

**CAPSTONE PROJECT 1**

**ROOMY**

**Product Backlog Document**

Version 1.0

Date: Sep 2nd, 2017

**MENTOR** **: MSc. Phan Van Son**

**PROJECT TEAM** : **Blue team**

**TEAM’S MEMBER** : Huynh Quoc Nhat

Ho Trung Anh

Le Hoang Quoc

Huynh Vo Nhat Lan

Huynh Thi Thanh Van

**INTERNATIONAL SCHOOL**

**Project Information**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Project acronym** | RMY | | | | |
| **Project Tittle** | Roomy | | | | |
| **Start Date** | Aug 17, 2018 | | **End Date** | | Dec 5, 2018 |
| **Lead Institution** | International School, Duy Tan University | | | | |
| **Project Mentor & contact details** | Msc. Son, Phan Van  Email: [sonpvs@duytan.edu.vn](mailto:sonpvs@duytan.edu.vn)  Tel: 01675373265 | | | | |
| **Scrum Master & contact details** | Huynh Vu Ha Lan  Email: [huynhvuhalan.97@gmail.com](mailto:huynhvuhalan.97@gmail.com)  Tel: 0989342173 | | | | |
| **Team members** | **Name** | **Email** | | **Tel** | |
| Le Hoang Quoc | [quocle28292@gmail.com](mailto:quocle28292@gmail.com) | | 01657708105 | |
| Huynh Quoc Nhat | [huynhquocnhat97@gmail.com](mailto:huynhquocnhat97@gmail.com) | | 0905740256 | |
| Ho Trung Anh | [hotrunganhht1912@gmail.com](mailto:hotrunganhht1912@gmail.com) | | 09055258534 | |
|  | Huynh Thi Thanh Van | [thanhvandb97@gmail.com](mailto:thanhvandb97@gmail.com) | | 01229444661 | |

**Product Backlog Document**

|  |  |  |  |
| --- | --- | --- | --- |
| **Document Title** | Product Backlog Document |  |  |
| **Reporting Period** | August, 2018 |  |  |
| **Author(s) &**  **project** | Huynh Thi Thanh Van,  Huynh Vu Ha Lan | |  |
| **Team Information** | **Name** | **Role** |  |
| Le Hoang Quoc | Member |  |
| Huynh Quoc Nhat | Member |  |
| Ho Trung Anh | Member |  |
|  | Huynh Thi Thanh Van | Product Owner |  |
| **Date** | Aug 29, 2018 | **Filename** | Productbacklog\_Ver1.0.docx |
| **Access** | Project and CMU Program | |  |

|  |  |  |
| --- | --- | --- |
| Document History | | |
| **Version** | **Date** | **Comments** |
| V1.0 | Aug 29, 2019 | Draft for Document |
|  |  |  |

**Document Approvals**

The following signatures are required for approval of this document.

|  |  |  |
| --- | --- | --- |
| MSc. Vo Van Son  *Mentor* |  | Date: |
| Huynh Vu Ha Lan  *Scrum Master* |  | Date: |

Table of Contents

[1. Introduction 6](#_Toc499911195)

[1.1. Purpose 6](#_Toc499911196)

[1.2. Scope 6](#_Toc499911197)

[1.3. References 6](#_Toc499911198)

[2. Product Backlog 7](#_Toc499911199)

[2.1. Product Backlog Specification 7](#_Toc499911200)

[3. Breakdown into Sprint Backlog: 10](#_Toc499911201)

[4. Constraint 12](#_Toc499911202)

[5. Stack holders and User Descriptions Summary 13](#_Toc499911203)

1. **Introduction**

The agile product backlog in Scrum is a prioritized features list, containing short descriptions of all functionality desired in the product. When applying Scrum, it’s not necessary to start a project with a lengthy, upfront effort to document all requirements. Typically, a Scrum team and its product owner begin by writing down everything they can think of for agile backlog prioritization. This agile product backlog is almost always more than enough for a first sprint. The Scrum product backlog is then allowed to grow and change as more is learned about the product and its clients.

