



Faculty of Computer Science and Information Technology

SMK BAU HOSTEL MANAGEMENT SYSTEM

Kuan Sze Li

Bachelor of Computer Science with Honors
(Network Computing)
2014

SMK BAU HO:



MENT SYSTEM

Kuan Sze Li

This project is submitted in partial fulfillment of the requirements for the degree of
Bachelor of Computer Science with Honours
(Network Computing)

Faculty of Computer Science and Information Technology

UNIVERSITI MALAYSIA SARAWAK

2014

UNIVERSITI MALAYSIA SARAWAK

THESIS STATUS ENDORSEMENT FORM

TITLE SMK BAW HOSTEL MANAGEMENT SYSTEM

ACADEMIC SESSION: 2013/2014

(CAPITAL LETTERS)

hereby agree that this Thesis* shall be kept at the Centre for Academic Information Services, Universiti Malaysia Sarawak, subject to the following terms and conditions:

1. The Thesis is solely owned by Universiti Malaysia Sarawak
2. The Centre for Academic Information Services is given full rights to produce copies for educational purposes only
3. The Centre for Academic Information Services is given full rights to do digitization in order to develop local content database
4. The Centre for Academic Information Services is given full rights to produce copies of this Thesis as part of its exchange item program between Higher Learning Institutions [or for the purpose of interlibrary loan between HLI]
5. ** Please tick (√)

- CONFIDENTIAL (Contains classified information bounded by the OFFICIAL SECRETS ACT 1972)
- RESTRICTED (Contains restricted information as dictated by the body or organization where the research was conducted)
- UNRESTRICTED



(AUTHOR'S SIGNATURE)

Validated by



(SUPERVISOR'S SIGNATURE)

Permanent Address

NO 466B, LOT 2650, JALAN BELATOK,
TAMAN ALAMANDA INDAH, 92050
KUCHING SARAWAK

Date: 27/06/2014

Date: 27/6/2014

Note * Thesis refers to PhD, Master, and Bachelor Degree
** For Confidential or Restricted materials, please attach relevant documents from relevant organizations / authorities

ACKNOWLEDGEMENT

My sincere gratitude goes to the project supervisor, Mdm. Eaqerzilla Phang for her patience in supervision; support and assistance given that truly help the progression and smoothness of this project.

Besides that, I also want to express my earnest appreciation to the final year project coordinator, Professor Dr Wang Yin Chai for his deep commitment and invaluable contributions to provide the guidelines that are essential for the successful completion of the project.

Apart from that, a great deal of appreciation also goes to Faculty of Computer Science and Information Technology (FCSIT) for providing the opportunity to undergo the final year project.

Finally, I would like to thank my family for the support and encouragement they have given me along the progress of my work.

Contents

ACKNOWLEDGEMENT	4
ABSTRACT.....	14
ABSTRAK.....	15
CHAPTER 1	16
INTRODUCTION	16
1.0 Introduction.....	16
1.1 Problem Statement	16
1.2 Objectives	17
1.3 Procedures / Methodologies.....	17
1.3.1 Object-Oriented Analysis (OOA)	18
1.3.2 Object-Oriented Design (OOD)	18
1.3.3 Object-Oriented Implementation	19
1.4 Scope.....	19
1.5 Significant of project.....	20
1.6 Project Schedule.....	21
1.7 Expected Outcome	21
CHAPTER 2	22
BACKGROUND STUDY	22
2.0 Introduction.....	22
2.1 Review on SMK BAU Hostel Management System using Object-Oriented Approach.....	22
2.2 Review on CUSAT Hostel Management System	23
2.2.1 The Major Functions of the CUSAT Hostel Management System:.....	24
2.3 Review on RANAK Hostel Management System	25
2.3.1 The Key features of RANAK Hostel Management System:.....	26
2.3.2 Review on Hostel Management System using Asp.Net.....	27
2.4 Comparison of the Review Systems	29
2.5 Review on Programming Tool.....	33
2.6 Review on ASP.Net	33
2.7 Summary.....	34
CHAPTER 3	35
REQUIREMENT ANALYSIS AND DESIGN.....	35

