

AN AUTOMATED HOSTEL MANAGEMENT SYSTEM

ARANSIOLA MATHEW AYODEJI

AN AUTOMATED HOSTEL MANAGEMENT SYSTEM

**ARANSIOLA, MATHEW AYODEJI
15010301006**

**BEING A PROJECT SUBMITTED IN THE DEPARTMENT OF COMPUTER SCIENCE
AND MATHEMATICS, COLLEGE OF BASIC AND APPLIED SCIENCES
IN PARTIAL FUFILLMENT OF THE REQUIREMENTS FOR THE
AWARD OF DEGREE OF BACHELOR OF SCIENCE
MOUNTAIN TOP UNIVERSITY, IBAFO,
OGUN STATE, NIGERIA**

2019

CERTIFICATION

This Project titled, **AN AUTOMATED HOSTEL MANAGEMENT SYSTEM**, prepared and submitted by **ARANSIOLA MATHEW AYODEJI** in partial fulfilment of the requirements for the degree of **BACHELOR OF SCIENCE (Computer Science)**, is hereby accepted

_____ (Signature and Date)
Dr. Olaniyan O.O.
Supervisor

_____ (Signature and Date)
Dr. Akinyemi I.O.
Head of Department

Accepted as partial fulfilment of the requirements for the degree of BACHELOR OF SCIENCE (Computer Science)

_____ (Signature and Date)
Prof. Akinwande A.I.
Dean, College of Basic and Applied Sciences

DEDICATION

This Project is dedicated to God Almighty

ACKNOWLEDGMENT

I would like to acknowledge the almighty God for giving good retentive wisdom, knowledge and understanding. My sincere and profound gratitude to my supervisor Dr. (Mrs.) Olaniyan O. O. for her time, effort and guidance throughout the course of this project work with the aim to ensure that this project will be a success. I specially appreciate the Dean of College of Basic and Applied sciences in person of Professor Akinwande A.I and also to thank the H.O.D (Head of Department) of the department of Computer Science and Mathematics in person of Dr. Akinyemi I.O for his guidance during my stay at the university. I want to thank the H.O.D (Head of Department) of computer science, University of Lagos, in person of Dr. Okunoye O.B. for his time and corrections on this project work. I would also like to thank the project coordinator in person of Dr. Alaba O.B. for his constant support and advice, I would like to thank all the lecturers of Department of Computer science and Mathematics Dr. Kasali F.A, Dr. Oyetunji M.O, Mr. Falana O.J, Mrs. Taiwo O.O, Dr. Oladeji F.A, for their advice and their words of encouragement. I also want to thank all the non-academic staff of the Department of Computer Science and Mathematics for making my stay a worth while one.

I would like to thank my Mother in person of Chief Matron Aransiola M.M for her financial support and words of encouragement, also I would like to thank my godmother in person of Mrs Funke Michael-Adenuga for her support towards me throughout my stay in this University, and also I would like to acknowledge my sisters for always being there for me and for always supporting me in all my project works, I pray that God will bless all of you abundantly and grant all your heart desires.

ABSTRACT

‘AUTOMATED HOSTEL MANAGEMENT SYSTEM’ is a software developed for managing different hostel activities. Within the past few years’ educational institutions are increasing fast. Therefore, the need for hostel is also increasing, by which accommodation is needed for the students studying in their various institutions, here the case study is Mountain Top. This project shows the design and implementation of an automated hostel management system aimed at creating a database that would store and manage the hostel management system data, and also creating a platform for applying and booking of rooms through the application rather than using the physical method of application, which is known to be always stressful and not as efficient as the automated method.

The methodology used in this project work was accomplished by using suitable software development cycle. Brackets was used as the development environment for the backend of this project and XAMPP Control Panel (Cross-Platform Apache MySQL PHP and Perl) for the database environment. Moreover, the authentication involves using both username and password to enable the user access the hostel management portal

The result gotten for this project led to an application, which can now allocate hostel rooms to students automatically, without the need for any physical application, or using of biro and paper for application, and it also keeps the data’s safe

It was concluded that without implementing this automation application, most schools will still find it stressful to allocate rooms to students using the physical method rather than the automation method, and they’ll still be struggling with the record keeping which would have to include having large files and a room for storing such records

Keywords: Hostel, Automation, Allocation, File Management, Data, Database, Application

TABLE OF CONTENTS

Content	Page
Title Page	i
Certification	ii
Dedication	iii
Acknowledgment	iv
Abstract	v
Table Of Contents	vi
List of Tables	ix
List of Figures	x
CHAPTER ONE: INTRODUCTION	
1.1 Background to the study	1
1.2 Statement to the problem	1
1.3 Aim and objectives	2
1.4 Scope of the study	2
1.5 Significance of the study	2
1.6 Definition of terms	2
1.7 Organization of subsequent chapters	4
CHAPTER TWO: LITERATURE REVIEW	
2.0 Introduction	5
2.1 History of Hostel	5
2.1. Impact of Hostel	5
2.2 Existing system	11

2.2.1	Analysis of the existing system	11
2.2.2	Problems of the existing system	12
2.3	Review of information system	12
2.3.1	Management information system	12
2.3.2	Transaction processing system	13
2.3.3	Decision Support system	13
2.3.4	Workflow automation	13
2.3.5	Process reengineering	13
2.4	File organization	14
2.5	Review of related words	16
2.6	Summary of literature review	16

CHAPTER THREE: METHODOLOGY

3.0	Introduction	17
3.1	Requirement gathering	17
3.2	Implemented system	17
3.3	Database structure and development	19
3.3.1	Database design(the table structure)	20
3.4	Input design	20
3.5	Process design	21

CHAPTER FOUR: SYSTEM DESIGN AND IMPLEMENTATION

4.0	Introduction	24
4.1	System design objective	24

4.2	System design	24
4.2.1	Logical design	24
4.2.2	Conceptual design	25
4.2.3	Physical design	25
4.3	System requirement and model	25
4.3.1	Software requirement and constraints	25
4.3.2	Hardware platform	26
4.3.3	Networking requirements	27
4.4	System development	27
4.4.1	Student login module	27

CHAPTER FIVE: SUMMARY, CONCLUSIONS AND RECOMMENDATIONS

5.0	Introduction	39
5.1	Summary	39
5.2	Conclusions	40
5.3	Recommendation	41
5.4	Contribution to knowledge	41
5.5	Limitation of the study	42

References

Appendix

LIST OF TABLES

Table		Page
3.1	Students table showing students details	20
3.2	Admins table showing admins details	20

LIST OF FIGURES

Figure	Page
3.1 Context-level DFD	21
3.2 Registration process	22
3.3 Admin module	22
3.4 Allotment process	22
3.5 Use case diagram	23
4.1 Showing the Home page	27
4.2 Showing the user registration	28
4.3 showing the User login dashboard	29
4.4 Showing the admin login interface	30
4.5 Showing the book hostel interface	31
4.6 Showing the room details	32
4.7 Showing the students test profile	33
4.8 Showing the Admin interface	34
4.9 Showing the add room interface	35
4.1.1 Showing the Manage rooms interface	36
4.1.2 Showing the registered students	37
4.1.3 Showing the access log	38

CHAPTER ONE
INTRODUCTION

1.1 BACKGROUND TO THE STUDY

Hostel by definition is an establishment which provides cheap food and lodging for specific group of people, and it is also seen as a home for students when staying away from their home. It has large well ventilated dormitories and single rooms and is situated in the school premises (Advanced Dictionary). Providing clean and calm hostel accommodation is one of the key responsible of school management. To manage the hostel facilities, a lot of data need to be maintained such as number of student hostel can accommodate, hostel rules and regulation, hostel fee, hostel in and out of student, guest and visitor record and so on (Ayaka, 2015). So, this needs the system which has an ability to capture all kind of data and information and analyze it properly for smooth functioning of the hostel. Hostel porters and can easily maintain the data. Hostel Management in schools often involves administering of all activities of students. All these still remain difficult and require some job for the top management. Hostel Management functions and responsibilities in modern day schools have always been a problem in managing, because of the manual system method of tools they use. Hostel Management System is well designed specially to meet challenges of administrative set up of any school. Hostel Management System (HMS) can be used to assist in student's allocation, setup hostel information, hostel application, and visitor management. In short, this system would assist the staff in managing some of the hostel activities (Ayaka 2015). Hostel Management system simply implies the way in which the allocation and management of the system is being done. In this context, this project examines how we can automate the allocation of rooms in Mountain Top University, since the University kicked off in 2015, the rate at which students are being admitted seem to increase every session, therefore, there is a need to develop an application which would make the allocation of rooms to the students much more easier than the traditional method being used, this application would

allow students who have payed their fees be given allocation without need for physical application.

1.2 STATEMENT OF THE PROBLEM

In this present age there has been advancement in technology were majority of life applications are automated, whereas discovery showed that an institution like Mountain Top University still use the traditional method (paper and pen) for collection and storing of students' hostel information. As a result, there is high chances of loss of information which could result from human error, improper storage of paper documents or fire outbreak.

In addition, there is usually more stress at the process of student's registration for accommodation where the students have to move from one office to the other, whereas this can be done online. Therefore, this project considers it important to develop an automated hostel management system to help solve the aforementioned problems.

1.3 AIM AND OBJECTIVES

The aim of this research is to design and implement an automated hostel management system using Mountain Top University as a case study. The specific objectives are to:

- i. analyze the existing methods of hostel management
- ii. develop an application for the hostel management

1.4 SCOPE OF THE STUDY

This project focused on the development of an automated hostel management system for Mountain Top University. The modules considered are: room allocation, rooms registration, student reports; and student data records.

1.5 SIGNIFICANCE OF THE STUDY

This project is majorly significant to Mountain Top University Hostel allocation system as it would enhance easy and accurate room allocation to the students. Likewise, this project would

benefit other institutions who would want to inculcate this system in their university website and practice.

In addition, this project would be a useful literature to other researchers, and the source code would be useful to those who intend to develop a similar application.

1.6 DEFINITION OF TERMS

Accommodate: To provide someone with a place to stay

Allocate: To set aside something specifically for a person or something.

Computer data: is information processed or stored by a computer. Computer data may be processed by the computer's CPU and is stored in files and folders on the computer's hard disk.

Computer Program: This is a set of instruction that guides the computer on the action to perform.

Data: Data is any sequence of one or more symbols given meaning by specific act of interpretation, Historically, data referred to known facts that could be recorded and stored on computer media. These are facts made up of text, numbers, images, and sounds.

