

**International School**

CMU-CS 445 SAIS

**Project Plan**

**COFFEE SHOP**

**Submitted by**

**Kiet,Nguyen Tuan**

**Khai,Nguyen Van**

**Diem,Dinh Thi Kim**

**Duc,Tran Van**

**Proposal Review Panel Representative:**

Name Signature Date

**Capstone Project 1- Mentor:**

Name Signature Date

**PROJECT INFORMATION**

|  |  |  |  |
| --- | --- | --- | --- |
| **Project acronym** | CFS | | |
| **Project Title** | COFFEE SHOP | | |
| **Start Date** | 01 July 2023 | **End Date** | 08 July 2023 |
| **Lead Institution** | International School, Duy Tan University | | |
| **Scrum master**  **/ Project Leader & contact details** | Kiet,Nguyen Tuan  Email:  Tel: | | |
| **Partner Organization** |  | | |
| **Project Web URL** |  | | |
| **Team members** | Name | Email | Tel |
|  | Khai,Nguyen Van |  |  |
|  | Diem,Dinh Thi Kim |  |  |
|  | Duc,Tran Van |  |  |

**TABLE OF CONTENTS**

1. **[PROJECT OVERVIEW 6](#_heading=h.30j0zll)**
   1. [Project Description 6](#_heading=h.3znysh7)
   2. [Scope and Purpose 6](#_heading=h.tyjcwt)
   3. [Assumptions and Constraints 6](#_heading=h.2s8eyo1)
   4. [Project Objectives 7](#_heading=h.3rdcrjn)
   5. [Project Risk 8](#_heading=h.lnxbz9)
2. **[PROJECT DEVELOPMENT APPROACH 9](#_heading=h.z337ya)**
   1. [Technical Process 9](#_heading=h.1y810tw)
   2. [Unit Testing Strategy 9](#_heading=h.2xcytpi)
   3. [Integration Testing Strategy 9](#_heading=h.3whwml4)
   4. [System Testing Strategy 10](#_heading=h.qsh70q)
3. **[ESTIMATION 14](#_heading=h.1hmsyys)**
   1. [Size 14](#_heading=h.vx1227)
   2. [Schedule 15](#_heading=h.2u6wntf)
   3. [Resource 17](#_heading=h.nmf14n)
   4. [Infrastructure 17](#_heading=h.1mrcu09)
   5. [Training Plan 18](#_heading=h.1mrcu09)
4. **[PROJECT ORGANIZATION 20](#_heading=h.206ipza)**
   1. [Organization Structure 20](#_heading=h.2zbgiuw)
   2. [Project Team 20](#_heading=h.3ygebqi)
5. **[COMMUNICATION & REPORTING 25](#_heading=h.sqyw64)**
6. **[CONFIGURATION MANAGEMENT 27](#_heading=h.4bvk7pj)**
7. **[SECURITY ASPECTS 28](#_heading=h.1664s55)**

# PROJECT OVERVIEW

# Project Description

|  |  |  |  |
| --- | --- | --- | --- |
| **Project code** | CFS | **Contract type** | Internal Project |
| **Customer** | N/A | **2nd Customer** | N/A |
| **Project Level** | Group | **Project rank** | A |
| **Group** | 3 | **Division** |  |
| **Project Type** | External | **Project Manager/ Scrum master** | Kiet,Nguyen Van |
| **Project Category** | N/A | **Business domain** |  |
| **Application type** | Web Application |  |  |

# Purpose and Scope

# Purpose

●Build a website to help students register for topics directly on the website, look up information about topics as well as related documents.

● On the management side, you can control the topics that students register for, approve topics, and cancel topics.

# Scope

Research Limitations: Clearly defining what will be included and excluded from the study.

* Research Methodology: Describing the approach to solving the problem, including research methods, tools used, and the data scope.
* Principles and Criteria: Outlining criteria or principles to be applied in the research process to ensure accuracy and reliability of results*.*

# Assumptions and Constraints

|  |  |  |
| --- | --- | --- |
| No | Description | Note |
| Assumptions | | |
| 1 | Laravel version v9.0 (or above) and lower version not supported. | Scope |
| 2 | 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 | The project is developed within 114 day and quarterly deployed on the market. | Schedule |
| 2 | The project shall conform to security requirements specified by the customer in the NDA | Security |
| 3 | The product operates at a high level of performance and has a page load of no more than 5 seconds. | Quality |
| 4 | The system can run on any web platform | Scope |
| 5 | The project will be implemented by a team including 4 members | Resources |

# Project Objectives

# Standard Objectives

|  |  |  |  |
| --- | --- | --- | --- |
| Metrics | Unit | Committed | Note |
| Start Date | dd-mmm-yy | 01-july-24 |  |
| End Date | dd-mmm-yy | 08-july-23 |  |
| Maximum Team Size | Person | 4 |  |

# Specific Objectives

# •We will build a website to help customers buy drinks faster and more conveniently.

# •The system operates with high performance and is safe for users. User security data is encrypted and stored carefully, avoiding data loss.