3.0 Requirement Analysis.....	35
3.1 Introduction.....	35
3.1.1 Data Gathering	35
3.1.2 Requirements	36
3.1.3 Functionality of the System	38
3.1.4 Hardware requirement.....	38
3.1.5 Software requirement.....	39
3.1.6 Identified Key Modules	40
3.1.7 User Requirement (System Requirement Specification)	40
3.2 System Design	59
3.2.1 System Design Introduction.....	59
3.2.2– Software Architecture	60
3.2.3 Diagram.....	61
3.2.4 – User Interface Design	145
3.2.4.1 – User Page Design.....	145
3.3 Conclusion	146
CHAPTER 4	147
IMPLEMENTATION.....	147
4.1 Introduction.....	147
4.2 Modules Implemented	147
4.3 Implementation by Phase.....	147
4.3.1 Phase 1 (System’s Authentication Module).....	148
4.3.1.1 Maintenance Module.....	148
4.3.1.2 Authentication Module.....	148
4.3.2 Phase 2 (System’s Disciplinary Module, Maintenance Module, and Operation	148
Module).....	148
4.3.2.1 Disciplinary Module	148
4.3.2.2 Operation Module.....	148
4.3.3 Phase 3 (System’s Report Module).....	149
4.4 Description of Implementation of System Architecture	149
4.4.1 Database	149
4.4.2 Data Layer	150
4.4.3 Control	151
4.4.4 Presentation.....	152

4.5	Implemented System User Interface	153
4.6	Difficulties Encounters during Implementation.....	162
4.6.1	Missing Function	162
4.7	Conclusion	163
CHAPTER 5		164
TESTING		164
5.1	Introduction.....	164
5.1.1	Unit Testing	164
5.1.2	Integration Testing.....	165
5.1.3	System Testing.....	165
5.1.4	User Acceptance Testing	165
5.2	Testing Strategy	166
5.3	Test Conducted	168
5.3.2	Integration Testing	189
5.3.3	System Testing.....	192
5.3.4	User Acceptance Testing	198
5.4	Conclusion	201
CHAPTER 6		202
CONCLUSION AND FUTURE WORK		202
6.1	Project Conclusion	202
6.1.1	User Requirements Met	202
6.1.2	Project Objectives	202
6.1.3	System Functionality.....	203
6.2	System Limitation.....	204
6.3	Future Enhancement (s)	205
6.3.1	Upgrade of Modules.....	205
6.3.2	Integrate the system with other system	205
6.3.3	Better User Interface design.....	205
6.3.4	Upgrade Maintenance Module.....	205
6.4	If Given a Second Try.....	206
6.4.1	– Better User Interface Design.....	206
6.4.2	– Spend More time On Analysis and Design Stage	206
6.5	Main Computational Challenge	207
6.6	Lesson Learnt.....	208

6.7 Conclusion.....	209
REFERENCES	210
APPENDIX.....	211

List of Table

Table 1: Administrator Module.....	28
Table 2: User Module	28
Table 3: Comparison in the Functionality.....	32
Table 4: Types of software with their purposes description	39
Table 5: Login Use Case.....	44
Table 6: Manage User Use Case.....	45
Table 7: Manage Role Use Case.....	46
Table 8: Change Password Use Case.....	47
Table 9: Merit Use Case	48
Table 10: Demerit Use Case	49
Table 11: Manage Parent Record Use Case.....	50
Table 12: Manage Dorm Room Record Use Case.....	52
Table 13: Manage Student Profile Use Case	52
Table 14: Manage Staff Record Use Case	53
Table 15: Room Availability Use Case.....	56
Table 16: Room Application Use Case.....	57
Table 17: Report Use Case	58