Databases: A systematically arranged collection of computer data, structured so that it can be automatically retrieved or manipulated. It is also called Databank.

Hostel Management System: Hostel Management System is a system for managing the various activities in the hostel. It is used for managing the hostel information. It manages the student information, room information, room allocation details, fee details, mess bill details and employee details of the hostel

Hostel: An inexpensive place for accommodation usually found in boarding schools

Information: Facts provided or learned about something or someone.

Management Information System (MIS): A management information system (MIS) is a computerized database of financial information organized and programmed in such a way that it

produces regular reports on operations for every level of management in a company. It is usually also possible to obtain special reports from the system easily.

Management: The planning, organizing and controlling of the affairs of an organization, institution or business.

Students: Someone who studies at a school, college or university.

1.7 ORGANIZATION OF SUBSEQUENT CHAPTERS

This work was organized in a way which this chapter one talks about the background to the study, the statement of the problem, aim and objectives, the scope of the study, the significance of the study and the definition of terms, Chapter 2 talked about the literature review of the project, chapter 3 discussed the methodology, chapter 4 discussed the system implementation and design, and chapter 5 discussed the conclusion of this project

CHAPTER TWO

LITERATURE REVIEW

2.0 INTRODUCTION

This chapter briefly discusses the review of the technique used for the existing system, and also techniques used in the development of the new system, here the projects that has been carried out concerning the hostel management system would be discussed, and also researches that had been done on the project work

2.1 HISTORY OF HOSTEL

The hostel is a place where students stay for pursuing formal education away from their homes. But the concept of hostel is not only limited to place of residence, hostel is a human practical laboratory. Therefore, hostel is not simply a place for living it is a center of education. Students learn as much as from their teachers as well as fellows during hostel stay. It enriches the understanding of the curriculum through analytical discussion among the students living in the hostels, and may contribute to character building as well. Students in hostel not only learn the theoretical material they also learn how to enhance their personal abilities and learn to live independently (Mishra, 1994)

2.1.1 IMPACT OF HOSTEL

Living away from family for a specific period of time leaves some enduring experiences in the life of the students. In this new life style student learns to live independently, and learn how to compromise with the other students and roommates (as cited in Khozaei et al., 2010).

Students living in hostels face many difficulties and hurdles such as financial crises, adjustment issues, personal helplessness, distress, changes in eating and sleep habits, and many other issues. Research suggests that Empathy, altruistic behavior, emotional stability will be more in hostel students. Hostel environment gives an opportunity for socialization among students (Mimrot, 2012).

While living in hostel students share their personal ideology with other students, and learn many new ideas from their hostel fellows. Hostel life also influences the students' views and perceptions about the religion. Hostel life also makes students more ambitious, those students who have stayed in hostels are more self-reliant and confident than other students. In hostels students learn courage and spirit from other students, and that may help students to face the practical life more confidently (Ahmad, 2006).

It is a common perception that, hostel life has a unique impact on the pattern of students' life. Living in the hostel makes students socially and behaviorally different. Boarding or hostel life is a combination of different cultural backgrounds, in the hostel life students learn to live with different cultural background people (Shah, 2010).

There is a popular quotation "Times change people changes" it is best applicable for the hostel students. Hostel life going to change the way a student is, its effect on the personality behavior, thinking, and dressing as well. In hostel students are surrounded by other students of about the same age as they are, all those students have different characteristics. In hostel life all students have to adjust to the other students stayed in the hostel (Thakkar, 2012).

Therefore, it leads to mushrooming of hostels for the accommodation of the student's study in this institution, and hence there is the appearance of Hostel Allocation Management System which helps with dealing with the problem of managing hostel and avoid the problem when doing it manually. (Muhammed, Muhammed, Vinod & Prasobh 2009) Hostel Allocation management by manual way is a tedious process, since it involves work load and time

consumption. In this system, we can easily manage the hostel details, room details, student records, easy way of room allocation and hostel attendance. Thus, there are a lot of repetition that can be easily evaded which has reduced the data redundancy (Deepika & Chitra 2010).

Function Features of the system are:

1. Creation of Building and Block information
2. Provision of various room types(single/Twin/Four/six)
3. Room allocation to the student.
4. Transfer of rooms.
5. Room evacuation. (Hostel System, 2006)

Disadvantages of the existing system includes:

1. More human power.
2. More strength and strain of manual labor needed.
3. Repetition of the same procedures.
4. Low security
5. Data redundancy
6. Difficult to handle
7. Difficult to update data
8. record keeping is difficult
9. Backup data cannot be easily generated

Business needs of the system includes five aspects, a summary about these five aspects as follows:

1. **Checking for room availability** – Currently there is no system to check for room availability, and the students have to do so by physically going to the hostel and enquiring. Meeting the

Principal – For any student to meet hostel House keepers and porters, they have to make a trip to the hostel, and many a times, this is a complete waste as the House keepers and the porters are busy or not available. There is no way to know when the House keepers and the porters would be in for meetings prior to going to the office. There is no way also, to make any kind of appointments to meet the House keepers and the porters. Meeting the hostel Principal could be for issues such as major complains, compound appeals, critical hostel matters etc.

2. **Delay Report response** – There are many cases where facilities provided in the hostel might not be functioning properly, or the residence might have another problem with their respective room. In such cases, the residence is required to go to the hostel office and log a report, which would be addressed to, in the near future. The residence at this point has no idea when his report would be taken into consideration, or would it be even considered or not.

3. **Information Exchange** – It is generally a very difficult and time consuming task to get any information from the hostel for any residence. This is because for doing so, they have to go to the hostel, which does not exactly fall under their timeline, as they usually have classes from the 9 a.m. to 5 p.m., which are the exact times these offices are open. At the same time, it is also difficult for the hostel Officer and the staff to propagate any information to all residence, as there is no central hub for them to do so.

4. **Room check in/ check out:** students always get compound because they forget to make check in and check out. The system would make this process an easy task since it is not required to go to hostel office to make check in and check out. Instead of this, students can make login to the system and do this process easily

5. **Compound appeal:** students can make compound appeal through the system. The system which is proposed as part of this project would be an online Information and activities system, which mainly aims to solve all the problems explained above. (HRMS) would serve the main purpose of making it easier for the relevant involved parties (residence, staff and Principal) to interact on various matters related to hostel and their maintenance as well as administration. The interaction would be via this system, which is like a portal for the students to log in and use. For the staff, it would be a portal to get quick updates on all issues and updates that are ongoing in different hostels around the campus, and would help them plan their solutions in an efficient and effective manner.

This project would have some interfaces which includes:

Administrator module:

In administrator module administrator manages the master data's like server details and student details. Accept the application of students, view the application forms, reject the fake applications, view the complaints of the students in the hostel, accept the vacating form and delete from the database, edit the notice boards and view complaints.

Student Module:

In student module, they can Submit application form, change password, can check status, view notice board and submit the vacating form.

View Application Module:

This module is where students can view their application status, if the application has been approved on status below it would appear approved, otherwise it would appear pending.

The Project Deliverable is web-based system, which would be integrated under MYIUM website, which is similar to hotel systems, where students can check room availability and perform booking, payment and many other processes online without any physical interaction which cause time and effort wastage.

Key functionalities of the suggested system are listed below as follows:

The proposed system contains two modules namely

(I) Student module and

(ii) Admin Module.

It is proposed that both work in an integrated manner to enhance work activities and facilitate the functionalities of the various units in the HRMS. The main characteristics of these two modules are presented as follows:

Student Module would contain the following functionalities:

- i. **Log in to Student Profile:** The system shall include a user authorization procedure where users must identify themselves using a login name and password. Only users who are authorized in this way may access the system data
- ii. **Update Student Information:** The system should allow students to update their information after login (only editable areas, such as address, contact info etc.)
- iii. **Checking Room Availability:** The system shall list all available/vacant rooms to allow student apply for room change or to book specified room.
- iv. **Room Booking:** The system shall allow student to book any vacant room he/she wants.
- v. **Report Maintenance needed:** Students can make report on Hostel facilities' failure through the system. This feature of the system would automate some of the present manually carried out reporting procedures.
- vi. **Report Complaints:** The system shall provide a function to the students to report any other kinds of complaints as well.
- vii. **Arrange Appointments:** The system shall allow students to make an appointment with the hostels principal for any specific purpose. For making an appointment, the student would be presented with a form to fill, which would contain the following fields: -Date of Appointment - Reason for Appointment -Choose Timeslot (This would be a calendar slot, whereby the available timeslots would be displayed and the student can select one of them)
- viii. **Room Check in/out:** When the student has completed the room reservation/Booking process, the system shall allow he/she to check in and check out at the end of every semester. Also, the system shall show the status details of the Hostel, Block, Room, Level and compartment registered by the student
- ix. **Notify user for Request Changing Status:** System should also be able to inform users via email if their reports, appointments have been approved/rejected.

Admin Module would contain the following functionalities:

- i. **Log in to Admin Profile:** The system shall include a user authorization procedure where users must identify themselves using a login name and password. Only users who are authorized in this way may access the system data
- ii. **Check Reports:** The system shall present to staff, a tabular view of all issues/defect reports raised by students. This table view can be sortable as well as searchable.
- iii. **Respond to Reports:** The staff should be able to mark the issues as done, pending, or rejected
- iv. **Upload Announcements/ important events:** The system shall allow staff an interface to upload any kind of events/news pertaining to Hostels. This would make the system a one stop information hub for all students.
- v. **Confirm/Reject Appointments:** The system should allow the relevant staff to approve/reject students' appointment to meet the Principal
- vi. **Approve/Reject Room Changing:** The system should allow relevant staff to approve/reject students' application for change of room/hostel. The system would also prompt the staff to enter a reason in case they choose rejection
- vii. **Notify User of New Requests:** System should also be able to inform users via email users get new requests

2.2 EXISTING SYSTEM

The population of students gaining admission to higher institutions is increasing on yearly basis. This is putting enormous pressure on the facilities in these institutions. Adopting the conventional manual scheduling methods to the facility management job is the common practice in most institutions here in developing world. This method is characterized by numerous drawbacks, some of which are human error, low security, data redundancy, difficulty in management, difficulty in data update, difficulty in record keeping, difficulty in data recovery in case of disaster etc. Thus, to enhance the administration of hostel facilities, there is need for application of effective optimization methods and scheduling algorithms on the part of the authority managing the facilities. The proposed system addresses the drawbacks of the existing system and ensures data integrity. The system is more user-friendly and GUI- oriented.

