

**CAPSTONE PROJECT 1**

**ARCHITECTURE DESIGN DOCUMENT**

**ROOMY SYSTEM MANAGEMENT**

**CODE:** ARCHITECTURE DESIGN - Version: 1.2

**DATE:** Sep 10th, 2018

**Blue Team**

International School – Duy Tan University

**PROJECT INFORMATION**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Project Title | Roomy System | | | |
| Star Date | Aug 15th, 2018 | **End Date** | Dec 5th, 2018 | |
| Lead Institution | International School, Duy Tan University | | | |
| Project Mentor | Mr. Phan Van Son | | | |
| Project Manager & Contact Details | Phan Van Son  Email: [sonpvs@duytan.edu.vn](mailto:sonpvs@duytan.edu.vn)  Phone: 0905151655 | | | |
| Scrum Master & contact details | Huynh Vu Ha Lan  Email: [huynhvuhalan.97@gmail.com](mailto:huynhvuhalan.97@gmail.com)  Tel: 0989342173 | | | |
| Team Members | 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 |

*Table 1. Project information*

**ARCHITECTURE DESIGN DOCUMENT**

|  |  |  |  |
| --- | --- | --- | --- |
| Document Title | Architecture Design | | |
| Reporting Period | Sep, 2018 |  |  |
| Author(s) & project | Blue Team | | |
| Date | Sep 20th 2018 | **Fieldname** | Architecture design ver 1.2 |
| Access | Project and CMU program | | |

*Table 2. Architecture design document*

**DOCUMENT HISTORY**

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Person | Date | Description |
| 1.0 | Team | Sep 7th, 2018 | Draft. |
| 1.1 | Huynh Vu Ha Lan | Sep 9th, 2018 | Update Introdustion |
| 1.2 | Le Hoang Quoc | Sep 20th, 2018 | Fix C&C view, Module view. |

*Table 3. Document history*

**SIGNATURE**

**Document Approvals:** The following signatures are required for approval of this document.

|  |  |  |
| --- | --- | --- |
| Phan Van Son Mentor |  | Date |
| Huynh Vu Ha Lan  Scrum Master |  | Date |
| Huynh Thi Thanh Van Member |  | Date |
| Le Hoang Quoc  Member |  | Date |
| Ho Trung Anh  Member |  | Date |
| Huynh Quoc Nhat  Member |  | Date |

*Table 4. Document approvals*

Contents

[**1.** **Introduction:** 1](#_Toc532004451)

[**1.1.** **Purpose:** 1](#_Toc532004452)

[**1.2.** **Documents References:** 1](#_Toc532004453)

[**2.** **Project Statement:** 1](#_Toc532004454)

[**2.1.** **Project Overview:** 1](#_Toc532004455)

[**2.2.** **Business Driver:** 2](#_Toc532004456)

[**3.** **Architecture Driver:** 3](#_Toc532004457)

[**3.1.** **High-Level Requirements:** 3](#_Toc532004458)

[**3.2.** **System Context:** 4](#_Toc532004459)

[**3.2.1.** **Use case diagram:** 4](#_Toc532004460)

[4](#_Toc532004461)

[*Figure 1. Use case diagram* 4](#_Toc532004462)

[**3.2.2.** **Context diagram:** 5](#_Toc532004463)

[**3.3.** **Quality Attributes:** 6](#_Toc532004464)

[**4.** **Constraint:** 8](#_Toc532004465)

[**4.1.** **Business Constraint:** 8](#_Toc532004466)

[**4.2.** **Technical Constraint:** 8](#_Toc532004467)

[**5.** **High level Architecture** 8](#_Toc532004468)

[**5.1.** **Component and Connection View (C&C View):** 9](#_Toc532004469)

[**5.2.** **Module view:** 11](#_Toc532004470)

[**5.3.** **Allocation view :** 13](#_Toc532004471)

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. **Documents References:**

|  |  |
| --- | --- |
| **No** | **References** |
| **1** | Product Backlog Document |
| **2** | Project Plan Document |

