|  |  |  |
| --- | --- | --- |
| **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 | 29 May 2014 | Kanittee,Pimchittra | Kanittee, Pimchittra | Draft |
| 1.1 | 2 July 2014 | Kanittee,Pimchittra | Kanittee, Pimchittra | edit |

**Document develops by**

Kanittee Hongron

Pimchittra Sukkasem

**Document Approved by**

Ms. Pattama Longanee

**Objective:**

-To provide guidelines to prepare a minimum Project Management Plan for the projects handled within a software 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 | Project Proposal | Overview of Project. |
| Project Management Plan | Management Schedule, Quality Stand, Estimate Risks, Costs, and Configuration Management. |
| Requirements Specification | Project requirements that were analyzed from developer teams and accepted by users. |
| Traceability | Traceability Document that can trace, change and update when errors occur. |
| System Design | Design of software that is based on Algorithm, diagrams, database design and so on. |
| Implementation | Construction process, which consists of implementation and integration. |
| Testing | Document about test case for the system. |

**Chapter One | Introduction**

The Chiang Mai Red Taxi Service Assistant is an application that allows passengers to book a red taxi in Chiang Mai and select their destination from a list of popular attractions. Passengers would be able to send a request to the driver and the driver would be able to either accept or deny the request, the system would then inform the passenger of the driver’s response. The application will run on an Android platform.

**Project Overview**

The overview of the project will be a mobile application functioning on an Android platform. The main objective of the application is that the system would be able to process the request from passenger to the driver, and the driver would be able to accept or deny the request and send it back to the system.

* 1. **Purpose**

The software development plan is a document that outlines the project plan and evaluates the project. The plan would help to budget the costs involved and set goals in the project timeline. The project plan before start project and when project start, project plan used to control project follow the plan.

* 1. **Scope**

The application will be divided into two parts. The server side will be developed using PHP and the Android part will be developed using the Android API.

* 1. **Acronyms and Definition**

**Acronyms**

**UI** User Interface

**UC** User Case

**SRS** Software Requirement Specification

**AD** Activity Diagram

**SDD** Software Design Document

**URS**  User Requirement Specification

**SRS** System Requirement Specification

**SD** Sequence Diagram

**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 | 5 March 2014 |
| 2. | **The Progress Report I**  -Project Management Plan version 1.1  -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 | 7 July 2014 |
| 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 | 3 September 2014 |
| 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 | 26 November 2014 |

**Chapter Two | Infrastructure**

**2.1 Hardware/Software Acquisition Plans**

**Hardware**

-computer Lenovo 512M

-Phone Lenovo android operating system version 4.2

**Software**

**-**Android Operating System version 4.0 or 4.2

- PHP

- Java Android SDK

-JSON

**2.2 Project Specific System Support**

**Software Required:**

-PHP

-Java

-Android SDK version 4.0

-Server

**Minimum Specifications:**

-Android Version 4.0

-A stable Internet connection/3G connection

-Smart phone with Android Operating System version 4.0 or above

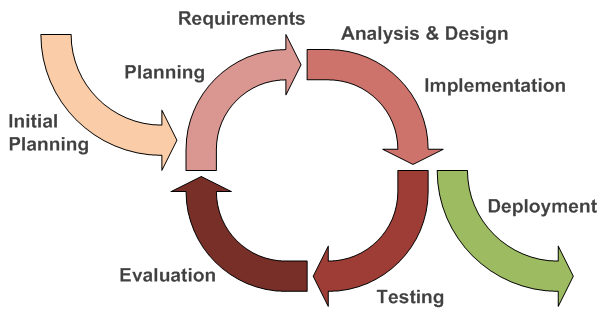
**2.3 Management Procedures**

**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 introduces the project, the technologies and tools used, quality standards and proposes the overall project plan.

**Document Plan Phase**: This document outlines the plan and designs the overall system from the requirements as stated by the user. These documents consist of the Project Management Plan, Software Requirement Specification and Software Design Document.

**Overall Iterative Features**: This phase is about separating the system into the individual features and then creating every single feature from the first till the final feature. This document will be divided into 4 parts:

- Plan: Planning the method of creating and testing each feature.

- Implement: Implement and coding each feature.