# •The system implemented by the project team minimizes errors and controls risks well.

# 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. Replan 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 |

.

# 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

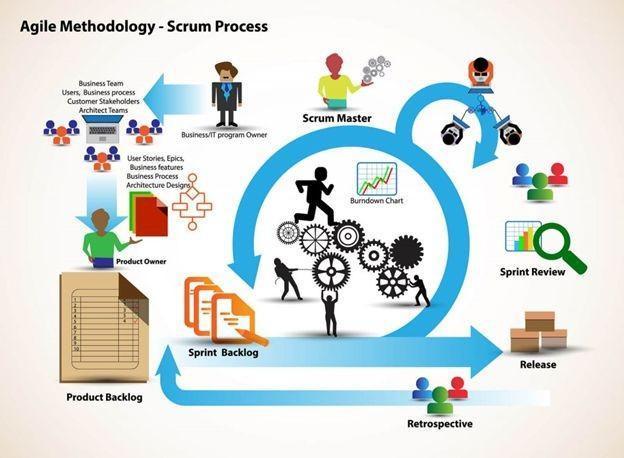
Agile software development refers to a group of software development methodologies based on iterative development, where requirements and solutions evolve through collaboration between self-organizing cross-functional teams.

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.

**2.1.2.1. Scrum Process**



#### About Scrum:

Scrum is a subset of Agile. It is a lightweight process framework for agile development, and the most widely-used one.

Scrum is most often used to manage complex software and product development, using iterative and incremental practices. Scrum significantly increases productivity and reduces time to benefits relative to classic “waterfall” processes. Scrum processes enable organizations to adjust smoothly to rapidly-changing requirements and produce a product that meets evolving business goals.

An agile Scrum process benefits the organization by helping it to

* Increase the quality of the deliverables
* Cope better with change (and expect the changes)
* Provide better estimates while spending less time creating them
* Be more in control of the project schedule and state

* + 1. **Strategy for Meeting Quality Objectives**

|  |  |
| --- | --- |
| Strategy | Expected Benefits |
| Implement defect prevention using standard guidelines and JavaScript coding standards | Aim for a 15–25% reduction in defect injection rate and approximately 5% improvement in productivity |
| Conduct group reviews of program specifications for initial and complex use cases | Expected improvement in quality by enhancing overall defect removal efficiency; some productivity benefits by detecting defects early |
| Organize a group review of design documents and first-time code by project leader, developer, and consultant | Aim for better quality by ensuring robustness in initial code development and design; early detection of potential issues |

* + 1. **Quality Control**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Review Item | | Type of Review | Reviewer | | | When | | |
| Proposal | | Group review | Kiệt  Khải  Diễm  Đức |  | | Initial | | |
| Project plan Project schedule  Test Plan | | Group review Group review  One-person review | Kiệt  Khải  Diễm  Đức | | | End of Initiation stage | | |
| Business analysis and requirements specification document, Use Case catalog | | Group review | Kiệt  Khải  Diễm  Đức | | | End of requirements | 90% | of |
| Design document, object model | | Group review | Kiệt  Khải  Diễm  Đức | | | End of 90% design | | |
| Complex/first specs incl. diagrams time test Genera ed cases, program interactive | Group review | Kiệt  Khải  Diễm  Đức | | | End of detailed design | | | |
| Code | | Group review | Kiệt  Khải  Diễm  Đức | | | After coding for first few programs | | |

* + 1. **Measurements Program**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data to be collected** | **Purpose** | **Responsible** | **When** |
| Size: No. of KLOC/ FP | Early estimate project cost | Team members | At the end of 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 | mentor | Right after the review/test |
| Schedule | Divide work and allocate resources properly, ensure the project is completed on time and on budget | Kiet,Nguyen Tuan | Weekly and at the end of stages |

## Unit Testing Strategy

* *Grey Box:*
  + It is a combination of a Black Box and White Box testing. It is the type of testing in which tester aware with internal functionality of a method or unit but not in a more deep level like white box testing. In this, the user partially aware of the internal functionality of a system.
  + Write test cases before fixing the defect and independent of each other.
  + Write cases to verify behavior, also write test cases to ensure the performance of the code
  + Execute test cases continuously and frequently.
  + Using tool: Install and run Jest for writing unit test in java(JDK)
* Isolation of a code – Isolate function to test it more rigorously. Isolate code to do Automated Unit Testing in a better way. Isolating functions/code helps to do testing in a good way. It helps to reveal dependencies between functions of code.

## Integration Testing Strategy

* *Bottom up Strategy:*
  + The components below are first written and these are integrated first. The integration happens from bottom to top. If the calling component is yet to be developed, it is replaced by a specially written component called a Drive
  + When we finish each product backlog, we test it out before we finish.
* *Bigbang Strategy:*
  + All components are put together at the same time, there is no order, except all are integrated at the same time.
  + Towards the end of the project, we started to apply this tactic to test the entire application.

