Applied Information Technology And Computer Science Vol. 1 No. 1 (2020) 108–119

AITCS

Homepage: https://publisher.uthm.edu.my/periodicals/index.php/aitcs
e-ISSN : 2773-5141

Resident Management System for Private Residential

Nur Sobah Ahmad Nizar[1], Muhaini Othman[1*]

[1]Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia, Parit Raja, Batu Pahat Johor,86400, MALAYSIA

DOI: https://doi.org/10.30880/aitcs.2020.01.01.012
Received 31 October 2020; Accepted 28 November 2020; Available online 30 December 2020

Abstract: Resident Management System for Private Residential is a system developed to effectively operates the management of residents in Bistari Hostel. The goal was to develop a web-based system which can be accessible by an administrator, fellow residents including public students who wishes to rent a room at Bistari Hostel. The problem arise from the current system was student may not be able to meet the operational time allocated due to class's schedule. Due to on-shelves manual record using physical form, the processing time for resident's registration may take longer as management staff were unaware of room's availability. Whereas forms' redundancy may happen due to unrestricted amount of forms submitted by each potential and existing resident. Hence, it maximizes the management's workload, and affected the flexibility of resident's information. This system was developed using Waterfall model of structured approach. A management staff who will act as an administrator will be able to manage the records of existing residents including monthly rental payment and request forms submitted. Whereas residents will be able to submit rent payment verification and maintenance request forms, including the accessibility to view monthly rent statement. Whereas public students will have access to a request for accommodation form and be able to view the submitted form's status. Through the pro-posed system, workload of management staff can be reduced since the system will systematically conquest the manual process of acquiring resident applications and rent payment verification and maintenance requests from every potential and existing residents of Bistari Hostel.

Keywords: Resident management system, hostel management, student residential

1. Introduction

In the current era of growing technologies, it is inadvisable to use manual methods in managing an operational system. However, the management of Bistari Hostel still manages the hostel operation of managing new resident's application and its existing residents manually. The management staff manually verifies the Request for Accommodation form that was written and personally submitted by students who wishes to rent a room. The forms received from students are then stored in shelves for future references. Whereas on monthly basis, the receipts for rent were manually generated by the management staff and residents were obligated to collect the copy of the receipt at the office while the

**Corresponding author: muhaini@uthm.edu.my*

original receipts are stashed in the respective resident's file. Moreover, residents who wish to request for maintenance must collect a Maintenance Request form at the office and return it back to the office once filled.

The objective of this paper is to present the development of Resident Management System using structured approach. Therefore, the designation of the new computerized system was led by the disadvantages from the existing manual system stated. The idea of using the computerized system was to save time on manual labor of registering new residents including the process of verifying and responding to every maintenance request submitted by existing residents. Moreover, the system may lead to the reduction of papers wasted on manual forms, filing for resident's information, and generating rent's receipts. With a proper computerized system in place, efficiency of the existing system can be improved, thus overcoming these disadvantages.

2. Literature Reviews

This section will discuss on literature review that had been done for the system and current systems that was being used by the hostel. Section 2.1 will dis-cuss about the background of case study for hostel management, and Section 2.2 will discuss about the technology used in the system. Consequently, Section 2.3 presents three selected existing management system, and the comparison with the developed system.

2.1 Review on Resident Application and Resident Management

Hostel has become an important part of students' life especially for those living far away from home. The students requiring hostel will be the senior students in university as their contract with university's residence has ended. The resident's registration is the process of applying as and accommodating new residents into the hostel. In the process of registering these senior students, a private hostel must display a proper management to attract students. Before the hostel's management registers new residents, they must implement appropriate residents planning and forecasting on the limited number of residents they are able to accommodate based on the different type of rooms. The basis for forecasting is the hostel's rooms availability, organization's monthly budget and its short and long-term plans such as the possibility of expanding room's quantity.

2.2 Comparison with the Existing Systems

There are total of three existing systems and one proposed system to be com-pared. As shown in Table 1, features such as technology, module and functional will be compared between existing systems and proposed system. There were some common features of the existing systems which were controlling user's access, evaluating requests submitted and generating reports. Moreover, there were additional features for the Resident Management System in comparison to the existing systems.

Table 1: Comparison with exisitng systems

Features / Systems	**MyUM Student Portal**	**SACAD UTHM**	**UiTM Student Portal**	**Resident Management System**
Login	√ Username and password	√ Username and password	√ Username and password	√ Username and password
Process Request	√	√	X	√
Hostel Application Request	√	X	X	√

Table 1: (cont.)

