**B. Report No. 2 Software Project Management Plan**

**1.**  **Problem Definition**

**1.1**   **Name of this Capstone Project**

▪   Official name: Group Travel Assistant

▪   Vietnamese name: Ứng dụng hỗ trợ du lịch tự túc theo nhóm.

▪   Abbreviation: EERS

**1.2**    **Problem Abstract**

Below are the problems encountered in this project:

* + **New framework:** The team needs time to learn about a new framework like spring boot.
  + **Algorithm:** We need to research how to implement a recommendation engine.
  + **Business:** We need to console and take surveys multiple times to understand the business of traveling.

**1.3**   **Project Overview**

**1.3.1**           **Current Situation**

Below are the problems encountered in this project:

* **Limit time and human resource**: Team has only 4 members and time for all projects is about 13 weeks for writing a document, implementing the products, and testing.
* **Schedule of team members**: team members can have a conflict in meeting schedules because of sickness, or class and work schedule, etc.
* **Framework study**: team members have a problem when applying the framework into a project. The team needs an amount of time to get familiar with new techniques.
* **New techniques, algorithms:** Some team members are new to the techniques, algorithms used in the project. The team needs an amount of time to get familiar with those techniques.
* **Lack of UI (user interface), UX (user experience) design skill**: Our team members all study IS major, and no one has studied UI, UX design. Therefore, some UI may be misunderstood or hard to use with normal users.

**1.3.2**           **The Proposed System**

**1.3.2.1 Mobile application**

* Support users to login/register.
* Support users to create travel group/ travel plan
* Support user to enter the group with an invitation
* Easy for non-technical users to use.
* Support expense/budget tracking and calculate money return after trip

**1.3.1.2 Web application**

Web management is designed for administrators to:

* Create/modify a place of interest.

**1.3.3**           **Boundaries of the System**

The system can:

* Create a travel group.
* Send an invitation to join the group.
* Review a place of interest and trip
* Create/end a trip.
* Track budget/expense on a trip.
* Suggest a place of interest that is based on the user’s hobbies and history.
* Support in-group chat.
* Feedback place on interest and trip

The system cannot:

* Transportation booking.
* Booking a hotel/motel.
* Money return with online payment

**1.3.4**           **Future Plans**

* Search flights, hotels, trains, …
* Scan receipt to expense tracking

**1.3.5**           **Development environment**

**1.3.5.1**        **Hardware requirements**

|  |  |  |
| --- | --- | --- |
| **Server** | **Minimum Requirements** | **Recommended** |
| **Internet Connection** | Cable (10 Mbps) | Cable (50 Mbps or more) |
| **Operating System** | 7,10, Window Server 2008 | 10, Window Server 2008 |
| **Computer Processor** | Intel® Xeon ® 1.4GHz | Intel® Xeon ® Quad-Core (12M Cache, 2.50 GHz) |
| **Computer Memory** | 6GB RAM | 32GB or more |
| **Storage space** | 8GB | 16GB or more |

**For PC**

|  |  |  |
| --- | --- | --- |
| **PC** | **Minimum Requirements** | **Recommended** |
| **Internet Connection** | **Cable, Wi-Fi (4 Mbps)** | **Cable, Wi-Fi (8 Mbps)** |
| **Operating System** | **None** | **None** |
| **Computer Processor** | **Intel® Core i3 1.4GHz** | **Intel® Core i5 2.50GHz** |
| **Computer Memory** | **4GB RAM** | **6GB RAM or more** |
| **Web Browser** | **None** | **None** |

**1.3.5.2**        **Software requirement**

|  |  |  |
| --- | --- | --- |
| **Name** | **Name/Version** | **Description** |
| **Environment** | JDK 1.8  Android API > 21 | Specification for developing Android and Server  application |
| **Operating System** | Window 10 | Operating system and platform for development |
| **Modeling tool** | LucidChart | Used to design diagram |
| **IDE** | Netbean IDE 11.3, Android Studio,IntelliJ | Programming tools |
| **DBMS** | My SQL 8.0 | Used to create & manage the database for system |
| **Web Server** | Apache Tomcat 8 | Deployment environment |
| **Web Browser** | Chrome 69 or above. | Testing browser |

Table 3 - Software Requirement

**2.**  **Project Organization**

**2.1 Software Process Model**

**What is agile?**

Simply stated, Agile is a flexible software development model, based on iterative and incremental methods. It will engage customers in the development process of the software, people try to make the product as fast as possible. Then give the customer a try and feedback, the development team will continue to develop the next stage. Depending on the project, the release time of the product is long or short (maybe 2 weeks, maybe 1 month ...)

