**Smart course Management**

**System in cloud**

Project Plan

By

**Mr. Chaichan Suttee 542115016**

**Mr. Tanadol Parn-ong 542115021**

Department of Software Engineering

College of Arts, Media and Technology

Chiang Mai University

Project Advisor

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Mrs. Yun Rim Park**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Document Name** | **Version** | **Status** | **Date** | **Viewable** | **Reviewer** | **Responsible** |
| **Documents** | | | | | | |
| Smart course management system in the cloud – Project plan\_v1.0.docx | Create  Chapter One – Introduction  Chapter Two– Infrastructure  Chapter Three – Quality Planning  Chapter Four – Software Standard  Chapter Five – Estimated of Tasks  Chapter Six – Risks  Chapter Seven – Version Control | Draft | 17 May 2014 | CS, TP, YP | CS,TP | TP |

**Document History**

**\*CS = Chaichan Suttee**

**\*TP = Tanadol Parn-ong**

**\*YP = Yun Rim Park**

Contents

[**Chapter One | Introduction** 4](#_Toc388121629)

[**1.1** **Project Overview** 4](#_Toc388121630)

[**1.2** **Acronyms and Definition** 5](#_Toc388121631)

[**1.3** **Work Product to be developed** 8](#_Toc388121632)

[**Chapter Two | Infrastructure** 10](#_Toc388121633)

[**2.1** **Hardware/Software Acquisition Plans** 10](#_Toc388121634)

[**2.2** **Management Procedure** 11](#_Toc388121635)

[**2.3** **Monitoring and Controlling Mechanisms** 12](#_Toc388121636)

[**2.4** **Change Management** 17](#_Toc388121637)

[**Chapter Three | Quality Planning** 18](#_Toc388121638)

[**3.1** **Quality Factors** 18](#_Toc388121639)

[**3.2** **Reviews/Responsibility** 19](#_Toc388121640)

[**3.3** **Testing** 20](#_Toc388121641)

[**Chapter Four | Software Standard** 20](#_Toc388121642)

[**4.1 Project Management (PM) process** 20](#_Toc388121643)

[**4.2 Software Implementation (SI) process** 22](#_Toc388121644)

[**Chapter Five | Estimated of Tasks** 24](#_Toc388121645)

[**5.1 Estimated Duration of Tasks** 24](#_Toc388121646)

[**5.2 Estimated Effort and Cost** 25](#_Toc388121647)

[**Chapter Six | Risks** 25](#_Toc388121648)

[**Chapter Seven | Version Control** 26](#_Toc388121649)

[**7.1 Naming Conversion** 26](#_Toc388121650)

[**7.2 Project Repository** 26](#_Toc388121651)

[**7.3 Configuration Item Table** 27](#_Toc388121652)

# **Chapter One | Introduction**

Smart course management system in the cloud is a system that helps to manage score, upload/download course materials, share files and assign quizzes which provides convenience to both lecturers and students. Maintaining course materials in the cloud storage, lecturers can manage scores, upload materials, and create assignments, and students can download the materials, take the assignments, and view the scores. Smart course management system in the cloud is also a tool for students and lecturers conveniently interact in studying together.

## **Project Overview**

The main objectives are a development Smart course management system in the cloud that supports both smartphone and personal computer in order to provide convenient to lectures and student. This designed system integrates the technologies of web service and iOS technology to access to the cloud storage. Such the system will provide convenience about managing the score, uploading/downloading course materials, assigning tests for lecturers and also student can take tests, view the score, download the course materials, and receive notification from lecturers.

* + 1. **Purpose**

The software project management plan is a document for plan the project, evaluating project. The planning can reduce the cost from project. The project plan is a plan before start project and when project start, project plan is used to tracking and control project follow the plan.

* + 1. **Scope**

The scopes of this project plan and quality plan are:

* For managing the schedule to develop Smart course management system in the cloud.
* For managing the risk those can occur during the development.
* Specify the tools those used for developing the project.
* Specify the developers who have responsible to develop the project.
* Specify the deliverable items in each phase.
* Describe the quality plan of this project.

