FPT University
TRƯỜNG ĐẠI HỌC FPT

CAPSTONE PROJECT REPORT

Report 2 – Project Management Plan

Eatery Management Systems

G11	
Group members	Lê Hà Nam HE 153319 Chu Tuấn Dũng HE 150178 Nguyễn Phương Linh HE 163269 Hà Tiến Dũng HE 151003 Chu Hồng Long HE 153093 Chu Văn Luân HE 151322
Supervisor	Nguyễn Thị Hải Năng
Capstone project code	RMS

– Hanoi, January 2024 –

Table of Contents

I. Record of Changes

Date	A* M, D	In charge	Change Description
12/01/2024	A	Le Ha Nam	Create report Project Management Plan
13/012024	M	Le Ha Nam	Overview and Management approach
15/01//2024	M	Le Ha Nam	Project deliverables and Responsibility assignments
16/01/2024	M	Le Ha Nam	Project communications and Configuration management

*A - Added M - Modified D - Deleted

II. Project Management Plan

1. Overview

1.1 Scope & Estimation

#	WBS Item	Complexity	Est. Effort (man-days)
1	**Specification**		**73**
1	Project Planning	Medium	3
2	Deliver Report 2 - Project Management Plan	Simple	20
3	Deliver Report 3 - System Requirement Specification	Complex	20
4	Deliver Report 5 - Test Documentation	Complex	10
5	Deliver Report 6 - Software User Guides	Medium	10
6	Deliver Report 7 - Final Project Report	Medium	10
2	**Design**		**30**
1	Design User Interfaces (UI) prototypes	Medium	10
2	Deliver Report 4 - Software Design Document	Complex	20
3	**Training**		**14**
1	Learn backend,frontend, and framework	Medium	14
4	**Implement code**		**341**
4.1	**Common Functions**		**11**
1	Login	Medium	6
2	Logout	Simple	2
3	Change password	Simple	3
4.2	**Eatery Owner**		**114**
1	View dashboard	Complex	15
2	View list table	Simple	5
3	Create table	Medium	6
4	Edit table	Medium	6
5	Delete table	Simple	2
6	View list category	Simple	3
7	View category detail	Medium	5
8	Create category	Medium	6

9	Edit category	Medium	5
10	Delete category	Simple	3
11	Search category	Simple	2
13	Change status category	Medium	4
14	View list product	Medium	6
15	View product details	Medium	6
16	Create product	Medium	6
17	Edit product	Medium	5
18	Delete product	Simple	3
19	Search product	Simple	2
20	Change status product	Medium	4
21	View list bill	Medium	4
22	View detail bill	Medium	5
23	View transaction	Medium	6
24	View detail transaction	Medium	5
4.3	**Waiter**		**76**
1	View list table	Medium	5
2	Filter table	Medium	6
3	Create order	Medium	20
4	View menu	Simple	3
5	Filter dish	Simple	3
6	Search dish	Simple	1
7	View cart	Medium	5
8	View list table	Medium	5
9	Edit order	Medium	10
10	View waiting Dish	Simple	3
11	View delivery Dish	Simple	3
12	View waiting dish list	Simple	2
13	View serve dish list	Simple	3
14	View finish dish list	Simple	2

15	Confirm bill	Medium	5
4.4	**Chef**		**16**
1	View list order dish	Medium	5
2	View list finish dish	Medium	5
3	Update status order dish	Medium	6
4.5	**Cashier**		**28**
1	View list bill	Simple	3
2	View detail bill	Medium	4
3	Filter bill	Medium	4
4	Export Invoice	Medium	10
5	View transaction	Medium	4
6	View detail transaction	Simple	3
4.6	**Manager**		**96**
1	View list user	Simple	3
2	View detail user	Medium	4
3	Create user	Medium	5
4	Delete user	Simple	3
5	Edit user	Medium	5
6	View list category	Simple	3
7	Create category	Medium	5
8	Edit category	Medium	4
9	Delete category	Simple	2
10	Search category	Simple	2
11	Change status category	Medium	4
12	View list product	Simple	3
13	View product details	Medium	4
14	Create product	Medium	6
15	Edit product	Medium	6
16	Delete product	Medium	5
17	Search product	Simple	2