*Table 5. Document references*

1. **Project Statement:**
   1. **Project Overview:**

* Project name: Roomy System.
* Development team: Blue Team

|  |  |  |  |
| --- | --- | --- | --- |
| Full Name | Phone | Email | Position |
| Huynh Vu Ha Lan | 0989342173 | [huynhvuhalan.97@gmail.com](mailto:huynhvuhalan.97@gmail.com) | Scrum Master |
| Huynh Thi Thanh Van | 01229444661 | [thanhvandb97@gmail.com](mailto:thanhvandb97@gmail.com) | Member |
| Huynh Quoc Nhat | 0905740256 | [huynhquocnhat97@gmail.com](mailto:huynhquocnhat97@gmail.com) | Member |
| Le Hoang Quoc | 01657708105 | [quocle28292@gmail.com](mailto:quocle28292@gmail.com) | Member |
| Ho Trung Anh | 09055258534 | [hotrunganhht1912@gmail.com](mailto:hotrunganhht1912@gmail.com) | Member |

*Table 6. Team Members*

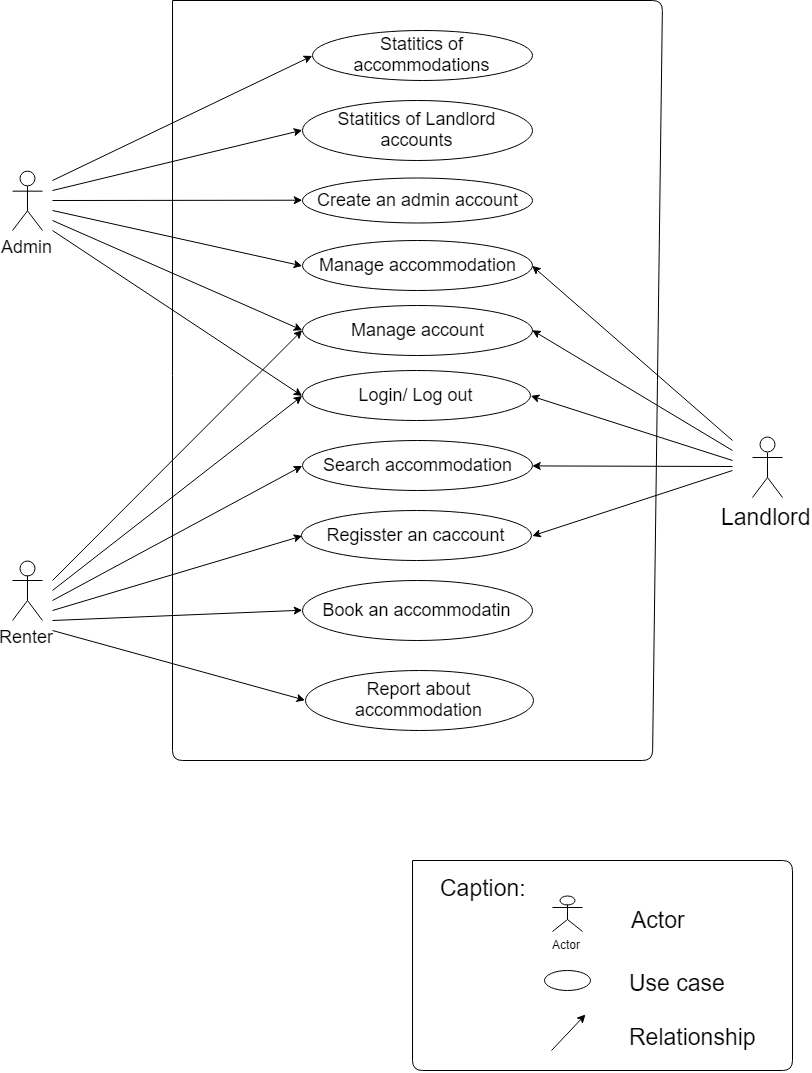
* 1. **Business Driver:**
* **Project Definition:**
  + - Roomy System is known as a system to help users find accommodation suitable for their needs.
    - This system allows landlords to provide information about the places they want to rent and manage them; allows renter to search for information on accommodation based on their needs from which to choose the right accommodation.
* **Business needs:**
* Landlords provide information about the accommodation they want to rent and manage them.
* Renters search for information on accommodation based on their needs.
* Users can see the visual position of the accommodations through maps and 360-degree imagery.
* Renters can booking an available accommodation and that booking will expired after 48 hours if it was not accepted by landlord.
* Landlords have he right to change prices and accept or reject the booking.
* Renters can report about accommodation if it contains incorrectly information.
* Admins can see list of report form renters, manage all accounts and accommodations in system.
* Admin can see statistics of the users and accommodations.