## **Acronyms and Definition**

**Acronyms**

AC Activity diagram

SDD Software design document

SPMP Software project management plan

SRS Software requirement specification

URS User requirement specification

UC Use case

UI User interface

CSS Cascading style sheets

CD Class diagram

SD Sequence diagram

STC System test case

UTC Unit test case

MDA Method design Attribute

MDM Method design method

**Definition**

**IEEE**

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

**UML Unified 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 programs or modules in order to remove a design or programming errors. [IEEE90]

## **Work Product to be developed**

* + 1. **Deliverables**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Deliverables/Release** | **Media** | **No. of**  **Copies** | **Data** |
| 1. | **The Proposal report**  Project Proposal version 3.0 | Hard copy | 1 | 26/03/2013 |
| 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 * Test record version 1.0 * Traceability record version 1.0 | Hard copy | 3 | 28/05/2013 |
| 3. | **The Progress report II**   * Project Management Plan version 2.0 * Software Requirement Specification version 2.0 * Software Design Document version 2.0 * version 2.0 * Test Record version 2.0 * Traceability record version 2.0 | Hard copy | 3 | 30/07/2013 |
| 4. | **The Progress report III**   * 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 | 30/09/2013 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Deliverables/Release** | **Media** | **No. of**  **Copies** | **Data** |
| 5. | **The Final progress report**   * Project Management Plan version 4.0 * Software Requirement Specification version 4.0 * Software Design Document version 4.0 * Test Plan version 4.0 * Test Record version 4.0 * Traceability record version 4.0 * Video clips for demo program | Hard copy  .mp4 file | 3 | 31/10/2013 |

# **Chapter Two | Infrastructure**

## **Hardware/Software Acquisition Plans**

* + 1. **Testing tools**
* **Hardware**
* iPhone 5 with iOS version 7.1.1
* Macbook white 2010 with OS X version 10.9.2
* Toshiba notebook L840 Series with Window 7 Ultimate sp1
* **Software**
* Firefox 29.0.1
* iOS 7.1.1
  + 1. **Design tools**
* Adobe Photoshop
  + 1. **Development tools**
* PhoneGap
* Microsoft visual studio
* Internet Information Service (IIS)
* Microsoft SQL Server
* Window Azure
  + 1. **Configuration management tools**
* GitHub
  + 1. **Document tools**
* Microsoft Word2013
* Visual Paradigm
* GanttProject

## **Management Procedure**

* + 1. **Project Team Structure**

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Participants | Roles | Responsibility |
| 1. | Mr.Chaichan Suttee | System Analysis, Designer, Programmer, Tester | - Project Proposal  - Software Requirement Specification in part of web application  - Software Design Document in part of web application  - Software coding in part of web application  - Test Plan document in part of web application  - Test Record documentin part of web application  - Traceability record document in part of web application |
| 2. | Mr.Tanadol Parn-ong | System Analysis, Designer, Programmer, Tester | - Project Proposal  - Project Management Plan  - Software Requirement Specification in part of mobile application  - Software Design Document in part of mobile application  - Software coding in part of mobile application  - Test Plan document in part of mobile application  - Test Record document in part of mobile application  - Traceability record document in part of mobile application |

## **Monitoring and Controlling Mechanisms**

* + 1. **Software Development Model**

Iterative development model

Iterative development model is one of the software development models which evolves from waterfall model. By changing process flow from step to step into iterative step. When the process flows into iterative, the process will start from the first step then go to the next step till the last. After that, the process will back to the first step and start again. The iteration will be repeat until all processes planned are complete then out from the loop and go to next main phase.

**Proposal phase:** This phase is about creating a proposal for Smart course management system in the cloud.

**Document plan phase:** This phase is about document for planning and designs the overall system from requirement given by the user. 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:** Implementing and coding each feature.

**- Test:** Testing and fixing each feature.

**- Review:** Reviewing and maintaining each feature to meet the feature plan.

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

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

* + 1. **Status Reporting**

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Progress Report** | **Software items** | **Date** |
| 1. | Progress I | * Project Management Plan version 1.0 * Software Requirement Specification version 1.0(Feature 1,2,3,4,5) * Software Design Document version 1.0(Feature 1,2,3,4,5) * Test Plan version 1.0(Feature 1,2,3,4,5) * Test Record version 1.0 (Feature 1,2,3,4,5) * Traceability record version 1.0 | 28/052014 |

**Features**

**Feature#1 User registration**

**Description:** This feature supports registration of users for using the system.

**Users:** Lecturer, Student, Administrator

**Details:**

**1-1 Lecturer/Student:** Lecturer/Student can request to register in the system.

**1-2 Administrator:** Administrator can approve registration requests from Lecturers and Students.

**Feature#2 Login to the system**

**Description:** This feature supports access to the system.

**Users:** Lecturer, Student, Administrator

**Details:**

**2-1 Lecturer/Student/Administrator:** User can login to the system.

**2-2 Lecturer/Student/Administrator:** User can logout from the system.

**Feature#3 Course management**

**Description:** This feature supports managing semesters to respond academic year that change over time by Administrator. Lecturer manages his/her courses offered in each semester. Student accesses course information registered.

