**BIT302**

**Software Engineering**

****

**ASSIGNMENT 1**

**Requirement Document**

**“Web-based Information System for MicroHousing System in Kuala Lumpur”**

**Team Leader:**

**Luh Wulandari Maharani**

**E1700873 / 170030401**

**luhwulandari@gmail.com**

**Member:**

**Rivaldo Bagus Soepardhy**

**E1700882 / 170030400**

**aldobagus@hotmail.co.id**

Table of Content

[Functional Requirements 1](#_Toc33546744)

[Non – functional Requirements 2](#_Toc33546745)

[Use Case Diagram 3](#_Toc33546746)

[Use Cases 4](#_Toc33546747)

[Expanded Use Cases 6](#_Toc33546748)

[Analysis Class Diagram 13](#_Toc33546749)

[Tasks Division 14](#_Toc33546750)

Functional Requirements

**HousingOfficer requirement:**

1. HousingOfficer has login menu that can be filled with user ID and password to go directly to HousingOfficer homepage.
2. HousingOfficer can change the password if housingofficer forget their password.
3. HousingOfficer must have “edit menu” which is can edit residence detail.
4. HousingOfficer must have “add menu” which is can set up new residence.
5. Housing officer must have “delete menu” which is can delete applicant and residence detail.
6. Housing officer must have “view menu” which is can view applications and residence details.
7. Payment menu to display payment details for applicant which is designed by HousingOfficer
8. Log out Menu to exit from the application.

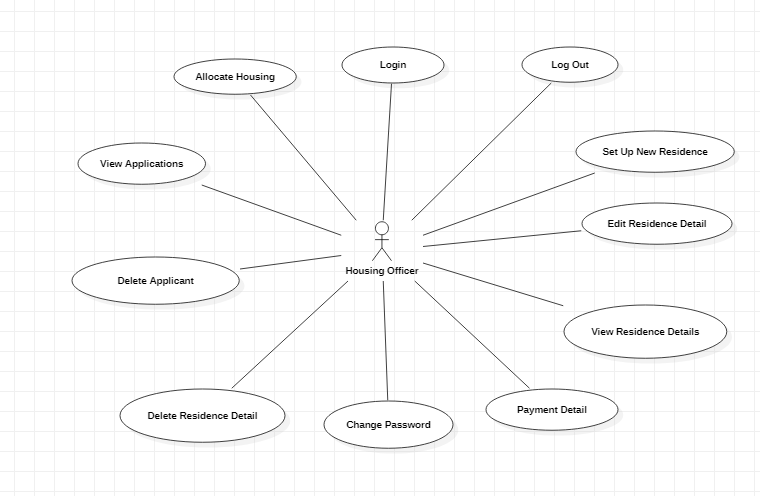
**Applicant Requirement:**

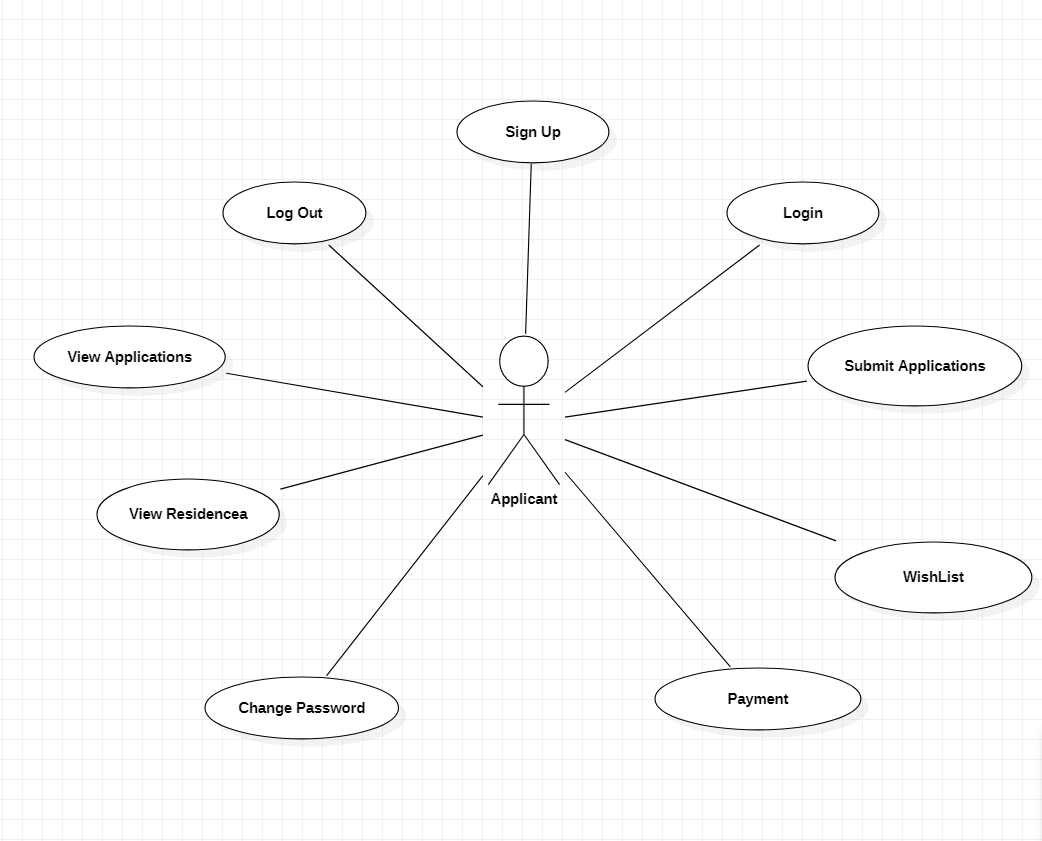
1. Applicant will register where the form has been given.
2. Applicant has login menu that can be filled with user ID and password so that the applicant can access the system without confusion, and will be directed to the home page.
3. Applicant can change the password if applicant forget their password.
4. The system has a menu with buttons or icons that applicants can choose from view application, view residences, wish list, submit applications and payment.
5. Log out Menu to exit from the application.

