delays in development, this approach welcomes changes for the customer’s competitive advantage. Second, this approach continually shows the project's results, not just its final output. This allows the customer to see the project and see if changes must be made. Third, based on the time intervals set, the team can reflect on becoming more effective and adapting accordingly.

The disadvantages of applying the Agile Development Methodology in this project include the following: First, the project can be less predictable due to the changes or demands that may happen between the phases. Second, this approach requires the time and commitment of both the team and the customer to communicate to ensure the project's quality throughout the development.

2.2. Organizational Structure

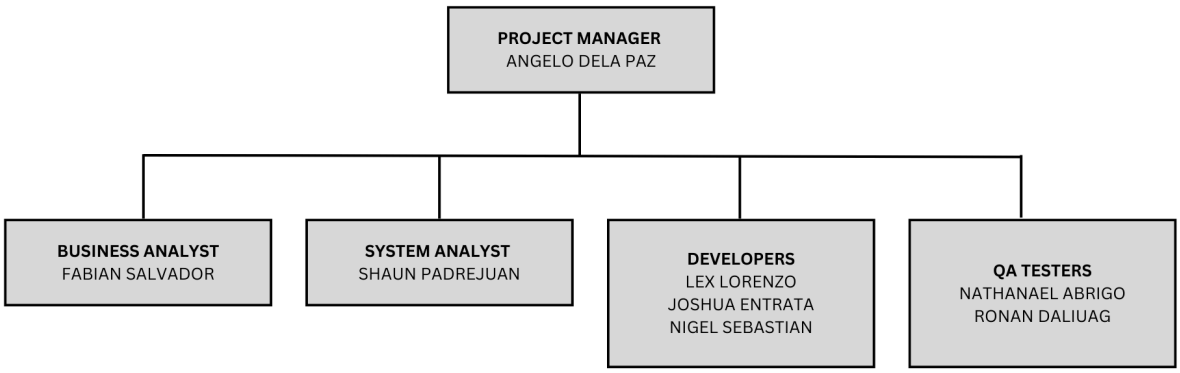

Figure 2.2 Tigre Tech Organizational Chart

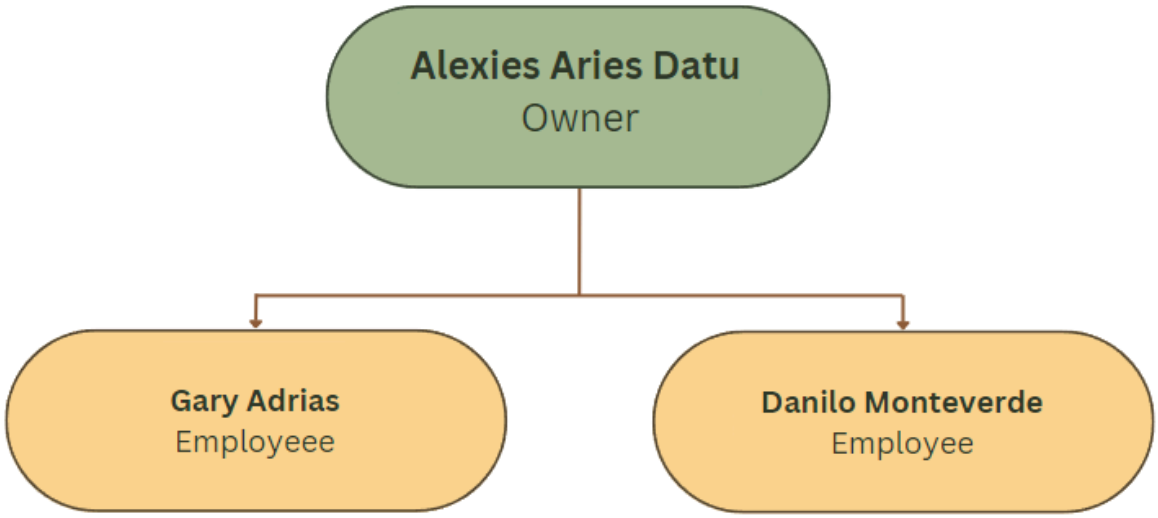

Figure 2.3 Datu Shop Organizational Chart

2.3. Project Responsibilities

Table 2.1. Project Responsibilities

Role	Name	Responsibilities	Deliverables
Project Manager	Angelo Dela Paz	Spearheads the creation of the whole project. Manages and oversees task allocation, project scheduling, and client communication (along with the business analyst)	timetable of the Project and Documentation (Software Project Management Plan, Validation Board).
Business Analyst	Fabian Salvador	Leads the communication efforts with stakeholders, being the most well-versed in business jargon and processes. Responsible for laying the groundwork and translating the requirements stated by said stakeholders for easier understanding of the system required.	Transcript of Meetings, Client-related Letters, Documentation (Software Requirements Specifications)
Systems Analyst	Shaun Kristoffer Padrejuan	Analyzes the business requirements and leads the system	UML Diagrams, System Design, and Model Documentation

