

International School

**Capstone Project 2**

CMU-SE 451

**Project Plan Document**

**Version 2.1**

**Date:** 27/02/2023

**Craft Village Pollution Monitor System**

**Submitted by**

**Ca, Van Cong Le**

**Huy, Bui Duc**

**Phuc, Hua Hoang**

**Trung, Nguyen Thanh**

**Nhan, Huynh Ba**

**Approved by**

**Ph.D. Nguyen Thanh Binh**

**Proposal Review Panel Representative:**

Name Signature Date

**Capstone Project 2 - Mentor:**



Name Signature Date

**PROJECT INFORMATION**

|  |  |  |  |
| --- | --- | --- | --- |
| **Project acronym** | CVPMS | | |
| **Project Title** | Craft Village Pollution Monitor System | | |
| **Start Date** | 01/03/2023 | **End Date** | 15/05/2023 |
| **Lead Institution** | International School, Duy Tan University | | |
| **Project Mentor** | Ph.D. Thanh Binh, Nguyen | | |
| **Scrum master / Project Leader & contact details** | Ca, Van Cong Le  *Email:* [cascabusiness@gmail.com](about:blank)  *Tel:* 0352707895 | | |
| **Partner Organization** |  | | |
| **Project Web URL** | [*https://github.com/Casca113s2/craft-village-pollution-monitor-system*](https://github.com/Casca113s2/craft-village-pollution-monitor-system) | | |
| **Team members** | **Name** | **Email** | **Tel** |
| 25211207666 | Ca, Van Cong Le | [cascabusiness@gmail.com](http://cascabusiness@gmail.com) | 0352707895 |
| 25211215894 | Huy, Bui Duc | [duchuyltt122@gmail.com](http://duchuyltt122@gmail.com) | 0818648090 |
| 25211204084 | Phuc, Hua Hoang | [phuchuho0402@gmail.com](http://phuchuho0402@gmail.com) | 0905639682 |
| 25211215133 | Trung, Nguyen Thanh | [nguyenttrung2601@gmail.com](http://nguyenttrung2601@gmail.com) | 0774496838 |
| 25211203702 | Nhan, Huynh Ba | [nhanhuynh1409@gmail.com](mailto:nhanhuynh1409@gmail.com) | 0935430785 |

DOCUMENT NAME

|  |  |  |  |
| --- | --- | --- | --- |
| **Document Title** | Project Plan Document | | |
| **Author(s)** | All team members | | |
| **Date** | 27/02/2023 | **File Name** | C2SE.01\_CVPMS\_Project-Plan-Document\_v2.1.docx |

REVISION HISTORY

| **Version** | **Date** | **Comments** | **Author** | **Approval** |
| --- | --- | --- | --- | --- |
| 2.0 | 25/02/2023 | Create documents | All team members |  |
| 2.1 | 27/02/2023 | Update after review | All team members |  |

**Approve Document:** Sign in to approve the document

|  |  |  |  |
| --- | --- | --- | --- |
| **Mentor** | Binh, Nguyen Thanh | Date | 27/02/2023 |
| Sign |  |
| **Scrum Master** | Ca, Van Cong Le | Date | 27/02/2023 |
| Sign |  |
| **Scrum Member** | Huy, Bui Duc | Date | 27/02/2023 |
| Sign |  |
| **Scrum Member** | Phuc, Hua Hoang | Date | 27/02/2023 |
| Sign |  |
| **Scrum Member** | Trung, Nguyen Thanh | Date | 27/02/2023 |
| Sign |  |
| **Scrum Member** | Nhan, Huynh Ba | Date | 27/02/2023 |
| Sign |  |

**Table Of Contents**

[1. Project Overview 7](#_Toc133137940)

[1.1. Project Description 7](#_Toc133137941)

[1.2. Scope and Purpose 7](#_Toc133137942)

[1.2.1. Scope 7](#_Toc133137943)

[1.2.2. Purpose 7](#_Toc133137944)

[1.3. Assumptions and Constraints 8](#_Toc133137945)

[1.4. Project Objectives 9](#_Toc133137946)

[1.4.1. Standard Objectives 9](#_Toc133137947)

[1.4.2. Specific Objectives 9](#_Toc133137948)

[1.5. Critical Dependencies 9](#_Toc133137949)

[1.6. Project Risk 10](#_Toc133137950)

[2. Project Development Approach 11](#_Toc133137951)

[2.1. Technical Process 11](#_Toc133137952)

[2.1.1. Reasons for selecting 11](#_Toc133137953)

[2.1.2. Agile Methodology 11](#_Toc133137954)

[2.1.3. Scrum Process 12](#_Toc133137955)

[2.2. Quality Management 13](#_Toc133137956)

[2.2.1. Estimates of Defects to be detected 13](#_Toc133137957)

[2.2.2. Quality Control 13](#_Toc133137958)

[2.2.3. Measurements Program 14](#_Toc133137959)

[2.3. Unit Testing Strategy 14](#_Toc133137960)

[2.4. Manual Testing Strategy 15](#_Toc133137961)

[3. Estimation 16](#_Toc133137962)

[3.1. Size 16](#_Toc133137963)

[3.2. Estimated Effort 18](#_Toc133137964)

[3.3. Schedule 19](#_Toc133137965)

[3.3.1. Work Breakdown Structure 19](#_Toc133137966)

[3.3.2. Detailed Schedule 20](#_Toc133137967)

[3.3.3. Project Schedule 25](#_Toc133137968)

[3.4. Resource 27](#_Toc133137969)

[3.5. Infrastructure 27](#_Toc133137970)

[3.6. Training Plan 28](#_Toc133137971)

[3.7. Budget for Project 28](#_Toc133137972)