List of Figure

Figure 1: General view of the RANAK Hostel Management System	25
Figure 2: General view of log-in page	26
Figure 3: General view of log-in page	27
Figure 4: Authentication Module.....	41
Figure 5: Disciplinary Module.....	41
Figure 6: Maintenance Module.....	42
Figure 7: Operation Module.....	43
Figure 8: Report Module.....	43
Figure 9: Login Activity Diagram	62
Figure 10: Change Password Activity Diagram	63
Figure 11: Inserts New User Record Activity Diagram.....	64
Figure 12: Modifies existing user record Activity Diagram	65
Figure 13: Deletes existing user record Activity Diagram.....	65
Figure 14: Views existing user record Activity Diagram	66
Figure 15: Filters user record list Activity Diagram	66
Figure 16: Inserts new role record Activity Diagram	67
Figure 17: Modifies existing role record Activity Diagram.....	67
Figure 18: Delete existing role record Activity Diagram.....	68
Figure 19: Views existing role record Activity Diagram.....	68
Figure 20: Filters role record list Activity Diagram.....	69
Figure 21: Logout Activity Diagram	69
Figure 22: Inserts new merit record Activity Diagram	70
Figure 23: Modifies existing merit record Activity Diagram	70
Figure 24: deletes existing merit record Activity Diagram.....	71
Figure 25: Confirms existing merit record Activity Diagram.....	71
Figure 26: Views existing merit record Activity Diagram.....	72
Figure 27: Filters the merit record list Activity Diagram	72
Figure 28: Inserts new demerit record Activity Diagram	73
Figure 29: Modifies existing demerit record Activity Diagram.....	73
Figure 30: Confirms existing demerit record Activity Diagram.....	74
Figure 31: Confirms existing demerit record Activity Diagram.....	74
Figure 32: Views existing demerit record Activity Diagram.....	75
Figure 33: Filters the demerit record list Activity Diagram.....	75
Figure 34: Inserts new parent record Activity Diagram.....	76
Figure 35: Modifies existing parent record Activity Diagram	76
Figure 36: Deletes existing parent record Activity Diagram	77
Figure 37: Views existing parent record Activity Diagram	77
Figure 38: Filters parent record list Activity Diagram.....	78
Figure 39: Inserts new dorm room record Activity Diagram.....	78
Figure 40: Modifies existing dorm room record Activity Diagram	79
Figure 41: Deletes existing dorm room record Activity Diagram	79
Figure 42: Views existing dorm room record Activity Diagram	80
Figure 43: Filters dorm room record list Activity Diagram.....	80
Figure 44: Inserts new student profile Activity Diagram.....	81

Figure 45: Modifies existing student profile Activity Diagram	81
Figure 46: Deletes existing student profile Activity Diagram	82
Figure 47: View existing student profile Activity Diagram.....	82
Figure 48: Filters the student profile record list Activity Diagram.....	83
Figure 49: Inserts new staff record Activity Diagram	83
Figure 50: Modifies existing staff record Activity Diagram.....	84
Figure 51: Deletes existing staff record Activity Diagram	84
Figure 52: Views existing staff record Activity Diagram	85
Figure 53: Filters staff record list Activity Diagram.....	85
Figure 54: Insert Application Priority Record	86
Figure 55: Modify Application Priority Record Activity Diagram.....	87
Figure 56: Delete Application Priority Record Activity Diagram	88
Figure 57 : Display Application Priority Record Activity Diagram	88
Figure 58 : Filter Application Record Activity Diagram	89
Figure 59: Inserts new dorm allocation record Activity Diagram	90
Figure 60: Modifies existing dorm allocation record Activity Diagram.....	91
Figure 61: Deletes existing dorm allocation record Activity Diagram	91
Figure 62: Views existing dorm allocation record Activity Diagram.....	92
Figure 63: Filters dorm allocation record list Activity Diagram.....	92
Figure 64 : Auto Dorm Allocation Activity Diagram.....	93
Figure 65: Views existing room availability Activity Diagram.....	94
Figure 66: Views existing room availability detail Activity Diagram	94
Figure 67: Inserts new room application record Activity Diagram	95
Figure 68: Modifies existing room application record Activity Diagram.....	95
Figure 69: Submits existing room application record Activity Diagram	96
Figure 70: Deletes existing room application record Activity Diagram	96
Figure 71: Views existing room application record Activity Diagram.....	97
Figure 72: Filters room application record list Activity Diagram.....	97
Figure 73 : Set Room Application' Activity Diagram	98
Figure 74 : Evict Student' Activity Diagram	99
Figure 75: Views report Activity Diagram	100
Figure 76: Filters report list Activity Diagram	100
Figure 77: Show record Sequence Diagram	101
Figure 78: Save Sequence Diagram	102
Figure 79: Delete Sequence Diagram	102
Figure 80: Retrieve by parameter Sequence Diagram	103
Figure 81: Login Sequence Diagram	104
Figure 82: Change Password Sequence Diagram	105
Figure 83: Inserts record of new user Sequence Diagram	105
Figure 84: Modifies existing user record Sequence Diagram	106
Figure 85: Deletes existing user record Sequence Diagram	106
Figure 86: Views existing record Sequence Diagram.....	107
Figure 87: Filters user record list Sequence Diagram	107
Figure 88: Inserts new role record Sequence Diagram	108
Figure 89: Modifies existing role record Sequence Diagram	108

