|  |  |  |
| --- | --- | --- |
| **Project Plan** | | |
| Cross Ref. VSE-29110 | Coverage Level: | Version : |
|  |  |  |

|  |  |
| --- | --- |
| **Process Ownership** | **Approving Authority** |
|  |  |
| **Scope** | **Approved Date** |
|  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Document History** | | | | |
| **Version Number** | **Record Date** | **Prepared/modified by** | **Reviewed by** | **Change Details** |
| 1.0 |  | Kanittee,Pimchittra | Kanittee, Pimchittra | Draft |
|  |  |  |  |  |

**Document develops by**

Kanittee Hongron

Pimchittra Sukkasem

**Document Approved by**

Ms. Yunrim Park

**Objective:**

-To provide guidelines to prepare a minimum Project Management Plan for projects handled within company.

-To provide checklists and templates that ensure the relevant aspects of project management are covered.

|  |  |  |
| --- | --- | --- |
| **Project Information** | | |
| **Name** | **Phase** | **Description** |
| The Chiang Mai Red Taxi service Assistant | Requirements Specification | Overview of Project. |
| Requirements Specification | Management Schedule, Quality Stand, Estimate risk, Cost, and configuration management. |
| Requirements Specification | Project requirement that has analyzed from developer teams and accepted from users. |
| Requirements Specification | Traceability Document that can find, change, update when errors occurs. |
| System Design | Design of software which is algorithm, diagrams, database design, so on. |
| Implementation | Construction process, which is implementation and integration. |
| Testing | Document about test case for a system. |

**Chapter One**

**Introduction**

The Chiang Mai Red Taxi Service Assistant is an application that passengers can book Red taxi in Chiang Mai and select where they want to go from the general lists. A passenger sends a request to a driver, and the driver can accept the request and the system will send a message to the passenger. The application will run on Android platform.

**Project Overview**

The overview of the project will be a mobile application in the form of Android platform. The main objective of the application is the system can respond the request from a passenger to a driver. The driver can accepts the request and sends the acceptance back to the system.

* 1. **Purpose**

The software development plan is a document for planning and evaluating the project. The plan can reduce the cost from project and set the time in milestone. The project plan before start project and when project start, project plan used to control project follow the plan.

* 1. **Scope**

The application will set in to two parts. The server side will be developed in PHP. The client side will be developed using android api.

* 1. **Acronyms and Definition**

**Acronyms**

UI User Interface

UC User Case

SRS Software Requirement Specification

AC Activity Diagram

SDD Software Design Document

**Definition**

**IEEE**

Institute for Electrical and Electronics Engineers. Biggest global interest group for engineers of different branches and computer scientists. [IEEE90]

**Integration Testing**

The progressive linking and testing of software component in order to ensure their proper functioning in the whole system. [IEEE90]

**Milestone**

A significant event in the project, usually completion of the main deliverable. [IEEE90]

**Plan**

A documented series of tasks requires meeting an objective, typically including the associated schedule, budget, resources, Organizational description and work breakdown structure. [IEEE90]

**Project management**

The application of knowledge, skills, tools, and techniques to project activities in order to meet or exceed stakeholder needs and expectations from a project. [IEEE90]

**Project Plan**

A formal, approved document used to guide both project execution and project control. The primary uses of the project plan are to document planning assumptions and the decision, to facilitate communication among stakeholders, and to document approved scope, cost, and schedule baseline. [IEEE90]

**Risk**

An uncertain event or condition that, if it occurs, has a positive or negative effect on a project’s objectives. It is a function of the probability of occurrence of a given threat’s occurrence. [IEEE90]

**Risk management**

The systematic application of management policies, procedures and practices to the tasks of identifying, analyzing, evaluating, treating and monitoring risk. [IEEE90]

**System testing**

Testing conducted on complete and integrated system for evaluate the system’s compliance with its specified requirements. [IEEE90]

**Traceability**

The ability to trace the history, application or location of an item or activity, or work products or activities, by means of recorded identification. The establishment and maintenance of relationships between such items. Horizontal traceability describes the relationship between work products of the same type (e.g., Customer requirements). Vertical traceability describes the relationship between work products, which build or derived from each other (e.g., From customer requirements to qualification test cases). Bidirectional traceability allows to directly following relationship in both directions. [IEEE90]

