**Project plan**

Admission system

**Contents**

[**List of table** 2](#_Toc374433955)

[**1.** **Revision** 3](#_Toc374433956)

[**2.** **Introduction** 4](#_Toc374433957)

[2.1. Purpose 4](#_Toc374433958)

[2.2. Project scope 4](#_Toc374433959)

[2.3. Project deliverables 4](#_Toc374433960)

[2.4. The implementation of project management 4](#_Toc374433961)

[2.5. Reference 4](#_Toc374433962)

[2.6. Definition and acronyms 4](#_Toc374433963)

[**3.** **Project Organization** 5](#_Toc374433964)

[3.1. Process model 5](#_Toc374433965)

[3.2. Description 5](#_Toc374433966)

[3.3. Mapping of Scrum and ACDM activities 5](#_Toc374433967)

[3.4. Role and responsibility 6](#_Toc374433968)

[3.5. Mapping role in Scrum and ACDM 9](#_Toc374433969)

[3.6. Team structure 9](#_Toc374433970)

[**4.** **Managerial Process** 10](#_Toc374433971)

[4.1. Management Objectives & Priorities 10](#_Toc374433972)

[4.2. Assumptions, Dependencies & Constraints 10](#_Toc374433973)

[4.3. Risk management plan 10](#_Toc374433974)

[4.4. Change management plan 10](#_Toc374433975)

[4.5. Configuration management plan 10](#_Toc374433976)

[4.6. Measurement plan 10](#_Toc374433977)

[4.7. Communication plan 10](#_Toc374433978)

[4.8. Architecture and design plan 10](#_Toc374433979)

[4.9. Implement plan 10](#_Toc374433980)

[4.10. Test plan 10](#_Toc374433981)

[4.11. Team charter 10](#_Toc374433982)

[4.12. Master Plans 10](#_Toc374433983)

[**5.** **Technical Process** 11](#_Toc374433984)

[5.1. Methods, Tools and Techniques 11](#_Toc374433985)

[5.2. Software documentation 11](#_Toc374433986)

[**6.** **Work Packages, Schedule** 12](#_Toc374433987)

[6.1. Work Packages 12](#_Toc374433988)

[6.2. Milestone 12](#_Toc374433989)

[6.3. Schedule 12](#_Toc374433990)

# **List of table**

[Table 1: Revision history 3](#_Toc374433944)

[Table 2: Definition and acronyms 4](#_Toc374433945)

[Table 3: Admission project development process description 5](#_Toc374433946)

[Table 4: A mapping of ACDM and Scrum activities 6](#_Toc374433947)

[Table 5: Role and responsibility in Admission project development process 7](#_Toc374433948)

[Table 6: Mapping role in Scrum and ACDM 9](#_Toc374433949)

[Table 7:Team structure 9](#_Toc374433950)

# **Revision**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Version** | **Update date** | **Author** | **Content** |
| 1 | 1.0 | 07/11/2013 | Le Ngoc Chau | Create document |
| 2 | 1.1 | 25/11/2013 | Le Ngoc Chau | Update Introduction |
| 3 | 1.2 | 10/12/2013 | Le Ngoc Chau | Update a mapping role ACDM and Scrum |

Table 1: Revision history

# **Introduction**

## 2.1. Purpose

Develope module that allows Infomation management centre can manage admission system easily, include:

- Compose information tool

- Category tool

- Ask and answer tool

Beside that, development an application that supports mobile device which use android from spring core framework of java

## 2.2. Project scope

The module is used by Infomation management centre of Van Lang University.

* Compose information tool shows the latest news of website tuyensinh.vanlanguni.edu.vn.
* Category tool is installed dynamic items by users
* Ask and answer tool will help user search title of their questions, if don’t have title that they want to search, they can make new title and send their question to server.

## Project deliverables

* Architectural driver document
* Architectural design document
* Detail design document
* Test document
* Install and user guide document

## The implementation of project management

* All changes must be approved from the Change control board before going into practice.
* All changes must be documented and updated continuously.

