

**Capstone Project Report**

**Report 2 – Project Management Plan**

– Ho Chi Minh, September 2025 –

**Table of Contents**

[I. Record of Changes 3](#_Toc83330363)

[II. Project Management Plan 4](#_Toc83330364)

[1. Overview 4](#_Toc83330365)

[1.1 Scope & Estimation 4](#_Toc83330366)

[1.2 Project Objectives 4](#_Toc83330367)

[1.3 Project Risks 4](#_Toc83330368)

[2. Management Approach 5](#_Toc83330369)

[2.1 Project Process 5](#_Toc83330370)

[2.2 Quality Management 5](#_Toc83330371)

[2.3 Training Plan 5](#_Toc83330372)

[3. Project Deliverables 5](#_Toc83330373)

[4. Responsibility Assignments 5](#_Toc83330374)

[5. Project Communications 6](#_Toc83330375)

[6. Configuration Management 6](#_Toc83330376)

[6.1 Document Management 6](#_Toc83330377)

[6.2 Source Code Management 6](#_Toc83330378)

[6.3 Tools & Infrastructures 6](#_Toc83330379)

# I. Record of Changes

|  |  |  |  |
| --- | --- | --- | --- |
| Date | A\* M, D | In charge | Change Description |
| 20/09/2025 | A\* | Minh Anh | Initial document creation |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

\*A - Added M - Modified D - Deleted

# II. Project Management Plan

## 1. Overview

### 1.1 Scope & Estimation

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **WBS Item** | **Complexity** | **Est. Effort**  **(man-days)** |
| ***1*** | ***Onboarding & Academic Data*** | ***Medium*** | ***5*** |
| ***2*** | ***AI Curriculum & Career Alignment*** | ***Complex*** | ***8*** |
| ***3*** | ***Skills & Stats Foundation*** | ***Complex*** | ***8*** |
| ***4*** | ***Dashboard, Skill Tree & Arsenal*** | ***Complex*** | ***7*** |
| ***5*** | ***Boss Fight & Leaderboards*** | ***Complex*** | ***9*** |
| ***6*** | ***Dynamic Quest & Notifications*** | ***Medium*** | ***5*** |
| ***7*** | ***Browser Extension Integration (FTPU & Arsenal)*** | ***Medium*** | ***3*** |
| ***8*** | ***Code Battle & Competitive Events*** | ***Complex*** | ***10*** |
| ***9*** | ***Event Management & Administrative Platform*** | ***Medium*** | ***5*** |
| ***10*** | ***Academic Integration (FPTU, Quest Memory & Recovery)*** | ***Complex*** | ***7*** |
| ***11*** | ***Admin Owned Educational Governance*** | ***Medium*** | ***5*** |
| ***12*** | ***Objective System, Knowledge Graph & Rewards*** | ***Complex*** | ***7*** |
| ***13*** | ***CI/CD Pipeline & Infrastructure*** | ***Medium*** | ***3*** |
| ***14*** | ***Observability & Monitoring*** | ***Simple*** | ***2*** |
| ***Total Estimated Effort (man-days)*** | | | ***84*** |

### 1.2 Project Objectives

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Testing Stage** | | **Test Coverage** | **No. of Defects** | **% of Defect** | **Notes** | | |
| 1 | Reviewing | | N/A | 24 | 20% | PR-based code reviews, checklists, static analysis | | |
| 2 | Unit Test | | ≥ 70% lines/branches | 42 | 35% | .NET unit tests + React component tests | | |
| 3 | Integration Test | | Key paths ≥ 60% | 30 | 25% | API + DB + services; Postman suites | | |
| 4 | System Test | | ≥ 80% critical flows | 18 | 15% | End-to-end user journeys in staging | | |
| 5 | Acceptance Test | | 100% of AC covered | 6 | 5% | UAT against PRD acceptance criteria | | |
| ***Allocated Effort (man-days)*** | | | | | ***84*** |

### 1.3 Project Risks

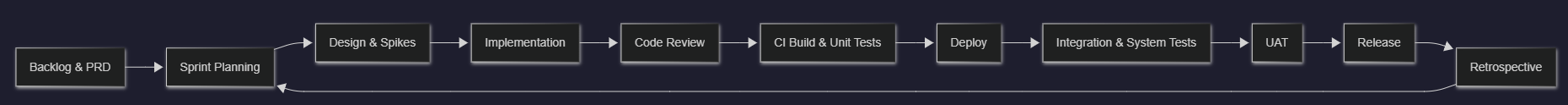
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Risk Description** | **Impact** | **Possibility** | **Response Plans** |
| 1 | Requirements churn or unclear acceptance criteria | High | Medium | Maintain PRD traceability; change control board; refine AC before sprint commit |
| 2 | Skill gaps in ASP.NET Core/React | High | Medium | Targeted training plan; pair programming; code review; spikes |
| 3 | Integration dependencies (OAuth, email) delayed | Medium | Medium | Mock services; feature flags; decouple integration behind interfaces |
| 4 | Performance/scalability issues in code battle module | High | Medium | Early load testing; profiling; optimize algorithms; caching |
| 5 | Data model misalignment with curriculum import | Medium | Medium | Schema review; sample imports; migration scripts |
| 6 | Schedule risk due to scope creep | High | Medium | Strict MVP scopes; backlog grooming; time boxing; communicate trade-offs |
| 7 | Team availability constraints | Medium | Low | Cross-training; buffer; reassign tasks quickly |