		design efforts. Identifies the hardware and software requirements and supersedes the PM in overseeing the actual implementation of the system	(Software Design Document)
Software Developers	Joshua Kyle Entrata Lez Zedrick Lorenzo Nigel Haim Sebastian	Develops the system following the design model of the Systems Analyst.	System Prototypes, Documentation
Quality Assurance Officers	Nathanael Chris Abrigo Ronan Manuel Daliuag	Manages the quality control of the outputs presented by the team. Inspects the system for issues and/or errors and resolves said issues	System's Error and Validation Reports, and Documentation (Software Test Plan)

3. Managerial Process

3.1. Management Objectives and Priorities

The project's primary objective is to create an inventory management system for Datu Shop that automates several critical processes related to the company's business practices. This primary objective may be fulfilled by developing a system with the following functions:

- An inventory management system will be used by the employees of the motor shop to track the flow of product stocks.
- The inventory system will notify users of any low product stock.
- The inventory system will include a dashboard for the admins to see current sales analytics using various graphs.

To further visualize the project's scope, shown below is a validation board that contains the problems discussed with the client and the proposed solutions that will be implemented through the development of the proposed system.

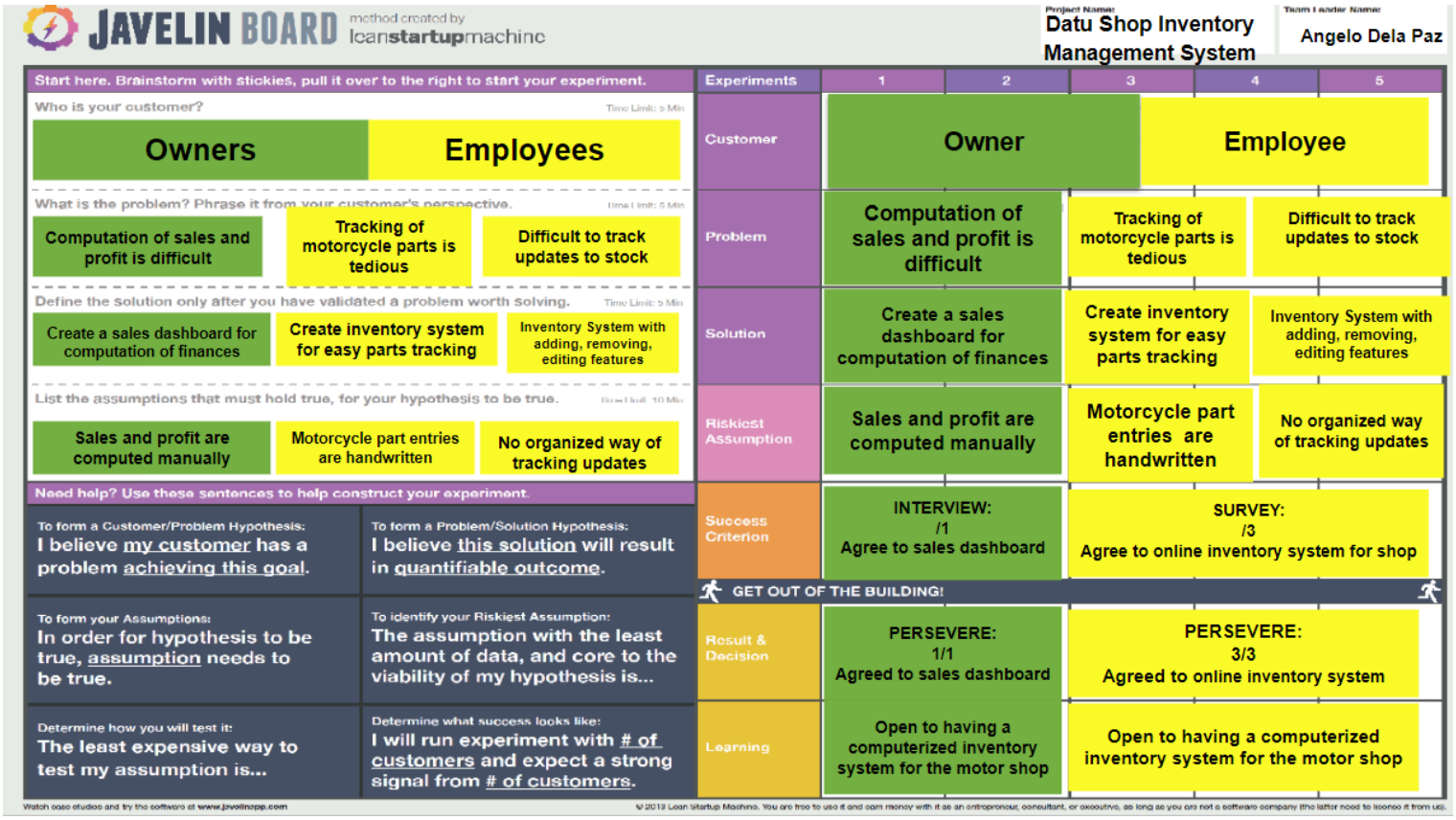

Figure 3.1 Validation Board

3.2. Assumptions, Dependencies, and Constraints

This document section provides the assumptions, dependencies, and constraints the development team will follow throughout the duration of the development.

3.2.1. Assumptions

The project has the following assumptions before continuing the project:

- The employees and admins of Datu Shop will use the system.
- Orders can be placed for inventory items that are not in stock.

3.2.2. Dependencies

The system will depend significantly on the availability of an internet connection within the shop. The system will be entirely online; therefore, the absence of the Internet would mean the users of the system would be forced to use other means. In the event of this, the users would opt to return to handwritten means in the meantime.

3.2.3. Constraints

The project constraints are the following:

- The system to be developed is not fully digital; therefore, clients would still need to go to the actual shop for any services.
- The logging of new inventory stock is not automatic.

3.3. Risk Management

The risk management section aims to identify the possible risks of project development. Such risks are mostly interrelated to the complexity and potential impact on the project development. Below is the table that shows the assessment and the significance of the possible problems that may arise during the project's development.

Failure Possibility Legends:

High

- The possibility is likely to occur. The developers need to prioritize finding a solution.