**Validation**

Confirmation by examination and provision of objective evidence that the particular requirements for a specific intended use are fulfilled (“doing the right thing”). Part of quality control. [IEEE90]

**Verification**

Confirmation at the end of the process by examination and provision of objective evidence that specified requirements to the process have been fulfilled (“doing things right”). Part of quality control. [IEEE90]

**UML United Modeling Languages**

Standardized notation for modeling design descriptions, architectures or scenarios. Not depending on a specific method. Issued and maintained by the object Management Group (OMG). [IEEE90]

**Unit test**

A test of individual program or modules in order to remove a design or programming errors. [IEEE90]

* 1. **Work Product to be developed**
     1. **Deliverables**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Deliverables/Release** | **Media** | **No. of Copies** | **Date** |
| 1. | **The Proposal Report**  -Project Proposal version 1.4 | Hard Copy | 3 |  |
| 2. | **The Progress Report I**  -Project Management Plan version 1.0  -Software Requirement Specification version 1.0  -Software Design Document version 1.0  -Test Plan version 1.0  -Traceability record version 1.0 | Hard Copy | 3 |  |
| 3. | **The Progress Report II**  - Project Management Plan  version 2.0  - Software Requirement  Specification version 2.0  - Software Design  Document version 2.0  - Test Plan version 2.0  - Test Record version 2.0  - Traceability record version  2.0 | Hard Copy | 3 |  |
| 4. | **The Final Progress Report**  - Project Management Plan  version 3.0  - Software Requirement  Specification version 3.0  - Software Design  Document version 3.0  - Test Plan version 3.0  - Test Record version 3.0  Traceability record version  3.0 | Hard Copy | 3 |  |

**Chapter Two**

**Infrastructure**

**2.1 Hardware/Software Acquisition Plans**

**Hardware**

-Smart phone with android operating system.

**Software**

-PHP

-SAMPP server

- Java Android SDK

**2.2 Project Specific system support needed**

**Software needed:**

-PHP

-JAVA

-Android SDK

-Server

**Minimum spec:**

-Android 4.3

-Internet connection

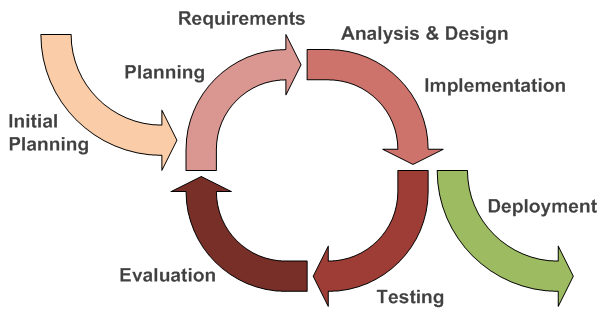
**2.3 Management Procedures**

**2.3.1 Project Team Structure**

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Participant** | **Roles** | **Responsibility** |
| 1. | Kanittee Hongron | Designer, Programmer, System Analysis, Tester | -Project Proposal  - Project Management  Plan  - Software  Requirement  Specifications  - System test document  - Software Implementation  - Traceability record |
| 2. | Pimchittra Sukkasem | Designer, Programmer, System Analysis, Tester | -Project Proposal  - Project Management  Plan  - Software  Requirement  Specifications  - System test document  - Software Implementation  - Traceability record |

**2.4 Monitoring And controlling Mechanisms**

**2.4.1 Software Development Model**

****

*Figure1: Iterative software development model*

The iterative software development model as shown in Figure 8, begins by implementing and specifying a portion of the software instead of specifying the full requirements. It is then reviewed along the way to find and add more requirements as needed. The model is broken down into increments containing a number of smaller life cycle stages with each part including a new function to the product.

**Proposal phase**: This phase is about creating a proposal for The Chiang Mai Red Taxi Service Assistant. The proposal contains about project introduction, technologies and tool involved, quality standard and project plan.

**Document Plan phase**: This phase is about document for planning and designs the overall system from requirement given by the user. These documents are Project Management Plan, Software Requirement specification and Software Design Document.