* 1. **Purpose**
* Provide a prioritized features list, containing short description of all functionality desired in the product.
* Lists everything that the product owner and Scrum team feels should be included in the software they are developing.
  1. **Scope**
* Lists the user’s role.
* Write all the user’s requirements.
* Lists some main function of website.
* Short description of all the functionality desired in the product.
* Given the priority of each feature and function of the product.
  1. **References**

***Table 1: References***

|  |  |  |
| --- | --- | --- |
| **No** | **Reference** | **Note** |
| 1 | <http://agilebench.com/blog/the-product-backlog-for-agile-teams> | How to create product backlog |
| 2 | <http://www.mountaingoatsoftware.com/agile/scrum/product-backlog/example/> | Product Backlog Example |

1. **Product Backlog**
   1. **Product Backlog Specification:**

**Priority and Estimates**

Priorities are set from 1 to 4.

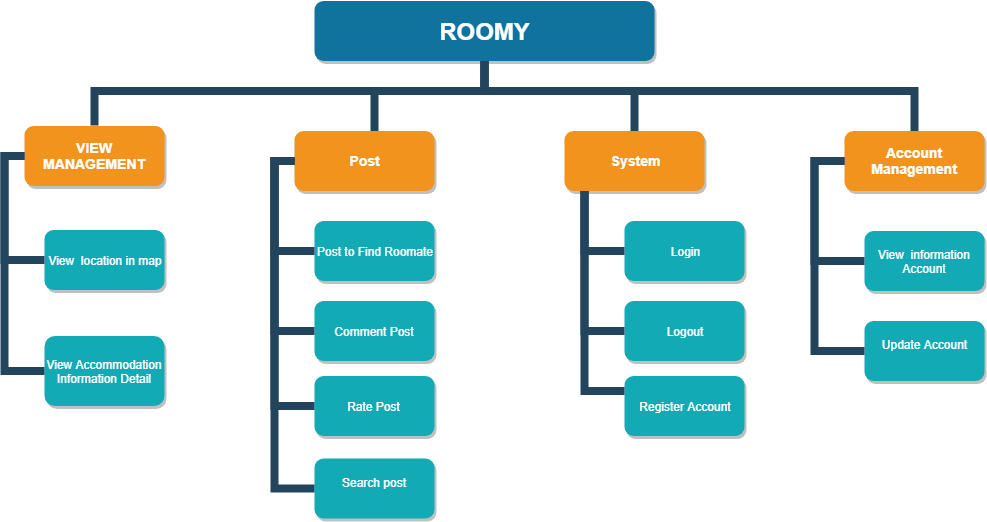
◦ The maximum priorities are 1

◦ The minimum prioritiies are 4

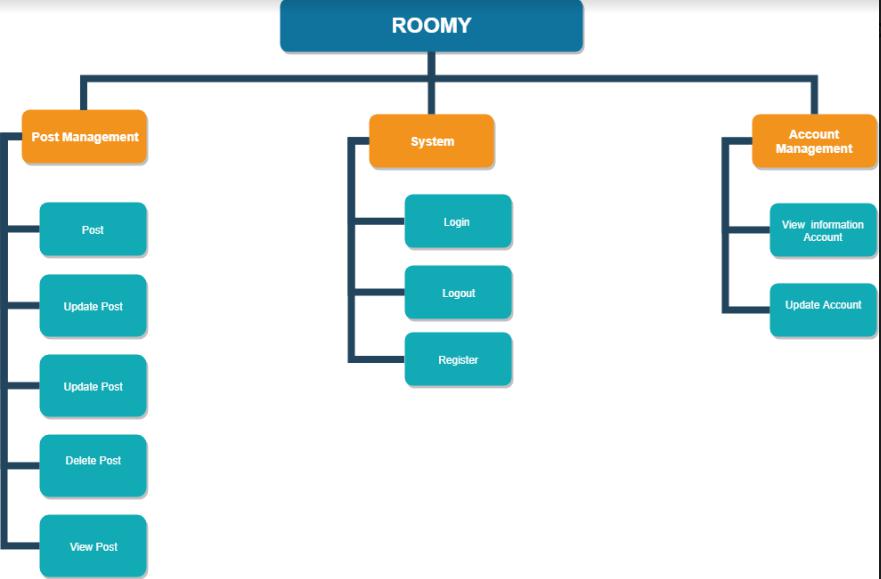
◦ Priorities of some user stories can be the same with others

◦ If a user story has dependencies, it must have lower priority than its dependencies