Features / Systems	MyUM Student Portal	SACAD UTHM	UiTM Student Portal	Resident Management System
Maintenance Request	X	√	X	√
Bill Transaction	√	X	√	√
Generate Report	√	√	√	√
Notification	X	√	X	√

3. System Methodology

Selecting an appropriate development approach can be easy when ways to handle the issues is known. From the studied case, Bistari Hostel is currently using manual operation in managing the residents. Therefore, structured approach is considered, and the Waterfall Model is chosen. According to [1], the model emphasizes phases of logical progression to be executed throughout the SDLC, similar to cascading down a waterfall. The model illustrates a linear sequential flow where each phase begins only when its preceding phase is completed.

Table 2: Software development activities

Phase	Task	Output
Planning	• Determining the project schedule, activities, and output	• Develop Gantt chart
Analysis	• Gathering the data • Analysing the information gathered	• User requirement • Data Flow Diagram • Entity relationship diagram • Hardware and software specification
Design	• Designing the user interfaces	• System's interfaces • Relationship schema and data dictionary
Implement ation	• Developing the system	• System programming coding • PHP and MySQL coding
Testing	• Run system testing • Fixing the error identified during system testing	• Error report • User acceptance testing

The project planning is structured by using Gantt Chart (refer to **Appendix A**). It was done by observation and study of the existing systems. The Gantt chart will show how the Resident Management System will be developed.

4. Analysis & Design

This section discusses on system's design to describe the implementation of the proposed system's development. The system requirement analysis and design will be the guideline in formulating the data flow diagram, entity relationship diagram and flowchart. The task performed is vital in the process of fulfilling the system requirement.

4.1 System Requirement Analysis

A functional requirement defined what has been done by identifying the necessary activity, task and action that must be accomplished by de-scribing the process of inputs' transformation to the desired output for the user.

Table 3 specifies the functional requirement for Resident Management System.

Table 3: Functional Requirements

Module	Functional Requirement
Login	• The system should allow user to log in into the system using a valid username and password • The system should allow user to input username and password • The system should alert user for any invalid input • The system should redirect user to their respective profile's upon successful login
Resident Application	• The system should allow public user to apply for residency
Resident	• The system should allow admin to register resident into the system • The system should allow resident to view their respective profile
Maintenance	• The system should allow resident to submit maintenance request form
Rent	• The system should allow resident to submit rent payment verification request form
Report	• The system should be able to generate the list of forms submitted

Non-functional requirement specifies the criteria that can be used to judge the operation of a system, rather than specific behaviors.

Table 4 specifies the functional requirement for Resident Management System.

Table 3: Non-Funstional Requirements

	Non-Functional Requirement
Operational	• The system should be able to operate on any web browser • The system should be user friendly
Implementation	• The interaction between the user and the system should be able to operate
Safety	• Every user is required to input the accurate username and password upon log in

4.2 System Analysis

A context diagram is a top-level view of an information system used to de-fines the scope and boundary for the system [3]. **Appendix B** illustrates the context diagram for Resident Management System for Private Residential which clarifies inputs and outputs between a system and its external entities.

A data flow diagram (DFD) is a tool that maps out the flow of information for any process through a system. Appendix C illustrates the data flow diagram level 0 for Resident Management System for Private residential which depicts the overall process of a system.

Appendix C, D, E and F illustrate the data flow diagram level 1 for Resident Management System for Private Residential which depicts the elaboration of processes from four related modules. Process selected were manage resident application, manage resident, manage maintenance, and manage rent.

4.3 Database Design

Entity Relationship Diagram (ERD) is a data model which depicts data in terms of the entities and relationships described by the data [2]. Entity is a class of certain criteria about which we need to store data. Whereas relationship is a natural business association that exists between one or more entities which represents an event that links the entities. **Appendix H** illustrate the entity relationship diagram for Resident Management System for Private Residential.

4.4 Flowchart

The administrator flowchart in **Appendix I** illustrates the process flow of an administrator's accessibility throughout the system. The system can only be accessible by a verified administrator who has been registered into the system upon development.

While the resident flowchart in **Appendix J** illustrates the process flow of a resident's accessibility throughout the system. The system can only be accessible by current resident who has been registered into the system by an administrator.

Whereas the applicant flowchart in **Appendix K** illustrates the process flow of applying for residency at Bistari Hostel throughout the system. The application form can be accessed by public students who wishes to apply for residency without restriction. Furthermore, the status of resident application submitted must be verified by an identification card number.

5. Implementation & Testing

The system was developed based on the specified design in the early phase of the project. The process involved software's installation, system's development where programming was included to develop every module planned in the system and system's control.

