

|  |
| --- |
|  |

|  |
| --- |
| **FPT UNIVERSITY** |
| **Project Plan** |
| Internet-connected Devices Checking System |
|  |
| |  |  |  | | --- | --- | --- | | **IDCS TEAM** | | | | **Group Members** | Nguyễn Quý Đôn | SE04468 | | Nguyễn Quý Tuấn | SE04330 | | Phạm Công Minh | SE04098 | | **Supervisor** | Lecture : Phan Duy Hùng | | |
|  |

|  |
| --- |
| - Hanoi, 10/2019 - |

1. ***Introduction***

This document provides an overview of the project plan, including software model, project organization and project schedule plan. Moreover, the document also has communication management, configuration management and risk management.

All team member must use this chapter as a guideline for tracking assigned task and deadlines.

1. ***Project Organization***
   1. ***Project Description***

|  |  |
| --- | --- |
| Project Name : | Internet-connected Devices Checking System |
| Project Code : | IDCS |
| Project Type : | Web Application Platform |
| Project Category : | Checking Security System |
| Business Type : | Online Customer Services |
| Project Instructor : | Phan Duy Hùng |
| Project Manager : | Phạm Công Minh |
| Time line : | 10/9/2019 – 24/12/2019 |

In this project, we will develop a Web application will named Internet-connected Devices Checking System, which help a customer to check open port, services running in to server and vulnerability if have. We also give solution based on vulnerabilities you have.

* 1. ***Scope***

The scope of this project includes these stages:

* Develop requirement team made and software requirement specification.
* Develop architecture and detailed design document.
* Coding and unit testing.
* Deployed the application in server.
* Develop test case and execute combination test

The application has these main function

* Check port, services, security check
* Export and Support user
  1. ***Standard Objectives***
* Project must be finished before 31 December 2019
* All team member give best effort to complete this project
* The final application covers more than 100% of requirement
  1. ***Milestone and deliverables***

|  |  |  |
| --- | --- | --- |
| *No* | *Milestone* | *Delivery Date* |
| 1 | Project Registration | 09/08/2019 |
| 2 | Submit Report No.1 | 23/09/2019 |
| 3 | Submit Report No.2 | 07/10/2019 |
| 4 | Submit Report No.3 | 24/10/2019 |
| 5 | Submit Report No.4 | 29/10/2019 |
| 6 | Submit Report No.5 | 13/12/2019 |
| 7 | Submit Report No.6 | 17/12/2019 |
| 8 | Submit Final Report | 18/12/2019 |
| 9 | Submit Project Resource | 25/12/2019 |
| 10 | Thesis Defense | 03/01/2019 |

1. ***Project Organization***
   1. ***Purpose***

This chapter provides an overview of the project plan, including software process model, project organization and project management plan. All team members must use this chapter as a guild line for tracking assigned task and deadlines. This chapter also included an overview of this project and team members.

* 1. ***Software Process Model***

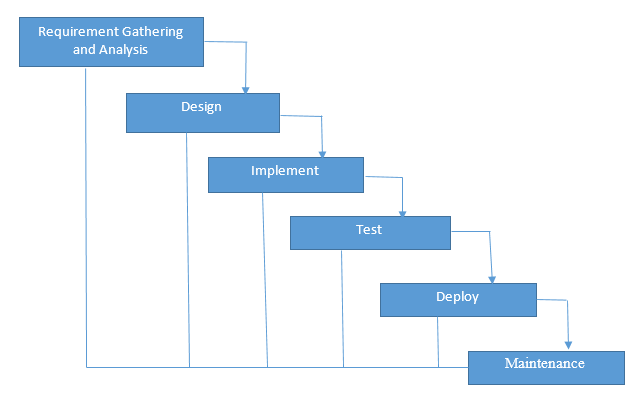


Figure 1

Figure 1-1 Waterfall Software Process Model (image: Internet)

Our Project uses The Waterfall Software Process Model.

The Waterfall Model is a linear or sequential approach to project management and works based on fixed dates, requirements, and outcomes. Teams do not require consistent communication and, unless specific integrations are required, can be self-contained. Team member can also work independently and are often required to provide status reports somewhat less frequently.

The Project scope stay relatively static, meaning cost and timelines can be determined early on in this project. A structure approach to a project means that everyone understands what needs to be done and when.