Medium

- The possibility could occur during the development. Attention is necessary to resolve the problem.

Low

- Failure in the development is not likely to happen. The implementation will be smooth to focus on other requirements.

Project Impact Legends:

High

- Failure to satisfy the requirements of the client

Medium

- Can be decided in the future whether it is still required due to constraints.

Low

- When failure occurs, it will not significantly affect. The requirement can be excluded at times due to constraints.

Table 3.1. Risk Management Table

Problem	Failure Possibility	Project Impact	Mitigation Technique
Difference of Schedule Availability	Medium	High	Meetings will be scheduled ahead of time and when most members and the client are available.
Absence of the Owner	Medium	High	Due to the nature of the owner's job, the owner may have to go to various places and be absent for some periods. The mitigation technique used to solve this problem is to have a proxy for the owner who can make decisions other than the owner.
Skill Deficiencies Among the Development Team	Medium	High	Since the development team are college students, they will learn and study the current technology trends. Merging their learnings from other sources and learning from the curriculum implemented by the university.
Server Integration	High	High	Integrating the server requires

			payment (monthly or annually) and maintenance. The developers will implement an affordable and secure server for the database and hosting.
Manual to Digital Integration	Medium	High	The team will understand the ins and outs of the process used by the client's staff and ensure that it is found in the inventory management system.
Staff Training	Low	Medium	The team will teach the staff and the owner the system's capabilities.

3.4. Monitoring and Controlling Mechanisms

The project will be monitored through Notion to keep track of progress, tasks, and assigned members. Each change in the program will be observed through the project's documentation. Upon the start of the development period, Git and Github will be used by the programmers for version control and could act as an online repository for collaborative work. Both front-end and back-end developers will push their source codes to the repository, pull, and review them before merging each branch to the origin before testing. Revisions will be recorded on the project's documentation and in the repository during testing.

Communication between the members of the group is through messenger. The members' concerns will be discussed through the group chat created. Members of the group are also updated on tasks completed and tracking deadlines. A separate group chat will be created for the members to communicate with the client. Further group inquiries on the business details are asked through the group chat. Concerning the privacy and information this project includes, all inquiries, messages, and information within the group are confidential and archived upon completion.