Figure 90: Deletes existing role record Sequence Diagram.....	109
Figure 91: Views existing role record Sequence Diagram.....	109
Figure 92: Filters role record list Sequence Diagram	110
Figure 93: Logout Sequence Diagram	110
Figure 94: Inserts new merit record Sequence Diagram.....	111
Figure 95: Modifies existing merit record Sequence Diagram	111
Figure 96: Deletes existing merit record Sequence Diagram.....	112
Figure 97: Confirms existing merit record Sequence Diagram	112
Figure 98: Views existing merit record Sequence Diagram	113
Figure 99: Filters merit record list Sequence Diagram	113
Figure 100: Inserts new demerit record Sequence Diagram	114
Figure 101: Modifies existing demerit record Sequence Diagram	114
Figure 102: Deletes demerit record Sequence Diagram	115
Figure 103: Confirms exiting demerit record Sequence Diagram	115
Figure 104: Views exiting demerit record Sequence Diagram	116
Figure 105: Filters demerit record list Sequence Diagram	116
Figure 106: Inserts new parent record Sequence Diagram	117
Figure 107: Modifies existing parent record Sequence Diagram.....	117
Figure 108: Deletes existing parent record Sequence Diagram	118
Figure 109: Views existing parent record Sequence Diagram.....	118
Figure 110: Filters parent record list Sequence Diagram.....	118
Figure 111: Inserts new dorm room record Sequence Diagram	119
Figure 112: Modifies existing dorm room record Sequence Diagram.....	119
Figure 113: Deletes existing dorm room record Sequence Diagram	120
Figure 114: Views existing dorm room record Sequence Diagram	120
Figure 115: Filters dorm room record list Sequence Diagram.....	120
Figure 116: Inserts new student profile Sequence Diagram	121
Figure 117: Modifies existing student profile Sequence Diagram.....	121
Figure 118: Delete existing student profile Sequence Diagram.....	122
Figure 119: Views existing student profile Sequence Diagram.....	122
Figure 120: Filters student profile record list Sequence Diagram	122
Figure 121: Inserts new staff record Sequence Diagram	123
Figure 122: Modifies existing staff record Sequence Diagram	123
Figure 123: Deletes existing staff record Sequence Diagram.....	124
Figure 124: Views existing staff record Sequence Diagram.....	124
Figure 125: Filters staff record list Sequence Diagram	124
Figure 126: Insert Application Priority Record Sequence Diagram	125
Figure 127: Modify Application Priority Record Sequence Diagram	125
Figure 128: Delete Application Priority Record Sequence Diagram.....	126
Figure 129: View Application Priority Record Sequence Diagram.....	126
Figure 130: Filters Application Priority Record Sequence Diagram	127
Figure 131: Inserts new dorm allocation record Sequence Diagram	127
Figure 132: Modifies existing dorm allocation record Sequence Diagram	128
Figure 133: Deletes existing dorm allocation record Sequence Diagram.....	129
Figure 134: Views existing parent dorm allocation record Sequence Diagram.....	129

