|  |
| --- |
| SOFTWARE PROJECT PLAN |
| HUMAN RESOURCE MANAGEMENT |
| [Type the abstract of the document here. The abstract is typically a short summary of the contents of the document. Type the abstract of the document here. The abstract is typically a short summary of the contents of the document.] |

10/10/2011

Project Information

usage notes:

<Describe the notes of your template here >

Table of template information :

|  |  |  |  |
| --- | --- | --- | --- |
| Issue Status: |  | | |
| Owner: |  | | |
| Author: |  | | |
| Location: |  | | |
| Confidential Class: |  | | |
| Template code: |  | | |
| Corporate template: |  |  |  |
| Reviewer information |  |  |  |
| Approver information |  |  |  |

* template Revision History

| Date | Version | Description | Revised by |
| --- | --- | --- | --- |
|  |  |  |  |

**Document Reviewer Information**

| Reviewer Name | Review Attendance (R/S) | Comments |
| --- | --- | --- |
|  |  |  |
|  |  |  |
| R/S: Required or Suggested participants or functions for the document review meeting | | |

**Document Approver Information**

| Approver Name | Approver Function | Comments |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |

**Document Revision History**

| Date | Revision | Description | Revised by |
| --- | --- | --- | --- |
| 10/15/2011 | 1.0 | Update template | Nhung Huynh |
| 10/16/2011 | 1.0.1 | Update part 5 | Loc Phan |
| 10/17/2011 | 1.0.2 | Update V Model Process | Tan Tran |
| 10/18/2011 | 1.0.3 | Requirement process | Quyet Nguyen |
| 10/23/2011 | 1.0.4 | Update project organization chart | Nhung Huynh |
| 10/23/2011 | 1.0.5 | Part 2 and detail design process | Dang Nguyen |
| 10/23/2011 | 1.0.6 | Update Architecture Process | Tuong Nguyen |
| 10/23/2011 | 1.0.7 | Update Assumption and Milestone | Tuong Nguyen |
| 10/23/2011 | 1.0.8 | Update Deliverable and Milestone | Quyet Nguyen |
| 10/25/2011 | 1.0.9 | Update testting process | Tung Nguyen |
| 10/30/2011 | 1.0.10 | Update testting process: Defect life cycle | Tung Nguyen |

**Table of Contents**

[I. Introduction 6](#_Toc316601411)

[1.1 Purpose 6](#_Toc316601412)

[1.2 Scope 6](#_Toc316601413)

[1.3 References 6](#_Toc316601414)

[2.4 Definitions, Acronyms and Abbreviations 6](#_Toc316601415)

[II. Project Overview 7](#_Toc316601416)

[2.1 Purpose 7](#_Toc316601417)

[2.2 Scope 7](#_Toc316601418)

[2.3 Deliverables 9](#_Toc316601419)

[2.4 Constraints 9](#_Toc316601420)

[2.5 Assumptions 9](#_Toc316601421)

[2.6 Project Processes 10](#_Toc316601422)

[2.6.1 Development process 10](#_Toc316601423)

[2.6.2 HRM Requirement Process 11](#_Toc316601424)

[2.6.3 Architecture Process 13](#_Toc316601425)

[2.6.4 Detail Design Process 17](#_Toc316601426)

[2.6.5 Testing Process 19](#_Toc316601427)

[2.6.6 Risk Management Process 25](#_Toc316601428)

[I. Project Quality Plan 29](#_Toc316601429)

[1.1 Acceptance Criteria 29](#_Toc316601430)

[1.2 Quality & Process Performance Objectives 29](#_Toc316601431)

[II. Project Estimation 29](#_Toc316601432)

[III. Project Resource Plan 30](#_Toc316601433)

[3.1 Human Resource 30](#_Toc316601434)

[3.2 Project Organization Chart 30](#_Toc316601435)

[3.2.1 Knowledge 30](#_Toc316601436)

[3.2.2 Skill 30](#_Toc316601437)

[3.3 Staffing 30](#_Toc316601438)

[3.5 Training Needs 31](#_Toc316601439)

[IV. Project Schedule 31](#_Toc316601440)

[4.1 Phases / Iterations 31](#_Toc316601441)

[4.2 Milestones 32](#_Toc316601442)

[4.3 Detailed Schedule 32](#_Toc316601443)

[V. Project Risk Management Plan 32](#_Toc316601444)

[5.1 Risk Source and Category 32](#_Toc316601445)

[5.1.1 Risk Sources 32](#_Toc316601446)

[5.1.2 Risk Categories 33](#_Toc316601447)

[5.2 Risk Management Approach 34](#_Toc316601448)

[5.3 Tools and Techniques 34](#_Toc316601449)

[5.4 Risk Items to be managed 34](#_Toc316601450)

[5.5 Risk Monitoring and Control 35](#_Toc316601451)

[VI. Project Stakeholders Involvements 35](#_Toc316601452)

[VII. Project Communication Plan 35](#_Toc316601453)

[Key Contacts List 35](#_Toc316601454)

[VIII. Communication Methodology 37](#_Toc316601455)

[IX. Project Factors 37](#_Toc316601456)

[Other Plans 38](#_Toc316601457)

# Introduction

## Purpose

This document is to establish reasonable plans for performing the software engineering and for managing the software project Human Resource Management. The customer of this project is Human Resource Department- Van Lang University

The plan includes information about the project, resource, schedule and processes that our team apply for management.

## Scope

This document is to cover parameters of a project and to establish the appropriate project management and quality environment required to complete the project:

1. Task Scheduling
2. Resource Planning
3. Organizational Planning
4. Risk Management
5. Change Management Process
6. Project Monitor &Control

## References

| **No.** | **Document ID** | **Description** | **Version** | **Location** |
| --- | --- | --- | --- | --- |
|  | Configuration Management Document | This document is used to manage the version of document and source code in HRM project | 1.0 |  |
|  | Measurement Plan | This document include all the metric and measurement collection process of HRM project | 1.0 |  |

## Definitions, Acronyms and Abbreviations

| **Term** | **Definition or description** |
| --- | --- |
| **HRM** | Human Resource Management |

# Project Overview

## Purpose

Human Resource System is a new system that will be a replacement forthe current Human Resource System at Van Lang University that is operated manually using Microsoft Excel.