Non – functional Requirements

1. Security requirement: using MD5 encryption for the password to make sure if someone without authorization can access our database, he/she can’t understand the password
2. Usability: the system should be easy to operate by admin and the features understood by students
3. Accuracy and precision: data entered by admin should be accurate and precise and free from error
4. Modifiability: data can only be modified by admin through admin page and students are only allowed to change his/her own password

Use Case Diagram





Use Cases

|  |  |  |
| --- | --- | --- |
| Requirement | Actor | Use Case |
| Login page created for HousingOfficer and applicants to login | HousingOfficer  Applicant | Login |
| HousingOfficer and Applicant can access the system and login with the new password | HousingOfficer  Applicant | Change Password |
| HousingOfficer dan Applicant can exit the application through the system | HousingOfficer  Applicant | Log Out |
| The system displays a page where that can display Applications and validation for the applicant | HousingOfficer | View Applications |
| The system displays a page where that can display residence details | HousingOfficer | View Residence Detail |
| The system allocates housing where the applicant has ordered a slot for a long time but will be verified by HousingOfficer | HousingOfficer | Allocate Housing |
| HousingOfficer can input several new residences to the system | HousingOfficer | Set Up New Residence |
| Residence details will be changed for some data and will be recorded in the system by HousingOfficer | HousingOfficer | Edit Residence Detail |
| Applicants can be removed if needed on the system by the HousingOfficer | HousingOfficer | Delete Applicant |
| Residence Detail can be removed if needed on the system by HousingOfficer | HousingOfficer | Delete Residence Detail |
| Payment Details will display the payment form which is done by the system and designed by HousingOfficer | HousingOfficer | Payment Detail |
| The system has a page which will display applications that can be seen by applicant | Applicant | View Applications |
| The system has a page which will display residence details that can be seen by applicant | Applicant | View Residences |
| The submit page will display a form to be filled by the applicant and will be saved by the system | Applicant | Submit Applications |
| WishList page will display which housing the applicant likes | Applicant | WishList |
| The payment page will automatically be display by the system and the applicant can see the payment details | Applicant | Payment |

Expanded Use Cases

1. Login

|  |  |
| --- | --- |
| Use Case | Login |
| Goal in Context | Allow HousingOfficer and Applicant access the main page |
| Primary Actor  Secondary Actor | HousingOfficer and Applicant  - |
| Typical Course of Events | |
| Actor Actions | **System Response** |
| 1. The process starts when the user input their ID user and password and press the login button | 1. The system will validate the information received |
|  | 1. System will display the homepage for HousingOfficer or Applicant |
| Alternative Course | |
| If the HousingOfficer or Applicant inputs an incorrect user ID or password then the system will display notification if the information entered is incorrect, and HousingOfficer or Applicant must fill in with correct user ID and password. | |

