

STUDENT REGISTRATION SYSTEM

PART 1: PROJECT PLAN

COURSE: SOFTWARE ENGINEERING

Fall 2019

CLASS: MONDAY MORNING (LABORATORY)

SATURDAY MORNING (THEORY)

INSTRUCTOR: Ms. N.T.T.LOAN

**Member:**

Trịnh Minh Đăng ITITIU16022

Nguyễn Đức Phi Hồng ITITIU17022

Võ Huy Thành ITITIU17026

|  |
| --- |
| **MEMBER CONTRIBUTION** |

|  |  |  |
| --- | --- | --- |
| **NAME** | **ID** | **CONTRIBUTION** |
| **Nguyễn Đức Phi Hồng** | **ITITIU17022** | **III. ORGANIZATION**  **IV. RESOURCE REQUIREMENTS**  **I. OVERVIEW**  **II. GOALS AND SCOPE**  **IX. ABBREVIATIONS AND DEFINITIONS**  **X. REFERENCES** |
| **Trịnh Minh Đăng** | **ITITIU16022** | **V. SCHEDULE**  **VII. DELIVERY PLAN** |
| **Võ Huy Thành** | **ITITIU17026** | **VI. RISK MANAGEMENT**  **VIII. SECURITY ASPECTS** |

**OUTLINE**

|  |
| --- |
|  |

I. OVERVIEW

II. GOALS & SCOPE

III. ORGANIZATION

IV. RESOURCE REQUIREMENTS

V. SCHEDULE

VI. RISK MANAGEMENT

VII. DELIVERY PLAN

VIII. SECURITY ASPECTS

IX. ABBREVIATIONS AND DEFINITIONS

X. REFERENCES

**I. OVERVIEW**

**1.1 About Us**

Tech World is a company specializing in developing software and web application. We focus on creating the products with high quality in design, functionality, and performance. Our customers range from small to medium enterprises, development partners.

​

|  |  |
| --- | --- |
| **Company Name:** | SE Team |
| **Team Name:** | SE Team |
| **Business:** | - Develop software - Develop and design web application |
| **Customer:** | Ms. N.T.T.LOAN |
| **Office:** | International University, Quarter 6, Linh Trung Ward, Thu Duc District, HCM City |
| **Email:** | shiuxing123@gmail.com |
| **Phone:** | (+083) 7300 909 (Mr Hong) |

**1.2. Product’s Information**

Student Registration System is a vital part of any university’s running because students are what keep a university alive. A poor system can mean fewer students being admitted into a university because of mistakes or an overly slow response time.

Student Registration System is the software which is helpful for students as well as the department. In this project, we are facilitated by the system for registration students, add subjects and fee structure. Our Student Registration System helps deal with the various activities related to the students

Aside from the costs for hardware resources and domain server used to develop and run the system, which our team decides to take care of by ourselves, we intend to distribute the final product absolute free for everyone. Any donations from the community are strongly appreciated but are not required. “We believe in what people make possible. Our mission is to empower every person and every organization on the planet to achieve more” is our slogan.  
Our early approximation:

|  |  |
| --- | --- |
| **Time Delivery** | The First Day of the Construction: 26th September 2019. The Last Day of the Construction: 28th December 2019. Duration: approximately 3 months. Demonstration Date: 4nd January 2019 |
| **Budget** | To implement this project, our team is going to need an amount of budget. This amount is separated into three kinds: hardware budget, software budget and human budget. - Hardware budget: $3325 - Software budget: $440 - Human budget: $2000 - Additional budget: $1000 - Total cost estimated: $5765  The detail cost will be present in section Resource Requirement |