The functions and features of Resident Management System was put into test in System Functionality Testing to ensure the effectiveness of the system based on all the previously designed specifications and requirements. The system has been implemented with interfaces to enable easier user interaction and navigation. Test plan was used to support the functional testing on different test cases. Test plan is important to ensure the outcome of the system was as how it was expected. The items included in test plan were test case, objective, expected result and result of testing.

System Functionality Testing was conducted once the system was fully developed. It was done to ensure the functionality of every module involved in the system functions as how it was originally planned. The System Functionality Testing of Resident Management System for Private Residential was repeatedly conducted to avoid error among every linked interface. Furthermore, every accessible button or link were tested to ensure that it was precisely accessible.

6. Conclusion

The conclusion should summarize the main findings of the study, and restate the key points inferred from trends observed and discussed regarding the data. Some suggestions should be included to encourage the continuation of the current research.

Overall, the system which has been developed is functioning well as a whole and has led to the easement of management operation in managing existing and potential residents of Bistari Hostel. The system can be redesigned using the recommendation discussed in this s to strive on improving the quality and upgrading the current system for future use. In conclusion, the developed system is able to achieve the objectives of improving the current system which has been operating manually and testing the functionality of Resident Management System.

Based on the testing result of the system, there are several advantages that can be seen in Resident Management System. The advantages are as follow:

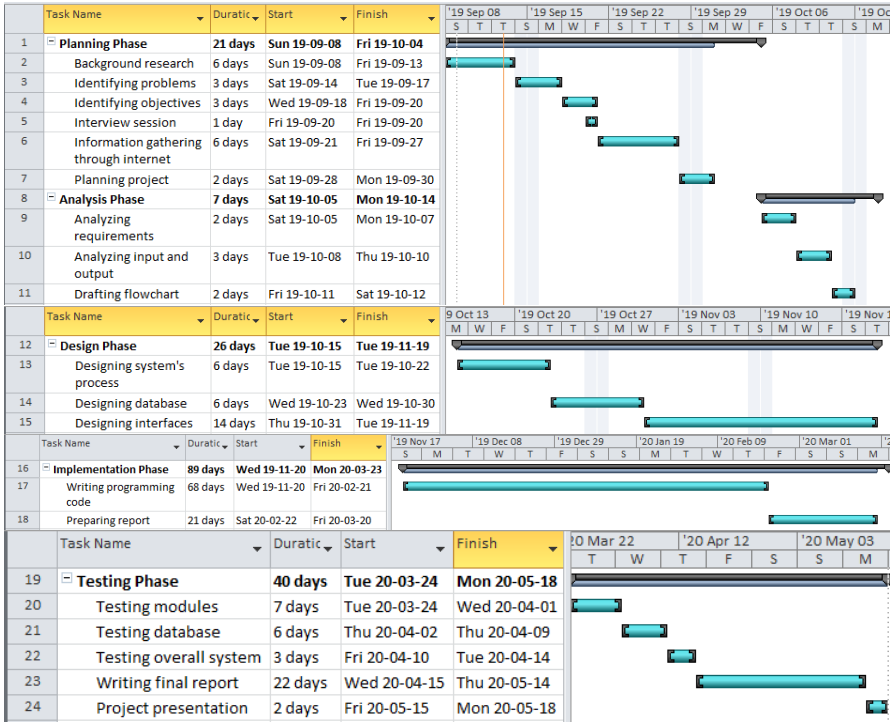
1. Public user can apply for residency and view the status of the ap-plication through Application page.
2. Administrator can register residents easily by viewing the applicant's list in the right side of the registration form's column.
3. The functionality in rent statement page will enable the administrator to be notified with a list of residents who have not yet pay the monthly rent.

There are also several disadvantages that can be seen in Resident Management System. Mainly, as an administrator, they are not able to change the resident's rent payment status if the resident were to pay the rent in cash and did not submit a rent payment verification request. Hence, the administrator's control is limited to only managing the resident who paid the payment through methods other than cash. Moreover, after residents have submitted a maintenance request through the system, they are no longer able to give feedback if the maintenance issue recured after it has been resolved. Hence, the resident is required to submit a new maintenance request which may lead to miscommunication with the management staff.