- Test: Test and debug each feature.

- Review: Review and maintain each feature to match the features in the plan.

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

**Deployment phase**: This phase is about deploying the whole system to the server and use the application it as a regular mobile application.

**2.4.2 Status Reporting**

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Progress Report** | **Software items** | **Date** |
| 1. | Progress I | -Project Management Plan v.1.1  -Software Requirement Specification v.1.0  -Software Design Document v.1.0  -Main Features (Feature#1-13)  -Traceability Record I v.1.0  -Test Plan v.1.0  -Test record v.1.0 | 7 July 2014 |
| 2. | Progress II | -Main features (Feature#1-13)  -Traceability Record II v.2.0  -Test Plan v.2.0  -Test Record v.2.0 | 3 September 2014 |
| 3. | Progress ShowPro | -Sub features (Feature#14-19)  -Review the features  -Traceability Record ShowPro  -Test Plan v.3.0  -Test Record v.3.0 | 5 November 2014 |
| 4. | Final Progress | -Final Document v.4.0  -Final System v.4.0 | 26 November 2014 |

**Features**

**Main Features**

-#1 Passengers can register themselves to the system.

-#2 Passengers can log in to the system.

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

-#4 Passengers can send a request to a driver.

-#5 Passengers can get the confirmation if driver accepts the request.

-#6 Passengers can see booking information of red taxi.

-#7 Passengers can view the current location of the red taxi matching their conditions.  
-#8 Drivers can register themselves to the system.

-#9 Drivers can log in to the system.

-#10 Drivers can update the number of available seats.

-#11 Drivers can tell where the taxi is heading.

-#12 Drivers can get the request from passenger.

-#13 Drivers can either accept or decline the request.

**Sub Features**

-#14: Passengers can cancel the request.

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

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

-#17: Passengers can change the scope of searching for red taxis.

-#18: Drivers can receive the cancellation request.

-#19: Drivers can choose to enable or disable the service (E.g. off duty).

**2.5 Change Management**

**Change Control Procedures**

1. State the change.
2. Analyze the reason for the change.
3. Send change form to Project advisor.
   1. If accepted: Make a change in the project from the change request form.
   2. If not accepted: Resume the project and find a solution to the problem.
4. Analyze the results from changing and modifying the 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 factors after completion:

**3.1.1 Product Operation factors**

**Correctness**

-The System should be able to meet the full requirements of displaying and tracking the red taxi on the mobile application.

-The information in the System should be updated regularly.

**Reliability**

-The System should have a failure rate of lower than 10% after deployed to server.

-The System should transfer information via the server.

**Integrity**

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

**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 supported using an Internet connection or 3G.

**Reusability**

-The software should be developed and able to function on a Windows phone or Apple 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.

**4.1 Project 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 complies with the project’s objectives in the area of 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 meetings 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 course 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 with the customer, which includes user, operation and maintenance documentations is integrated, baselined and stored in 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

