WEB-BASED HOUSE RENTAL APPLICATION SYSTEM

NURUL AFIQAH BINTI SAMSURI

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

WEB-BASED HOUSE RENTAL APPLICATION SYSTEM

NURUL AFIQAH BINTI SAMSURI

This report is submitted in partial fulfillment of the requirements for the Bachelor of Computer Science (Software Development) with Honours.

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2021

DECLARATION

I hereby declare that this project report entitled

WEB-BASED HOUSE RENTAL APPLICATION SYSTEM

is written by me and is my own effort and that no part has been plagiarized without citations.

STUDENT : nurul afiqah Date : 13/9/2021
(NURUL AFIQAH BINTI SAMSURI)

I hereby declare that I have read this project report and found this project report is sufficient in term of the scope and quality for the award of Bachelor of Computer Science (Software Development) with Honours.

SUPERVISOR : Date : 13 September 2021
(EMALIANA BINTI KASMURI)

DEDICATION

This project is dedicated to my loving parents, who have always been a source of inspiration for me, giving me strength when I was on the verge of giving up, and who continue to support me morally, spiritually, emotionally, and financially.

To my supervisor, siblings, friends, and classmates all offered words of support and guidance to help me finish this project.

Finally, I'd like to express my gratitude to the Almighty God for his guidance, strength, and protection in providing us with a healthy life.

ACKNOWLEDGEMENTS

First and foremost, I want to thank God the Almighty for showering me with blessings throughout my life, allowing me to successfully complete the project.

I would like to express my deep and sincere gratitude to my supervisor, PN.Emaliana Binti Kasmuri, for giving me the opportunity to do the project and providing invaluable guidance throughout this project. Her vision, sincerity, and motivation have profoundly influenced me. She has taught me the methodology to carry out the project and to present project work as clearly as possible. It was a great privilege and honour to work and study under her guidance.

Furthermore, I would like to express my gratitude to my evaluator, Dr Nor Hafeizah Binti Hassan, for providing me with critical recommendations and evaluating my work.

Lastly, I am extremely grateful to my parents for their prayers, care and sacrifices for educating and preparing me for my future. I am grateful to my friends and classmates for their patience, prayers, and ongoing support in completing this project.

ABSTRACT

The House Rental Application System is a web-based system to manage the process of renting a house systematically. The majority of landlords and tenants currently use a manual process. The manual process is complicated, time-consuming, and inconvenient, especially for landlords who manage multiple tenants at the same time. It is also hard for the landlord to keep track of the monthly rental payment for every tenant. With the development of this project, it may be possible to solve the current problems that landlords and tenants are experiencing. The purpose of this system is to develop a systematic web application to rent houses to keep all the data regarding the landlord, tenant, houses and payment. The business operation is computerized and systematic by using the House Rental application system. The scope of this project covers the administrator module, landlord module and tenant module. The significance of the project is to develop a web application to rent houses where it can simplify the entire process of renting and viewing, payment and minimize human error.

ABSTRAK

Sistem Permohonan Sewaan Rumah adalah sistem berasaskan web untuk menguruskan proses menyewa rumah secara sistematik.Majoriti tuan-tuan tanah dan penyewa kini menggunakan proses manual. Proses manual adalah rumit dan memakan masa terutamanya bagi tuan-tuan tanah yang menguruskan maklumat penyewa yang banyak pada masa yang sama. Ia juga sukar bagi tuan tanah untuk memantau bayaran sewa bulanan untuk setiap penyewa. Dengan pembangunan projek ini, ia dapat membantu menyelesaikan masalah semasa yang dialami oleh tuan-tuan tanah dan penyewa. Tujuan sistem ini adalah untuk membangunkan aplikasi web yang sistematik untuk menyewa rumah, untuk menyimpan semua data mengenai tuan tanah, penyewa, rumah dan pembayaran. Operasi perniagaan dikomputerkan dan sistematik dengan menggunakan sistem aplikasi Permohonan Sewaan Rumah. Skop projek ini meliputi modul pentadbir, modul tuan tanah dan modul penyewa. Kepentingan projek adalah untuk membangunkan aplikasi web untuk menyewa rumah di mana ia boleh memudahkan keseluruhan proses menyewa dan menguruskan pembayaran dan meminimumkan kesilapan manusia.

TABLE OF CONTENTS

PAGE

LIST OF TABLES

LIST OF FIGURES

PAGE

LIST OF ABBREVIATIONS

FYP	**-**	**Final Year Project**
ERD	**-**	**Entity Relationship Diagram**
HRAS	**-**	**House Rental Application System**
NRIC	**-**	**National Registration Identity Card Number**
DDL	**-**	**Data Definition Language**
SDLC	**-**	**Software Development Life Cycle**

CHAPTER 1: INTRODUCTION

1.1 Introduction

This project is regarding on the web application to rent houses. The purpose of the web application is to manage the process of renting a house systematically for the landlord and tenant. The web based house rental management is a system where the users must access to the web page. The system is required the users to login into the system after they had register their account. The system have 3 modules which is administrator, landlord and tenant. The system will allow administrator to manage the landlord information, tenant information and also managing the house advertising. The landlord will able to manage the rental house information, appointments, contract management and rental payment information. While for the tenant, the system will allow the tenant to search the house, view the condition of the house, booking appointment, continue the contract, and payment through online. This system will provide an easy access to the tenants to get the house depending on their preferences.

At the some moments, there is a limited number of web based house rental management system. Majority of the landlord and tenant are using manual process. The manual process is complicated, time consuming and inconvenient. Due to the time constraints, the tenant need to pay lot of money to the house agency to help them searching the houses. By developing the system, all the information regarding on the house, facilities, landlord details can be access quickly and efficiently.

With this application, it can reduces the searching time and also can provide a convenient platform for the tenant for searching the houses and also the landlord for ease the management. Thus, this system can overcome the problem that face by both tenants and landlords without need to follow a lengthy and hectic process anymore.

1.2 Problem Statement

Every individuals want his rental house to be the strategic location with the best facilities. The manual approach to search the rent houses through agency which have to follow a length and hectic process. It takes a long time to look for the suitable house at the desired location. It is quite difficult to handle with the agency or brokers to find the house as it may ask a lot of money from the tenant if they need more information about the houses, payment and also landlord details. People need to meet the agent or broker in person, checking the house details and also need to visit the location for several times. Due to the money, energy and time constraint, the tenant does not have much time to search the desired location and house.

Aside from that, when it comes to organizing the information on each specific property, landlord face a variety of challenges. Each property may contain a considerable amount of critical that the landlord must keep track of at all times. Each area property must identified and stored by the landlord for future reference, from the address to the services provided. It may be difficult for the landlord to keep track of the properties' information because everything is saved on paper, and it takes times to locate the records when they are needed.

Property management does not manage the manual handling of properties well. The traditional approach, with no appropriate storage for the important property management. These techniques employ a lot of paper to document property information, which is not environmentally friendly. This technique also may result in the loss of crucial information about tenants and properties if it is not adequately stored in secure storage and may result in human errors.

1.3 Objective

The objective of this project are as follows:

1. To design a systematic web application to rent house for keeping all data regarding the landlord, tenant, houses, and payment.
2. To develop a rental house management system that will ease the process of renting.
3. To test the accuracy that is consistent with the system.

1.4 Scope

The scope of the project is to manage house rental process for system administrator, landlord and tenant using web application. The features design for the application is according to the role of the users. The features are:

1.4.1 Module to Be Develop

1.4.1.1 Administrator Module

(a) *Authentication - Admin must login to the system to have the privilege to manage the system according to their roles.*

(b) *Change Password - Admin able to update their password.*

(c) *Manage House - Admin able to manage the advertisement such as update, add, and delete the advertisement details.*

(d) *Manage Landlord - Admin able to manage landlord such as update and delete landlord's information.*

(e) *Manage Tenant - Admin able to manage tenant such as update and delete tenant's information.*

1.4.1.2 Landlord Module

(a) *Authentication - Landlord must login to the system to have the privilege to manage the system.*

(b) *Change password - Landlord can update their password.*

(c) *Manage rent house - Landlord can add, update, and delete the rent house information.*

(d) *Manage contract - Landlord can update the tenant's contract if they want to pursue rent the house.*

(e) *Manage rental payment - Landlord can update the rental information and send notification to the tenants if there have due payment to the tenants.*

1.4.1.3 Tenant Module

(a) *Authentication - Tenant must login to have the privilege to access to the system.*

(b) *Change password - Tenant can update their password.*

(c) *Search rent houses - Tenant can search the houses according to their preferences and also view the condition, facilities and other utilities available in the house.*

(d) *Manage Rent House - Tenant can continue the renting the house once it about to end.*

(e) *Rental payment - Tenant can pay the monthly rental through online payment.*

1.5 Project Significant

The significant of the project is to create a web application to rent house that will provide service to the user, easy to use and provide the management features to store information. On the other hand, the system can provide better user experience where it can simplify the entire process of renting and viewing, payment and minimize the human error.

1.6 Expected Output

The project will be develop a web based application for rental house management system that will have a database for store all the information about the landlord, tenat, house, payment details, booking and others setup by administrator, tenant and landlord.

1.7 Conclusion

As a conclusion, a web application to rent house will be great help for the tenants, landlord and also property agency who need a proper house management system. This system will ease the landlord to manage the house, schedule appointment, upload the house information, finding the suitable tenants. The system can also ease the tenant to find the available house information, comparing the different houses and price without undergoes hectic process.

CHAPTER 2: LITERATURE REVIEW AND PROJECT METHODOLOGY

2.1 Introduction

The housing sector remains proactive in the face of transition , employing a modern approach that makes it easier to handle rental properties. The tenant and landlord are the centre of the web application. As a result, it allow the landlord to register their house in order to aid the tenant in searching for the ideal house. The web application to rent a house is suitable for the owner to manage the house details and rent details, while the tenant searches for a rental house and is able to gain information about available rooms, house rent, address, and other information.

Each process of a project's life cycle is represented by a set of steps and activities known as project methodologies. Any research project's design necessitates careful consideration of the research process and the planned data analysis. Within the section, we have attempted to provide some information about how to handle and present project during the course of consistent implementation process until it completed successfully.

2.2 Facts and Findings

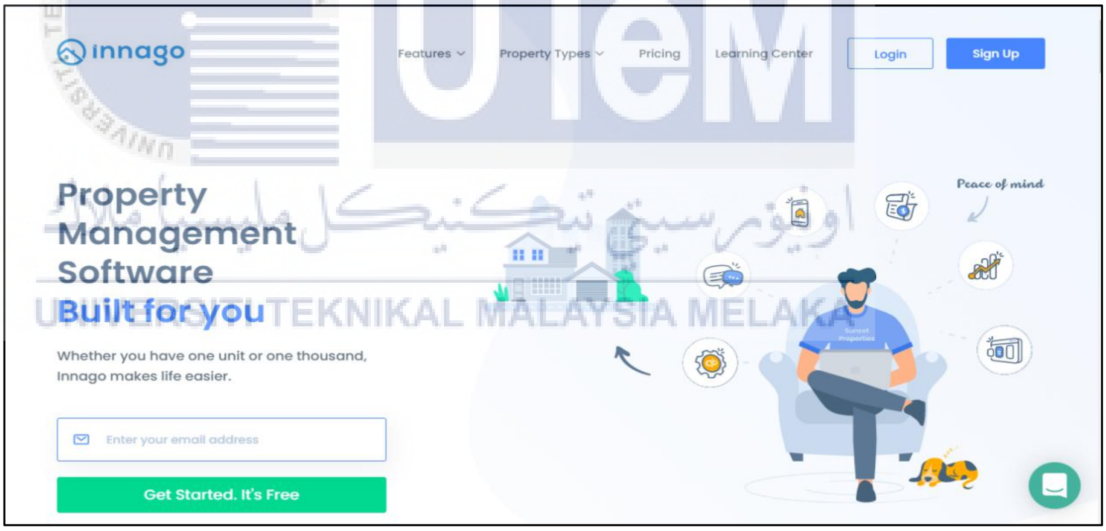
2.2.1 Domain

Industrial real estate is a related domain that is close to this project. One type of industrial real-estate that manages the renting process is the house rental application system. In industrial real estate, property managers or landlords are in charge of maintaining the houses and dealing with individual tenants. Industrial real estate simplifies the process of renting or owning a house for both landlords and tenants by searching for affordable house and also managing the houses through an online web application, eliminating the need for a complicated process. The house rental application system will enable landlords to manage their properties by screening potential tenants and signing and renewing contracts, and collecting the rent.

2.2.2 Existing System

This section will be focusing on the existing system and its advantage and disadvantage. This web-based property management system will allow the user to manage all the properties, rental process, rental payment and tenant management in the system. The figure below is few example of existing property management system.

a) Innago



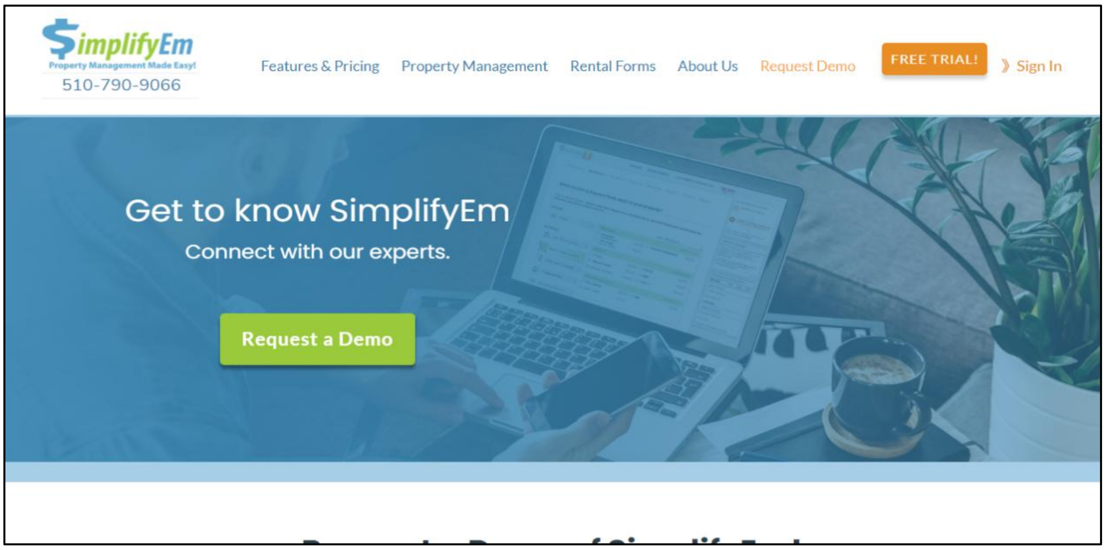
Property Management Software Built for you

Whether you have one unit or one thousand, Innago makes life easier.

Figure 2.1 Front interface of Innago System

This system is one of the similar system that are going to be develop. The Innago system is designed to handle the property management, make renting simple and accessible for the landlord. This system has similar features to Web-based House Rental Application System(HRAS), such as collect rent, screen tenants, list properties, manage tenants, and more.

b) SimplifyEm



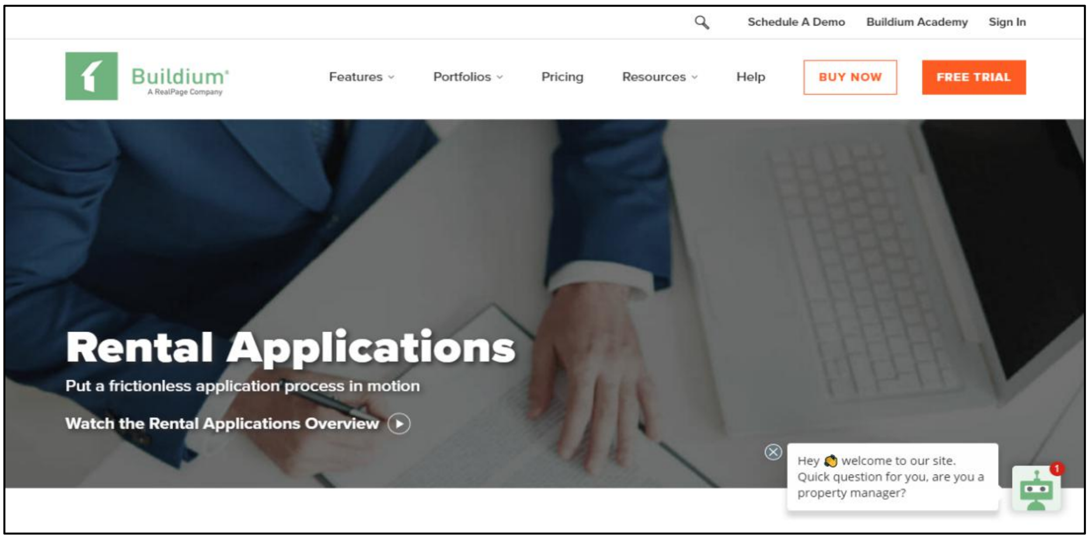
Get to know SimplifyEm

Connect with our experts.

Figure 2.2 Front Interface of SimplifyEm System

This system has offers almost similar function for the system that going to be developed. This system is also provide the both version which is free trial and free version for user to use. However, the system focuses on the internal use for the landlord which is collecting online rental payments, manage tenants information, find and screen prospective tenants.

c) Buildium



Rental Applications

Put a frictionless application process in motion

Hey 👋 welcome to our site. Quick question for you, are you a property manager?

Figure 2.3 Front Interface of Buildium System

The system has similar function for the system that is going to be developed. However, this system also focuses on internal use for the landlord to manage their properties. However, this system allow user to use for limited time for the free trial and the user need to paid every month to continue using their services.

Table 2.1 Comparison between the existing system

Function	Innago	Buildium	SimplifyEm
Login	Yes	Yes	Yes
Registration	Yes	Yes	Yes
Property management	Yes	Yes	Yes
Rent tracking	Yes	Yes	Yes
Online rental payment	Yes	Yes	Yes
Tenant management	Yes	No	Yes
Contact management	Yes	No	Yes
Generate detailed report	Yes	Yes	Yes

2.3 Project Methodology

Before developing the system, there are few steps need to be done in-order to keep the system development well organized and completed on the given time. This section describe system life cycle that related with the project. SDLC stands for Software Development Life Cycle and also referred as Application Development Life Cycle. SDLC is a software development life cycle that defines their own process and deliverable in every phases. Using SDLC is a systematic process to ensure the quality and correctness of the software built.

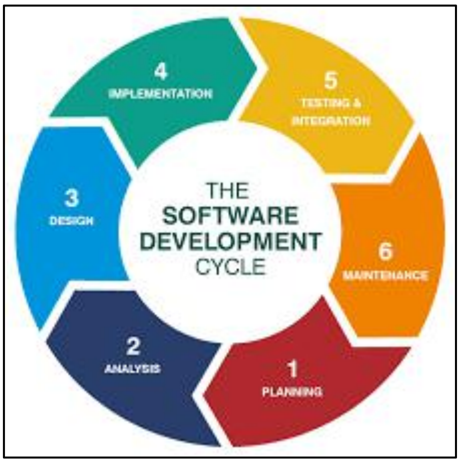


Figure 2.4 SDLC Methodology

i. Planing

The first phase involve in the SDLC life cycle is planing. In this phase, user will start choose the title of the project. The project chosen is the web application to rent house. The web application to rent house is a system that use by the landlord and tenant for ease of rental management and also searching for the rental house. During the phase, it will start identifying the stakeholder involve in the project, the scope of the project and also the objective that need to be achieve. The researcher also will start identify the list of hardware and software that will be using during the development of the project.

ii. Analysis

In this phase, the process of identifying the problem face on the manual process is begin. Once the problem had been identified, the user requirement need to be gather for collect the information regarding on the project. The process involves by identify the functional and non-functional requirement that involve in the project. This stage will give a clearer picture of the scope of entire project and the module that will be develop on the project.

iii. Design

In this phase, the design process involved. The design process involve two parts which is database design and system design. For the database design, it will be included the entity relation diagram(ERD) along with the data dictionary. For the system design, it involve designing the initial mock up interface for the project. The Balsamiq wire frame software had been use to design the interface of the project. This process helps to define the overall system's flow and give the visualization to the user on how the system look alike.

iv. Development

Once the design phase is complete, the next phase is development. Developer start build the entire project and writing the code using the chosen programming language. In the development phase, the system is divide into a few modules. For the development environment is using the visual studio which as IDE for the web development. Xampp is use during deployment of the system as it ease to use the services as it comes bundle with other packages such as Apache, MySQL, PHP,etc. The language that are using in this project is HTML, CSS, JavaScript, PHP, SQL queries and also Bootstrap for the front-end development. For the Back-end, the data in the project is store and synchronize from application to MySQL Database or vice versa.

v. Testing & Integration

Once the system is complete, and it is deployed in testing environment. The developer start testing the functionality of the entire system. This stage is to make sure that the system is working well according to the customer requirement. The developer fixes the bug and it will re-test the system. This process continues until the system is bug-free and run smoothly.

vi. Implementation

Once the testing phase is done and no bug in the system, it start with the final deployment process and provide to the end user for the demonstration. The system is tested by our supervisor and lectures. If there any issue during the deployment phase, a fix need to be done to maintain the access to system.

vii. Maintenance

After the success implementation of the system, start with maintenance phase where there are three activities occur which is fixing bug, upgrade and enhancement of the system. This phase continue to ensure that it meet with the requirements.

2.4 Project Requirements

The project requirements are the list of software and hardware needs that must be used to ensure the project success. The software and hardware are determine before the development process to ensure the smooth progress.

2.4.1 Software Requirement

Below are the list of software that been used to develop the system. The software that are using in the development of the system are:

2.4.1.1 Development Tools

Table 2.2 Development Tools

Software	**Description**
Microsoft Visual Studio	Integrated Development Environment Version: 1.58.2 Language: PHP, HTML, JS
MySQL	Version: 8.0.3

2.4.1.2 Operating System

Table 2.3 Operating System

Software	**Description**
Window 10	Microsoft Windows Operating System Edition: Windows 10 Home Single Language Version: 20Hz

2.4.1.3 Software Tools

Table 2.4 Software Tools

Software	**Description**
Draw.Io	Version: 14.4.3

WPS Office	Version: 11.2.0.10223
Balsamiq Wire frame	Version: 4.2.4

2.4.1.4 Web Server

Table 2.5 Web Server

Software	**Description**
Xampp	Version: 8.0.3

2.4.2 Hardware Requirement

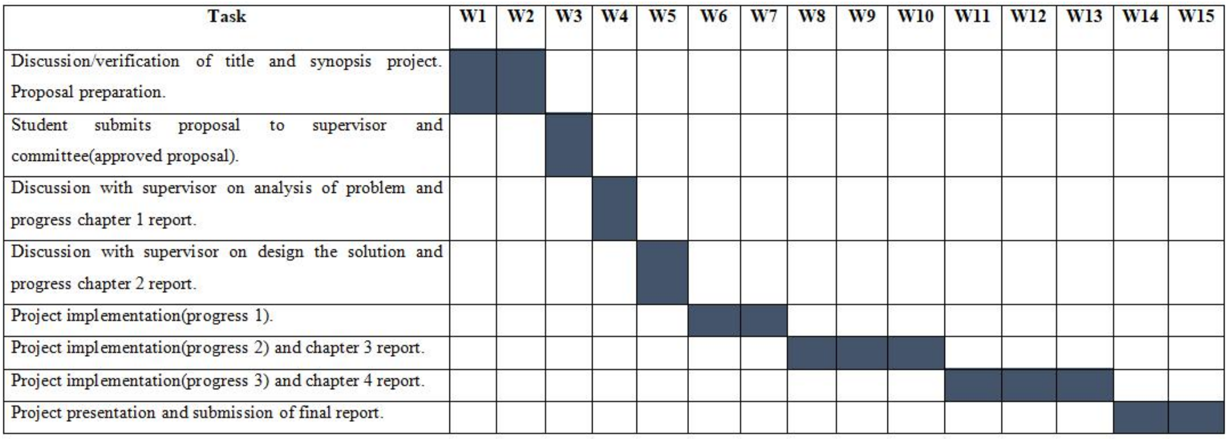
Below are the list of hardware that been used to develop the system. The hardware that are using in the development.

Table 2.6 Hardware Requirement

Software	**Description**
Laptop	Intel(R) Core(TM) i3-7020U CPU @ 2.30GHz 2.30 GHz

2.5 Project Schedule and Milestone

The management of this project uses a Gantt chart for project planning and milestones. Figure 2.1 below shows the Gantt chart for the whole project. The system's development took 14 weeks to complete.



Task	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14	W15
Discussion/verification of title and synopsis project. Proposal preparation.	■	■													
Student submits proposal to supervisor and committee(approved proposal).			■												
Discussion with supervisor on analysis of problem and progress chapter 1 report.				■											
Discussion with supervisor on design the solution and progress chapter 2 report.					■										
Project implementation(progress 1).						■	■								
Project implementation(progress 2) and chapter 3 report.								■	■	■					
Project implementation(progress 3) and chapter 4 report.											■	■	■		
Project presentation and submission of final report.														■	■

Figure 2.5 GanttChart of House Rental Application System

Table 2.7 FYP 1& FYP 2 Milestones

Final Year Project 1 (Semester 2 2020/2021)		
Target Week	**Start Date**	**Deliverable**
Week 1	15 March 2021	Proposal PSM
Week 2	22 March 2021	Proposal Submission
Week 3	29 March 2021	Chapter 1
Week 6	19 April 2021	Chapter 2 Project Progress
Week 7	26 April 2021	Chapter 3
Week 10	17 May 2021	Chapter 4 Project Progress
Week 11	24 May 2021	Project Demonstration
Week 14	14 June 2021	FYP 1 Draft Report
Week 15	21 June 2021	Final Presentation Submission PSM 1 Report
Final Year Project 2 (Semester 3 2020/2021)		
Target Week	**Start Date**	**Deliverable**
Week 1	19 July 2021	Chapter 4
Week 2	26 July 2021	Progress Presentation 1
Week 3	2 August 2021	Chapter 5
Week 4	9 August 2021	Progress Presentation 2
Week 5	16 August 2021	Chapter 6
Week 6	23 August 2021	FYP 2 Draft Report
Week 7	30 August 2021	Presentation
Week 8	6 August 2021	FYP 2 Logbooks
Week 9	13 September 2021	Final FYP Report

2.6 Conclusion

In conclusion, this chapter focuses on the domain and project methodology that are used in the project. For the project methodology, the agile method has been chosen to be implemented during the project development process. In addition, the chapter discusses the project requirements. It is divided into two sections for the project requirements, which are the software and hardware requirements that will be used throughout the project to develop the house rental application system.

CHAPTER 3: ANALYSIS

3.1 Introduction

This chapter provides an initial overview of the development plan and aims to help the development process run smoothly and meet the user's requirements. The analysis stage encompasses the assessment of the system development planning carried out ahead of time. At this stage, the suitability of the chosen method and the sequence of the work is the aim of ensuring the system can be effectively setup. Functional requirements, non-functional requirements, and other requirements were all analyzed during the requirement analysis process.The requirements involved in the development process will be described to assist users in analyzing and understanding the required process.The requirement should be clearly identified so that it meets all of the system development criteria and does not cause any undesirable problems.

3.2 Problem Analysis

The current rental housing business process is still carried out by manual. There are numerous papers that must be organized, as well as numerous papers used for data recording. The current system has two types of users: landlords and tenants.The current approach to handling the rental process does not have a proper system to store all the user information because some of them use a manual way to keep all the records.