The repository of documents and relevant files regarding the initial designs and diagrams of the project is stored in a Google Drive Folder and Figma Project, respectively. All necessary

drafts, such as the SPMP, SRS, and SDD, are stored in Google Drive, the web application mockups and relevant diagrams are found in the Figma Project, and the actual source code used for the project will be stored in the Github repository.

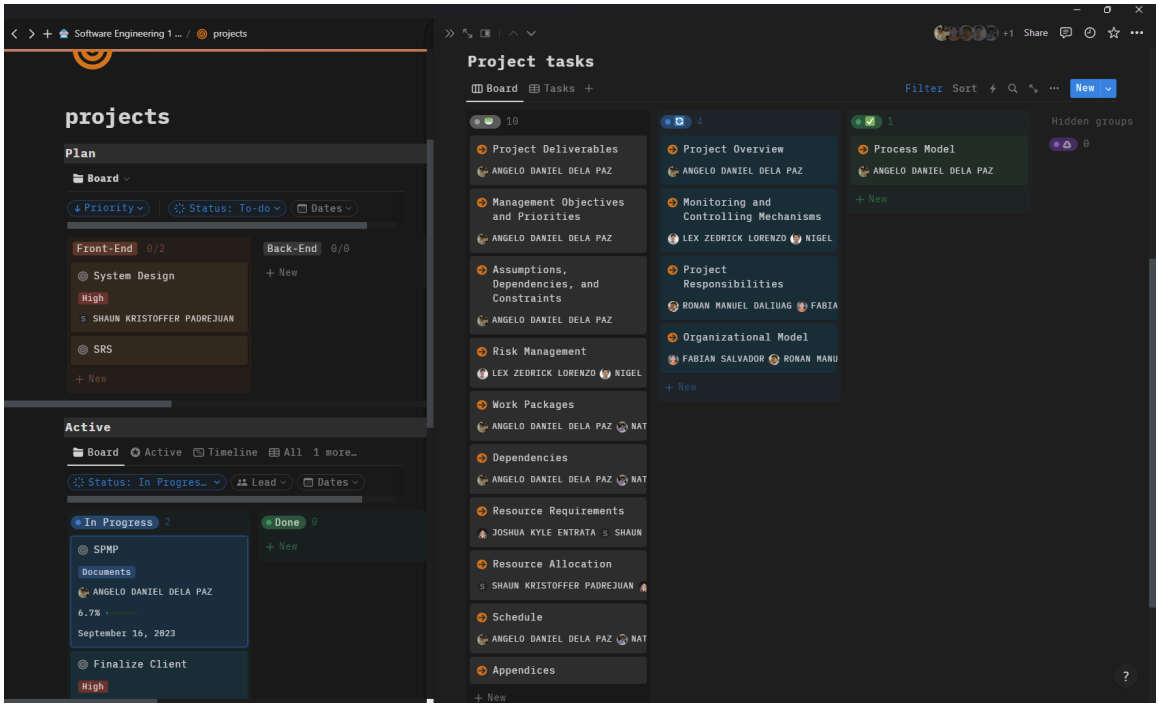

Figure 4.1 Notion Board for Projects

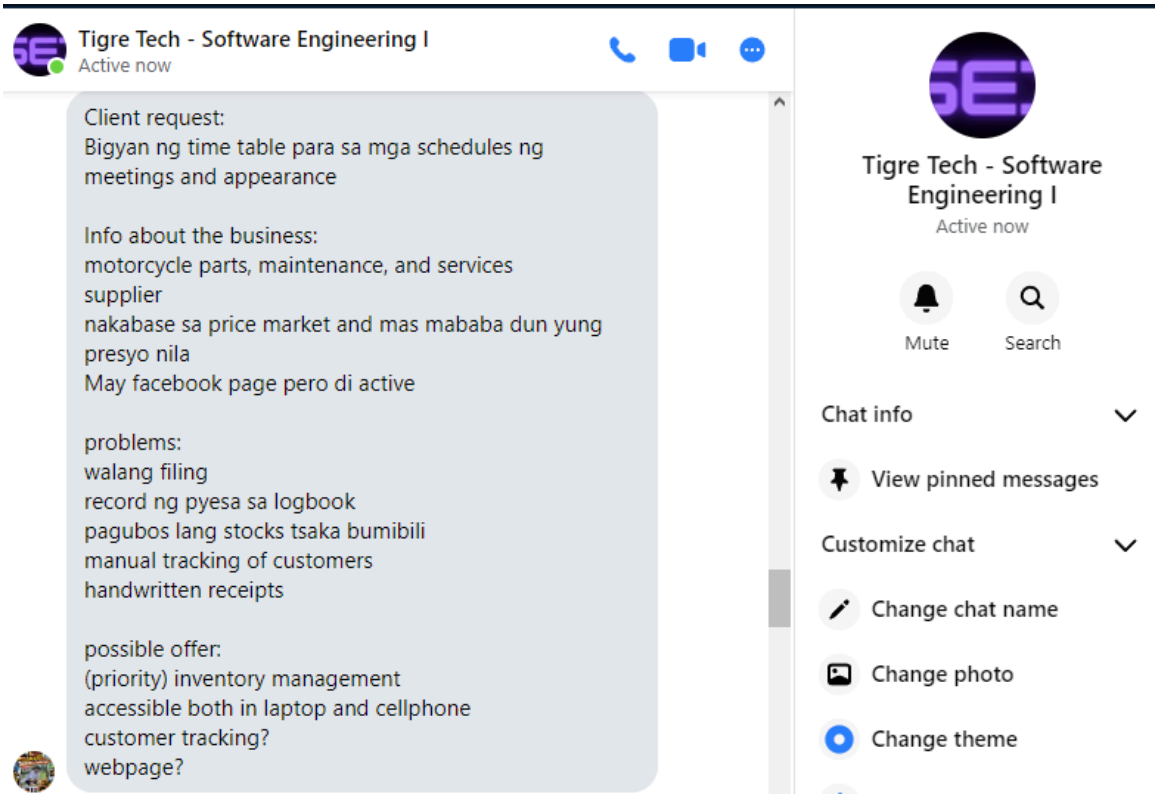