2.2.1 ANALYSIS OF THE EXISTING SYSTEM

In different universities, there are various mechanical methods in which rooms could be distributed evenly to students. The most common method used follows steps;

1. Getting the total number of halls available
2. Taking note of the maximum capacity of each hall
3. Taking note of the number of rooms available in each hall
4. Taking note of the available bed spaces in each room

Most students are allocated based on the first room in the halls, but levels and course of study are also taken into consideration, in some situations, students offering the same courses or on the same level are not allowed to stay together in the same room, while some students are given rooms based on health issues, whatever happens, when the first room allocated is filled up, we'll have to move to the next room, till all the rooms are filled up, then we move to another hall of residence

2.2.2 PROBLEMS OF THE EXISTING SYSTEM

After the comprehensive study, it was noticed that the mechanical way of the allocation of hostels has some drawbacks:

1. Too many operators are involved.
2. It takes a whole lot of time to carry out the whole process of allocating rooms to students
3. The allocation is not effectively done: Some spaces are given unrealized

These drawbacks listed for the current method used in allocation of rooms to students needs to be improved, so that brings us to the development of this new application, which would automate the allocation of hostels to the students

2.3 REVIEW OF INFORMATION SYSTEM

In meeting up with the needs of organizations, many different systems have been developed. In small organizations, a single computer might meet all of the organization's needs, Larger organizations supplement single user systems with minicomputers, mainframe computers, local area networks(LANs)and wide area networks (WANs). Some of these larger systems are designed to assist with group works (team of two or more people working on the same project);

others are enterprise-wide systems, available throughout an organization (including all of the organizations branch offices)

2.3.1 MANAGEMENT INFORMATION SYSTEM

A management information system(MIS) is a computer-based system that supports the information's needed of different levels of the management. This type of systems is designed to help the management make formal decisions. Most management information systems are designed to work with transaction processing systems. They produce reports that tell managers how well they are meeting goals.

Although, management information systems continue to play an important role in organizations, they have drawbacks. They generate predefined reports that may not be available when it is needed and it might be buried in reams of printout.

2.3.2 TRANSACTION PROCESSING SYSTEM(TPS)

A transaction processing system (TPS) also called an operational system or a data processing system handles day-to-day accounting needs of an organization. In commercial organizations, a TPS is often linked with an inventory control system so that sales personnel would know whether an item is in stock. Transaction processing systems date to the earliest years of business computing, and the cost saving they introduced help create a huge market for business computers. A TPS saves money by automating routine, labor intensive record keeping.

2.3.3 DECISION SUPPORT SYSTEM(DSS)

A decision support system (DSS) addresses the deficiencies of management information systems by enabling questions to be asked that cannot be answered by fixed, predefined MIS reports, for example, a retail chain manager can ask how an advertising campaign affect sales of advertised versus non-advertised items. Many DSS applications enable managers to create simulations that begin with real data and ask what-if questions, such as 'what would happen to profits if we went with a shipper who could cut our packaging costs by certain percentage? Online analytical processing (OLAP) provides decision support by enabling managers to import up-to-the minute data from transaction databases.

2.3.4 WORKFLOW AUTOMATION

In most organizations, important documents must go through a series of personnel for approval. For example, in a department of a university, every student collects a course registration form. The students fill the form and take it to a course handler for approval, on approval of all the courses the student is registering for he/she takes it to the class coordinator before the student satisfies the process of registration. Workflow automation can greatly speed this process by automatically sending a document to the next person who needs to see it and approve it.

2.3.5 PROCESS REENGINEERING

Process reengineering, also known as re-engineering, refers to the uses of information technology to bring about a major change in an organization's structure. At the core of re-engineering is the principle that information technology (IT) does not bring big payoffs if you simply automate existing work processes. The key to big payoffs lies in using IT to change existing processes.

In process re-engineering, designers ignore an organizations functional divisions and focus instead on work processes: activities that have identifiable output and value to the organizations clients. For example, management of academic records is a process that involves many functional divisions, including students' registration, course registration, result compilation, analysis and storage and report generation. By employing information technology such as electronic academic records management system, a school's academic records system can be re-designed around work processes rather than functional divisions. The work is then carried out by a cross-functional team, in which team members perform functions that were formerly separated out among the company or the school's functional units. For example, at University of Lagos, each semester begins with the collection of course forms by students and registration of courses by taking the course forms round lecturers for approval by signatures. This process can be re-engineered through the simple expedient of placing workstations with the electronic academic records management system software installed right on the course coordinators' desk where students can register for all courses at a time. Then reports of students registered for each course can be generated, printed out and sent to the course handler or lecturer, invariably, a major bombardment in lecturer's offices with the traffic of registering students can be eliminated.

Re-engineering can lead to big payoffs, but it usually gives redundancy of some employees. Most experts agree to the use of a more modest approach, building on existing information systems if possible

2.4 FILE ORGANIZATION

The advent of modern software was celebrated for its unique characteristics of been able to store information for a considerable long-time. Initially, the major method of storage, which could be relied upon for a period, was physical records, files and magnetic tapes that were rather cumbersome. Thus, the introduction of modern devices such as compact disc, Flash drive, External hard-drive and so on, information storage on long term basis got into a new dimension. Information security and protection was a necessity since the first time a programmer has actually wiped out an already debugged program, was loaded with other one still under developing due to problems yet to be ascertained with the latter.

Today, one of the fundamental functions of modern operating system is protecting information, without which the present day increases use of large databases and online information system would fail.

Ideally, file system facilitates the storage maintenance and working with collections of information existing at two distinct levels, the entire file and its component records. However, sophisticated application software system deals with more complex data relationship providing not only storage but also manipulation of data, thus, creating a complete data management system. File and data management functions can be considered as part of a basic operating system or as independent system. They are an integral part of most software systems today, In fact, one of the main features of many contemporary systems is their provision for long-term storage or large databases, thus, the fundamental of file and data management are extremely important.

In short, a file is a collection of inter-related information, mostly stored in binary form representing characters, core-images of a program or some other information encoding devices. Files can be divided into records, which can further be sub-divided into fields. Data is

information about entities of interest dealt in a system. A file system provides facilities for working with files of information. Some of these facilities are:

- Create, change and delete files
- Provide symbolic names for files
- Move files between storage devices (copying of files)
- Move files to and from computational storage
- Keep account of the data of creation and size and protection characteristics
- Permit and control sharing of files between users
- Manage physical space allocation
- Providing a mapping from the symbolic names, known to the user to the physical address necessary to access information.

However, for the purpose of this project some of these mentioned facilities would be used

2.5 REVIEW OF RELATED WORDS

1. **Room Allocation-** It maintains different type of room allocation (one man/two man etc.) depending on the waiting list
2. **Hostel Account management system-** Separate Hostel account would be maintained for each student. It would be linked with central account system as well
3. **Inventory management system-** It would control the allotment of Tables, Chairs, Beds etc. to rooms
4. **Fee and fine collection-** Any fine or fee can be collected, receipt can be issued through the system.
5. **Reports-** Waiting list, allocation list, Fine report, due list report etc. can be generated by the system

2.6 SUMMARY OF LITERATURE REVIEW

Is a review of literature relevant to the research topic; introduction to the chapter, overview of hostel and brief history of hostel would be discussed. Also to be discussed are the type of hostels and a quick look on several literatures on this work before ending with my proposed contribution.

CHAPTER THREE

METHODOLOGY

3.0 INTRODUCTION

This chapter discussed the methodology used in the achievement of this project work, Requirement gathering, implemented system, database structure and development, database design, input and process design and so on, we also considered the methods used for the creation of the application that is meant to automate the allocation of the hostels for the students of the university.

3.1 REQUIREMENT GATHERING

The way the requirement of the system that is being used and the one that is to be created were thoroughly gathered through personal interview with the hostels House keepers, portals, security

personnel's and other hostel workers, by knowing their views and what is wrong with the current system, and what improvement that can be made to the new system that is about to be created.

3.2 IMPLEMENTED SYSTEM

The system was based on different system programs and software's

Technology Implemented: Apache Server

Language Used: PHP 5.3

Database: My SQL 5.5

User Interface: HTML, AJAX

Web Browser: Chrome or Internet Explorer 8(or newer)

Software: XAMPP Server

Operating System: Windows XP or higher versions.

This project work is also aimed at the development of a system which would keep records and show information about or in a hostel, this system is designed to help the officers (House keepers and porters) manage the affairs of things going on in the hostel, the system provides the full information needed about a student in a hall of residence, it would show available and unavailable rooms and the number of students allocated to a particular room. Also would include the user modules for the officers (Housekeepers and porter). There would be an administrator module which would be accessed by the administrator and has the ability to delete and edit the records of the students. This system was based on Software Development Life Cycle(SDLC) with HTML and CSS used for the front end of the development of the application, while PHP and MY SQL server would be used for the backend development of the application. PHP is good for the development and designs of web based applications and SQL is good in the usage of databases because its security reliant and because the improved, advance property and features.

Here we are going to give basic definition of some terms used in this methodology:

HTML:HTML stands for Hyper Text Markup Language, HTML is the standard markup language for Web pages, HTML element are the building blocks of HTML pages, HTML elements are represented by <> tags.

CSS:CSS stands for Cascading Style Sheets, CSS describes how HTML elements are to be displayed. CSS is a stylesheet language that describes the presentation of an HTML (or XML) document. CSS describes how elements must be rendered on screen, on paper, or in other media. CSS saves a lot of work. It can control the layout of multiple web pages all at once. External stylesheets are stored in CSS files

MYSQL: MySQL is an Oracle-backed open source relational database management system (RDBMS) based on Structured Query Language (SQL). MySQL runs on virtually all platforms, including Linux, UNIX and Windows. Although it can be used in a wide range of applications, MySQL is most often associated with web applications and online publishing. SQL is a standard language for accessing databases. What can SQL do?

- SQL can execute queries against a database
- SQL can retrieve data from a database
- SQL can insert records in a database
- SQL can update records in a database
- SQL can delete records from a database
- SQL can create new databases
- SQL can create new tables in a database
- SQL can create stored procedures in a database
- SQL can create views in a database
- SQL can set permissions on tables, procedures, and views

PHP: PHP (recursive acronym for PHP: Hypertext preprocessor) is a widely-used open source general-purpose scripting language that is especially suited for web development and can be embedded into HTML.