The current system also stores payment records, contracts, and tenant information manually, which can lead to serious mistakes if the landlord is not careful when recording the information.Next, the landlord will need to organize a lot of paperwork. It is because the current system does not have a system to manage all that paperwork, including the contract, the tenant’s information, the rental house and others. Because of the number of tasks that must be managed and organized, the landlord's workload will increase.The landlord must also thoroughly review all of the documents provided to ensure that no errors occur during the property management task.

3.3 Requirement Analysis

This section will focus on context diagram where the flow of the system is shown with the data flow diagram. The data flow diagram will have level 0 and 1 of the Web-Based House Rental Application System.

3.3.1 Data Requirement

This section will show how the data is stored in the database. MySQL is chosen as the database to store all the information for the Web-Based House Rental Application System.

3.3.1.1 Data Dictionary

Data dictionary is develop to provide detailed information about the information that store in the database such as the entity, variable, data type and description. It is also use to show the structure of database and relationship between each entity. The table below consists of the data dictionary table that will be implement on the system.

a) Table User

Table 3.1 Data Dictionary for User

User				
Field Name	**Description**	**DataType/Length**	**Constraint**	**Remarks**
userID	A unique number use to identify each user	Integer (11)	Primary Key	
fullName	Full name of the user	Varchar(255)		
username	Surname of the user	Varchar(50)		

email	Electronic mail address	Varchar(100)		
phoneNo	User contact number	Varchar(15)		
password	Hashed Password	Varchar(50)		
role	User role	Varchar(10)		
profileImage	User profile picture	Varchar(100)		

b) Table Tenant

Table 3.2 Data Dictionary for Tenant

Tenant				
Field Name	**Description**	**DataType/Length**	**Constraint**	**Remarks**
tenantID	A unique number use to identify each tenant	Integer (11)	Primary Key	
fullName	Tenant Name	Varchar (50)		
emil	Electronic mail address	Varchar (50)		
IdentificationNo	National registration identification number	Varchar (20)		
occupation	Tenant occupation	Varchar (20)		
phoneNo	Tenant contact number	Varchar (10)		
address	Tenant address	Varchar (30)		
userID	A unique number to identify each user	Integer (11)	Foreign Key	

c) Table Landlord

Table 3.3 Data Dictionary for Landlord

Landlord				
Field Name	**Description**	**DataType/Length**	**Constraint**	**Remarks**
landlordID	A unique number use to identify each landlord	Integer (11)	Primary Key	
name	Landlord name	Varchar (30)		
username	Landlord username	Varchar (30)		
identityNo	National registration	Integer		

	identification number			
email	Electronic mail address	Varchar (30)		
contactNo	Landlord contact numbet	Varchar (30)		
address	Landlord address	Varchar (50)		
userID	A unique number use to identify each landlord	Integer (11)	Foreign Key	

d) Table Notifications

Table 3.4 Data Dictionary for Notifications

notifications				
Field Name	**Description**	**DataType/Length**	**Constraint**	**Remarks**
id	A unique number use to identify each notification	Integer (11)	Primary Key	
senderName	Name of the sender	Varchar(100)		
senderEmail	Email of the sender	Varchar(100)		
senderNumber	Contact Number of the sender	Varchar(15)		
message	Message that sender's send	Text		
houseID	A unique number use to identify each house	Integer(11)	Foreign Key	
status	Status of the message	Integer(11)		
cr_date	Date of the message's send	Timestamp		

e) Table House

Table 3.5 Data Dictionary for House

Houses				
Field Name	**Description**	**DataType/Length**	**Constraint**	**Remarks**
houseID	A unique number use to identify each house	Integer (11)	Primary Key	
houseName	Name of house	Varchar (20)		

category	Category of house	Varchar (20)		
address1	First line address	Varchar (50)		
address2	Second line address	Varchar (50)		
postcode	Postcode	Integer(11)		
district	District	Varchar (50)		
state	State	Varchar (20)		
size	Size area of house			
noRoom	Number of room in the house	Integer(11)		
noToilet	Number of toilet in the house	Integer(11)		
floorType	Type of floor	Varchar (30)		
livingRoom	Living room of the house	Varchar (20)		
airCond	Number of air conditioner in the house	Integer (11)		
kitchen	Kitchen of the house	Varchar (20)		
typeKitchen	Tyoe of kitchen house	Varchar (20)		
wifi	Wifi availability	Varchar (20)		
furniture	Furniture	Varchar (20)		
gate	Type of gate	Varchar (20)		
cctv	Availability of cctv	Boolean		
gateNguarded	Gate and guarded availability	Boolean		
house_image1	The Front view of house	Varchar (30)		
house_image2	Second house image	Varchar (30)		
house_image3	Third house image	Varchar (30)		
house_image4	Forth house image	Varchar (30)		
monthlyPaid	Monthly house rental	Decimal		
negotiable	Negotiable	Boolean		
deposit	House deposit	Decimal		
description	Description	Varchar (50)		
contract	House's contract	Varchar (30)		
changeSubject	Subject to change	Boolean		
userID	Aunique number yo identify each user	Integer(11)	Foreign Key	
houseStatus	Status of the house	Varchar (30)		

f) Table House Category

Table 3.6 Data Dictionary for House Category

houseCategory				
Field Name	**Description**	**DataType/Length**	**Constraint**	**Remarks**
id	A unique number use to identify each house information	Integer (11)	Primary Key	
categoryName	Category name of the house	Varchar (20)		

g) Table Contact Information

Table 3.7 Data Dictionary for Contact Information

Contact				
Field Name	**Description**	**DataType/Length**	**Constraint**	**Remarks**
contactID	A unique number use to identify each contact information	Integer (11)	Primary Key	
contactName	Contact name	Varchar (50)		
contactAddress	Contact address	Varchar (100)		
contactNo	Contact mobile phone numbet	Varchar (12)		
relationship	nature of the relationship with the tenant	Varchar (10)		

h) Table Payment

Table 3.8 Data Dictionary for Payment

Payment				
Field Name	**Description**	**DataType/Length**	**Constraint**	**Remarks**
paymentID	A unique number use to identify each payment	Integer (11)	Primary Key	
tenantID	Aunique number use to identify each tenant	Integer (11)	Foreign Key	
paymentDate	Date for rental payment paid	DATE		

paymentAmount	Total amount paid for the rental payment	Decimal		
status	Status of the payment	Varchar (20)		
paymentType	Method of payment pay	Varchar (20)		

3.3.2 Functional Requirement

This section define and describe the functional requirement in HRAS. Functional requirements are statement which shows a service that a system should have. It will provide a high-level statement to help provide the detailed requirement of specification of the system.

Table 3.9 Functional Requirement of HRAS

FR_No	Requirement	Description
HRAS 1_1	Login and Logout	The system shall allow user to login and logout into the system by validate through email and password.
HRAS 1_2	User Account Registration	The system shall allow the new user to register their account and allow the user to sign up based on their roles.
HRAS 2_1	Landlord	The system shall allow the admin to create, update and delete the landlord.
HRAS 3_1	Tenant	The system shall allow the admin to create, update and delete the tenant.
HRAS 4_1	Contract	The system shall allow the admin to create, update and delete the tenant's contract.
HRAS 5_1	House	The system shall allow the landlord and admin to create, update and delete the house information.

HRAS 6_1	Rental Payment	The system shall allow the tenant to make payment through the system.

3.3.2.1 Use Case Diagram

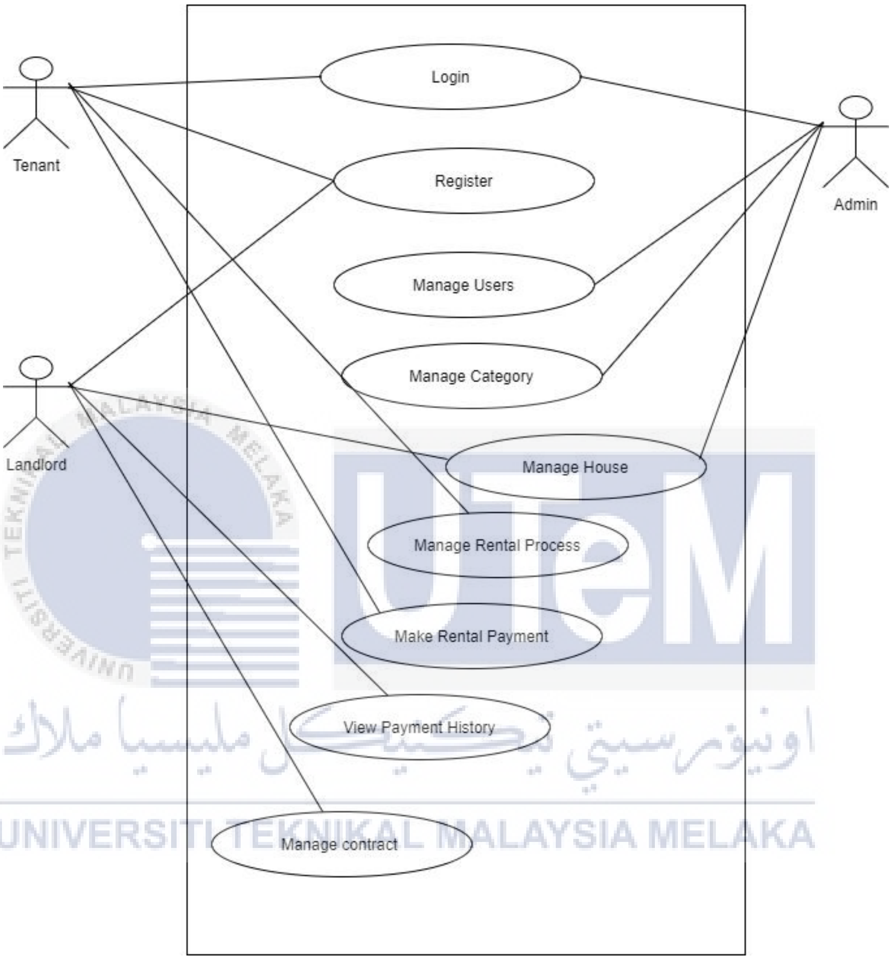


Figure 3.1 Use Case Diagram for HRAS

The Figure 3.8 above shows the use case diagram for House Rental Application System(HRAS). The are three user involve in this use case which is administrator, tenant, and landlord.

3.3.3 Non Functional Requirement

This section describe the non-functional requirement on the system. The non-functional requirement can be defined as the requirement specifying criteria that can be used to judge the operation of the system.

3.3.3.1 Data Integrity

The data store in the system should accurate and consistence.

3.3.3.2 Security

The system should allow only the authorized users to access the system. The system validates the user based on their email and password. The password should saved in database encrypted.

3.3.3.3 Performance

The system should be response within 3 seconds.

3.4 Conclusion

In this chapter discuss about the analysis of the current system and proposed system. In the proposed system, there are few features that have been added into the system. The use case diagram would be the overview of the process involve in the system to the user. Lastly, the requirement specification and analysis provides a details description and understanding of the process involve in the system.

CHAPTER 4: DESIGN

4.1 Introduction

The design phase specification will be briefly described in this chapter. The process of defining the elements of a system, such as the architecture, modules, and components, as well as the many interfaces between those components in the system that flows through it, is known as system design. It is intended to meet specific needs and requirements of a business or organization by engineering a coherent and well-functioning system.

The conceptual system design phase is the first stage of the system design methodology. The entity relationship model is a method of representing the logical relationship of entities that have been created a database in graphical form.

The logical database design phase is the second stage of the system design methodology. A data dictionary is a document that is used to control system access and manipulation. At the same time, the local logical data's integrity constraints are defined and reviewed model in collaboration with a user.

The physical database, which produces a description of the system's implementation, is the final phase of the system design methodology. It describes the file, organizations, base relations, and index design, all of which contribute to efficient data access, as well as any associated integrity constraints and security measure.

4.2 High-Level Design

The high-level design focuses on the design of the system. It also covers the system architecture with an interface and its behaviour in high-level design. In this section, specifies how the system responds to user input and attract user interaction. This design can also be used as a guide for users to understand how the components interact.

4.2.1 System Architecture

The system architecture on figure 4.1 define the structure of House Rental Application System. The process renting is the fore front process of the House Rental Application System. It synced up with the database as it will update the available house in the system. To complete the process of rental house booking, potential tenant enters all the tenant details in the tenant management module. It will notifies the landlord management module so that landlord can prepare the document needed to send to the tenant for the confirmation of their booking.

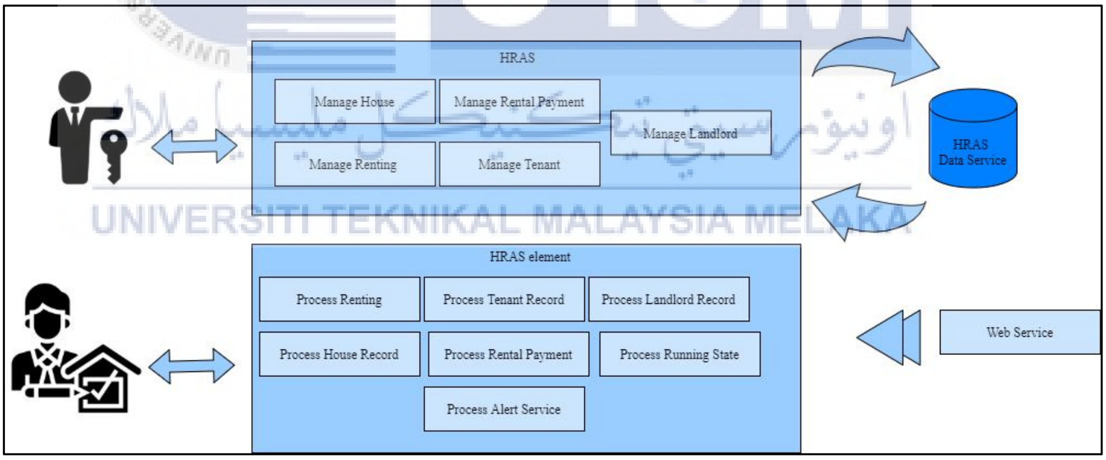
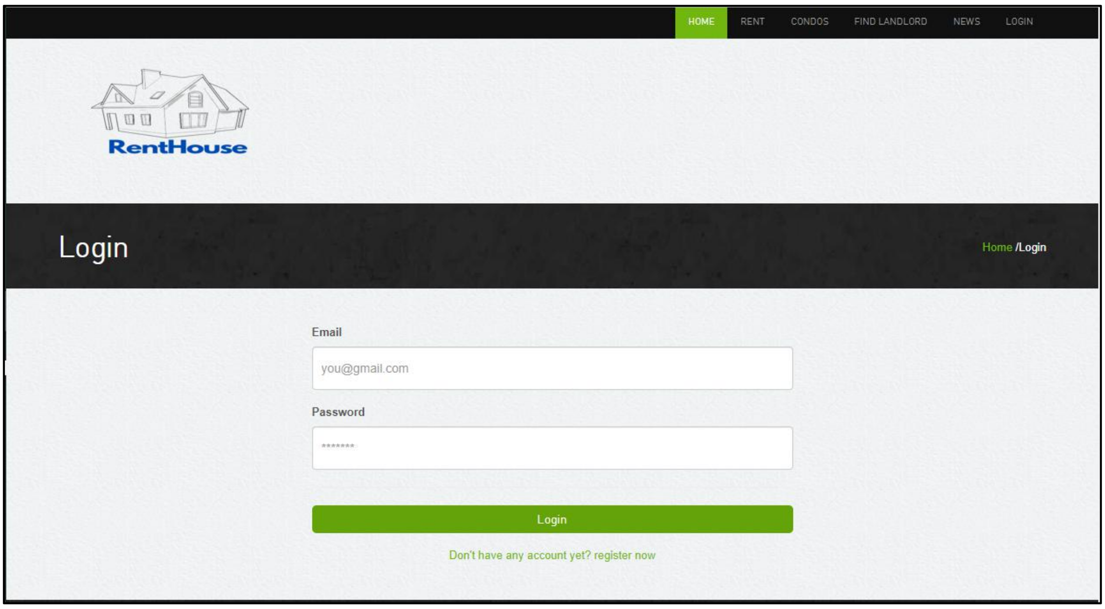


Figure 4.1 System architecture of web-based house rental system

4.2.2 User Interface Design



Email

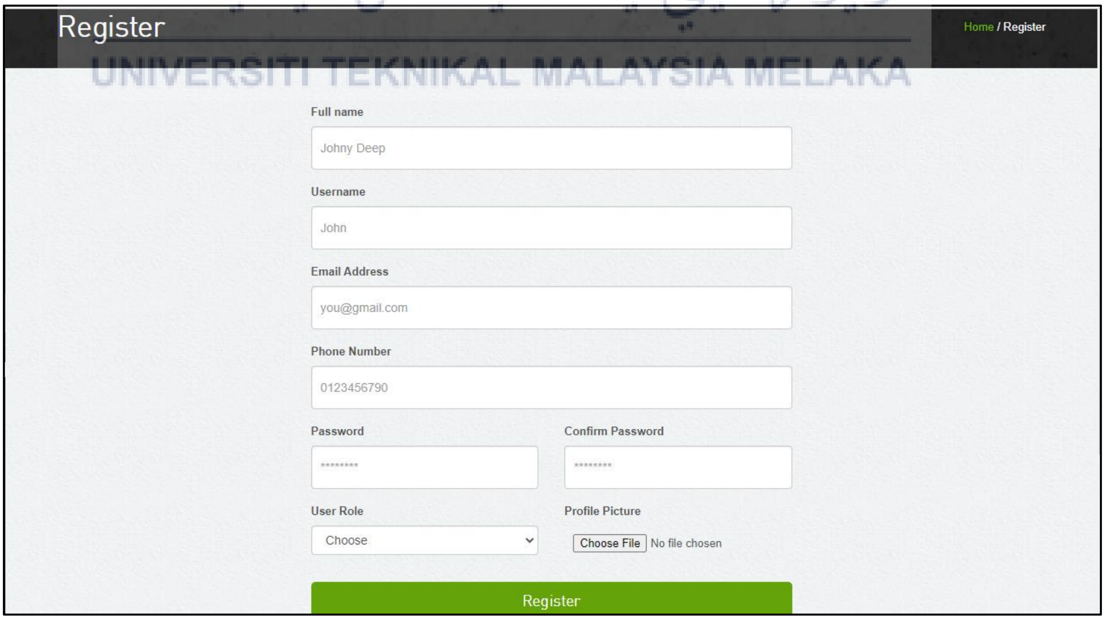
you@gmail.com

Password

****

Figure 4.2 Login page

Based on figure 4.2, the user must provide their email address and password, which are already stored in the system. The system will authenticate the user and direct them to their page depending on their roles based on the email and password entered.



Full name

Johny Deep

Username

John

Email Address

you@gmail.com

Phone Number

0123456790

Password

****

Confirm Password

****

User Role

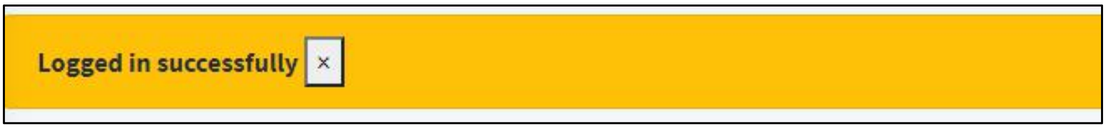
Choose

Profile Picture

Choose File No file chosen

Figure 4.3 Register page

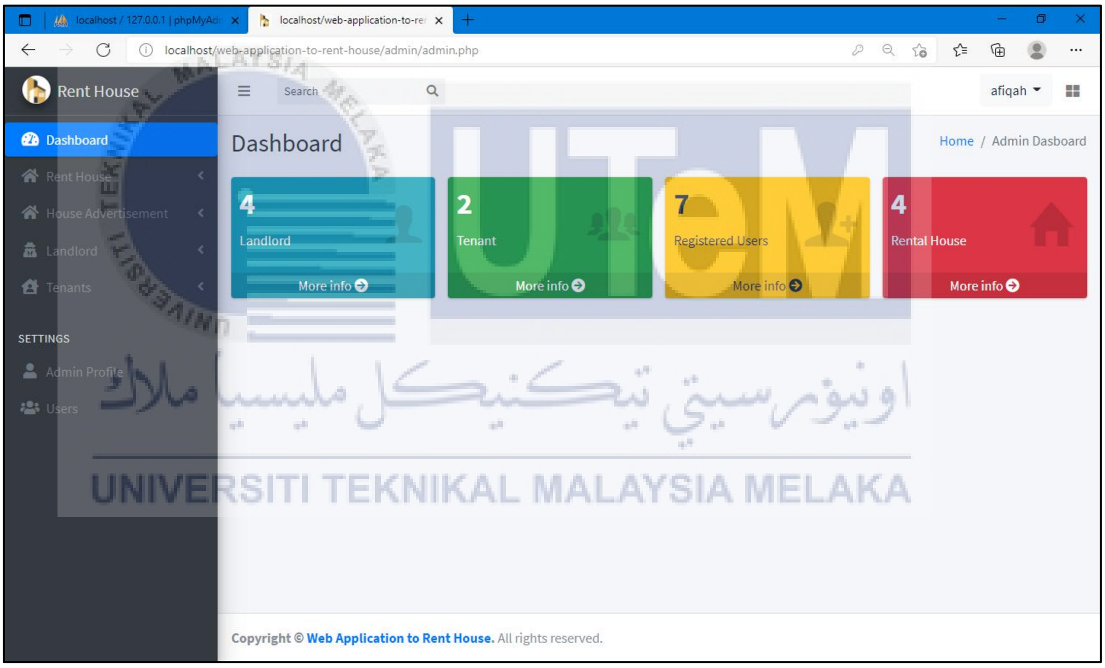
Users must register before logging into the system, as shown in Figure 4.3. In the database, all of the data entered will be saved, and the passwords will be encrypted. After completing the registration process, the user will log in using the email and password they registered.



Logged in successfully ×

Figure 4.4 Success Alert Message

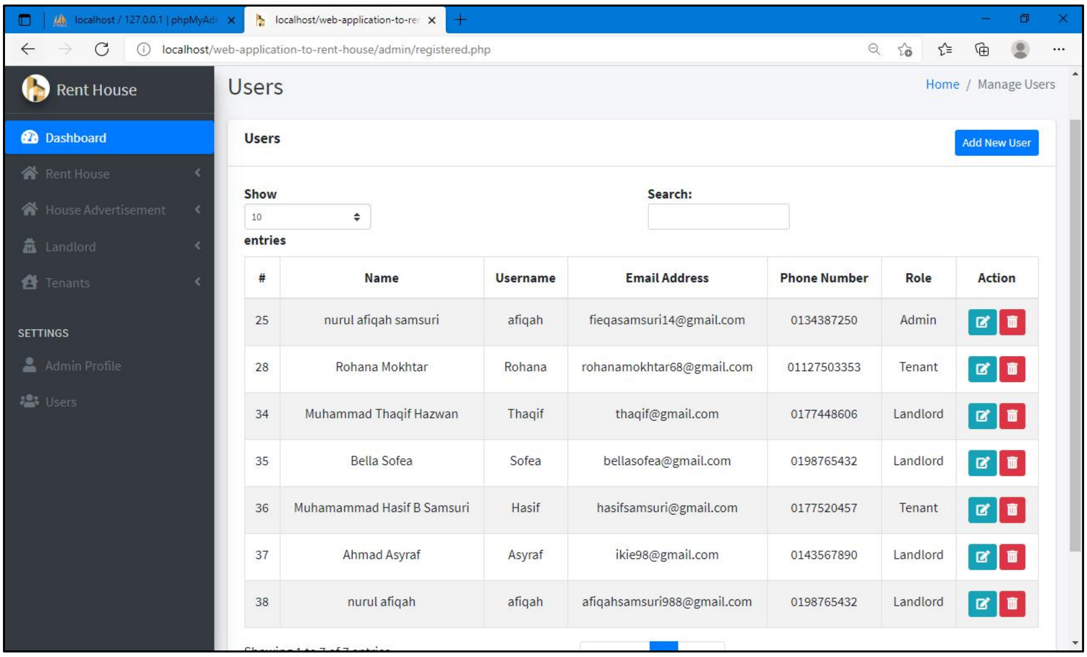
Based on figure 4.4, once the user has been validated, the system will display a successful alert to acknowledge the user that their login was successful.



Copyright © **Web Application to Rent House.** All rights reserved.

Figure 4.5 Home page for Admin

Figure 4.5 shows how the user is directed to the dashboard once they have successfully entered into the system. The user will be able to view the management overview and information via the dashboard.

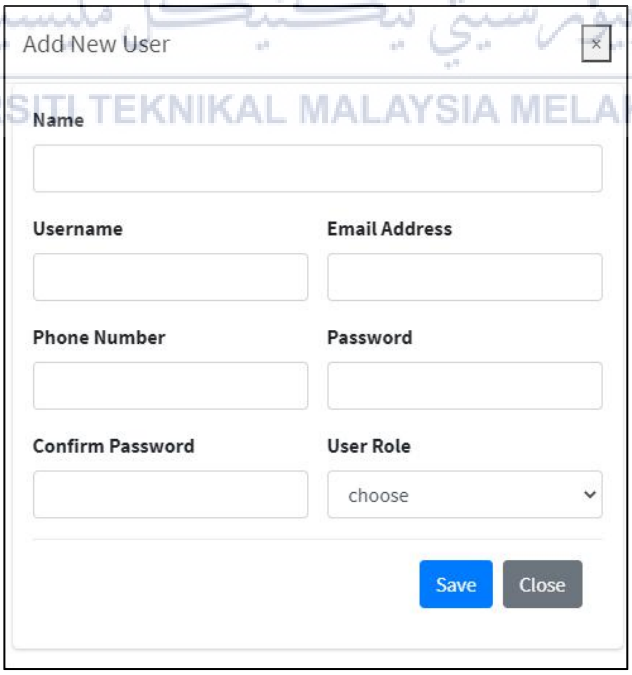


Users

#	Name	Username	Email Address	Phone Number	Role	Action
25	nurul afiqah samsuri	afiqah	fieqasamsuri14@gmail.com	0134387250	Admin	
28	Rohana Mokhtar	Rohana	rohanamokhtar68@gmail.com	01127503353	Tenant	
34	Muhammad Thaqif Hazwan	Thaqif	thaqif@gmail.com	0177448606	Landlord	
35	Bella Sofea	Sofea	bellasofea@gmail.com	0198765432	Landlord	
36	Muhamammad Hasif B Samsuri	Hasif	hasifsamsuri@gmail.com	0177520457	Tenant	
37	Ahmad Asyraf	Asyraf	ikie98@gmail.com	0143567890	Landlord	
38	nurul afiqah	afiqah	afiqahsamsuri988@gmail.com	0198765432	Landlord	

Figure 4.6 Manage Users Page for Admin

The admin will be able to see a list of users who have been registered in the system when they click on users settings. From here, they will be able to update the user’s information. If the user is no longer active on the system, the admin can delete the user with a delete button and the system will delete all the user’s information.