Figure 4.2 Facebook Messenger Group Chat

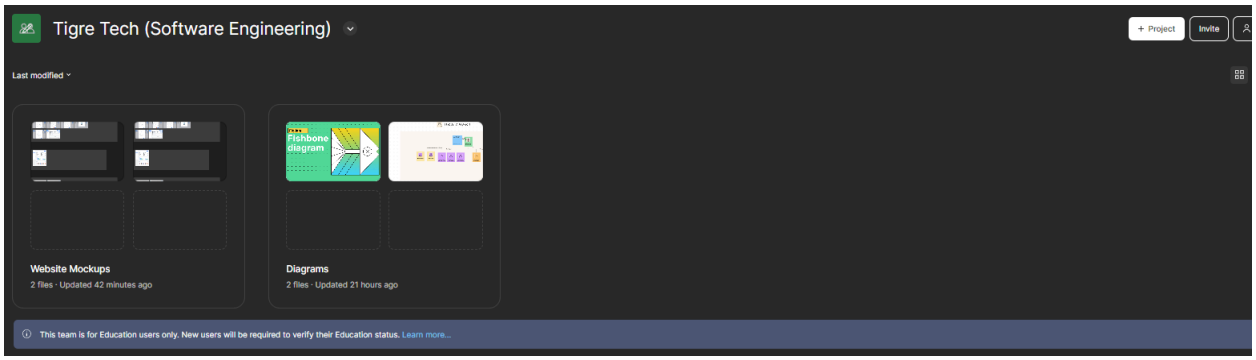

Figure 4.3 Figma Project Files

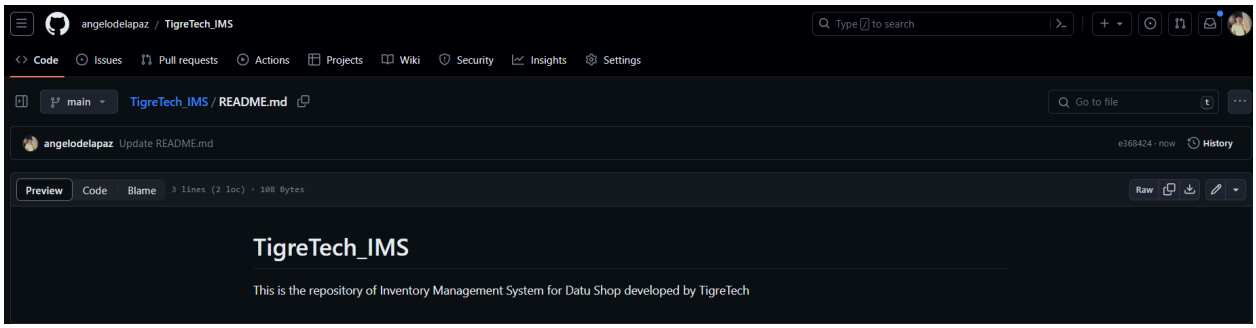

Figure 4.4 Github Repository

4. Work Packages, Schedule, and Budget

4.1. Work Packages

Found in the table below are the significant tasks delegated to the developers throughout the duration of the conception and development of the project. Each task is given a certain number of days to be completed for a systematic and organized workflow between the developers and the client.