|  |
| --- |
| **II. GOALS AND SCOPES** |

**Priority: 1 – Highest 5 – Lowest**

**2.1. Project Goals**

|  |  |  |
| --- | --- | --- |
| **PROJECT GOALS** | **PRIORITY** | **DESCRIPTION** |
| **FUNCTIONAL GOALS** | | |
| **1. Requirement Specification** | 1 | All documents should be clear, easy to understand and collected from all stakeholder |
| **2. UI Design** | 1 | The User Interface of final system should be simple, well-organized and convenience. |
| **3. Implementation** | 2 | All functions must be clear, maintainable and flexible. |
| **4. Testing** | 3 | Testing phase should be included testing plan and testing the whole system with test cases, make sure paying attention into the result of process. |
| **STRATEGIC GOALS** | | |
| **1. Customer services** | 1 | System shall be able to achieve and maintain outstanding services for users |
| **2. Popularity** | 1 | System shall attract many users and make a well- known status in the Internet community |
| **BUSINESS GOALS** | | |
| **1. Released time** | 2 | The final product shall be delivered within 3 months from first day of the development process. |
| **2. Cost** | 2 | The final system (Online Retail Application) shall be delivered to users with the appropriate price. Additionally, we also charge for advertisements from other organizations and use that profit to cover the cost of running the system. |
| **3. Market growth** | 3 | The final product shall produce significant growth in the market. |
| **TECHNOLOGICAL GOALS** | | |
| **1. Basic functions** | 1 | System shall deliver standard functions for Online Retail Application. |
| **2. Future enhancement** | 2 | System shall leave rooms for future improvement (can implement new functions) |
| **QUALITY GOALS** | | |
| **1. Functions execution** | 1 | System shall be able to execute all the implemented functions smoothly, free of errors. |
| **2. Response time** | 2 | System shall be able to respond to user within an acceptable amount of time. |
| **3. System interface** | 1 | System shall display its components in an organized and well-formatted way; clearly and easy to use. |
| **ORGANIZATIONAL GOALS** | | |
| **1.Competence development** | 2 | System shall be designed and implemented so that it is able to compete with the top product in the market. |
| **2. Maintenance** | 3 | System shall be developed such that it is easy to be maintained. |
| **3. New technologies** | 4 | System shall be able to adapt to new technologies. |
| **4. Modern tools exploitation** | 4 | Up-to-date tools and technologies shall be used to develop the whole system. |
| **CONSTRAINTS** | | |
| **1. Environment constraints** | 2 | System shall be able to access by multi-platform devices. |
| **2. Application standards** | 1 | System shall satisfy most common standard rules of a web application. |
| **3. National and cultural standards** | 3 | System shall not be illegal |

**2.2. Project Scope**Customer classifications:  
1. Unprivileged user: Student  
2. Privileged users: Admin

**2.2.1. Included**Due to time and human constrains, our final project can only provide some standard functions that is typical for the Student Registration System:

- Register function (Register account).  
 - Authentication function (Login/Logout function).

- Allow student to enroll for course.

- Allow student to view enrolled course

- Allow student to delete course from list.

- Allow student to view their profile

- Allow student to edit their profile

- Allow admin/student to edit their password  
 - Allow admin to add/edit course.

- Allow admin to view student profile.

- Allow admin to add/edit student profile.

- Allow admin to view enroll history.  
  
**2.2.2. Excluded**There are some functions our group decided to exclude from the project but may intrigue customers will be left for future improvements, including:  
 - Allow student to make payment.

- Allow student to contact admins.  
 - Allow student to receive their password when they forget it.

- Allow student to search for specific course

- Allow student to create their account that does not depend on the acceptation of admins

**III. ORGANIZATION**

|  |
| --- |
|  |

**3.1. Organizational Boundaries and Interfaces**

The project is built on a web-based platform and can be accessed by every device that has internet connection. And since this is only a project developed for the university’s course, the stakeholders are the members that work on this project.

All the resources used in this project belong to the team member. The final product will be received by the internet users. This project is free to use by anyone that has interest in it. Nevertheless, any commercial intention on this product is not allowed.

List of suppliers for the project:

|  |  |
| --- | --- |
| **Company** | **Product** |
| Microsoft Corporation | Microsoft Office 365 |
| Xampp | Database and web server |
| JetBrains | Pycharm |

**3.2. Project Organization**

Each member of the group has their own role in this project, nonetheless, we share our work to one another. Some possess skills that’s required for specific tasks, which will improve the product quality.