Acknowledgement

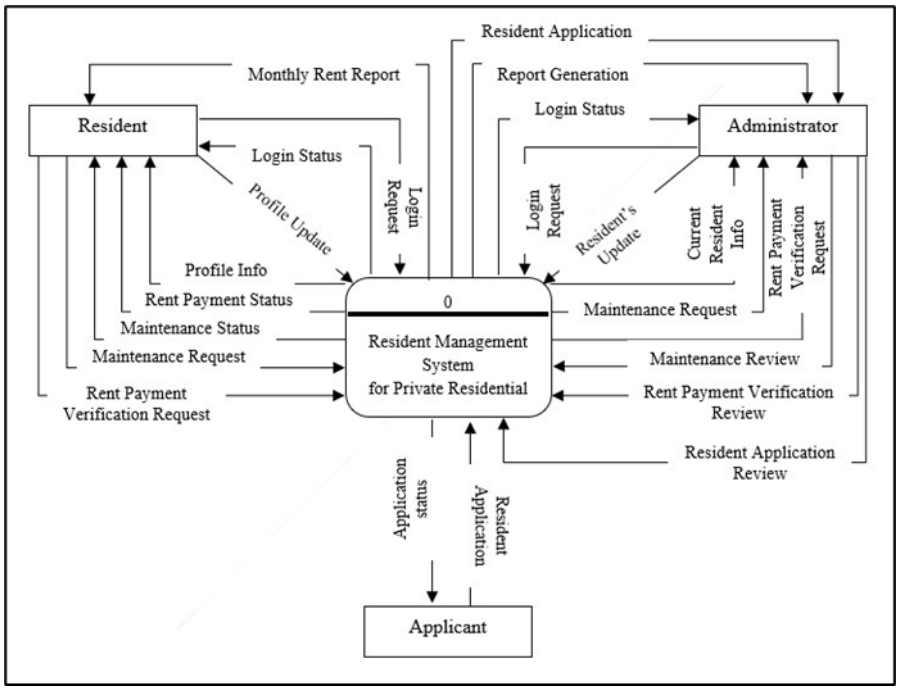
The authors would like to thank the Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia for its support and encouragement throughout the process of conducting this study.