1. Change Password

|  |  |
| --- | --- |
| Use Case | Change Password |
| Goal in Context | Allow HousingOfficer dan Applicant changed their password |
| Primary Actor  Secondary Actor | HousingOfficer dan Applicant  - |
| Typical Course of Events | |
| Actor Actions | **System Response** |
| 1. Process occurs when the HousingOfficer or Applicant login |  |
| 1. HousingOfficer or Applicant will select the Forget Password button | 1. System will display a form for change password that must be filled with a new password |
| 1. HousingOfficer or Applicant will fill in with new password and one – time validation |  |
| 1. HousingOfficer or Applicant will press the submit button |  |
|  | 1. System will save the update in the database |
| Alternative Course | |
| If there is a third of mismatch password event, then a message will appear and the new password will not be submitted | |

1. Log Out

|  |  |
| --- | --- |
| Use Case | Log Out |
| Goal in Context | Allow HousingOfficer dan Applicant exit from the application |
| Primary Actor  Secondary Actor | HousingOfficer dan Applicant  - |
| Typical Course of Events | |
| Actor Actions | **System Response** |
| 1. The process starts when the user clicks the Log Out button | 1. System will exit the user from the main page |
| Alternative Course | |
|  | |

1. View Applications

|  |  |
| --- | --- |
| Use Case | View Applications |
| Goal in Context | Allow HousingOfficer to see the whole applications form |
| Primary Actor  Secondary Actor | HousingOfficer  Applicant |
| Typical Course of Events | |
| Actor Actions | **System Response** |
| 1. HousingOfficer can access the page where there is a list of applicants | 1. System will display page that containing the applicant |
| Alternative Course | |
|  | |

1. View Residence Detail

|  |  |
| --- | --- |
| Use Case | View Residence Detail |
| Goal in Context | Allow HousingOfficer to see the whole residence detail form |
| Primary Actor  Secondary Actor | HousingOfficer  - |
| Typical Course of Events | |
| Actor Actions | **System Response** |
| 1. HousingOfficer can access the page where there is a list of residence | 1. System will display page that containing the residence detail |
| Alternative Course | |
|  | |

1. Allocate Housing

|  |  |
| --- | --- |
| Use Case | Allocate Housing |
| Goal in Context | Allow HousingOfficer to book applicant who have booked slots that have long been ordered and will be sorted again |
| Primary Actor  Secondary Actor | HousingOfficer  - |
| Typical Course of Events | |
| Actor Actions | **System Response** |
| 1. HousingOfficer will check the applicants one by one |  |
| 1. HousingOfficer will select the applicant who has chosen the residence first | 1. The system will save the information that has been entered by HousingOfficer |
| Alternative Course | |
|  | |

1. Set Up New Residence

|  |  |
| --- | --- |
| Use Case | Set Up New Residence |
| Goal in Context | Allow HousingOfficer to input new residence information |
| Primary Actor  Secondary Actor | HousingOfficer  - |
| Typical Course of Events | |
| Actor Actions | **System Response** |
| 1. The process occurs when HousingOfficer input new residence information | 1. The system will save new data |
| Alternative Course | |
|  | |

1. Edit Residence Detail

|  |  |
| --- | --- |
| Use Case | Edit Residence Detail |
| Goal in Context | Allow HousingOfficer to change the information about the residence |
| Primary Actor  Secondary Actor | HousingOfficer  - |
| Typical Course of Events | |
| Actor Actions | **System Response** |
| 1. HousingOfficer checks the residence that want to change |  |
| 1. HousingOfficer will choose which residence to change | 1. The system will display the residence page |
| 1. HousingOfficer will input ne data that change and selected by HousingOfficer | 1. System will save changes and will update the information that will be displayed in residence detail page |
| Alternative Course | |
| HousingOfficer akan | |

1. Delete Applicant

|  |  |
| --- | --- |
| Use Case | Delete Applicant |
| Goal in Context | Allow HousingOfficer to delete information about the Applicant |
| Primary Actor  Secondary Actor | HousingOfficer  - |
| Typical Course of Events | |
| Actor Actions | **System Response** |
| 1. HousingOfficer will select the Applicant that they want delete | 1. System will receive information which will be deleted |
|  | 1. System will update information and will be updated in the database |
| Alternative Course | |
| - | |

1. Delete Residence Detail

|  |  |
| --- | --- |
| Use Case | Delete Residence Detail |
| Goal in Context | Allow HousingOfficer to delete information about the Residence |
| Primary Actor  Secondary Actor | HousingOfficer  - |
| Typical Course of Events | |
| Actor Actions | **System Response** |
| 1. HousingOfficer will choose which residence they want delete | 1. System will receive information which will be deleted |
|  | 1. System will update information and will be updated in the database |
| Alternative Course | |
| - | |