Figure 135: Filters dorm allocation record list Sequence Diagram	130
Figure 136: Auto Dorm Allocation Sequence Diagram.....	130
Figure 137: Views existing room availability Sequence Diagram.....	131
Figure 138: Views existing room availability detail Sequence Diagram.....	131
Figure 139: Inserts new room application record Sequence Diagram	132
Figure 140: Modifies existing room application record Sequence Diagram	132
Figure 141: Submits existing room application record Sequence Diagram.....	133
Figure 142: Deletes existing room application record Sequence Diagram	133
Figure 143: Views existing room application record Sequence Diagram.....	134
Figure 144: Filters room application record list Sequence Diagram	134
Figure 145: Set Room Application's Distance and Priority and Confirm	135
Figure 146: Evict Student Sequence Diagram	136
Figure 147: Filters dorm room status report Sequence Diagram	137
Figure 148: Views dorm room status report Sequence Diagram	137
Figure 149: Filters disciplinary status report Sequence Diagram	138
Figure 150: Views disciplinary status report Sequence Diagram	138
Figure 151: Filters student information report Sequence Diagram.....	139
Figure 152: Views student information report Sequence Diagram.....	139
Figure 153: User, Role and Access Control Class Diagram	140
Figure 154: Merit Class Diagram.....	141
Figure 155: Demerit Class Diagram	141
Figure 156: Staff and User Class Diagram	142
Figure 157: Parent and Student Profile Class Diagram	142
Figure 158: Dorm Room and Tenant Class Diagram.....	143
Figure 159: Application Priority Class Diagram	143
Figure 160: Room Application Class Diagram.....	144
Figure 161: Dorm Allocation Class Diagram	144
Figure 162: User Interface Design	145
Figure 163: Home Page Screen Shot	153
Figure 164: Login Page Screen Shot.....	154
Figure 165: After Login Screen Shot.....	154
Figure 166: Dorm Allocation Page Screen Shot.....	155
Figure 167: Room Application Page Screen Shot	155
Figure 168: Room Availability Page Screen Shot	156
Figure 169: Auto Dorm Allocation Screen Shot.....	156
Figure 170: Evict Student Page Screen Shot	156
Figure 171: Demerit Page Screen Shot.....	157
Figure 172: Merit Page Screen Shot	157
Figure 173: Student Detail Page Screen Shot	158
Figure 174: Application Priority Page Screen Shot.....	158
Figure 175: Dorm Room Page Screen Shot	159
Figure 176: Parent Page Screen Shot.....	159
Figure 177: Staff Page Screen Shot	160
Figure 178: Student Profile Page Screen Shot	160
Figure 179: User Page Screen Shot.....	161

Figure 180: Room Application Form Screen Shot.....	161
Figure 181: Final Year Project 1 schedule.....	211
Figure 182: Final Year Project 2 schedule.....	212

ABSTRACT

Hostel Management system is software developed for managing the school hostel, which includes storing the information of the students, generate report for room availability and manage room allocation. This project also deals with the problems on managing a hostel and prevents problems which occur during the manual work that had been carried out for managing the hostel. Identification of the drawbacks of the manual work leads to the designing of this computerised system. Through computerizing the existing system with the help of C#.NET which are an object-oriented programming language, it is hoped that it will be compatible with the existing system and will be more efficient and user friendly.

ABSTRAK

Sistem Pengurusan Hostel adalah perisian yang dibangunkan untuk menguruskan asrama termasuk menyimpan maklumat pelajar, menjana laporan untuk menyemak bilik dan menguruskan peruntukan bilik. Projek ini akan menyelesaikan masalah yang wujud dalam pengurusan asrama dan masalah yang berlaku semasa kerja manual sedang dijalankan untuk menguruskan asrama. Kelemahan yang telah dikenalpasti terdapat dalam kerja manual telah membawa kepada terbentuknya sistem berkomputer ini. Melalui system komputeran yang sedia ada dengan bantuan C#.NET yang merupakan bahasa pengaturcaraan yang berorientasikan objek, diharap ia akan serasi dengan sistem yang sedia ada bagi menghasilkan sistem yang lebih efisien dan mesra pengguna.