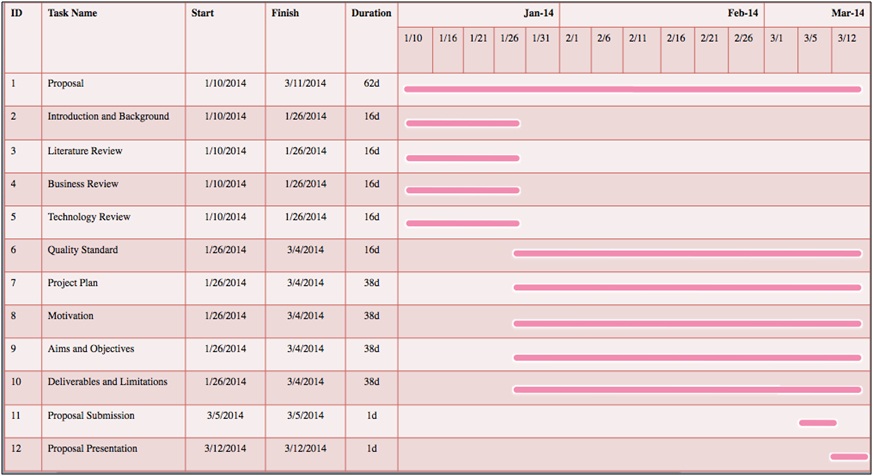
-SI.5 Software Integration and Tests

-SI.6 Product Delivery

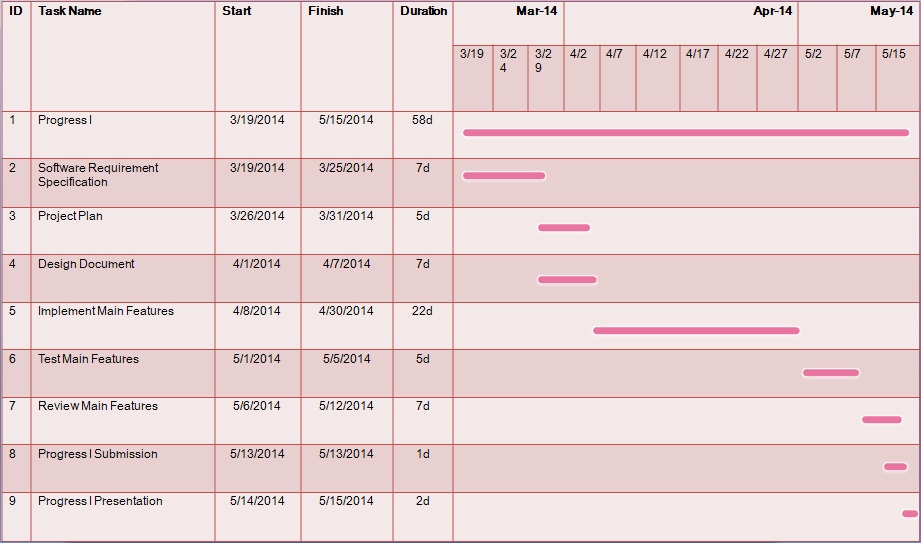
**Chapter Five | Schedule and Milestones**

**Estimated Duration of Tasks**

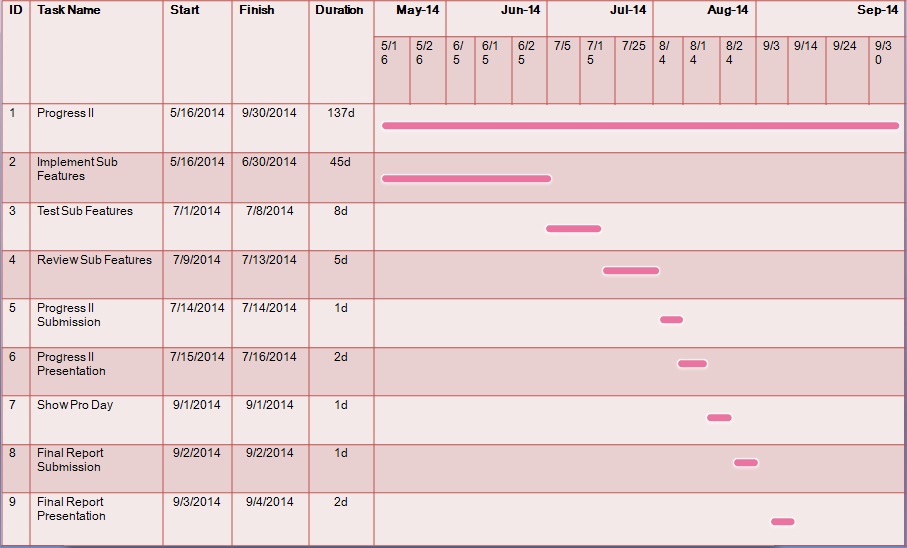
|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Progress** | **Features** | **Submission** |
| 1. | **Progress I** | **Features #1 to #13**  **(URS-01 to URS-20, SRS-01 to SRS-74)**  #1 Passengers can register themselves to the system.  #2 Passengers can log in to the system.  #3 Passengers can set the destination and the number of passengers.  #4 Passengers can send a request to a driver.  #5 Passengers can get the confirmation if driver accepts the request.  #6 Passengers can see booking information of red taxi.  #7 Passengers can view the current location of the red taxi matching their conditions.  #8 Drivers can register themselves to the system.  #9 Drivers can log in to the system.  #10 Drivers can update the number of available seats.  #11 Drivers can tell where the taxi is heading.  #12 Drivers can get the request from passenger.  #13 Drivers can either accept or decline the request. | **7 July 2014** |
| 2. | **Progress II** | **Features #14 to #19**  #14 Passengers can cancel the request.  #15 Passengers can get the notification when the red taxi arrives.  #16 Passengers can create schedules (for planning of routes).  #17 Passengers can change the scope of searching for red taxis.  #18 Drivers can receive the cancellation request.  #19 Drivers can choose to enable or disable the service (E.g. off duty). | **3 September 2014** |
| 3. | **Final Progress** | **Final version of system and documents**  **Features #1 to #19** | **26 November 2014** |