HRM is particularly developed for human resource management in university / colleges. The system consists of key modules:

* Personal information management
* Employee labor contract management
* Recruitment & training processing
* Payroll
* Administration panel - Utilities

## Scope

The following figure shows an overall all the key features of HRM project



*Figure 1: Features data relationship of HRM project*

As shown above, HRM consists of eight key modules/features, which is described and prioritized as followed:

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Name | Description | Priority |
| HRM.FE1 | System Management | Managing HRM system, include user management, authentication, configuration, etc. This feature will be hidden and filtered by users' permission; only administrator of HRM can access this feature | 1 |
| HRM.FE2 | Recruitment Management | This feature is responsible for managing recruitment process. It includes interviewing, evaluating, managing probation information. | 8 |
| HRM.FE3 | Employee Labor Contract Management | This feature is responsible for managing employee contract information: salary ratio, class, grade, contract date, staff name… Besides, this feature helps HR easy to manage payroll. Therefore, it has a definite link to Payroll Management module. | 3 |
| HRM.FE4 | Insurance Information Management | This feature is responsible forcollecting and gathering information about insurance types, and manage premium. Beside, this feature bases on salary table to update insurance types of staff (including lecturer). | 6 |
| HRM.FE5 | Assessment Management | This feature is responsible forgathering information about work of staff, lectures, discipline, reward, etc. Moreover, assessments will be updated at the end of each year. This will helps HR to calculate salary. | 7 |
| HRM.FE6 | Employee Labor Management | This feature is responsible for receiving information about staff from Personal Information Management to manage working day, working hour. This will helps HR to calculate salary. | 4 |
| HRM.FE7 | Personal Information Management | It is one of the most important features. It will responsible formanagingresource information and provide it for other features.  ***In the first release of this project, our team will responsible for developing this feature.*** | 2 |
| HRM.FE8 | Payroll Management | Reporting and manage information about staff, lectures, salary etc. However, it is mainly about managing income of a staff of VLU | 5 |
| HRM.FE9 | Report Statistic Management | Getting information that customer wants to report |  |

## Deliverables

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Deliverable** | **Date** | **Method** | **Remark** |
| **1** | Concept Operation | 22/10/2011 | Email |  |
| **2** | Software Requirement Specification | 20/11/2011 | Email |  |
| **3** | Architecture Driver Document | 06/11/2011  06/12/2011 | Email |  |
| **4** | Software Architecture Specification | 13/11/2011  25/12/2011 | Email |  |
| **5** | Detail Design Specification |  |  |  |
| **6** | Source code |  |  |  |
| **7** | Test Plan |  |  |  |
| **8** | Test case specification |  |  |  |
| **9** | Test Report |  |  |  |
| **10** | Meeting report |  |  |  |
| **11** | Software Planning Document |  |  |  |

## Constraints

|  |  |  |
| --- | --- | --- |
| **No.** | **Constraint** | **Reference Doc  (Ref Doc ID, if any)** |
| Business Constraints | | |
| 1 | Resource: 8 members |  |
| 2 | Schedule: 200 days (include 22 days for Tet holiday)  Started date: 1/10/2011  Finished date: 1/5/2011 |  |
| Technical Constraints | | |
| 3 | Databases: MS SQL Server |  |
| 4 | Development languages: C#, WCF |  |
| 5 | Development platform: .Net 4.0 |  |

## Assumptions

| **No.** | **Assumption** | **Risks / Impacts  if the assumption is not true** |
| --- | --- | --- |
|  | Requirement will finish approximately 50 day and approach on contract between customer and the people get requirement. | Customer will change requirement any time on the project, The project groups will get Requirement again. |
|  | The Architecture Driver Document (ADD) must be reviewed before moving to Architecture Design phase | If the assumption does not occur, the ADD needs to rework |
|  | The Architecture Design document must be reviewed and get the consensuses before moving to Design phase | If the assumption does not occur, the Architecture Design document needs to rework |

## Project Processes

## Development process

Human resource management project is developed following the V-model methodology. There are four reasons for choosing this model.

1. The V-model helps to minimize the project risks by specifying standardized approaches and describing the corresponding results and responsible roles. It permits an early recognition of planning deviations and risks and improves process management, thus reducing the project risk.
2. Improvement and Guarantee of Quality: the V-model ensures that the result to be provided is complete and has the desired quality.
3. Reduction of total cost: The V-model can help you to calculate the effort of development, production, operation and maintenance of a system.
4. Improvement of Communication between all stakeholders: each step in V-Model (requirement, design, code, test …) must be verified and validated between stakeholders and it can help to improve the communication between the stakeholders.



Figure: V-model process

|  |  |
| --- | --- |
| **Phase** | **Description** |
| Requirement | In this phase, the requirement engineers will start to get the requirement list from stakeholder, analyse requirement and document them. These documents will be validated by the satkeholder and verify by team members.  After the requirements is all agreed by stakeholders. The testing team will start to design the acceptance test cases base on the customer acceptance criteria |
| Architecture | After the requirement is finalized, the chief architect will start to create the architecture and refine them until there is no issue on architecture.  After this phase, the testing team will start to design system test cases. |
| Detail design | The system will be design in more detail. The class diagram and sequence diagram will be design to ensure that the developer can build the system as the architecture.  After this phase, the testing team will start to design intergration test cases |
| Build the system | The developers will start build the system based on the detail deisign  The unit testing will also implement as the same time the coding phase to ensure each module or function is tested deeply. |
| Testing | After the system is build. The testers will execute test cases which are design in previous phases. *[See more in Test plan about each kind of testing]* |