[4. Project Organization 30](#_Toc133137973)

[4.1. Organization Structure 30](#_Toc133137974)

[4.2. Project Team 30](#_Toc133137975)

[5. Communication & Reporting 32](#_Toc133137976)

[6. Configuration Management 33](#_Toc133137977)

[7. Security Aspects 33](#_Toc133137978)

[8. References 34](#_Toc133137979)

**Table Of Tables**

[**Table 1.** Project Description 7](#_Toc122214094)

[**Table 2.** Assumptions and Constraints 8](#_Toc122214095)

[**Table 3.** Standard Objectives 9](#_Toc122214096)

[**Table 4.** Critical Dependencies 9](#_Toc122214097)

[**Table 5.** Project Risk 10](#_Toc122214098)

[**Table 6**. Pre-release review defects 13](#_Toc122214099)

[**Table 7.** Quality Control 13](#_Toc122214100)

[**Table 8.** Measurements Program 14](#_Toc122214101)

[**Table 9.** Software Scale Drivers 16](#_Toc122214102)

[**Table 10.** Software Cost Drivers 16](#_Toc122214103)

[**Table 11.** Acquisition Phase Distribution 17](#_Toc122214104)

[**Table 12.** Software Effort Distribution for RUP/MBASE 18](#_Toc122214105)

[**Table 13.** The estimated effort 18](#_Toc122214106)

[**Table 14.** Detailed Schedule 20](#_Toc122214107)

[**Table 15**. Project Schedule 25](#_Toc122214108)

[**Table 16**. Infrastructure 27](#_Toc122214109)

[**Table 17.** Training Plan 28](#_Toc122214110)

[**Table 18.** Total Cost Estimate 28](#_Toc122214111)

[**Table 19**. Cost Description 29](#_Toc122214112)

[**Table 20**. Estimate Budget 29](#_Toc122214113)

[**Table 21.** Project Team 30](#_Toc122214114)

[**Table 22.** Communication Methodology 32](#_Toc122214115)

[**Table 23.** Configuration Management 33](#_Toc122214116)

[**Table 24.** Acronym 34](#_Toc122214117)

**Table Of Figures**

[**Figure 1**. Principle and Different Stages 12](#_Toc122214135)

[**Figure 2.** Work Breakdown Structure 19](#_Toc122214136)

[**Figure 3.** Organization Structure 30](#_Toc122214137)

# Project Overview

## Project Description

**Table 1.** Project Description

|  |  |  |  |
| --- | --- | --- | --- |
| **Project code** | CVPMS | **Contract type** | Internal Project |
| **Customer** | Duy Tan University | **End-User** | Private Person  Household  Local authority  Admin |
| **Project Type** | Internal | **Project Manager/ Scrum master** | Van Cong Le Ca |
| **Project Category** | Development and Maintenance | **Business domain** | Environment |
| **Application type** | Mobile application, Web application |  |  |

## Scope and Purpose

### Scope

The system runs on mobile platforms and website platform, the windows operating system includes:

* Deploy on an Android/iOS device.
* Deploy on a PC or Laptop.
* During: 93 days.

### Purpose

The project name is “Craft Village Pollution Monitor System”. With the goal of building a software system to help automate the process of collecting and managing the pollution data from craft villages in Vietnam, in order to overcome the limitations and weaknesses of the current exist systems. The system is designed so that everyone from a wide range of ages can help to collect the pollution data from craft villages and other stake holders can use the system to monitor and manage the collected data. The process of collecting data will be more quick, efficient and required no deep understanding of environment science from the end user (private user to be more precisely). The project is developed in the form of a mobile application for collecting pollution data and a web application for data management, the work is handled automatically, saving effort and time for the users. Automate information storage and processing, and provide accurate and timely information at the request of the users. Synthesize, report statistics, get better results.

## Assumptions and Constraints

**Table 2.** Assumptions and Constraints

|  |  |  |
| --- | --- | --- |
| **No** | **Description** | **Note** |
| **Assumptions** | | |
| 1 | The mobile application can be run on both Android and iOS | Scope |
| 2 | The website can be run on Chrome, CocCoc, Edge | Scope |
| 3 | Customer reviewers will get seven days to approve a milestone document. If no comments are received within this time period, it will be considered as approved. | External Interfaces |
| **Constraints** | | |
| 1 | All module must be completed and delivered to customer before 10 – December – 2022 because customer has to demo to its end user after 12 – December | Schedule |
| 2 | The project shall conform to security requirements specified by the customer in the NDA | Security |
| 3 | Network is available | Environment |
| 4 | Flutter, Spring Boot, Bootstrap, Dart, Java, HTML/CSS, JavaScript, Python, SQL | Programming languages ​​and supporting libraries |
| 5 | Multi-Lable classification in machine learning, Location based services | Technology |

## Project Objectives

### Standard Objectives

**Table 3.** Standard Objectives

|  |  |  |  |
| --- | --- | --- | --- |
| **Metrics** | **Unit** | **Committed** | **Note** |
| Start Date | Date | 21/02/2023 |  |
| End Date | Date | 25/05/2023 |  |
| Duration | Date | 93 days |  |
| Team Size | Person | 5 peoples |  |
| Billable Effort | Person-day | 2$ \* 5 \* 5 \* 67 (For workday)  2$ \* 8 \* 5 \* 26 (For weekend)  (Working hours = Number of working hours per day \* Number of members \* Number of days.  Cost = Working hours \* The cost per member per hour = Working hours \* 2) |  |
| Number of work hours per day for one engineer | Person-hour | 5 (For workday)  8 (For weekend) |  |

### Specific Objectives

* + - * No defect
      * Done on time, completion of project early by December 10th
      * Apply new technology to the project
      * The system is easy to use and user friendly
      * Complete the functions of the system