## Reference

* K14T, K15T graduation projects
* Technical reading software project management
* Final project of SPM course - FGS Logistic

## 2.6. Definition and acronyms

|  |  |  |
| --- | --- | --- |
| **No** | **Acronyms** | **Description** |
| 1 | UC | Use case |
| 2 | WBS | Work breakdown structure |
| 3 | PM | Project manager. |

Table 2: Definition and acronyms

# **Project Organization**

## Process model



Figure 1: Admission system development process

## Description

|  |  |  |
| --- | --- | --- |
| **No** | **Step** | **Description** |
| 1 | ACDM | * Ref 3.3. A Mapping of ACDM and Scrum activities |
| 2 | Sprint planning meeting | * Scrum master creates a release plan - A very high-level plan for multiple sprints. |
| 3 | Review and choose story items and pull it into sprint backlog | * The (estimated) velocity of the scrum team * The SM pulls the first PBI that needs estimating and gives a verbal description of the PBI to the Development Team * Development team analyzes the relative “bigness” of a product backlog items by gut feel and come to a consensus * Development team discusses differences in the numbers assigned to each PBI, and then vote again |
| 4 | Daily meeting and review | * Only development team member can actively talk during this meeting, until the very end and the scrum master keeps the meeting on track ensures that any discussions don’t go too far outside these constraints |
| 5 | Implement sprint | * Implement detail design * Implement task and unit test * Update daily report |
| 6 | Review sprint | * Development Team demonstrates sprint accomplishments to the attendees and tester will run User Acceptance Test case with customer. |
| 7 | Release sprint product | * Release install and user guide document |
| 8 | Pre- next sprint | * Repair and decision to choose story item for next sprint |

Table 3: Admission project development process description

## Mapping of Scrum and ACDM activities

|  |  |  |
| --- | --- | --- |
| **Sprint Type** | **ACDM Activity or Stage** | **Scrum Activity** |
| Pre- Sprint | Stage 1 | Develop stories |
| Pre-Sprint | Stage 2 | Create product backlog |
| Pre- Sprint | Stage 1 and Stage 2 | Prioritize the project backlog |
| Design Sprint 1 | Stage 3 Scrum team creates architecture design. | No equivalent Scrum activity |
| Stage 4: Evaluate the architecture design with the product owner |  |
| Design Sprint 2 | Stage 5: Plan experiments for next Design Sprint | Sprint planning meeting |
| Stage 6: Conduct experiments and review results with product owners | Sprint review meeting |
| Sprint retrospective meeting |  |  |
| N/A | Stage 7: Plan the construction of the system or product based on the architecture design | Create product backlog |
| Scrum Sprints | Stage 8: Production (repeated in successive Sprints) | * Sprint planning meeting traditional Scrum Sprint * Sprint review meeting * Sprint retrospective meeting |