CHAPTER 1

INTRODUCTION

1.0 Introduction

The number of students in SMK BAU has increased as compared to the previous years, and this has raised the number of students that requested to stay in the hostel. The increase of hostel application at the school hostel also has burden up the hostel staff in keeping and managing the student's information as the existing manual work to record the student application is slowing down the application progress, creating human error and others problems. In this section, there will be stating the problems or issues faced by the hostel management regarding the existing manual system will be stated. The objectives for this project will also be listed and the introduction of the procedure or methodologies that will be used for the proposed system. Project scope, significance of the project, project schedule and expected outcome for this project will also be explained.

1.1 Problem Statement

The hostel management system is designed for enable the hostel management to record the information about the accommodation for the student and to manage the hostel application in an easy way. Currently, the hostel and the registration form verification are managed manually by the hostel management. The manual work will often create human error, data redundancy, less efficiency and time consuming. To reduce this problem, there is a need to develop the computerized system to overcome the existing problems.

1.2 Objectives

The goal of this project is to create a hostel management system that will manage the hostel room application and allocation which to resolve the issue or problem areas faced by the management with the current manually work managing the hostel.

The objectives intended to be achieved after successfully completing this project are outlined as shown below:

- Develop a Hostel Management System software that are using object-oriented concept
- To computerized the process for hostel application
- To create a database for saving more information

1.3 Procedures / Methodologies

Object-Oriented Systems Development Approach had been chosen as the methodology for this project. The main reason for choosing this model is to ensure the fulfilment of the specifications and requirements of the client that include frequent changes and modifications through improvement of the hostel management system.

In Object-Oriented methodology, the main phase in Object-Oriented System Development Approach is as below:

1.3.1 Object-Oriented Analysis (OOA)

This phase produced a conceptual model and system requirements implied as an object model which is the population of cooperating objects of a system. For instance, the individual who will be using the hostel management system is hostel management and parents.

First, process model which includes activity diagrams and use cases where all functionalities of the system are developed. For instance, room allocation for the student is the process that will be taken by the hostel staff on the system. For activity diagram, the process and flow in the system will be identified. Then, the interaction diagram is prepared follow by Unified Modeling Language (UML) class diagram that will identify classes, attributes, methods and associations. Finally, it is iterate and refine if necessary.

1.3.2 Object-Oriented Design (OOD)

This phase will transforms the analysis classes into a computerized model that fit to the solution space. Initially, design axioms are applied where the static UML class diagram which are done in OOA phase are refined and completed by adding up more information to the diagram. After that, access layer and view layer classes are designed. The purpose of access layer is to create classes that communicate with the data source while view layer involves designing the interface for displaying and obtaining info through the system. For instance, in the access layer, there is the primary and foreign keys in the database table such as name, ID card number, and other relevant data whereas the view layer will include the forms & data entry windows, dialog boxes and application windows. All of the process will be iterated and refine.

1.3.3 Object-Oriented Implementation

The final phase is the implementation phase. This phase activities includes developing the source code, testing and maintenance. A prototype will be built to test the usability and the user satisfaction of the system. Lastly, prototype of the hostel management system that contains the standard Object-oriented Programming feature will allows better reusable codes and produces system that is more organized, easier to maintain, alter and debug.

1.4 Scope

Hostel management system is generally developed for hostel management and the parents of the students. Parents can help their children to apply for the hostel room using this system. This system facilitates application procedure by displaying the application form online.

Furthermore, hostel management would be able to retrieve the data which shows the details of the students in each room, the empty room available in the hostel and also the hostel staff details. The hostel administrator can perform the adding, modifying and deleting operation in the system.

This particular project deals with the problems on managing a hostel and prevents the problems which occur when worked manually.