Add New User

Name

Username

Email Address

Phone Number

Password

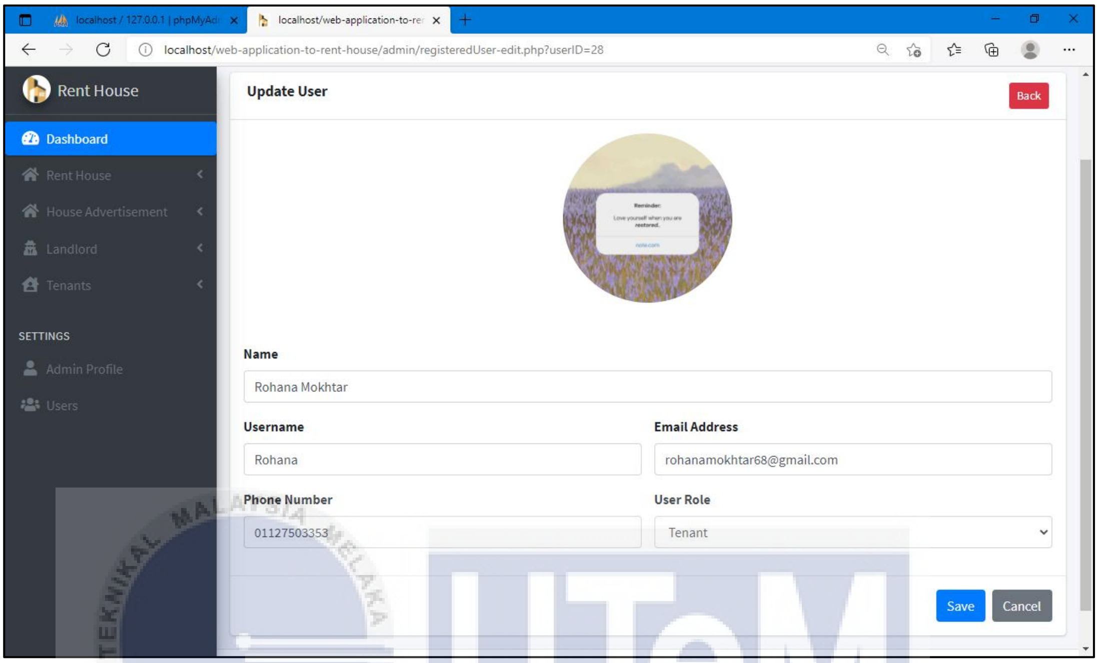
Confirm Password

User Role

choose

Figure 4.7 Add New Users Page

The Admin can add new users into the system. The default user's password will be given and the user can change the default password after logging into the system.


localhost / 127.0.0.1 | phpMyAd
localhost/web-application-to-re
localhost/web-application-to-rent-house/admin/registeredUser-edit.php?userID=28
Rent House
Dashboard
Rent House
House Advertisement
Landlord
Tenants
SETTINGS
Admin Profile
Users
Back
Save
Cancel

Update User

Name: Rohana Mokhtar

Username: Rohana

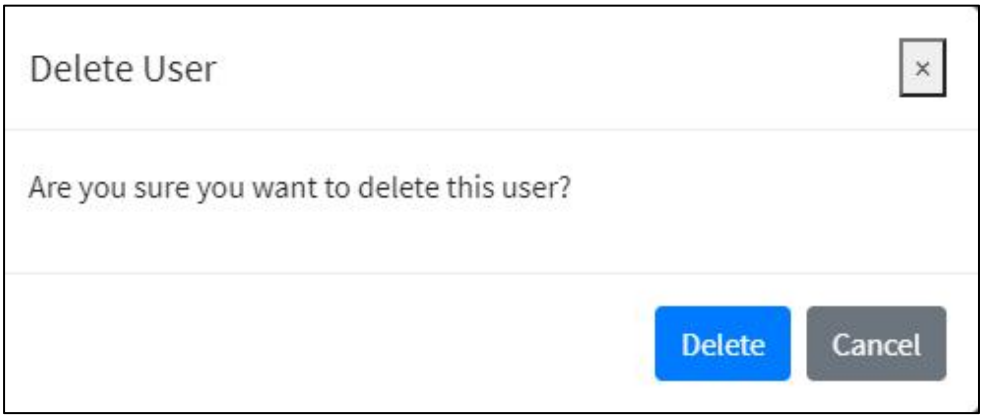
Email Address: rohanamokhtar68@gmail.com

Phone Number: 01127503353

User Role: Tenant

Figure 4.8 Update User Page

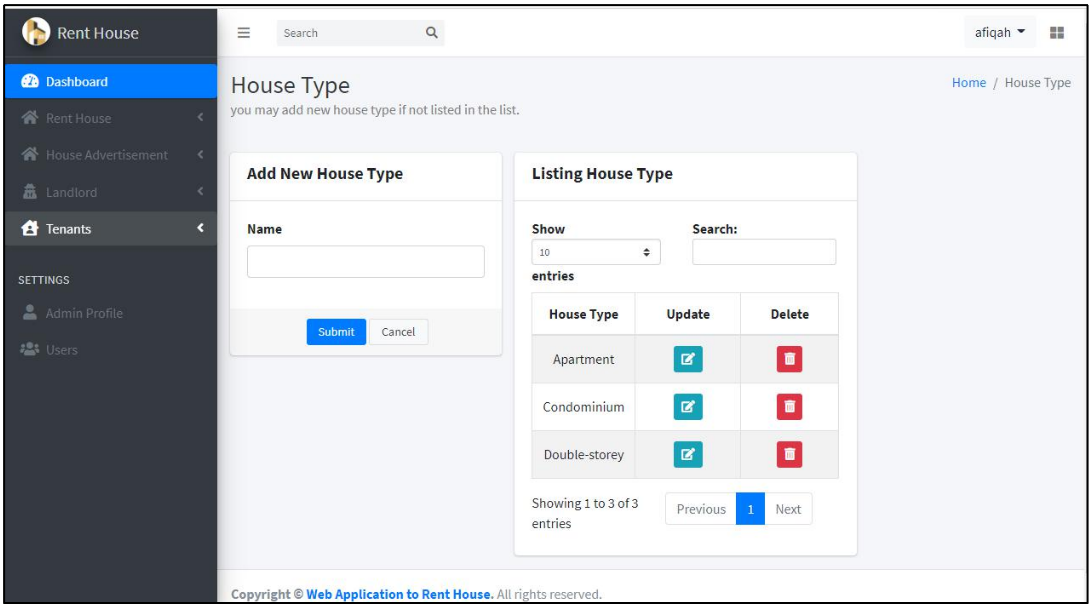
Based on figure 4.8 above, the user needs to fill in the blank fields that are given by the system after the first login into the system. Then, the system will update the user's information on the system.



Delete User

Are you sure you want to delete this user?

Figure 4.9 Delete user page



House Type

you may add new house type if not listed in the list.

Add New House Type

Name

Listing House Type

Show 10 entries

Search:

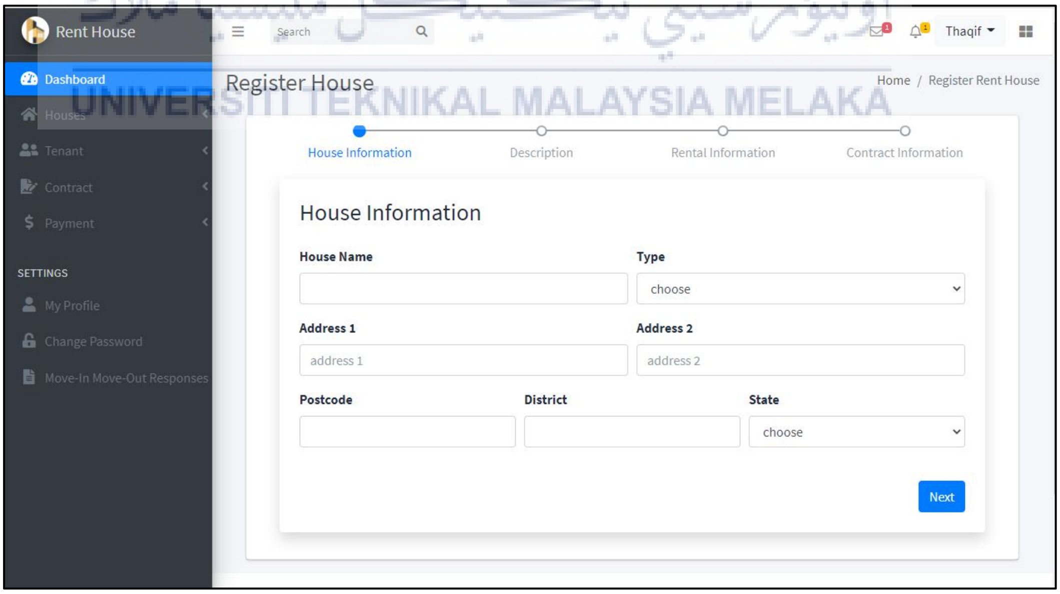
House Type	Update	Delete
Apartment		
Condominium		
Double-storey		

Showing 1 to 3 of 3 entries

Copyright © Web Application to Rent House. All rights reserved.

Figure 4.10 Manage House Category Page

Based on figure 4.10 above, the admin can add the new house category to the system. The house ID will be given automatically by the system. The admin can make changes or update the house types. If the house type is no longer used on the system, the admin can delete the house type by clicking the delete button in the settings.



Register House

House Information

House Name

Type: choose

Address 1: address 1

Address 2: address 2

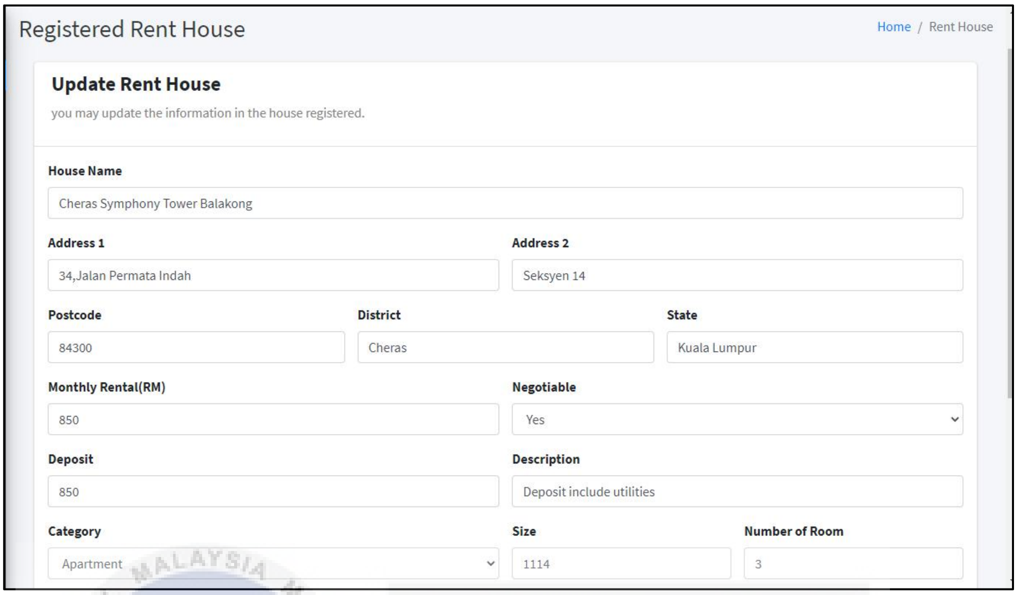
Postcode

District

State: choose

Figure 4.11 Add House Information

Landlords can register their residences to be rented on the system by filling in all of the fields provided by the system, as shown in figure 4.11. Once the landlord fills out the form, the system will advertise the property using the information provided to assist the landlord in finding a tenant.



Registered Rent House

Update Rent House

you may update the information in the house registered.

House Name: Cheras Symphony Tower Balakong

Address 1: 34,Jalan Permata Indah

Address 2: Seksyen 14

Postcode: 84300

District: Cheras

State: Kuala Lumpur

Monthly Rental(RM): 850

Negotiable: Yes

Deposit: 850

Description: Deposit include utilities

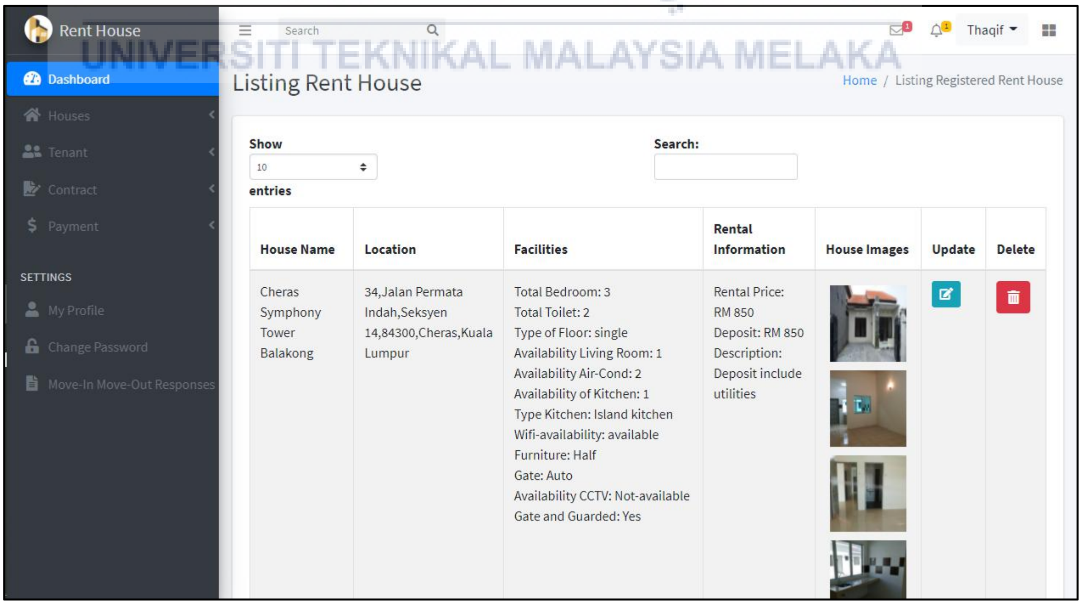
Category: Apartment

Size: 1114

Number of Room: 3

Figure 4.12 Update House Information

If there are any modifications to the house, the landlord can update the information by clicking the update button on the list of houses. The information on the system will be automatically updated by the system.



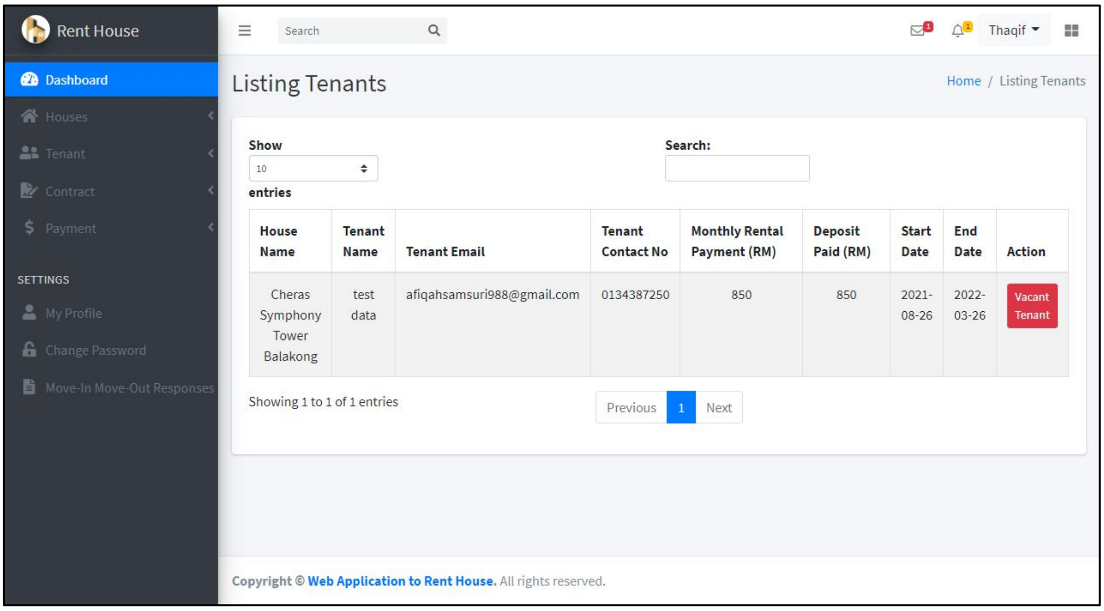
Listing Rent House

Show 10 entries

Search:

House Name	Location	Facilities	Rental Information	House Images	Update	Delete
Cheras Symphony Tower Balakong	34,Jalan Permata Indah,Seksyen 14,84300,Cheras,Kuala Lumpur	Total Bedroom: 3 Total Toilet: 2 Type of Floor: single Availability Living Room: 1 Availability Air-Cond: 2 Availability of Kitchen: 1 Type Kitchen: Island kitchen Wifi-availability: available Furniture: Half Gate: Auto Availability CCTV: Not-available Gate and Guarded: Yes	Rental Price: RM 850 Deposit: RM 850 Description: Deposit include utilities			

Figure 4.13 List of House



Listing Tenants

Show 10 entries

Search:

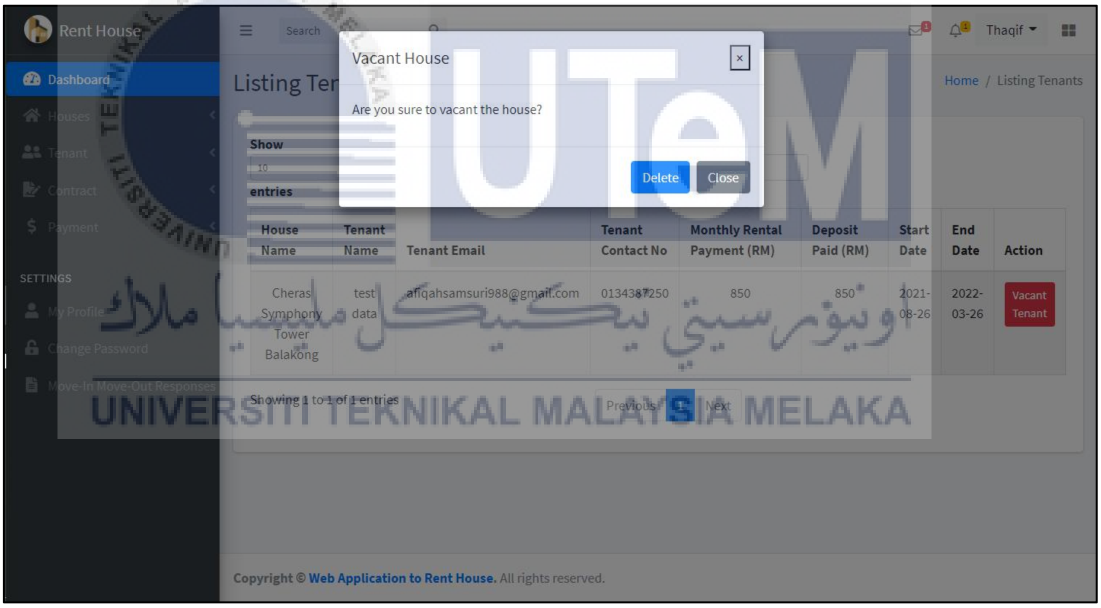
House Name	Tenant Name	Tenant Email	Tenant Contact No	Monthly Rental Payment (RM)	Deposit Paid (RM)	Start Date	End Date	Action
Cheras Symphony Tower Balakong	test data	afiqahsamsuri988@gmail.com	0134387250	850	850	2021-08-26	2022-03-26	Vacant Tenant

Showing 1 to 1 of 1 entries

Previous 1 Next

Copyright © Web Application to Rent House. All rights reserved.

Figure 4.14 Listing Tenant



Vacant House

Are you sure to vacant the house?

Listing Tenants

Show 10 entries

House Name	Tenant Name	Tenant Email	Tenant Contact No	Monthly Rental Payment (RM)	Deposit Paid (RM)	Start Date	End Date	Action
Cheras Symphony Tower Balakong	test data	afiqahsamsuri988@gmail.com	0134387250	850	850	2021-08-26	2022-03-26	Vacant Tenant

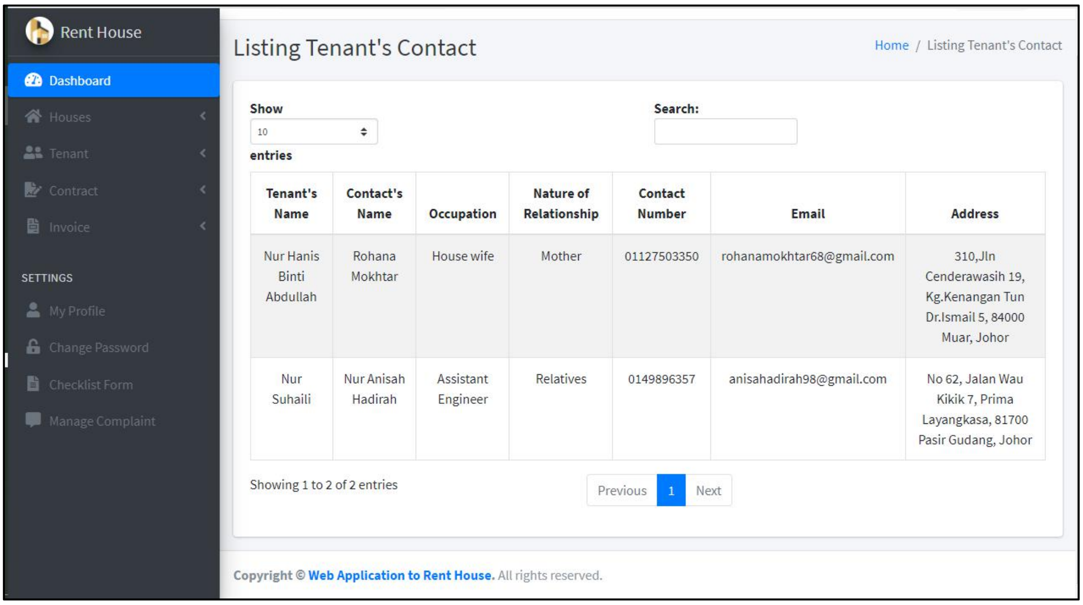
Showing 1 to 1 of 1 entries

Previous 1 Next

Copyright © Web Application to Rent House. All rights reserved.

Figure 4.15 Vacant Tenant Page

The landlord can see the list of tenants who rent their property in figure 4.15. The landlord can eliminate the tenant's information by selecting the vacant tenant option if the tenant's contract has expired or the tenant no longer rents the house. The tenant's rental and contract information will be deleted from the system, and the house status will be updated.



Listing Tenant's Contact

Show 10 entries

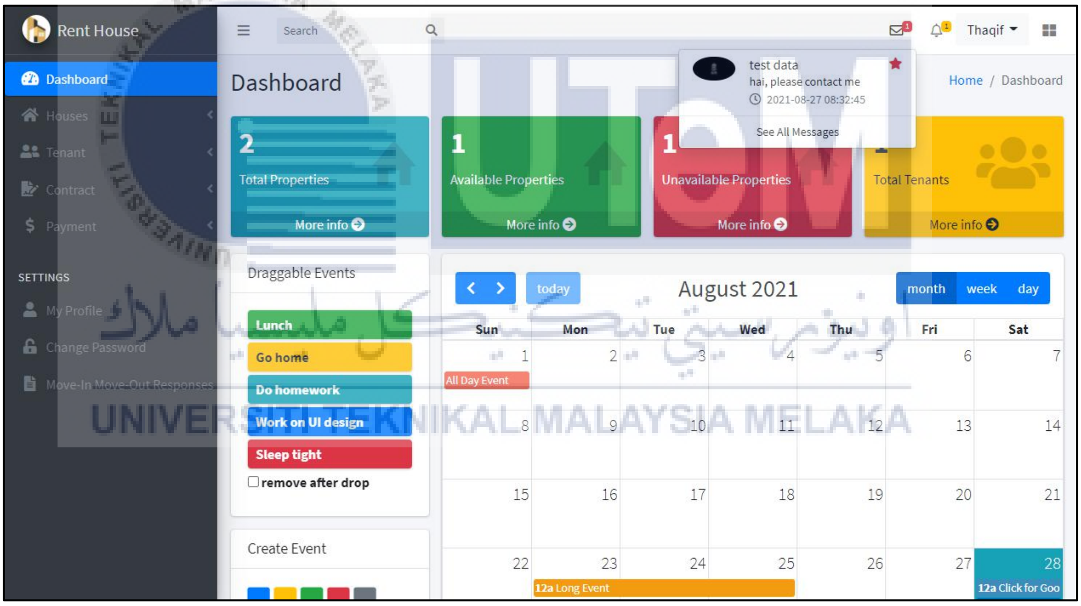
Search:

Tenant's Name	Contact's Name	Occupation	Nature of Relationship	Contact Number	Email	Address
Nur Hanis Binti Abdullah	Rohana Mokhtar	House wife	Mother	01127503350	rohanamokhtar68@gmail.com	310,Jln Cenderawasih 19, Kg.Kenangan Tun Dr.Ismail 5, 84000 Muar, Johor
Nur Suhaili	Nur Anisah Hadirah	Assistant Engineer	Relatives	0149896357	anisahadirah98@gmail.com	No 62, Jalan Wau Kikik 7, Prima Layangkasa, 81700 Pasir Gudang, Johor

Showing 1 to 2 of 2 entries

Copyright © **Web Application to Rent House.** All rights reserved.

Figure 4.16 List of Tenant's Contact Member

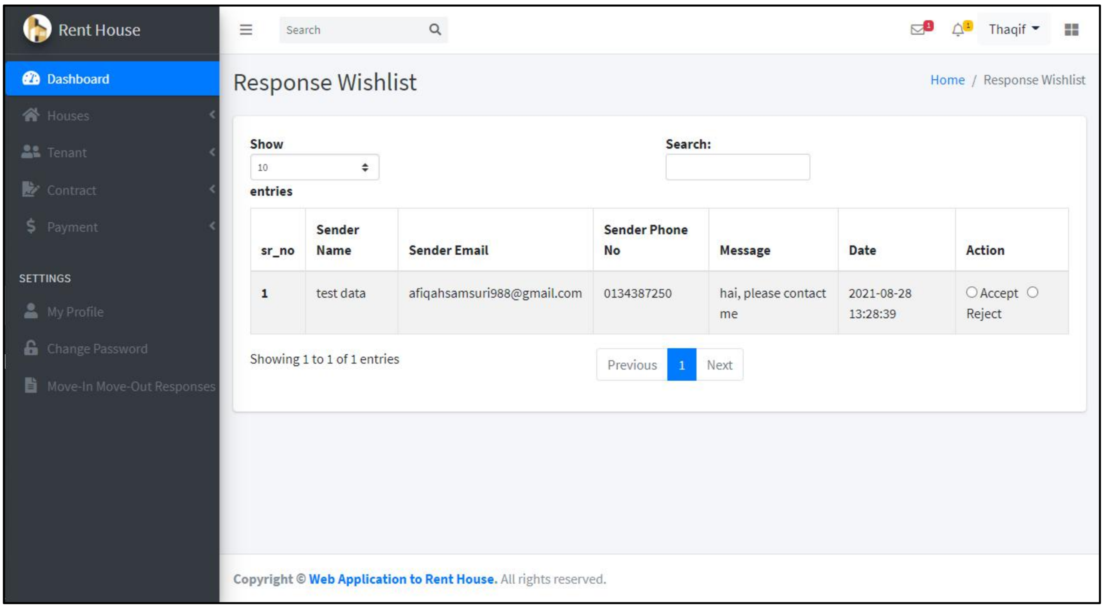


Dashboard

test data
hai, please contact me
2021-08-27 08:32:45

Figure 4.17 List of Notifications

If the system identified the credential as a landlord, it would direct the landlord into the system and display the landlord dashboard, as shown in figure 4.17. The landlord can see the overall number of properties, available properties, unavailable properties, and the number of tenants via the dashboard. On the top bar of the page, the landlord will be able to see the list of notifications and mark the important events on the calendar.