**2.6.2 HRM Requirement Process**

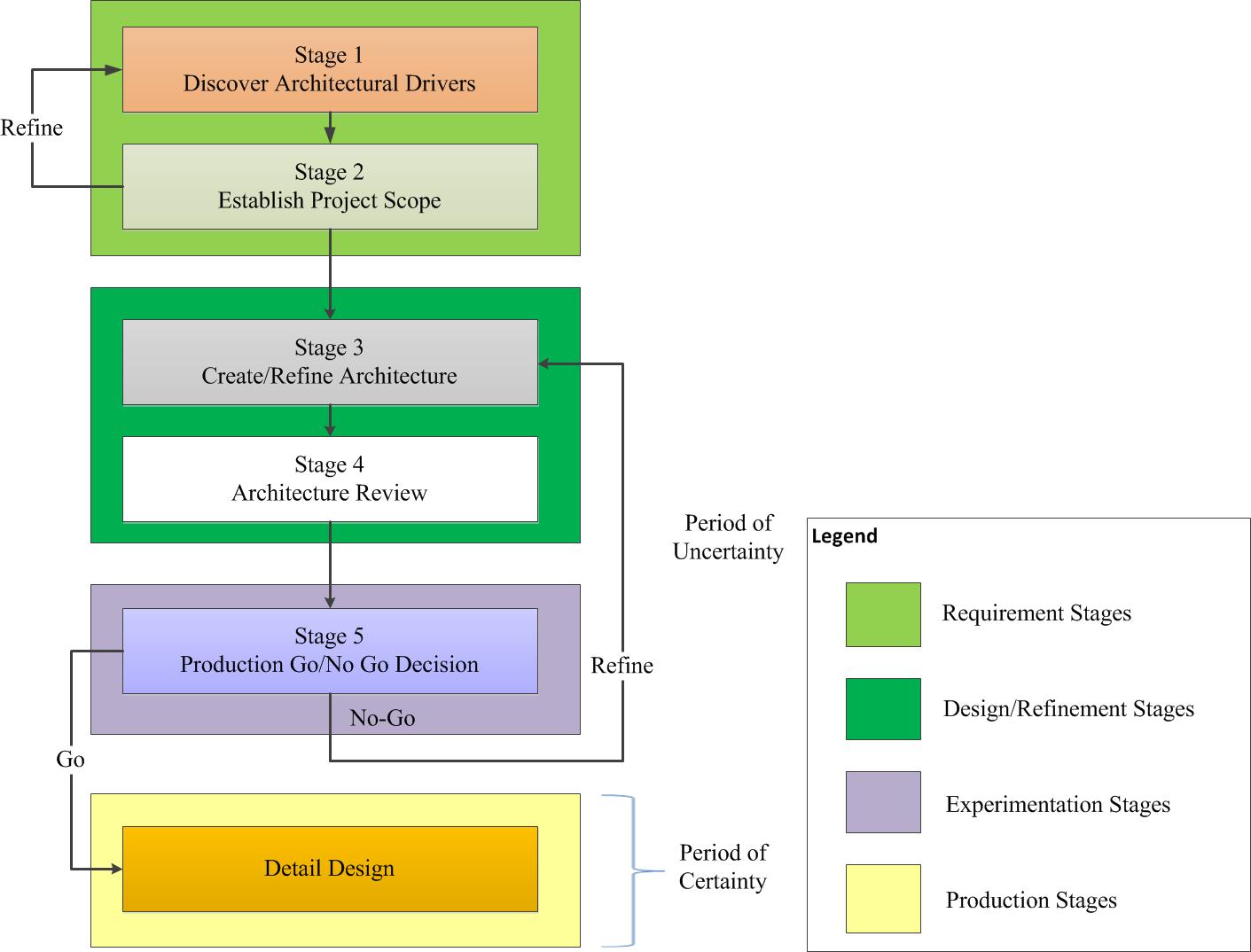
## HRM Requirement Process



|  |  |
| --- | --- |
| **Stage Name** | **Plan** |
| *Purpose* | - |
| *Precondition* |  |
| *Output* |  |
| *Tool & techniques* |  |
| *Post condition* |  |
| *Activities* |  |
| **Stage Name** | **Elicitation** |
| *Purpose* | - |
| *Precondition* |  |
| *Output* |  |
| *Tool & techniques* |  |
| *Post condition* |  |
| *Activities* |  |
| **Stage Name** | **Analysis** |
| *Purpose* | - |
| *Precondition* |  |
| *Output* |  |
| *Tool & techniques* |  |
| *Post condition* |  |
| *Activities* |  |
| **Stage Name** | **Specification** |
| *Purpose* | - |
| *Precondition* |  |
| *Output* |  |
| *Tool & techniques* |  |
| *Post condition* |  |
| *Activities* |  |
| **Stage Name** | **Verification** |
| *Purpose* | - |
| *Precondition* |  |
| *Output* |  |
| *Tool & techniques* |  |
| *Post condition* |  |
| *Activities* |  |
| **Stage Name** | **Validation** |
| *Purpose* | - |
| *Precondition* |  |
| *Output* |  |
| *Tool & techniques* |  |
| *Post condition* |  |
| *Activities* |  |

|  |  |
| --- | --- |
| **Phase** | **Description** |
| Plan (1 excel for making a plan every two week) | * Understand vision and scope of Project * Define requirements check list * Receive and research on any document provided by customers and mentors * Create assumptions about HRM requirements * Define role for each integration (per week) |
| Elicitation | * Apply requirement check list * Acquire human resource information * Get requirements from customers * Validate assumptions with customer |
| Analysis | * Assemble information * Check and classify requirements from what customer provides * Feedback on Requirement * Analyze requirement based on customer’s point of views. * Analyze any issues from verification among team members. |
| Specifications | * Concepts of Operation * Software Requirement Specification Document |
| Verification | * Review and check ConOps and SRS among team members |
| Validation | * Review and check ConOps with customers. * Accept and close RE process |