Table 4.1 Work Packages Table

Activity	Deliverables
Preliminaries	
Creation of Letters for the Client	Proposal Letter
	Certification of Non-Relation
	Interview Letter
Interview	Transcript
Discussion of Possible Solutions and Approach	
Development Team Meeting	Initial Validation Board
Meeting with Client	Validation Board with Experiments

Data Gathering for the new action plan of javelin board	Complete Validation Board
Aesthetics, Designs, and Functionality	
Discussion of Use-Cases	Use-Case Diagram and Use-Case Descriptions
Initial SRS	Initial Software Requirement Specifications Document
Discussion of Design with the client	Transcript
Creation of Prototype	Prototype System
Presentation	Validated Prototype System
Project Defense	Proposal Defense
Revised Documentation	Software Project Management Plan
	Software Design Document
	Software Requirement Specifications
	Software Test Plan
Implementation and integration of systems	
Front - End Design Implementation	Front-end of the System
Back - End Design Implementation	Back-end of the System
System Testing	Successful Test Cases
Project Defense	Final Defense
Final Revisions	Final Documentation
Project Deployment	System Installation
	Maintenance Training

4.2. Dependencies

Table 4.2 Project Dependency Table

Letter	Activity	Prerequisites
A	Finding a Client	-
B	Proposal Letter	A
C	Certificate of non-relation	
D	Interview Letter	
E	Interview with the Client	B C D
F	Validation Board	E
G	Software Project Management Plan	F
H	Use-Case Diagram	G
I	Use-Case Description	H
J	SRS	I
K	SDD	J
L	Project Defense	G, J, K
M	Revised Documentation	L
N	Start of Front-End Development	M
O	Start of Back-End Development	M
P	System Testing	N, O
Q	Final Defense	P

R	Final Documentation Submission	Q
S	System Installation/Training	N, O, P

4.3. Resource Requirements

The proposed Datu Shop Inventory Management System will have the following requirements. The developers will need a machine to code the website properly, develop the data analytics, and make a coherent database process to ensure orders will be given and received by the business's necessary people. A web domain is needed to host the website, along with a database for the orders and the data analytics. Regarding software, the developer will need IDEs and SDKs to develop the deliverables.

Table 4.3 Resource Requirements Table

RESOURCE REQUIREMENTS			
Type of Resource	**Resource Details/Specs**	**Source**	**Assumptions**
Software	Python Django	Internet	Expertise in this backend framework
Software	React	Internet	Expertise in this frontend library
Software	PostgreSQL	Internet	Expertise in database administration
Software	Figma	Internet	Expertise in using this tool to design and create prototypes

Hardware	Personal Computers or Laptops	Group Members	Members have their own devices that is capable of developing and testing the system
Software/Hardware	Internet	Internet Service Providers	Everyone must have an internet access
Person	Development Team (*Project Manager, System Analyst, Business Analyst, Web Developers, Quality Assurance Testers*)	Group Members	Capable of fulfilling their responsibilities and tasks
Person	Motoshop Mechanics	Client	Trained to use Inventory Management System
Person	Owner	Client	Involved in the development and implementation of the system

4.4. Resource Allocation

Table 4.4 below shows the breakdown of how the developers are assigned to a specific activity in the project's development. Some activities and documentation are managed by one lead, while the other developers may help complete the document.