18	Change status product	Medium	5
19	View list bill	Simple	3
20	View detail bill	Medium	4
21	View Detail Transactions	Medium	4
22	View list table	Simple	3
23	Create table	Medium	5
24	Edit table	Medium	4
25	Delete table	Simple	2
4.7	**Customer**		**36**
1	View menu	Simple	3
2	Search dish	Simple	2
3	Filter menu	Medium	4
4	Order dish	Medium	10
5	Edit order	Medium	10
6	View cart	Simple	3
7	View history order	Medium	4

Total Estimated Effort (man-days)   **458**

One man-days   **5h**

Allocated Effort (man-days):458(6 member * 5 workday/week * 15 weeks).

1.2 Project Objectives

1.2.1 Quality

#	Testing Stage	No. of Defects	Notes
1	Reviewing	60	
2	Unit Test	20	Identify and fix bugs or issues early in the development process
3	Integration Test	60	Ensuring that different parts of a software system work seamlessly together
4	System Test	10	Ensure the overall quality, reliability, and performance of the software system before it is deployed for actual use.
5	Acceptance Test	0	Confirm that the software meets the specified requirements and is suitable for production use.

1.2.2 Project Objectives
- Project will be completed on time
- Complete with allocated effort (458 man-days)

1.2.3 Team Objectives

- The team finishes the project before April 26th, 2024.
- All team members will follow the task assigned.
- Team members improve their skills working in a group.
- Each member improves both technical skills and soft skills.
- Team successfully applied a software development process.

1.3 Project Risks

#	Risk Description	Impact	Possibility	Response Plans
1	Lack of skill & technology knowledge.	Medium	Medium	Training members quickly to keep up with the project progress
2	Misinterpreting the requirement leads to the project going in the wrong direction	High	Medium	Work together with the requirement team immediately and check back to make sure the requirement is correct
3	Members do not attend meetings or come in late	Medium	Medium	Record the meeting minutes in text and hand over the work to another member, inform about meeting information to the member, and provide rules and penalties for violations
4	Hard to understand business requirements	High	High	Learn more about how a restaurant management systems Find ways to solve possible problems from the website as well as customers
5	Extended time to define functional requirements.	Hight	Medium	Focus on fixing bugs from the previous phase and will continue to develop new functionality once the requirements are clearly defined.