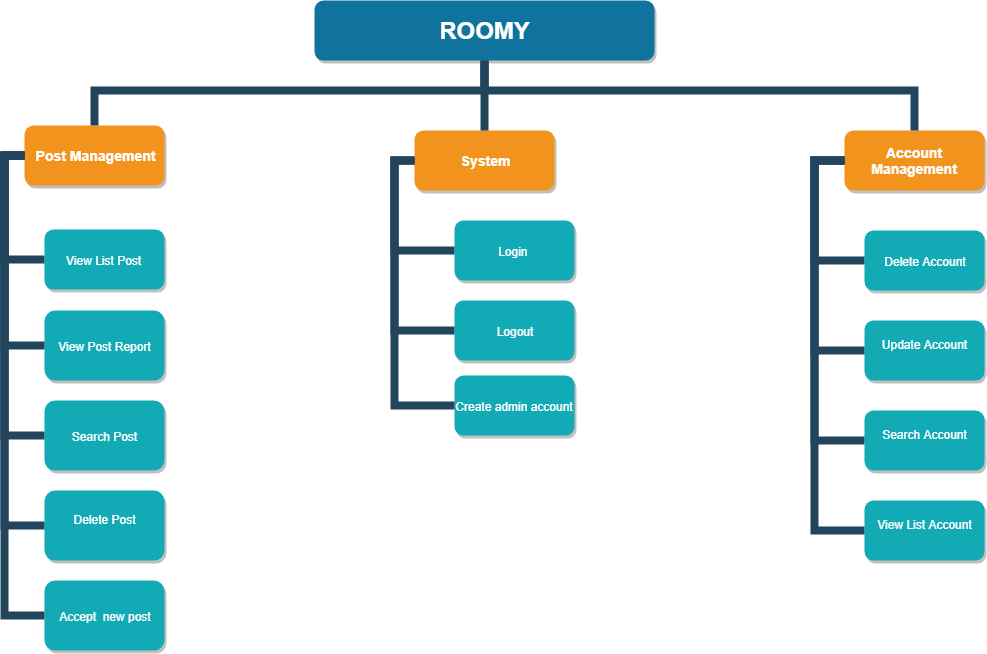
[**Work breakdown structure**](https://en.wikipedia.org/wiki/Work_breakdown_structure)

****

*Figure 1. Renter Diagram*

****

*Figure 2. Landlord Diagram*

****

*Figure 2. Admin Diagram*

***Table 1: Product Backlog Specification (Renter)***

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Theme** | **As a/an** | **I want to** | **So that** | **Priority** |
| **View** | | | | | |
| PB01 | View location on Map. | Renter | View location on Map | I can view my location myself on the map | 1 |
| PB02 | View detail. | Renter | View all information about accommodation | I can know exactly about accommodation information where I want to rent | 1 |
| **Post** | | | | | |
| PB03 | Post to find Rommate. | Renter | Post to find Rommate | I want to post to find roommate for me | 1 |
| PB04 | Search post. | Renter | Search posts by location | I can search for posts by the location I want | 2 |
| PB06 | Rate | Renter | Rate about quality of accommodation | I can rate the quality and authenticity of the accommodation | 4 |
| PB06 | Comment | Renter | Comment about accommodation | I can comment on the post for accommodation | 4 |
| **System** | | | | | |
| PB07 | Login. | Renter | Access to the system | I can use the system | 3 |
| PB08 | Logout. | Renter | Log out my account from system | I can stop using system and I can log in later | 3 |
| PB09 | Register. | Renter | Register account | I can register account to access system | 3 |
| **Account** | | | | | |
| PB10 | View information | Renter | View detail account information of myself. | I can know information of me such as username, password, name, address, phone number… | 1 |
| PB11 | Update Account | Renter | Edit account information of myself or password | I can edit any account information such as username, name, address, phone number… or password | 3 |