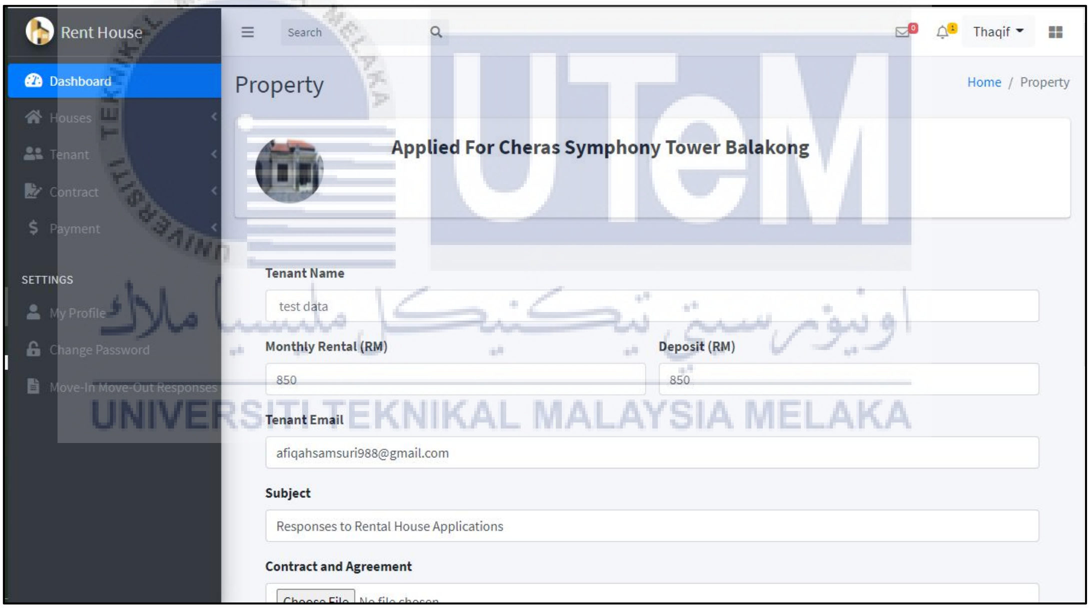
Response Wishlist

sr_no	Sender Name	Sender Email	Sender Phone No	Message	Date	Action
1	test data	afiqahsamsuri988@gmail.com	0134387250	hai, please contact me	2021-08-28 13:28:39	○ Accept ○ Reject

Showing 1 to 1 of 1 entries

Copyright © Web Application to Rent House. All rights reserved.

Figure 4.18 List of Application's Responses



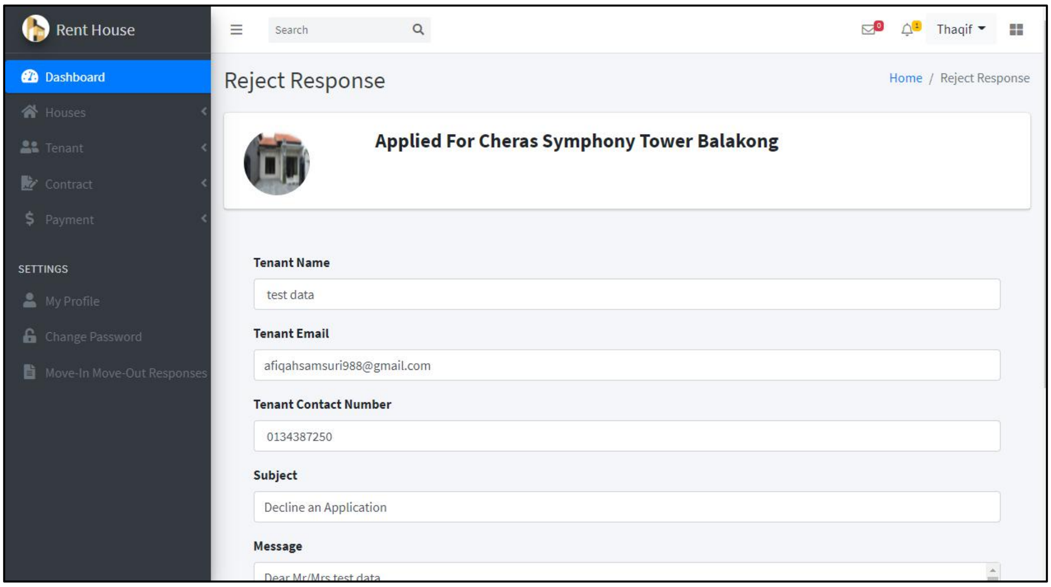
Property

Applied For Cheras Symphony Tower Balakong

Tenant Name	test data
Monthly Rental (RM)	850
Deposit (RM)	850
Tenant Email	afiqahsamsuri988@gmail.com
Subject	Responses to Rental House Applications
Contract and Agreement	

Figure 4.19 Accept Application Response Page

The list of tenants who want to apply for the residence can be viewed by the landlord. The system would send an email to the landlord informing him or her of the situation. The landlord will be directed to the response wish list page after clicking on the notification. If the landlord intends for the tenant to rent the house, he will click the accept button. The landlord will then be directed to accept the application response page as shown on figure 4.19.



Reject Response

Applied For Cheras Symphony Tower Balakong

Tenant Name

test data

Tenant Email

afiqahsamsuri988@gmail.com

Tenant Contact Number

0134387250

Subject

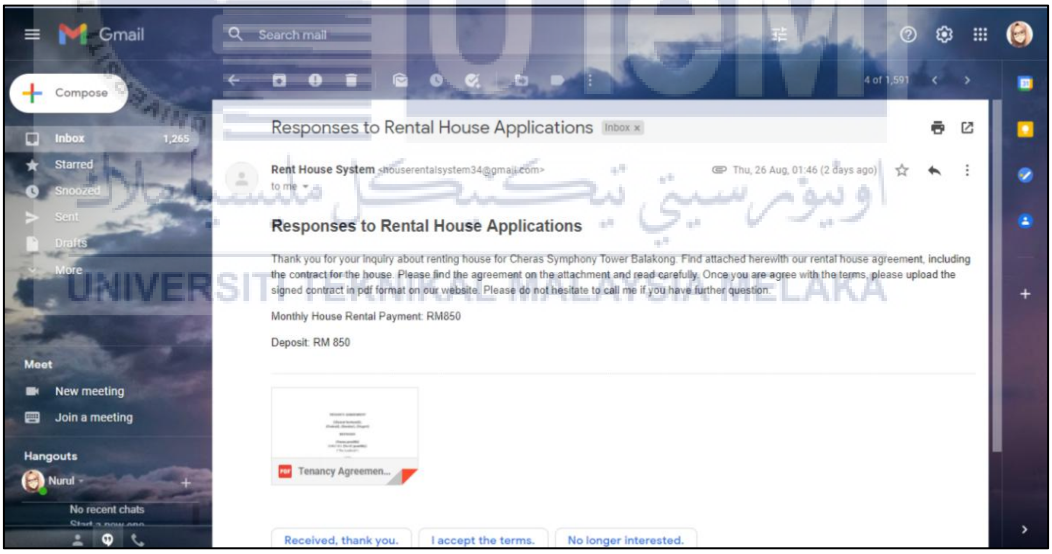
Decline an Application

Message

Dear Mr/Mrs test data

Figure 4.20 Reject Application Response Page

Figure 4.20 shows the page for the landlord to reject the other tenants' responses if the house is no longer available to rent.



Responses to Rental House Applications

Rent House System <houserentalsystem34@gmail.com>
to me

Thu, 26 Aug, 01:46 (2 days ago)

Responses to Rental House Applications

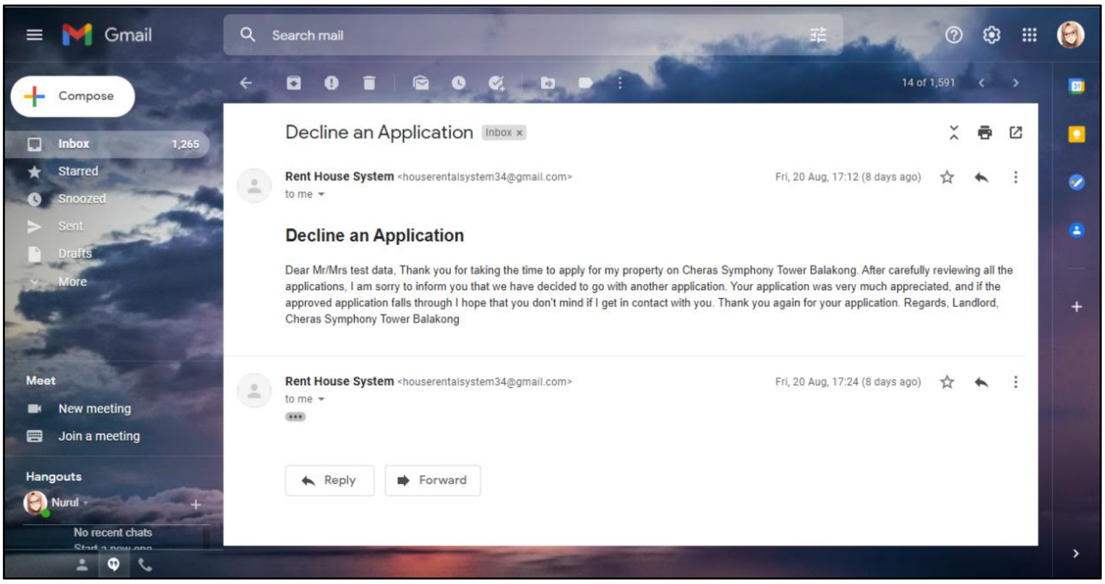
Thank you for your inquiry about renting house for Cheras Symphony Tower Balakong. Find attached herewith our rental house agreement, including the contract for the house. Please find the agreement on the attachment and read carefully. Once you are agree with the terms, please upload the signed contract in pdf format on our website. Please do not hesitate to call me if you have further question.

Monthly House Rental Payment: RM850

Deposit: RM 850

Figure 4.21 Responses of Email with Contract Attachment

Figure 4.21 depicts an example of an email from the landlord to a prospective tenant. When the landlord approves their rental house response, the system will send an email including the lease agreement. Before uploading the signed contract to the system and moving on to the next stage, the possible tenant must read and agree to the agreement.



Decline an Application

Rent House System <houserentalsystem34@gmail.com>
to me
Fri, 20 Aug, 17:12 (8 days ago)

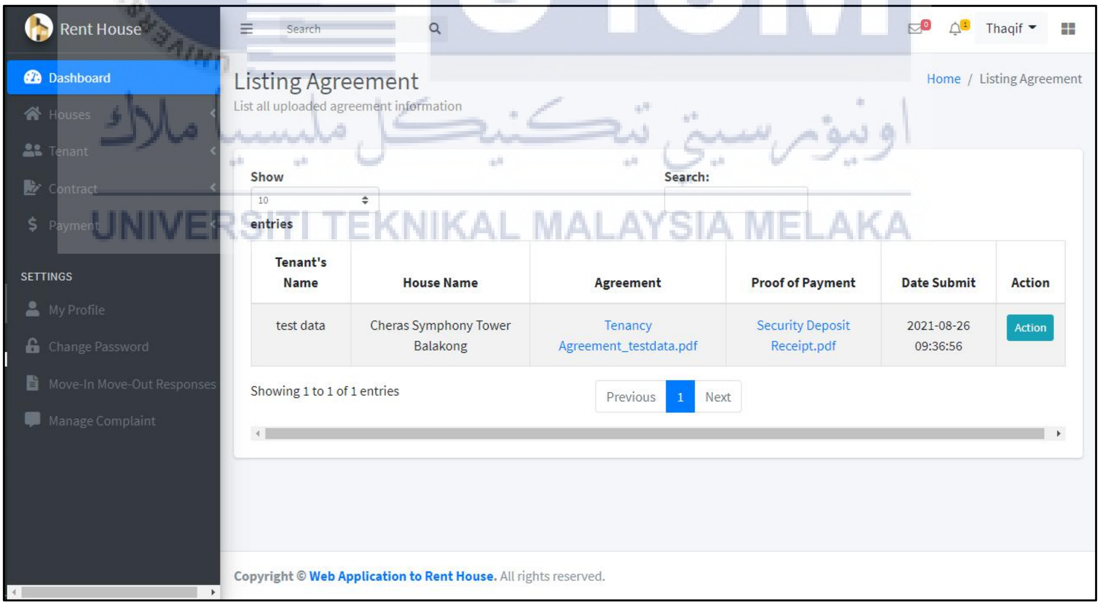
Decline an Application

Dear Mr/Mrs test data, Thank you for taking the time to apply for my property on Cheras Symphony Tower Balakong. After carefully reviewing all the applications, I am sorry to inform you that we have decided to go with another application. Your application was very much appreciated, and if the approved application falls through I hope that you don't mind if I get in contact with you. Thank you again for your application. Regards, Landlord, Cheras Symphony Tower Balakong

Rent House System <houserentalsystem34@gmail.com>
to me
Fri, 20 Aug, 17:24 (8 days ago)

Figure 4.22 Reject Email Responses

If the landlord rejects their response, the system will send an email to the possible tenant informing them of the rejection. Figure 4.22 shows an example of a landlord's rejection of an tenant application.



Listing Agreement

List all uploaded agreement information

Show 10 entries

Search:

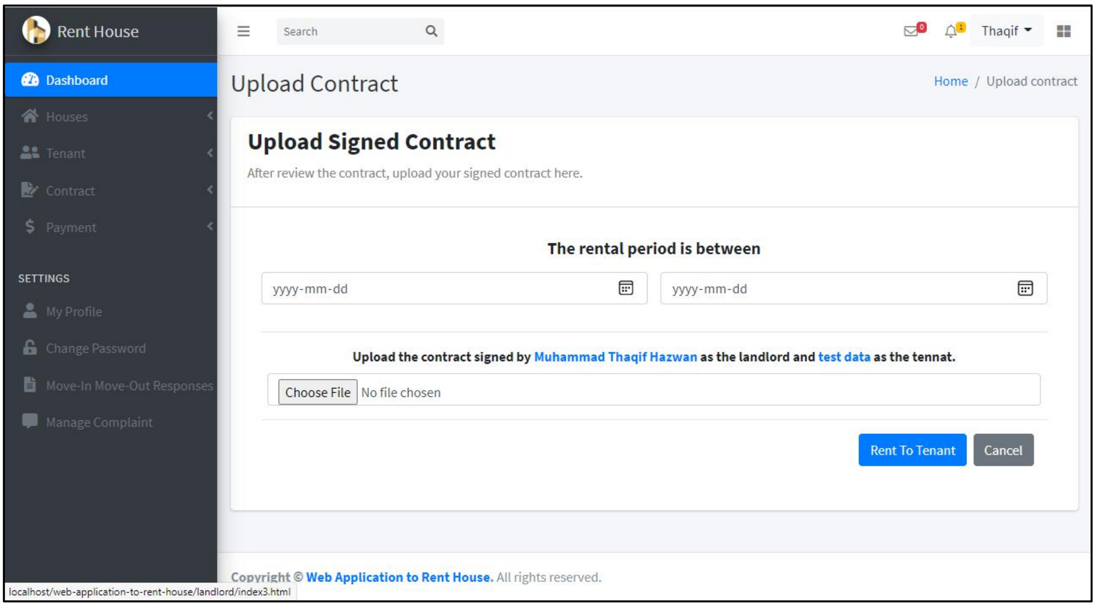
Tenant's Name	House Name	Agreement	Proof of Payment	Date Submit	Action
test data	Cheras Symphony Tower Balakong	Tenancy Agreement_testdata.pdf	Security Deposit Receipt.pdf	2021-08-26 09:36:56	Action

Showing 1 to 1 of 1 entries

Copyright © Web Application to Rent House. All rights reserved.

Figure4.23 List all contract uploaded by tenant

Based on figure 4.23 shows the list proof of payment and lease agreement that had been signed by the tenant. The landlord will be able to download and view the lease agreement and proof of payment that had been uploaded by the tenant.



Upload Contract

Upload Signed Contract

After review the contract, upload your signed contract here.

The rental period is between

yyyy-mm-dd   yyyy-mm-dd

Upload the contract signed by Muhammad Thaqif Hazwan as the landlord and test data as the tennat.

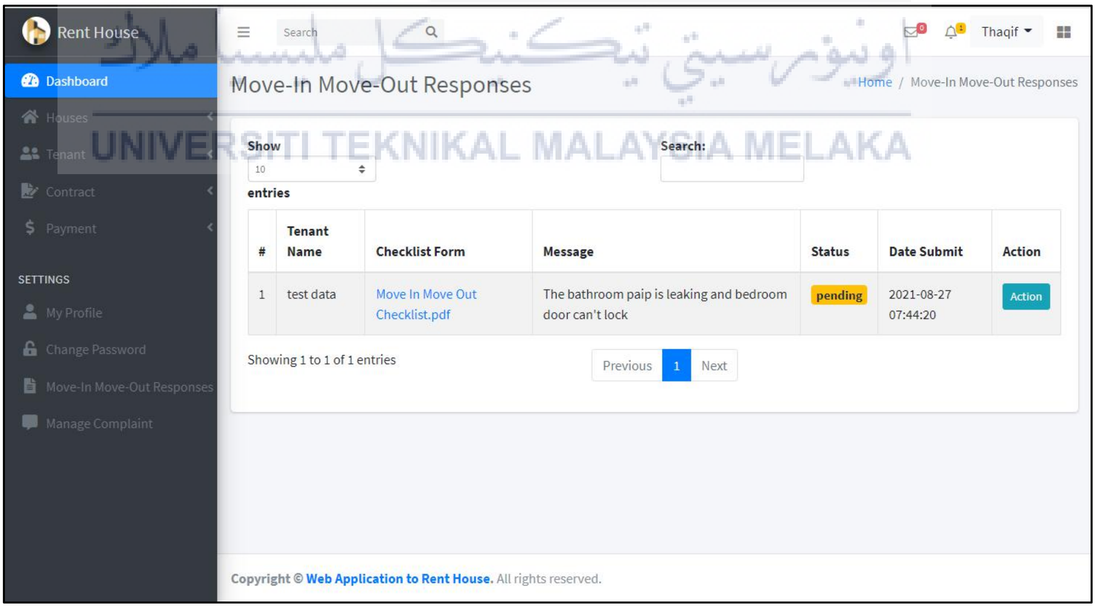
Choose File No file chosen

Copyright © Web Application to Rent House. All rights reserved.

localhost/web-application-to-rent-house/landlord/index3.html

Figure 4.24 Landlord upload contract page

According to figure 4.24, once the landlord has reviewed the lease agreement, the landlord will upload the signed agreement and allocate the rental period agreed upon by the tenant from the agreement by selecting the Rent to Tenant button. The system will then change the status of the house from available to in-agreement.



Move-In Move-Out Responses

Show 10 entries

Search:

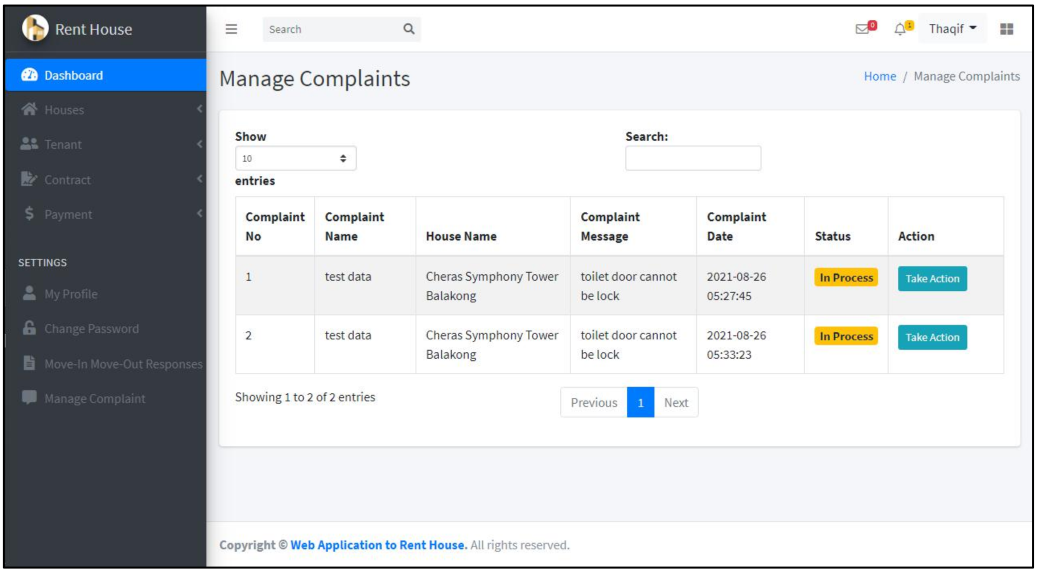
#	Tenant Name	Checklist Form	Message	Status	Date Submit	Action
1	test data	Move In Move Out Checklist.pdf	The bathroom paip is leaking and bedroom door can't lock	pending	2021-08-27 07:44:20	Action

Showing 1 to 1 of 1 entries

Copyright © Web Application to Rent House. All rights reserved.

Figure 4.25 List of move-in move-out responses

Based on figure 4.25, the list of checklist responses has been uploaded by the tenant once they occupy the house.



Manage Complaints

Show 10 entries

Search:

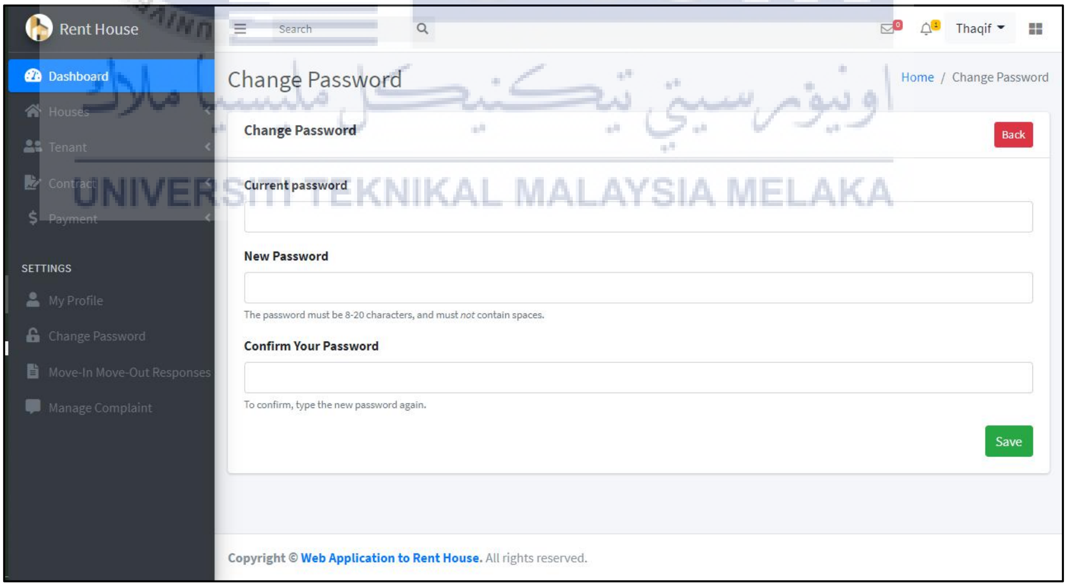
Complaint No	Complaint Name	House Name	Complaint Message	Complaint Date	Status	Action
1	test data	Cheras Symphony Tower Balakong	toilet door cannot be lock	2021-08-26 05:27:45	In Process	Take Action
2	test data	Cheras Symphony Tower Balakong	toilet door cannot be lock	2021-08-26 05:33:23	In Process	Take Action

Showing 1 to 2 of 2 entries

Copyright © Web Application to Rent House. All rights reserved.

Figure 4.26 List of complaints

The landlord will be able to manage the complaint that has been reported by the tenant. Based on figure 4.26, it shows the list of complaints that have been reported by the tenants. The system will show the in-process status if the complaint has not been resolved yet.



Change Password

Change Password

Current password

New Password

The password must be 8-20 characters, and must *not* contain spaces.

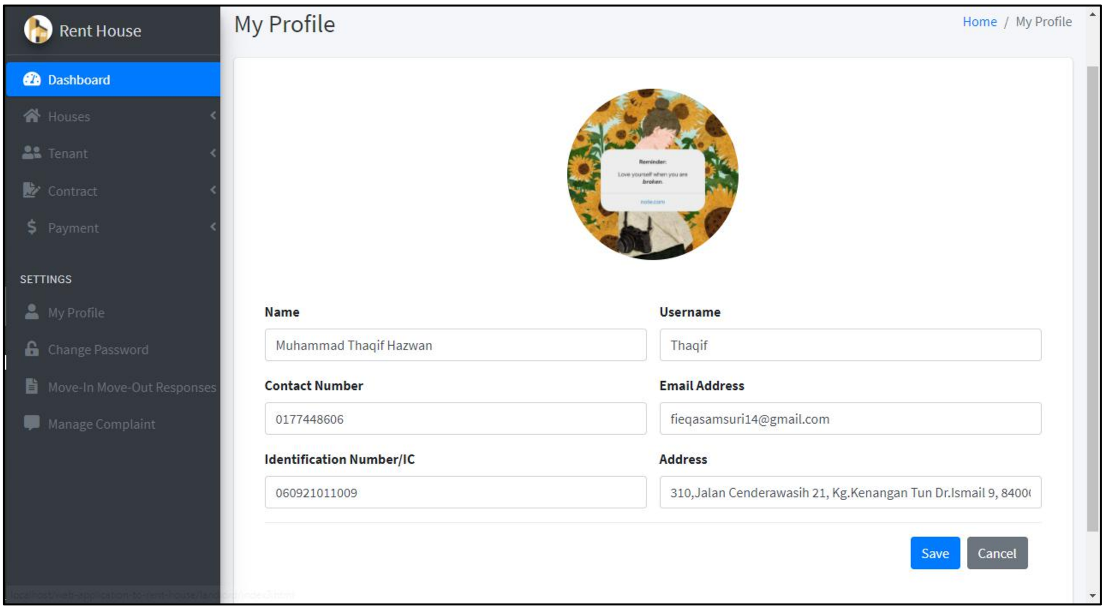
Confirm Your Password

To confirm, type the new password again.

Copyright © Web Application to Rent House. All rights reserved.

Figure 4.27 Change password page

Based on figure 4.27, the landlord will be able to change their password and the system will automatically update their new password. After the reset password is successful, the landlord needs to login again into the system.