**Iterative all features**: This phase is about separate system into many features and then iterative create all feature from the first feature till the final feature. For this phase, it will be divided into 4 phases. There are;

- Plan: Planning the method for creating and test each feature.

- Implement: Implement and coding each feature.

- Test: Test and debug each feature.

- Review: Review and maintain each feature to meet the feature plan.

**System test phase**: This phase will integrate all features together into one system and then create test document from system testing.

**Deploy phase**: This phase is about deploy the whole system to server and use as a regular mobile application.

**2.4.2 Status Reporting**

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Progress Report** | **Software items** | **Date** |
| 1. | Progress I | -Project Management Plan v.1.0  -Software Requirement Specification v.1.0  -Software Design Document v.1.0  -Main Features (Feature#1-5)  -Traceability Record I v.1.0  -Test Plan v.1.0  -Test record v.1.0 |  |
| 2. | Progress II | -Main features (Feature#6-10)  -Traceability Record II v.2.0  -Test Plan v.2.0  -Test Record v.2.0 |  |
| 3. | Progress ShowPro | -Sub features (Feature#11-15)  -Review the features  -Traceability Record ShowPro  -Test Plan v.3.0  -Test Record v.3.0 |  |
| 4. | Final Progress | -Final Document v.4.0  -Final System v.4.0 |  |

**Features**

**Main Features**

- #1: Both passengers and drivers can register themselves to the system.

- #2: Passengers can set the destination and the number of passengers.

-Feature#3: Passengers can send a request to a driver.

-Feature#4: Passengers can get the confirmation if driver accepts the request.

-Feature#5: Passengers can see booking information of red taxi.

-Feature#6: Passengers can view the current location of the red taxi matching their conditions.

-Feature#7: Drivers can update the number of available seats.

-Feature#8: Drivers can tell where the taxi is heading.

-Feature#9:Drivers can get the request from passenger.

-Feature#10: Drivers can either accept or decline the request.

**Sub Features**

-Feature#11: Passengers can cancel the request.

-Feature#12: Passengers can get the notification when the red taxi arrives.

-Feature#13: Passengers can create schedules (for planning of routes).

-Feature#14: Passengers can change the scope of searching for red taxis1.

-Feature#15: Drivers can receive the cancellation request.

-Feature#16: Drivers can choose to enable or disable the service (e.g. off duty).

**2.5 Change Management**

**Change control procedure**

1. Admit the change.
2. Analyze the reason for the change.
3. Send change form to Project advisor.
   1. If accept: Make a changed in project from the change request form.
   2. If not accept: Continued in the project and find the way to solve the problem.
4. Analyze the result from changing and modify document or system to match with a change document.

**Chapter Three**

**Quality Planning**

**3. Quality Planning [V&V]**

**3.1 Quality Factors**

According to McColl’s factors model, The Chiang Mai Red taxi Service Assistant should meet these quality factor after completed:

**3.1.1 Product Operation factors**

**Correctness**

-The system should be able to 100% expected result from displaying and tracking the red taxi on mobile application.

-The information in the system should be up to date on time.

**Reliability**

-The system should have failure rate lower than 10% after deployed to server.

-The system should transfer information via on server.

**Integrity**

-The system should be able to identify authentication of the drivers.

-The system should be able to

**3.1.2 Product revision factors**

**Maintainability**

-The software should have at least 20% of comment to the whole Line of code.

-The software should return output for maintenance.

**Testability**

-The system should be able to test all features flow in process.

**Portability**

-The system should be support on wifi or 3G internet.

**Reusability**

-The software should be developed and used in window phone or IOS.

**3.2 Reviews/Responsibility**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Stage Exit Review** | | | | |
| **No.** | **Stage** | **Review Item** | **Responsibility** | **Reviewed** |
| 1 | Requirement gathering and analysis | Project Proposal | Kanittee, Pimchittra | Kanittee, Pimchittra |
| 2 | Requirement gathering and analysis | Project Management Plan | Kanittee ,Pimchittra | Kanittee, Pimchittra |
| 3 | Requirement gathering and analysis | Software Requirement specification | Kanittee, Pimchittra | Kanittee, Pimchittra |
| 4 | Requirement gathering and analysis | Traceability Record | Kanittee, Pimchittra | Kanittee, Pimchittra |
| 5 | System design | Software Design Document | Kanittee, Pimchittra | Kanittee, Pimchittra |
| 6 | Implementation | Code | Kanittee, Pimchittra | Kanittee, Pimchittra |
| 7 | Testing | System Test Record | Kanittee, Pimchittra | Kanittee, Pimchittra |
| 8 | Testing | Unit Test Record | Kanittee ,Pimchittra | Kanittee, Pimchittra |