## Critical Dependencies

**Table 4.** Critical Dependencies

| **No** | **Dependency** | **Expected delivery date** | **Note** |
| --- | --- | --- | --- |
| 1 | Craft Village Mobile Application | 25/05/2023 |  |
| 2 | Detection Pollution AI | 25/05/2023 |  |

## Project Risk

**Table 5.** Project Risk

| **Risk** | **Description** | **Probability** | **Impact** | **Mitigation Strategy** |
| --- | --- | --- | --- | --- |
| Incorrect requirements | Developing the product which does not accord with the requirements | 3 | 5 | Discuss and communicate frequently with Stakeholders |
| Estimate working time | Actual working time is not enough to finish a task compared to the estimated previous time | 2 | 4 | Review old tasks and evaluations to estimate for the new task. Re-plan for each sprint. |
| People | Team member who is ill, has health problems, or busy | 4 | 3 | Notify the scrum master (or ask a colleague to help you)  Complete the assigned tasks when possible |
| Lack of technical experiences | Detect harmful content in the video is a difficult technique that all members need to research and develop. | 4 | 4 | Spend a lot of time for learning and training |
| Team Communication | Team members can conflict with each other while discussing something related to the project | 4 | 2 | Conduct a meeting to share knowledge, experience and learning methods |
| External problems | It has power problems, laptop, personal computer, network system | 3 | 3 | Find another workplace (library, coffee shop, ...)  Notify the scrum master to assign appropriate tasks |
| Market | Other products are deployed at the same time and compete with the project team's product | 2 | 3 | Develop newer features and organize promotional activities |

# Project Development Approach

## Technical Process

### Reasons for selecting

To keep up with today's increasingly changing technology trends, we want a truly flexible and easy project development model to adapt to that change. Our project will develop more new features in the future. We will continuously update and apply new technologies that increase the attractiveness and intelligence of the application.

Currently, our team is a small team with little experience in project development. Therefore, we cannot avoid problems that arise in the software development stages and requirements can be changed to be more suitable. For the traditional model that requires managerial skills and high accuracy, it will not suit our team. Applying Agile Scrum model will help us to solve these problems, bring a lot of experience and best performance for project development.

### Agile Methodology

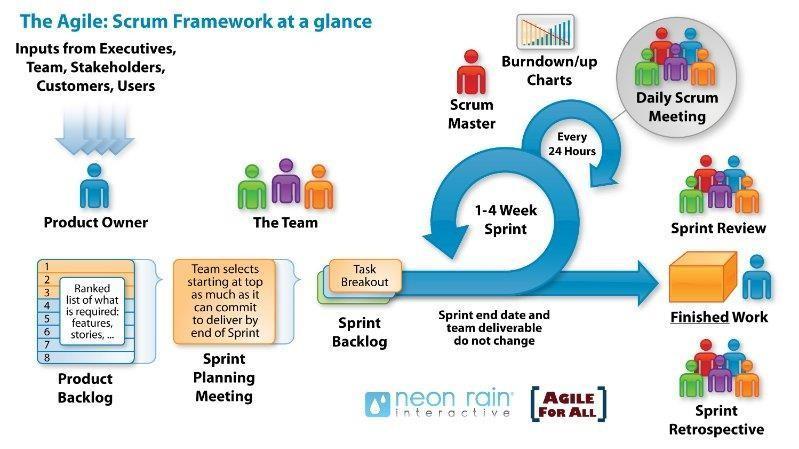
Agile software development refers to a group of software development methodologies based on iterative development, where requirements and solutions evolve through collaboration between self-organizing cross-functional teams.

Agile software development is more than frameworks such as Scrum, Extreme Programming, or Feature-Driven Development (FDD).

Agile software development is more than practices such as pair programming, test-driven development, stand-ups, planning sessions, and sprints.

Agile software development is an umbrella term for a set of frameworks and practices based on the values and principles expressed in the Manifesto for Agile Software Development and the 12 Principles behind it. When you approach software development in a particular manner, it’s generally good to live by these values and principles and use them to help figure out the right things to do given your particular context.

### Scrum Process

****

**Figure 1**. Principle and Different Stages

* + - * Scrum is an iterative and incremental agile software development framework for managing software projects and product or application development.
      * Scrum focuses on project management institutions where it is difficult to plan ahead.
      * Mechanisms of empirical process control, where feedback loops that constitute the core management technique are used as opposed to traditional command-and-control management.
      * Its approach to planning and managing projects is by bringing decision-making authority to the level of operation properties and certainties.
      * Scrum has three roles: product owner, scrum master and the development team members.
* **Benefit of the methodology:**
  + - * Project can respond easily to change.
      * Problems are identified early.
      * Customer gets most beneficial work first.
      * Work done will better meet the customer’s needs.
      * Improved productivity.
      * Ability to maintain a predictable schedule for delivery.