My Profile

Name: Muhammad Thaqif Hazwan

Username: Thaqif

Contact Number: 0177448606

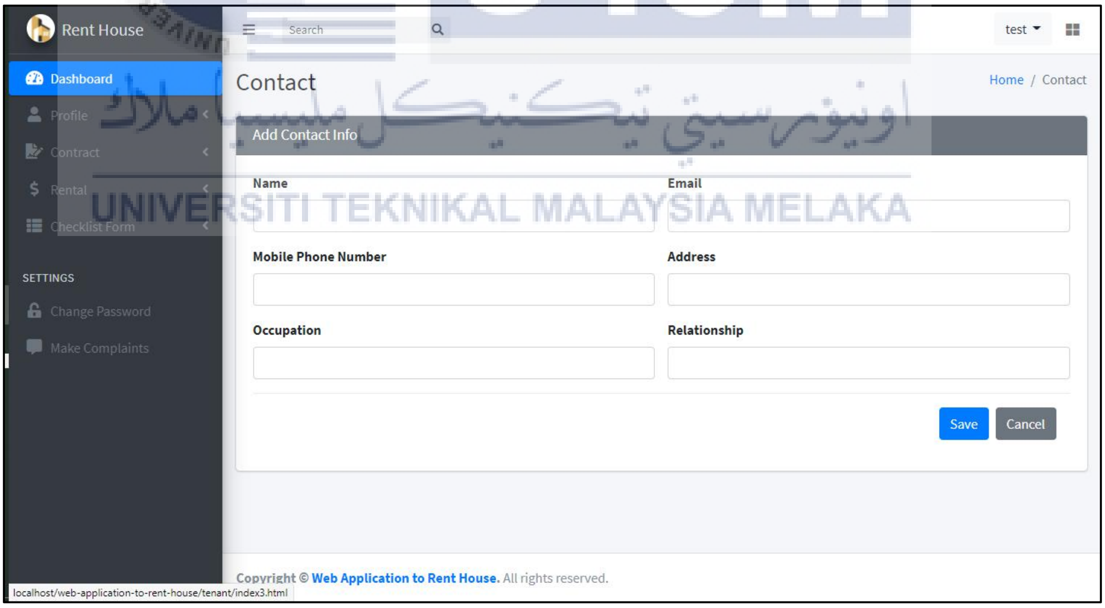
Email Address: fieqasamsuri14@gmail.com

Identification Number/IC: 060921011009

Address: 310,Jalan Cenderawasih 21, Kg.Kenangan Tun Dr.Ismail 9, 8400(

Figure 4.28 User profile page

Based on figure 4.28, which shows the tenant profile. Once the user has been registered and logged into the system for the first time, the tenant needs to update their information by filling in the fields given by the system.

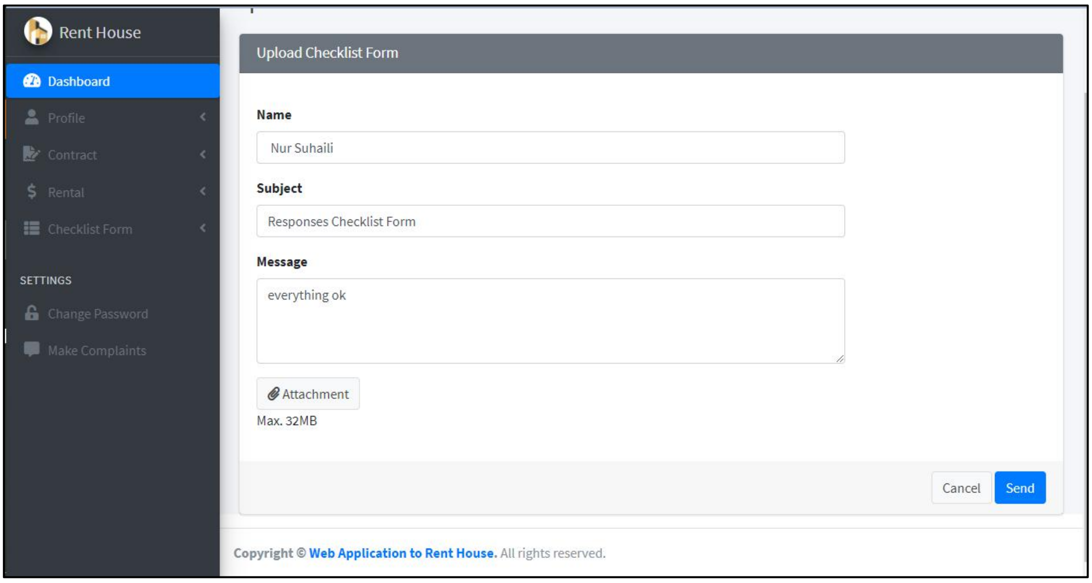


Contact

Copyright © Web Application to Rent House. All rights reserved.

Figure 4.29 Add contact info page

The tenant needs provide their contact information into the system as a reference for the landlord in case the tenant cannot be reached as shown on figure 4.29.



Upload Checklist Form

Name: Nur Suhaili

Subject: Responses Checklist Form

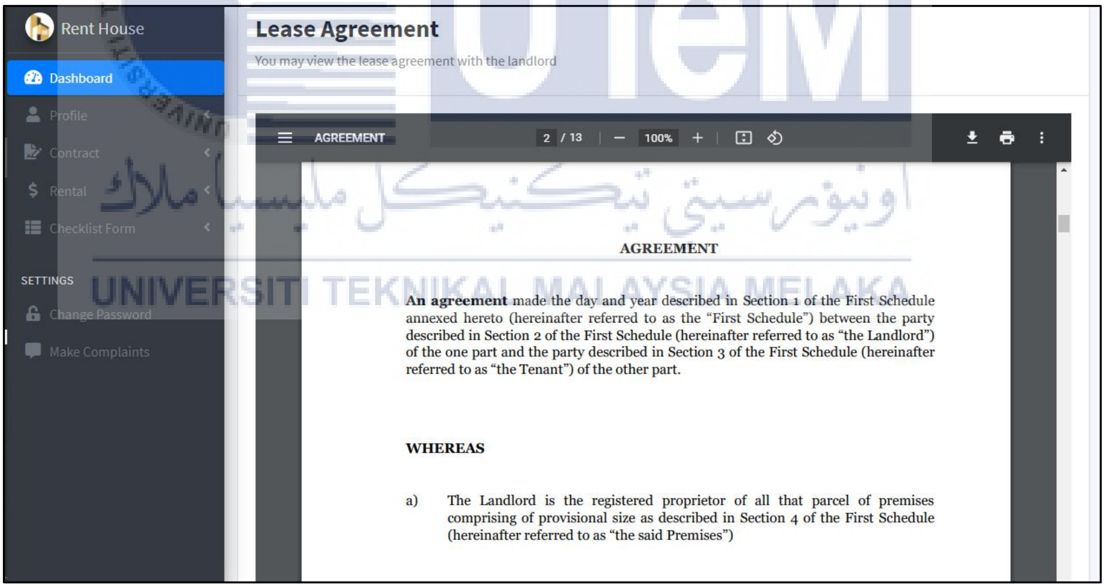
Message: everything ok

Max. 32MB

Copyright © **Web Application to Rent House.** All rights reserved.

Figure 4.30 View tenant's profile with contact info

The tenant can view their profile information with the contact information that had been added by the tenant, as shown in figure 4.30.



Lease Agreement

You may view the lease agreement with the landlord

AGREEMENT

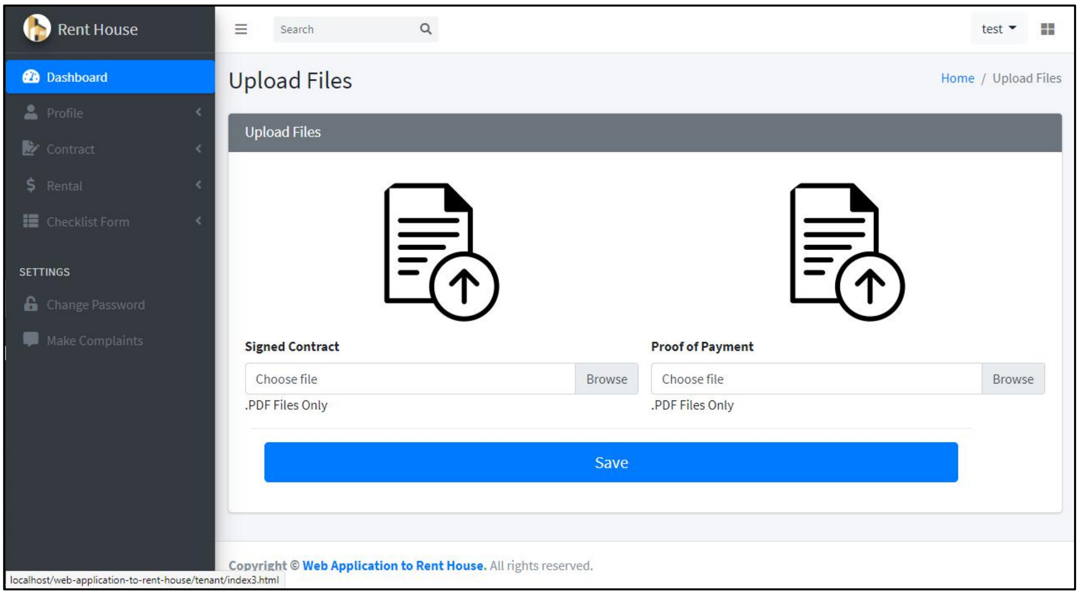
An agreement made the day and year described in Section 1 of the First Schedule annexed hereto (hereinafter referred to as the "First Schedule") between the party described in Section 2 of the First Schedule (hereinafter referred to as "the Landlord") of the one part and the party described in Section 3 of the First Schedule (hereinafter referred to as "the Tenant") of the other part.

WHEREAS

a) The Landlord is the registered proprietor of all that parcel of premises comprising of provisional size as described in Section 4 of the First Schedule (hereinafter referred to as "the said Premises")

Figure 4.31 View tenant contract

As shown in figure 4.31, the tenant can view and download their lease agreement with the landlord for future reference.



Upload Files

Upload Files

Signed Contract

Choose file

.PDF Files Only

Proof of Payment

Choose file

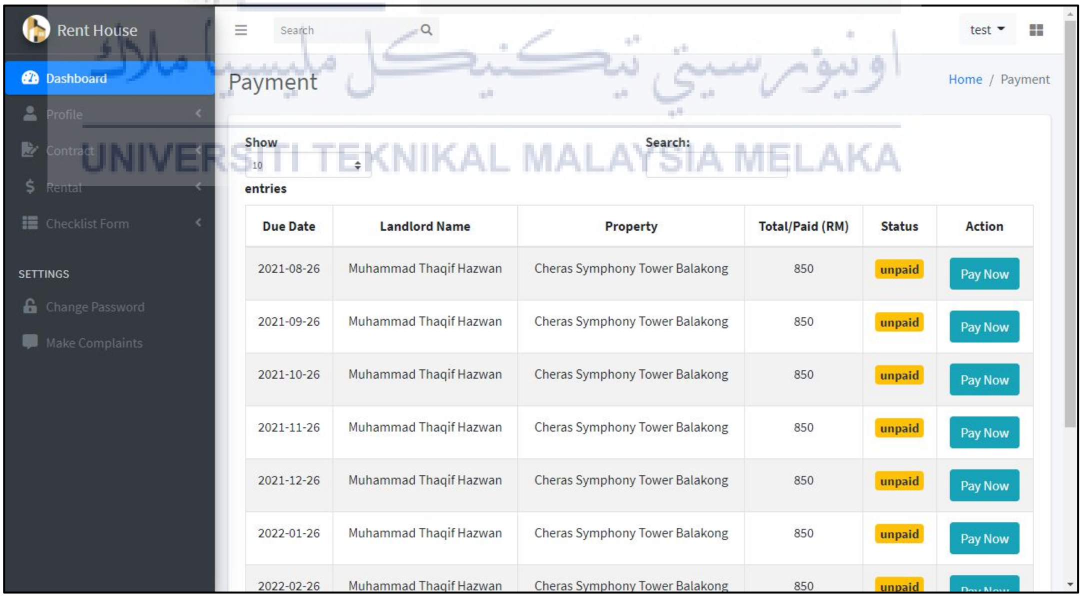
.PDF Files Only

Copyright © Web Application to Rent House. All rights reserved.

localhost/web-application-to-rent-house/tenant/index3.html

Figure 4.32 Upload contract and proof of payment page

Based on 4.32, the tenant need upload the signed lease agreement as well as proof of payment to the system in order for the landlord to proceed with the rental placement. The system will automatically notify the tenant through email once the tenant has been uploaded.



Payment

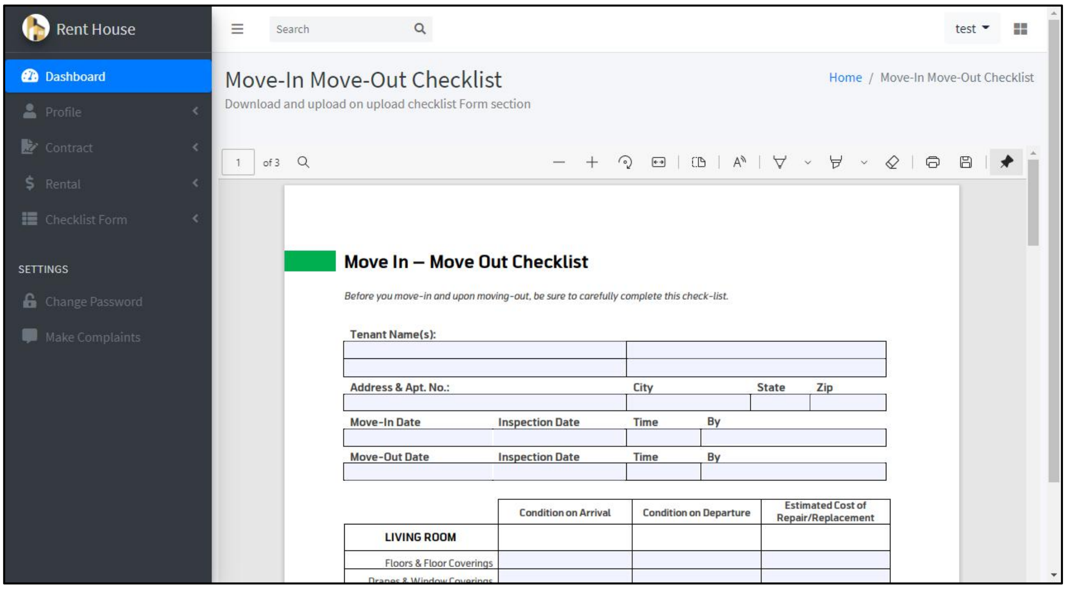
Show 10 entries

Search:

Due Date	Landlord Name	Property	Total/Paid (RM)	Status	Action
2021-08-26	Muhammad Thaqif Hazwan	Cheras Symphony Tower Balakong	850	unpaid	Pay Now
2021-09-26	Muhammad Thaqif Hazwan	Cheras Symphony Tower Balakong	850	unpaid	Pay Now
2021-10-26	Muhammad Thaqif Hazwan	Cheras Symphony Tower Balakong	850	unpaid	Pay Now
2021-11-26	Muhammad Thaqif Hazwan	Cheras Symphony Tower Balakong	850	unpaid	Pay Now
2021-12-26	Muhammad Thaqif Hazwan	Cheras Symphony Tower Balakong	850	unpaid	Pay Now
2022-01-26	Muhammad Thaqif Hazwan	Cheras Symphony Tower Balakong	850	unpaid	Pay Now
2022-02-26	Muhammad Thaqif Hazwan	Cheras Symphony Tower Balakong	850	unpaid	Pay Now

Figure 4.33 List of payment rental

The tenant will be able to see a list of rental payments that must be made before the due date throughout the rental period. The list of due dates, total rent to be paid, and payment status is shown in Figure 4.33.



Move-In Move-Out Checklist

Download and upload on upload checklist Form section

Move In – Move Out Checklist

Before you move-in and upon moving-out, be sure to carefully complete this check-list.

Tenant Name(s):

Address & Apt. No.: City State Zip

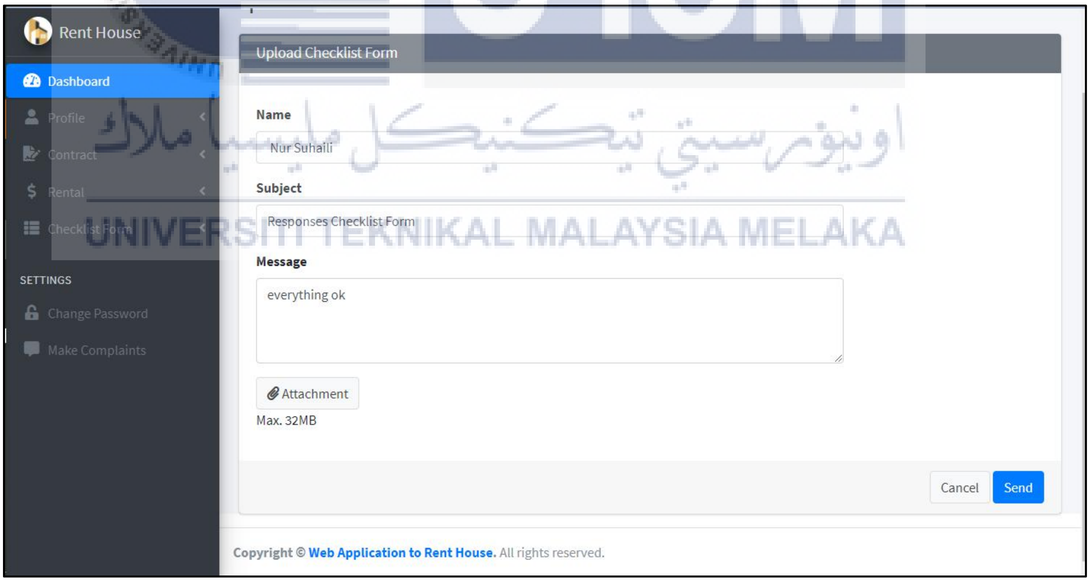
Move-In Date Inspection Date Time By

Move-Out Date Inspection Date Time By

	Condition on Arrival	Condition on Departure	Estimated Cost of Repair/Replacement
LIVING ROOM			
Floors & Floor Coverings			
Drapes & Window Coverings			

Figure 4.34 Download checklist form page

Based on figure 4.32, the checklist form needs to be completed by the tenant once they occupy the house. The tenant needs to download the checklist form and fill in all the information given.



Upload Checklist Form

Name: Nur Suhaili

Subject: Responses Checklist Form

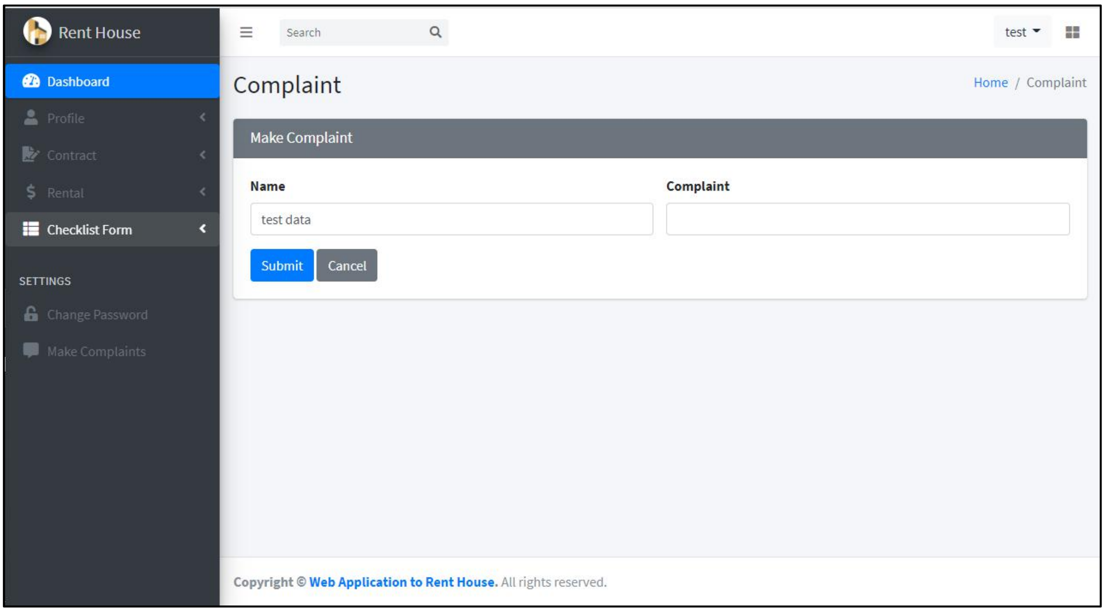
Message: everything ok

Max. 32MB

Copyright © Web Application to Rent House. All rights reserved.

Figure 4.35 Upload checklist form page

After the tenant has completed the checklist form, the tenant will need to upload the form to the checklist form page as shown in figure 4.35.



Complaint

Make Complaint

Name: test data

Complaint:

Copyright © Web Application to Rent House. All rights reserved.

Figure 4.36 Make complaint page

Based on figure 4.36, a tenant will be able to make a complaint if there is any damage to the utilities or facilities in the rental house during the rental period.

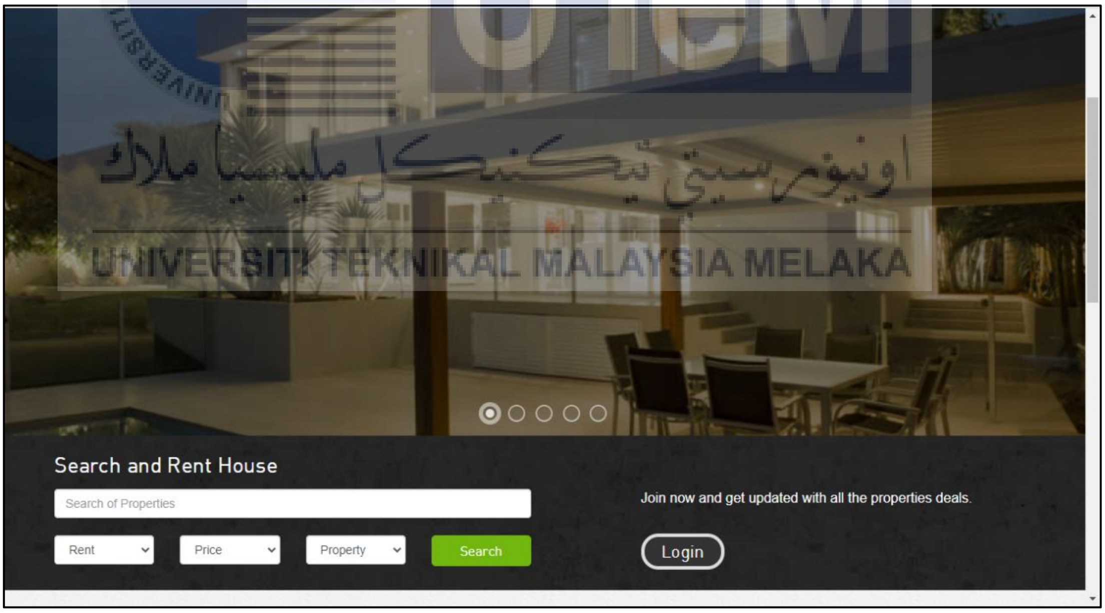


Figure 4.37 Home page of the system

Figure 4.37 shows the home page of the rental house system. The tenant will be able to search and rent the house on the system.

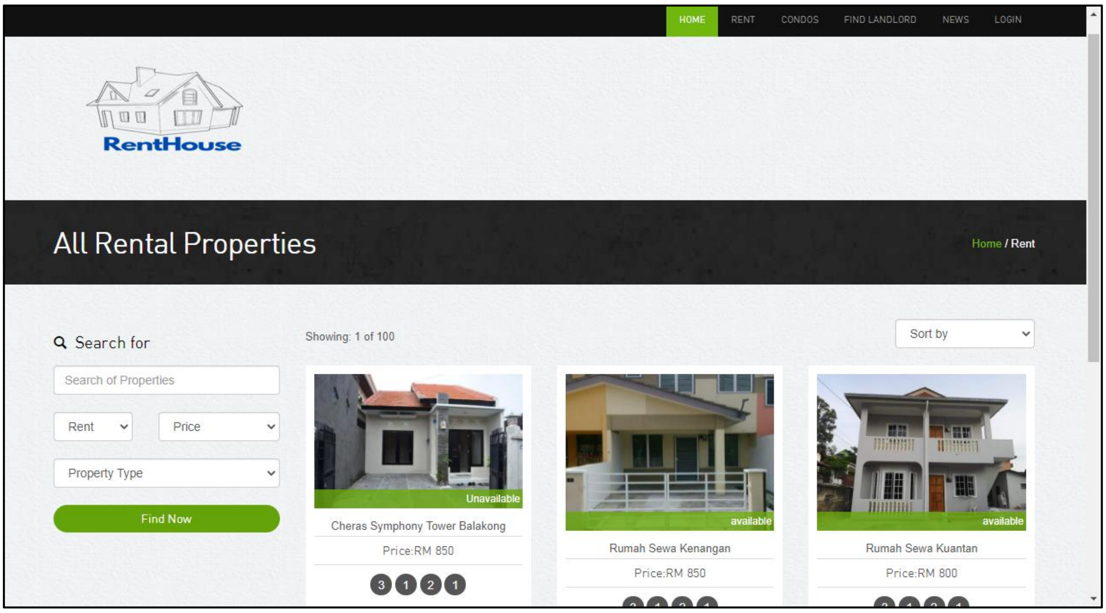


Figure 4.38 List of rental houses

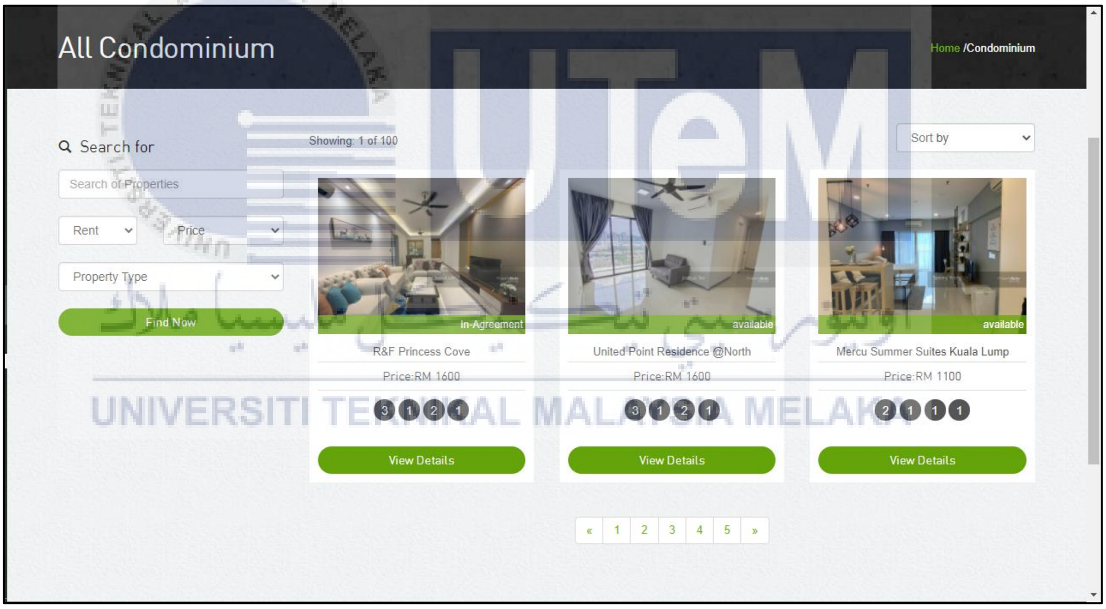
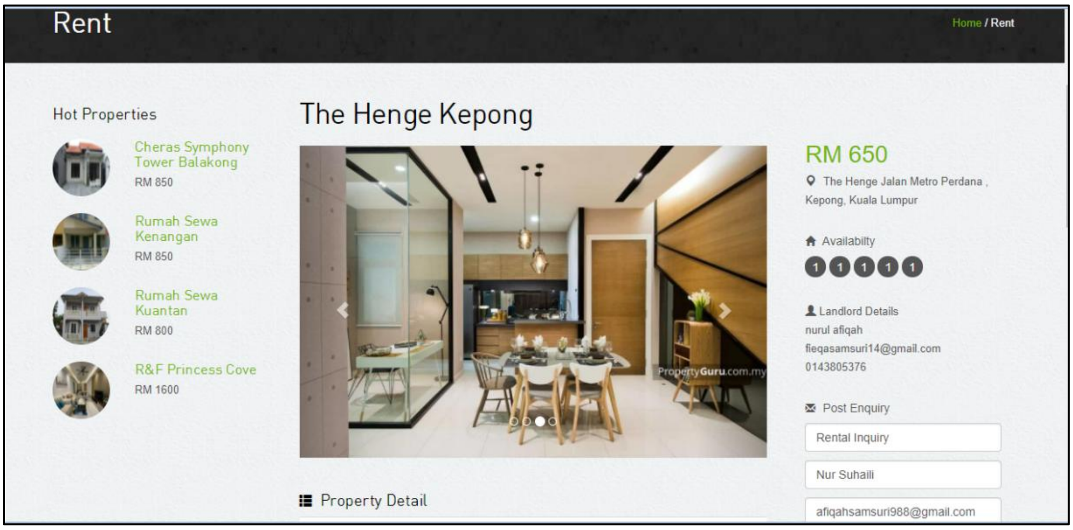


Figure 4.39 List of condominium

Based on figure 4.38, figure 4.39 shows the list of rental houses available on the system for the tenant to rent. The houses are divided into two parts, which are terrace houses and condominiums, and are categorized by the house type. The tenant will be able to view the details of the rental house by clicking on the view details button.