List of the team member:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Role** | **Contact** | **Email** |
| Trịnh Minh Đăng | Tester, Business analyst |  |  |
| Nguyễn Đức Phi Hồng | Designer, Developer | +0837300909 | [shiuxing123@gmail.com](mailto:shiuxing123@gmail.com) |
| Võ Huy Thành | Scrum Master |  |  |

**IV. RESOURCE REQUIREMENTS**

|  |
| --- |
|  |

**4.1. Hardware Resource**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Purpose** | **Quantity** | **Price** | **Total Estimation** |
| Asus A510 | Designing/Testing | 1 | $725 | $725 |
| Asus GL552 | Designing/Testing | 1 | $900 | $900 |
| Acer Aspire E5 | Designing/Testing | 1 | $600 | $600 |
| Asus FX504 | Designing/Testing | 1 | $1100 | $1100 |
| **Total estimation of Hardware resource:** | | | | **$3325** |

**4.2. Software Resource**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Purpose** | **Quantity** | **Price** | **Total Estimation** |
| Microsoft Office 365 | Documenting | 4 | $110 | $440 |
| Umlet | Diagram/Designing | 1 | Free | $0 |
| Draw.io | Diagram | 1 | Free | $0 |
| Xampp | Designing | 1 | Free | $0 |
| Pychram | Designing |  | Free | $0 |
| **Total estimation for Software resource:** | | | | **$440** |

**4.3. Human Resource**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Skillset** | **Time** | **Salary per month** |
| Trịnh Minh Đăng | Tester, Business analyst | Full-time | $500 |
| Nguyễn Đức Phi Hồng | Designer, Developer | Full-time | $1000 |
| Võ Huy Thành | Scrum Master | Full-time | $500 |
| **Total estimation for Human resources:** | | | **$2000** |

**V. SCHEDULE**

|  |
| --- |
|  |

**5.1. Work Breakdown Structure**

**5.2. Schedule and Milestones**

Here is the product backlog that our team comes up with after applying Scrum Framework to the project development process:

**Priority: 1- Highest**

**5- Lowest**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Sprint** | **Date** | **Backlog Items** | **Estimate (week)** | **Priority** |
| 1 | **Sprint 1** Register account | 21/10/2019-27/10/2019 | As an admin, I want to register a student so that I can have all student’s accessibilities. | 2 | 1 |
| 2 | **Sprint 2** Login/Logout | As a student and admin, I want to login/logout my account. | 1 |
| 3 | **Sprint 3**  Add student | 28/10/2019-3/11/2019 | As an admin, I want to add student. | 1 | 4 |
| 4 | **Sprint 4**  Add/edit course | 4/11/2019-10/11/2019 | As an admin, I want to Add/edit courses to course list | 3 | 1 |
| 5 | **Sprint 5** Add course (Student) | 11/11/2019-17/11/2019 | As a student, I want to add courses and edit/update in my registration list. | 2 | 2 |
| 6 | **Sprint 6**  View student | 18/11/2019-24/11/2019 | As a admin, I want to see details of student. | 2 | 2 |
| 7 | **Sprint 7**  View enrolled course | 25/11/2019- 1/12/2019 | As a student, I want to see my enrolled course | 2 | 4 |
| 8 | **Sprint 8**  View enrolled course (Admin) | 2/12/2019- 8/12/2019 | As an admin, I want to see student’s enrolled course. | 2 | 1 |
| 9 | **Sprint 9** Change Password | 9/12/2019- 15/12/2019 | As a student, I want to change my password. | 2 | 2 |