* 1. ***Roles and Responsibilities***

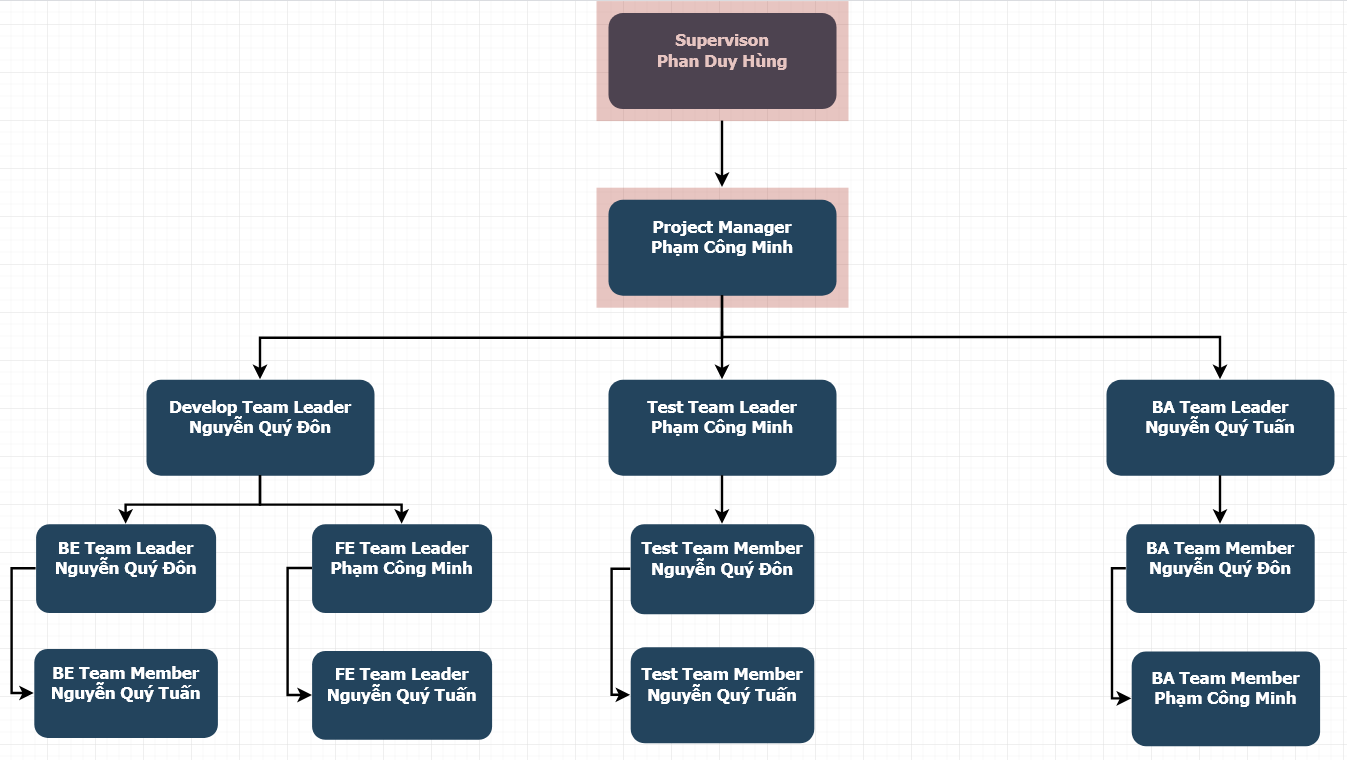


Figure 2

* + 1. ***Organizational Structure***

|  |  |
| --- | --- |
| *Role* | *Responsibility* |
| Project Manager | Responsible for leading the team and managing the whole project, planning, defining scope, developing schedules, coordinating communication, analyzing and managing risks. |
| Technical Leader | Responsible for choosing and deciding what technologies should be used, as well as for overseeing the work being done by other developers. |
| Business Analyst | Responsible for analyzing business, processes and systems. |
| Tester | Responsible for conducting tests. |
| Back-end Developer | Responsible for server-side web application logic and integration of the work front-end developers do. |
| Front-end Developer | Responsible for implementing visual elements that users see and interact with in a web application. |

Table 1

* + 1. ***Project Team Member***

|  |  |
| --- | --- |
| *Team Member* | *Roles* |
| Phạm Công Minh | Project Manager, Front-end Team Leader, Tester, Business Analyst, Designer |
| Nguyễn Quý Đôn | Back-end Team Leader, Technical Leader, Tester, Business Analyst, Designer |
| Nguyễn Quý Tuấn | Front-end Back-end Developer, Tester, Business Analyst, Designer |

