**Report 2 – Software Project Management Plan**

1. **Problem Definition**
   1. **Name of this Capstone Project**

* **Official name:** Just Walk Out Library
* **Vietnamese name:** Thư viện mượn sách tự động
* **Abbreviation:** JWL
  1. **Problem Abstract**

To help librarians reduce work overload and give borrowers a much more comfortable experience in acquiring books, we provide the JWL system. Our system will automate the book-borrowing process with the help of RFID, NFC, QR Code, and iBeacon. Those great technologies may come with some of the difficulties:

* They require knowledge about hardware, which our team is lacking of. Thus, it takes more time for us to do research.
* Not every borrower brings smart phone to the library. Thus, JWL needs to cooperate with the library’s traditional way in order to bring the most comfortable service to its user.
  1. **Project Overview**
     1. **Current Situation**

Below are the problems encountered in this project:

* + **New technologies:** as stated above, our team have no experience in hardware. Therefore, we need time to do research about each technology, which devices to choose, and how to use them.
  + **Traditional habit:** Borrowers have been using traditional way to obtain books for a long time, so libraries should provide proper introduction and assistance for their users to get use to JWL.
  + **Devices problem:** RFID/NFC devices may be crashed or damaged after long time using, so backup solution should be prepared and periodically maintenance should be carried out.
    1. **The Proposed System**

To help library’s borrower ‘just walk out library’, JWL will be launched on two platforms:

* + Mobile Application:
    - For borrower: to borrow book automatically when he/she steps out of the library.
    - For emulator: to read NFC/QR Code from the borrower’s mobile application.
  + Web Application:
    - For Admin: to manage user accounts.
    - For librarian: to manage books and borrowers.
      1. **Mobile Application**

For borrower:

* Make NFC connection.
* Show QR Code.
* Manage user’s information.
* Search books.
* Add books to wish list.
* Track list of borrowed books.
* Notify user about:
  + Borrowed books.
  + Book return date.
  + Available books in user’s wish list.

For emulator:

* Read NFC/QR Code from the borrower’s mobile application.
* Send the read data above to serve for validation.
  + - 1. **Web Application**

For admins:

* + - Manage accounts (CRUD).
    - Deactivate/activate accounts.

For librarians:

* + - Manage borrowers (CRUD).
    - Deactivate/activate borrowers.
    - Manage books (CRUD).
    1. **Boundaries of the System**
* The system is mostly built based on real processes of library management. Our main target is to improve the current process and make it more convenient and efficient:
  + Allow borrower to check in the library by scanning their phone to the RFID reader of the library.
  + Allow borrower to check out with their borrowed books by just walking out of the library.
  + Allow borrower to search for wanted book by title, author, category.
  + Allow borrower to add a book to a wish list.
  + Allow librarian to manage borrowers and their borrowed book list.
  + Allow librarian to manage books.
  + Allow admin to manage all accounts.
* Any libraries which deployed this system must set up their devices as required below:
  + One RFID tag on each book.
  + RFID reader, QR Code reader, and iBeacon at the entrance of the reading room.
  + Computer system with internet connection.
* Borrowers that use JWL have to make sure that:
  + They open the application when checking in and out of the library.
  + Their phones have internet and Bluetooth connection open.
* The completed product includes:
  + Android mobile application.
  + Website application.
    1. **Future Plans**

With further research and development, the system can apply the following features:

* Library can place RFID reader and iBeacon at the gate. Borrowers carrying books and mobile device just walk out. It is suitable for any borrowers who lack of knowledge in information technology.
* Library store book’s position and its status. System provide function allows borrows can search books, position and book status before going to library.
  + 1. **Development Environment**
       1. **Hardware requirements**
* **For web application server**

|  |  |  |
| --- | --- | --- |
| Windows | Minimum Requirements | Recommended |
| Internet Connection | Cable, Wi-Fi (4 Mbps) | Cable, Wi-Fi (8 Mbps) |
| Operating System | Window Server 2008 R2 | Window Server 2012 R2 |
| Computer Processor | Intel® Xeon ® 1.4GHz | Intel® Xeon ® Quad Core |
| Computer Memory | 2GB of RAM | 4GB of RAM or more |

Table 2 : Hardware Requirement for Server.