## Quality Management

### Estimates of Defects to be detected

**Table 6**. Pre-release review defects

|  |  |  |
| --- | --- | --- |
| **Process** | **Planned found by review** | **Actual found by review** |
| **Requirement** | 10 | 7 |
| **Design** | 20 | 22 |
| **Coding** | 50 | 42 |
| **Other** | 10 | 3 |
| **Total** | **90** | **74** |

### Quality Control

**Table 7.** Quality Control

| **Review Item** | | | | **Type of Review** | **Reviewer** | **When** |
| --- | --- | --- | --- | --- | --- | --- |
| Project Plan  Project schedule  CM Plan | | | | Group review  One-person review | Mentor  Team members | End of the Initiation stage |
| Business analysis and requirements specification document, Use Case Catalog | | | | Group review | All members | End of 90% of requirements |
| Design document, object model | | | | Group review | All members | End of 90% design |
| Stage plans | | | | One-person review | Mentor | Beginning of each stage |
| Complex/first specs incl. diagrams | time test | generated cases, | program interactive | Group review | Mentor  Team members | End of detailed design |
| Code | | | | Group review | All members | After coding for first few programs |

### Measurements Program

**Table 8.** Measurements Program

|  |  |  |  |
| --- | --- | --- | --- |
| **Data to be collected** | **Purpose** | **Responsible** | **When** |
| Size: No. of KLOC// FP | Early estimate project cost | PM/SM | At the end of the stages |
| Effort: No. person-day | Calculate project effort for scheduling | Team members | Daily |
| Quality: No. defects detected | Early evaluate product quality and the feasibility of the project | Reviewer, Tester | Right after the review/test |
| Schedule | Divide work and allocate resources properly, ensure the project is completed on time and on budget | PM/SM | Weekly and at the end of stages |

## Unit Testing Strategy

**Completion criteria:** Completion criteria are stated to for two purposes:

* + - * Identify acceptance criteria for product quality.
      * Identify when the testing is successfully executed

A clear statement of completion criteria should include the following items:

* + - * Function, behavior, or condition being measured
      * Method of measurement

Criteria or degree of conformance to measurement Special considerations:

This section should identify any influences or dependencies, which may impact or influence the test effort described in the test strategy. Influences might include:

Human resources (such as availability or need for non-test resources to support/participate in test) Constraints, (such as equipment limitations or availability, or the need/lack of special equipment) Special requirements, such as test scheduling or access to systems

Testing may be stopped when

* + - * It becomes unproductive
      * It requires a certain coverage
      * It requires a certain number of errors to be found
      * Schedule time runs out

## Manual Testing Strategy

Manual testing is a software testing process in which test cases are executed manually without using any automated tool. All test cases executed by the tester manually according to the end user's perspective. It ensures whether the application is working, as mentioned in the requirement document or not. Test cases are planned and implemented to complete almost 100 percent of the software application. Test case reports are also generated manually.

Manual Testing is one of the most fundamental testing processes as it can find both visible and hidden defects of the software. The difference between expected output and output, given by the software, is defined as a defect. The developer fixed the defects and handed it to the tester for retesting.

Manual testing is mandatory for every newly developed software before automated testing. This testing requires great efforts and time, but it gives the surety of bug-free software. Manual Testing requires knowledge of manual testing techniques but not of any automated testing tool.

Manual testing is essential because one of the software testing fundamentals is "100% automation is not possible". The advantages of Manual Testing:

* It does not require programming knowledge while using the Black box method.
* It is used to test dynamically changing GUI designs.
* Tester interacts with software as a real user so that they are able to discover usability and user interface issues.
* It ensures that the software is a hundred percent bug-free.
* It is cost-effective.
* Easy to learn for new testers.

# Estimation

## Size

**Table 9.** Software Scale Drivers

|  |  |
| --- | --- |
| **Software Scale Drivers** | |
| Precedentedness | *Nominal* |
| Development Flexibility | *High* |
| Architecture / Risk Resolution | *Nominal* |
| Team Cohesion | *High* |
| Process Maturity | *Nominal* |

**Table 10.** Software Cost Drivers

|  |  |  |  |
| --- | --- | --- | --- |
| **Software Cost Drivers** | | | |
| **Product** | | **Personnel** | |
| Required Software Reliability | *High* | Analyst Capability | *Nominal* |
| Data Base Size | *Nominal* | Programmer Capability | *High* |
| Product Complexity | *High* | Personnel Continuity | *Nominal* |
| Developed for Reusability | *High* | Application Experience | *Nominal* |
| Documentation Match to Lifecycle Needs | *Nominal* | Platform Experience | *Low* |
|  | | Language and Toolset Experience | *High* |
| **Project** | | **Platform** | |
| Use of Software Tools | *Nominal* | Time Constraint | *High* |
| Development | *Nominal* | Storage Constraint | *Nominal* |
| Required Development Schedule | *Nominal* | Platform Volatility | *Nominal* |

**Software Development (Elaboration and Construction)**

Effort = 14.6 Person-months

Schedule = 2.92 Months

Cost = $5118

Total Equivalent Size = 3000 SLOC

Effort Adjustment Factor (EAF) = 1.52

**Acquisition Phase Distribution**

**Table 11.** Acquisition Phase Distribution

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Phase** | **Effort (Person-months)** | **Schedule (Months)** | **Average Staff** | **Cost (Dollars)** |
| Inception | 0.9 | 1.1 | 0.8 | $307 |
| Elaboration | 3.5 | 3.2 | 1.1 | $1229 |
| Construction | 11.1 | 5.3 | 2.1 | $3890 |
| Transition | 1.8 | 1.1 | 1.6 | $614 |

**Software Effort Distribution for RUP/MBASE (Person-Months)**

**Table 12.** Software Effort Distribution for RUP/MBASE

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Phase/Activity** | **Inception** | **Elaboration** | **Construction** | **Transition** |
| Management | 0.1 | 0.4 | 1.1 | 0.2 |
| Environment/CM | 0.1 | 0.3 | 0.6 | 0.1 |
| Requirements | 0.3 | 0.6 | 0.9 | 0.1 |
| Design | 0.2 | 1.3 | 1.8 | 0.1 |
| Implementation | 0.1 | 0.5 | 3.8 | 0.3 |
| Assessment | 0.1 | 0.4 | 2.7 | 0.4 |
| Deployment | 0.0 | 0.1 | 0.3 | 0.5 |