1. **Architecture Driver:**
   1. **High-Level Requirements:**

|  |  |
| --- | --- |
| User Story ID | Priority |
| US01 | 3 |
| US02 | 3 |
| US03 | 3 |
| US04 | 2 |
| US05 | 1 |
| US06 | 1 |
| US07 | 3 |
| US08 | 2 |
| US09 | 1 |
| US10 | 1 |
| US11 | 1 |
| US12 | 2 |
| US13 | 1 |
| US14 | 2 |
| US15 | 2 |

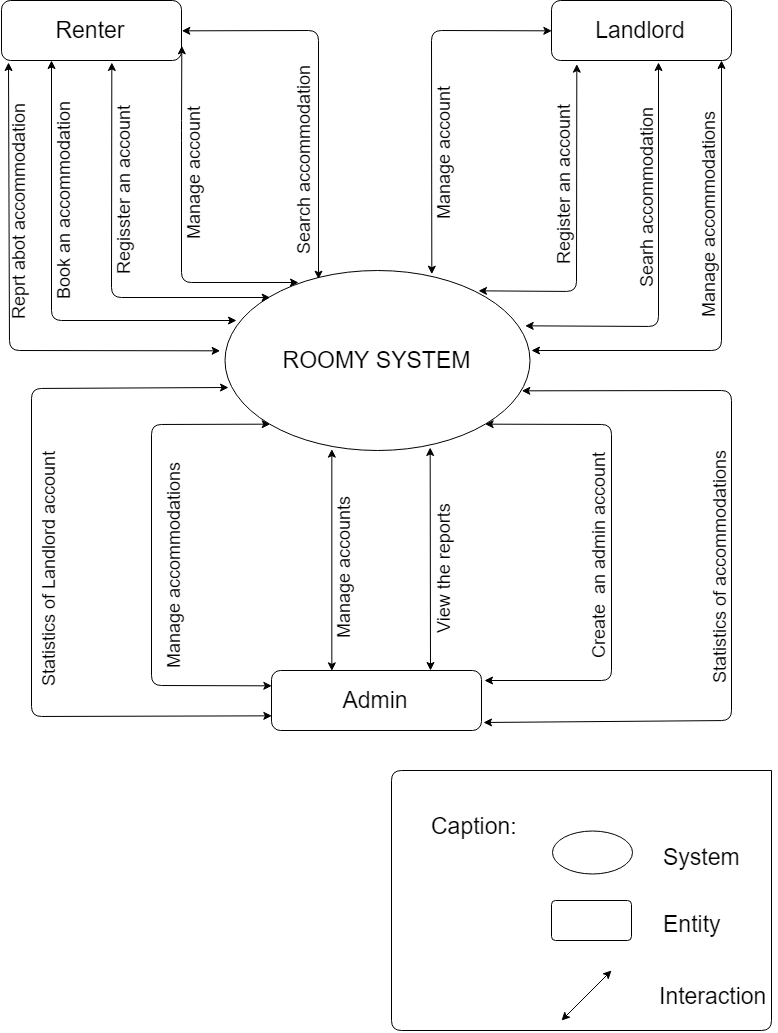
*Table 7. High-Level Requirements*

* 1. **System Context:**
     1. **Use case diagram:**

****

*Figure 1. Use case diagram*

* + 1. **Context diagram:**

****

*Figure 2. Context Diagram*

* 1. **Quality Attributes:**

|  |  |
| --- | --- |
| **Quality Attributes** : Availability | **ID** : QA01 |
| **Stimulus** | User access in system any where and any time |
| **Source of the stimulus** | Human |
| **Environment** | Website |
| **Artifact** | Normal operation |
| **Response** | Accept access |
| **Response measure** | System work 24/7 |

Table 8. Quality Attributes: Availability

|  |  |
| --- | --- |
| **Quality Attributes** : Performance | **ID** : QA02 |
| **Stimulus** | User request |
| **Source of the stimulus** | Human |
| **Environment** | Normal operation |
| **Artifact** | The system |
| **System response** | Process the request and return the result |
| **Response measure** | Response in less than 3 seconds |

Table 9: Quality Attributes: Performance

|  |  |
| --- | --- |
| **Quality Attributes** : Security | **ID** : QA03 |
| **Stimulus** | User log in system with account and pasword which was regisstered. |
| **Source of the stimulus** | Human |
| **Environment** | System |
| **Artifact** | Log in process |
| **System response** | User log in system and use system functions. |
| **Response measure** | An e-mail only use for one account. |

*Table 10: Quality Attributes: Security*

|  |  |
| --- | --- |
| **Quality Attributes** : Compatibility | **ID** : QA04 |
| **Stimulus** | There are a lot of users access in the system. |
| **Source of the stimulus** | Human |
| **Environment** | Normal operation |
| **Artifact** | The system. |
| **System response** | Accept access |
| **Response measure** |  |