Table 2

1. ***Project Management Plan***
   1. ***Project Schedule***

To deploy this project effectively, according to this schedule, we will follow the each tasks in project

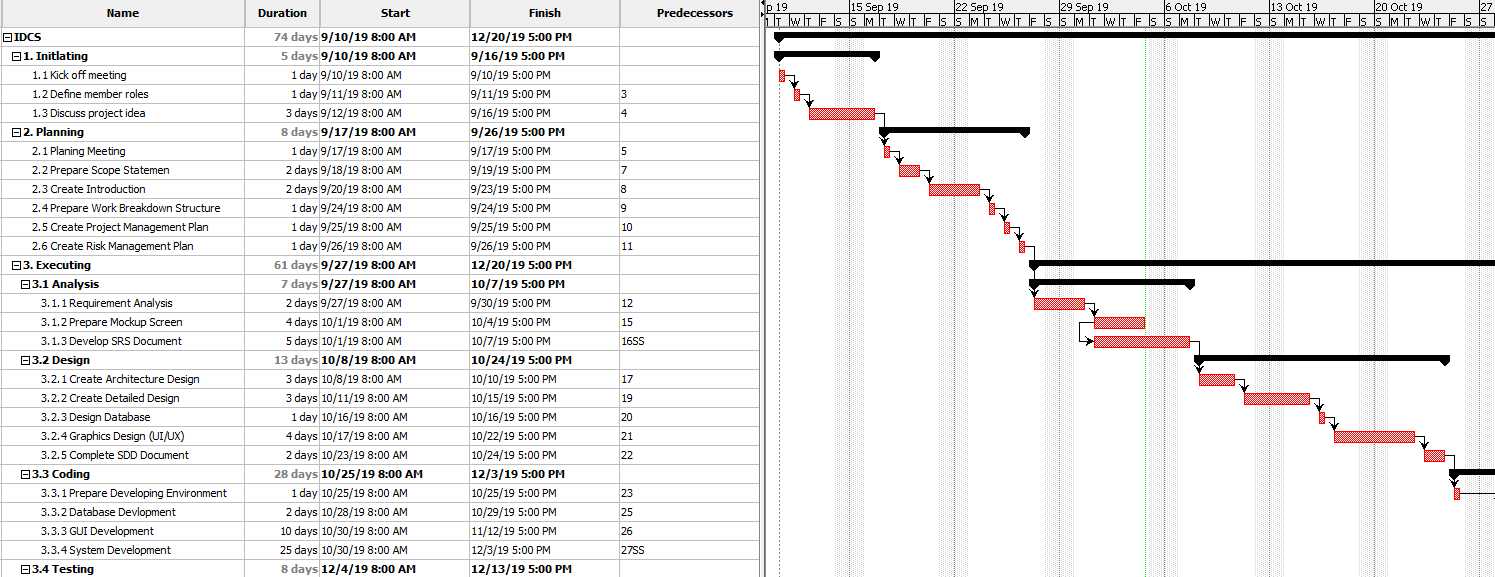


Figure 3

* 1. ***Communication Management***
     1. ***Communication between members***

**Weekly meeting schedule:**

Team will have a meeting every Saturday to report the process of whole team’s current task and assign new task. If have 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 advises.

**Unscheduled meeting:**

If someone has an important problem want to be solved immediately, we will have a meeting with social network like Skype, FaceTime because we are apart.

**Communication channel:**

Our main communication channels are Facebook Messenger, Email, Skype, Trello, face-to-face meeting. However, sometimes can make a phone call or instant message if someone has problem.

* + 1. ***Communication with supervisor***

**Face-to-Face meeting:**

Weekly on every Thursday afternoon to make sure that supervisor can keep tracking of the team’s progress.

**E-mail:**

Gmail is the faster way to get advice and document checking form supervisor.