2. Management Approach

2.1 Project Process

We have chosen to develop this project by using the Iterative and Incremental model. This system will be developed and continuously in the circle model
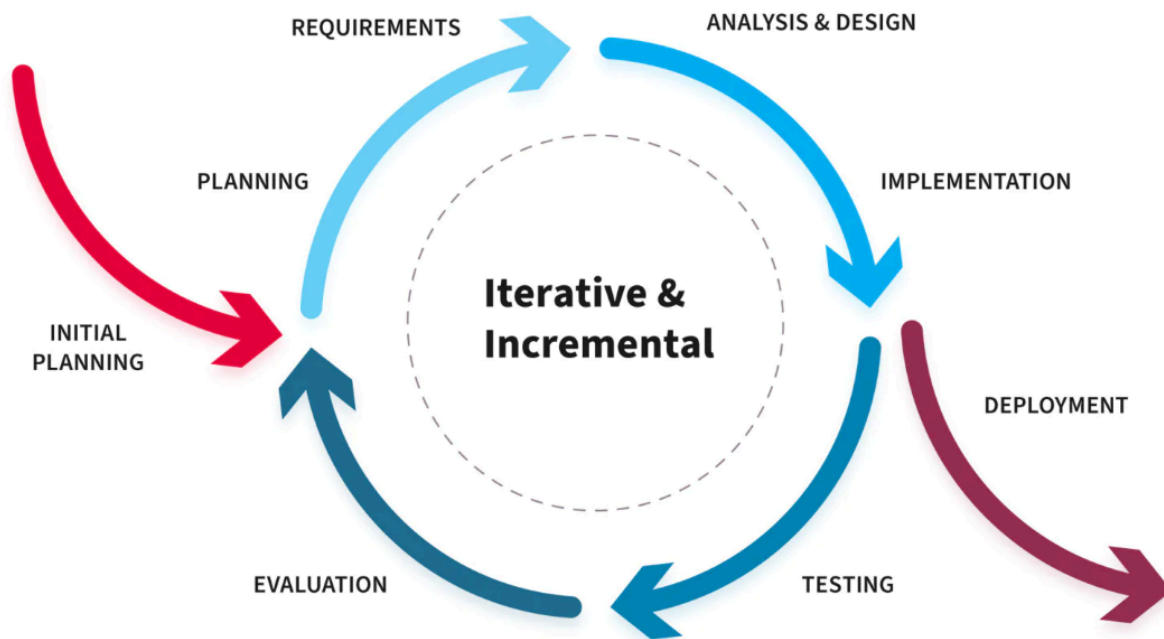
Figure 2.1. Iterative & Incremental Models

We selected this approach due to the reasons listed below:

    – This application has many features for users and may take a while to develop. By using this approach, we can prioritise and deliver the most important features first, followed by the remaining ones.

    – By testing and reviewing functions early on, any necessary fixes or upgrades can be made, resulting in an improvement in the overall quality of the system.

    – By receiving feedback after each increment of the application, we can prevent unexpected outcomes at the end of the development

    – Accommodating easily for any changes in requirements.

Apply Iterative & Incremental Model to project, we provided to 3 loops as follows:

- Iteration 1:
    - Requirement collect & analyse
    - Design and analysis project architecture
    - Implementation
    - Test execution: Unit Test and Integration Test
    - Evaluation
- Iteration 2:
    - Requirement analyse

- ● Design and analysis project architecture

- ● Implementation

- ● Test execution: Unit Test and Integration Test

- ● Evaluation

- Iteration 3:
  - ● Requirement analyse

  - ● Design and analysis project architecture

  - ● Implementation

  - ● Test execution: Unit Test, Integration Test and System Test

  - ● Evaluation

2.2 Quality Management

2.2.1 Defect Prevention
- ● Preventive Measures: Implement preventive measures at the early stages of the project. This includes clear requirements gathering, thorough analysis, and detailed design specifications.
- ● Continuous Improvement: Adopt a continuous improvement mindset, where lessons learned from past projects are used to improve processes.
- ● Training and Skill Development: Regular training sessions for the team to ensure everyone is up-to-date with the latest best practices and technologies.
- ● Coding Standards: Enforce coding standards to ensure consistency and reduce errors.

2.2.2 Reviewing
- ● Peer Reviews: Conduct peer reviews of code, designs, and documentation. This helps in identifying potential issues early.
- ● Walkthroughs and Inspections: Regularly schedule walkthroughs and inspections of both the project's progress and its various components.
- ● Client Feedback: Regularly involve the client or stakeholders in reviews to ensure the project meets their expectations and requirements.

2.2.3 Unit Testing
- ● Develop unit test suites to test the smallest components of the software, such as functions, modules or classes, and ensure that they work as expected.
- ● Use unit testing tools to automate the execution of test suites and identify errors automatically.

2.2.4 Integration Testing
- ● Create integration test scenarios to test the performance of software components when combined and integrated together.
- ● Testing the interaction and data exchange between different components to ensure system compatibility and integrity.

- ● Check if components do not conflict or cause errors when combined together.

