A Real Time and Interactive Dashboard in Tourism Industry

Project Plan

By

Junyu Zhou 592115508

Yawei Li 592115518

Department of Software Engineering,

College of Arts, Media and Technology,

ChiangMai University

Project Advisor



Table of Contents

[1. Document History 4](#_Toc13523636)

[2. Introduction 5](#_Toc13523637)

[2.1 Identification 5](#_Toc13523638)

[2.2 Project Overview 5](#_Toc13523639)

[2.3 Purpose 5](#_Toc13523640)

[2.4 Work Product to be developed 5](#_Toc13523641)

[2.4.2 Non-Deliverables 7](#_Toc13523642)

[2.5. Acronyms and Definitions 7](#_Toc13523643)

[3. Infrastructure 10](#_Toc13523644)

[3.1 Development Tools 10](#_Toc13523645)

[3.2 Hardware and Material Resources 10](#_Toc13523646)

[4. Management Procedures 11](#_Toc13523647)

[4.1 Project Team Structures 11](#_Toc13523648)

[4.2 Monitoring and Controlling Mechanisms 12](#_Toc13523649)

[4.2.2 Software Development Process Model 13](#_Toc13523650)

[5. Quality Standard 14](#_Toc13523651)

[5.1 ISO 29110 for Very Small Entity (VSE) 14](#_Toc13523652)

[5.2 Project Management Process 14](#_Toc13523653)

[5.3 Software Implementation Process 16](#_Toc13523654)

[6. Quality Planning 18](#_Toc13523655)

[6.1 Reviews/Responsibility 18](#_Toc13523656)

[6.2 Testing 19](#_Toc13523657)

[7. Estimated Duration of Tasks 20](#_Toc13523658)

[8. Software Configuration Management 21](#_Toc13523659)

[8.1 Naming Convention 21](#_Toc13523660)

[8.2 Change Management 21](#_Toc13523661)

[8.3 Project Repository 22](#_Toc13523662)

[9. Risk Management 24](#_Toc13523663)

[9.1 Risk Identifications and Solutions 24](#_Toc13523664)

[10. Schedule 25](#_Toc13523665)

# 1. Document History

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| History | Status | Date | Viewable | Editable | Responsible |
| ProjectPlan\_v1.docx  Create:  - user requirement specification  - system requirement specification | Draft | July 7, 2019 | ZJY, LYW,  AJP | ZJY, LYW | ZJY,  LYW |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

ZJY = Junyu Zhou

LYW = Yawei Li

AJP = Dr. Pree Thiengburanathum

# 2. Introduction

## 2.1 Identification

This Project Management Plan is the document for planning, scheduling activities and evaluating overall of the project so that it could complete successfully despite all the risks. It provides detailed scheduled, assigned task and identified risks of the project.

## 2.2 Project Overview

A real time and interactive dashboard in tourism industry is developed for the decision maker to view and manage the data easily and efficiently. Due to a large number of data, there are numerous work and extra things to do for doing statistics. A real time and interactive dashboard in tourism industry will provide the effective platform to decision maker to manage and statistics the massive data.

## 2.3 Purpose

A real time and interactive dashboard in tourism industry is a web-based application. It is for decision maker to do the easy statistics of mess data in a real time way. For decision maker to manage and keep tracks all the data.

## 2.4 Work Product to be developed

**2.4.1 Deliverable**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Deliverables/Release | Media | Copies | Date |
| 1 | **Project Proposal**  Proposal Version | Hard Copy | 3 | July 7, 2019 |
| 2 | **Progress Report I**  Project Management Plan  Software Requirement Specification  Software Design Document  Test Plan  Traceability Record  Application Demo | Hard Copy | 3 | July 8, 2019 |
| 3 | **Progress Report II**  Project Management Plan Version 2.0  Software Requirement Specification  Version 2.0  Software Design Document Version 2.0  Test Plan Version 2.0  Traceability Record Version 2.0  Software Version 0.2 | Hard Copy | 3 |  |

## 2.4.2 Non-Deliverables

|  |  |  |
| --- | --- | --- |
| No. | Deliverables/Release | Media |
| 1 | 10 mins presentation | File |
|  |  |  |
|  |  |  |