* **For Mobile (As Emulator)**

|  |  |  |
| --- | --- | --- |
| Android | Minimum | Recommended |
| Internet Connection | Wi-Fi or 3G (1 Mbps) | Wi-Fi or 3G (8 Mbps) |
| Operating System | Android 4.4.2 | Android 6.0.0 |
| Mobile Processor | Cortex-A7 Dual-Core 1.3GHz | Cortex-A7 Dual-Core 1.3GHz |
| Mobile Memory | 1GB of RAM | 2GB of RAM or more |
| Mobile Feature | NFC, Camera supporting, Google play service version more than 7.0 | NFC, Camera supporting, Google play service version more than 7.0 |

*Table 3 : Hardware Requirement for Mobile as Emulator.*

* **For Mobile (As User’s device)**

|  |  |  |
| --- | --- | --- |
| Android | Minimum | Recommended |
| Internet Connection | Wi-Fi or 3G (1 Mbps) | Wi-Fi or 3G (8 Mbps) |
| Operating System | Android 4.4.2 | Android 6.0.0 |
| Mobile Processor | Cortex-A7 Dual-Core 1.3GHz | Cortex-A7 Dual-Core 1.3GHz |
| Mobile Memory | 1GB of RAM | 2GB of RAM or more |
| Mobile Feature | - | NFC |

*Table 4: Hardware Requirement for Mobile as User’s device.*

* + - 1. **Software requirements**

|  |  |  |
| --- | --- | --- |
| Software | Name / Version | Description |
| Operating system | Window 7 Professional SP1 | Operating system and platform for development |
| Environment | JDK version 1.8 | Specification for developing web application |
| IDE | IntelliJ IDE 2016.3.2, Android Studio v2.2.3, XCode | Used for implement website and Android Mobile App. |
| Design Model tool | StartUML v2.5.1 | Used for creating modal and diagrams. |
| DBMS | PostgreSQL | Used to create & manage the database for system |
| Document storage | Github | Used for storing document |
| Store and manage source code | Github & SourceTree | Used to store all source code |
| Web browser | Chrome 42 or above | Testing browser |

Table 5: Software requirements

1. **Project organization**
   1. **Software Process Model**

The project is developed using Scrum – Agile Model. We choose the model for the following reasons:

* RFID, NFC, iBeacon, and QR Code are new to our team. We need more time to learn about those technologies and apply them gradually to the system.
* Prototypes are delivered frequently for evaluation.
* Team members can involve more in the development process.