**Mobile Phone:**

Is used to get time and place arranged for the meeting every weeks

* + 1. ***Meeting Plan***
       1. ***Meeting minutes***

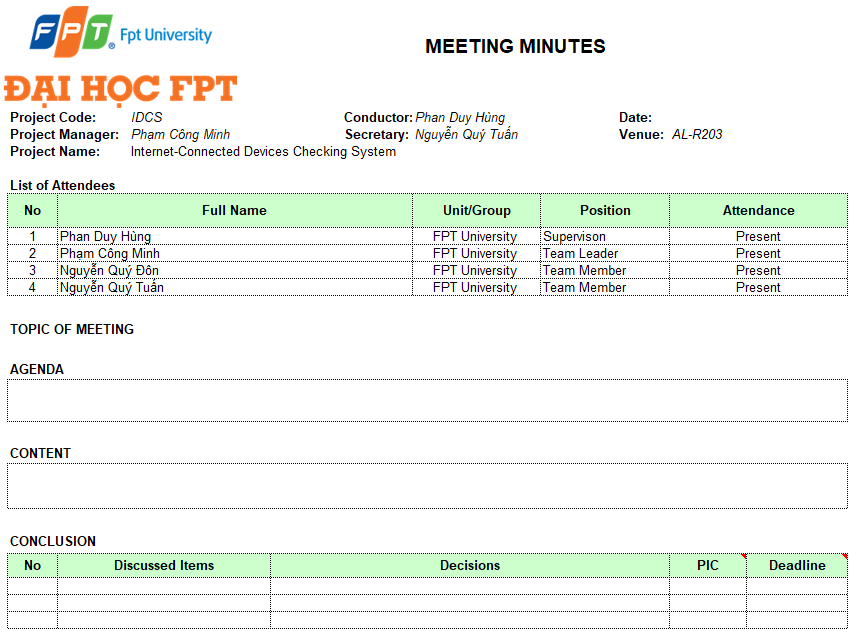


Table 3

* + - 1. ***Progress Report***

Project Progress reports are delivered Supervisor every week. Below is the sample of our progress report:

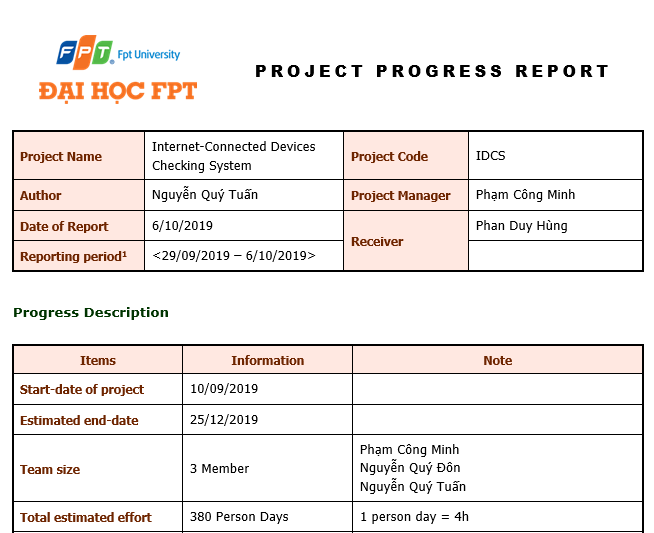


Figure 4

* 1. ***Coding Conventions***
     1. ***Front end***

Folder/files naming:

* Written in “camelCase” and “PascalCase”, for example: “Components”, “Test”, and “login.js”. With some sub-class or sub-folder, we will use” underscore”, for example: “login\_s”, “home\_result\_s”.

Javascript Style Guide:

* We strictly follow javascript Standard Style Guide, with linter and automatic code fixer. Please refer to the official website at <https://standardjs.com/rules-en.html>
  + 1. ***Back end***

Folder/files naming:

* Written in “camelCase” and “PascalCase”, for example: “Controller”, “Model”, and “LoginController.java”.

Java Style Guide:

* We strictly follow Java Coding Convention with linter and automatic code fixer. Please refer to coding convention form oracle at : <https://www.oracle.com/technetwork/java/codeconventions-150003.pdf>

Python Style Guide:

* We strictly follow Python Coding Convention with linter and automatic code fixer. Please refer to official coding convention website at : <https://www.python.org/dev/peps/pep-0008/>
  1. ***Tools and Techniques***
     1. ***Tools***

|  |  |
| --- | --- |
| **Project Management tools** | Trello, Microsoft Project 2016, ProjectLibre |
| **Document tools** | Microsoft Office 2016 (Word,Excel,PowerPoint) |
| **UML tools** | Draw.io, lucidchart.com, |
| **Ides** | Notepad ++, NetBean 8.2, IDLE |
| **DBMS** | Microsoft SQL Server 2016 |
| **Source code version control** | GitHub, SourceTree, Git SCM 2.19.1 |
| **Testing tool** | Postman 6.4.4, TestNG |
| **Communication tools** | Facebook Group, Messenger, Email, Trello |
| **File Management tools** | GitHub, Google Driver, DropBox |