***Table 2: Product Backlog Specification (Landlord)***

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | | **Theme** | **As a/an** | | **I want to** | | **So that** | **Priority** | |
| **Post** | | | | | | | | | |
| PB01 | | Post the accommodation | Landlor | | Post the accommodation | | I can post all the information about accommodation | 1 | |
| PB02 | | Update post | Business | | access to the website by my account | | I can update my post which information has changed | 3 | |
| PB03 | | Delete post | Business | | Delete post | | I can delete my post when I find it is not useful | 3 | |
| PB04 | | View posts | Business | | View posts | | I can know the posts I posted. | 1 | |
| PB05 | | Search post |  | | Search posts in the system | | I can search for posts on the system | 2 | |
| **System** | | | | | | | | | |
| PB06 | | Login | Business | | Access to the system | | I can use the system | 3 | |
| PB07 | | Logout | Business | | Log out my account from system | | I can stop using system and I can log in later | 3 | |
| PB08 | | Register | Business | | Register account | | I can register account to access system | 3 | |
| **Account** | | | | | | | | | |
| PB09 | View information | | | Renter | View detail account information of myself. | I can know information of me such as username, password, name, address, phone number… | | | 1 | |
| PB10 | Update Account | | | Renter | Edit account information of myself or password | I can edit any account information such as username, name, address, phone number… or password | | | 3 | |

***Table 3: Product Backlog Specification (Admin)***

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Theme** | **As a/an** | **I want to** | **So that** | **Priority** |
| **View** | | | | | |
| PB01 | View list post | Admin | View posts that the user has posted | I can see the list of posts posted by the user | 1 |
| PB02 | View list report about Post | Admin | View user reports when their posts have problems | I can view user reports when their posts have problems to solve it. | 4 |
| PB03 | Search post | Admin | Searching list of posts in the system by post ID | I can see a port and its rate and comment. | 2 |
| PB04 | Accept new post | Admin | Accept posts of user posted | I can accept posts of user posted | 4 |
| PB05 | Deletele post | Admin | delete invalid post | During the review process I can delete posts that are not valid with the system's rules | 3 |
| **System** | | | | | |
| PB06 | Login | Admin | Access to the system | I can use the system | 3 |
| PB07 | Logout | Admin | Log out my account from system | I can stop using system and I can log in later | 3 |
| PB08 | Create account | Admin | Create account for new admin | I can create account for new admin to manage the system with me | 3 |
| Account | | | | | |
| PB09 | Delete account | Admin | delete the user's account | I can delete the user’s account when users have not used for a long time | 3 |
| PB10 | Update account | Admin | Update account password | I can update the password for my account | 3 |