**Users:** Administrator, Lecturer, Student

**Details:**

**3-1 Administrator:** Administrator can create, edit, and delete semesters.

**3-2 Administrator:** Administrator can create, edit, and delete a course in the semester.

**3-3 Administrator:** Administrator can register a list of students to a course.

**3-4 Lecturer:** Lecturer can access information of his/her course.

**3-5 Student:** Student can see access information of his/her course registered.

**Feature#4 Assignment and Quiz management**

**Description:** This feature supports assignment and quiz management by Lecturer. This feature can process Student’s answer to each question of an assignment or quiz and compute the score. Moreover, this feature supports preventing Student from copying assignments or quizzes which are multiple choices and true/ false questions. The system can randomize the order of questions.

**Users:** Lecturer

**Details:**

**4-1 Lecturer:** Lecturer can assign, edit and delete assignments or quizzes which are multiple choices, true/false questions, short answers.

**4-2 Lecturer:** Lecturer can limit access to the assignment or quiz of authority.

**4-3 Lecturer**: Lecturer can review overall student’s answer in each choice of assignment or quiz in chart format.

**4-4 Lecturer:** Lecturer can post an assignment or quizzes.

**4-5 Lecturer:** Lecturer can check whether the assignments are submitted in time.

**4-6 Lecturer:** Lecturer can randomize the order of questions in an assignment or a quiz.

**Feature#5 Taking assignments and quizzes**

**Description:** This feature supports taking assignments and quizzes by student.

**Users:** Student

**Details:**

**5-1 Student:** Student can take assignments or quizzes which are multiple choices, true/false questions, short answers.

## **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 change in project from change request form.
   2. If reject: Continue in the project and find the way to solve a problem.
4. Analyze the result from changing and modify the document or system to match with change.

# **Chapter Three | Quality Planning**

**Quality Planning [V&V]**

## **Quality Factors**

According to McCall’s factor model, Smart course management system in the cloud should meet these quality factor after complete;

* + 1. **Product operating factors**
* **Correctness**
* The software product should able to provide 100% correctness of data from user request.
* The information in the system should be up to date at all time.
* **Reliability**
* The system should have a failure rate lower than 10% after deployed to the cloud server.
* **Integrity**
* The system should able to identify the authentication of every user.
* The system should able to limit some features and pages from unauthorized user.
  + 1. **Product revision factors**
* **Maintainability**
* The software should have at least 20% of comment comparing with the whole Line of code to support the future maintenance activity.
* The software should return measurable output for future maintenance activity.
* **Testability**
* The software should able to be tested 100% of all feature and flow of processes.
  + 1. **Product transition factors**
* **Portability**
* The system should able to support Firefox internet browser and iPhone 5 with iOS 7.1.1.
* **Reusability**
* More than 20% part of finished software product should able to be reused in future development.

## **Reviews/Responsibility**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Stage Exit Review** | | | | |
| **No.** | **Stage** | **Review Item** | **Responsibility** | **Reviewer** |
| 1. | Proposal | Project proposal version 3.0 | CS,TP | CS,TP,YP |
| 2. | Progress I | Project Management Plan version 1.0 | TP | CS,TP,YP |
| 3. | Progress I | Software Requirement Specification version 1.0 | CS,TP | CS,TP,YP |
| 4. | Progress I | Software Design Document version 1.0 | CS,TP | CS,TP,YP |
| 5. | Progress I | Test Plan version 1.0 | CS,TP | CS,TP,YP |
| 6. | Progress I | Test Record version 1.0 | CS,TP | CS,TP,YP |
| 7. | Progress I | Traceability record version 1.0 | CS,TP | CS,TP,YP |
| 8. | Progress II | Software Requirement Specification version 2.0 | CS,TP | CS,TP,YP |
| 9. | Progress II | Software Design Document version 2.0 | CS,TP | CS,TP,YP |
| 10. | Progress II | Test Plan version 2.0 | CS,TP | CS,TP,YP |
| 11. | Progress II | Test Record version 2.0 | CS,TP | CS,TP,YP |
| 12. | Progress II | Traceability record version 2.0 | CS,TP | CS,TP,YP |
| 13. | Progress III | Software Requirement Specification version 3.0 | CS,TP | CS,TP,YP |
| 14. | Progress III | Software Design Document version 3.0 | CS,TP | CS,TP,YP |
| 15. | Progress III | Test Plan version 3.0 | CS,TP | CS,TP,YP |
| 16. | Progress III | Test Record version 3.0 | CS,TP | CS,TP,YP |
| 17. | Progress III | Traceability record version 3.0 | CS,TP | CS,TP,YP |
| 18. | Final progress | Software Requirement Specification version 4.0 | CS,TP | CS,TP,YP |
| 19. | Final progress | Software Design Document version 4.0 | CS,TP | CS,TP,YP |
| 20. | Final progress | Test Plan version 4.0 | CS,TP | CS,TP,YP |
| 21. | Final progress | Test Record version 4.0 | CS,TP | CS,TP,YP |
| 22. | Final progress | Traceability record version 4.0 | CS,TP | CS,TP,YP |