## Estimated Effort

**Table 13.** The estimated effort

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Activity/ Process** | **Total Budget Effort Usage (pd)** | **Total % Budget Effort Usage (%)** | **Sprint 1** | | **Sprint 2** | | **Sprint 3** | | **Sprint 4** | | **Sprint 5** | |
| **No.** | **%** | **No.** | **%** | **No.** | **%** | **No.** | **%** | **No.** | **%** |
| Requirement | 55 | 12.6 | 25 | 23.8 | 7 | 6.8 | 7 | 8.1 | 12 | 15.2 | 4 | 6.5 |
| Design | 88 | 20.1 | 20 | 19 | 27 | 26.2 | 12 | 13.6 | 18 | 22.8 | 11 | 17.7 |
| Coding | 141 | 32.2 | 20 | 19 | 33 | 32 | 36 | 40.9 | 25 | 31.6 | 27 | 43.5 |
| Unit Testing | 18 | 4.1 | 3 | 2.9 | 4 | 3.9 | 5 | 5.7 | 3 | 3.8 | 3 | 4.8 |
| Testing | 35 | 8 | 6 | 5.7 | 8 | 7.8 | 8 | 9.1 | 7 | 8.9 | 6 | 9.7 |
| Support for Acceptance Test | 11 | 2.5 | 0 | 0 | 4 | 3.9 | 1 | 1.1 | 2 | 2.5 | 2 | 3.2 |
| Project Planning | 14 | 3.2 | 7 | 6.7 | 2 | 1.9 | 1 | 1.1 | 2 | 2.5 | 1 | 1.6 |
| Project monitoring | 22 | 5 | 6 | 5.7 | 4 | 3.9 | 4 | 4.5 | 4 | 5.1 | 4 | 6.5 |
| Quality Assurance | 22 | 5 | 3 | 2.9 | 5 | 4.9 | 6 | 6.8 | 4 | 5.1 | 4 | 6.5 |
| Training | 32 | 7.3 | 15 | 14.3 | 9 | 8.7 | 8 | 9.1 | 2 | 2.5 | 0 | 0 |
| **Total** | **438** | **100** | **105** | **100** | **103** | **100** | **88** | **100** | **79** | **100** | **62** | **100** |