1. **Breakdown into Sprint Backlog:**

***Table 4: Breakdown into Sprint Backlog***

|  |  |  |  |
| --- | --- | --- | --- |
| **Sprint** | **Function** | **Started Date** | **Due Date** |
|  | **Design user interface** |  |  |
| Design user interface of Home page(Web) | Sep 13, 18 | Sep 13, 18 |
| Design user interface of Register Account(Web) | Sep 14, 18 | Sep 14, 18 |
| Design user interface of Login (Web) | Sep 13, 18 | Sep 13, 18 |
| Design user interface of View Accommodation location Detail (Web) | Sep 14, 18 | Sep 14, 18 |
| Design user interface of Home page(App) | Sep 13, 18 | Sep 13, 18 |
| Design user interface of Register Account(App) | Sep 14, 18 | Sep 14, 18 |
| Design user interface of Login (App) | Sep 13, 18 | Sep 14, 18 |
| Design user interface of View Accommodation location Detail (App) | Sep 14, 18 | Sep 14, 18 |
| **Test Design Interface** | **Sep 15, 18** | **Sep 17, 18** |
| Design user interface of Home page(Web, App) | Sep 15, 18 | Sep 15, 18 |
| Design user interface of Register Account (Web, App) | Sep 15, 18 | Sep 15, 18 |
| Design user interface of Login (Web, App) | Sep 15, 18 | Sep 16, 18 |
| Design user interface of View Accommodation location Detail (Web, App) | Sep 15, 18 | Sep 17, 18 |
| **Coding** |  |  |
| Code front-end for Home page(Web) | Sep 16, 18 | Sep 16, 18 |
| Code back-end for Home page(Web) | Sep 16, 18 | Sep 18, 18 |
| Code front-end for Register Account (web) | Sep 17, 18 | Sep 17, 18 |
| Code back-end for Register Account (Web) | Sep 17, 18 | Sep 17, 18 |
| Code front-end for Login (Web) | Sep 18, 18 | Sep 18, 18 |
| Code back-end for Login (Web) | Sep 19, 18 | Sep 20, 18 |
| Code front-end View Accommodation location Detail (Web) | Sep 19, 18 | Oct 20, 18 |
| Code Back-end for View Accommodation location Detail (Web) | Sep 20, 18 | Sep 22, 18 |
| Code front-end for Home  page(App) | Sep 16, 18 | Sep 16, 18 |
| Code back-end for Home page(App) | Sep 17, 18 | Sep 19, 18 |
| Code front-end for Register(App) | Sep 17, 18 | Sep 17, 18 |
| Code back-end for Register (App) | Sep 17, 18 | Sep 18, 18 |
| Code front-end for Login (App) | Sep 19, 18 | Sep 20, 18 |
| Code back-end for Login (App) | Sep 21, 18 | Sep 22, 18 |
| Code front-end for View Accommodation location Detail (App) | Sep 18, 18 | Sep 19, 18 |
| Code back-end for View Accommodation location Detail (App) | Sep 19, 18 | Sep 22, 18 |
| **Testing** |  |  |
| Design Test Case for Homepage | Sep 23, 18 | Sep 23, 18 |
| Design Test Case for Register | Sep 23, 18 | Sep 23, 18 |
| Design Test Case for Login | Sep 23, 18 | Sep 23, 18 |
| Design Test Case for View Accommodation location Detail | Sep 23, 18 | Sep 23, 18 |
| **Fix Bug & Release** |  |  |
| Implement Test Case | Sep 24, 18 | Sep 26, 18 |
| Fix bugs and Report | Sep 27, 18 | Sep 30, 18 |
| Retest | Oct 1, 18 | Oct 2, 18 |
| Closing | Oct 28, 18 | Oct 28, 18 |
| Sprint 2 | **Design user interface** |  |  |
| Design user interface of Post to find Rommate (Web, Renter) | Oct 6, 18 | Oct 6, 18 |
| Design user interface of Search Post(Web, Renter) | Oct 6, 18 | Oct 6, 18 |
| Design user interface of Post the accommodation (Landlord) | Oct 6, 18 | Oct 7, 18 |
| Design user interface of Update Post ( Landlord) | Oct 6, 18 | Oct 7, 18 |
| Design user interface of Delete Post (Landlord) | Oct 6, 18 | Oct 7, 18 |
| Design user interface of View Post (Landlord) | Oct 7, 18 | Oct 7, 18 |
| Design user interface of Search Post (landlord) | Oct 7, 18 | Oct 7, 18 |
| Design user interface of Update Account | Oct 8, 18 | Oct 8, 18 |
| Design user interface of View Account Information | Oct 8, 18 | Oct 8, 18 |
| Design user interface of Post to find Rommate (Web, Renter) | Oct 6, 18 | Oct 6, 18 |
| Design user interface of Search Post(Web, Renter) | Oct 6, 18 | Oct 6, 18 |
| Design user interface of Post the accommodation (Landlord) | Oct 7, 18 | Oct 8, 18 |
| Design user interface of Update Post ( Landlord) | Oct 7, 18 | Oct 7, 18 |
| Design user interface of Delete Post (Landlord) | Oct 7, 18 | Oct 8, 18 |
| Design user interface of View Post (Landlord) | Oct 7,18 | Oct 8, 18 |
| **Coding** |  |  |
| Code front-end of Post to find Rommate(Web,Renter) | Oct 9, 18 | Oct 9, 18 |
| Code back-end of Post to find Rommate(Web,Renter) | Oct 10, 18 | Oct 11, 18 |
| Code front-end of Search Post(Web, Renter) | Oct 9, 18 | Oct 10, 18 |
| Code back-end of Search Post(Web, Renter) | Oct 11, 18 | Oct 12, 18 |
| Code front-end of Post the accommodation (Landlord) | Oct 10, 18 | Oct 11, 18 |