**Schedule and Milestones**

|  |  |
| --- | --- |
| Whole team |  |
| Designer |  |
| Tester |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Sprint 1 + 2  Register account + Login/Logout (21/10/2019-27/10/2019)** | | | | | | | | | | | | | | | | | |
| **Tasks** | | **Mon** | | **Tue** | | | **Wed** | | **Thu** | | | | **Fri** | | **Sat** | **Sun** | |
| Analyze the requirements and define the concept of the functions | |  | |  | | |  | |  | | | |  | |  |  | |
| Design interface for registration page | |  | |  | | |  | |  | | | |  | |  |  | |
| Design interface for Login/ Logout page | |  | |  | | |  | |  | | | |  | |  |  | |
| Design + insert data into database | |  | |  | | |  | |  | | | |  | |  |  | |
| Test functions with database | |  | |  | | |  | |  | | | |  | |  |  | |
|  | | | | |  | | | | |  | | |  | | |  | |  |  |  |
| **Sprint 3 Add menu (28/10/2019-3/11/2019)** | | | | | | | | | | | | | | | | | |
| **Tasks** | | | **Mon** | | **Tue** | | | **Wed** | | **Thu** | | **Fri** | | | **Sat** | **Sun** | |
| Analyze the requirements and define the concept of the functions | | |  | |  | | |  | |  | |  | | |  |  | |
| Design interface for Add course | | |  | |  | | |  | |  | |  | | |  |  | |
| Design + insert data into database | | |  | |  | | |  | |  | |  | | |  |  | |
| Test functions with database | | |  | |  | | |  | |  | |  | | |  |  | |
|  | | |  | |  | | |  | |  | |  | | |  |  | |
| **Sprint 4**  **Add/edit course (4/11/2019-10/11/2019)** | | | | | | | | | | | | | | | | | |
| **Tasks** | | | **Mon** | | **Tue** | | | **Wed** | | **Thu** | | **Fri** | | | **Sat** | **Sun** | |
| Analyze the requirements and define the concept of the functions | | |  | |  | | |  | |  | |  | | |  |  | |
| Design interface for Course | | |  | |  | | |  | |  | |  | | |  |  | |
| Design + insert data into database | | |  | |  | | |  | |  | |  | | |  |  | |
| Test functions with database | | |  | |  | | |  | |  | |  | | |  |  | |
|  | | |  | |  | | |  | |  | |  | | |  |  | |
|  | | |  | |  | | |  | |  | |  | | |  |  | |
| **Sprint 5  Update Course (Student) (11/11/2019-17/11/2019)** | | | | | | | | | | | | | | | | | |
| **Tasks** | | | **Mon** | | **Tue** | | | **Wed** | | **Thu** | | **Fri** | | | **Sat** | **Sun** | |
| Analyze the requirements and define the concept of the functions | | |  | |  | | |  | |  | |  | | |  |  | |
| Design interface for enrolled course (Student) | | |  | |  | | |  | |  | |  | | |  |  | |
| Design + insert data into database | | |  | |  | | |  | |  | |  | | |  |  | |
| Test functions with database | | |  | |  | | |  | |  | |  | | |  |  | |
|  | | |  | |  | | |  | |  | |  | | |  |  | |
| **Sprint 6 View Student (18/11/2019-24/11/2019)** | | | | | | | | | | | | | | | | | |
| **Tasks** | | | **Mon** | | **Tue** | | | **Wed** | | **Thu** | | **Fri** | | | **Sat** | **Sun** | |
| Analyze the requirements and define the concept of the functions | | |  | |  | | |  | |  | |  | | |  |  | |
| Design interface for view student page | | |  | |  | | |  | |  | |  | | |  |  | |
| Design + insert data into database | | |  | |  | | |  | |  | |  | | |  |  | |
| Test functions with database | | |  | |  | | |  | |  | |  | | |  |  | |
|  | | |  | |  | | |  | |  | |  | | |  |  | |
| **Sprint 7 View enrolled course (25/11/2019- 1/12/2019)** | | | | | | | | | | | | | | | | | |
| **Tasks** | | | **Mon** | | **Tue** | | | **Wed** | | **Thu** | | **Fri** | | | **Sat** | **Sun** | |
| Analyze the requirements and define the concept of the functions | | |  | |  | | |  | |  | |  | | |  |  | |
| Design interface for enrolled course (admin) | | |  | |  | | |  | |  | |  | | |  |  | |
| Design + insert data into database | | |  | |  | | |  | |  | |  | | |  |  | |
| Test functions with database | | |  | |  | | |  | |  | |  | | |  |  | |
|  | | |  | |  | | |  | |  | |  | | |  |  | |
| **Sprint 8 View enrolled course (Admin) (2/12/2019- 8/12/2019)** | | | | | | | | | | | | | | | | | |
| **Tasks** | | | **Mon** | | **Tue** | | | **Wed** | | **Thu** | | **Fri** | | | **Sat** | **Sun** | |
| Analyze the requirements and define the concept of the functions | | |  | |  | | |  | |  | |  | | |  |  | |
| Design interface for view enrolled course | | |  | |  | | |  | |  | |  | | |  |  | |
| Design + insert data into database | | |  | |  | | |  | |  | |  | | |  |  | |
| Test functions with database | | |  | |  | | |  | |  | |  | | |  |  | |
|  | | |  | |  | | |  | |  | |  | | |  |  | |
| **Sprint 9 Change Password (3/12/2019 - 9/12/2019)** | | | | | | | | | | | | | | | | | |
| **Tasks** | | | **Mon** | | **Tue** | | | **Wed** | | **Thu** | | **Fri** | | | **Sat** | **Sun** | |
| Analyze the requirements and define the concept of the functions | | |  | |  | | |  | |  | |  | | |  |  | |
| Design interface for Change password | | |  | |  | | |  | |  | |  | | |  |  | |
| Design + insert data into database | | |  | |  | | |  | |  | |  | | |  |  | |
| Test functions with database | | |  | |  | | |  | |  | |  | | |  |  | |