## 2. Management Approach

We will use an agile, sprint-based approach (Scrum-like) with one-week sprints, continuous integration, and incremental releases, Planning emphasizes MVP scope, measurable milestones, and quality gates.

* Iterations: 10 sprints (1 week each)
* Parallel workstreams: Core Player, Competitive & Events, Governance & Academic Integration
* Backlog management: PRD-derived stories prioritized by value and risk
* Definition of Ready: Clear AC, dependencies identified, designs prepared
* Definition of Done: Code + tests + review + CI green + docs updated
* Governance: Weekly steering meeting: change control for scope changes

### 2.1 Project Process

**

Key practices:

* Branching: trunk-based with short-lived feature branches; PR required
* Reviews: mandatory reviewers; linting and static analysis in CI
* Automated checks: build, unit tests, integration tests, coverage thresholds
* Releases: tagged, semantic versioning; release notes linked to PRD stories

### 2.2 Quality Management

Approach to achieve quality objectives:

* Defect Prevention: design reviews, coding standards, static analysis, secure defaults
* Reviewing: PR templates, checklists, mandatory code review for all changes
* Unit Testing: .NET xUnit, React Testing Library, minimum coverage ≥ 70%
* Integration Testing: API + DB tests using in-memory or test containers; Postman collections
* System Testing: end-to-end flows; cross-browser checks; performance smoke tests
* Test Data: deterministic fixtures and seeded data

### 2.3 Training Plan

*[You need to plan the training activities in case any of your team member lack of knowledge/skills to handle the project works]*

|  |  |  |  |
| --- | --- | --- | --- |
| Training Area | Participants | When, Duration | Waiver Criteria |
| Java Spring Boot |  |  | Mandatory |
| Git, Github |  |  | Mandatory |

## 3. Project Deliverables

*[Given the main project deliverables. Those can be internal and/or external deliverables. Students can prepare master schedule like the table format as below or in the more detailed structure as the sample in the attached sample file -* ***Report2\_Sample High Level Project Schedule.pdf****]*

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Deliverable** | **Due Date** | **Notes** |
| 1 | … | dd/MM/yyyy | … |

## 4. Responsibility Assignments

*[Describe the main responsibilities in your project (to complete the outputs as defined in the above section), in the format as the sample below]*

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

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Responsibility** | **KhanhNTHE03579** | **VanNTTHE04680** | **PhuongDTHE03246** | **HuyenDTHE04671** | **ThuyVT04278** |
| Project Planning & Tracking | S | D | R | R | R |
| Prepare Project Introduction Document | S | S | D | R | I |
| Prepare SRS Document (Overview Part) | R | D | S | S | R |
| Prepare SRS Document (User Requirements) | D | R | S | S | R |
| … |  |  |  |  |  |

## 5. Project Communications

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Communication Item** | **Who/ Target** | **Purpose** | **When, Frequency** | **Type, Tool, Method(s)** |
| Daily Standup | Development Team, PM | Discuss progress, blockers and plan for the day | Daily | Discord |
| Sprint Planning | Development Team, PM | Plan the work for the upcoming sprint | Every 2 weeks | Discord |
| Sprint Review | All Stakeholders | Demonstrate the work completed in the sprint and gather feedback | Every 2 weeks | Google Meets |
| Product Demo | All Stakeholders | Showcase new features and functionality | As needed | Google Meets, Offline |

## 6. Configuration Management

### 6.1 Document Management

*[Describe how you would manage project documents & their changes/versions]*

### 6.2 Source Code Management

*[Describe how you would manage project source codes & their changes/versions]*

### 6.3 Tools & Infrastructures

|  |  |
| --- | --- |
| **Category** | **Tools / Infrastructure** |
| **Technology** | Next.js 14+ (FrontEnd), .NET 9 (BackEnd) |
| **Database** | Supabase (PostgreSQL) |
| **IDEs/Editors** | Visual Studio Code, Visual Studio |
| **Diagramming** | DrawIO |
| **Documentation** | Ms Office, Google Docs/Sheets/Slides, Canva |
| **Version Control** | Github (Source Codes), Google Drive (Documents) |
| **Deployment server** | Google Cloud |
| **Project management** | Github (Repository, Projects) |