1. Payment Detail

|  |  |
| --- | --- |
| Use Case | Payment Detail |
| Goal in Context | Allow HousingOfficer to make a detailed payment form |
| Primary Actor  Secondary Actor | HousingOfficer  - |
| Typical Course of Events | |
| Actor Actions | **System Response** |
| 1. HousingOfficer will edit the payment form details that have been made | 1. The system will save all information that has been entered by the applicant |
| Alternative Course | |
| - | |

1. View Applications

|  |  |
| --- | --- |
| Use Case | View Applications |
| Goal in Context | Allow Applicant to see the whole other applicants |
| Primary Actor  Secondary Actor | Applicant  HousingOfficer |
| Typical Course of Events | |
| Actor Actions | **System Response** |
| 1. Applicant can access the entire Applications page | 1. The system will display the Applications page |
| Alternative Course | |
|  | |

1. View Residences

|  |  |
| --- | --- |
| Use Case | View Residences |
| Goal in Context | Allow Applicant to see the whole other applicants |
| Primary Actor  Secondary Actor | Applicant  HousingOfficer |
| Typical Course of Events | |
| Actor Actions | **System Response** |
| 1. Applicant can access the entire residences page | 1. The system will display the Residences page |
| Alternative Course | |
|  | |

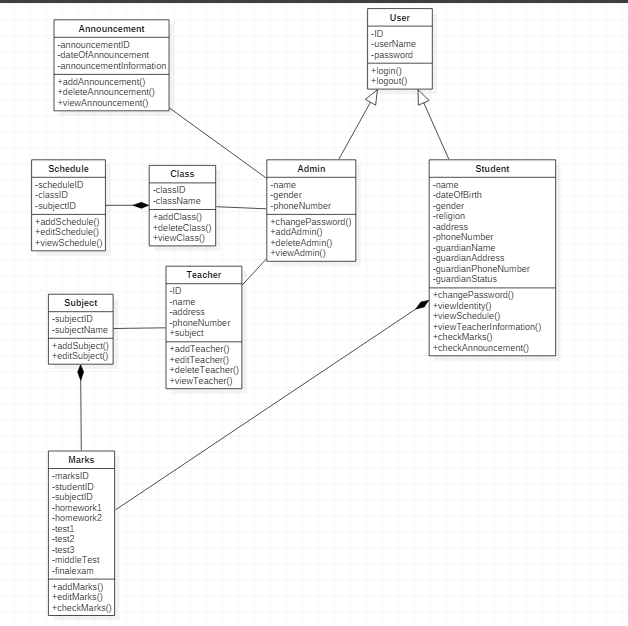
1. Submit Applications

|  |  |
| --- | --- |
| Use Case | Submit Applications |
| Goal in Context | Allow Applicant to enter information about the new applicant |
| Primary Actor  Secondary Actor | Applicant  HousingOfficer |
| Typical Course of Events | |
| Actor Actions | **System Response** |
| 1. Applicant will choose which Residence you want to occupy |  |
| 1. Applicant will fill in the form provided | 1. The system will save all information that has been filled |
| Alternative Course | |
| If the information entered by the applicant an incorrect, the applicant will refill it again | |

1. Payment Detail

|  |  |
| --- | --- |
| Use Case | Payment Detail |
| Goal in Context | Allow Applicant to see payment details |
| Primary Actor  Secondary Actor | Applicant  - |
| Typical Course of Events | |
| Actor Actions | **System Response** |
| 1. The process occurs when the applicant has finished entering all the information that has been submitted | 1. The system will check all information that has been submitted |
|  | 1. The system will display the payment in detail |
| 1. Applicants will choose their payment method | 1. The system will direct the applicant to make a transaction |
|  | 1. The system will validate |
| Alternative Course | |
| If the Applicant incorrectly enters information or chooses the payment method, the transaction will not occur. | |

Analysis Class Diagram



Tasks Division

|  |  |
| --- | --- |
| Required Behaviour | Member’s Name |
| HTML and CSS script for designing the website menu and navigation | Lovely |
| PHP script for login | Lovely |
| PHP script for add new admin, delete and admin | Lovely |
| PHP script to change password | Denny |
| HTML and CSS script to design how data will be displayed when being viewed | Denny |
| PHP script for edit student, teacher, marks, class and schedules | Lovely and Denny |
| PHP script for search column in admin home page | Lovely |
| PHP script for posting announcement from admin to student home page | Denny |