2.2.5 System Testing

- Testing the functionality and operation of a software system to ensure that it functions properly and meets the stated requirements.
- Evaluates the performance of a software system, including load, response time and resource consumption, to ensure that it operates effectively.
- Testing the security of a software system to ensure that it meets security requirements and is free of security vulnerabilities.

2.3 Training Plan

Training Area	Participants	When, Duration	Waiver Criteria
Java Spring Boot	All members	09/01/2024	Mandatory
MySQL	All members	12/01/2024	Mandatory
Github,Slack	All members	14/1/2024	Mandatory
Angular	Chu Tuấn Dũng + Lê Hà Nam + Chu Văn Luân	16/1/2024	Mandatory
Postman	All members	19/1/2024	Mandatory

3. Project Deliverables

| #	Deliverable	Due Date	Notes
1	Project Plan	6/01/2024	Make a project goal plan.
2	Report 1 - Project Introduction Document	11/01/2024	Project introduction.
3	Report 2 - Project Management Plan Document	16/01/2024	Project management plan.
4	Report 3 - SRS Document	6/03/2024	Software requirement specification.
5	Report 4 - Software Design Document	29/03/2024	Software design document
6	Report 5 - Software Testing Document	16/04/2024	Test case document, test documentation
7	Report 6 - Release Package and User Guides	15/04/2024	Software user guides
8	Report 7 - Appendix	16/04/2024	Final project final
9	Final package	17/04/2024	Final Codes & documents, User manual