Rent

Home / Rent

Hot Properties

Cheras Symphony Tower Balakong

RM 850

Rumah Sewa Kenangan

RM 850

Rumah Sewa Kuantan

RM 800

R&F Princess Cove

RM 1600

The Henge Kepong

RM 650

The Henge Jalan Metro Perdana , Kepong, Kuala Lumpur

Availabilty

1 1 1 1 1

Landlord Details

nurul afiqah

fieqasamsuri14@gmail.com

0143805376

Post Enquiry

Rental Inquiry

Nur Suhaili

afiqahsamsuri988@gmail.com

Property Detail

Figure 4.40 House rental details

Figure 4.40 shows a list of landlords together with their contact information. The tenant will be able to view it if they want to know about their landlord.

4.2.3 Database Design

4.2.3.1 Conceptual and Logical Database Design

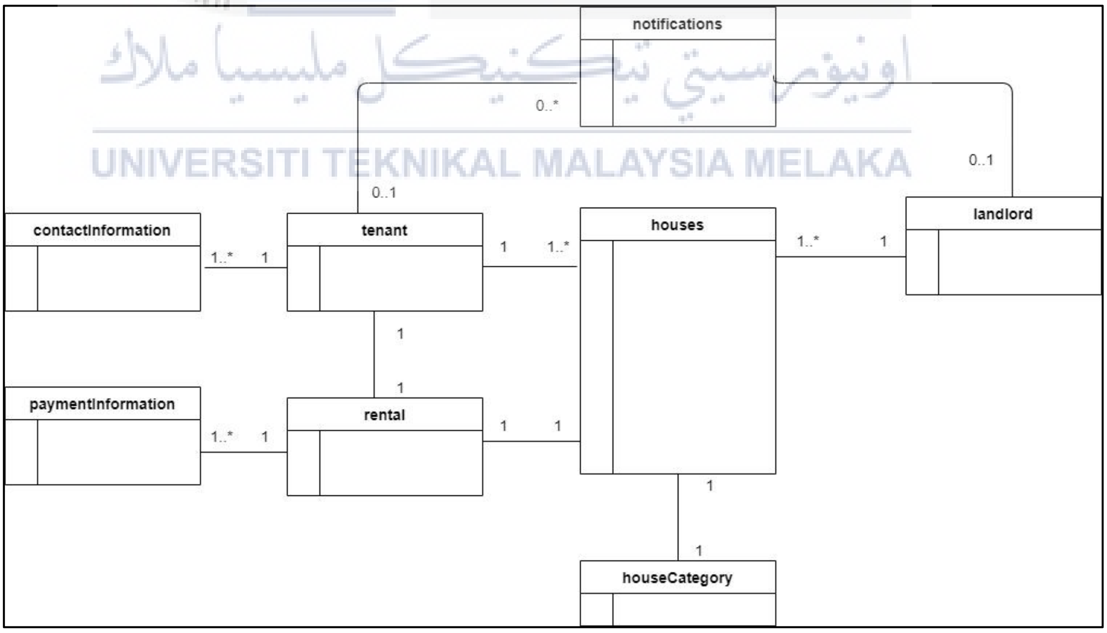


Figure 4.41 Conceptual ERD for House Rental Application System

Based on the figure 4.27, the Entity Relationship Diagram for Web-based House Rental System will be explained as below:

- Notification table - An alert will be recorded in this table for displaying the notification on the system.

- Contact information table - All the tenant's contact information are stored in the table.

- Tenant table - All of the tenant details are stored in this table for ease the business process.

- Landlord table - All of the landlord detailed information are stored in the table.

- Houses table - All of the house details are stored in this table. The house id is the primary key, which is a unique key used as reference to other table.

- House category table - All of the house category are recorded in the table.

- Rental table - Part of rental information such as rental no, tenant details, contract, proof of payment, start date and end date of renting the house is stored in this table.

- Payment information table - Part of payment information such as payment amount, payment type, date of payment is stored in the table. The rental id is a reference key from the rental table.

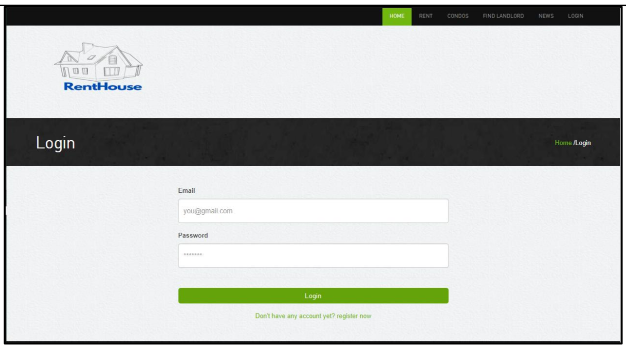
4.3 Detailed Design

Detailed design will show how web-based House Rental Application System (HRAS) should operate when receive interaction from user. In this section, it will describe in detailed the system design with database.

4.3.1 Software Design

This section will describe on how the web-based House Rental Application System (HRAS) operates on their module and the expected output that are shown from the system from the interaction between the user and the system.

4.3.1.1 Login User

Program Name	**HRAS_001**
Description	- Receive information of users and save the data into the database. - Admin/Tenant/Landlord can log into the system as the database contains all information of users in the system as shown on Figure 4.42.
Input/Output	Input: Email address and password Output: The users will direct into their page based on their role.
Pseudo code	Step 1: Initializes all working variables to zeroes. Step 2: Fill in the needed inputs(email and password). Step 3: Click the button "Login" to the system.
Screen Format	 HOME RENT CONDOS FIND LANDLORD NEWS LOGIN RentHouse Login Home /Login Email you@gmail.com Password Login Don't have any account yet? register now **Figure 4.42 Login page**

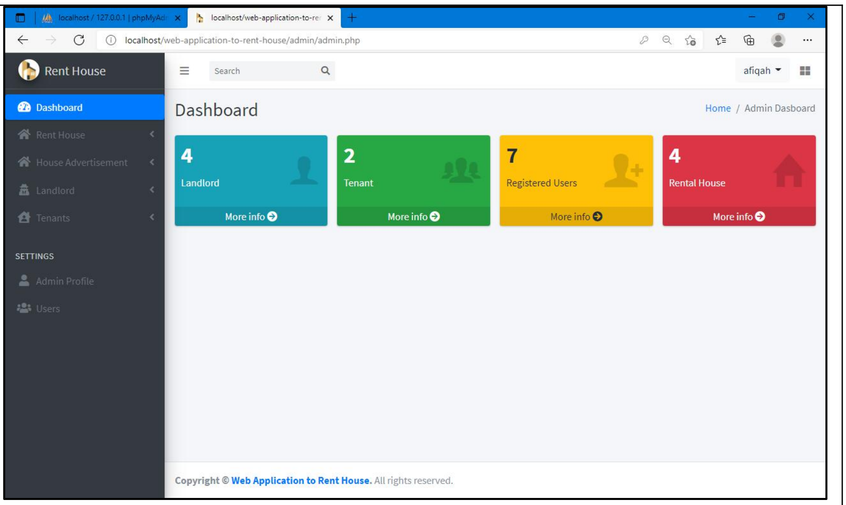
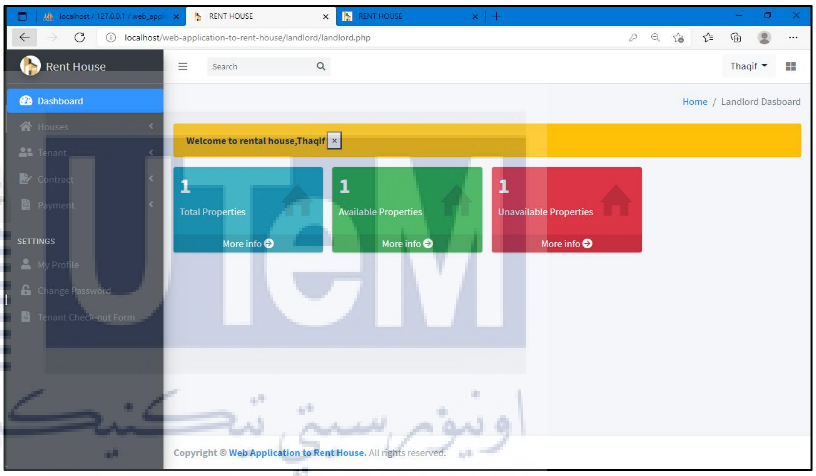


Figure 4.43 Admin Dashboard



Welcome to rental house,Thaqif

Figure 4.44 Landlord Dashboard

4.3.1.2 Manage Landlord

Program Name	**HRAS_002**
Description	- Sent and receive information of users and save into the database. - Admin can add, update, delete landlord in the system as shown on Figure 4.45, 4.46 and 4.47.
Input/output	- Add Landlord

	Input: name, username, NRIC/IC, email, contact number, address. Output: The information of landlord successfully inserted in the manage landlord page. - Delete Landlord: Output: The information of landlord has successfully delete in the manage landlord page. - Update Landlord: Input: name, username, NRIC/IC, email, contact number, address. Output: The information of landlord successfully updated in the manage landlord page.
Pseudo code	- Add Landlord Step 1: Click button “Add Landlord” to add landlord in the system. Step 2: Fill in the needed inputs(name, username, NRIC/IC, email, contact number, address). Step 3: Click button “Save” to add landlord to the system. - Delete Landlord Step 1:Click the icon “delete” and display a confirmation message to delete the landlord from the system. Step 2: Click button “Delete” to delete the landlord. - Update Landlord

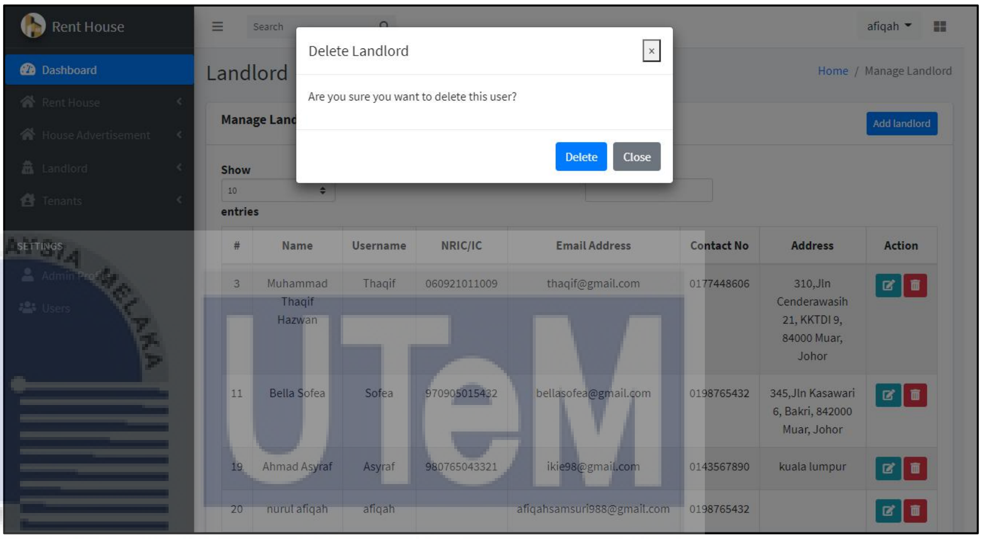
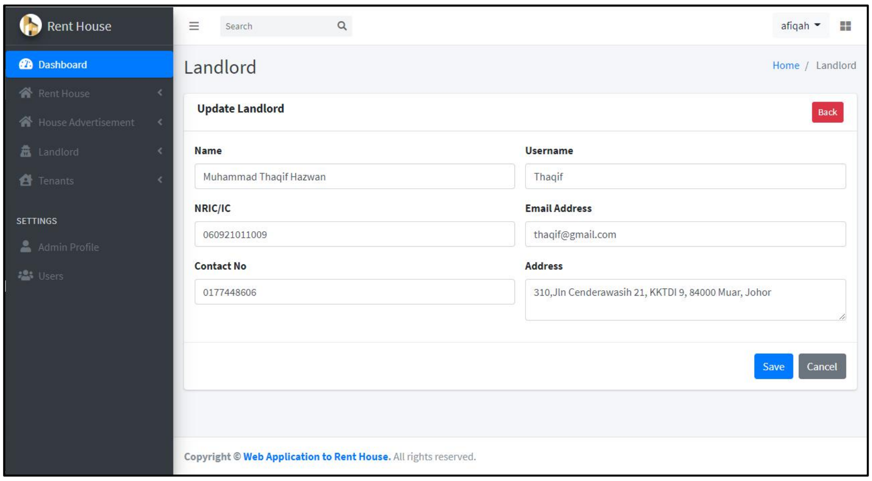
	Step 1: Click icon “update” to update landlord in the system. Step 2: Fill in the needed inputs(name, username, NRIC/IC, email, contact number, address). Step 3: Click the button “Save” to update the landlord.
Screen Format	- Delete Landlord  **Figure 4.45 Delete Landlord page** - Update Landlord  **Figure 4.46 Update Landlord Page**

Figure 4.45:

Landlord

Manage Land

Delete Landlord

Are you sure you want to delete this user?

Delete Close

Show 10 entries

#	Name	Username	NRIC/IC	Email Address	Contact No	Address	Action
3	Muhammad Thaqif Hazwan	Thaqif	060921011009	thaqif@gmail.com	0177448606	310,Jln Cenderawasih 21, KKTDI 9, 84000 Muar, Johor	
11	Bella Sofea	Sofea	970905015432	bellasofea@gmail.com	0198765432	345,Jln Kasawari 6, Bakri, 842000 Muar, Johor	
19	Ahmad Asyraf	Asyraf	980765043321	ikie98@gmail.com	0143567890	kuala lumpur	
20	nurul afiqah	afiqah		afiqahsamsuri988@gmail.com	0198765432		

Figure 4.46:

Landlord

Update Landlord

Name: Muhammad Thaqif Hazwan

Username: Thaqif

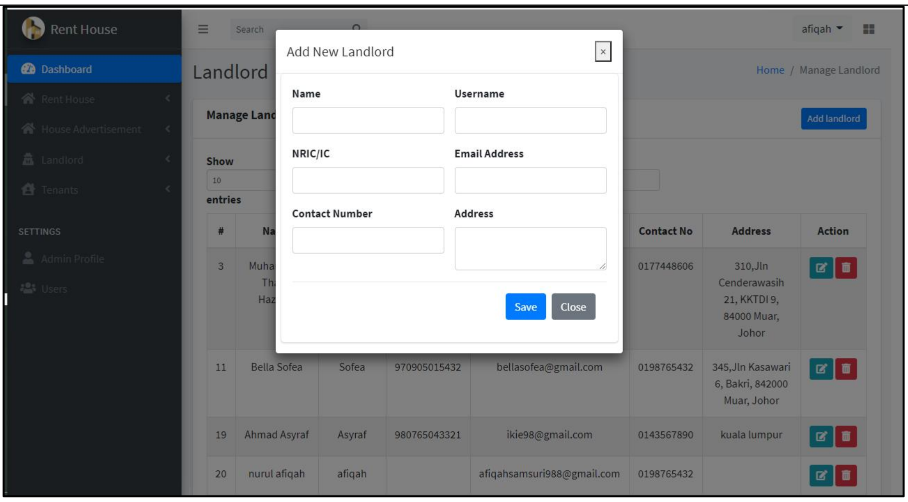
NRIC/IC: 060921011009

Email Address: thaqif@gmail.com

Contact No: 0177448606

Address: 310,Jln Cenderawasih 21, KKTDI 9, 84000 Muar, Johor

Copyright © Web Application to Rent House. All rights reserved.



Add New Landlord ×

Name

Username

NRIC/IC

Email Address

Contact Number

Address

Save Close

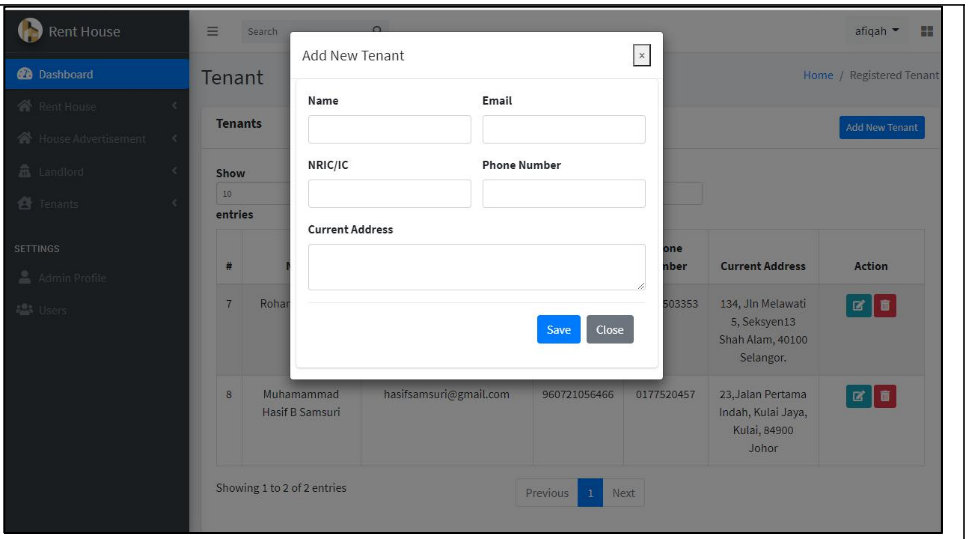
#	Na				Contact No	Address	Action
3	Muha Th Haz				0177448606	310,Jln Cenderawasih 21, KKTDI 9, 84000 Muar, Johor	
11	Bella Sofea	Sofea	970905015432	bellasofea@gmail.com	0198765432	345,Jln Kasawari 6, Bakri, 842000 Muar, Johor	
19	Ahmad Asyraf	Asyraf	980765043321	ikie98@gmail.com	0143567890	kuala lumpur	
20	nurul afiqah	afiqah		afiqahsamsuri988@gmail.com	0198765432		

Figure 4.47 Add Landlord Page

4.3.1.3 Manage Tenant

Program Name	**HRAS_003**
Description	- Sent and receive information of users and save into the database. - Admin can add, update, delete tenant in the system as shown on Figure 4.48, 4.49 and 4.50.
Input/Output	- Add Tenant Input: name, username, NRIC/IC, email, contact number, address. Output: The information of tenant successfully inserted in the manage tenant page. - Delete Tenant: Output: The information of tenant has successfully delete in the manage tenant page. - Update Tenant:

	Input: name, username, NRIC/IC, email, contact number, address. Output: The information of tenant successfully updated in the manage tenant page.
Pseudo code	- Add Tenant Step 1: Click button “Add Tenant” to add landlord in the system. Step 2: Fill in the needed inputs(name, username, NRIC/IC, email, contact number, address). Step 3: Click button “Save” to add tenant to the system. - Delete Tenant Step 1:Click the icon “delete” and display a confirmation message to delete the tenant from the system. Step 2: Click button “Delete” to delete the tenant. - Update Tenant Step 1: Click icon “update” to update Tenant in the system. Step 2: Fill in the needed inputs(name, username, NRIC/IC, email, contact number, address). Step 3: Click the button “Save” to update the Tenant.
Screen format	- Add Tenant



Tenant

Tenants

Show 10 entries

Add New Tenant

Name

Email

NRIC/IC

Phone Number

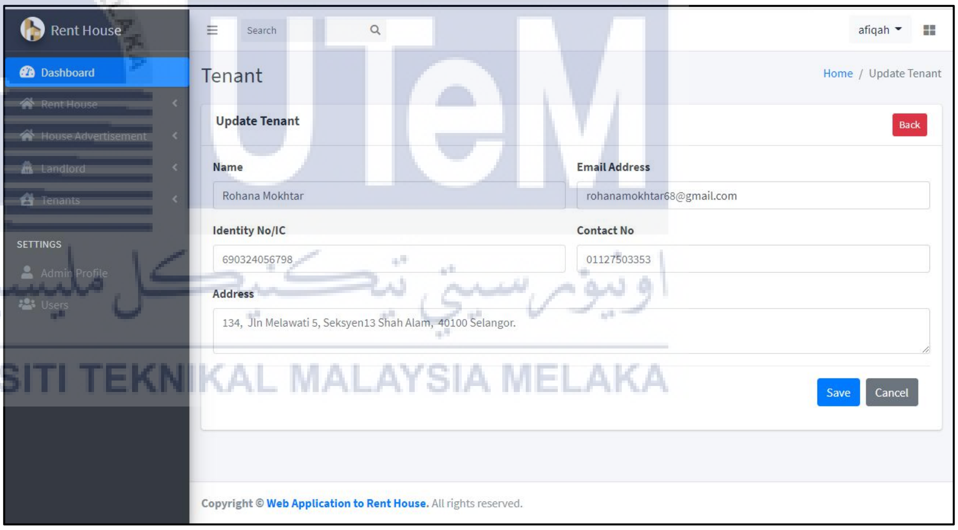
Current Address

#				one nber	Current Address	Action
7	Rohan			503353	134, Jln Melawati 5, Seksyen13 Shah Alam, 40100 Selangor.	
8	Muhamammad Hasif B Samsuri	hasifsamsuri@gmail.com	960721056466	0177520457	23,Jalan Pertama Indah, Kulai Jaya, Kulai, 84900 Johor	

Showing 1 to 2 of 2 entries

Figure 4.48 Add Tenant Page

- Update Tenant



Tenant

Update Tenant

Name: Rohana Mokhtar

Email Address: rohanamokhtar68@gmail.com

Identity No/IC: 690324056798

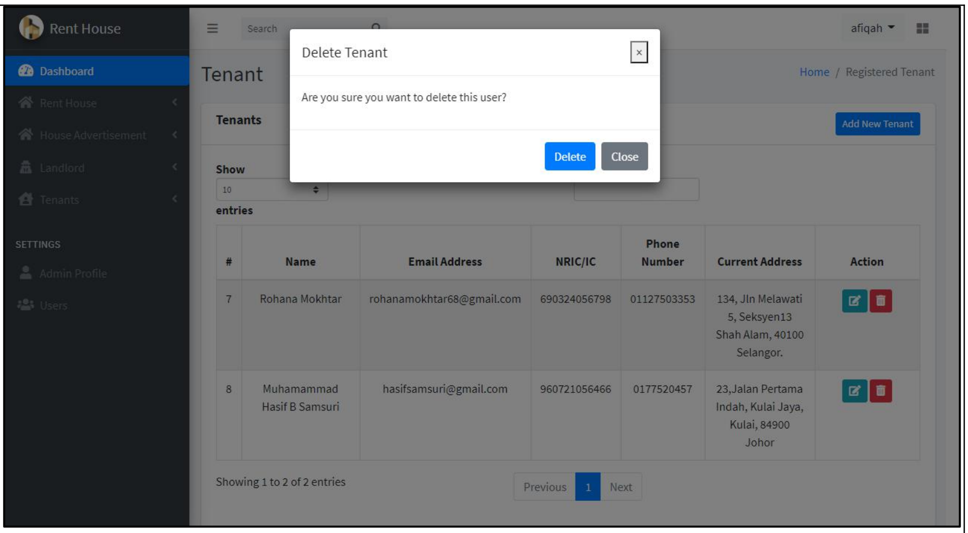
Contact No: 01127503353

Address: 134, Jln Melawati 5, Seksyen13 Shah Alam, 40100 Selangor.

Copyright © Web Application to Rent House. All rights reserved.

Figure 4.49 Update Tenant Page

- Delete Tenant



Tenant

Delete Tenant

Are you sure you want to delete this user?