## **Testing**

|  |  |  |
| --- | --- | --- |
| **Testing Process** | | |
| **No.** | **Test** | **Responsibility** |
| 1. | Unit Testing | CS,TP |
| 2. | System Testing | CS,TP |

# **Chapter Four | Software Standard**

**Software Development Standard**

**ISO29110 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 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 validated 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. Corrections to remediate problems and deviations from the plan are taken when project targets 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 conduct of the project.

**PM.O6:** A software Version Control Strategy is developed. Items of Software Configuration are identified, defined and base lined. 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 that 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, base lined and communicated.

**SI.O3:** Software architectural and detailed design is developed and base lined. 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 the design. Traceability to the 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 Requirements Specification as agreed to with the Customer, which includes user, operation and maintenance documentations is integrated, base lined and stored at the Project Repository. Needs for changes to the Software Configuration are detected 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**

## **5.1 Estimated Duration of Tasks**

|  |  |  |
| --- | --- | --- |
| **Task and Estimated Duration** | | |
| **No.** | **Phase** | **Estimated Duration (Days)** |
| 1. | Proposal | 35 |
| 2. | Progress I | 63 |
| 3. | Progress II | 63 |
| 4. | Progress III | 62 |
| 5. | Final progress | 31 |
|  | Total | 254 |

## **5.2 Estimated Effort and Cost**

Smart course management system in the cloud is supported by College of Arts, Media and Technology, Chiang Mai University as a senior project. This project is limited supporting. The faculty will provide 500 baths for poster presentation, and will provide hardware for development base on request and other justification.

# **Chapter Six | Risks**

**Identification of Project Risks**

|  |  |
| --- | --- |
| **Risk** | **Solution** |
| **Human Risks** | |
| Group members are lack in skill and knowledge. | - Lean from a textbook and websites.  - Asking about solutions in group member together. |
| One of group members has a lot of duty to do | - Separation of works should have equal works in each group members |
| There is bad communication in group members. | - Try to more communicate in group members |
| **Process Risks** | |
| Scope of the project should be changed. | - Team project advises with project’s advisor for suggestion.  - Create a change request document and then sent to project’s advisor. |
| Project’s item cannot trace to its source. | - Create traceability record documents. |

# **Chapter Seven | Version Control**

## **7.1 Naming Conversion**

Smart course management system in the cloud – [File name]\_[Version].[File format]

## **7.2 Project Repository**

- GitHub: For keep all file related to project and source codes.

## **7.3 Configuration Item Table**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Item** | | | **File Name** | **File Type** | **Owner** | **Path** | **Baseline Version** |
| 1. | Project Proposal | | | Smart course management system in the cloud – Proposal\_v3.0.docx | .docx | CS,TP | /Smart-Course-Management-system-in-cloud/Proposal | 3.0 |
| 2. | Project Management Plan | | | Smart course management system in the cloud – Project plan\_v4.0.docx | .docx | CS,TP | /Smart-Course-Management-system-in-cloud/ Project Plan | 4.0 |
| 3. | Software Requirement Specification | | | Smart course management system in the cloud – Requirement\_v4.0.docx | .docx | CS,TP | /Smart-Course-Management-system-in-cloud/ Requirement | 4.0 |
| 4. | Software Design Document | | | Smart course management system in the cloud – Design \_v4.0.docx | .docx | CS,TP | /Smart-Course-Management-system-in-cloud/Design | 4.0 |
| 5. | Test Plan | | | Smart course management system in the cloud – Test Plan \_v4.0.docx | .docx | CS,TP | /Smart-Course-Management-system-in-cloud/Test | 4.0 |
| **No.** | | **Item** | **File Name** | | **File Type** | **Owner** | **Path** | **Baseline Version** |
| 6. | | Test Record | Smart course management system in the cloud – Test Record \_v4.0.docx | | .docx | CS,TP | /Smart-Course-Management-system-in-cloud/Test | 4.0 |
| 7. | | Traceability Record | Smart course management system in the cloud – Traceability \_v4.0.docx | | .docx | CS,TP | /Smart-Course-Management-system-in-cloud/ Traceability | 4.0 |
| 8. | | Software code | Smart course management system in the cloud – code \_v4.0.docx | | .rar | CS,TP | /Smart-Course-Management-system-in-cloud/code | 4.0 |