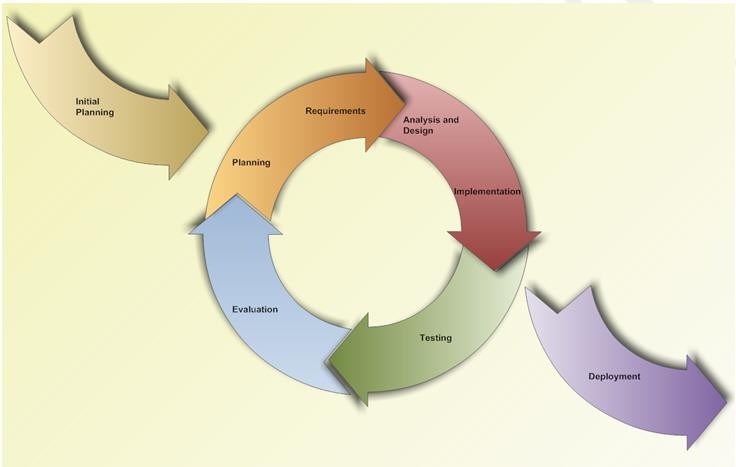
# SOFTWARE PROJECT MANAGEMENT PLAN

In this section, we will describe our organization and plan of project. Each member of team must use this section as a guideline tracking assigned tasks and deadlines. In addition, we introduce our team and overview about project. Daily meeting and meeting minute also present in this section. Moreover, we will describe risks of project and the way communication of our team.

## 2.1 Project organization

### 2.1.1 Software Process Model

This figure below will describe the information and products flow lifecycle process model.



*Figure 2-1 Iterative and Incremental software process model*

MANAPART project uses the Iterative and Incremental Software Process Model. We have chosen the Iterative and Incremental Software Process Model because the scope of our project is big, the major requirements were defined clearly, some more detail will be added in time, and for the newbie group in software development. By using this software process model, we generates working software quickly and early during the software life cycle. This model is more flexible – less costly to change scope and requirements. We easier to manage risk because risky pieces are identified and handled. Besides, the iterative model is easier than other models when the issues are discovered. All issue will be raised to the team. They will be solved while the project is still developed.

In our project, we have two cycles:

Cycle 1: Start Monday 09st – May -2016 to Wednesday 10st – August – 2016. In this iterative we will to survey, analyze, design, code and test the main feature of project.

Cycle 2: Start Thursday 11th – August – 2016 to Friday 26th – August – 2016. In this iterative we will to fix bug in iterative 1, complete other feature, test again end packing product.

### 2.1.2 Roles and Responsibilities

#### Project Team Member

|  |  |
| --- | --- |
| **Team Member** | **Role** |
| LongCQ | Project Manager, Developer, Requirement Analyst |
| HuyTLN | Technical Leader, Developer, Tester |
| TrungVD | Technical Leader, Developer, Tester |
| SonNC | Requirement Analysist, Developer, Tester Leader |
| TuanHV | Requirement Analysist, Developer, Tester, Designer Leader |

#### Role & Responsibilities

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Member** | **Role** | **Responsibilities** |
| 1 | Chu Quang Long | Project Manager,  Developer,  Requirement Analyst | * Planning and defining scope * Developing schedule * Write and Manage document * Identify resources and assign individual responsibilities * Assign task team * Coordinating communication * Developing |
| 2 | Tran Le Ngoc Huy | Technical Leader,  Developer,  Tester | * Assign task code team * Developing * Manage configuration * Framework * Deployment * Solve issue of project * Document * Testing |
| 3 | Vu Dang Trung | Technical Leader,  Developer,  Tester | * Developing * Manage configuration * Framework * Deployment * Solve issue of project * Document * Testing |
| 4 | Nguyen Cong Son | Developer,  Tester Leader  Requirement Analysist | * Developing * Responsible for test execution * Test results recording * Evaluation of test run and error recovery * Developing |
| 5 | Ho Viet Tuan | Developer,  Tester,  Designer Leader,  Requirement Analysist, | * Developing * Responsible for test execution * Test results recording * Evaluation of test run and error recovery * Developing * Designer |