Delete Close

Tenants

Show 10 entries

#	Name	Email Address	NRIC/IC	Phone Number	Current Address	Action
7	Rohana Mokhtar	rohanamokhtar68@gmail.com	690324056798	01127503353	134, Jln Melawati 5, Seksyen13 Shah Alam, 40100 Selangor.	
8	Muhamammad Hasif B Samsuri	hasifsamsuri@gmail.com	960721056466	0177520457	23,Jalan Pertama Indah, Kulai Jaya, Kulai, 84900 Johor	

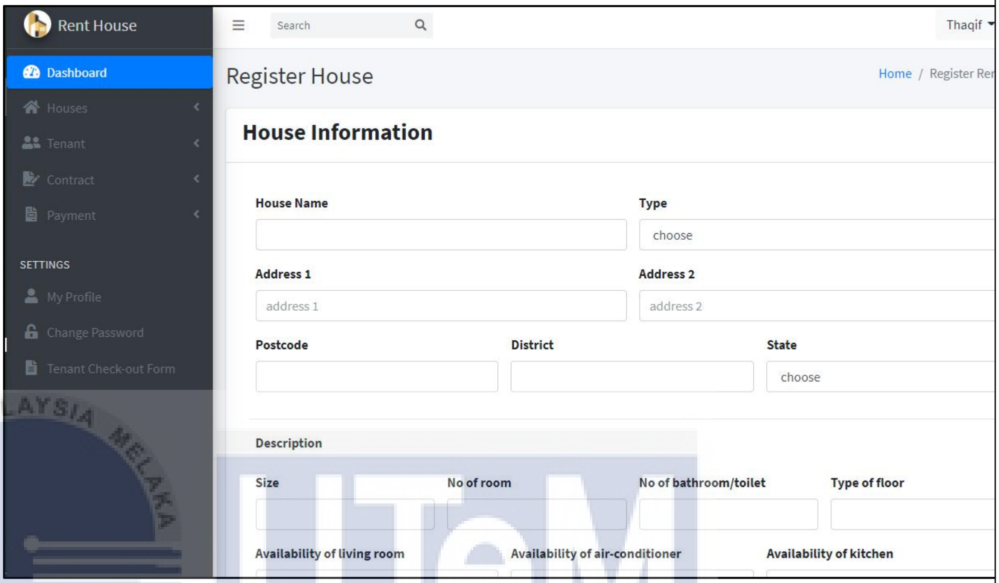
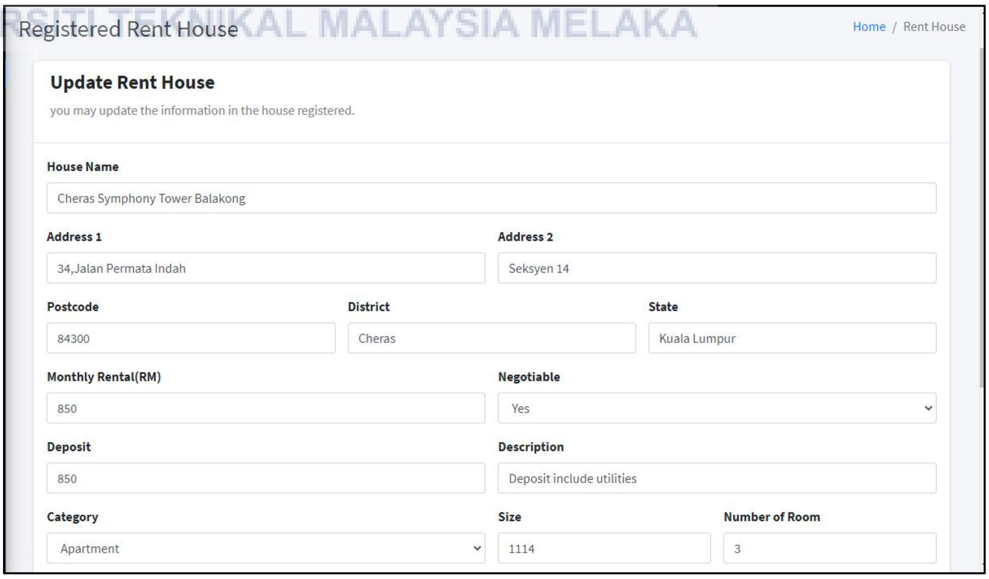
Showing 1 to 2 of 2 entries

Figure 4.50 Delete Tenant Page

4.3.1.4 Manage House

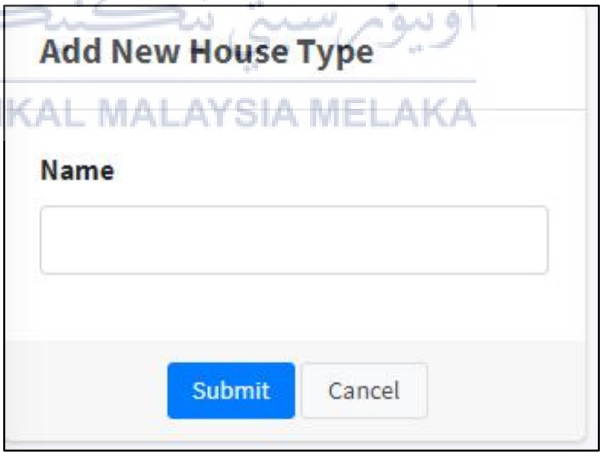
Program Name	**HRAS_004**
Description	- Sent and receive information of house and save into the database. - Landlord can add, update, delete house in the system as shown on Figure 4.51 and 4.52.
Input/Output	- Add House Input: house name, address1,address2, address3, postcode, district, state, monthlypaid,negotiable, deposit, category, noRoom, noToilet, wifi, kitchen, CCTV, furniture, gate. Output: The information of house successfully inserted in the manage house page. - Delete House: Output: The information of house has successfully delete in the manage

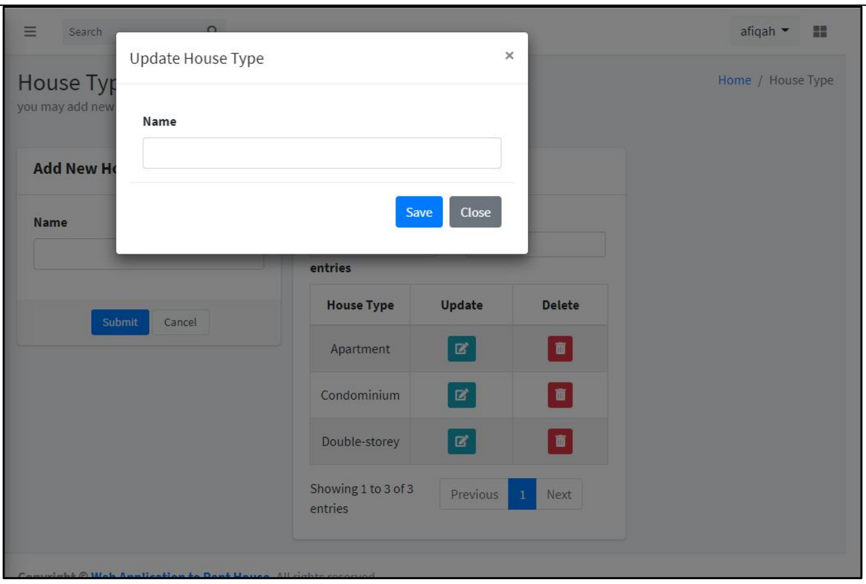
	house page. - Update House: Input: house name, address1,address2, address3, postcode, district, state, monthlypaid,negotiable, deposit, category, noRoom, noToilet, wifi, kitchen, CCTV, furniture, gate Output: The information of house successfully updated in the manage house page.
Pseudo code	- Add House Step 1: Click button "Add House" to add house in the system. Step 2: Fill in the needed inputs(housename, address1,address2, address3, postcode, district, state, monthlypaid,negotiable, deposit, category,noRoom, noToilet, wifi, kitchen, CCTV, furniture, gate). Step 3: Click button "Save" to add house to the system. - Delete House Step 1:Click the icon "delete" and display a confirmation message to delete the house from the system. Step 2: Click button "Delete" to delete the house. - Update House Step 1: Click icon "update" to update house in the system. Step 2: Fill in the needed inputs(house name, address1,address2, address3, postcode, district, state, monthlypaid,negotiable, deposit,

	category, noRoom, noToilet, wifi, kitchen, CCTV, furniture, gate). Step 3: Click the button “Save” to update the house.
Screen format	- Add House  Rent House Dashboard Houses Tenant Contract Payment SETTINGS My Profile Change Password Tenant Check-out Form Search Thaqif Home / Register Ren Register House House Information House Name Type: choose Address 1: address 1 Address 2: address 2 Postcode District State: choose Description Size No of room No of bathroom/toilet Type of floor Availability of living room Availability of air-conditioner Availability of kitchen **Figure 4.51 Add House Information** - Update House  Registered Rent House Update Rent House you may update the information in the house registered. House Name: Cheras Symphony Tower Balakong Address 1: 34,Jalan Permata Indah Address 2: Seksyen 14 Postcode: 84300 District: Cheras State: Kuala Lumpur Monthly Rental(RM): 850 Negotiable: Yes Deposit: 850 Description: Deposit include utilities Category: Apartment Size: 1114 Number of Room: 3 **Figure 4.52 Update House Information**

4.3.1.5 Manage Category

Program Name	**HRAS_005**
Description	- Saves and receives information of category and save into the database. - Admin can add, update, and delete category in the system as shown on Figure 4.53, 4.54 and 4.55.
Input/Output	- Add Category Input: name Output: The information of category has successfully inserted in the manage category page. - Update Category Input: name Output: The information of category has successfully updated in the manage category page. - Delete Category Input: name Output: The information of category has successfully deleted in the manage category page.
Pseudo code	- Add Category Step 1: Click the option "HouseType" to add category in the system. Step 2: Fill in the needed inputs(name).

	Step 3: Click the button “Save” to add category to the system. - Update Category Step 1: Click the icon “Update” to edit the category in the system. Step 2: Fill in the needed inputs(name). Step 3: Click the button “Save” to update category to the system. - Delete Category Step 1: Click the icon “Delete” to delete the category in the system. Step 2: The system will prompt confirmation message to delete category. Step 3: Click the button “Delete” to delete category to the system.
Screen format	- Add Category Add New House Type Name **Figure 4.53 Add Category House** - Update Category



Update House Type

Name

entries

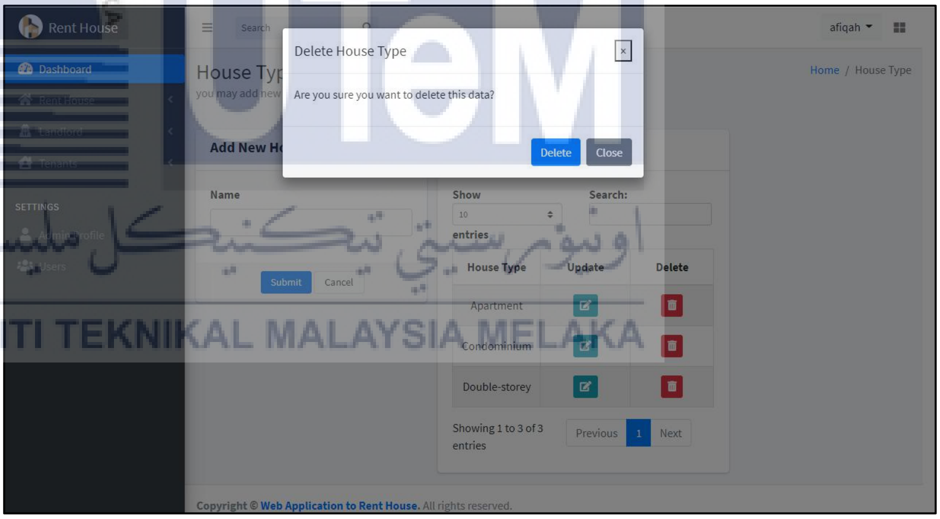
House Type	Update	Delete
Apartment	✎	🗑
Condominium	✎	🗑
Double-storey	✎	🗑

Showing 1 to 3 of 3 entries

Previous 1 Next

Figure 4.54 Update Category House

- Delete Category



Delete House Type

Are you sure you want to delete this data?

Show 10 entries Search:

House Type	Update	Delete
Apartment	✎	🗑
Condominium	✎	🗑
Double-storey	✎	🗑

Showing 1 to 3 of 3 entries

Previous 1 Next

Copyright © Web Application to Rent House. All rights reserved.

Figure 4.55 Delete Category House

4.3.2 Physical Database Design

Physical database design is a logical data model that works as SQL statement in the database. It gives functionalities within the system on display, manipulate, and insert data.

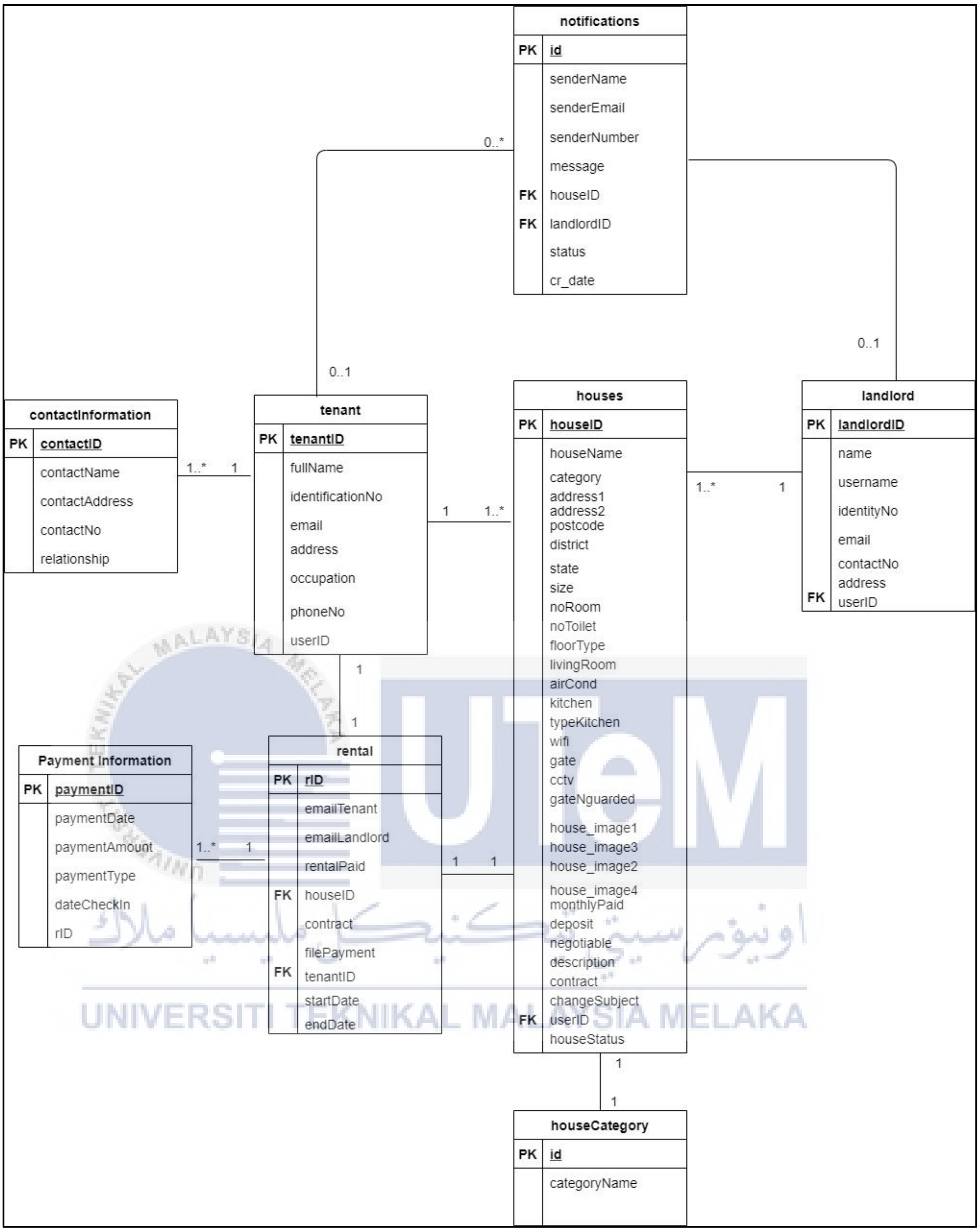


Figure 4.56 Physical design ERD of HRAS

4.3.2.1 Data Definition Language(DDL)

The figure below shows the syntax of creating database table. List of syntax for the table in the database as shown below:

1. User

```
CREATE TABLE `users` (
  `userID` int(11) NOT NULL,
  `fullName` varchar(255) NOT NULL,
  `username` varchar(50) NOT NULL,
  `email` varchar(100) NOT NULL,
  `phoneNo` varchar(100) NOT NULL,
  `password` varchar(50) NOT NULL,
  `role` varchar(20) NOT NULL,
  `profileImage` varchar(100) NOT NULL
);
```

2. Landlord

```
CREATE TABLE `landlord` (
  `landlordID` int(11) NOT NULL,
  `name` varchar(150) NOT NULL,
  `username` varchar(50) NOT NULL,
  `identityNo` varchar(50) NOT NULL,
  `email` varchar(100) NOT NULL,
`contactNo` varchar(50) NOT NULL,

  `address` varchar(150) NOT NULL,

  `userID` int(11) NOT NULL

) ;
```

3. Tenant

```
CREATE TABLE `tenant` (

  `tenantID` int(11) NOT NULL,

  `fullName` varchar(100) NOT NULL,

  `email` varchar(50) NOT NULL,

  `identificationNo` varchar(20) NOT NULL,

  `occupation` varchar(50) NOT NULL,

  `phoneNo` varchar(20) NOT NULL,

  `address` varchar(90) NOT NULL,

  `userID` int(11) NOT NULL

);
```

4. House

```
CREATE TABLE `houses` (
  `houseID` int(11) NOT NULL,
  `houseName` varchar(30) NOT NULL,
  `category` varchar(20) NOT NULL,
  `address1` varchar(30) NOT NULL,
  `address2` varchar(30) NOT NULL,
  `postcode` int(11) NOT NULL,
  `district` varchar(20) NOT NULL,
  `state` varchar(20) NOT NULL,
  `size` decimal(10,0) NOT NULL,
  `noRoom` int(11) NOT NULL,
  `noToilet` int(11) NOT NULL,
  `floorType` varchar(20) NOT NULL,
  `livingRoom` varchar(20) NOT NULL,
  `airCond` varchar(20) NOT NULL,
  `kitchen` varchar(20) NOT NULL,
  `typeKitchen` varchar(20) NOT NULL,
  `wifi` varchar(20) NOT NULL,
  `furniture` varchar(20) NOT NULL,

  `gate` varchar(20) NOT NULL,

  `cctv` varchar(20) NOT NULL,

  `gateNguarded` varchar(20) NOT NULL,

  `house_image1` varchar(30) NOT NULL,

  `house_image2` varchar(30) NOT NULL,

  `house_image3` varchar(30) NOT NULL,

  `house_image4` varchar(30) NOT NULL,

  `monthlyPaid` decimal(10,0) NOT NULL,

  `negotiable` varchar(20) NOT NULL,

  `deposit` decimal(10,0) NOT NULL,

  `description` varchar(50) NOT NULL,

  `contract` varchar(30) NOT NULL,

  `changeSubject` varchar(30) NOT NULL,

  `userID` int(11) NOT NULL,

  `houseStatus` varchar(11) NOT NULL

);
```

5. House Type

```
CREATE TABLE `housecategory` (
  `id` int(11) NOT NULL,
  `categoryName` varchar(50) NOT NULL
) ;
```

6. Tenant’s Contact Information

```
CREATE TABLE `contactinformation` (
  `contactID` int(11) NOT NULL,
  `contactName` varchar(100) NOT NULL,
  `email` varchar(80) NOT NULL,
  `contactNo` varchar(15) NOT NULL,
  `address` varchar(100) NOT NULL,
  `occupation` varchar(50) NOT NULL,
  `relationContact` varchar(20) NOT NULL,
  `tenantID` int(11) NOT NULL
) ;
```

7. Notifications

```
CREATE TABLE `notifications` (

  `id` int(11) NOT NULL,

  `senderName` varchar(100) NOT NULL,

  `senderEmail` varchar(100) NOT NULL,

  `senderNumber` varchar(15) NOT NULL,

  `message` text NOT NULL,

  `landlordID` int(11) NOT NULL,

  `houseID` int(11) NOT NULL,

  `status` int(11) NOT NULL DEFAULT 0,

  `cr_date` timestamp NOT NULL DEFAULT

 current_timestamp() ON UPDATE current_timestamp()

) ;
```

8. Rental

```
CREATE TABLE `rental` (

  `rID` int(11) NOT NULL,

  `emailTenant` varchar(50) NOT NULL,

  `emailLandlord` varchar(30) NOT NULL,
  `rentalPaid` decimal(10,0) NOT NULL,

  `deposit` decimal(10,0) NOT NULL,

  `userID` int(11) NOT NULL,

  `houseID` int(11) NOT NULL,

  `contract` varchar(50) NOT NULL,

  `filePayment` varchar(50) NOT NULL,

  `tenantName` varchar(100) NOT NULL,

  `tenantID` int(11) NOT NULL,

  `startDate` date NOT NULL,

  `endDate` date NOT NULL,

  `Date` timestamp NOT NULL DEFAULT

current_timestamp() ON UPDATE current_timestamp()

) ;
```

4.4 Conclusion

In conclusion, the designing phase can help to give an overview of the process involved in HRAS in the analysis phase. The design of HRAS can be divided into architecture, physical and logical design. The entity relationship diagram in database design is used to show the relationship between the entities in the HRAS table. A data dictionary is used to describe the information about each entity's attributes.

CHAPTER 5: IMPLEMENTATION

5.1 Introduction

The implementation phase of the Web Application to Rent House System will be the emphasis of this chapter. The system environment that includes database, are setup for the execution of the implementation phase. The implementation phase is able to proceed when the system environment setup has been done. The process of developing the system will be extensively explained in this phase, including installation, configuration, deployment, and testing to ensure that it meets the requirements.

5.2 Software Development Environment Setup

This section explains how to setup the system environment for the process of developing the system. Visual Studio is a programming tool that helps with the development process by allowing the user to integrate any extension into the visual studio to improve the user's coding experience that helps writing and debugging code easier. This chapter will go over the specifics of the software that was used throughout the project's development.

5.2.1 Windows 10 Operating System

The Windows operating system is one of the most user-friendly systems as it is most commonly used among users. The Windows 10 Operating System is the latest version of the Windows system which supports a wider range of software and hardware options that are compatible with the programmer's needs. The OS version that was used to develop the system is Windows 10 Home Single Language, which is able to carry out most of the system development process. Aside that, Windows OS also offers access to development Integrated Environment (IDE) through Visual Studio Community.

5.2.2 Xampp Version 8.0.3

Xampp is an open source programme that allows developers to test their website or system locally before deploying it to the main server. It's also a platform for testing PHP, MySQL, and Apache-based projects using the host's own infrastructure. Xampp Control Panel version 8.0.3 is the Xampp version used to design the system.

5.2.3 Visual Studio Version 1.58.2

Microsoft Visual Studio is a Microsoft integrated development environment (IDE) for developing computer programme such as websites, web apps, online services, and mobile apps. For students and individual developers, Visual Studio Community Edition provides a free source and fully-featured IDE. Microsoft Visual Studio also can be used across to any platform such as Linux, mac OS and Windows. It is also an effective development environment (IDE) which it can be use to edit, debug, build code, publish an app and it also provide extension from an outside source to be added to allow the ease of the development process.

5.3 Software Configuration Management

This section explains the procedure of the software configuration on the system and life cycle of the software.

5.3.1 Configuration Environment Setup

The Web Application to Rent House System requires XAMPP to deploy the system on the local server into the browser. With XAMPP acting as the local server, the system also requires a database that acts as storage to store data on the system. The Web Application to Rent House System uses the MySQL database. Other than that, the XAMPP is configured to allow the system to use the built-in PHP Mail function to send email from the local server to users through Gmail. With the configuration, it allows the users to notify the landlord, send the attachment, and handle the renting process on the system.

5.3.2 Version Control Procedure

The procedure for tracking and managing the changes made to the HRAS is described in the version control procedure section. In this section, version control is the programming process that manages the changes made during the development process. It will be handled manually with the file name based on the respective version change. The table below shows the version control for HRAS.

Table 5.1 Version of HRAS

Version	Author	Description
1.0	Nurul Afiqah Binti Samsuri	The first prototype of the system was reviewed by the supervisor.
1.1	Nurul Afiqah Binti Samsuri	Correcting the system based on the comments made by the supervisor.
1.2	Nurul Afiqah Binti Samsuri	Change the house advertisement interface and change the flow of the rental process to enhance user interaction.
1.3	Nurul Afiqah Binti Samsuri	Improve the previous version by correcting the errors in a few modules to enhance the efficiency of the system.
1.4	Nurul Afiqah Binti Samsuri	Testing most of the modules on the HRAS system. The version includes verification and error handling.
1.5	Nurul Afiqah Binti Samsuri	The final touch up of the system. The whole module of the system is tested and improves the functionality of the system.

5.4 Implementation Status

The implementation status is the duration for each module to complete. The table below shows the system implementation status for HRAS.

Table 5.2 Implementation Status of HRAS

Module Name	Description	Duration to Complete	Month Complete
Interface Design	Design the interface for the HRAS system.	5 days	February 2021
Database Design and configuration	Design the relationship between the entities in and setup the connection into the system.	5 days	March 2021
Login and administrator	Login for three role which is admin, landlord and tenant.	7 days	March 2021
Landlord Information Management	Allow the admin to add new landlord information in the system.	7 days	April 2021
Tenant Information Management	Allow the admin to add new tenant information in the system.	7 days	April 2021
House Management	Allow the landlord to create and manage their house in the system.	7 days	May 2021

House Advertisement	Allow the admin to manage the house advertisement in the system.	14 days	June 2021
Contract Management	Allow the landlord to create and manage the contract between the tenant in the system.	7 days	June 2021
Rental Management	Allow the landlord to manage the renting process of the house between the tenant in the system.	14 days	July 2021
Payment Management	Allow the tenant manage the monthly rental payment by paying online through the system.	14 days	August 2021

5.5 Conclusion

This chapter covers the system implementation procedure, as well as the tools that were stated in the previous part. It described how the system is implemented using the tool mentioned, as well as the time it takes for each module to complete. Lastly, the next chapter will focus on the testing phase in order to create a test plan for the Web Application to Rent House System's testing process.

CHAPTER 6: TESTING

6.1 Introduction

The testing phase of the development system will be discussed in this chapter. The testing phase evaluates and tests the system's requirements, features, and expectations to ensure that the system meets the project's requirements. It will test the entire system in this chapter, including the functional and non-functional requirements that were defined in the previous chapter. A test plan, strategy, result, and analysis will all be included in every test. The system's testers will be focused on the users who are stated in the project scope based on their roles. The white-box and black-box testing approaches will be used to test the HRAS system.

6.2 Test Plan

This section will describe how the test plan is implemented. It will define the scope of testing for the system's product. This test plan will assist in checking every detail of the developed product and ensuring its efficiency. The test plan will be explained in more detail below.

6.2.1 Test Organization

There will be two roles in this section: one for the test manager and one for the tester as shown on table 6.1. The test manager is in charge of ensuring that the project development goes off without a hitch. As a result, the tester will test the system's interface and functionality that have been developed. The testing results will be recorded for future improvements.

Table 6.1 Test Organization

Tester ID	Name	Roles
Test_01	Nurul Afiqah Binti Samsuri	Test Manager
Test_02	Nur Suhaili Binti Haruddin	Test User

6.2.2 Test Environment

The testing was conducted online on the developer's laptop. Table 2.2 of chapter 2 lists of software requirements specifications. The list of software required for the testing phase uses the same environment as the implementation phase, which is shown in table 6.2.

Table 6.2 Testing Specification Software

Component	**Description**
Operating System	Windows 10 Home Single Language
Database	MySQL Version 8.0.3
Development tool	Microsoft Visual Studio Version 1.58.2

6.2.3 Test Schedule

The table 6.3 below shows the duration of completing the testing process. The modules are categorized based on their scope and are given a unique identifier to be organized easily.

Table 6.3 Test Schedule

Test Case	**Total Module**	**Duration**
Test case for administrator	7	1 Day
Test case for landlord	9	2 Day
Test case for tenant	5	1 Day

6.3 Test Strategy

The strategy used in the testing phase will be explained in this section. Dynamic testing is the testing phase that will be used in this technique. Dynamic testing is a technique for evaluating the behaviour of software code that operates dynamically in response to human input entered into the system during the testing process. Based on table 6.3, white-box and black-box testing approaches are used in dynamic testing. This function is required as it will assess the Web Application to Rent House System.

Table 6.3 Black-Box and White-Box Testing

Type of method	**Description**
Black-box testing	A testing method in which the internal structure is not known to the tester. Black box testing is used to verify the functionality of the system, which includes functional and non-functional testing. The testing can be done by the tester and does not require any programming language.
White-Box Testing	The internal structure of the system is known to the tester during the testing process. White box testing is a technique for determining how well a system performs based on its code. The tester is required to have knowledge of programming as it will test the logic of the system.

6.3.1 Classes of Test

It is performed out to assure that the system's functionalities are operating as expected. The function and user input will be examined by the tester. The next part will go over all of the relevant test cases in detail.

Aside that, non-functionality testing, such as security testing, is performed on the system. It is to ensure the confidentiality of the data inside the system. It will check whether the system is vulnerable to attack by a hacker or third-party. The attack may come from various ways when there are vulnerabilities in the system which can be exploited by the hacker to gain access to the system or do any destruction to the system.