**VI. RISK MANAGEMENT**

|  |
| --- |
|  |

When an entity makes an investment decision, it exposes itself to several financial risks. Therefore, in order to minimize and control the exposure of investment to such risks, we listed a list of risks that our project would likely to come across as well as some strategy to solve those problems.

|  |  |  |  |
| --- | --- | --- | --- |
| **PROJECT RISKS** | | | |
| **Risk** | **Probability** | **Description** | **Strategy** |
| **Stakeholders fail to support project** | Low | When stakeholders have a negative attitude towards the project and wish to see it fail. | Perform weekly meeting with customer to have their ideas about the project and keep track of their demands. |
| **Resource shortfalls** | Moderate | Inability to secure sufficient resources for the project. | Have a specific project’s resource checklist and plan some backup plan in case of insufficient resource |
| **Training is inadequate** | Moderate | Training is often a poor substitute for professional experience. Projects shouldn't assume that resources will be fully productive in a new skill. | Have teammate prepare for necessary skills as well as techniques and tools before the project start executing to create a good product.  Choosing right member with suitable skills for the project. |
| **Low team motivation** | High | Team members lack motivation | Have daily meeting with team member to discuss about the difficulty and trouble that they are having and find a good way to solve each problem. |
| **COMMERCIAL RISKS** | | | |
| **Product doesn't sell** | Moderate | Demand risk for the new product in market. | Have a strategy of advertising the product via media and online ads to get the interest of as many people as possible.  Add some unique features to increase the project competitiveness among the market. |
| **Technology components aren't reliable** | Moderate | Components that fail after a short time. | Make sure that the project is implemented with latest technology |
| **PRODUCT RISKS** | | | |
| **User interface is low quality** | High | The user interface is buggy, slow or difficult to use. | Have some surveys about customers’ opinions of the interface and summarize those ideas to create an optimal interface. |
| **Project management tool problems & issues** | Moderate | Technical problems with the project management tools themselves. | The project manager must know and get use to using a suitable tool to develop the project. |
| **Project team misunderstand requirements** | High | When requirements are misinterpreted by the project team a gap develops between expectations, requirements and work packages. | Revise the requirement analysis phase carefully before adjusting the role of each team members. |