Table 11: Quality Attributes: Capability

|  |  |
| --- | --- |
| **Quality Attributes** : Usability | **ID** : QA05 |
| **Stimulus** | Users can easily use the system at the first time. |
| **Source of the stimulus** | Human |
| **Environmental** | Normal operation |
| **Artifact** | System. |
| **System response** | System supports Vietnamese |
| **Response measure** | A hotline to support users |

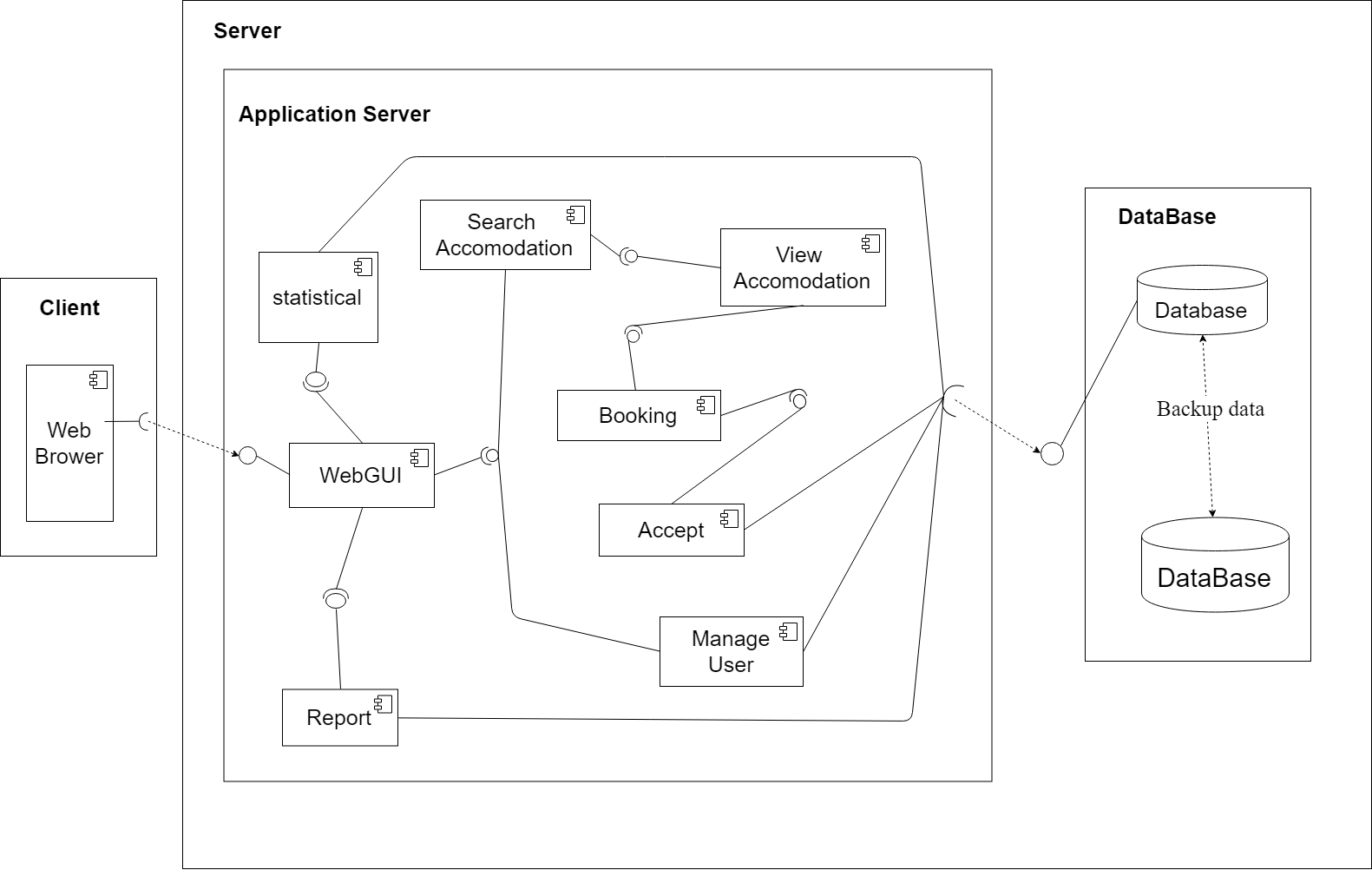
Table 12: Quality Attributes: Usability

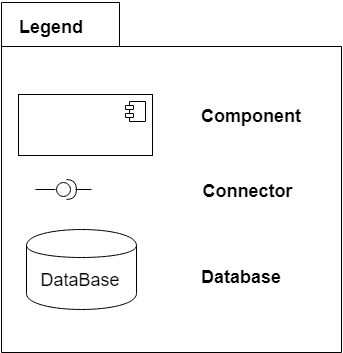
1. **Constraint:**
   1. **Business Constraint:**
      * Project will be started on: 15/08/2018
      * Project will be finished on: 5/12/2018
      * Project will be finished in: 90 days.
   2. **Technical Constraint:**