## 2.5. Acronyms and Definitions

**2.5.1 Acronyms**

ZJY = Junyu Zhou

LYW = Yawei Li

AJP = Dr. Pree Thiengburanathum

SRS = Software Requirement Specification

SDD = Software Design Document

OS = Operating System

SE = Very Small Entity

PM = Project Management

SI = Software Implementation

IID = Iterative and Incremental Development

SCI = Software Configuration Item

**2.5.2 Definitions**

|  |  |
| --- | --- |
| Name | Definition |
| Acceptance test | Test activities for sample checks to verify that the system (or product, solution) has the right quality for deployment or usage. Often acceptance test is done by the customer. [IEEE90] |
| Feature | Transformation of input parameters to output parameters based on a specified algorithm. It describes the functionality of the product in the language of the product. Used for requirements analysis, design, coding, testing or maintenance. [IEEE90] |
| IEEE | Institute for Electrical and Electronics Engineers. Biggest global interest group for engineers of different branches and computer scientists. [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 is to document planning assumptions and  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 or occurrence of a given threat’s occurrence. [IEEE90] |

# 3. Infrastructure

## 3.1 Development Tools

PyCharm

Visual Studio Code

MongoDB

Pusher

React

Flask

GitHub

Draw.io

## 3.2 Hardware and Material Resources

**Computer:**

MacBook Pro (Retina, 15-inch, Mid 2015)

Processor: 2.2 GHz Intel Core i7

Memory: 16 GB 1600 MHz DDR3

OS: MacOS Catalina 0.15 Beta (19A501i)

Dell (Inspiron15-7557)

Processor: Intel Core i7-4720HQ

Memory: 8 GB

OS: Windows 10

# 4. Management Procedures

## 4.1 Project Team Structures

|  |  |
| --- | --- |
| Participants | Activities |
| ZJY,  LYW | Feasibility Study  Project Proposal  Project Requirements  Project Plan  Project Design  Implementation  Record Changes  Testing |

## 4.2 Monitoring and Controlling Mechanisms

**4.2.1 Project Meeting**

|  |  |
| --- | --- |
| Participants | Roles |
| ZJY,  LYW | Development team member |
| AJP | Project advisor |

## 4.2.2 Software Development Process ModeC:\Users\42916\Desktop\Iterative Model.pngl

*Figure1-Iterative Model*

The iterative model is an implementation of a software development life cycle that focuses on an initial, simplified implementation, which then progressively gains more complexity and a broader feature set until the final system is complete. In short, iterative development is a way of breaking down the software development of a large application into smaller pieces.

# 5. Quality Standard

## 5.1 ISO 29110 for Very Small Entity (VSE)

ISO 29110[16] is a guide applies to a Very Small Entity (VSE), enterprise, organization, department or project up to 25 people, dedicated to software development. The Guide provides Project Management and Software Implementation processes which integrate practices based on the selection of ISO/IEC 12207- Systems and Software Engineering —Software Life Cycle Processes and ISO/IEC 15289 Software Engineering – Software Life Cycle Process – guidelines for the content of software life cycle process information products (documentation) standards elements.

## 5.2 Project Management Process

**Purpose:**

The purpose of the software management process is to establish and carry out in a systematic way the task of the software implementation project that allows complying with the project's objectives in the expected quality. Time and cost.

**Objective:**

**PM. 01.** 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.02.** 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.03.** The Change Requests are addressed through their reception and analysis. Changes to software requirements are evaluated for cost, schedule and technical impact.

**PM.04.** Review meetings with the Work Team and the Customer are held. Agreements are registered and tracked.

**PM.05.** Risks are identified as they develop and during the conduct of the project.

**PM.06.** A software Version Control Strategy is developed. Items of Software Configuration are identified, defined and baseline. 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.07**. Software Quality Assurance is performed to provide assurance that work products and processes comply with the Project Plan and Requirements Specification.

**Activities**

**PM.01.**  Project Planning

**PM.02.** Project Plan Execution

**PM.03.** Project Assessment and Control

**PM.04.** Project Closure

## 5.3 Software Implementation Process

**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.

**Objectives:**

**SI.01.** Tasks of the activities are performed through the accomplishment of the current Project Plan.

**SI.02.** Software requirements are defined, analyzed for correctness and testability, approved by the Customer, baseline and communicated.

**SI.03.** Software architectural and detailed design is developed and baseline. It describes the software items and internal and external interfaces of them. Consistency and traceability to software requirements are established.

**SI.04.** 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.05.** 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.06.** A Software Configuration that meets the Requirements Specification as agreed to with the Customer, which includes user, operation and maintenance documentations is integrated, baseline and stored at the Project Repository. Needs for changes to the Software Configuration are detected and related Change Requests are initiated.