**VII. DELIVERY PLAN**

|  |
| --- |
|  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Sprint 1 & 2** | **Main Tanks** | **Day 1** | **Day 2** | **Day 3** | **Day 4** | **Day 5** | **Day 6** | **Day 7** | **Estimate  (Total hours)** |  |  |  |  |  |  |
|  | **Register account + Login/Logout (21/10/2019-27/10/2019)** | **User Story:** As an admin, I want to register a student so that I can have all student’s accessibilities. | | | | | | | | |  |  |  |  |  |  |
|  | **User Story**: As a student and admin, I want to login/logout my account. | | | | | | | | |  |  |  |  |  |  |
|  | Design UI for Register page | 4 |  |  |  |  |  | 2 | 27 |  |  |  |  |  |  |
|  | Design UI for Login / Logout page | 4 |  |  |  |  |  | 2 |  |  |  |  |  |  |
|  | Design use cases and class diagrams |  |  | 4 |  |  |  | 4 |  |  |  |  |  |  |
|  | Create database to manage data of users |  |  |  |  |  |  | 3 |  |  |  |  |  |  |
|  | Testing Register page |  |  |  |  |  | 2 |  |  |  |  |  |  |  |
|  | Testing Login /Logout page |  |  |  |  |  | 2 |  |  |  |  |  |  |  |
|  | **Sprint 3** | **Main Tanks** | **Day 1** | **Day 2** | **Day 3** | **Day 4** | **Day 5** | **Day 6** | **Day 7** | **Estimate  (Total hours)** |  |  |  |  |  |  |
|  | **Add menu (28/10/2019-3/11/2019)** | **User Story:** As an admin, I want to add student. | | | | | | | | |  |  |  |  |  |  |
|  | Design UI for add page |  |  |  |  |  | 5 |  | 18 |  |  |  |  |  | 23 |
|  | Test functions with database |  |  |  |  |  |  | 3 |  |  |  |  |  |  |
|  | Design use cases and class diagrams |  |  | 2 |  |  | 4 |  |  |  |  |  |  |  |
|  | Testing page |  |  | 2 |  |  |  | 2 |  |  |  |  |  |  |
|  | **Sprint 4** | **Main Tanks** | **Day 1** | **Day 2** | **Day 3** | **Day 4** | **Day 5** | **Day 6** | **Day 7** | **Estimate  (Total hours)** |  |  |  |  |  |  |
|  | **Add/edit course (4/11/2019-10/11/2019)** | **User Story**: As an admin, I want to Add/edit courses to course list | | | | | | | | |  |  |  |  |  |  |
|  | Design UI for course |  |  | 2 |  |  | 2 | 5 | 30 |  |  |  |  |  |  |
|  | Test functions with database |  |  | 2 |  |  |  | 4 |  |  |  |  |  |  |
|  | Design use cases and class diagrams |  |  | 3 |  |  | 3 | 4 |  |  |  |  |  |  |
|  | Create database of Course | 1 |  |  | 1 |  |  | 3 |  |  |  |  |  |  |
|  | **Sprint 5** | **Main Tanks** | **Day 1** | **Day 2** | **Day 3** | **Day 4** | **Day 5** | **Day 6** | **Day 7** | **Estimate  (Total hours)** |  |  |  |  |  |  |
|  | **Update Course (Student) (11/11/2019-17/11/2019)** | **User Story**: As a student, I want to add courses and edit/update in my registration list. | | | | | | | | |  |  |  |  |  |  |
|  | Design UI for enrolled course |  |  | 2 |  |  | 2 | 4 | 26 |  |  |  |  |  |  |
|  | Test functions with database |  |  | 2 |  |  |  | 3 |  |  | 26 |  |  |  |
|  | Design use cases and class diagrams |  |  |  |  |  |  | 3 |  |  |  |  |  |  |
|  | Create database of Shopping cart |  |  | 3 |  |  | 3 | 4 |  |  |  |  |  |  |
|  | **Sprint 6** | **Main Tanks** | **Day 1** | **Day 2** | **Day 3** | **Day 4** | **Day 5** | **Day 6** | **Day 7** | **Estimate  (Total hours)** |  |  |  |  |  |  |
|  | **View Student (18/11/2019-24/11/2019)** | **User Story**: As a admin, I want to see details of student.. | | | | | | | | |  |  |  |  |  |  |
|  | Design UI for paying page |  |  | 3 |  |  |  | 4 | 24 |  |  |  |  |  |  |
|  | Create database of paying |  |  |  |  |  |  | 2 |  |  |  |  |  |  |
|  | Test functions with database |  |  | 2 |  |  |  | 2 |  |  |  |  |  |  |
|  | Test Order with system |  |  | 1 |  |  | 2 |  |  |  |  |  |  |  |
|  | Design use cases and class diagrams |  |  | 3 |  |  | 2 | 3 |  |  |  |  |  |  |
|  | **Sprint 7** | **Main Tanks** | **Day 1** | **Day 2** | **Day 3** | **Day 4** | **Day 5** | **Day 6** | **Day 7** | **Estimate  (Total hours)** |  |  |  |  |  |  |
|  | **View enrolled course (25/11/2019- 1/12/2019)** | **User Story**: As a student, I want to see my enrolled course. | | | | | | | | |  |  |  |  |  |  |
|  | Design UI for view enrolled course |  |  |  |  |  |  | 4 | 13 |  |  |  |  |  |  |
|  | Create database contact Admin |  |  | 2 |  |  |  |  |  |  |  |  |  |  |
|  | Test functions with database |  |  |  |  | 2 |  |  |  |  |  |  |  |  |
|  | Design use cases and class diagrams |  |  | 2 |  |  |  | 3 |  |  |  |  |  |  |
|  | **Sprint 8** | **Main Tanks** | **Day 1** | **Day 2** | **Day 3** | **Day 4** | **Day 5** | **Day 6** | **Day 7** | **Estimate  (Total hours)** |  |  |  |  |  |  |
|  | **View enrolled course (Admin) (2/12/2019- 8/12/2019)** | **User Story**: As an admin, I want to see students enrolled course. | | | | | | | | |  |  |  |  |  |  |
|  | Design UI for view enrolled course for admin |  |  | 2 |  |  |  | 4 | 18 |  |  |  |  |  |  |
|  | Test functions with database |  |  | 2 |  |  |  | 3 |  |  |  |  |  |  |
|  | Test product of database |  |  |  |  | 1 | 1 | 1 |  |  |  |  |  |  |
|  | Design use cases and class diagrams |  |  |  |  |  |  | 4 |  |  |  |  |  |  |
|  | **Sprint 9** | **Main Tanks** | **Day 1** | **Day 2** | **Day 3** | **Day 4** | **Day 5** | **Day 6** | **Day 7** | **Estimate  (Total hours)** |  |  |  |  |  |  |
|  | **Change Password (3/12/2019 - 9/12/2019)** | **User Story:** As a student, I want to change my password. | | | | | | | | |  |  |  |  |  |  |
|  | Design UI for change password |  |  |  |  |  | 1 | 3 | 15 |  |  |  |  |  |  |
|  | Test functions with database | 1 |  |  |  |  |  |  |  |  |  |  |  |  |
|  | Test product of database |  |  | 1 |  |  |  |  |  |  |  |  |  |  |
|  | Design use cases and class diagrams |  |  | 4 |  |  | 2 | 3 |  |  |  |  |  |  |