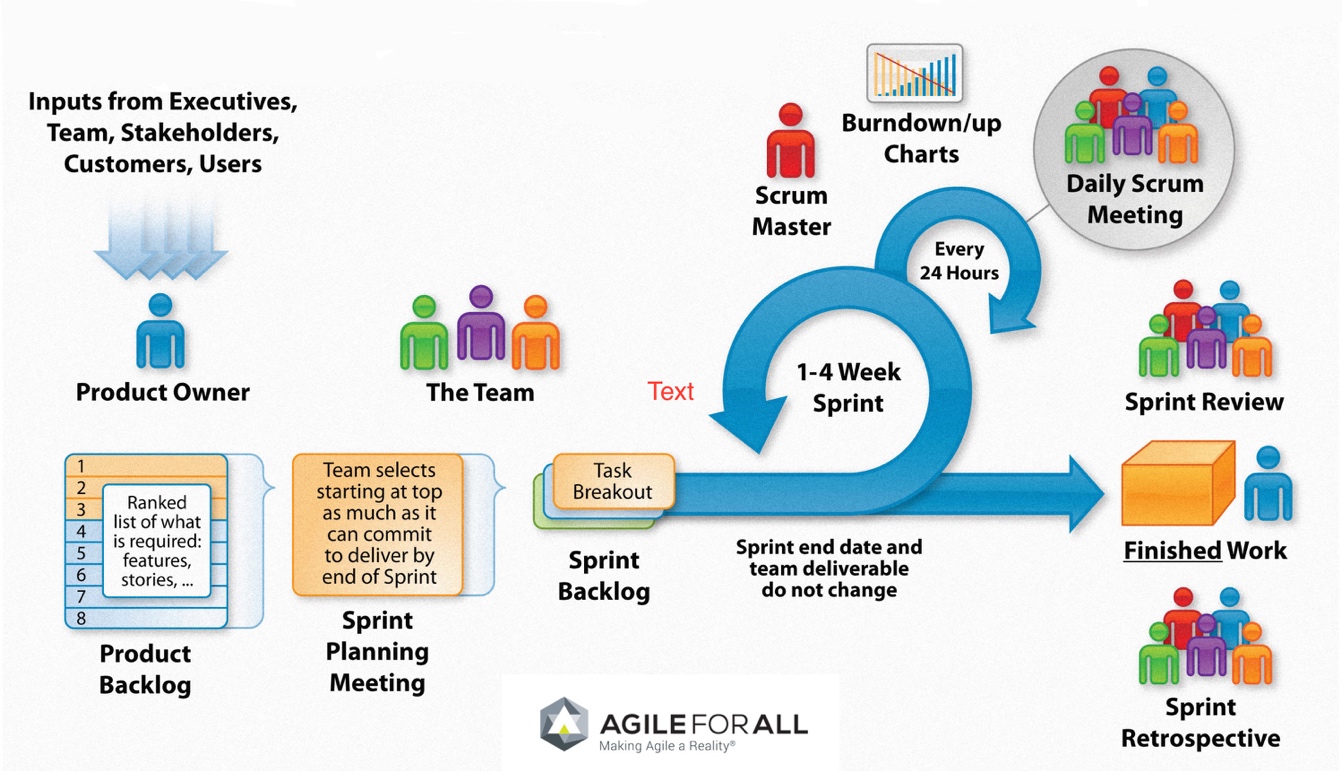


Figure 1 : Scrum model.

*Reference:* [*http://agileforall.com/resources/introduction-to-agile/*](http://agileforall.com/resources/introduction-to-agile/)

* 1. **Roles and responsibilities**

|  |  |  |  |
| --- | --- | --- | --- |
| No | Full name | Role in Group | Responsibilities |
| 1 | Kiều Trọng Khánh | Product owner, Project Manager | * Specify user requirements * Control the development process * Provide technical and business analysis support |
| 2 | Võ Hồng Hà | Scrum leader, B.A, Developer, Tester | * Manage process * Design database * Clarify requirements * Prepare and combine documents * GUI design * Create test plan * Code * Test |
| 3 | Nguyễn Tuấn Anh | Team member,  B.A, Developer,  Tester | * Design database * Clarify requirements * Prepare documents * GUI design * Create test plan * Code * Test |
| 4 | Đặng Nhật Thiên | Team member,  B.A, Developer,  Tester | * Design database * Clarify requirements * Prepare documents * GUI design * Create test plan * Code * Test |

* 1. **Tools and Techniques (WIP)**

|  |  |
| --- | --- |
| Tool | Name / version |
| Web server | Tomcat |
| Development tool | IntelliJ IDEA 2016.3.2, Android Studio, Webstorm |
| DBMS | Postgresql |
| Source control | Github & SourceTree |
| Modeling tool | StarUML v5.0.1 |
| Document tool | Microsoft Word 2010 |

Table 6: Tools List

|  |  |
| --- | --- |
| Technique | Name / version |
| Frontend | HTML5, CSS, ReactJS |
| Backend | Spring Boot |
| Mobile | Android, NFC, RFID, QR Code |