XAMPP: Xampp is a free and open source cross-platform web server stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, Maria DB database, and interpreters for scripts written in the PHP and Perl programming.

3.3 DATABASE STRUCTURE AND DEVELOPMENT

A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently. The general objective is to make data access easy, quick inexpensive and flexible for user. Relationships are established between the data access easy, quick, inexpensive and flexible for the user Relationships are established between the data items and unnecessary data items are removed. Normalization is done to get an internal consistency of data and to have minimum redundancy and maximum stability. This ensures minimizing data storage required, minimizing chances of data inconsistencies and optimization for updates. The MySQL database was chosen for developing relevant databases. In the creation of the database for this project, the most suitable database platform to use for this project was MYSQL (My structured query language) which was used to store username and passwords for the different modules (i.e. the Admin and the Student module), Also the use of PHP (Hypertext processor) would be used to connect to the SQL database.

3.3.1 DATABASE DESIGN (THE TABLE STRUCTURES)

Here we would be looking at the student and admin mode, the structures and the characters meant to use in the development of the proposed system

COLUMN NAME	DATA TYPE	NULLABLE	SIZE
ID	INT	NO	11
STUDENTS NAME	VARCHAR	NO	255

MATRIC NUMBER	INT	NO	11
GENDER	VARCHAR	NO	255
STUDENT LEVEL	INT	NO	255
COLLEGE	VARCHAR	NO	255
DEPARTMENT	VARCHAR	NO	255
COURSE	VARCHAR	NO	255

Table 3.1: Student's table showing students details

COLUMN NAME	DATA TYPE	NULLABLE	SIZE
NAME	VARCHAR	NO	255
ID NUMBER	INT	NO	255
GENDER	VARCHAR	NO	255
DESIGNATION	VARCHAR	NO	255

Table 3.2 Admins table showing admin's detail

3.4 INPUT DESIGN

The design of input focuses on controlling the amount of input required, detecting errors, avoiding delay, avoiding extra steps and keeping the process simple. The system needs data regarding the asset items, depreciation rates, asset transfer, verification/validation checks, calculation and report generation. The error detection method included in the software helps to raise flag when wrong entry is fed into the system

3.5 PROCESS DESIGN

Process design plays an important role in project development. In order to understand the working procedure, process design is necessary. We employed data flow diagram tool for process design. Data Flow Diagram is the logical representation of the data flow in the system. The DFD is drawn using various symbols. Every process has a source and a destination. The

process is represented using circles and source and destination are represented using squares, also arrows are used to represent data flow.

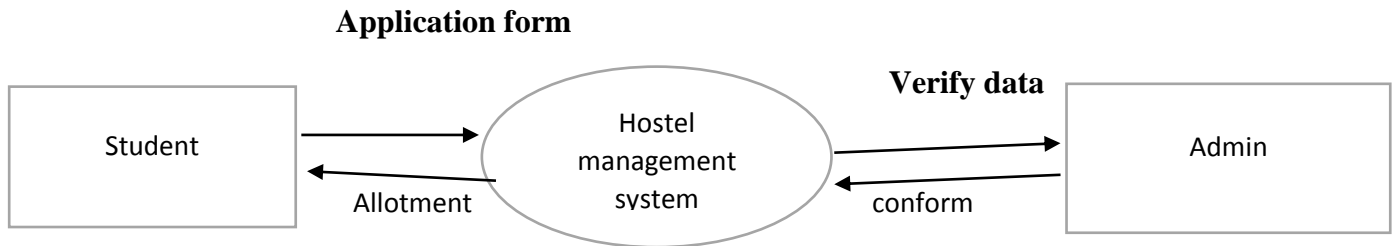
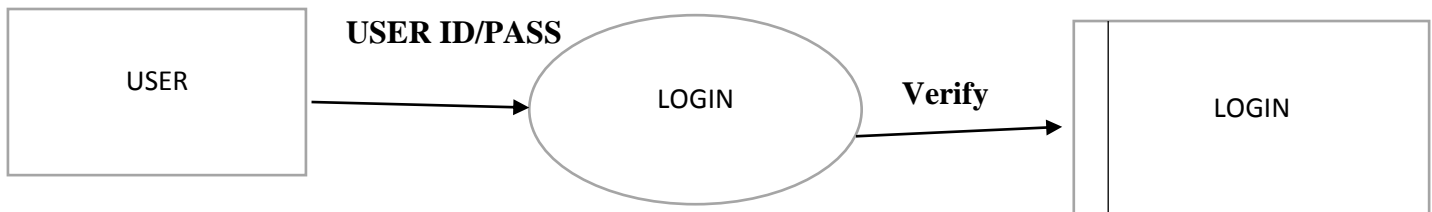
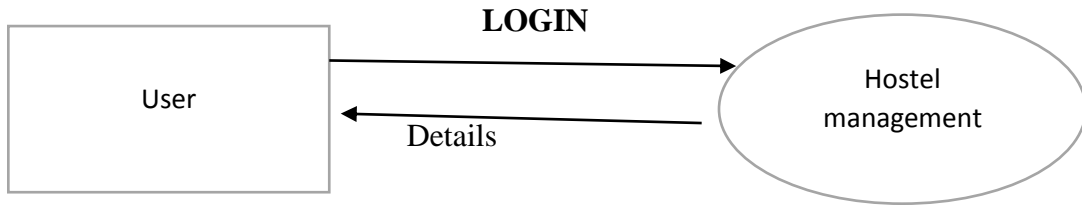