Appendix A

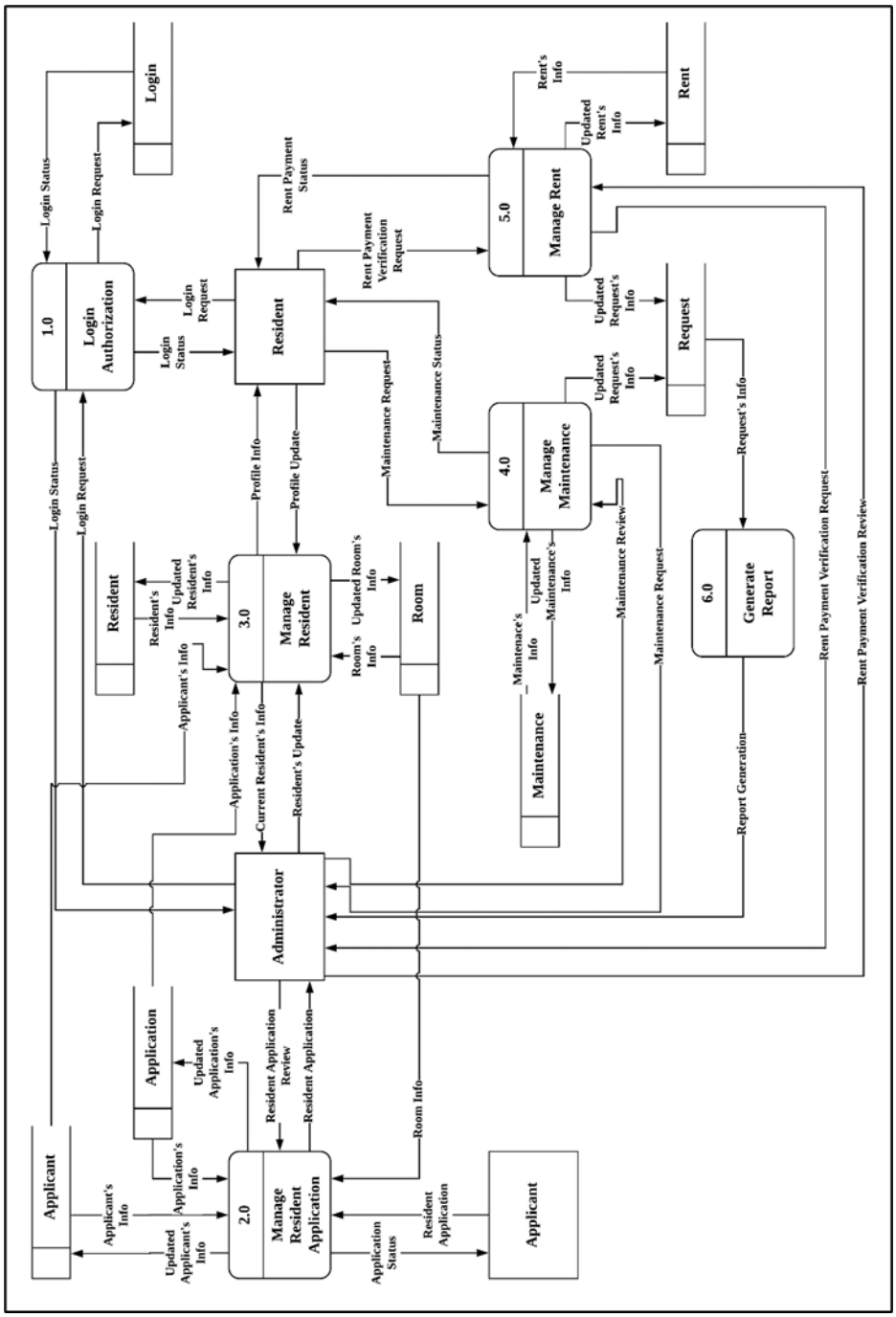


	Task Name	Duration	Start	Finish
1	**Planning Phase**	**21 days**	**Sun 19-09-08**	**Fri 19-10-04**
2	Background research	6 days	Sun 19-09-08	Fri 19-09-13
3	Identifying problems	3 days	Sat 19-09-14	Tue 19-09-17
4	Identifying objectives	3 days	Wed 19-09-18	Fri 19-09-20
5	Interview session	1 day	Fri 19-09-20	Fri 19-09-20
6	Information gathering through internet	6 days	Sat 19-09-21	Fri 19-09-27
7	Planning project	2 days	Sat 19-09-28	Mon 19-09-30
8	**Analysis Phase**	**7 days**	**Sat 19-10-05**	**Mon 19-10-14**
9	Analyzing requirements	2 days	Sat 19-10-05	Mon 19-10-07
10	Analyzing input and output	3 days	Tue 19-10-08	Thu 19-10-10
11	Drafting flowchart	2 days	Fri 19-10-11	Sat 19-10-12
12	**Design Phase**	**26 days**	**Tue 19-10-15**	**Tue 19-11-19**
13	Designing system's process	6 days	Tue 19-10-15	Tue 19-10-22
14	Designing database	6 days	Wed 19-10-23	Wed 19-10-30
15	Designing interfaces	14 days	Thu 19-10-31	Tue 19-11-19
16	**Implementation Phase**	**89 days**	**Wed 19-11-20**	**Mon 20-03-23**
17	Writing programming code	68 days	Wed 19-11-20	Fri 20-02-21
18	Preparing report	21 days	Sat 20-02-22	Fri 20-03-20
19	**Testing Phase**	**40 days**	**Tue 20-03-24**	**Mon 20-05-18**
20	Testing modules	7 days	Tue 20-03-24	Wed 20-04-01
21	Testing database	6 days	Thu 20-04-02	Thu 20-04-09
22	Testing overall system	3 days	Fri 20-04-10	Tue 20-04-14
23	Writing final report	22 days	Wed 20-04-15	Thu 20-05-14
24	Project presentation	2 days	Fri 20-05-15	Mon 20-05-18