Table 7: Technique List

1. **Project Management Plan**
   1. **Product Backlog**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Story ID** | **Features** | **Task ID** | **Task description** | **Sprint** |
| 1 | Create Product Backlog | 1 | Create Product Backlog |  |
| 2 | Write Introduction document | 2 | Write Introduction document |  |
| 3 | Write Project Management Plan | 3.1 | Problem definition |  |
| 3.2 | Project organization |  |
| 3.3 | Project management plan |  |
| 3.4 | Coding convention |  |
| 3.5 | Review document |  |
| 4 | Study RFID, NFC, iBeacon, QR Code | 4.1 | Research common devices |  |
| 4.2 | Find and buy RIFD reader, RFID tags, and iBeacon |  |
| 4.3 | Generate QR Code on Android Application |  |
| 4.4 | How to emulate NFC Reader on Android Application |  |
| 4.5 | How iBeacon works |  |
| 5 | Build System Structure | 5.1 | Backend Structure |  |
| 5.2 | Web Structure |  |
| 5.3 | Mobile Structure |  |
| 6 | Write Software Requirements | 6.1 | User Requirement Specification |  |
| 6.2 | External Interface Requirement |  |
| 6.3 | Use case diagram |  |
| 6.4 | Software System Attributes |  |
| 7 | Write Software Design Description | 6.1 | Design Overview |  |
| 6.2 | System Architectural Design |  |
| 6.3 | Component Diagram |  |
| 6.4 | Detailed Description of Components |  |
| 6.5 | Sequence Diagram |  |
| 6.6 | User Interface Diagram |  |
| 6.7 | Database Design |  |
| 6.8 | Entity Diagram |  |
| 6.9 | Class Diagram |  |
| 8 | Implementation | 8.1 | User Signs In |  |
| 8.2 | User Signs Up |  |
| 8.3 | User Logs Out |  |
| 8.4 | User\_Searchs For Books |  |
| 8.5 | User Views Borrowed Books |  |
| 8.6 | User Adds Books To Wish List |  |
| 8.7 | User Extends Borrowing Books Deadline |  |
| 8.8 | Staff Searchs Account |  |
| 8.9 | Librarian Adds Borrower |  |
| 8.10 | Librarian Updates Borrower |  |
| 8.11 | Librarian Deactivates Borrower |  |
| 8.12 | Librarian Activates Borrower |  |
| 8.13 | Librarian Adds Book |  |
| 8.14 | Librarian Updates Book |  |
| 8.15 | Librarian Removes Book |  |
| 8.16 | Librarian Confirms Returned Book |  |
| 8.17 | Librarian Searchs Borrower |  |
| 8.18 | Librarian Renews Subscription |  |
| 8.19 | Admin Adds Account |  |
| 8.20 | Admin Updates Account |  |
| 8.21 | Admin Deletes Account |  |
| 8.22 | Admin\_Searchs Account |  |
| 8.23 | Automatic Handler Sends Notification about available book in Borrower’s Wish List |  |
| 8.24 | Automatic Handler Sends Notification about Book Return Date |  |
| 8.25 | Automatic Handler Sends Notification about Subscription End Date |  |
| 8.26 | Automatic Handler Checks In Borrower |  |
| 8.27 | Automatic Handler Checks Out Borrower |  |
| 8.28 | Automatic Handler Borrows Books For Borrower |  |
| 9 | Create Software Test Documentation | 9.1 | Test Plan |  |
| 9.2 | Test Cases |  |
| 9.3 | Check lists |  |
| 10 | Quality Assurance | 10.1 | Quality Assurance for Backend |  |
| 10.2 | Quality Assurance for Web |  |
| 10.3 | Quality Assurance for Mobile |  |
| 11 | Software User’s Manual | 11.1 | Installation Guide |  |
| 11.2 | User’s Guide |  |
| 12 | Mobile prototype | 12 | Mobile prototype |  |
| Table 5: Product backlog | | | | |

* 1. **Sprint Backlog**
  2. **Deliverables**
  3. **All Meeting Minutes**

Table 8: Software Development Life Cycle Detail

1. **Coding Convention**

**Java:** Using to develop website and web service.

**Android:** Using to develop mobile application