**3.3 Testing**

|  |  |  |
| --- | --- | --- |
| **Test Process** | | |
| **No.** | **Test** | **Responsibility** |
| 1 | Unit Testing | Kanittee, Pimchittra |
| 2 | System Testing | Kanittee, Pimchittra |

**Chapter Four**

**Software Standard**

**4.Software Development Standard**

**ISO 29110 for Very Small Entity (VSE)**

ISO 29110 is a guide applies to Very Small Entities (VSEs), enterprise, organization, department or project up to 25 people, dedicated to software development. The Guide provides Project Management and Software Implementation process which integrate practices based on the selection of ISO/IEC12207

- Systems and Software Engineering

–Software Life Cycle process

– guidelines for the content of software life cycle process information products (documentation) standards elements.

**4.1Project Management (PM) process**

**PM purpose**

The purpose of the Project Management Process is to establish and carry out in a systematic way the tasks of the software implementation project, which allows complying with the project’s objectives in the expected quality, time and costs.

**PM Objectives**

**PM.O1.** The Project plan for the execution of the project is developed according to the Statement of work and validates with the customer. The tasks and resources necessary to complete the work are sized and estimated.

**PM.O2.** Progress of the project is monitored against the project plan and recorded in the progress status record. Correction to remediate problems and deviations from the plan are taken when project targeted are not achieved. Appropriate treatment is taken to correct or avoid the impact of risk. Closure of the project is performed to get the Customer acceptance documented in the Acceptance record.

**PM.O3.** The Change Requests are addressed through their reception and analysis. Changes to software requirements are evaluated for cost, schedule and technical impact.

**PM.O4.** Review meeting with the work team and the customer are held. Agreements are registered and tracked.

**PM.O5.** Risks are identified as they develop and during the conduct of the project.

**PM.O6.** A software version control strategy is developed. Items of Software Configuration are identified, defined and baselined. Modifications and releases of the items are controlled and made available to the customer and work team including the storage, handling and delivery of the items.

**PM.O7.** Software Quality Assurance is performed to provide assurance the work products and processes comply with the Project Plan and Requirements Specification.

**PM Activities**

The Project management process has the following activities:

-PM.1 Project Planning

-PM.2 Project Plan Execution

-PM.3 Project Assessment and Control

-PM.4 Project Closure

**4.2 Software Implementation (SI) process**

**SI purpose**

The purpose of the Software Implementation process is the systematic performance of the analysis, design, construction, integration and tests activities for new or modified software products according to the specified requirements.

**SI objectives**

**SI.O1.** Tasks of the activities are performed through the accomplishment of the current Project Plan.

**SI.O2.** Software requirements are defined, analyzed for correctness and testability, approved by the Customer, baselined and communicated.

**SI.O3.** Software architectural and detailed design is developed and baselined. It describes the software items and internal and external interfaces of them. Consistency and traceability to software requirements are established.

**SI.O4.** Software components defined by the design are produced. Unit test are defined and performed to verify the consistency with requirements and design are established.

**SI.O5.** Software is produced performing integration of software components and verified using Test Cases and Test Procedures. Results are recorded at the Test Report. Defects are corrected and consistency and traceability to software design are established.

**SI.O6.** A Software Configuration, that meets the requirement specification as agreed to with the customer, which includes user, operation and maintenance documentations is integrated, baselined and stored at the Project Repository. Needs for changes to the Software Configuration are defected and related Change Requests are initiated.

**SI.O7.** Verification and Validation tasks of all required work products are performed using the defined criteria to achieve consistency among output and input products in each activity. Defects are identified, and corrected; records are stored in the Verification/Validation Results.