#### 2.1.2.3 Tools and Techniques

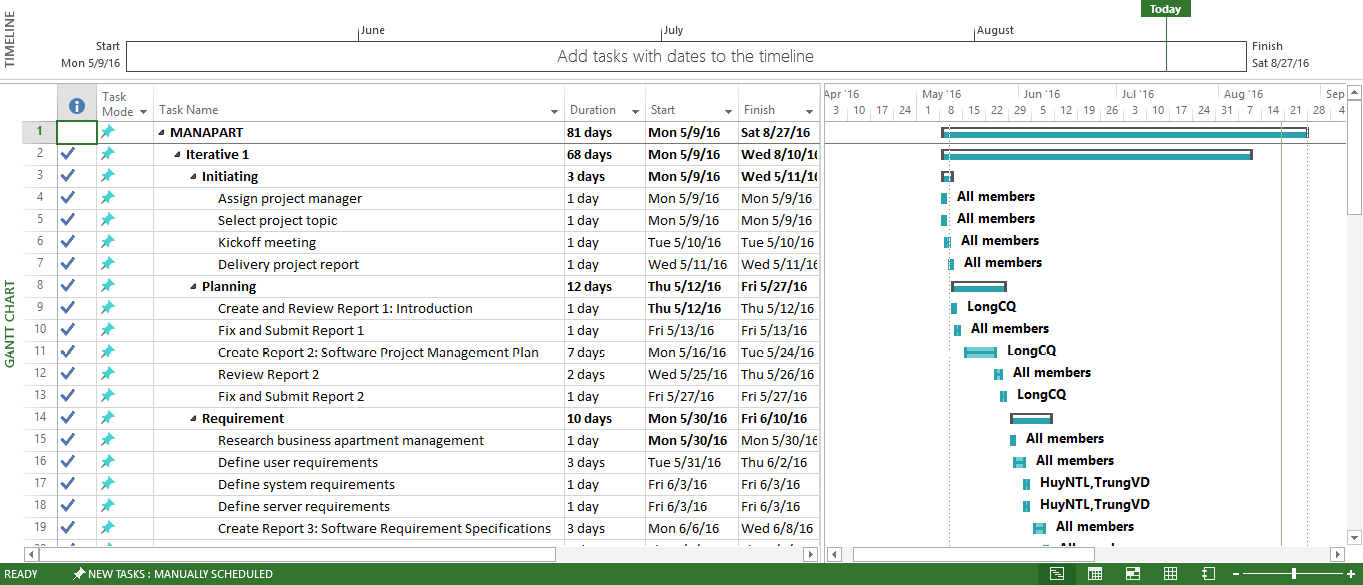
|  |  |
| --- | --- |
| Programming languages | Ruby, HTML, CSS, Javascript |
| Framework | Ruby on Rails, Bootstrap, AngularJS |
| Version Control | Github, Bitbuket(Back up) |
| IDEs | Vim, SublimeText, Cloud9 |
| DBMS | PostgreSQL |
| Deployment tools | Heroku |
| Communication tools | Slack |
| UML tools | Astah Professional 7.0 |
| Management tools | Microsoft Project Professional 2013, Bitrix24 |
| Web server | Ubuntu OS 14.04 |
| Cloud Service | Azure Web Service |
| Other | Microsoft Office 2013 |

## Project management plan

### 2.2.1 Tasks schedule

Iterative 1: Start Monday 09th – May -2016 to Wednesday 10th – August – 2016.

Iterative 2: Start Thursday 11th – August – 2016 to Friday 26th – August – 2016.



*Figure 2-2 Work Breakdown Structure*

Refer file MANAPART\_WBS.mpp to view details

### 2.2.2 Meeting Minutes

In all meeting minutes will be written follow this template:

**Meeting Minutes 1**

|  |  |  |  |
| --- | --- | --- | --- |
| **Subject** | Comment on Report 1 | **Date** | 5/11/2016 |
| **Conductor** | Prof. AnhBN | **Time** | 8 A.M |
| **Location** | P306 | **Secretary** | SonNC |
| **Attendees** | SonNC, HuyTLN, TuanHV | | |
| **Absent** | TrungVD | | |

| **Key Points Discussed** | | |
| --- | --- | --- |
| No. | Topic | Highlights |
| 1 | Common work progress | Done writing report 1, report 2 in progress |
| 2 | Assigning someone to write meeting minutes.  Submit time: no more than 2 days after metting. | Assign SonNC |
| 3 | Review report 1 | Something needs to be fixed:   * Add MANAPART to definitions * Incorrect grammar, vocabulary * Add interact with residents function, residents can see notify, warning, schedule, fee, … * Think about how performers ground |
| 4 | Server environment, coding convention, drawing a graph, estimated workload | * Server environment is ok * All of rest is in progress |
| 5 | Learn the business | Done |
| 6 | Submit report 2 | In progress |