**SI.07.** 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.

**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

# 6. Quality Planning

|  |  |  |  |
| --- | --- | --- | --- |
| Stage Exit Review | | | |
| No. | **Stage** | **Review Item** | **Responsibility** |
| 1 | Project Planning | Project plan | ZJY, LYW |
| 2 | Requirement Specification | Software Requirement Specification | ZJY, LYW |
| 3 | Architecture and Detailed Design | Software Design Document | ZJY, LYW |
| 4 | Implementation | Code | ZJY, LYW |
| 5 | Software Testing | Test Plan | ZJY, LYW |
| 6 | Software Testing | Test Record | ZJY, LYW |
| 7 | Project Monitoring and Control | Traceability Record | ZJY, LYW |

## 6.1 Reviews/Responsibility

## 6.2 Testing

|  |  |  |
| --- | --- | --- |
| Stage Exit Review | | |
| No. | **Test** | **Responsibility** |
| 1 | Unit Testing | ZJY, LYW |
| 2 | System Testing | ZJY, LYW |

# 7. Estimated Duration of Tasks

|  |  |  |
| --- | --- | --- |
| Task and Estimated Duration | | |
| No. | **Phase** | **Estimated Duration (Days)** |
| 1 | Proposal | 20 |
| 2 | Progress I | 60 |
| 3 | Progress II | 62 |
| 4 | Final Progress | 64 |
| Total | | 206 |

# 8. Software Configuration Management

## 8.1 Naming Convention

For the filename format that we use for all documents is:

Project- [File Name] \_v[Version]. [File Type]

* File Name - This part will depend on substance of that file.
* Version - This part is the version of file. Version number will be in this format “[Main version]. [Sub version]”
* File Type - This part is type of file.

(I.E. Project- ProjectPlan\_v1.0.docx)

## 8.2 Change Management

Change management manages all of the changes in the project during development process. All of the change requests will be record into the change record document.

The procedures for managing changes are:

1. Discuss with advisor about the change.

2. Record the change information to change document.

3. Send the change request to advisor.

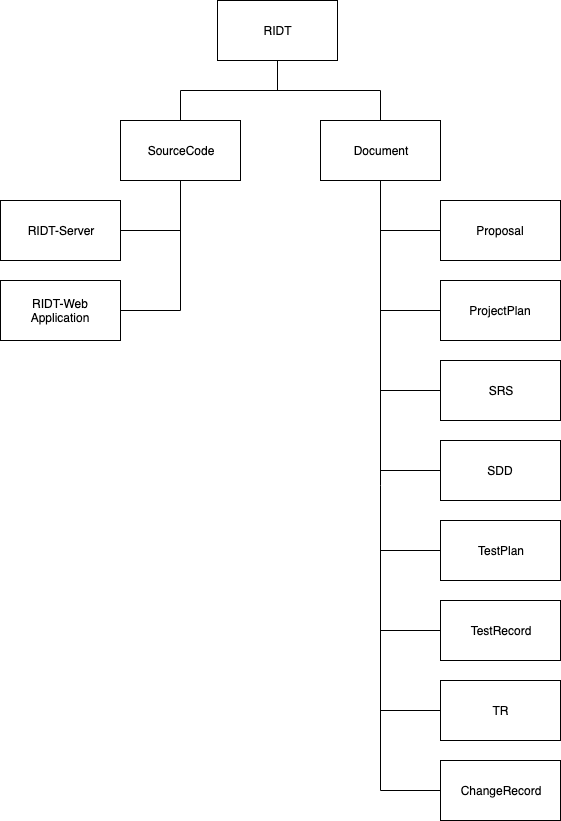
3.1 Request accepted: change document and software follow the change information.

3.2 Request rejected: continue on and find alternative solution.

## 8.3 Project Repository

This project uses “GitHub” to manage the version of document and software. It can be used to store and share code or binary files for software development projects.

The repository’s directories will be created as following:



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No. | Item | File Name | Type | Owner | Path | Version |
| 1 | Project  Proposal | Project-Proposal\_v8.docx | .docx | ZJY, LYW | /RIDT/Documents/Proposal | 8.0 |
| 2 | Project plan | Project-ProjectPlan\_v1.docx | .docx | ZJY, LYW | /RIDT/Documents/ProjectPlan | 1.0 |
| 3 |  |  |  |  |  |  |
| 4 |  |  |  |  |  |  |
| 5 |  |  |  |  |  |  |
| 6 |  |  |  |  |  |  |
| 7 |  |  |  |  |  |  |
| 8 |  |  |  |  |  |  |
| 9 |  |  |  |  |  |  |

**8.4 Software Configuration Table**

# 9. Risk Management

## 9.1 Risk Identifications and Solutions

|  |  |  |
| --- | --- | --- |
| No. | Risk statement | Solution |
| 1 | The developer lacks necessary skills for project development | - Study from textbook or online resources. |
| 2 | The requirements might change. | - Make change request form and discuss with advisor to reprioritize the changed requirements. |
| 3 | The process flow might not keep up with the project schedule. | - Start tasks execution before their schedule.  - Prioritize tasks and to do more important works first. |

# 10. Schedule

图片包含 音乐

描述已自动生成





**