*Figure2: Proposal*



*Figure3: Progress Report I*



*Figure4: Progress Report II & Final Progress*

**Estimates Effort and Cost**

Most of the cost will come from referencing articles, textbooks or previous research available. This project only uses open source language and freeware tools for its development. So, most of the budget for this project will be used to buy some textbooks and print whole the documents for each progress.

**Chapter Six | Risks Management**

Risk management is concerned with identifying risks and drawing up plan to minimize their effect on project.

A risk is probability that some adverse circumstance will occur.

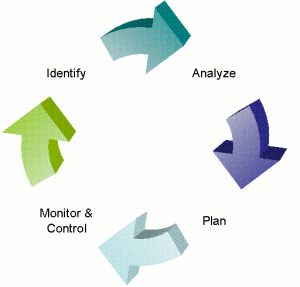
-Project Risks affect schedule or resources.

-Product Risks affect the quality or performance of the software being developed.

-Business risks affect the project team during developing or procuring the software.

Identified risks when start project and start development phase. All identified risks are documented and assessed in the Risk Management Process by the Project Team. In the Risk Management Process defined the possible risks and solution of them, and who is responsible for.

**6.1 Risk Management Process**



*Figure5: Risk Management Process*

1.Risk identification : To identify project, product and business risks.

2.Risk analysis : To assess the likelihood and consequences of the risks.

3.Risk planning : Draw up plans to avoid minimize the effects of the risks.

4.Risk monitoring : Monitor the risks throughout the project.

**6.2 Identification of Project Risks**

|  |  |
| --- | --- |
| **Risk statement** | **Risk Solution** |
| **Human Risks** | |
| Group members lack skills and knowledge. | -Learn from textbooks, reference to websites and seeking help from teacher. |
| Team members may be lack of knowledge or skills and cannot finish tasks in time. | -Discuss between team members and assign work more suitable for each member’s knowledge and skills.  -Find information and help both online, offline and consult with people who have experience. |
| **Technology Risks** | |
| Server can be busy, if the software must transfer more information. | -Selected the server which good performance and manage transfer information. |
| Internet connection can be unstable sometimes. | -Set and check Internet connection or scope Wi-Fi. |
| Technology that is used for the project may have problem. (e.g. Google map version 3 is slow and unstable.) | -Analyze and search for the solution to the problems. Try to find alternatives when necessary. (e.g. Check the status of Google map with related implementation.) |
| **Process Risks** | |
| Some features of project can be changed. | -Create Change request form and to the advisor for suggestion. |
| Project’s item cannot trace to its source. | -Create traceability record. |
| The requirements might change. | -Hold regular meeting and plan for the possible changes.  -Discuss with stakeholder involved and set the priority of changed requirement.  -Update the system with changed requirements and also other requirements related. |
| The project might not be done in time. | -Review the progress on a regular basis and identify the tasks that are likely to cause a delay. Monitor those tasks closely.  -Plan and schedule the progress and discuss between team members and advisor. |

**Chapter Seven | Software Configuration Management**

**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 contains all documents, codes, and any supporting materials .

**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 | 1.0 |
| 2 | Project Development and Quality Plan | The Chiang Mai Red Taxi Service Assistant\_ Project Plan\_V.1.1 | .docx | Kanittee, Pimchittra | /The Chiang Mai Red Taxi Service Assistant/Project Plan | 1.1 |
| 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 | 1.0 |
| 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 | 1.0 |
| 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 | 1.0 |
| 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 | 1.0 |
| 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 | 1.0 |
| 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 | 1.0 |