## Architecture Process



|  |  |  |
| --- | --- | --- |
| **Stage Name** | **Discover Architecture Driver** | **R**  **E**  **Q**  **U**  **I**  **R**  **E**  **M**  **E**  **N**  **T**  **S**  **T**  **A**  **G**  **E** |
| *Purpose* | The primary purpose of stage 1 is for the architecture design team to initiate one or more meetings with the client stakeholder community (or communities) to discover and document the system’s architectural drivers, to include high-level functional requirements, business constraints, technical constraints, and quality attributes. |
| *Precondition* | The architecture design team must be established and the ACDM roles  must be assigned to the architecture team members |
| *Output Document* | The raw architectural drivers describing what the stakeholders expect of the system. |
| *Post condition* | * The initial master design plan has been created and is updated as required after each architecture driver elicitation workshop. * All or key stakeholders or stakeholder groups have been engaged using the architecture drivers elicitation workshop. * The raw architectural drivers have been collected from the stakeholders and consolidated and documented. The focus of stage 1 is to collect data, not analyze or structure it. |
| *Activities* | Interact with stakeholders to discover and document the raw architectural drivers. |
|  | |
| **Stage Name** | **Establish Project Scope** |
| *Purpose* | The primary purpose of stage 2 is for the architecture design team to analyze the consolidated raw architecture driver information gathered in stage 1 to clarify and refine the architectural drivers and firmly establish the scope of the system/product. |
| *Precondition* | The consolidated raw architecture drivers from stage 1 must be available. |
| *Output Document* | The architectural driver specification and the updated master design plan. |
| *Post condition* | The architecture driver specification is completed and reviewed and formally accepted by the stakeholders. |
| *Activities* | Refine raw architectural drivers into an architectural driver specification, and define the scope of the work. |
|  | | |
| **Stage Name** | **Create/Refine Architecture** | **D**  **E**  **S**  **I**  **G**  **N**  **/**  **R**  **E**  **F**  **I**  **N**  **E**  **M**  **E**  **N**  **T**  **S**  **T**  **A**  **G**  **E** |
| *Purpose* | * Create the initial architectural design, or refine the architectural design based on the results of the architectural evaluation.   + If this is the first iteration in stage 3, then the initial notional architecture design will be created.  + If the decision is to continue refining the design (stage 5), then issues uncovered in the evaluation are addressed after stage 5. |
| *Precondition* | The architectural drivers must be analyzed and documented as described in stage 2 vis-à-vis the architecture driver specification. If this is the second (or nth) time through stage 3, the issues raised in the stage 4 evaluation must have been addressed by the architecture design team through experimentation after stage 5. |
| *Output Document* | The initial architectural design or the refined architectural design and the associated documentation artifacts. |
| *Post condition* | * The notional architecture design is completed (first time through stage 3), or the architecture design is refined based on experiments conducted after stage 5 (nth time through stage 3). * The notional architecture design is documented (first time through stage 3), or the architecture design documented is updated after refining the architecture based on experimentation after stage 5 (nth time through stage 3). |
| *Activities* | Create or refine the architecture design. After initial design the architect (or architecture team) will return to this step after the productions go/no go decision (stage 5) to refine the architecture. |
|  | |
| **Stage Name** | **Architecture Review** |
| *Purpose* | The primary purpose of stage 4 is for the architecture design team to evaluate the initial architectural design, or reevaluate the refined design after architectural evaluation and experimentation. |
| *Precondition* | The architecture design must be sufficiently complete to facilitate the design evaluation. At a minimum, the architecture design must be designed and documented in preliminary fashion with representation from the three primary perspectives. In addition to drawings, there must be sufficient prose to describe the design and its rationale. |
| *Output Document* | A list of issues uncovered during the evaluation that impact the design’s ability to satisfy the architectural drivers. |
| *Post condition* | Architecture design is evaluated and key issues identified and documented |
| *Activities* | Review the architecture to discover and document issues that may compromise the satisfaction of the architectural drivers. |
|  | | |
| **Stage Name** | **Production Go/No Go Decision** | **E**  **X**  **P**  **E**  **R**  **I**  **M**  **E**  **N**  **T**  **S**  **T**  **A**  **G**  **E** |
| *Purpose* | * The architecture design team to analyze the issues uncovered in stage 4 during the architectural design evaluation and devise concrete strategies for how to address each issue * The architecture design team to resolve issues uncovered during the evaluation in stage 4 by carrying out the actions described for each issue in the issue deposition document developed in stage 5 |
| *Precondition* | * The architecture design must have been evaluated and the issues from the evaluation recorded and available to all of the architecture design team * The architecture design team must have developed the issue deposition   document and assigned responsible engineers to each issue for experimentation |
| *Output Document* | * Issue deposition document template * The experimentation template. |
| *Post condition* | * There is a concrete strategy for how the issues uncovered during the stage 4 evaluation will be addressed by the architecture design team. A decision is made as to whether the team will **further refine the architecture design through the production Go/No Go decision** (stage 5), or if the team will **begin planning the implementation** of the design in the production stages (detail design stage). * The experiments have been conducted for each issue according to the issue deposition document. |
| *Activities* | 1. Preparation before meeting 2. Conduct analysis meeting. Team discusses or debates on the issues and reaching consensus on these matters can be challenging. 3. Make decision: refine or implement the design 4. Review The issue disposition document should be reviewed and circulated among the broader stakeholder community for comment 5. Experimentation planning: Each responsible engineer will develop a short plan for how he or she will perform the action described for each issue he or she is responsible for according to the issue deposition document. 6. Experimentation: Each responsible engineer will conduct the experiments specified in his or her experimentation plans. He or she will collect all relevant technical data and will track the time spent on each experiment. 7. Experimentation review: One the experiments are concluded, the team will share and review the outcomes and the data collected during the experiments |
|  | | |

## Detail Design Process



|  |  |
| --- | --- |
| **Stage Name** | **Definition** |
| *Purpose* | - |
| *Precondition* |  |
| *Output* |  |
| *Tool & techniques* |  |
| *Post condition* |  |
| *Activities* |  |
| **Stage Name** | **Analysis** |
| *Purpose* | - |
| *Precondition* |  |
| *Output* |  |
| *Tool & techniques* |  |
| *Post condition* |  |
| *Activities* |  |
| **Stage Name** | **Evaluation** |
| *Purpose* | - |
| *Precondition* |  |
| *Output* |  |
| *Tool & techniques* |  |
| *Post condition* |  |
| *Activities* |  |
| **Stage Name** | **Detail Design** |
| *Purpose* | - |
| *Precondition* |  |
| *Output* |  |
| *Tool & techniques* |  |
| *Post condition* |  |
| *Activities* |  |
| **Stage Name** | **Review** |
| *Purpose* | - |
| *Precondition* |  |
| *Output* |  |
| *Tool & techniques* |  |
| *Post condition* |  |
| *Activities* |  |

|  |  |
| --- | --- |
| **Phase** | **Description** |
| Definition | * + This task describes detail function like simple step     - Input: interface of requirement, architecture and scenario     - Output: action for each function |
| Analysis | * + This task bases on action of each function to analyzes, finds solution and appraise the complexity of each action.     - Input: action of each function       * Identify action       * Find Solution       * Appraise the complexity of action     - Output: the complexity of each action |
| Evaluation | * + This task bases on the complexity of action to evaluate the impact to system and identify priority of them.     - Input: the complexity of action     - Output: the priority of action and function |
| Design Detail | * + This task develops Sequence for each function.     - Input: the priority of function     - Output: Detail Design document |
| Review | * + This task exams design, which is correct solution on Analysis task     - Input: Detail Design Document     - Output: Report |

## Testing Process

**Definition:**

The course of software being tested in a well-planned way is known as Software test life cycle.