| Code back-end of Post the accommodation (Landlord) | Oct 12, 18 | Oct 15, 18 |
| Code front-end of Update Post ( Landlord) | Oct 12, 18 | Oct 12, 18 |
| Code back-end of Update Post ( Landlord) | Oct 13, 18 | Oct 14, 18 |
| Code front-end of Delete Post (Landlord) | Oct 13, 18 | Oct 13, 18 |
| Code back-end of Delete Post (Landlord) | Oct 14, 18 | Oct 14, 18 |
| Code front-end of View Post(Lordland) | Oct 15, 18 | Oct 15, 18 |
| Code back-end of Post(Lordland) | Oct 15, 18 | Oct 18, 18 |
| Code front-end of View Account Information | Oct 16, 18 | Oct 16, 18 |
| Code back-end View Account Information | Oct 16, 18 | Oct 17, 18 |
| Code front-end of Update Account | Oct 17, 18 | Oct 18, 18 |
| Code front-end of Update Account | Oct 19, 18 | Oct 20, 18 |
| Code front-end of Post to find Rommate(App,Renter) | Oct 9, 18 | Oct 10, 18 |
| Code back-end of Post to find Rommate(App,Renter) | Oct 10, 18 | Oct 13, 18 |
| Code front-end of Search Post(App, Renter) | Oct 9, 18 | Oct 10, 18 |
| Code back-end of Search Post(App, Renter) | Oct 11, 18 | Oct 14, 18 |
| Code front-end of Post the accommodation (Landlord) | Oct 12, 18 | Oct 13, 18 |
| Code back-end of Post the accommodation (Landlord) | Oct 13, 18 | Oct 16, 18 |
| Code front-end of Update Post ( Landlord) | Oct 15, 18 | Oct 15, 18 |
| Code back-end of Update Post ( Landlord) | Oct 15, 18 | Oct 16, 18 |
| Code front-end of Delete Post (Landlord) | Oct 16, 18 | Oct 17, 18 |
| Code back-end of Delete Post (Landlord) | Oct 17, 18 | Oct 18, 18 |
| Code front-end of View Post(Lordland) | Oct 17, 18 | Oct 17, 18 |
| Code back-end of View Post(Lordland) | Oct 18, 18 | Oct 18, 18 |
| Sprint 3 | **Design user interface** |  |  |
| Design user interface of Register | Nov 6, 17 | Nov 6, 17 |
| Design user interface of Manage Post | Nov 6, 17 | Nov 6, 17 |
| Design user interface of Update information account | Nov 6, 17 | Nov 6, 17 |
| **Coding** |  |  |
| Code front-end of Register | Nov 6, 17 | Nov 6, 17 |
| Code back-end of Register | Nov 6, 16 | Nov 7, 17 |
| Code front-end of Change password | Nov 6, 17 | Nov 7, 17 |
| Code back-end of Change password | Nov 7, 17 | Nov 7, 17 |
| Code front-end of Manage post on services/products | Nov 6, 17 | Nov 9, 17 |
| Code back-end of Manage post on services/products | Nov 6, 17 | Nov 10, 17 |
| Code front-end of Update information account | Nov 7, 17 | Nov 8, 17 |
| Code back-end of Update information account | Nov 7, 17 | Nov 7, 17 |
| Code back-end of View transaction | Nov 6, 17 | Nov 14, 17 |
| Code front-end of View transaction | Nov 7, 17 | Nov 8, 17 |
| Code front-end of post | Nov 7, 17 | Nov 11, 17 |
| Code back-end of post | Nov 6, 17 | Nov 15, 17 |
| Code back-end of Send ads | Nov 6, 17 | Nov 7, 17 |
| **Testing & Fix Bug** |  |  |
| Design Test Case for Register | Nov 8, 17 | Nov 14, 17 |

1. **Constraint**

**Table 5: Constraint**

|  |  |
| --- | --- |
| **Constraint** | **Condition** |
| **Time** | Project completion time limit in 3 months so time to complete project be restricted |
| **People constraint** | 4 people working together to finish the project |
| **Requirements** | According to Product owner’s Requirements |

1. **Stack holders and User Descriptions Summary**

***Table 6: Stakeholders and User Descriptions Summary***

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Role** |
| **Product Owner** | The Person who give the Requirement | Provide information to develop the website. Make the decision to accept and implement the project, do the unit test |
| **Scrum Master** | This is the stakeholder who leading, manage the website development Team | Controlling, managing, monitoring, make sure the project complete on time, within budget, according to plan and according to requirements |
| **Requirement Analyzer** | This is a stakeholder that works with the Analysts to correctly translate requests or needs into requirements to be used for design | Specifies the details of one or more a parts of the website’s functionality by describing one or the aspects of the requirements,  This will include functional and non-functional |
| **Software Architect** | This is a stakeholder that is primary for leading the website development | Responsible for the software architecture, which includes the key technical decisions that constrain the overall design for the project |
| **Coder** | This is a stakeholder that programs the software | Implement the project |