Table 4.4 Resource Allocation Table

Activity	Project Manager	Systems Analyst	Business Analyst	Project Developers	Quality Assurance Officers
Brainstorming	✔	✔	✔	✔	✔
Client Hunting	✔	✔	✔	✔	✔
Meeting with the Client	✔		✔		
Letter for Approval	✔	✔	✔	✔	✔
Requirements Gathering	✔		✔		
Software Project Management Plan Document	✔	✔	✔	✔	✔
Software Requirements Specifications			✔		
Software Design Document		✔			
Designing and Prototyping	✔			✔	

Software Test Plan	✔				✔
Coding				✔	
Testing					✔
Installation Manuals		✔		✔	

LEGEND

✔ = Needed

4.5. Schedule

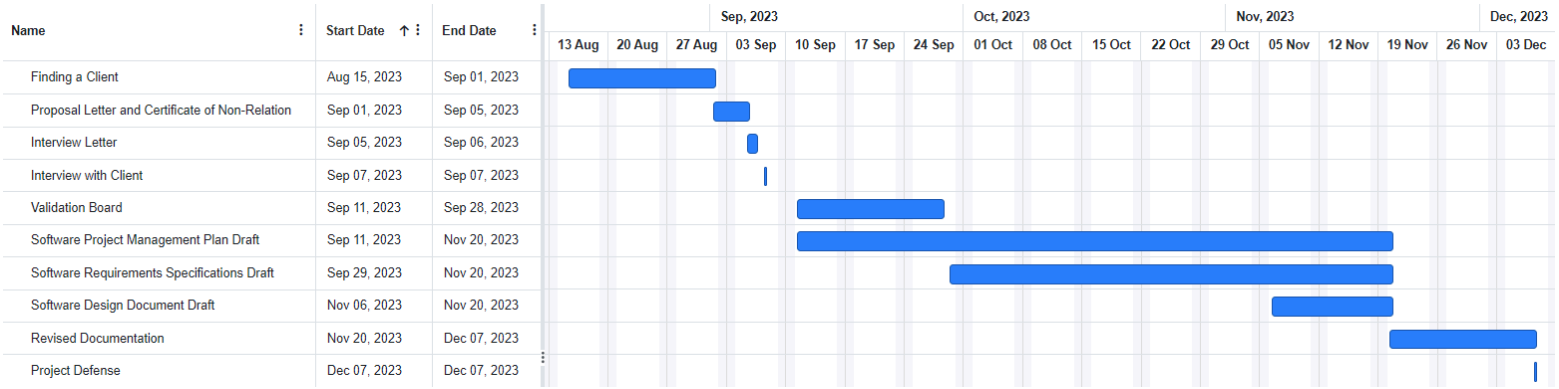

Figure 4.2.1 Gantt Chart for Software Engineering I

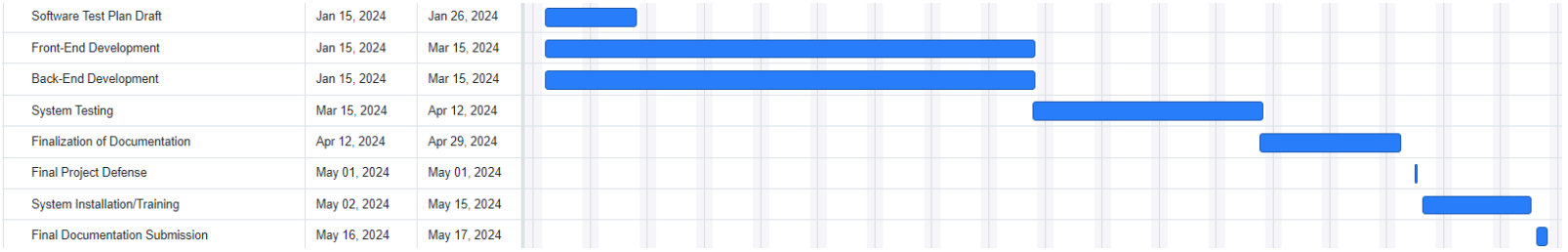

Figure 4.2.2 Gantt Chart for Software Engineering II

5. Additional Components

5.1. Appendices

Table 5.1 Development Cost Breakdown

Developer:	
Project Manager *(190/200hrs)*	38,000
Business Analyst *(150PHP/200hrs)*	30,000
System Analyst (160PHP/200hrs)	32,000
Project Developer *(170PHP/180hrs/3 members)*	91,800
Quality Assurance Developer *(170PHP/180hrs/2 members)*	61,200
	253,000
Hardware/Software Cost:	
Tablet	20,000
Server	20,000
	40,000
Transportation	**2,500**
Miscellaneous	**3,000**
Total:	**P298,500**

Table 5.2. Payback Analysis Table

Cash Flow Description	Year 0	Year 1	Year 2	Year 3	Year 4	Year 5
Development Cost	-298,500					