**Test Execution**

**Test Development**

**Test Planning**

**Product Delivery**

**Retest Defects**

**Defect Reporting**

**Stages Involved:**

|  |  |
| --- | --- |
| **Stage Name** | **Test Planning** |
| *Purpose* | - |
| *Precondition* |  |
| *Output* |  |
| *Tool & techniques* |  |
| *Post condition* |  |
| *Activities* |  |
| **Stage Name** | **Test Development** |
| *Purpose* | - |
| *Precondition* |  |
| *Output* |  |
| *Tool & techniques* |  |
| *Post condition* |  |
| *Activities* |  |
| **Stage Name** | **Test Execution** |
| *Purpose* | - |
| *Precondition* |  |
| *Output* |  |
| *Tool & techniques* |  |
| *Post condition* |  |
| *Activities* |  |
| **Stage Name** | **Defect Reporting** |
| *Purpose* | - |
| *Precondition* |  |
| *Output* |  |
| *Tool & techniques* |  |
| *Post condition* |  |
| *Activities* |  |
| **Stage Name** | **Retest Defects** |
| *Purpose* | - |
| *Precondition* |  |
| *Output* |  |
| *Tool & techniques* |  |
| *Post condition* |  |
| *Activities* |  |
| **Stage Name** | **Product Delivery** |
| *Purpose* | - |
| *Precondition* |  |
| *Output* |  |
| *Tool & techniques* |  |
| *Post condition* |  |
| *Activities* |  |

|  |  |  |
| --- | --- | --- |
| **Stage Name** | **Description** | **T**  **E**  **S**  **T**  **I**  **N**  **G**  **P**  **R**  **O**  **C**  **E**  **S**  **S** |
| **Test Planning** | * The testing team creates a test plan * The test plan defines the objectives and scope of the testing effort, and identifies the methodology that your team will use to conduct tests * The test plan identifies the hardware, software, and tools required for testing and the features and functions that will be tested * All members of team should review and approve its team’s test plan before it is integrated into the general test plan * It include: * Define testing scope and objectives * Define testing methodology * Identify required resources * Identify the features and functions to text * Identify risk factors * Establish a testing schedule   **Documents Involved:**   * Test Plan * Test cases |
| **Test Development** | * The test development is focused on test requirement analysis and documentation in the form of a test specification * Gather initial information and test intent * Propose a test solution with preliminary requirements * Solidify test requirement * Develop/Document test solution * Validate test solution * Deliver test solution to production environment   **Documents Involved:**   * Test Plan * Test cases |
| **Test Execution** | * The following activities are involved in performance test execution: * Executing Test cases * Testing Test Scripts * Capture, review and analyze Test Results * Raising the defects and tracking for its closure   **Documents Involved:**   * Test Cases * Test Execution report * Bug report |
| **Defect Reporting** | * A Defect Report describes defects in released software and documentation products. Defect Reports have a priority rating that indicates the impact the problem has on the customer * Defect reports are probably primary work product for most of the software testers * Write reports for: * Defect logging * Assigning defect and fixing * Retesting * Defect closing   **Documents Involved:**   * Test report * Bug Report |
| **Retest Defects** | * Defects fixed without changing code as in environment defects were retested and closed immediately. * Defect fixes requiring code changes were included in a build package and were retested and either closed or reopened depending on the outcome of the test.   **Documents Involved:**   * Test report |
| **Product Delivery** | * Analysis of the results of running tests * The results analysis provide management and the development team with a readout of the product quality   **Documents Involved:**   * Test summary reports |

**Defect Life Cycle:**

In software development process, the defect has a life cycle. The defect should go through the life cycle to be closed. A specific life cycle ensures that the process is standardized. The defect attains different states in the life cycle.

The life cycle of the defect can be shown diagrammatically as follows:



**Defect life cycle description:**

The testers or test group will report the bug to the change control board and Project manager.

|  |  |
| --- | --- |
| **Status** | **Description** |
| **Open** | * When the defect is posted for the first time, the lead of tester approves that the bug is new and change the status of defect as “OPEN”. The project manager will collect the number of new defect every week. * After the defect is report, we have a special team for analysis the impact and evaluate the benefit of benefit or change if we fix this defect. If the defect is real and impact on the project, the defect status will be change to “Approved”. And the project manager or QA team will assign the defect to corresponding developers and the status will be change to “Assigned” * Whenever the developers receive the defect and they don’t approve that it really is the defect and has the reason to believe that. The status will be changed to “Rejected” or "Deferred". |
| **Assigned** | * Once the defect is change to “Approved”, lead of testers assign the defect to corresponding developer or developer team. * We also have a special team to check all the opened defects and decide or find the defect that is not real defect. Defect checking will start whenever the defect change the status from opened to assigned. * After the defect is assigned to developers, developers start to resolve the defect with provided solution from QA team. After the defect is resolved by developers, the status of defect will be changed to “Resolved”. |
| **Resolved** | * The software engineer responsible for fixing the defect believes that the defect is fixed. The defect status will be changed to “RESOLVED”. * After the defect is resolved by the developers, the QA team start verify the resolving, if it is really resolved with right solution, the defect status will be changed to “ Tested” |
| **Tested** | * The defect is considered resolved by the QA team or the QA verifies and accepts the defect is fixed. * After the defect is tested and team QA accept to close the defect. The status of defect will be changed to “Closed”. |
| **Reopened** | * A defect that was previously Closed, Pending or Rejected has been reopened. Typically this means that the QA is not satisfied with the solution of developer. The status will be change to “REOPENED” |
| **Closed** | * The updated product is available for release. Sometimes the original defect submitter is responsible for closing defects, but often this isn't possible. |

**Guidelines on deciding the Severity of Bug:**

Indicate the impact each defect has on testing efforts or users and administrators of the application under test. This information is used by developers and management as the basis for assigning priority of work on defects.