## Schedule

### Work Breakdown Structure



**Figure 2.** Work Breakdown Structure

### Detailed Schedule

**Table 14.** Detailed Schedule

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Task Name** | **Start** | **Finish** | **Effort** |
| **1** | **Initial** | **21/02/2023** | **25/08/2022** | **28 hrs** |
| 1.1 | Gathering Requirement Meeting |  |  |  |
| 1.2 | Analyze Requirement |  |  |  |
| **2** | **Create Document** | **26/02/2023** | **28/02/2023** | **18 hrs** |
| 2.1 | Create Project Plan |  |  |  |
| 2.2 | Create User Stories |  |  |  |
| 2.3 | Create Product Backlog |  |  |  |
| **3** | **Development** | **01/03/2023** | **13/05/2023** | **2140 hrs** |
| **3.1** | **Sprint 1** | **01/03/2023** | **14/03/2023** | **428 hrs** |
| 3.1.1 | Sprint 1 Planning Meeting |  |  |  |
| 3.1.2 | Create Sprint Backlog |  |  |  |
| 3.1.3 | Create Test Plan document |  |  |  |
| 3.1.4 | Update Database |  |  |  |
| 3.1.5 | Change Oracle Database to MySQL Database |  |  |  |
| 3.1.6 | Update Database Structure |  |  |  |
| **3.1.7** | **Design user interface** |  |  | **80 hrs** |
| 3.1.7.1 | Design Household Survey |  |  |  |
| 3.1.7.2 | Design Report Form for Web |  |  |  |
| 3.1.7.3 | Design Report Form for App |  |  |  |
| **3.1.8** | **Coding** |  |  | **120 hrs** |
| 3.1.8.1 | Code Household Survey Feature |  |  |  |
| 3.1.8.2 | Code Report Form Feature |  |  |  |
| **3.1.9** | **Testing & Fix bug** |  |  | **120 hrs** |
| 3.1.9.1 | Design Test Case for Sprint 1 |  |  |  |
| 3.1.9.2 | Conduct test sprint 1 |  |  |  |
| 3.1.9.3 | Fix bug |  |  |  |
| **3.1.10** | **Release Sprint 1** |  |  |  |
| 3.1.10.1 | Sprint 1 Review Meeting |  |  |  |
| 3.1.10.2 | Sprint 1 Retrospective |  |  |  |
| **3.2** | **Sprint 2** | **15/03/2023** | **29/03/2023** | **428 hrs** |
| 3.2.1 | Sprint 2 Planning Meeting |  |  |  |
| 3.2.2 | Create Sprint 2 Backlog |  |  |  |
| 3.2.3 | Enhance Web UI |  |  |  |
| 3.2.4 | Deploy Mobile Application To Google Play Store |  |  |  |
| 3.2.5 | Deploy to Server |  |  |  |
| 3.2.6 | Collect District Coordinate |  |  |  |
| **3.2.7** | **Design user interface** |  |  | **80 hrs** |
| 3.2.7.1 | Design Dashboard Layout |  |  |  |
| 3.2.7.2 | Re-design UI for Login, Register, Forget Password |  |  |  |
| **3.2.8** | **Coding** |  |  | **120 hrs** |
| 3.2.8.1 | Code Dashboard Feature |  |  |  |
| 3.2.8.2 | Implement Map |  |  |  |
| 3.2.8.2 | Code Login, Register, Forget Password pages |  |  |  |
| **3.2.9** | **Testing & Fix Bug** |  |  | **120 hrs** |
| 3.2.9.1 | Design test case for Sprint 2 |  |  |  |
| 3.2.9.2 | Conduct test Sprint 2 |  |  |  |
| 3.2.9.3 | Fix Bug |  |  |  |
| **3.2.10** | **Release Sprint 2** |  |  |  |
| 3.2.10.1 | Sprint 2 Review Meeting |  |  |  |
| 3.2.10.2 | Sprint 2 Retrospective |  |  |  |
| **3.3** | **Sprint 3** | **30/03/2023** | **13/04/2023** | **428 hrs** |
| 3.3.1 | Sprint 3 Planning Meeting |  |  |  |
| 3.3.2 | Create Sprint 3 Backlog |  |  |  |
| 3.3.3 | Update New Version of Mobile Application to Store |  |  |  |
| 3.3.4 | Research Method to develop AI (Combine Household Information With Image Results To Make Decision) |  |  |  |
| 3.3.5 | Build Map Feature |  |  |  |
| **3.3.6** | **Design user interface** |  |  | **80 hrs** |
| 3.3.6.1 | Design UX/UI Edit Village page |  |  |  |
| **3.3.7** | **Coding** |  |  | **120 hrs** |
| 3.3.7.1 | Code Edit Village Feature |  |  |  |
| 3.3.7.2 | Implement Map |  |  |  |
| **3.3.8** | **Testing & Fix Bug** |  |  | **120 hrs** |
| 3.3.8.1 | Design Test Case for Sprint 3 |  |  |  |
| 3.3.8.2 | Conduct test Sprint 3 |  |  |  |
| 3.3.8.3 | Fix Bug |  |  |  |
| **3.3.9** | **Release Sprint 3** |  |  |  |
| 3.3.9.1 | Sprint 3 Review Meeting |  |  |  |
| 3.3.9.2 | Sprint 3 Retrospective |  |  |  |
| **3.4** | **Sprint 4** | **14/04/2023** | **28/04/2023** | **428 hrs** |
| 3.4.1 | Sprint 4 Planning Meeting |  |  |  |
| 3.4.2 | Create Sprint 4 Backlog |  |  |  |
| 3.4.3 | Update Database |  |  |  |
| 3.4.5 | Fake Training Data |  |  |  |
| 3.4.6 | Process Training Data |  |  |  |
| 3.4.7 | Change Workflow |  |  |  |
| **3.4.7** | **Fake training data** |  |  | **80 hrs** |
| 3.4.7.1 | Generate Script |  |  |  |
| 3.4.7.2 | Fake data |  |  |  |
| 3.4.7.3 | Code API Export Data |  |  |  |
| **3.4.8** | **Coding** |  |  | **120 hrs** |
| 3.4.8.1 | Update Detect Logic |  |  |  |
| 3.4.8.2 | Code API Data Set |  |  |  |
| **3.4.9** | **Testing & Fix Bug** |  |  | **120 hrs** |
| 3.4.9.1 | Design Test Case for Sprint 4 |  |  |  |
| 3.4.9.2 | Conduct test Sprint 4 |  |  |  |
| 3.4.9.3 | Fix Bug |  |  |  |
| **3.4.10** | **Release Sprint 4** |  |  |  |
| 3.4.10.1 | Sprint 4 Review Meeting |  |  |  |
| 3.4.10.2 | Sprint 4 Retrospective |  |  |  |
| **3.5** | **Sprint 5** | **29/04/2023** | **13/04/2023** | **428 hrs** |
| 3.5.1 | Sprint 5 Planning Meeting |  |  |  |
| 3.5.2 | Create Sprint 5 Backlog |  |  |  |
| 3.5.3 | Auto Update/Enhance AI model |  |  |  |
| 3.5.4 | Implement New Model |  |  |  |
| 3.5.5 | Enhance Pollution Detection API |  |  |  |
| 3.5.6 | Deploy New AI Version |  |  |  |
| 3.5.7 | Update New Version of Mobile Application to Store |  |  |  |
| **3.5.8** | **Design user interface** |  |  | **50 hrs** |
| 3.5.8.1 | Design Question Setting Page |  |  |  |
| 3.5.8.2 | Design Notification Feature |  |  |  |
| **3.5.9** | **Coding** |  |  | **120 hrs** |
| 3.5.9.1 | Code Question Setting Feature |  |  |  |
| 3.5.9.2 | Code Notification Feature |  |  |  |
| 3.5.9.3 | Code Update/Enhance AI model Feature |  |  |  |
| **3.5.10** | **Testing & Fix bug** |  |  | **120 hrs** |
| 3.5.10.1 | Design Test Case for machine learning Model |  |  |  |
| 3.5.10.2 | Conduct test machine learning Model |  |  |  |
| 3.5.10.3 | Design test case for Sprint 5 |  |  |  |
| 3.5.10.4 | Conduct test Sprint 5 |  |  |  |
| 3.5.10.5 | Fix Bug |  |  |  |
| **3.5.11** | **Release Sprint 5** |  |  |  |
| 3.5.11.1 | Sprint 5 Review Meeting |  |  |  |
| 3.5.11.2 | Sprint 5 Retrospective |  |  |  |
| **3.6** | **Review Project** | **15/05/2023** | **15/05/2023** | **2 hrs** |