Table 4: A mapping of ACDM and Scrum activities

## Role and responsibility

|  |  |  |
| --- | --- | --- |
| **No** | **Role** | **Responsibility** |
| 1 | Product owner | This is normally a person in the division who is responsible for feature prioritization and overall acceptance of developed code. |
| 2 | Scrum master | The scrum master is a certified scrum master who oversees and ensures compliance with the admission project development process. The scrum master takes ownership for manage all product backlog items, removes impediments, and ensures proper estimation of each product backlog item. The scrum master is an expert and works directly with the product owner. |
| 3 | Development team | This is team that include persons who implement sprint to release sprint product in the end of sprint |
| 4 | Managing engineer | The managing engineer is responsible for coordinating the overall system design and development effort.  They must be capable engineers and architects and able to confidently lead the design team without regularly resorting to draconian authoritarian means for leading the team. Such tactics do not work when leading talented engineers—the best engineers will simply leave under these conditions.  The managing engineer will plan, coordinate, track, and direct the overall activities of the design team. He or she is responsible for creating and maintaining the programmatic plans and schedules in both the period of uncertainty and the period of certainty. |
| 5 | Requirement engineer | The requirements engineer leads the effort to gather and document the architectural drivers. He or she will also help to manage the change and evolution of the architectural drivers—preferably throughout the system or product life cycle. The requirements engineer will also serve as the primary customer liaison. It is essential that the designing and developing organization presents a single, united face to the customer. Nothing can be more frustrating to customers than constantly receiving incongruent messages from the organization designing and building their product. The requirements engineer will also assist the quality engineer in coordinating architecture design review and in defining “black box” system or product tests. This testing will be explained in detail later, but in short, blackbox element and systemic tests are derived directly from the architectural drivers without any insight into the underlying implementation. |
| 6 | Chief Architecture | The chief architect is responsible for overall system design. He or she will work with all of the other members of the design team to coordinate the system design, beginning with gathering the architectural drivers, designing the architecture, reviewing it,  refining it, and documenting it until production and deployment—preferably throughout the system or product life cycle. The chief architect can provide enormous value throughout the system or product life cycle in managing change and evolution.  The chief architect is responsible for coordinating the creation and maintenance of the architecture design documentation. |
| 7 | Chief Scientist | The chief scientist is the project technologist and is primarily responsible for coordinating the planning, tracking, and documentation of experiments that are used to refine the architecture design. The chief scientist assists the architect with detailed technical issues concerning architectural design. In addition to a technical focus, the chief scientist assists the quality engineer in the architectural design reviews and in the development of “clear box” tests. |
| 8 | Support engineer | The support engineer is responsible for setting up and maintaining the design team’s support tools and environments. He or she may also be responsible for the system or product infrastructure or environment. As such, support engineers will play a key role in the design of the system from a physical perspective |
| 9 | Quality process engineer | The quality process engineer ensures that ACDM and other  defined processes are followed as prescribed to ascertain project quality goals are met. The quality process engineer is responsible for coordinating architecture design reviews as well as product test development, planning, and execution. The quality process engineer will work with the requirements engineer and the chief scientist to coordinate the architecture design reviews and in planning product or system tests. During architectural reviews, the quality process engineer is responsible for capturing, documenting, and tracking architectural issues uncovered during architectural evaluation, and that they are addressed and closed. The quality process engineer will also work with the team to establish the processes for configuration management, defect tracking, and so forth that the design team uses. These processes may also be used by the detailed designers or implementers throughout the production stage and for the life cycle of the system or product as well. |
| 10 | Product engineer | These are team members whose focus is on detailed design, implementation of the architectural elements, and integration of the elements to compose the system. |

Table 5: Role and responsibility in Admission project development process

## Mapping role in Scrum and ACDM

|  |  |
| --- | --- |
| **Scrum** | **ACDM** |
| Scrum master | Managing engineer |
| Product owners | Requirement engineer |
| Development team | Chief architecture |
| Chief scientist |
| Support engineer |
| Quality process engineer |
| Product engineer |

Table 6: Mapping role in Scrum and ACDM

## Team structure

|  |  |  |
| --- | --- | --- |
| **No** | **Team member** | **Roles** |
| 1 | Le Ngoc Chau | Development team |
| 2 | Khau Thanh Dao | Development team |
| 3 | Ngo Quang Huy | Scrum master |
| 4 | Nguyen Phan Xuan Huy | Development team |
| 5 | Huynh Trong Khang | Product owner |
| 6 | Ta Ngoc Thien Phu | Development team |

Table 7:Team structure

# **Managerial Process**

## Management Objectives & Priorities

* Provide product in the 30-weeks period and within the proposed budget.
* Identify priority to complete the module to meet the requirements of customers in each period.
* Members of the team must meet and report progress of work assigned with person who has
* responsibility and scrum master weekly
* Meet the customers to report the progress of work with customers at the same time control and determine if any changes from the customers side weekly.

## Assumptions, Dependencies & Constraints

* Time and cost must be satisfied.
* Products must be reliable, satisfactory.
* Products must be consistent with functional and non-functional requirements which were identified
* Products must be friendly with user and easy to use
* All architect document have to review by customer and transfer for customer.

## Risk management plan

* Refer to folder: Risk management plan

## Change management plan

* Refer to folder: Change management plan

## Configuration management plan

* Refer to folder: Configuration plan

## Measurement plan

* Refer to: folder: Measurement plan

## Communication plan

* Refer to folder: Communication plan

## Architecture and design plan

* Refer to folder: Design

## Implement plan

* Refer to folder: Code

## Test plan

* Refer to folder: Test

## Team charter

* Refer to folder: Team charter

## Master Plans

N/A

# **Technical Process**

## Methods, Tools and Techniques

* Java 6
* Spring 3 + hibernate
* Maven 2
* Eclipse
* DB: MySQL
* Android app: sdk 4

## Software documentation

* Document software must comply with the standards of the development team.
* Review of documents will be carried out by the Scrum master at each stage of the job done.

# **Work Packages, Schedule**

## Work Packages

N/A

## Milestone



## Schedule

N/A