A sample guideline for assignment of Priority Levels during the product test phase includes:

|  |  |
| --- | --- |
| **Status** | **Description** |
| **Critical** | * An item that prevents further testing of the product or function under test can be classified as Critical Bug. No workaround is possible for such bugs. * Examples of this include a missing menu option or security permission required to access a function under test |
| **High** | * A defect that does not function as expected / designed or cause other functionality to fail to meet requirements can be classified as Major Bug. The workaround can be provided for such bugs. * Examples of this include inaccurate calculations; the wrong field being updated, etc. |
| **Medium** | * The defects which do not conform to standards and conventions can be classified as Medium Bugs. Easy workarounds exists to achieve functionality objectives. * Examples include matching visual and text links which lead to different end points. |
| **Low** | * Cosmetic defects which does not affect the functionality of the system can be classified as Minor Bugs. |

## Risk Management Process

**RISK MANAGEMENT PLANNING**

These process interact with each other and with the process in the other areas

Each process generally occur at least once in the project

|  |  |
| --- | --- |
| **RISK MANAGEMETN PLANNING** | |
| ***Purpose*** | * The process decide how approach and plan the risk management activities for a project * Ensure that the level, type and visibility of risk management are commensurate with both the risk and importance of the project to the organization |
| ***Input*** | Defined roles and responsibilities  Templates  Work breakdown structure (WBS) |
| ***Tool & techniques*** | Planning meeting |
| ***Output*** | Risk management plan:   * Methodology * Role & responsibility * Timing * Scoring & interpretation * Thresholds * Reporting format   Tracking |
| **RISK IDENTIFICATION** | |
| ***Purpose*** | * Determining which risks might affect the project and documenting their characteristics * Participants: project team, risk management team, project manager, primary stakehoders * It is an iterative process. The first iteration may be performed by a part of the project team/the risk management team. The entire project team and primary stakeholders may make a second iteration. * Often simple & effective risk responses can be developed and even implemented as soon as the risk identified. |
| ***Input*** | Risk management plan  Project planning outputs  Risk categories |
| ***Tool & techniques*** | Documentation reviews  Information-gathering techniques  Checklists  Assumption analysis |
| ***Output*** | Risks  Triggers  Inputs to other processes |
| **RISK ANALYSIS** | |
| **QUALITATIVE RISK ANALYSIS** | |
| ***Purpose*** | * The process of assessing the impact and likelihood of identified risks * Priorities risks according to their potential effect on project objectives * Determine the importance of addressing actions may magnify the importance of a risk * Help modify the assessment of the risk * Analysis requires that the probability and consequences of the risks be evaluated using established qualitative analysis methods and tools * Qualitative analysis should be revisited during the project’s life cycle to stay current with changes in the project risks |
| ***Input*** | Risk management plan  Identified risks  Project status  Project type  Data precision  Scales of probability & impact  Assumptions |
| ***Tool & techniques*** | Risk probability and impact  Probability/impact risk rating matrix  Project assumptions testing  Data precision ranking |
| ***Output*** | Overall risk ranking for the project  List of prioritized risks  List of risk additional analysis & management  Trends in qualitative risk analysis result |
| **QUANTITATIVE RISK ANALYSIS** | |
| ***Purpose*** | * The quantitative risk analysis process aims to analyze numerically the probability of each risk and its consequence on project objectives. Give decision analysis to: * Determine the probability of achieving a specific project objective * Quantify the risk exposure for the project, and determine the size of cost and schedule contingency reserves may be needed * Identify risks requiring the most attention by quantifying their relative contribution to project list * Identify realistic and achieve schedule, scope targets |
| ***Input*** | Risk management plan  Identified risks  List of prioritized risks  List of risks additional analysis and management  Other planning inputs |
| ***Tool & techniques*** | Interviewing  Sensitivity analysis  Decision tree analysis  Simulation |
| ***Output*** | Prioritized list of quantified risks  Probabilistic analysis of the project  Probability of achieving the cost and time objectives  Trends in quantitative risk analysis results |
| **RISK RESPONSE PLANNING** | |
| ***Purpose*** | * The process of developing options and determining actions to enhance opportunities and reduce threats to the project objectives * It includes the identification and assignment of individuals or parties to take responsibility for each agree risk response |
| ***Input*** | Risk management plan  List of prioritized risks  Risk ranking of the project  Prioritized list of quantified risks  Probabilistic analysis of the project  Probability of achieving the time objectives  List of potential responses  Risk thresholds  Risk owners  Common risk causes  Trend in qualitative and quantitative risk analysis results |
| ***Tool & techniques*** | Avoidance  Transference  Mitigation  Acceptance |
| ***Output*** | Risk response plan  Residual risk |
| **RISK MONITORING & CONTROL** | |
| ***Purpose*** | * The process of keeping track of the identified risks, monitoring residual risks & identify new risks, evaluating effectiveness in reducing risks. |
| ***Input*** | Risk management plan  Risk response plan  Project communication  Scope changes |
| ***Tool & techniques*** | Project risk response audits  Earn value analysis  Technical performance measure  Addiction risk response planning |
| ***Output*** | Corrective action  Project change requests  Update to the risk response plan  Risk database  Updates to risk identification checklist |

# Project Quality Plan

## Acceptance Criteria

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Work Product** | **Criteria** | **Threshold** |
|  | Alpha-build | Defects found must be fixed before delivery | > 95% |
|  | Alpha-build | P1 CRs not completed | < 2 |
|  |  |  |  |
|  |  |  |  |

## Quality & Process Performance Objectives

|  |  |
| --- | --- |
| Objectives | Description |
| OBJ\_01 | Deliverables to customer are ensured to meet: >= 95 % on-time |
| OBJ\_02 | The variance of schedule is under control with the following criteria:  Schedule deviation is between [–7%, 7%] |
| OBJ\_03 | The variance of effort is under control with the following criteria:  Effort deviation is between  [–20%, 20%] |

# Project Estimation

[This section lists the total/overall estimation data (size, effort, and cost) per project phase]

| **No.** | **Estimation Type** | **Phase/Iteration** | **Unit** | **Value** |
| --- | --- | --- | --- | --- |
|  | **Size** |  | FP/UP/TP | 4,567 |
|  |  | Phase #1 | FP/UP/TP | 123 |
|  |  | Phase #2 | FP/UP/TP | 456 |
|  | **Effort** |  | Man-hours | 23,456 |
|  |  | Phase #1 | Man-hours | 1234 |
|  |  | Phase #2 | Man-hours | 2345 |

For further detail, please refer to **Estimation Form / Estimation Output**.