### Project Schedule

**Table 15**. Project Schedule

| **No.** | **Activity** | **Start date** | **Responsible** | **Note** |
| --- | --- | --- | --- | --- |
| **Defect Prevention** | | | | |
|  | Sprint 1 | 01/03/2023 | Team members |  |
|  | Sprint 2 | 15/03/2023 | Team members |  |
|  | Sprint 3 | 30/03/2023 | Team members |  |
|  | Sprint 4 | 14/04/2023 | Team members |  |
|  | Sprint 5 | 29/04/2023 | Team members |  |
| **Quality Control** | | | | |
|  | Review: Work Product 1 | 14/03/2023 | Mentor - Team members |  |
|  | Review: Work Product 2 | 29/03/2023 | Mentor - Team members |  |
|  | Review: Work Product 3 | 13/04/2023 | Mentor - Team members |  |
|  | Review: Work Product 4 | 28/04/2023 | Mentor - Team members |  |
|  | Review: Work Product 5 | 13/05/2023 | Mentor -Team members |  |
| **Project Tracking** | | | | |
|  | Project initiation meeting | 21/02/2023 | Team members |  |
|  | Sprint 1 Planning Meeting | 01/03/2023 | Team members |  |
|  | Sprint 1 Review Meeting | 14/03/2023 | Team members |  |
|  | Sprint 2 Planning Meeting | 15/03/2023 | Team members |  |
|  | Sprint 2 Review Meeting | 29/03/2023 | Team members |  |
|  | Sprint 3 Planning Meeting | 30/03/2023 | Team members |  |
|  | Sprint 3 Review Meeting | 13/04/2023 | Team members |  |
|  | Sprint 4 Planning Meeting | 14/04/2023 | Team members |  |
|  | Sprint 4 Review Meeting | 28/04/2023 | Team members |  |
|  | Sprint 5 Planning Meeting | 29/04/2023 | Team members |  |
|  | Sprint 5 Review Meeting | 13/05/2023 | Team members |  |
| **QA** | | | | |
|  | Final Inspection: Deliverable 1 | 14/03/2023 | Mentor - Team members |  |
|  | Final Inspection: Deliverable 2 | 29/03/2023 | Mentor - Team members |  |
|  | Final Inspection: Deliverable 3 | 13/04/2023 | Mentor - Team members |  |
|  | Final Inspection: Deliverable 4 | 28/04/2023 | Mentor - Team members |  |
|  | Final Inspection: Deliverable 5 | 13/05/2023 | Mentor - Team members |  |
|  | Baseline audit: Startup | 20/05/2023 | Mentor - Team members |  |
|  | Baseline audit: Wrap-up | 20/05/2023 | Mentor - Team members |  |

## Resource

Specified as in the section [Project Team](#_heading=h.34g0dwd)

## Infrastructure

**Table 16**. Infrastructure

|  |  |  |
| --- | --- | --- |
| **Work/Product** | **Purpose** | **Expected Availability by** |
| **Development Environment** | | |
| Flutter | Development framework | Construction stage |
| Spring Boot | Development framework | Construction stage |
| Oracle SQL | Database | Construction stage |
| Python | Development language | Construction stage |
| HTML/CSS/JavaScript | Development language for Web | Construction stage |
| Material UI | Supporting library for UI Web | Construction stage |
| Katalon | Testing | Construction stage |
| **Hardware & Software** | | |
| 1GB space on server |  | Initiation stage |
| Browser |  | Construction stage |
| Emulator |  | Construction stage |
| **Other Tools** | | |
| Github | Source version control | Initiation stage |
| Trello | Task management tool | Initiation stage |

## Training Plan

**Table 17.** Training Plan

|  |  |  |  |
| --- | --- | --- | --- |
| **Training Area** | **Participants** | **When, Duration** | **Waiver Criteria** |
| Technical | | | |
| Flutter | Van Cong Le Ca | 7 days | If already trained |
| Spring Boot | Hua Hoang Phuc | 3 days | If already trained |
| Material UI | Nguyen Thanh Trung | 5 days | If already trained |
| Python | Bui Duc Huy | 7 days | If already trained |
| Spring Boot, AWS | Huynh Ba Nhan | 7 days | If already trained |
| Process | | | |
| Task management | Van Cong Le Ca | 8 hrs | If already trained |
| Human management | Van Cong Le Ca | 8 hrs | If already trained |
| Defect prevention | Van Cong Le Ca | 1 day | Mandatory |

## Budget for Project

**Table 18.** Total Cost Estimate

| **No** | **Criteria** | **Price** | **Amount** | **Total (USD)** |
| --- | --- | --- | --- | --- |
| 1 | Working hours | $ 2 | 2188 | $ 4376 |
| 2 | Online server and services | $ 100 | 5 | $ 500 |
| 3 | Party | $ 150 | 2 | $ 300 |
| **Total cost** | | | | **$ 5176** |

**Table 19**. Cost Description

|  |  |  |
| --- | --- | --- |
| **Description** | **Amount** | **Unit** |
| Number of members | 5 | Person |
| Number of working hours per weekdays | 5 | Hour |
| Number of working hours per two weekends | 8 | Hour |
| Number of working days per week | 7 | Day |
| The duration of the project | 3 | Month |
| The cost per member per week | 82 | USD |
| Party cost per time | 10 | USD |
| The number of working days | 93 | Day |

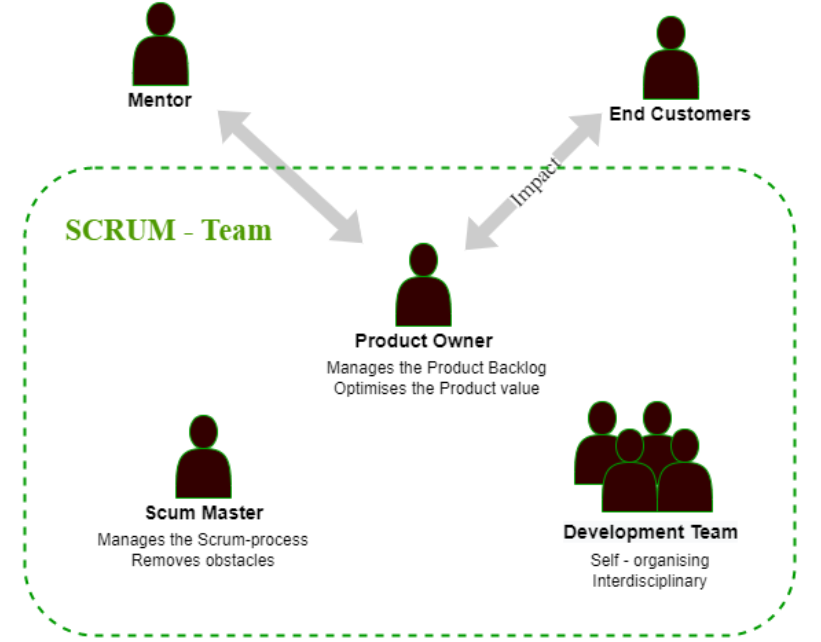
***\*\* Explain:*** *Amount of working hours = 5 members \* (5 hours \* 67 Days + 8 hours \* 26 days)*