Figure 3.1 Context-level DFD

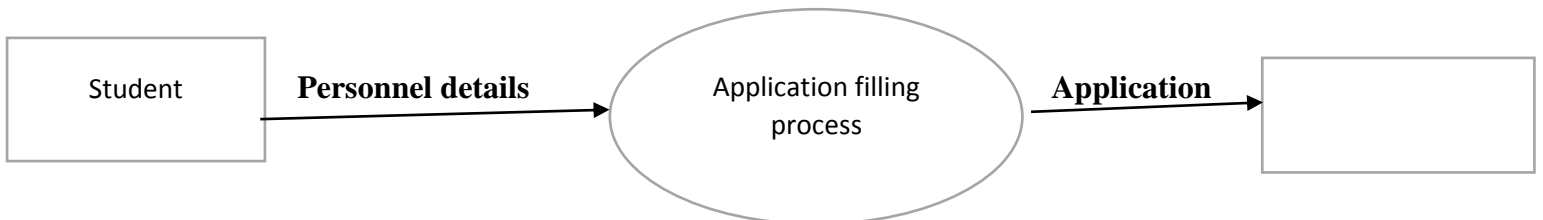


Figure 3.2 Registration process

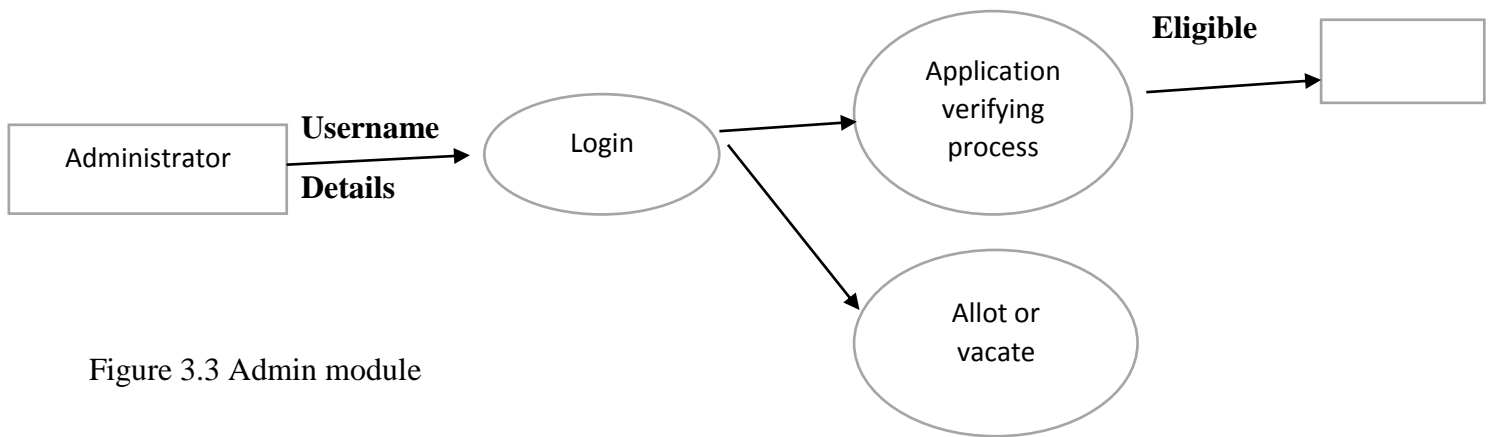


Figure 3.3 Admin module

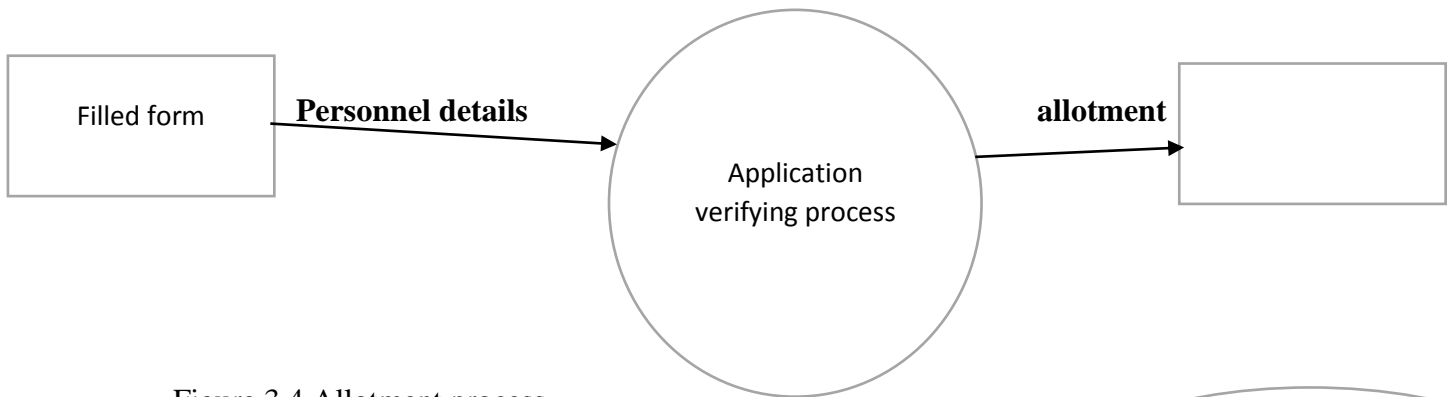


Figure 3.4 Allotment process

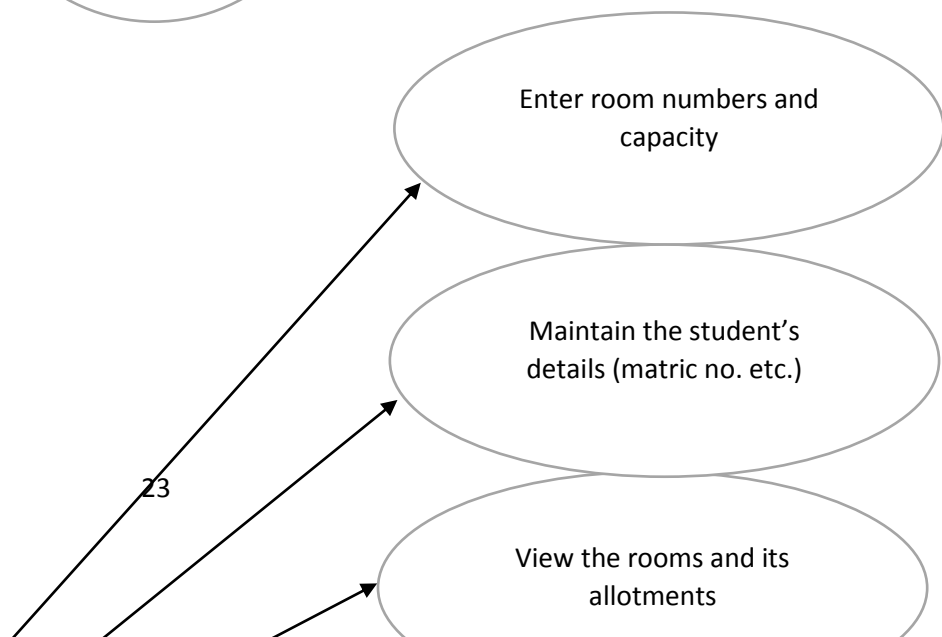




Figure 3.5 Use case diagram

CHAPTER FOUR
SYSTEM DESIGN AND IMPLEMENTATION

4.0 INTRODUCTION

This section talks about the implantation of this project, this chapter discussed about the system design and analysis, it includes pictures from the application and the interfaces involved in the development of the application and also the methods used in the development, the whole object is building a system which are set of things working together as. parts of a mechanism or an interconnecting system, it is a set of interacting or interdependent component forming an integrated whole or a set of elements to other elements.

Some characteristics of a system include:

- i. A system has a particular behavior; it contains processes that transforms inputs to outputs.
- ii. A system has interconnectivity, the parts and processes are connected by structural and behavioral relationship

4.1 SYSTEM DESIGN OBJECTIVE

The hostel management system is a software application that would offer help and assist the students in an effective way, which would give them less stress and trouble in the allocation of their hostel rooms.

4.2 SYSTEM DESIGN

The system design is majorly the implementation of the application to be developed and this is divided into 3 parts:

- i. Logical design
- ii. Conceptual design
- iii. Physical design

4.2.1 LOGICAL DESIGN

This model was developed indicating all the vital steps the system development went through. In this step, used case tools like flow charts and data flow diagrams were used. These models were vital and important in the development of the system. This stage included the graphical user interface design, input design in which the user inputs in data, the output design which displays the results of what a user would or have entered and database design where data is stored for easy management. These designs provided the technical blueprint from which the system was built. A combination of layout tools such as hand sketches and CASE tools were used to come up

with both input, output designs and the view of the system. The database management system employed was MySQL.

4.2.2 CONCEPTUAL DESIGN

This was just a description of the proposed system in terms of a set of integrated ideas and concepts about what the system is expected to do, behave and look like, that would easily be comprehended by the users in the manner intended. This process was started by identifying several entities required by the users and also identifying all the important relationships that exist between the entities. The result was the model of the user interface that has been developed.

4.2.3 PHYSICAL DESIGN

This was the physical realization of logical design. Forms, reports and tables were created and relationships defined among these tables and security constraints set. During the physical the expected schemas were translated into actual database structure.

4.3 SYSTEM REQUIREMENT AND MODEL SPECIFICATION

Usually, when developing software, it is expedient to develop it around the target users of the software. Administrative staff of the university would use this software, so it is important that it requires minimal experience or know-how on their part order to be able to use this system.

Our hall allocation system was implemented using HTML and CSS for the front end of the application and MYSQL database end was incorporated.

It is noteworthy to state that the system is designed to be used in a networked environment. To this end, we shall discuss the different specific requirements of the system

4.3.1 SOFTWARE REQUIREMENT AND CONSTRAINTS

This system was designed using Brackets. Although Brackets allows compilation and packaging of software into executables that can be setup or installed on a system, for reason of this project, the system was let alone with its open-source code state. This then implies that to run the program, the local computer on which it must run must have at least the enterprise edition of Brackets installed on it.

A prior edition of Brackets may not be able to effectively run the program, although a more recent version or Brackets would.

In addition, the database end was developed using MYSQL (my structured query language). This invariably implies that the local machine must have XAMPP server installed. The good thing about XAMPP is that prior versions would support databases built in more recent versions. So this software would run perfectly.

4.3.2 HARDWARE PLATFORM

The section of hardware configuration is an important task related to the software development insufficient random access memory may affect adversely on the speed and efficiency of the entire system. The process should be powerful to handle the entire operations. The hard disk should have sufficient capacity to store the file and application.

- i. Processor: Pentium IV and above
- ii. Processor speed: 1.4 GHz Onwards
- iii. System memory: 128 Mb minimum 256 Mb recommended
- iv. Cache size: 512 KB
- v. RAM: 512 MB(Minimum)
- vi. Network card: Any card can provide a 100mbps speed
- vii. Network connection: UTP or Coaxial cable connection
- viii. Printer: Inkjet/Laser Color printer provides at least 1000 Dpi
- ix. Hard disk: 80Gb
- x. Monitor: SVGA Color 15"
- xi. Mouse: 104 keys US Key Serial, USB or PS/2
- xii. Modem: 56.6 Kbps

4.3.3 NETWORKING REQUIREMENTS

As stated earlier, it is expected that the system is used in a networked environment. By design, it is expected that at least on a machine with a minimum of the specification enumerated above would be in each of the Students Affairs' department, the Students representative council and the

Health center. In addition, all these systems are networked in a local area network (LAN). This may be through optic fibers or radio links. Coaxial cables may not be appropriate regarding the distance between these sections of the institution. With coaxial cables, attenuation sets in at a certain distance

4.4 SYSTEM DEVELOPMENT

The design of the system was discussed in chapter three and the system requirements and model specifications were enumerated earlier in this chapter. Now we shall look into the system development and discuss the hard core coding involved. We would deal with the system window-by-window or form-by-form

4.4.1 STUDENT LOGIN MODULE

a) Home Page

This is the welcome page of the project, it is opened when the URL of the hostel management system, when you open this page, it would bring the User registration, User login and the Admin login interfaces.

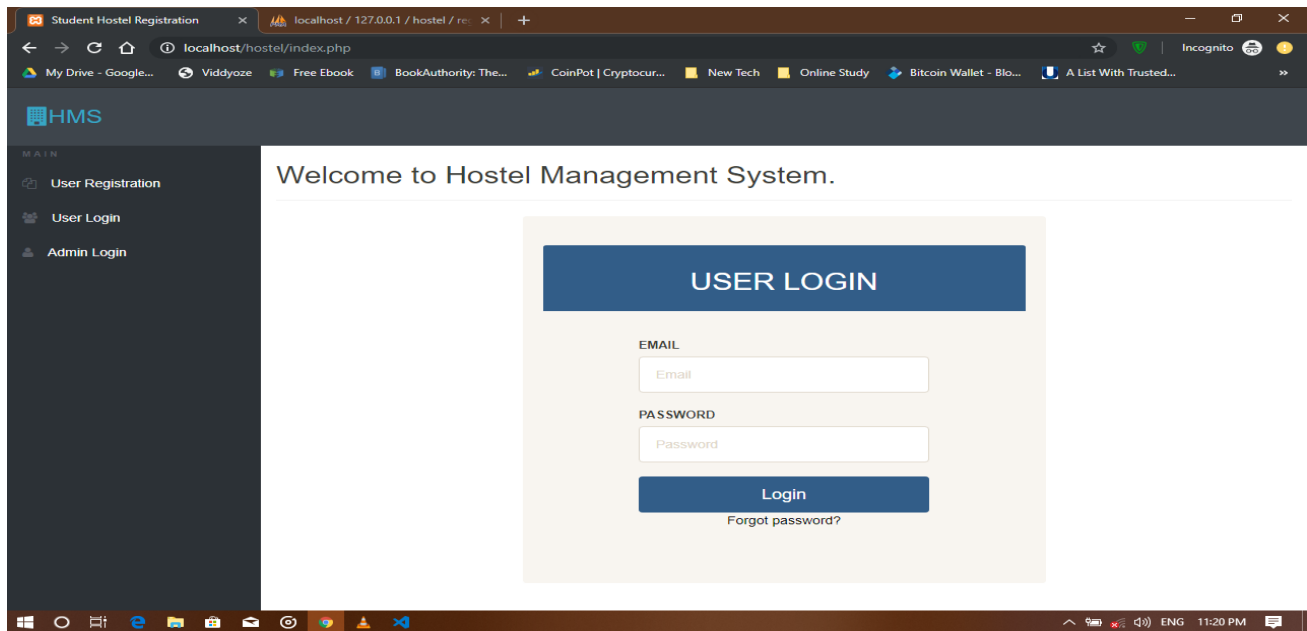


Figure 4.1 Showing the Home page

b) User registration

Before a user can login into the application interface, he/she has to create an account, that is, he/she has to register, this page includes the registration number (matric number), first name, middle name, last name, gender, college, department and level.