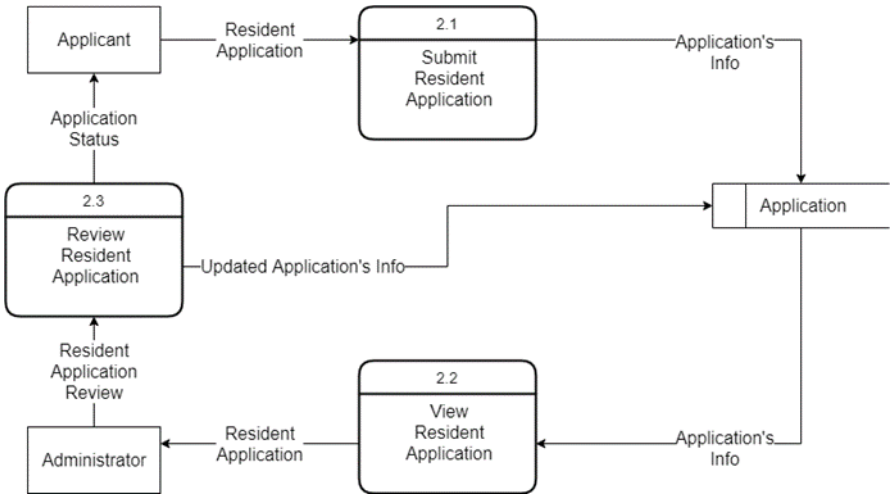
Appendix B



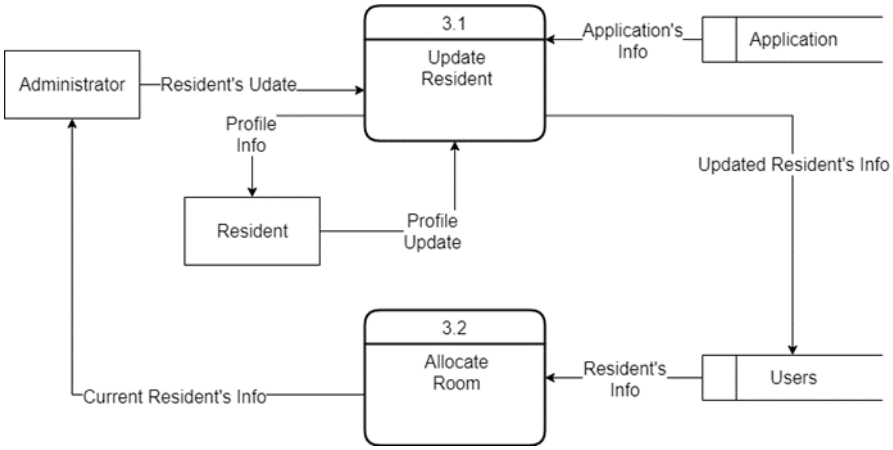
Appendix C


Login
Login Status
Login Request
1.0
Login Authorization
Resident
Rent Payment Status
Rent Payment Verification Request
5.0
Manage Rent
Rent's Info
Updated Rent's Info
Rent
Request
Updated Request's Info
Request's Info
Maintenance Request
Maintenance Status
4.0
Manage Maintenance
Maintenance Review
Profile Info
Profile Update
Resident's Info
Updated Resident's Info
3.0
Manage Resident
Updated Room's Info
Room's Info
Room
Maintenance's Info
Updated Maintenance's Info
Maintenance
Applicant's Info
Application's Info
Current Resident's Info
Resident's Update
6.0
Generate Report
Report Generation
Rent Payment Verification Request
Rent Payment Verification Review
Administrator
Application
Updated Application's Info
Resident Application Review
Resident Application
Room Info
Applicant
Applicant's Info
Updated Applicant's Info
2.0
Manage Resident Application
Application's Info
Application Status
Resident Application
Applicant