**Table 20**. Estimate Budget

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Item** | **Total Budget** | **% Budget** | **Budget in Period** | | | | | **Note** |
| **Sprint 1** | **Sprint 2** | **Sprint 3** | **Sprint 4** | **Sprint 5** |  |
| Purchases (COTS) | 4376 | 84.5% | 875.2 | 875.2 | 875.2 | 875.2 | 875.2 |  |
| Team Building | 300 | 5.8% | 150 | 0 | 0 | 0 | 150 |  |
| Tools | 500 | 9.7% | 100 | 100 | 100 | 100 | 100 |  |
| Travel Costs | 0 | 0 | 0 | 0 | 0 | 0 | 0 |  |
| Training | 0 | 0 | 0 | 0 | 0 | 0 | 0 |  |
| Review Activities | 0 | 0 | 0 | 0 | 0 | 0 | 0 |  |
| Other | 0 | 0 | 0 | 0 | 0 | 0 | 0 |  |
| **Total** | **5176** | **100%** | **1125.2** | **975.2** | **975.2** | **975.2** | **1125.2** |  |

# Project Organization

## Organization Structure



**Figure 3.** Organization Structure

## Project Team

**Table 21.** Project Team

| **Role** | **Responsibility** | **Name** |
| --- | --- | --- |
| **Product Owner** | - A spokesperson for the customer and needs to represent them   * Gathers, manages, and prioritizes the product backlog. * Has technical product knowledge or specific domain expertise. * Tracks progress towards the release of a product. | Mr. Nguyen Thanh Binh |
| **Scrum Master** | * Communicate the value of Scrum * Teach the organization on Scrum to maximize business value * Attend all Scrum meetings * Preserve the integrity and spirit of the Scrum framework * Maintain the focus of the Team and facilitate efforts to resolve them * Serve as a coach and mentor to members of the Team * Respectfully hold the Team, Product Owner and Stakeholders accountable for their commitments * Continually work with the Team and business to find and implement improvements * As a timekeeper * Record team meeting   Make the Team aware of impediments | Van Cong Le Ca |
| **Developer** | - Responsible for quality  - Responsible for delivering the potentially shippable product of the Application each sprint  - Report progress based on the remaining time  - Self-organized  - Owns the Sprint backlog | All members |
| **Tester** | * - Do the Test plan * - Creation of test designs, test processes, test cases and test data. * - Carry out testing as per the defined procedures. * - Graph the results and make sure people know when test results decline. * - Prepare all reports related to software testing carried out. * - Analysis and evaluate the Test result. * - Ensure that all tested related work is carried out as per the defined standards and procedures. | All Members |
| **Mentor** | * Guide on the process. * Monitoring all activities of the Team. * Help with anything. * Reviews project documents   - Reviews product | Mr. Nguyen Thanh Binh |

# Communication & Reporting

**Table 22.** Communication Methodology

| **Audience/ Attendees** | **Topic/ Deliverable** | **Frequency** | **Method** |
| --- | --- | --- | --- |
| **Mentor and Team member** | Project Progress Review | Weekly | Skype Meeting |
| **Team Member** | Project Progress Review and Daily Meeting | Daily | Remotely, Face to Face |

# Configuration Management

**Table 23.** Configuration Management

|  |  |  |
| --- | --- | --- |
| **No** | **Tool** | **Content** |
| 1 | Google Sheet | Track member activities. At the end of each day, team members will post on time log and scrum master will check. |
| 2 | Google Document | Track the changing of documents & manage versions of documents. |
| 3 | GitHub | Repositories for source code version management |
| 4 | Weekly Meeting | Hold a meeting every week to assign tasks to each member.  If there are some emergencies but we cannot sit together then we can use Google Meet to discuss online. |
| 5 | Document | All meetings must be documented and pictured. |
| 6 | Google Drive | Store document resources and designed components |
| 7 | Google Meet | Discuss online, stream and share problems |

# Security Aspects

* **About copyright:**

Use images, logos, and information about the university to be allowed for use

* **About security:**

All documents, and software products must be authorized by team members before they can be obtained or referenced

* **About integrity:**

Software products run continuously 24 / except for external problems such as natural disasters, power outages ... and when upgrading the software

To meet the above criteria requires action:

**Copyright**: ask for the permission of the provider before using the resource

**About security**: Use 3rd party services of reputable providers and security notices for each team member

Integrity: software product data must be backed up continuously

# References

**Table 24.** Acronym

|  |  |  |
| --- | --- | --- |
| **Acronym** | **Definition** | **Note** |
| CVPMS | Craft Village Pollution Monitoring System |  |
| FDD | Feature-Driven Development |  |

**References:**

* [WMS]Proposal
* What is Scrum: <https://www.scrum.org/resources/what-is-scrum>
* https://www.javatpoint.com/manual-testing