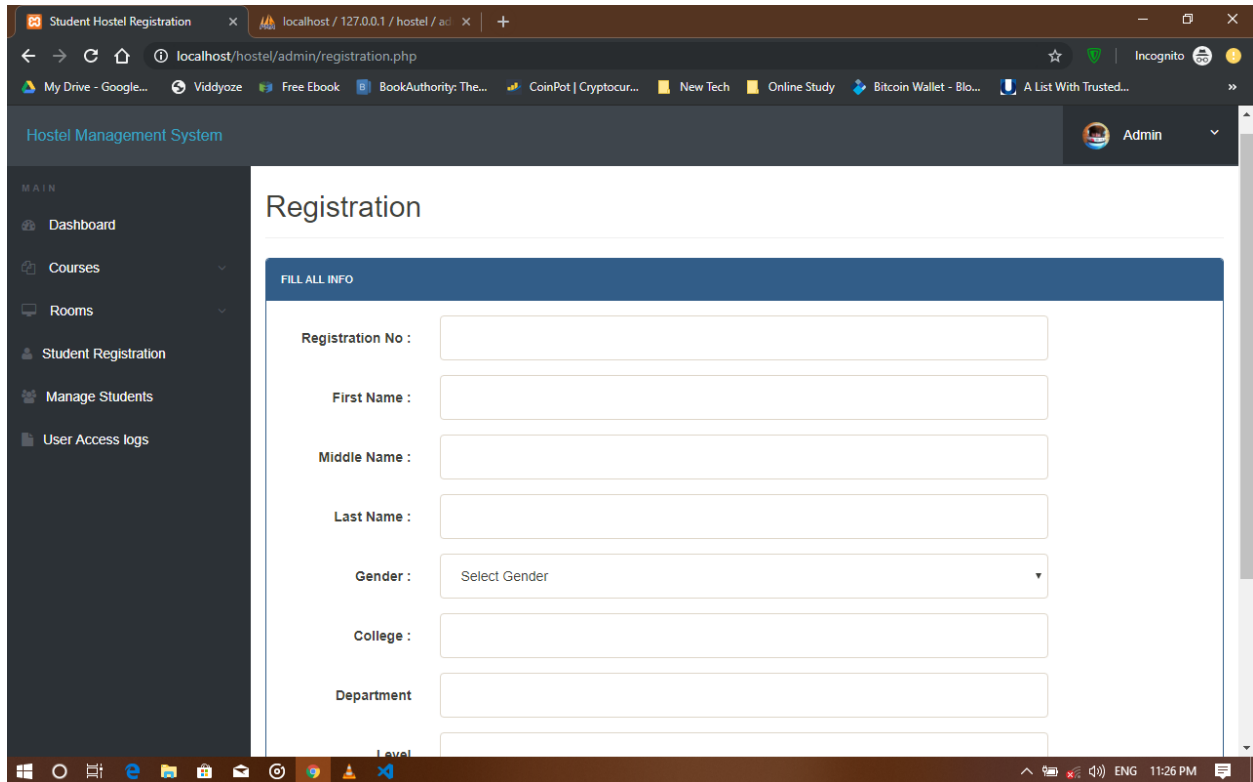


Figure 4.2 Showing the user registration

c) User login

This interface is accessed by clicking on the login, which is on the homepage, and it brings you to the place where you see your profile and also request for room, it comprises of the profile, change password, book hostel, room details and access log

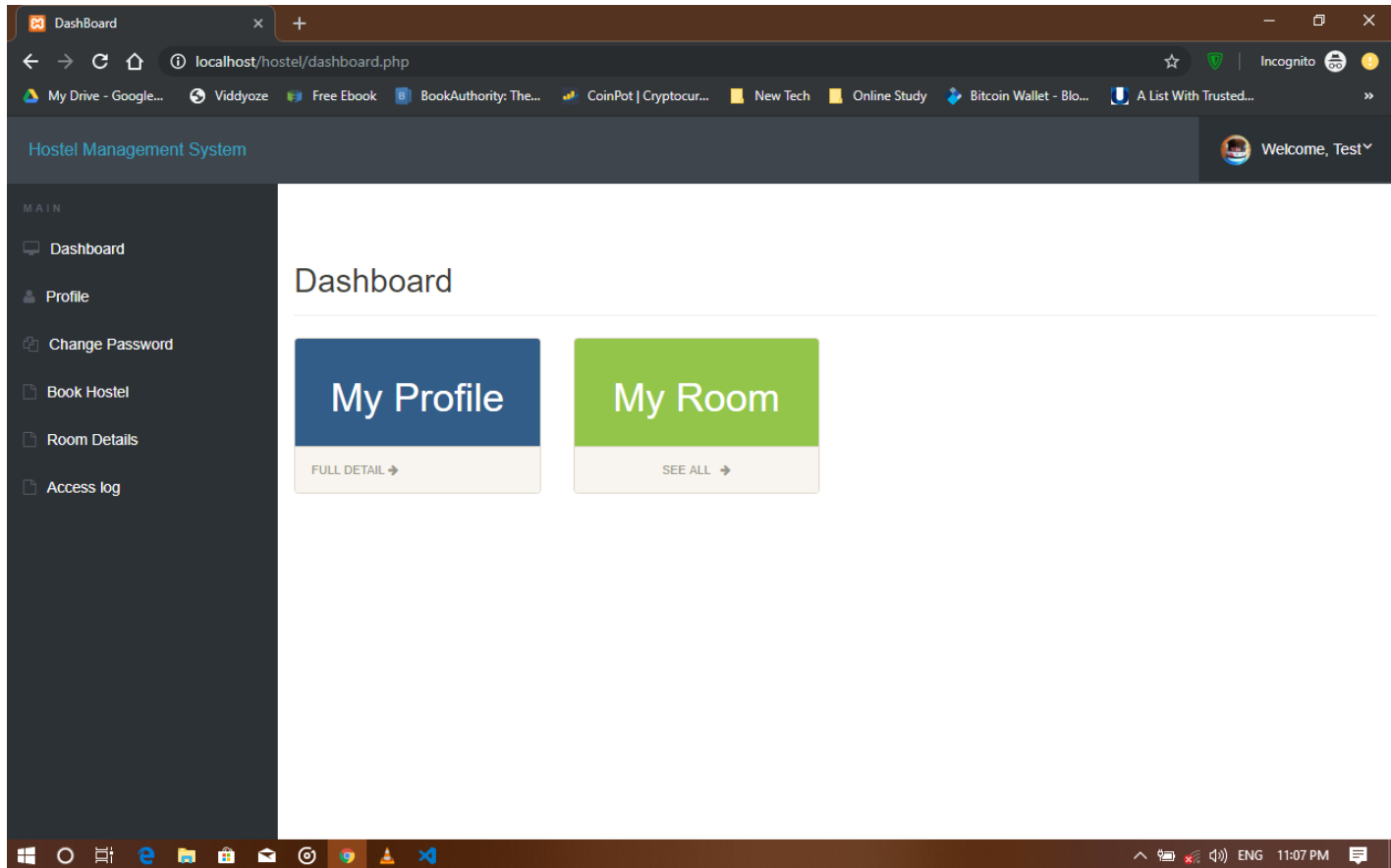


Figure 4.3 showing the User login dashboard

d) Admin login

This is the interface where the admin logs in to check the students that are about to book rooms and the rooms that are being booked

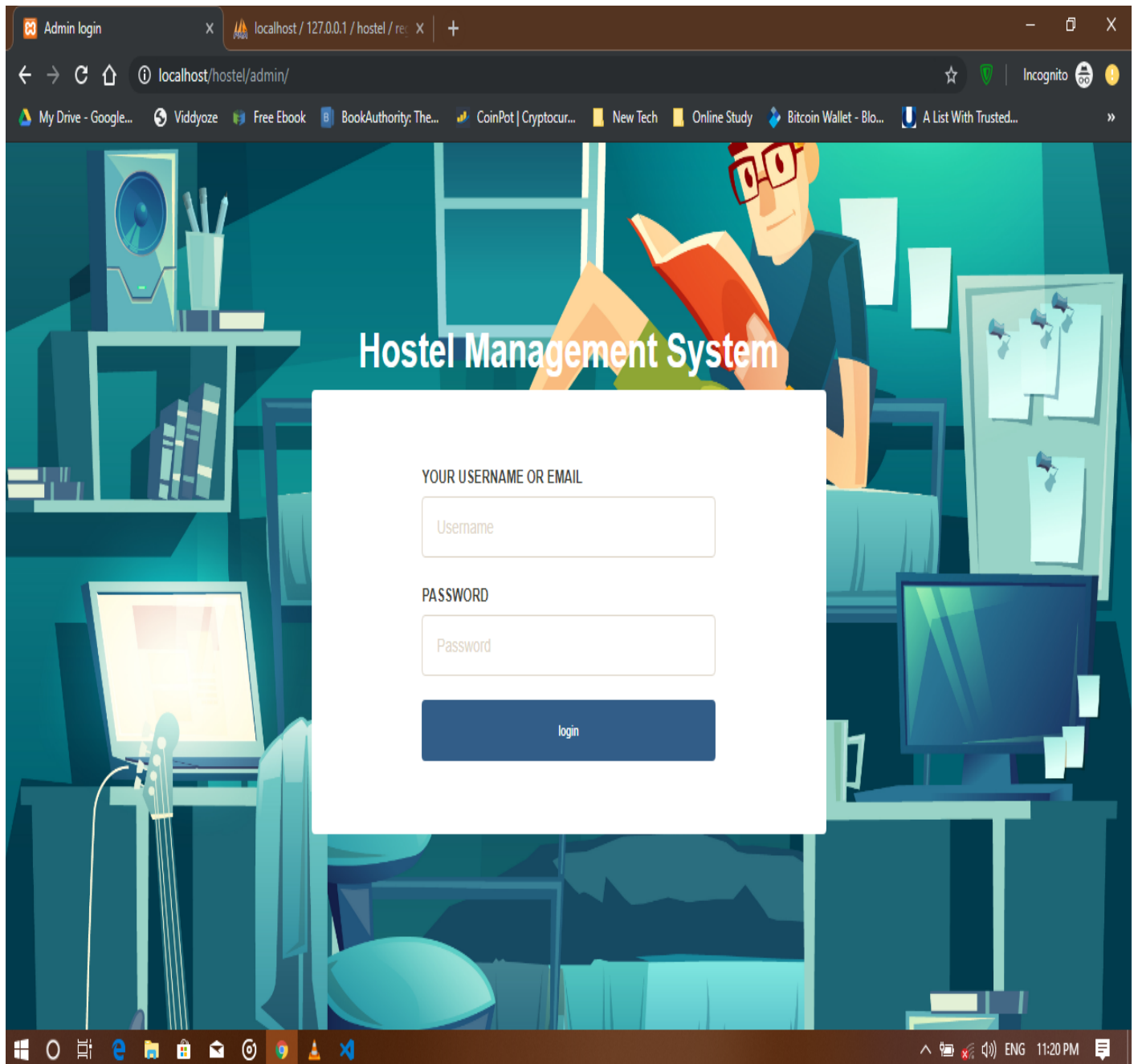


Figure 4.4 Showing the admin login interface

e) Book hostel interface

This is the part of the app, where you request and book for room, it is part of the login interface section