6.4 Test Design

This section will show the process of identifying the test case for each module in the test description. To obtain an accurate result, the correct and incorrect data are prepared and the result will be recorded on the test case.

6.4.1 Test Description

In test description section, each test will have a unique identifier, description and the expected result for each module. This project has designed different test cases for each role in HRAS. There are 21 number of test cases for administrator. The details of the test cases of the administrator is described in table 6.4.

Table 6.4 Test Case for Administrator

Module	Test Case ID	Description	Expected Result
Login	AD001_01	To authenticate user credential when logged into the system with the correct email and password	System will direct the user to dashboard page
	AD001_02	To authenticate the user credential when logged into the system with the incorrect email and password	System will display an error message of 'Invalid email or password'
	AD001_03	To authenticate the user when any of the field is left blank	System will ask the user to fill out the blank field

House	AD002_01	To authenticate user can view list of houses	System display the list of houses
	AD002_02	To authenticate user can add new house by fill in the required field and click on ‘Add New House’ button	System will display the message of ‘New house added’
	AD002_03	To authenticate user can update by click on edit icon	System will display the message of ‘House updated successfully ’
	AD002_04	To authenticate user can delete by click on delete icon	System will display the message of ‘House deleted successfully’
House Category	AD003_01	To authenticate user can view list of house category	System will display list of house category
	AD003_02	To authenticate user can add new house category by fill in the required field and click on ‘submit’ button	System will display message of ‘House category added successfully’
	AD003_03	To authenticate user can update by click on edit icon	System will display message of ‘House category updated successfully’
	AD003_04	To authenticate user can delete by click on delete icon	System will display message of ‘House Category deleted successfully’

Landlord	AD004_01	To authenticate user can view list of landlord	System will display list of landlords
	AD004_02	To authenticate user can add new landlord by fill in the required field and click on ‘Add Landlord’ button	System will display the message of ‘New Landlord added successfully’
	AD004_03	To authenticate user can update landlord details by click on edit icon	System will display message of ‘Landlord update is successful ’
	AD004_04	To authenticate user can delete landlord by click on delete icon	System will display message of ‘Landlord delete is successful’
Tenant	AD005_01	To authenticate user can view list of tenant	System will display list of tenants
	AD005_02	To authenticate user can add new tenant by fill in the required field and click on ‘Add Tenant’ button	System will display the message of ‘New tenant added successfully’
	AD005_03	To authenticate user can update tenant details by click on edit icon	System will display message of ‘Tenant update is successful ’
	AD005_04	To authenticate user can delete tenant by click on delete icon	System will display message of ‘Tenant delete is successful’
Admin Profile	AD006_01	To authenticate user can	System will display the

		view their information	user information
Users	AD007_01	To authenticate user can view list of register users	System will display list of register users
	AD007_02	To authenticate user can add new users by fill in the required field and click on 'Add New User' button	System will display the message of 'New user added successfully'
	AD007_03	To authenticate user can update users details by click on edit icon	System will display message of 'User update is successful '
	AD007_04	To authenticate user can delete user by click on delete icon	System will display message of 'User delete is successful'

There are 25 number of test cases for Landlord. The details of the test cases of the landlord is described in table 6.5.

Table 6.5 Test Case for Landlord

Module	**Test Case ID**	**Description**	**Expected Result**
Login	S001_01	To authenticate user credential when logged into the system with correct email and password	System will direct the user to dashboard page
	S001_02	To authenticate user credential when incorrect email or password	System will display an error message of 'Invalid email or password'

	S001_03	To authenticate user when any field is left blank	System will ask the user to fill in all the blank field
Profile	S002_01	To authenticate user can view personal details	System will display personal details
	S002_02	To authenticate user can update details by click on the 'Save' button	System will display message of 'Data successfully inserted'
Change Password	S003_01	To authenticate if user want to change password	System will ask user to insert the old password
	S003_02	To authenticate user if user password reset is successful	System will direct user to login page
	S003_03	To authenticate user if user password reset is unsuccessful	System will display an error message 'Password does not match!'
House	S004_01	To authenticate user can view the list of house had been register	System will display list of houses
	S004_02	To authenticate user can register new house by fill in the required field and click on 'Send' button	System will display message of 'New house successful added'
	S004_03	To authenticate user can update house details by click on update icon	System will display message of 'Update house successful'

	S004_04	To authenticate user can delete house by click on delete icon	System will display message of 'Delete house successful'
Response wish list	S005_01	To authenticate user can view all the list of application to rent house	System will display the list of notification
	S005_02	To authenticate user can click on accept radio button to accept the application	System will direct the user to property page and sending the email to the applicant with the contract on the attachment.
	S005_03	To authenticate user can click on reject radio button to reject the application	System will direct the user to reject response page and sending the reject email to teh applicant.
Tenant	S006_01	To authenticate user can view list of tenant	System will display the list of tenant
	S006_02	To authenticate user can view list of tenant's contact	System will display list of tenant contact
Contract	S007_01	To authenticate user can view list of agreement that had been upload by tenant	System will display list of agreement
	S007_02	To authenticate user can upload the signed contract and set the rental period by	System will direct the user to upload contract and will update status of

		specify start date and end date by click on 'action' button	the house into unavailable
Move-In Move-Out Response	S008_01	To authenticate user can view list of checklist response had been upload by the tenant	System will display the checklist form details

There are 14 number of test cases for tenant. The details of the test cases of the tenant is described in table 6.6.

Table 6.6 Test Case of Tenant

Module	**Test Case ID**	**Description**	**Expected Result**
Login	T001_01	To authenticate user credential when logged into the system with correct email and password	System will direct user to dashboard page
	T001_02	To authenticate user credential when logged into the system with incorrect email and password	System will display an error message of 'Invalid email or password'
	T001_03	To authenticate user when any field is left bank	System will ask user to fill out the blank field
Profile	T002_01	To authenticate user can view personal details	System will display personal details with the contact information

	T002_02	To authenticate user can update detail by click on 'save' button	System will display message of 'Updated successful'
Contact Information	T003_01	To authenticate user can 'add contact info'	System will display message of 'Contact information added successful'
Contract	T004_01	To authenticate user can view the rental agreement	System will display the rental agreement
	T004_02	To authenticate user can upload the signed rental agreement and proof of payment and send the email to the landlord	System will display message of 'Contract upload successful
Checklist Form	T005_01	To authenticate user can download the checklist form	System will display the checklist form
	T005_02	To authenticate user can upload checklist form	System will display message of 'your checklist form had been successfully sent'
Rental Payment	T006_01	To authenticate user can view rental payment information	System will display the monthly rental fee
	T006_02	To authenticate user can 'Pay Now' the rental fee	System will display the rental fee details
	T006_03	To authenticate user can	System will accept

		submit payment	payment and display “The mail have been sent to rohanamokhtarr69@gmail.com”

6.4.2 Test Data

In this section, it will explain how the system will respond when the correct or incorrect data is inserted or left blank into the system. The purpose of test data are to test the system's ability to handle unusual, exceptional, and unexpected inputs. We have designed different test data for each role in HRAS. The test data to be used with the test cases that was designed for Administrator is described in table 6.7.

Table 6.7 Test Data for Administrator

Test Case ID	Description	Steps
AD001_01	Email:fieqasamsuri14@gmail.com Password:12345	1. Fill in the required field with given data 2. Click ‘login’ button
AD001_02	Email:fieqasamsuri14@gmail.com Password: 12345uio (incorrect data)	
AD001_03	Email: (left blank) Password: 12345uio (incorrect data)	
AD002_01	Category Name: Terrace	1. Click ‘House’ Page 2. Fill in the required field with given data 3. Click ‘Add/Update/Delete’
AD002_02	Category Name: Terrace <u>UPDATE</u>	

AD002_03	Category Name: Terrace DELETE	button
AD003_01	House Name: Cheras Symphony Tower Balakong Type: Terrace Address 1: 34, Jalan Pertama Indah Address 2: Seksyen 14 Postcode: 84300 District: Cheras State: Kuala Lumpur	1. Click 'House' page 2. Click 'Register New House' page 3. Fill in the required field with given data 4. Click 'Add/Update/Delete' Button
AD003_02	House Name: Cheras Symphony Tower Balakong Type: Terrace Address 1: 34, Jalan Pertama Indah Address 2: Seksyen 14 Postcode: 84300 District: Cheras State: Kuala Lumpur UPDATE	
AD003_03	House Name: Cheras Symphony Tower Balakong Type: Terrace Address 1: 34, Jalan Pertama	

	Indah Address 2: Seksyen 14 Postcode: 84300 District: Cheras State: Kuala Lumpur <u>DELETE</u>	
AD003_04	House Name: Cheras Symphony Tower Balakong Type: Terrace Address 1: 34, Jalan Pertama Indah Address 2: Seksyen 14 Postcode: 84300 District: Cheras State: (left blank)	
AD004_01	Name: Nur Suhaili Username:Lily NRIC/IC:980614016760 Email :nursuhaili98@gmail.com Contact Number: 01137360403 Address: No 6, Jalan Seri Austin 3/6 Taman Seri Austin Hills, 81100 Johor Bahru, Johor	
AD004_02	Name: Nur Suhaili	1. Click ‘Landlord’ page

	Username:Lily NRIC/IC:980614016760 Email :nursuhaili98@gmail.com Contact Number: 01137360403 Address: (left blank)	2. Click 'Add Landlord' button 3. Fill in the required field with given data 4. Click 'Add/Update/Delete' button
AD004_03	Name: Nur Suhaili Username:Lily NRIC/IC:980614016760 Email :nursuhaili98@gmail.com Contact Number: 01137360403 Address: No 6, Jalan Seri Austin 3/6 Taman Seri Austin Hills, 81100 Johor Bahru, Johor UPDATE	
	Name: Nur Suhaili Username:Lily NRIC/IC:980614016760 Email :nursuhaili98@gmail.com Contact Number: 01137360403 Address: No 6, Jalan Seri Austin 3/6 Taman Seri Austin Hills, 81100 Johor Bahru, Johor DELETE	
AD005_01	Name: Nur Anisah Hadirah	1. Click 'Tenant' Page

	Email:anisahadirah98@gmail.com NRIC/IC: 981206019860 Phone number: 0149896375 Current address:62, Jalan Wau Kikik 7, Prima Langkasa, 81700 Pasir Gudang, Johor	2. Click ‘Add Tenant’ button 3. Fill in the required field with given data 4. Click ‘Add/Update/Delete’ button
AD005_02	Name: Nur Anisah Hadirah Email:anisahadirah98@gmail.com NRIC/IC: (left blank) Phone number: 0149896375 Current address:62, Jalan Wau Kikik 7, Prima Langkasa, 81700 Pasir Gudang, Johor	
AD005_03	Name: Nur Anisah Hadirah Email:anisahadirah98@gmail.com NRIC/IC: 981206019860 Phone number: 0149896375 Current address:62, Jalan Wau Kikik 7, Prima Langkasa, 81700 Pasir Gudang, Johor <u>UPDATE</u>	
AD005_04	Name: Nur Anisah Hadirah Email:anisahadirah98@gmail.com NRIC/IC: 981206019860 Phone number: 0149896375	

	Current address:62, Jalan Wau Kikik 7, Prima Langkasa, 81700 Pasir Gudang, Johor DELETE	
AD006_01	Name: Rohana Mokhtar Username:Rohana Email Address: rohana 67@gmail.com Phone Number: 01127503353 Password: abc12345 Confirm Password: abc12345 User Role: Tenant	1. Click ‘Users’ page 2. Click ‘Add New User’ button 3. Fill in the required field with given data 4. Click ‘Add/Update/Delete’ button
AD006_02	Name: Rohana Mokhtar Username:Rohana Email Address: rohana 67@gmail.com Phone Number: 01127503353 Password: abc12345 Confirm Password: (left blank) User Role: Tenant	
AD006_03	Name: Rohana Mokhtar Username:Rohana Email Address: rohana 67@gmail.com	

	Phone Number: 01127503353 Password: abc12345 Confirm Password: abc12345 User Role: Tenant UPDATE	
AD006_04	Name: Rohana Mokhtar Username:Rohana Email Address: rohana 67@gmail.com Phone Number: 01127503353 Password: abc12345 Confirm Password: abc12345 User Role: Tenant DELETE	

The test data to be used with the test cases that was designed for the landlord is described in table 6.8.

Table 6.8 Test Data for Landlord

Test Case ID	**Description**	**Steps**
S001_01	Email: fieqasamsuri14@gmail.com Password: ik@12345	1. Fill in the required field with given data 2. Click ‘login’ button
S001_02	Email: fieqasamsuri14@gmail.com	

	Password: ik@1234578 (Incorrect data)	
S001_03	Email: fieqasamsuri14@gmail.com Password: (left blank)	
S002_01	Update landlord details Name: Nur Suhaili Username: Suhaili Contact Number: 01137360403 Email:nursuhaili98@gmail.com Identification No/IC: 980609016780 Address: No 6, Jalan Seri Austin 3/6 Taman Seri Austin Hills, 81100 Johor Bahru, Johor	1. Click on 'My Profile' page 2. Fill in the required field with given data 3. Click 'Save' button
S003_01	New House Data(Correct data) House Name: Cheras Symphony Tower Balakong Type: Terrace Address 1: 34, Jalan Pertama Indah Address 2: Seksyen 14 Postcode: 84300 District: Cheras State: Kuala Lumpur	1. Click 'House' page 2. Click 'Register New House' page 3. Fill in the required field with given data 4. Click 'Add/Update/Delete' Button
S003_02	New House Data	

	House Name: Cheras Symphony Tower Balakong Type: Terrace Address 1: 34, Jalan Pertama Indah Address 2: Seksyen 14 Postcode: 84300 District: Cheras State: (left blank)	
S003_03	House Name: Cheras Symphony Tower Balakong Type: Terrace Address 1: 34, Jalan Pertama Indah Address 2: Seksyen 14 Postcode: 84300 District: Cheras State: Kuala Lumpur (update data) <u>UPDATE</u>	
S003_04	House Name: Cheras Symphony Tower Balakong Type: Terrace Address 1: 34, Jalan Pertama Indah Address 2: Seksyen 14 Postcode: 84300	

	District: Cheras State: Kuala Lumpur (update data) DELETE	
S004_01	Response Application Sender Name: Nur Alia Sender Email: aliasyahirah97@gmail.com Sender Phone No: 0143567890 Message:Hi, please contact me Date: 19-08-2021 ACCEPT	1. Click on notification icon. 2. System will direct the user to response wishlist page. 3. Click on 'Accept' radion button. 4. System will direct the user to Property page. 5. Add the contract on the 'attachment'. 6. Click on 'Save' button
S004_02	Accept Response Tenant Name: Nur Alia Monthly Rental: RM 850 Deposit: RM 850 Tenant Email: aliasyahirah97@gmail.com Subject: Responses to Rental House Applications Contract and Agreement: TenantContract.pdf Message:Thank you for your inquiry about renting house for Cheras Symphony Tower Balakong. SAVE	

S005_01	Upload Signed Contract Start Date: 2021-08-25 End Date: 2022-08-26 File: TenantContact.pdf Rent To Tenant	1. Click 'Contract' page 2. Click 'Action' button 3. Fill in the required field with given data 4. Click 'Rent To Tenant' button
S006_01	Current password: 12345678 New password: qwerty67 Confirm password: qwerty679 (incorrect data)	1. Click on Change Password Page 2. Fill in the required field with given data 3. Click 'Save' button 4. Login again
S006_02	Current password: 12345678 New password: qwerty67 Confirm password: (left blank)	
S006_03	Current password: 12345678 New password: qwerty67 Confirm password: qwerty67	

The test data to be used with the test cases that was designed for the tenant is described in table 6.9.

Table 6.9 Test Data for Tenant

Test Case ID	Description	Steps
T001_01	Email: (left blank) Password: ik@12345	1. Login page 2. Fill in the required field with given data

T001_02	Email: (left blank) Password: ik@1234578 (incorrect data)	3. Click 'login' button
T001_03	Correct data Email: afiqahsamsuri988@gmail.com Password: ik@12345	
T002_01	Update Personal Details Name: Nurul Afiqah Email Address: afiqahsamsuri988@gmail.com NRIC/IC: 980814015790 Occupation: Student Contact Number: 0134387250 Address: 310. Jalan Cenderawasih 19, KKTDI 5, 84000 Muar, Johor	1. Click on 'Personal Information' Page 2. Field in the required field with given data 3. Click 'Save' button
T003_01	New Contact Name: Muhammad Hasif Mobile Number: 01128697649 Occupation: Technician Email: hasif96@gmail.com Address: (left blank) Relationship: Sibling	1. Click 'Profile' page 2. Click on 'Contact Information' page 3. Fill in the required field with given data 4. Click on 'Save' button
T003_02	New Contact (correct data)	

	Name: Muhammad Hasif Mobile Number: 01128697649 Occupation: Technician Email: hasif96@gmail.com Address: 310, Jalan Cenderawasih 19, KKTDI 5, 84000 Muar, Johor Relationship: Sibling	
T003_03	Name: Muhammad Hasif Mobile Number: 01128697649 Occupation: Technician Email: hasif96@gmail.com Address: 310, Jalan Cenderawasih 19, KKTDI 5, 84000 Muar, Johor Relationship: Sibling CANCEL	
T004_01	Upload Files Signed contract: TenantContarct.pdf Proof of Payment: Receipt.pdf	1. Click on 'Contract' page 2. Click on 'Contract Management' page 3. Fill in the required field with given data 4. Click 'Save' button
T005_01	Upload Checklist Form	1. Click on 'Checklist Form' page

	Name: Nurul Afiqah Subject:Responses Checklist Form Message: Please repair asap Attachment: checklistForm.pdf	2. Click on 'Upload Checklist Form' page 3. Fill in the required field with given data 4. Click 'Send' page
T006_01	Current password: ik@12345 New password: ik@12345gh Confirm password: (incorrect data)	1. Click on 'Change Password' page 2. Fill in the required field with given data. 3. Click 'Save' button 4. Login again
T006_02	Current password: 12345678 New password: qwerty67 Confirm password: (left blank)	
T006_03	Current password: 12345678 New password: qwerty67 Confirm password: qwerty67	

6.5 Test Result and Analysis

In this section, all the results from the test case will be recorded. The test case would be considered as 'failed' if any of the test cases did not pass the test. And if it passes the test, it will be considered as 'success'. The test results and analysis are necessary to ensure that the system operates normally on every user action.

This project has carried out 60 test cases. This project has executed 21 number of test cases for administrator's role in HRAS using 21 number of data. 21 test cases met the expected results. The detail of the test result for Administrator's role is described in table 6.10.

Table 6.10 Test Result for Administrator

Test Case ID	Actual Result	Result (Success/Fail)
AD001_01	System directs the user into dashboard page	Success
AD001_02	System display an error message of ‘invalid email/password’	Success
AD001_03	System ask user to fill out the blank field	Success
AD002_01	System display list of houses	Success
AD002_02	System display message of ‘New house added’	Success
AD002_03	System display message of ‘Update is successful’	Success
AD002_04	System display message of ‘Delete is successful’	Success
AD003_01	System display list of landlord	Success
AD003_02	System display message of ‘New landlord added’	Success
AD003_03	System display message of ‘Update landlord successful’	Success
AD003_04	System display message of ‘Delete landlord successful’	Success
AD004_01	System display list of tenant	Success
AD004_02	System display message of ‘New tenant added’	Success
AD004_03	System display message of ‘Update is successful’	Success

AD004_04	System display message of 'Delete is successful'	Success
AD005_01	System direct user to change new password	Success
AD005_02	System display error message 'Password and confirm password does not match'	Success

This project has executed 25 number of test cases for Landlord's role in HRAS using 14 number of data. 25 test cases met the expected results. The detail of the test result for Landlord's role is described in table 6.11.

Table 6.11 Test Result for Landlord

Test Case ID	**Actual Result**	**Result (Success/Fail)**
S001_01	System direct user to login page	Success
S001_02	System display error message 'Invalid email/password'	Success
S001_03	System ask user to fill out the blank field	Success
S001_04	System directs the user into dashboard page	Success
S002_01	System display user personal details	Success
S002_02	System display message of 'Update is successful'	Success
S003_01	System display list of houses	Success
S003_02	System display message of 'New house added'	Success

S003_03	System display message of 'Update is successful'	Success
S003_04	System display message of 'Delete is successful'	Success
S004_01	System display list of applicant to rent house	Success
S004_02	System display 'Your house contract has been submitted successfully'	Success
S004_03	System update the selected house status to 'In-agreement'	Success
S005_01	System display list of agreement upload by tenant	Success
S005_02	System display message of 'signed contract had been upload'	Success
S005_03	System update the status of selected house to 'unavailable'	Success

This project has executed 14 number of test cases for Tenant's role in HRAS using 12 number of data. 14 test cases met the expected results. The detail of the test result for Tenant's role is described in table 6.12.

Table 6.12 Test Result for Tenant

Test Case ID	Actual Result	Result (Success/Fail)
T001_01	System direct user to login page	Success
T001_02	System display error message 'Invalid email/password'	Success
T001_03	System direct user into dashboard	Success

T002_01	System display personal details	Success
T002_02	System display message of ‘update is successful’	Success
T002_03	System display list of contact	Success
T002_04	System display message of ‘contact information added successful’	Success
T003_01	System display contract details	Success
T003_02	System display a message ‘Contact uploaded successful’	Success
T004_01	System display checklist form	Success
T004_02	System display a message ‘Your checklist form submitted successful’	Success

Summary for recorded Test Case

A total of 60 test cases had been carried our in this project. We summarized the result in Figure 6.1.

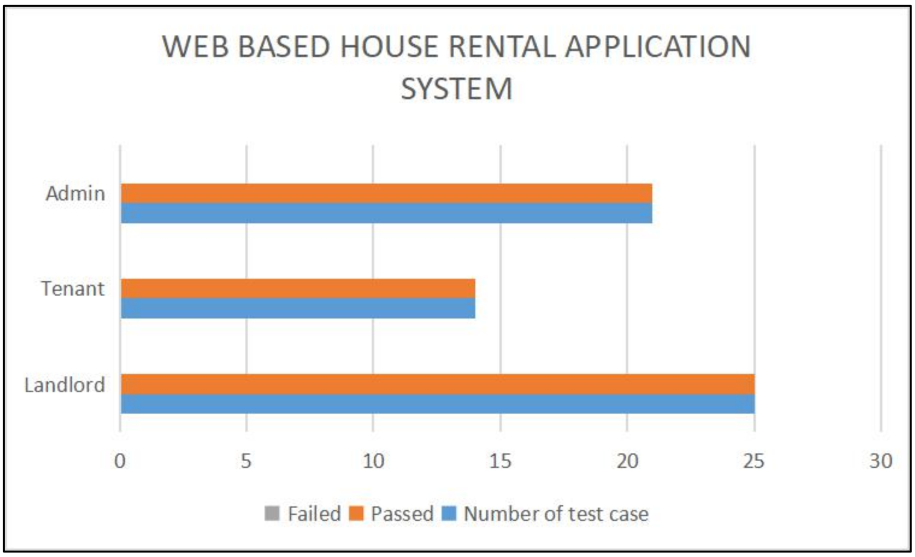


Figure 6.1 Bar Chart of test result for HRAS.

6.6 Conclusion

In conclusion, this chapter discusses the categories of testing that must be performed before to the end-user deployment. The testing process is critical for ensuring that the system performs as intended in accordance with the specifications. The strengths and weaknesses of the Web Application to Rent House System, as well as future enhancements that can be made on the system, will be discussed in the following chapter.

CHAPTER 7: PROJECT CONCLUSION

7.1 Observation on Weakness and Strengths

During the testing phase of the Web Application to Rent House System, a few strengths and drawbacks were discovered. The system's strengths and weakness will set this project apart from the others.

7.1.1 The Strength of HRAS

- The passwords stored in the database are encrypted and protected to protect the information from unauthorized users.
- For each account, the system restricts the use of the email account and password. Users are are not permitted to register for more than one account using the same email address.
- The system makes the rental process easier for both the tenant and landlord bu automating the renting process.
- The system can alleviate the problem of searching for a property by allowing the tenant to search rental houses on the system, saving the tenant a significant amount of time, energy, and money compared to searching for a house manually.
- The tenant will be able to pay rent on the system and it will ease the landlord's tracking of the rent for every tenant that rents their house.
- The landlord will be able to advertise their properties and it can reduce the time taken for the landlord to find a suitable tenant to rent the house.

- The system will notify the landlord through email of every application to rent a house.

7.1.2 The limitation of HRAS

- If there is any incorrect data that was recorded, the system will not be able to fix the mistakes made by the landlord and tenant.
- The system is unable to automatically collect each tenant's monthly rental payment, and tenant must pay directly through the system.
- The tenant is not notified of any changes to the lease agreement that the landlord had signed.

7.2 Propositions of Improvement

The Web Application to Rent House System can be enhanced based on the flaws mentioned in the previous section. A few recommendations should be implemented. One recommendation that can be integrated into the system is to set up an auto deduction for monthly rental payments for each tenant, which will make it easier to keep track of rent payments and make sure they are paid on time. As a result, the tenant will be notified of any deductions made from his or her account for rental payments. In addition, the tenant module can be used to implement a notification system. Every change made by the landlord to the lease agreement will be acknowledged by the tenant using the notification function, allowing the tenant to stay informed at all times. Finally, a new module can be added to the system, such as a message module. The module will allow the tenant to communicate with the landlord in real time through the system. It will make it easier for both the landlord and tenant to communicate with each other if there is any problem with the rental house.

7.3 Project Contribution

The Web Application to the Rent House System belongs to the University Teknikal Malaysia Melaka (UTeM) and Fakulti Teknologi Maklumat and Komunikasi (FTMK). The system's target is to provide an alternative solution for the tenant and landlord that will help to solve the renting process and find the house easily, which can save time, energy, and money when searching for a new rental house.

7.4 Conclusion

In conclusion, the Web Application to Rent House System objectives have been achieved. The problems that were mentioned in the previous chapter have been solved. The landlord is able to manage the renting process easily and find a suitable tenant to rent their house. Other than that, the tenant can search for a rental house easily that will meet their criteria and all the payments related to the rental can be made on the system. With the development of the system, it can overcome the problems that are faced by both tenants and landlords without the need to follow the hectic process anymore. Lastly, the system was developed to ensure that it provides a platform that is user-friendly for the end-user to find and manage the rental house.

REFERENCES

Microsoft. (2016, April 14). *PHP Programming with Visual Studio Code*. https://code.visualstudio.com/docs/languages/php

Khan, A. (2019, March 30). *StackPath*. Https://Www.Cloudways.Com/Blog/Connect-Mysql-with-Php/. https://www.cloudways.com/blog/connect-mysql-with-php/

Editor. (2019, January 29). *How to Choose and Integrate Payment Gateway: Online Payments, Transaction Processing, and Payment Gateways. . .* AltexSoft. https://www.altexsoft.com/blog/business/how-to-choose-and-integrate-payment-gateway-online-payments-transaction-processing-and-payment-gateways-providers/

Free online rental property management software. Innago. (2020, June 1). Retrieved September 12, 2021, from https://innago.com/.

SimplifyEm. (n.d.). *Property management made easy!* SimplifyEm. Retrieved September 12, 2021, from https://www.simplifyem.com/.

Property management software. Buildium. (2021, February 25). Retrieved September 12, 2021, from https://www.buildium.com/.