Operation & Maintenance Cost		-55,000	-40,000	-35,000	-40,000	-45,000
Present Value	1	0.893	0.797	0.712	0.636	0.567
Time-Adjusted Costs	-298,500	-49,107.14	-31,887.76	-24,912.31	-25,420.72	-25,534.21
Cumulative Time-Adjusted Cost Over	-298,500	-249,392.86	-217,505.10	-192,592.7	-167,172.07	-141,637.86
Benefits	0	131,250	141,093.75	151,675.78	163,051.46	175,280.32
Time-Adjusted Benefits	0	117,187.50	112,479.07	107,959.83	103622.15	99,458.76
Cumulative Benefits	0	117,187.50	229,666.57	337,626.4	441,248.55	540,707.32
Net Cost+Benefits	-298,500	-132,205.36	12,161.47	145,033.61	274,076.48	499,069.45

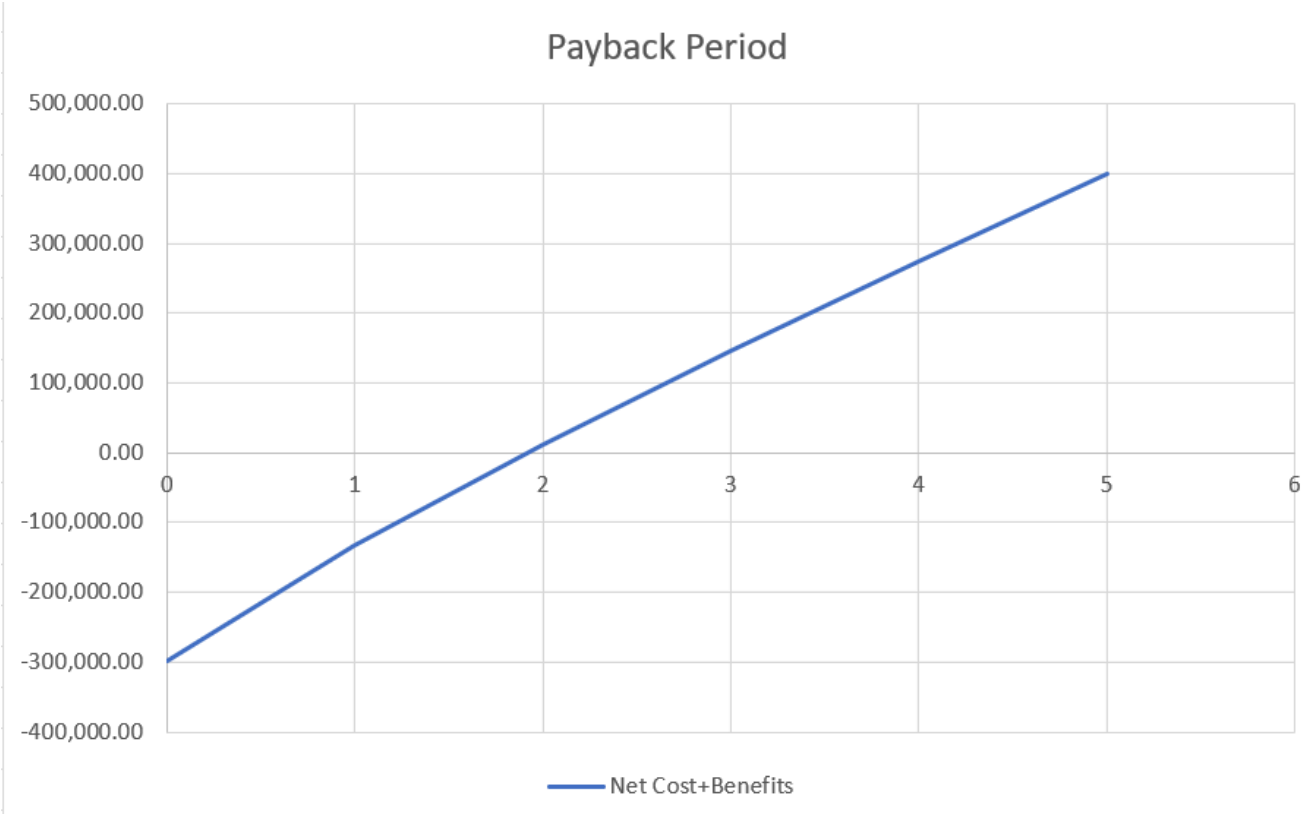

Figure 5.1. Payback Analysis Graph