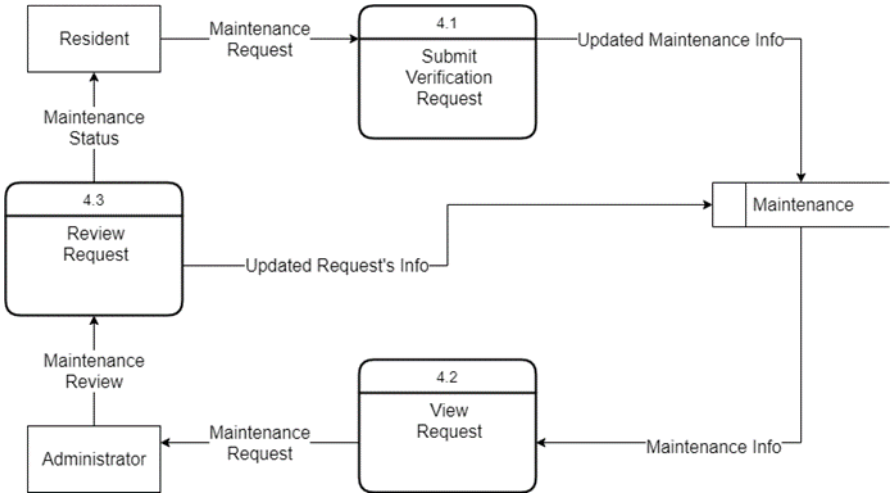
Appendix D



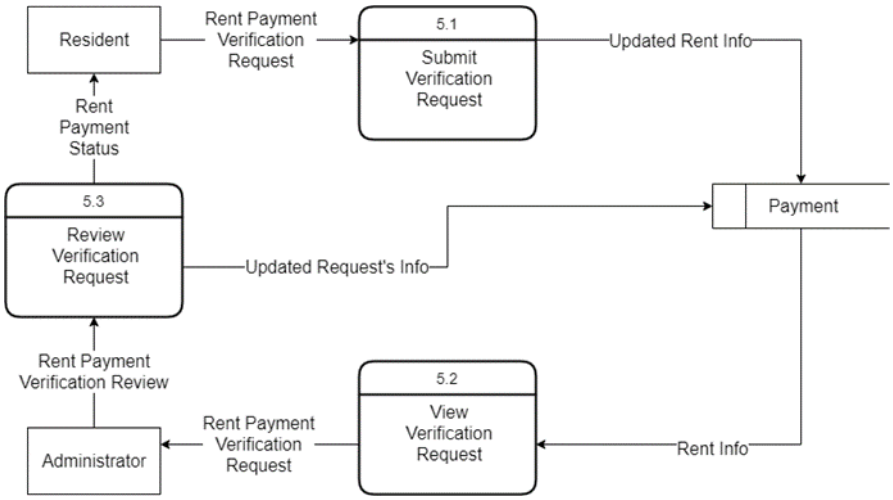
Appendix E



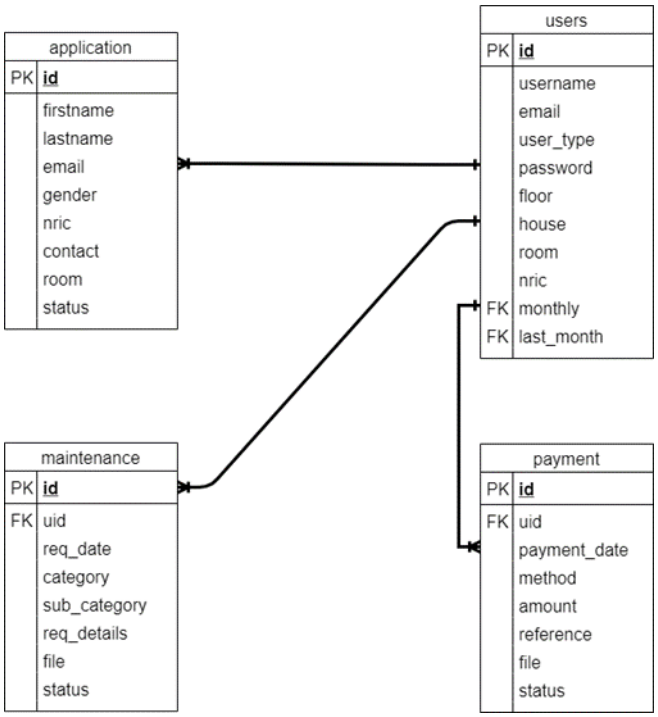
Appendix F



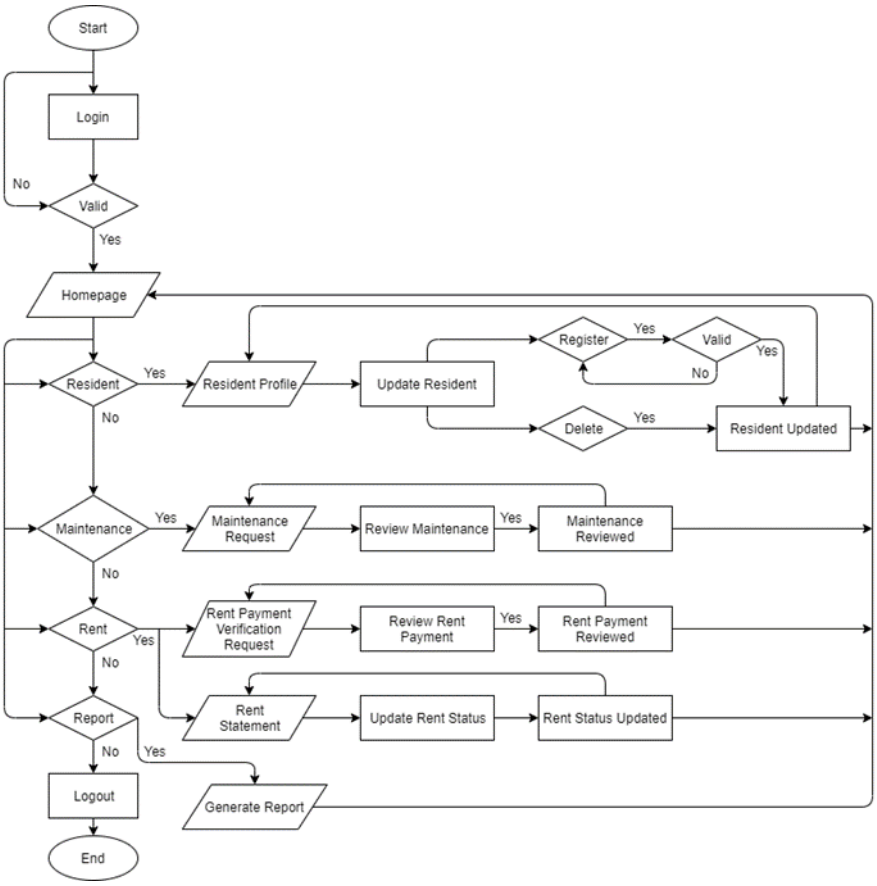
Appendix G



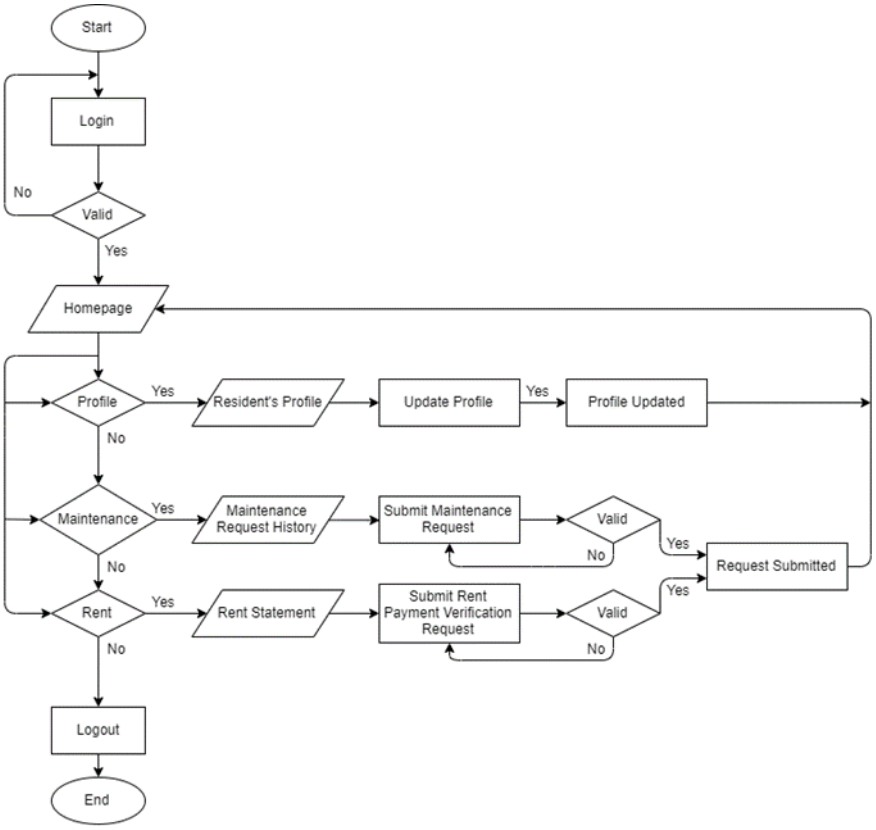
Appendix H



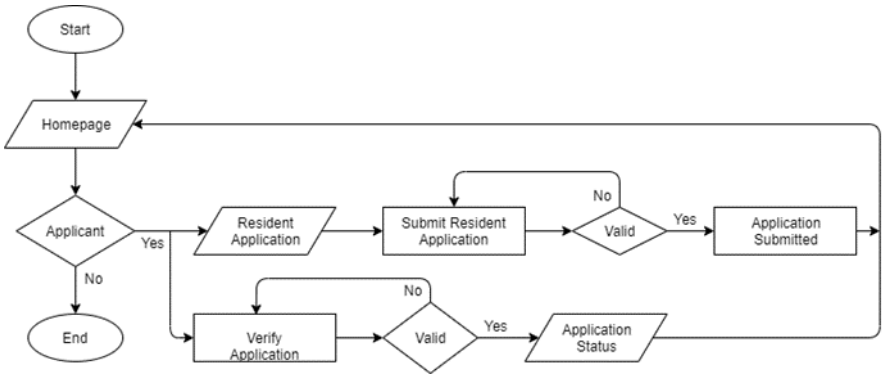
Appendix I



Appendix J



Appendix K



References

[1] Y. Bassil, "A simulation model for the waterfall software development life cycle". arXiv preprint arXiv:1205.6904 , 2012.

[2] J. L .Whitten, & L. D. Bentley, "Systems analysis and design methods". 4th ed., Boston, Massachusetts: Tom Casson 1998.

[3] G.B. Shelly, & H.J Rosenblatt, "Essentials of Systems Analysis and Design". 9th ed., Course Technology, Cengage Learning, Boston, USA, 2012.