**What is SCRUM?**

Scrum is a software development process based on the agile method so that it follows the rules of agile.

**Why is SCRUM?**

Product status can be a factor for us to decide whether to use Scrum. When the product is in the early stages of development, there are many changes, but not too urgent, there are many risks involved in people and technology, Scrum is very appropriate.

**2.2 Project management**

This project is developed under the Scrum model. My team applies the Scrum model to suit the current situation in our team. We choose this model due to the following reasons:

·     Enhanced collaboration and communication: This project is new for our team. Therefore, we have to discuss everyday – Scrum model suggests that project progress via a series of sprints. At each sprint, we need to prepare a planning meeting where our team has many items we can commit to, and then create a sprint backlog (a list of tasks to do during the sprint). During the daily meeting, team members share what they worked or what they learned on the prior day.

·     High performance: Sprints are very short so results are delivered and are ready for testing with 1 -2 week(s). Therefore, our team needs to focus on the sprint backlog.

·     The project implements new ideas, so maybe the product owner changes requirements and scope. The team will adapt to change better.

·     In this project, there are many new technologies that need to be learned. With the Scrum model, our team can learn and develop in parallel to meet deadlines.

*Figure 2 Scrum Model*

For more information:

·    <http://scrummethodology.com/>

**2.3 Roles and responsibilities**

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Role in Scrum** | **Name** | **Responsibilities** |
| 1 | **Product owner** | Nguyễn Phan Quang Nhật | Specify scope and user requirements.  Supervise the development progress.  Provide professional techniques and business analysis support.  Support knowledge about algorithms. |
| 3 | **Scrum team member** | Nguyễn Tiến Thiên  Đỗ Trần Nhật Duy  Lê Hoài Ân  Nguyễn Văn Dương | Collecting data  Preprocessing data  Transforming the data to features  Evaluating with other models  Designing database  Clarifying requirements  Prepare documents  GUI design  Coding  Testing |

Table 4 - Roles and responsibilities

**2.3**   **Tools and Techniques**

|  |  |
| --- | --- |
| Tool/Technique | Name |
| Front-end | HTML, CSS, JavaScript, Java Android |
| Back-end | Spring Boot Framework |
| IDE | NetBeans 11.3, IntelliJ, Android Studio |
| DBMS | MySQL |
| Source Control | GitLab |
| Modeling tool | StarUML, Draw.io |

Table 5 - Tools and techniques

**3.**  **Project Management Plan**

**3.1.** **Product Backlog**

Product Backlog could be found

**3.2.** **Sprint Backlog**

Sprint Backlog can be found

**3.3.** **Deliverables**

|  |  |  |
| --- | --- | --- |
| **No** | **Deliverable** | **Note** |
| **1** | **Introduction, Mock UI** | **Sprint 1** |
| **…** |  |  |
|  |  |  |

**Table 6 - Deliverables**

**3.4.** **All Meeting Minutes**

All sprint meeting minutes could be found

**4.**  **Coding Convention**

**Naming Convention:**

●   Variable names should be short yet meaningful.

●  The choice of a variable name should be stick with the intent of its use.

●   Methods should be verbs, the first letter lowercase then the first letter of each internal word capitalized.

● A class should be nouns, the first letter lowercase then the first letter of each internal word capitalized.

**General Convention:**

●  One declaration per line is recommended to encourage commenting.

●  Variables and functions can’t be declared on the same line.

● Always put spaces at the beginning and the end of an operator ( = + - \* / ).

**Declarations Convention:**

●  One declaration per line is recommended to encourage commenting.

Reference:

<http://www.oracle.com/technetwork/java/codeconvtoc-136057.html>