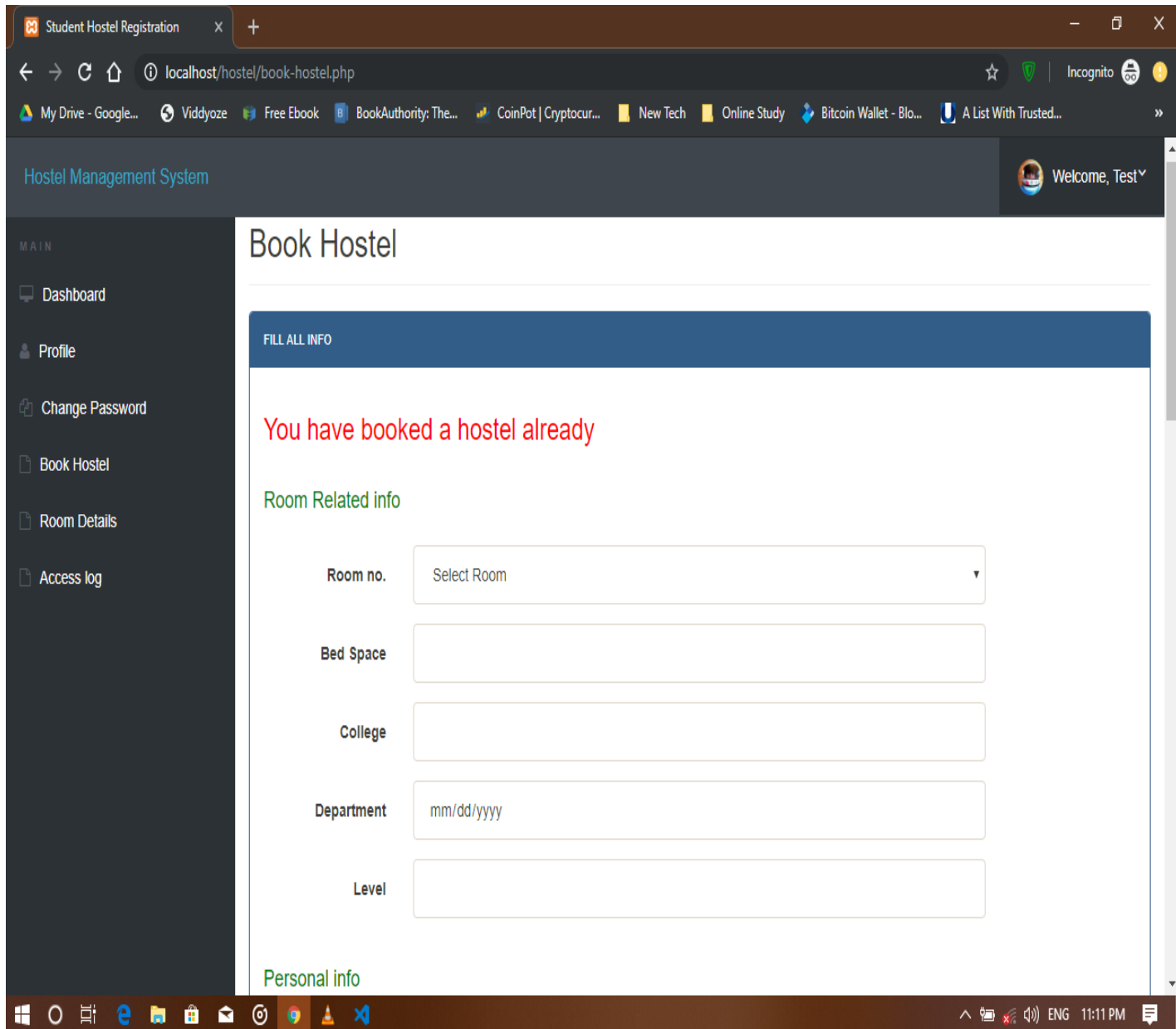


Figure 4.5 Showing the book hostel interface

f) Room details

This is the section where you see the room that you booked and the information about the room booked, and also personal information and so on

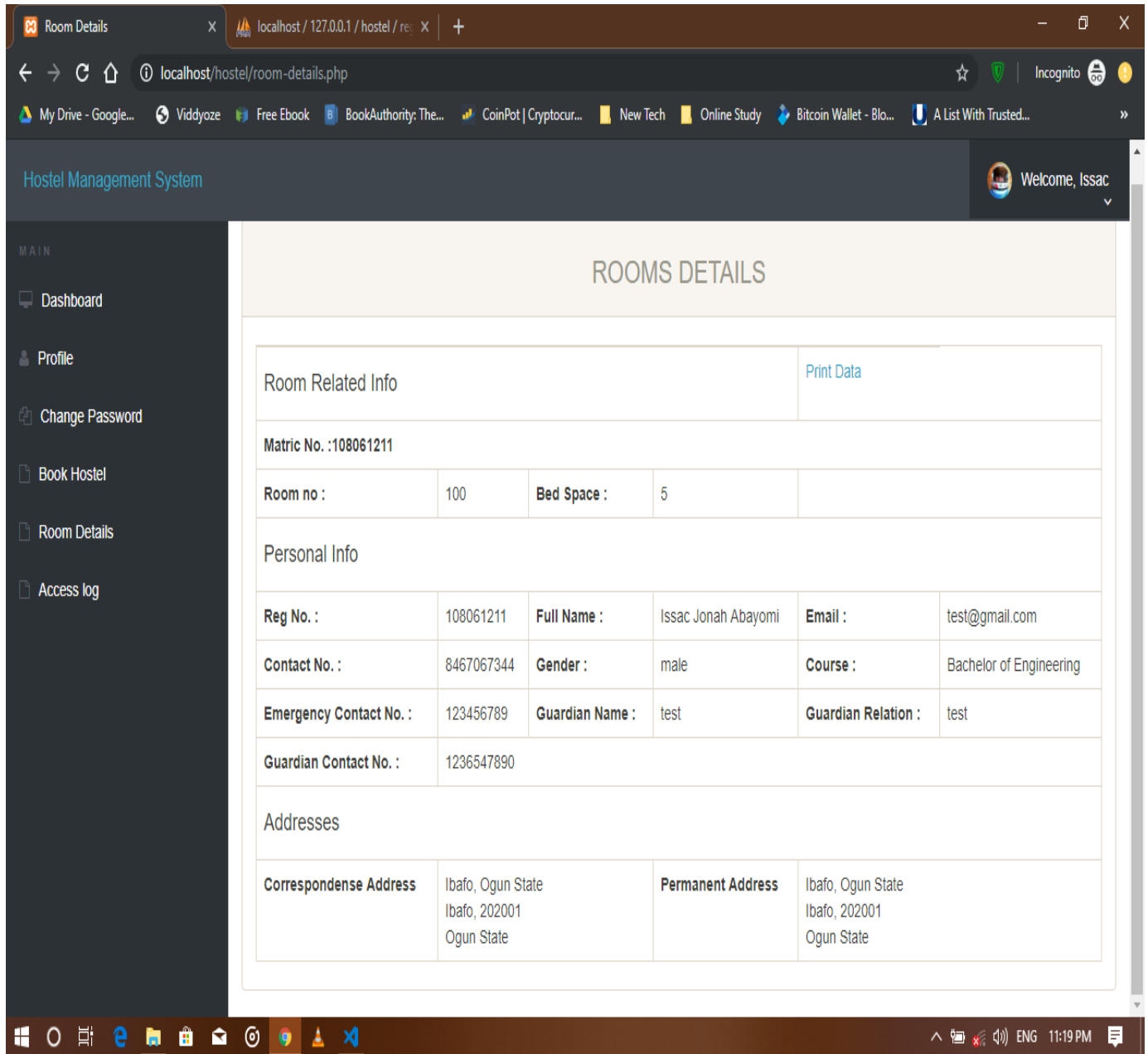


Figure 4.6 Showing the room details

g) Test profile

This section comprises of the filled form from the registration module, it shows the profile of the S