**SI Activities**

The Software Implementation Process has the following activities:

-SI.1 Software Implementation Initiation

-SI.2 Software Requirements Analysis

-SI.3 Software Architectural and Detailed Design

-SI.4 Software Construction

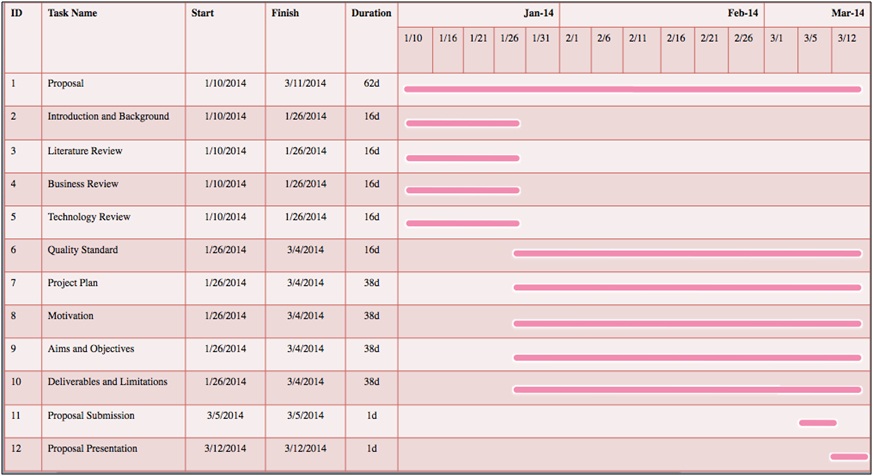
-SI.5 Software Integration and Tests

-SI.6 Product Delivery

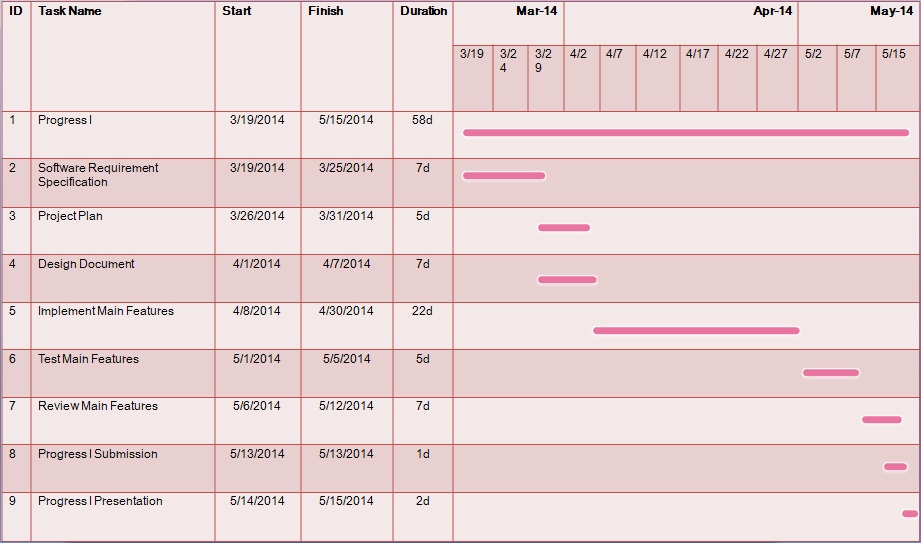
**Chapter Five**

**Estimated of Tasks**

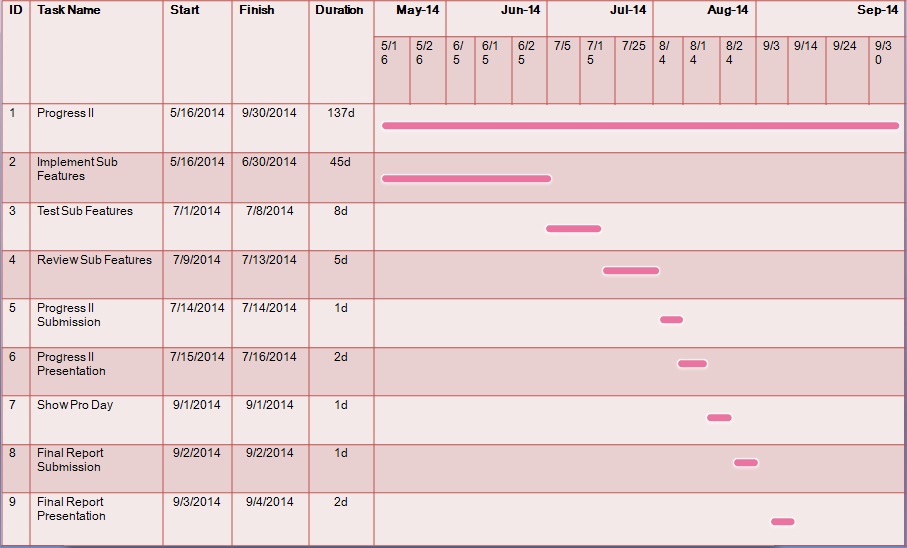
**5.Estimated Duration of Tasks**



*Figure2: Proposal*



*Figure3: Progress Report I*



*Figure4: Progress Report II & Final Progress*

**5.1Estimates Effort and Cost**

Most of the cost will be come from reference or learning textbook and hard copy document. Because this project use only open source language and freeware tool for development. So, most of the cost from this project will be used to buy some textbook and print whole documents for each progress.

**Chapter Six**

**Risks**

6.Identification of Project Risks

|  |  |
| --- | --- |
| **Risk** | **How to solve** |
| **Human Risks** | |
| Group member are lack of skills and knowledge. | Learn from textbook, website and teacher. |
|  |  |
| **Technology Risks** | |
| Server can be busy, if the software must transfer more information. | Selected the server which good performance and manage transfer information. |
| Internet can be lack on sometimes. | Set and check Internet connection or scope wifi. |
| **Process Risks** | |
| Some features of project can be changed. | Created Change request form and to the advisor for suggestion. |
| Project’s item cannot trace to its source. | Created traceability record. |
|  |  |

**Chapter Seven**

**Version Control**

**7.Version Control Strategy**

**7.1Naming Conversion**

The Chiang Mai Red Taxi Service Assistant – [File name]\_[Version].[File Format]