1.5 Significant of project

Referring to the difficulties that are faced by the hostel management and the student's parents in manual file system, it is essentials to have a computerized Hostel Management System. As the computerized system, the parents can easily apply the hostel online and no need to fill the paper application every year. The hostel management can easily manage and identify the identity of the students for each room clearly and effectively. Finally, reports regarding the room availability and student information can be generated for backup data.

1.6 Project Schedule

The project schedule will include the progress and development for Final Year Project 1 and Final Year Project 2. Please refer to Gantt Charts in Appendix for the complete project schedule.

1.7 Expected Outcome

This system can reduce manual effort by storing the student information and staff information effectively, enabling the parents to help their children to apply the hostel online and improves the management work for the hostel. The hostel management can also manage the hostel in a more systematic way. Thus, this system will provide a lot of benefits for both the parents and hostel management team.

CHAPTER 2

BACKGROUND STUDY

2.0 Introduction

As the number of student staying in the hostel is growing, number is very important in develop a system to manage all the hostel room application and allocation which can resolve the issue or problem areas face by the management with the current manual work managing the hostel. This section will cover the review on the proposed system and other three similar systems, and also the comparison of those systems for solving the problems that occurred in the hostels.

2.1 Review on SMK BAU Hostel Management System using Object-Oriented Approach

Hostel Management System for SMK BAU School is proposed to manage the information of students who stay in the hostel, generate report for room availability, manage room allocation, and allow the parents to apply for hostel online. It is designed to assist the hostel management to manage the hostel information which will also reduce their manual effort when storing the student information and staff information. This proposed system will also store the details include demerit points where the student who had broken the hostel rules will be given points. The development of this system will be focusing on using object-oriented concept. In order to develop a suitable system, it is essential for the developer to gain understanding on the requirement and functionality of the system. Furthermore, the existing management systems should be reviewed and compared from every aspect. Thus, literature review on the existing systems is very important and should be carried out. There are several systems on Hostel Management System that have been chosen for review and comparison.

2.2 Review on CUSAT Hostel Management System

This system is a hostel management system that is developed for hostel of COCHIN University of Science and Technology (CUSAT) to manage the various activities that are conducted in the institution. The increasing number of students who study in the institutions had led to the needs of designing the hostel management system to computerize the existing system which was done manually so that accommodation problem can be solved easily. In this case, the efficiency of the system has been improved through designing a system that is more users friendly and more Graphical User Interface (GUI) oriented. The use of computer is very important to help the staff to manage the hostel activities systematically and to avoid the occurrence of these problems such as human error, the need of strength and strain of manual labor, repetition of the same procedures, low security, data redundancy, difficulties to handle and update data, difficulties on record keeping and hard to generate backup data.

In this system, the hostel managements and the students are able to view the details of the different hostels in the institution. The details of the 9 hostels include the notice boards that consist of the notifications on the changes of hostel fee.

The hostel authorities could assign different students to the different hostels, vacate the students from the hostels, control the status of the fee payment, and also edit and modify the information of the students. For the students, they are only allowed to view the details of the hostels and access the registration page which the students will be able to fill the registration form online and the filled form can be printed out by them. The hostel authorities will verify the registered students before allocating them to their hostels.

The hostel management system also includes the link that can direct the user to the CUSAT official website. The added links at the system enable the system interrelation

between the official website and allow the users to gain access to both sides. This could provide the more effective and faster way for the users to access the official website.

Apart from that, the hostel management system developed for CUSAT is using the PHP and MySQL. The installing of full PHP/MySQL server environment that also known as PHP Triad which include PHP, MySQL, Apache, and PHPMyAdmin will make the system provide flexibility to have the potential for future enhancement. The software development methodologies that have been used in the system are System Development Life Cycle (SDLC) that consists of four main phases which is analysis, design, implement and testing.

2.2.1 The Major Functions of the CUSAT Hostel Management System:

- Allocation of the hostels
- Vacate the rooms
- Delete the record
- Display the Student administration division of hostels
- Control the status of the fee payment
- Edit and modify the information of students
- Register Online
- Display Notice Board