# Project Resource Plan

## Human Resource

* Resource

### Project Organization Chart

### Knowledge

* Developers must have basic knowledge about C#, WCF, SQL Server, Silverlight.
* Besides developers know to get requirement, design, architect, basic code.

### Skill

* Project management skill
* Teamwork skill
* Negotiate skill
* Listen skill
* Problem resolve skill

### Staffing

* Customer: staff of training room and accountancy room.
* Development Group who is responsibility for development process...
  1. Project Management team

| No. | Role | Name | Planned Join Date | Planned Leave Date |
| --- | --- | --- | --- | --- |
|  | Project Manager | HuỳnhThịHồngNhung | 09-19-2011 |  |
|  | Team member | TrầnNguyễnHoàngTân | 09-19-2011 |  |
|  | Team member | Nguyễn Kim Tường | 09-19-2011 |  |
|  | Team member | ĐinhNguyễnKhôiNguyên | 09-19-2011 |  |
|  | Team member | PhanGiaBáLộc | 09-19-2011 |  |
|  | Team member | NguyễnKhắcQuyết | 09-19-2011 |  |
|  | Team member | NguyễnNgọcTùng | 09-19-2011 |  |
|  | Team member | NguyễnTiếnĐặng | 09-19-2011 |  |

### Training Needs

Training about C#, Silverlight

[List any special training that project team members will require, with target dates for when this training should be completed.

# Project Schedule

## Phases / Iterations

| **No.** | **Phase** | **Iteration** | **Star Date** | **End Date** | **Comment** |
| --- | --- | --- | --- | --- | --- |
| 1. | Requirement Analysis |  | 10/1/2011 | 11/20/2011 | Update until week 22 |
| 2. | Architect Design Analysis |  | 11/6/2011 | 12/25/2011 | Update until week 22 |
| 3. | Detail Design Analysis |  | 12/25/2011 | 3/4/2012 | Update until week 22 |
| 4. | Implement |  | 3/4/2012 | 4/4/2012 |  |
| 5. | Testing |  | 4/4/2012 | 4/14/2012 |  |
| 6. | Deploy Product |  | 4/15/12 | 4/17/2012 |  |
| 7. | Maintenance |  | 4/17/2012 | 4/30/2012 |  |

## Milestones

[List all project milestones. For **major milestones** must be remark as “**Major**” in the **Comment** column of the table, for not major milestones, could be keep blank or mark as “**interim**”.

Notes: A major milestone must have a milestone review meeting.

A **major milestone** is a mandatory milestone:

1. Defined/required by customer
2. End of iteration

PM can define internal milestones, duration of an internal milestone must <= 3 months]

| **No.** | **Milestone** | **Description** | **Due Date** | **Comment** |
| --- | --- | --- | --- | --- |
| 1 | Requirement | Beginning get Requirement | 10.1.2011 | Major |
| 2 | Requirement | Finishing get Requirement | 11.20.2011 | Major |
| 3 | Architecture Driver Document (ADD) | Review between team members and mentor about the ADD | 11.12.2011 | Major |
| 4 | Architecture Design document | Review between team members and mentor about the Architecture Design document | 12.10.2011 | Major |

## Detailed Schedule

[Insert Gantt Chart, or MS Project or MrProject® tool here]

[List all project main tasks in high level that will be performed and relevant schedule and resources allocated.]

# Project Risk Management Plan

## Risk Source and Category

### Risk Sources

| **No.** | **Source of risk** | **Description** |
| --- | --- | --- |
|  | **Requirement** | This source covers risk introduced due to different attributes of the requirement elements such as -- Stability, Completeness, Clarity, Validity, Feasibility, Precedent and Scale. |
|  | **Design** | It is "Translation of requirements into effective design". Design covers the risk introduced due to following attributes - Functionality, Difficulty, Interfaces, Performance, Testability, Hardware Constraints and Non developmental Software. |
|  | **Code & Unit Test** | It is "Translation of software design into code that satisfies the specified requirements. This source covers risk introduced due to following attributes -- Feasibility, Unit Test and Coding/Implementation. |
|  | **Integration & Test** | It is “Integration of code into a system and the validation that the software performs as required. This source covers risk introduced due to following attributes -- Environment, Product & System |
|  | **Engineering Specialties** | It is a set of Product requirements or development activities that may need specializes expertise such as -- Maintainability, Reliability, Safety, Security & Human Factors |
|  | **Development Process** | It is the definition, planning, documentation, suitability enforcement and communication of the methods & procedures used to develop the product. This element groups risks due to -- Formality, Suitability, Process control, Familiarity & Product Control. |
|  | **Development System** | It addresses the hardware and software tools & supporting equipments used in product development. The risk for this source may be due to -- Capacity, Suitability, usability, Familiarity, Reliability, System Support and deliverability. |
|  | **Management Process** | It address the risks related to following attributes -- Planning, Project Organization, Management Experience & Program Interfaces. |
|  | **Management Methods** | It refers to the methods for managing both the development of the product program personnel. It groups risks related to following attributes -- Monitoring, Personnel Management, Quality Assurance & Configuration Management. |
|  | **Work Environment** | It refers to the subjective aspects of environment such as -- Quality Attitude, Cooperation, Communication & Morale |
|  | **Resources** | It refers to the factors on which a project is totally dependent upon such as -- Schedule, Staff, and Budget & Facilities. |
|  | **Contract** | Risks related to the contracts are classified according to -- Type of contract, Restrictions and Dependencies. |
|  | **Program Interfaces** | Risks related to other groups that are interfaced with the project --Customer, Suppliers and Corporate management |