4. Responsibility Assignments
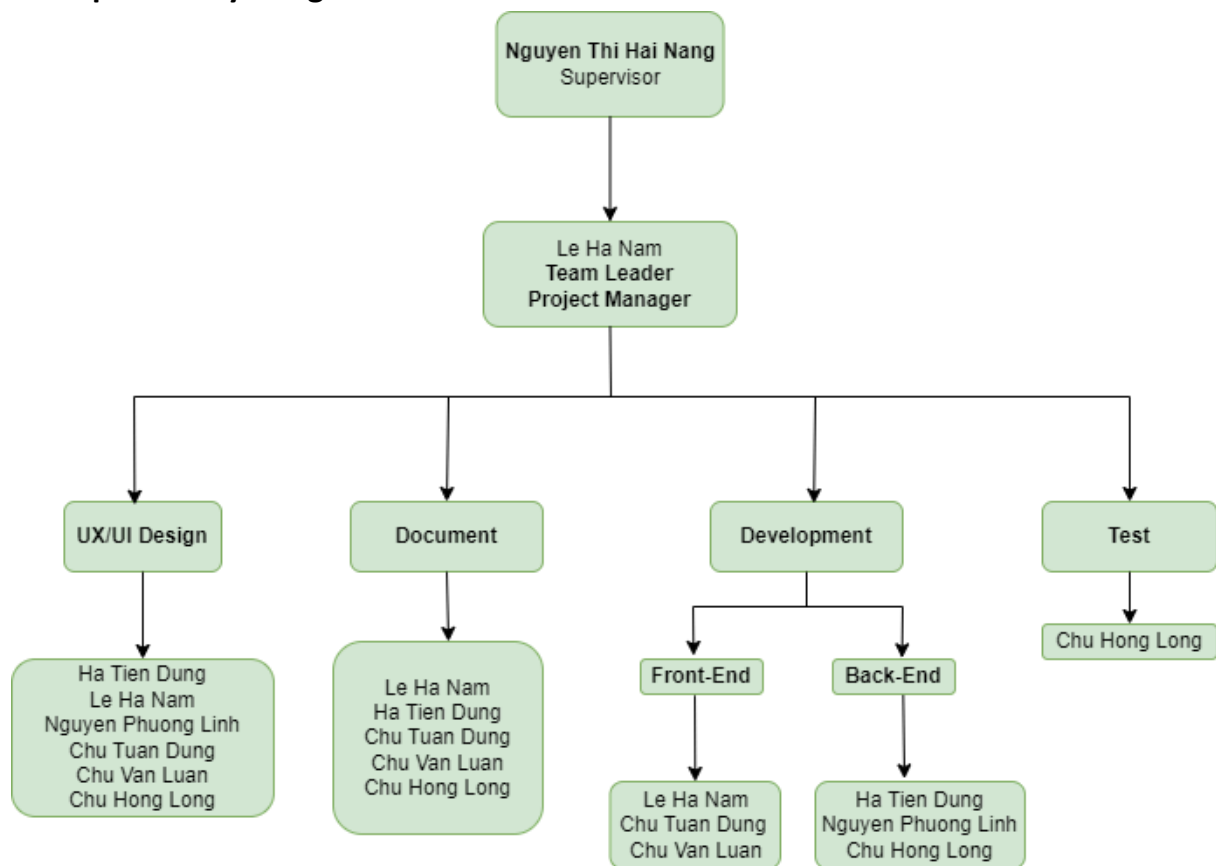
Figure 4.1. Team & Structures Diagram

D~Do; R~Review; S~Support; I~Informed; <blank>- Omitted

Responsibility	Project Manager	Business Analyst	Designer	Developer	Tester
Start Project	D	I	I	I	I
Prepare Project Introduction Document	R	D	D	D	D
Project Planning & Tracking	D	D	S	S	S
Prepare SRS Document	R	D	S	S	S
Prepare Software Design Document	R	D	S	S	S
Implement Features	D	D	D	D	D
Testing	I	S	S	S	D
Maintenance	S	S	D	D	S

Prepare Users Guides	R	D	D	D	D
Prepare Thesis Presentation	D	D	D	D	D
Prepare Final Project	D	D	D	D	D

5. Project Communications

Communication Item	Who/ Target	Purpose	When, Frequency	Type, Tool, Method(s)
Weekly Meeting With Supervisor	All team members and Supervisor	Review members' work achievements. Report the project's progress and the status of each member's work.	Once a week	Offline, Online (Google Meet)
Team daily meeting	All team members	Report the project's progress and the status of each member's work each day.	Everyday	Slack,Messenger, Google Meet
Team daily working	All team members	Work and communicate issues.	Everyday	Slack,Messenger, Google Meet,Zalo
Unscheduled Meeting	All team members	Discuss and solve the urgent problem	When there are urgent problems	Google Meet

6. Configuration Management

6.1 Document Management

6.1.1 Documentation tools
- Google Docs: Report documents, User guides.
- Google Sheets: QA documents, Weekly reports, Project Tracking.
- Microsoft Office (Word, Powerpoint)

6.1.2 Document Conventions
- Each report has code starting with SEP490_G11
- Font family : Calibri

6.1.3 Document Version
- All documents will be stored on Google Drive and Microsoft OneDrive.
- All reports will be delivered to the supervisor through Drive.

After the meeting between all members and supervisor describing requirements, all documents will be created and submitted to management tools to store and to keep track of changes. Leader then creates tasks through Slack. The member will then edit documents if necessary then the Leader will collect documents to submit to the supervisor.

6.2 Source Code Management
- Source code will be stored and maintained by members on Github.

- Each function in the function list will be coded in a separate branch.
- Each commit to the branch will have to follow specific rules.
- A code review will be executed before merging code from function branch to main branch.
- Technical leader is the person who will review and merge code of team members

6.3 Tools & Infrastructures

Category	Tools / Infrastructure
Technology	Java/SpringBoot (BackEnd),Angular(FrontEnd)
Database	MySQL
IDEs/Editors	Visual Studio Code, Intellij IDEA
Diagramming	StarUML, DrawIO
Documentation	Ms Office, Google Docs/Sheets/Slides
Version Control	GitHub (Source Codes), Google Drive (Documents)
Deployment server	Amazon Web Service (AWS)
Communication tools	Facebook - Messenger, Google Meet, Zalo,Slack
Project management	Word,Excel