**7.2 Project Repository**

-Github : For updates documents.

-Google Drives : For collected all documents.

**7.3 Configuration Item Table**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Item** | **File Name** | **File Type** | **Owner** | **Path** | **Baseline Version** |
| 1 | Project Proposal | The Chiang Mai Red Taxi Service Assistant\_Proposal\_V.1.0 | .docx | Kanittee, Pimchittra | /The Chiang Mai Red Taxi Service Assistant/Proposal |  |
| 2 | Project Development and Quality Plan | The Chiang Mai Red Taxi Service Assistant\_ Project Plan\_V.1.0 | .docx | Kanittee, Pimchittra | /The Chiang Mai Red Taxi Service Assistant/Project Plan |  |
| 3 | Software Requirement Specification | The Chiang Mai Red Taxi Service Assistant\_SRS\_V.1.0 | .docx | Kanittee, Pimchittra | /The Chiang Mai Red Taxi Service Assistant/SRS |  |
| 4 | Software Design Document | The Chiang Mai Red Taxi Service Assistant\_SDD\_V.1.0 | .docx | Kanittee, Pimchittra | /The Chiang Mai Red Taxi Service Assistant/SDD |  |
| 5 | Traceability Record | The Chiang Mai Red Taxi Service Assistant\_Traceability\_Record\_V.1.0 | .docx | Kanittee, Pimchittra | /The Chiang Mai Red Taxi Service Assistant/traceability |  |
| 6 | Software Implementation | The Chiang Mai Red Taxi Service Assistant\_Code\_V.1.0 | .rar | Kanittee, Pimchittra | /The Chiang Mai Red Taxi Service Assistant/Code |  |
| 7 | Unit Test Record | The Chiang Mai Red Taxi Service Assistant\_Unit\_Test\_V.1.0 | .docx | Kanittee, Pimchittra | /The Chiang Mai Red Taxi Service Assistant/Unit test |  |
| 8 | System Test Record | The Chiang Mai Red Taxi Service Assistant\_System\_Test\_V.1.0 | .docx | Kanittee, Pimchittra | /The Chiang Mai Red Taxi Service Assistant/System test |  |