* + 1. ***Techniques***
       1. ***Front end***
* Programming languages: JavaScript, HTML, CSS
  + - 1. ***Back end***
  1. ***Risk and Issue Management Plan***
     1. ***Issue Management***

The table below is some issues has been met in making project. Besides the issues, we have solution with each issue

|  |  |  |
| --- | --- | --- |
| No | Name issue | Solution |
| 1 | Conflict between member | The Project Manager has to find out the root that caused problem and solve it. Face-to-face meeting is required method. Voting is also good method |
| 2 | A team member’s absence. | Motivate member and create positive work environment. Other member replace that team member’s work |

* + 1. ***Risk Management***
       1. ***Risk Categories***

The table below is some Risk Categories may be had in this Project.

|  |  |  |
| --- | --- | --- |
| Category | Sub-Category | Acronym & Abbreviation |
| Technical | Requirement Definition | T-RD |
| Technology | T-T |
| Complexity and Interfaces | T-CI |
| Performance and Reliability | T-PR |
| Quality | T-Q |
| Management | Estimating | M-E |
| Human Resources | M-HR |
| Communication | M-C |
| Source | M-S |
| Controlling | M-C |

Table 4

* + - 1. ***Risk Register***

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Name | Risk | Category | Root Cause | Probability | Impact |
| 1 | R1 | Team don’t meet the deadline on time. | M-HR | Members lack of responsibility, lack of Management | High | High |
| 2 | R2 | Lack of knowledge about technology needs for project | T-T | Depend on requirement, new technology should be applied in the project, which some team members haven’t used before or training members before use new technology | Medium | High |
| 3 | R3 | The morale of work goes down | T-HR | Conflict among team members/ working alone ,meet a difficult problem | High | Low |
| 4 | R4 | Team member misunderstands requirement | T-RD | In the beginning, team member doesn't clear the requirement | Low | High |

* + - 1. ***Risk Respond***

Risk in project cannot be avoid, so we must respond with risk suitably to continue the project conveniently. The table below is some risk respond which team will respond to risk.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Mitigation Plan | Contingency Plan | Fallback Plan | Status |
| 1 | R1 | Make rules and penalties for member who misses the deadline. | Working overtime to complete tasks on time. | Reduced time of another task to require | Active/Solved |
| 2 | R2 | Each team members must study to understand the technology, framework that is needed for project | Technical Leader needs to support carefully for team member in group to increase required skill and knowledge. Review complete task to make comment for team members | Discuss with supervisor about technical, may change to another easy technical or not | Active/Solved |
| 3 | R3 | Members have to read requirement specification and related document carefully. | Making sure that any miscommunication would be resolved | All team members will take a time together to fix requirement and adjust project on the right way | Active/Solved |
| 4 | R4 | Warm-up by teambuilding | Project Manager should talk to a member who gets the problem and helps that member solves the problem | Giving that member’s problems for the supervisor solve the solution | Active/Solved |

* + - 1. ***Risk Probability and Impact***

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Probability | |  |  |  |  | | --- | --- | --- | --- | | High | R3 |  | R1 | | Medium |  |  | R2 | | Low |  |  | R4 | |  | Low | Medium | High | |
|  | Impact |

The probability and impact of occurrence for each identified risk will be assessed by the project manager, with input from the project team using the following approach:

**Probability:**

* **High -** Likely or very likely to occur.
* **Medium -** May occur about half the time.
* **Low -** Very unlikely and unlikely to occur.

**Impact:**

* **High -** Will jeopardize the project or bring the project to a halt
* **Medium -** Will impact the project in terms of timeline, cost, quality etc. However, the project will still move forward
* **Low -** Minimal impact to the project, not critical to project deliverables
  + - 1. ***Watch List***

Risk could be closed in six case:

* When the time of a risk happening is over
* When the scope of a project is amended and risk becomes irrelevant (avoided).
* When a risk is addressed well and reduced to acceptable level.
* Risk closure at the direction of the Project Manager
* When risks become no longer relevant to the project.
* When the risk is considered successfully mitigated.