### Risk Categories

|  |  |  |
| --- | --- | --- |
| **No.** | **Risk Category** | **Description** |
| **1** | **Technical risks** | Risks related to scope, technologies, external dependencies |
| **2** | **Resource risks** | Risks related to people, time, budget & organization |
| **3** | **Process risks** | Risks related to processes define, compliance |
| **4** | **Business risks** | Risks related to competitor, contract and finance |
| **5** | **Customer risks** | Risks related to customer problem |
| **6** | **Others** | Include all risks that are not in the above categories |

## Risk Management Approach

|  |  |
| --- | --- |
| **Risk Category** | **Description** |
| **Avoid** | The risk can be avoided by choosing a lower-risk alternative. For example, choose a less exotic technology (but accepting the higher weight), or go with an on-time, more expensive supplier (but accepting the higher cost) |
| **Transfer** | The risk will be transferred to another party (customer, external groups,...) for handling |
| **Mitigate** | Mitigation seeks to reduce the probability and/or consequences of an adverse risk event to an acceptable threshold. Taking early action to reduce the probability of risk’s occurring or its impact on the project is more effective than trying to repair the consequences after it has occurred. |
| **Accept** | The project team has decided not to change the project plan to deal with a risk or is unable to identify any other suitable response strategy |

## Tools and Techniques

## Risk Items to be managed

|  |  |
| --- | --- |
| **Risk Status** | **Description** |
| **Identified** | Correlative with Risk Identification |
| **Analysis Complete** | Correlative with Risk Analysis |
| **Planning Complete** | Correlative with Risk Response Planning |
| **Triggered** | Correlative with Risk |
| **Resolved** | Risk resolved |
| **Retired** | Risk retired |

## Risk Monitoring and Control

1. Weekly, PM monitors identified risks and new risks. When risk threshold is reached, PM implements the mitigation action(s).
2. Monthly, PM re-assessments the risks that are controlled. If a risk occurred, PM implements the contingency plan(s).
3. If a risk becomes issue, PM records that issue to the Issue List.

# Project Stakeholders Involvements

|  |  |
| --- | --- |
| **ROLE** | **NAME** |
| Mentor | NguyễnThếQuang  ĐinhĐứcTrí  NguyễnAnhNhân |
| Development team | Human Resource Team |
| Customer | Van Lang uni.PhongNhan Luc |

# Project Communication Plan

## Key Contacts List

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Role** | **Represent for** | **Contact Info** | **Interface** |
|  | PhòngNhânLực | Customer | VLU | Address: 45 Nguyen KhacNhu, Q1, HCM  Email:[p.kh@vanlanguni.edu.vn](mailto:p.kh@vanlanguni.edu.vn) |  |
|  | NguyễnThếQuang | Mentor | HRM team | Phone:0989990017  Email: [quangsm1994@gmail.com](mailto:quangsm1994@gmail.com) |  |
|  | ĐinhĐứcTrí | Mentor | HRM team | Email: [nguyenanhnhan@vanlanguni.edu.vn](mailto:nguyenanhnhan@vanlanguni.edu.vn) |  |
|  | NguyễnAnhNhân | Mentor | HRM team | Email: [dinhductri@vanlanguni.edu.vn](mailto:dinhductri@vanlanguni.edu.vn) |  |
|  | HuỳnhThịHồngNhung | Team Leader | HRM team | Email: [nhunghuynhthihong@gmail.com](mailto:nhunghuynhthihong@gmail.com) |  |
|  | TrầnNguyễnHoàngTân | Team Member | HRM team | Email: [hoangtanvlu@gmail.com](mailto:hoangtanvlu@gmail.com) |  |
|  | Nguyễn Kim Tường | Team Member | HRM team | Email: [kimtuongvlu@gmail.com](mailto:kimtuongvlu@gmail.com) |  |
|  | ĐinhNguyễnKhôiNguyên | Team Member | HRM team | Email: [shadow141206@gmail.com](mailto:shadow141206@gmail.com) |  |
|  | PhanGiaBáLộc | Team Member | HRM team | Email: [tuongcuop.ali@gmail.com](mailto:tuongcuop.ali@gmail.com) |  |
|  | NguyễnKhắcQuyết | Team Member | HRM team | Email: [nguyenkhacquyet89vl@gmail.com](mailto:nguyenkhacquyet89vl@gmail.com) |  |
|  | NguyễnNgọcTùng | Team Member | HRM team | Email: [haycogang0207@gmail.com](mailto:haycogang0207@gmail.com) |  |
|  | NguyễnTiếnĐặng | Team Member | HRM team | Email: [dangnguyen2409@gmail.com](mailto:dangnguyen2409@gmail.com) |  |

## Communication Methodology

[Describe the audiences, topic of information delivered, frequencies, and method in each communication]

| **Audience/ Attendees** | **Topic/ Deliverable** | **Frequency** | **Method** |
| --- | --- | --- | --- |
| Senior Management | Progress Review | Weekly / Monthly | - Meeting  - Project Weekly Report  - Project Monthly Report |
| * + Customer   + Senior Management   + SQA Rep | Milestone Review | Per milestone | - Meeting  - Milestone Report |
| Project Steering Committee | Gate Review | End of each Phase | - Meeting  - Gate Review Report |
| Customer | Status Report | Weekly / Monthly | Status Report |

## Project Factors

*[This list of project factors is using for monitoring and controlling the project execution]*

.

| **No.** | **Project Factor** | **Frequency** | **Tracker** | **Method** | **Threshold** |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |

# Other Plans

| **No.** | **Plan name** | **Plan Description** | **Ver.** | **Plan Location** |
| --- | --- | --- | --- | --- |
|  | <Filename of Plan> | Budget Plan | 1.1 |  |
|  | <Filename of Plan> | QA Plan | 1.0 |  |
|  | <Filename of Plan> | CM Plan | 1.2 |  |
|  | <Filename of Plan> | MA Plan | 1.1 |  |
|  | <Filename of Plan> | OCP (Operational continuity plan) | 1.0 |  |
|  | <Filename of Plan> | Project Procurement Plan | 1.0 |  |
|  | <Filename of Plan> | Test Plan | 1.3 |  |
|  | <Filename of Plan> | <any others plan> |  |  |

----------- End of Document -----------