## System Testing Strategy

* *Automation strategy:*
  + Automation Testing or Test Automation is a software testing technique that performs using special automated testing software tools to execute a test case suite.
  + The automation testing software can also enter test data into the System Under Test, compare expected and actual results and generate detailed test reports. Software Test Automation demands considerable investments of money and resources.
  + Testing tools: Katalon Studio, Appium.
* *Customer testing(Beta testing) strategy:*
  + Beta testing is a type of user acceptance testing where the product team gives a nearly finished product to a group of target users to evaluate product performance in the real world.
  + We are rolling out a beta app on the Google Store early on for testing. After that, we gathered all the feedback and improved our system.

## Schedule

* + 1. **Project Milestone & Deliverables**

|  |  |  |
| --- | --- | --- |
| **No** | **Activities** | **Deliverable** |
| **1** | **Project Proposal** | **Project Proposal Document 1.1** |
| **2** | **Database Design** | **Database Design Document v1.1** |
| **3** | **Interface Design** | **Interface Design Document v1.1** |

## Resource

Specified as in the section [Project Team](#_heading=h.i17xr6)

## Infrastructure

|  |  |  |  |
| --- | --- | --- | --- |
| **Work/Product** | **Purpose** | **Expected Availability by** | **Note** |
| **Development Environment** | | | |
| Windows 11 | Operating System | Initiation stage |  |
| My SQL ,visual studio code | DBMS | Initiation stage |  |
| C++ | Development language for Web interface | Initiation stage |  |
| **Hardware & Software** | | | |
| 4 Personal Laptop | Design, Develop and emulation | Initiation stage |  |
| **Other Tools** | | | |
| Git | Source version control | Definition stage |  |

# PROJECT ORGANIZATION

## Organization Structure

|  |  |  |
| --- | --- | --- |
| **Role** | **Responsibility** | **Name** |
| **Scrum Master** | * Communicate the value of Scrum * Teach the organization on Scrum to maximize business value * Preserve the integrity and spirit of the Scrum framework * 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 * Helping the team agree on what they can achieve during each development sprint (or other period of time). * Facilitating the daily standup (sometimes called the daily scrum) and helping the team reach consensus on each of the three questions. * Helping the team continuously make progress on the project by making sure each person is working on the right tasks, helping to remove any obstacles to the team members’ progress, and protecting the team from distractions. | Kiet,Nguyen Tuan |
| **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 |

## Project Team

|  |  |
| --- | --- |
| **Full Name** | **Position** |
| Kiet,Nguyen Tuan | Scrum Master, Dev-team |
| Duc,Tran Van | Dev-team |
| Diem, Dinh Thi Kim | Dev-team |
| Khai,Nguyen Van | Dev-team |

# COMMUNICATION & REPORTING

|  |  |  |  |
| --- | --- | --- | --- |
| **Audience / Attendees** | **Topic / Deliverable** | **Frequency** | **Method** |
| Scrum Master, Members | Daily meeting | Daily | Face to Face / Zoom Meeting |
| Scrum Master, Members | Sprint Planning Meeting | When starting a sprint | Zoom Meeting |
| Scrum Master, Members, Mentor | Sprint Review Meeting | When finishing a sprint | Face to face, Zoom Meeting |
| Scrum Master, Members | Sprint Retrospective | When the sprint review finish | Face to Face |
| Scrum Master, Members | Individual Meeting | When need | Face to Face, Zoom ,Discord,Meeting, Message |
| Scrum Master, Members, Mentor | Working report, review problems | Once a week | Face to face |

# CONFIGURATION MANAGEMENT

In student final project management, configuration management holds significant importance in ensuring effective control and maintenance of project configurations. A configuration management plan is established to delineate objectives, scope, processes, tools, supporting software, version control, change management, testing, validation, risk management, and handling configuration-related documentation. Adherence to the configuration management plan ensures the stability of the project and its alignment with user requirements.

# SECURITY ASPECTS

* The credential data is carefully secured by multi-layer encryption and data integrity is ensured. Regularly back up system data.
* Research on network attack prevention solutions to ensure data security, avoid being exploited and stolen data by hackers.
* Deploy project architecture with a high priority in security. Optimized architectural solutions enable the deployment of data security with 99% reliability.
* Social media, sharing and use of data must be approved by the end user and verified by the organization's management.

### DEFINITIONS AND ACRONYMS

|  |  |  |
| --- | --- | --- |
| **Acronym** | **Definition** | **Note** |
| PM | Project Manager |  |
| PTL | Project Technical Leader |  |
| QA | Quality Assurance Officer |  |
| CC | Infrastructure Configuration Controller |  |
| DV | Developer |  |
| URD | User Requirement Document |  |
| SRS | Software Requirement Specification |  |
| ADD | Architecture Design Document |  |
| DDD | Detail Design Document |  |
| TP | Test Plan |  |
| TC | Test Case |  |
| SC | Source Code |  |
| CM | Configuration Management |  |
| CSCI | Computer Software Configuration Items |  |
| CI | Configuration Item |  |
| CCB | Change Control Board |  |