| **Action Plan** | | | |
| --- | --- | --- | --- |
| No. | Action Item(s) | Owner | Target Date |
| 1. | Define the roles & responsibilities of each team member | LongCQ | 5/12/2016 |
| 2. | Make plan and assign to each team member for doing the first stage (propose for: Category, Appointment management) include:  2.1. Analyze => requirement written in template format  2.2. Design: ERD, System architecture, choose programming model, Class diagram, Sequence diagram  2.3. Development:  2.3.1. Setup programming environments, sync code, Application server, DB server, coding convention  2.3.2. Implement following 2.1 and 2.2 | LongCQ + SonNC+ TrungVD + HuyTLN | 5/15/2016 |
| 3. | Estimate number of use case, 60 use case is appropriate | TuanHV + HuyTLN | 5/13/2016 |
| 4 | Gen sample data, use <http://generatedata.com/> | TuanHV | 5/20/2016 |
| 4. | Write the report 1, 2 | SonNC | 5/12/2016 |

### 2.2.3 Coding Convention

The following rules follow the standard rule of developing applications using Ruby on Rails Framework. It refers to the coding convention on websites:

<https://github.com/bbatsov/ruby-style-guide>

### 2.2.4 Risks management

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Name** | **Probability** | **Prevention** | **Correction** | **Impact** |
| 1 | Member has problem for health or absence | Medium | Team members should warn their team leader timely before a planned period of absence. Team leader has plan to assign task to member again | By ensuring that knowledge is shared between team members, work can be taken over quickly by someone else if a person gets ill. | High |
| 2 | Member has conflicts | Medium | Clear the role and responsibility. | Manager must find cause the conflict and solve them. Team building every week and have meeting if member have conflicts. | Medium |
| 4 | Requirement changed | Low | Carefully brainstorm system’s features among team members. Regularly hold meeting to define and discuss all features of systems. Design system carefully. Analyze all the possible cases to minimize the change. | Team meetings with supervisor to determine which features is suitable for implementation  Team leaders create implementation plan for implemented features and assign to team members. | High |
| 5 | Data or Source Code Lost | Medium | Backup database automatically daily. Use Github for version control of source code and need to commit when small task completed | Restore data from backed up data. Restore source code from history versions | High |
| 6 | Over Deadline | Medium | Assign tasks to suitable team members depending on their strength and weakness. Define punishment for team member who miss deadlines. | Find the root cause of the problems.  Assign task to other team member.  Break down task to smaller tasks and assign new tasks to team members. | Medium |
| 7 | Lack of knowledge about technology | Medium | Each team member must understand the technology, framework, ides, which we using Encourage team members to study on themselves for their technical personal needs | Technical leader assists team member personally or in group to develop required skills and knowledge. | Medium |

### 2.2.5 Communication plan

#### 2.2.5.1 Communication between Team Members

* Weekly meeting schedule: MANAPART will have meeting every Wednesday. We present the task of each person and another members comment it. If there is any issue, we will discuss and find solution together. If it is too difficult and can’t be solved by ourselves, we will ask our supervisor for advices.
* Daily report: each member must write daily report. We update the tasks status (complete, ongoing, assign) in task list (It’s on Trello). All member can see how tasks is and when they complete.
* Unscheduled meeting: if someone has an important problem want to be solved immediately, we will have meeting for discussion.
* Communication channel: Our main communication channels are face-to-face meeting, email, phone, group on Facebook and Slack.

#### 2.2.5.2 Communication with supervisor

* Face-to-face meeting: weekly on every Wednesday afternoons to make sure that supervisor can keep tracking of the team’s progress.
* Supervisor can to tracking progress on the tasks list. It is updated at 9:00 pm every day.
* Group on Facebook, email: It’s the fastest way to get advice and document checking from supervisor.
* Phone: is used to get time and place arranged for the meeting every weeks.

### 2.2.6 Project millstones

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Millstone** | **Completion Date** | **Verification** |
| 1 | Project start | May 09th, 2016 | Supervisor |
| 2 | Submit report 1 | May 26th, 2016 | Supervisor |
| 3 | Submit report 2 | June 03rd, 2016 | Supervisor |
| 4 | Submit report 3 | June 17th, 2016 | Supervisor |
| 5 | Submit report 4 | July 01st, 2016 | Supervisor |
| 6 | Submit report 5 | July 11th, 2016 | Supervisor |
| 7 | Submit report 6 | August 19st, 2016 | Supervisor |
| 8 | Final report | August 20st, 2016 | FPT University , Supervisor |
| 9 | Defend project | August 31st, 2016 | FPT University , Supervisor |