**Capstone Project**

**Document Template**

Note: The following are the candidate sections of the document. They are presented here for guidance. Questions in each section could be used as possible aspects to cover. Some questions may not be applied to each project. On the other hand, additional information may be needed.

# Introduction

## Purpose

* What is the problem or the opportunity that the project is investigating?
* Why is this problem valuable to address?
* What is the current state (e.g. unsatisfied users, lost revenue)?
* What is the desired state?
* Has this problem been addressed by other projects? What were the outcomes?

## Industry/ domain

* What is the industry/ domain?
* What is the current state of this industry? (e.g. challenges from startups)
* What is the overall industry value-chain?
* What are the key concepts in the industry?
* Is the project relevant to other industries?

## Stakeholders

* Who are the stakeholders? (be as specific as possible as to who would have access to the software)
* Why do they care about this software?
* What are the stakeholders’ expectations?

# Product Description

## Architecture Diagram

Include a diagram of the building blocks of the design including users and how they interact with the product.

## User Stories

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| # | User Story Title | User Story Description | Priority | Additional Notes |
| 1 |  |  |  |  |
| 2 |  |  |  |  |
| … |  |  |  |  |

## User Flow

Present as a flow diagram the steps a user may make in interacting with the software.

## Wireframe Design

Show elements of the user interface, either manually or via a tool such as Figma.

## Open Questions/Out of Scope

* What features are considered out of scope?

## Non-functional Requirements

* What are the key security requirements? (e.g. login, storage of personal details, inactivity timeout, data encryption)
* How many transactions should be enabled at peak time?
* How easy to use does the software need to be?
* How quickly should the application respond to user requests?
* How reliable must the application be? (e.g. mean time between failures)
* Does the software conform to any technical standards to ease maintainability?

# Project Planning

Include a Gantt chart or screenshot of a Trello board showing key milestones (with dates) to complete the project.

# Testing Strategy

* What were steps undertaken to achieve product quality?
* How was each feature of the application tested?
* How did you handle edge cases?

# Implementation

* What were the considerations for deploying the software?

# End-to-end solution

* How well did the software meet its objectives?

# References

* Where is the code used in the project? (link to GitHub)
* What are the resources used in the project? (libraries, APIs, databases, tools, etc)