**VIII. SECURITY ASPECT (OPTIONAL)**

|  |
| --- |
|  |

Data security refers to protective digital privacy measures that are applied to prevent unauthorized access to computers, databases and websites. Data security also protects data from corruption. Data security is an essential aspect of IT for organizations of every size and type.

Here are some vulnerabilities that are likely to happen to a web-based project system:

|  |  |  |
| --- | --- | --- |
| **Vulnerability** | **Description** | **Solution** |
| **Memory safety** | Memory safety is the state of being protected from various software bugs and security vulnerabilities when dealing with memory access, such as buffer overflows and dangling pointer. | Uses static program analysis and automated theorem proving to ensure that the program is free of memory errors |
| **Malicious software** | An internet user can be tricked or forced into downloading software onto a computer that is of malicious intent. Such software comes in many forms, such as viruses, Trojan horses, spyware, and worms. | Add static analysis (also called “white-box” testing) to your software development lifecycle to review your code for the presence of malicious code  Use some trusted security application to detect malicious code. |
| **Application vulnerabilities** | Applications used to access Internet resources may contain security vulnerabilities such as memory safety bugs or flawed authentication checks. | Use runtime application self-protection (RASP) technologies deployed within or alongside the application runtime environment that instruments an application and enables detection and prevention of attacks |
| **Phishing** | Phishing is an attack which targets online users for extraction of their sensitive information such as username, password and credit card information | Transaction verification and signing through using mobile phone as a second channel for verification and authorization of banking transactions. |

|  |
| --- |
| **IX. ABBREVIATIONS AND DEFINITIONS** |

|  |  |
| --- | --- |
| **WORD** | **MEANING** |
| **UI Design** | User Interface Design |
| **MySQL** | A freely available open source Relational Database  Management System (RDBMS) that uses Structured Query Language (SQL) |
| **Hardware** | Physical component of a computer system |
| **Software** | The programs and other operating information used by a computer. |
| **Use case** | A use case is a methodology used in system analysis to identify, clarify and organize system requirement |
| **Product backlogs** | A list of all things that needs to be done within the project |
| **Agile** | A group of software development methods |
| **Scrum** | A lightweight process framework for agile development |
| **Iteration** | The process of repeating computing process |
| **HTML** | Hyper Text Markup Language - the standard markup language for creating Web pages. |
| **CSS** | The language for describing the presentation of Web pages, including colors, layout, and fonts |
| **Bug** | An error, flaw, failure or fault in a computer program or system that causes it to produce an incorrect or unexpected result |

**X. REFERENCES**

|  |
| --- |
|  |

1. <https://continuingprofessionaldevelopment.org/risk-management-steps-in-risk-management-process/>

2. <https://searchcompliance.techtarget.com/definition/risk-management>

3. <https://www.peterjoubert.com/resource-requirements/>

4. <https://www.projectinsight.net/project-management-basics/project-management-schedule>

5. <https://www.liquidplanner.com/blog/back-school-special-project-management-tips-successful-project-delivery/>

6. <https://www.wbdg.org/project-management>

7. <https://www.commonplaces.com/blog/6-common-website-security-vulnerabilities/>

8. <https://www.creativebloq.com/web-design/website-security-tips-protect-your-site-7122853>