* **Technical for development:**
  + - * Language: PHP, HTML, CSS, Java, Android, JavaScript.
      * Develop Tools: Sublime Text, Notepad++, Android studio, Xampp.
      * Database: MySQL
  + Operating system: Multi-platform
* **Environment:**
* Web browsers: Internet Explorer, Mozilla Firefox, Apple Safari, Google Chrome, and so on.
* Operator systems: Microsoft Windows, Linux, and Apple Mac OS.
* **Other Constraints:**
  + Resource: 5 people.
  + Budget: Limited.
  + Time: The project must be completed within 4 months frame.

1. **High level Architecture** 
   1. **Component and Connection View (C&C View):**

The diagram below shows the overview architecture including component and other related component. We have representations and behaviors for important components in the following sections.





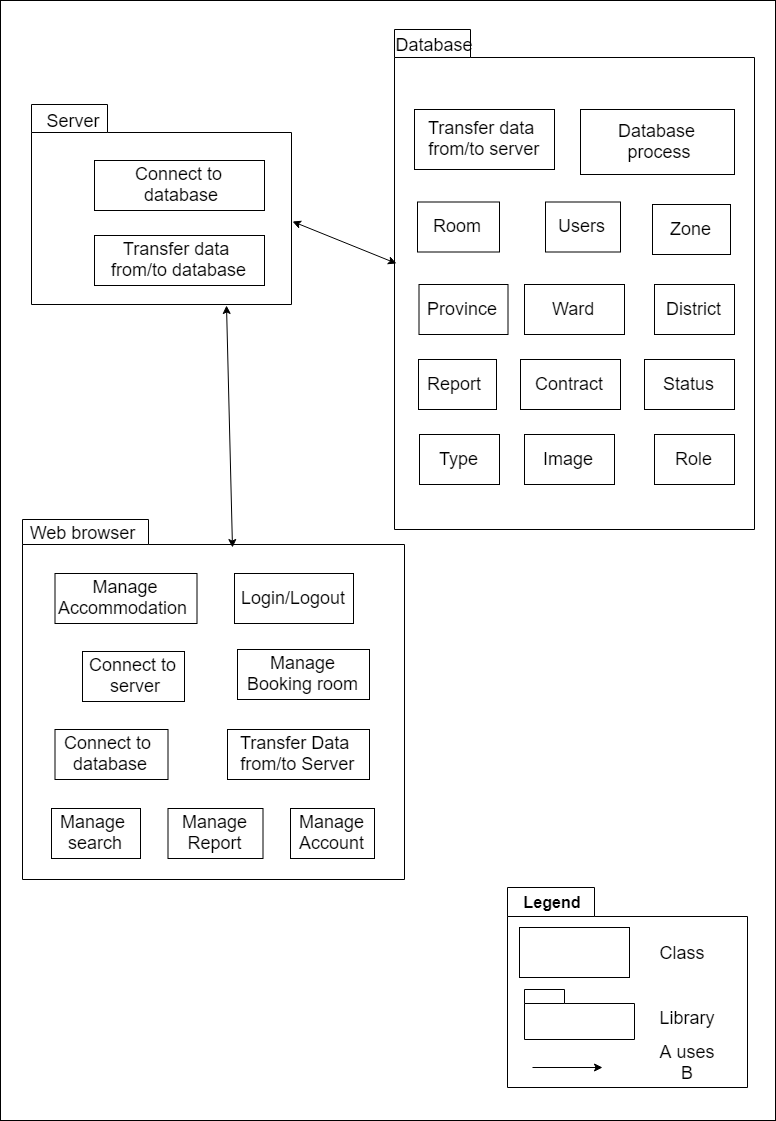
*Figure 3: C&C View*

* Prose:
* One user accessed to system, the front-end interface will be displayed. With these front-end interfaces, user can go around the entire of the application. They can know which functions are on system easily. From that, database from server can be transacted to users by back-end working. The entire of website will have some main functions that will be presented in the table blow.
* The table below gives a brief explanation for the key elements in the C&C view:

|  |  |
| --- | --- |
| **Element** | **Description** |
| Homepage | The element will allow users to see homepage of tools. |
| Draft | The element will allow users to open [Drafting] Form |
| Select Text | The element will allow users to select type of administrative documents which they want editor. |
| Text Editor | The element will allow users to text editor. |
| Preview | The element will allow users to preview the text after they are text editor complete. |
| Save | The element will allow users to save data. |
| Print | The element will allow users to print document. |
| Export | The element will allow users to export document. |
| History | The element will allow users to open [History] Form |
| Select | The element will allow users to select type of administrative documents which they want view history. |
| Search | The element will allow users to search administrative documents. |
| Edit Text | The element will allow users to edit the old administrative documents. |
| Delete Text | The element will allow users to delete the administrative documents out of history. |
| Administrative documents management | The element is save all information about administrative documents. |

*Table 13. Brief explanation for the key elements in C&C view*

* 1. **Module view:**

****

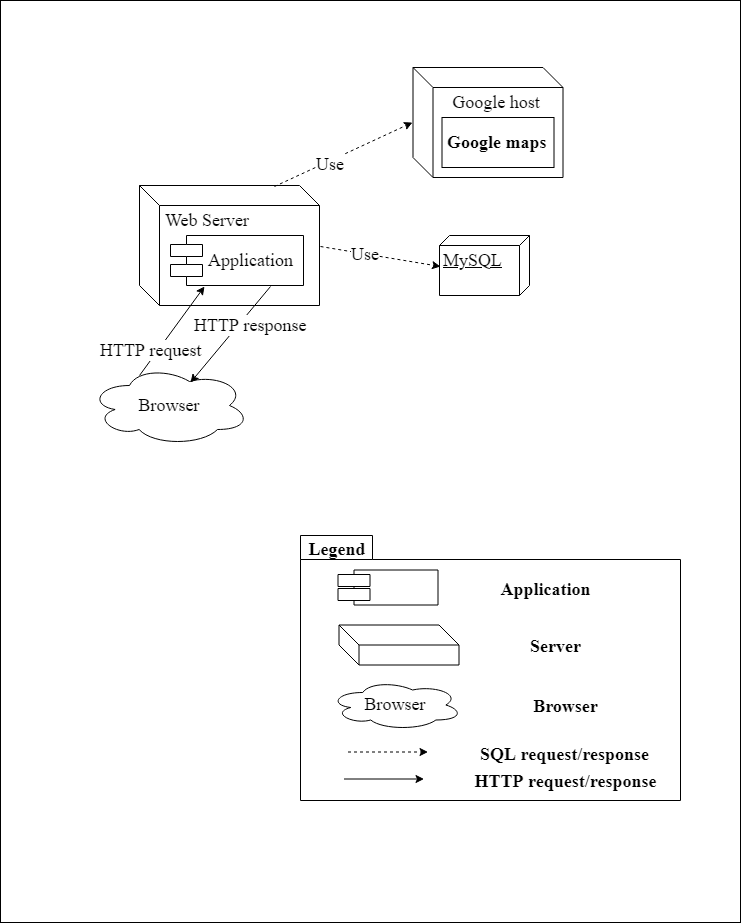
*Figure 4: Module View*

* Prose:
* These modules have relationship and interactive each other to create this tool. The functionality interact each other, relationship will be defined by arrows.

|  |  |
| --- | --- |
| Element | Description |
| Homepage | This module is responsible for showing homepage of tool. |
| Draft | This module is responsible for showing [Drafting] form. |
| Select Text | This module is responsible for selecting type of administrative documents which they want editor. |
| Text Editor | This module is responsible for editing the text by form that user want to edit. |
| Preview | This module is responsible for previewing administrative documents after user is complete. |
| Save | This module is responsible for saving administrative document. |
| Print | This module is responsible for printing administrative documents. |
| Export | This module is responsible for exporting administrative documents. |
| History | This module is responsible for showing [History] form. |
| Select | This module is responsible for selecting type of administrative documents which they want view history. |
| Search | This module is responsible for searching administrative documents. |
| Edit Text | This module is responsible for editing the old administrative documents. |
| Delete Text | This module is responsible for deleting the administrative documents out of history. |

*Table 14: Brief explanation for the key elements in Module view*

* 1. **Allocation view :**



*Figure 5: Allocation View*