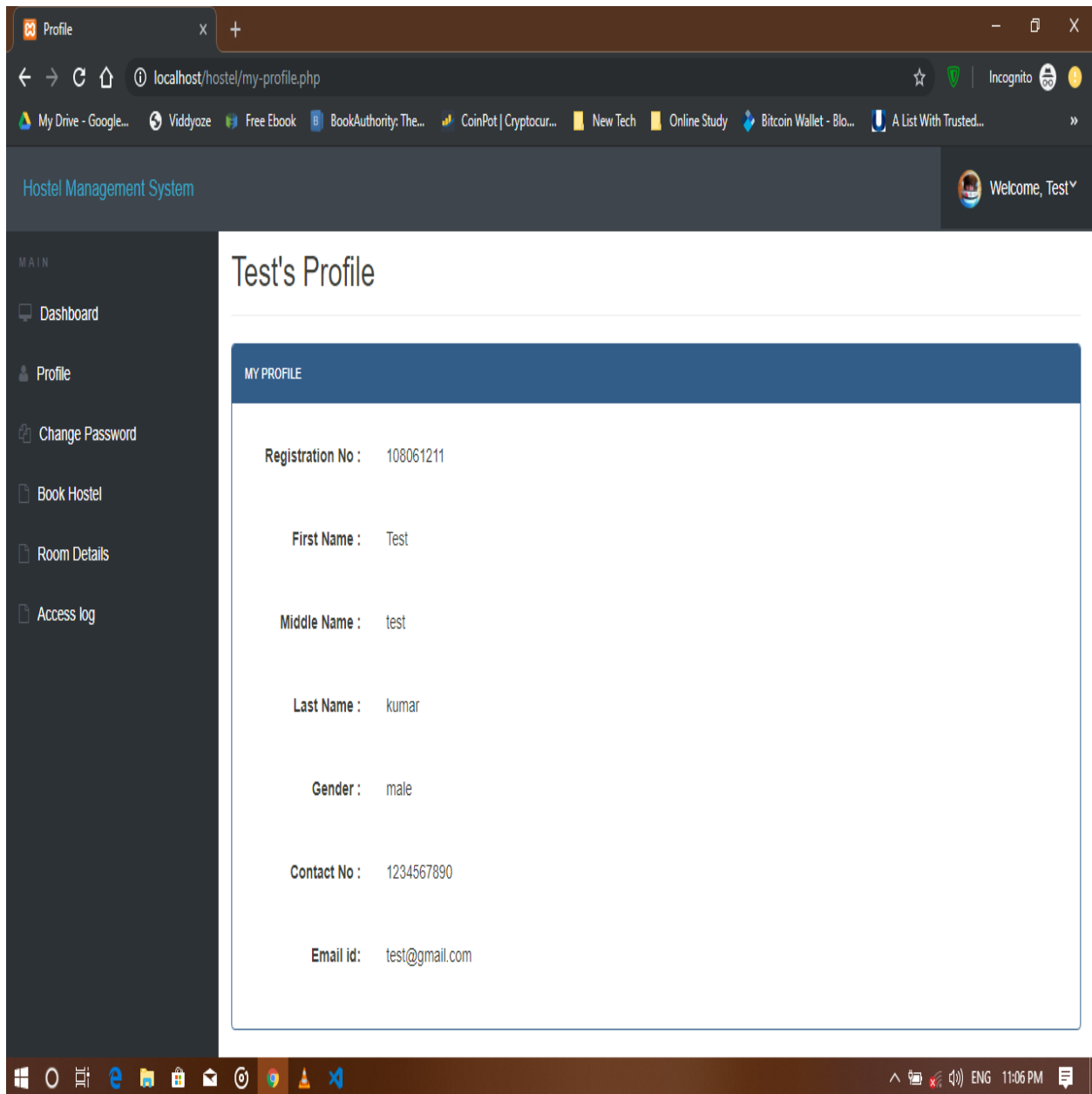


Figure 4.7 Showing the students test profile

h) Admin profile

This module is the page that appears after which the admin logs in to the interface, it comprises of the username, which cannot be changed, the email of the admin, the registration date, and It also has the change password interface in it.

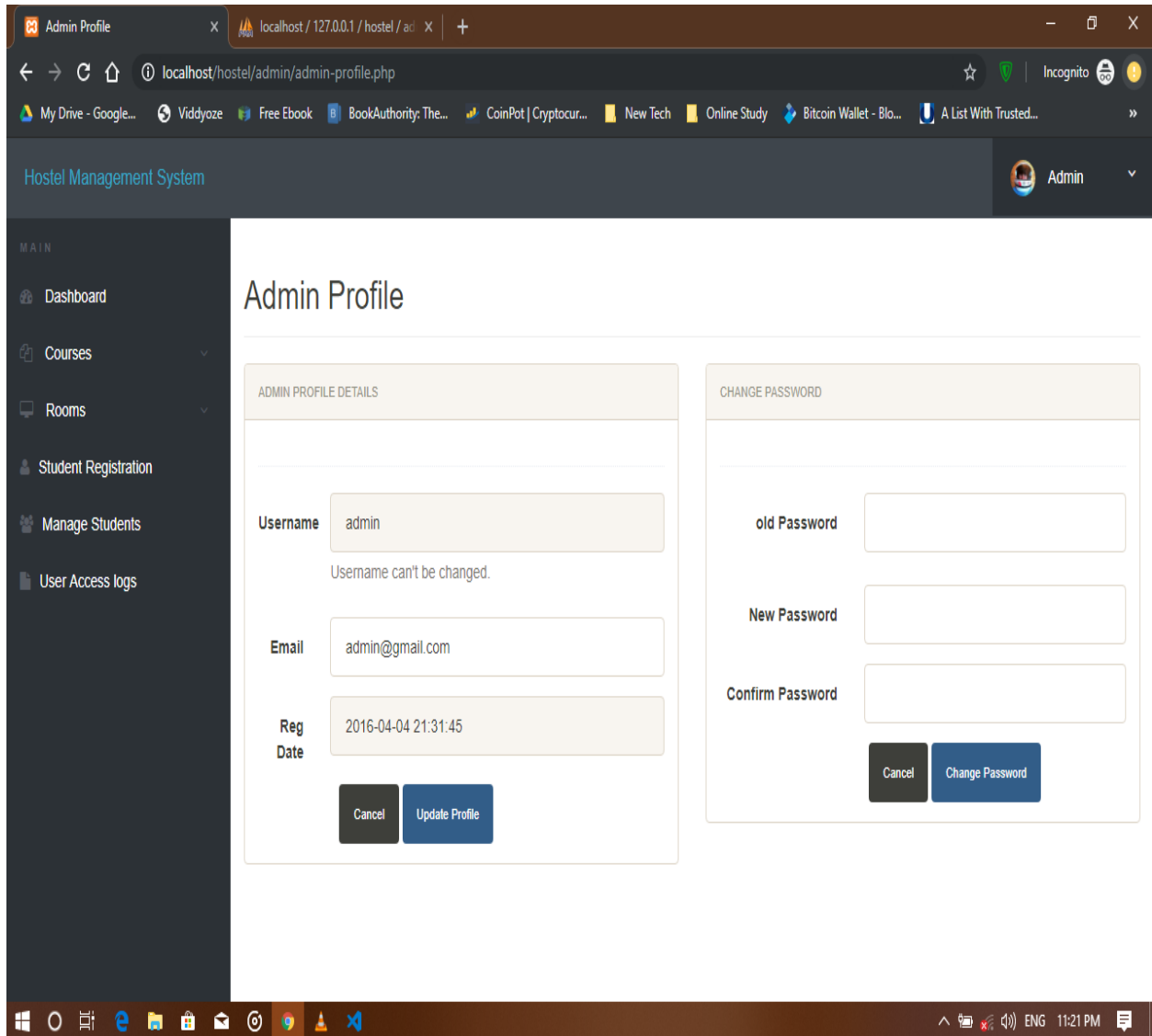


Figure 4.8 Showing the Admin interface

i) Add room

This interface is in the admin region, where the admin adds room, or create rooms that should be included in the rooms that should be booked, while applying for allocation

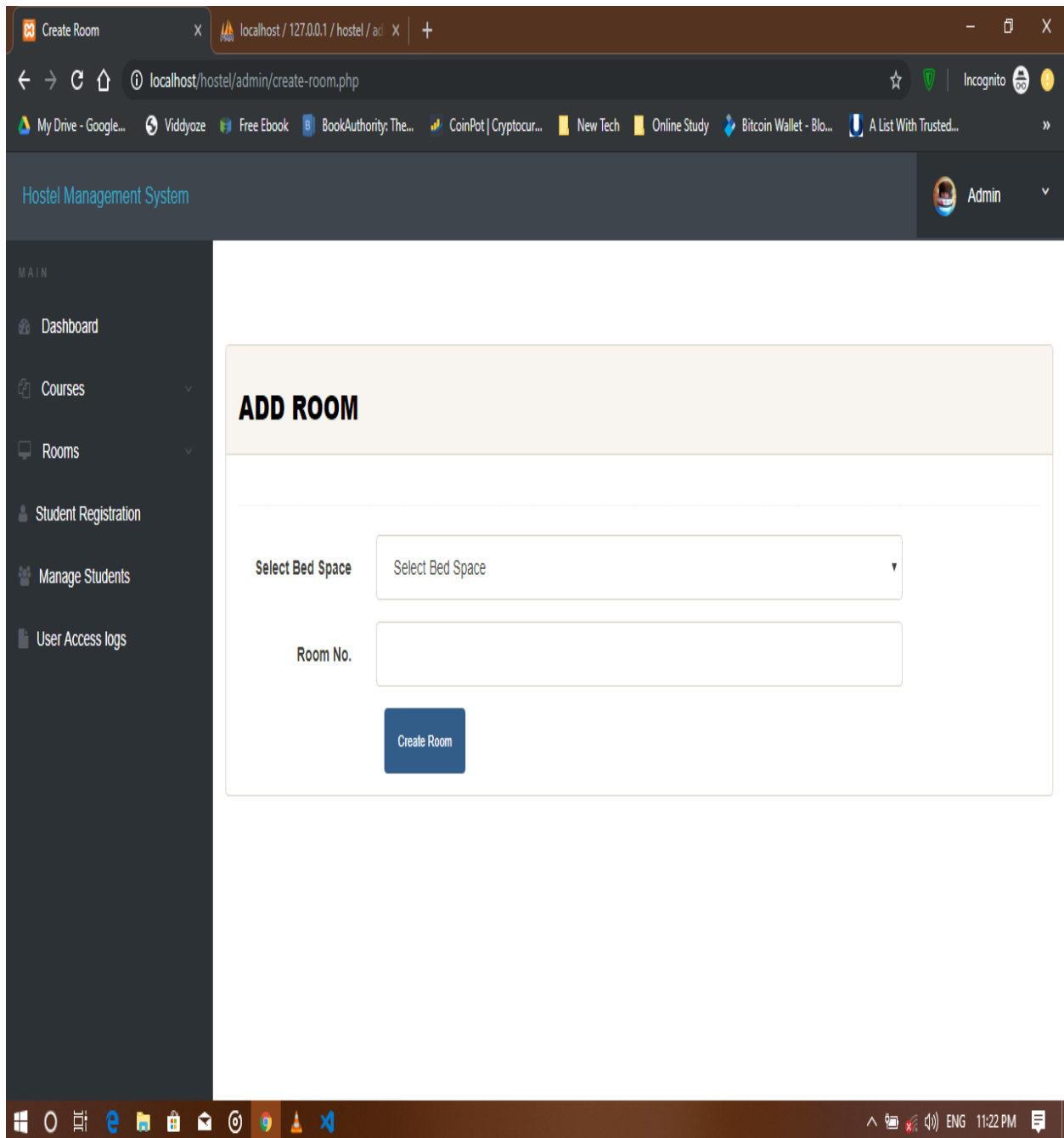


Figure 4.9 Showing the add room interface

j) Manage rooms

This interface shows all the rooms that have been included and created, the ones that have been allocated to students and the ones that are yet to be allocated to students, and also viewing of the application status of the students

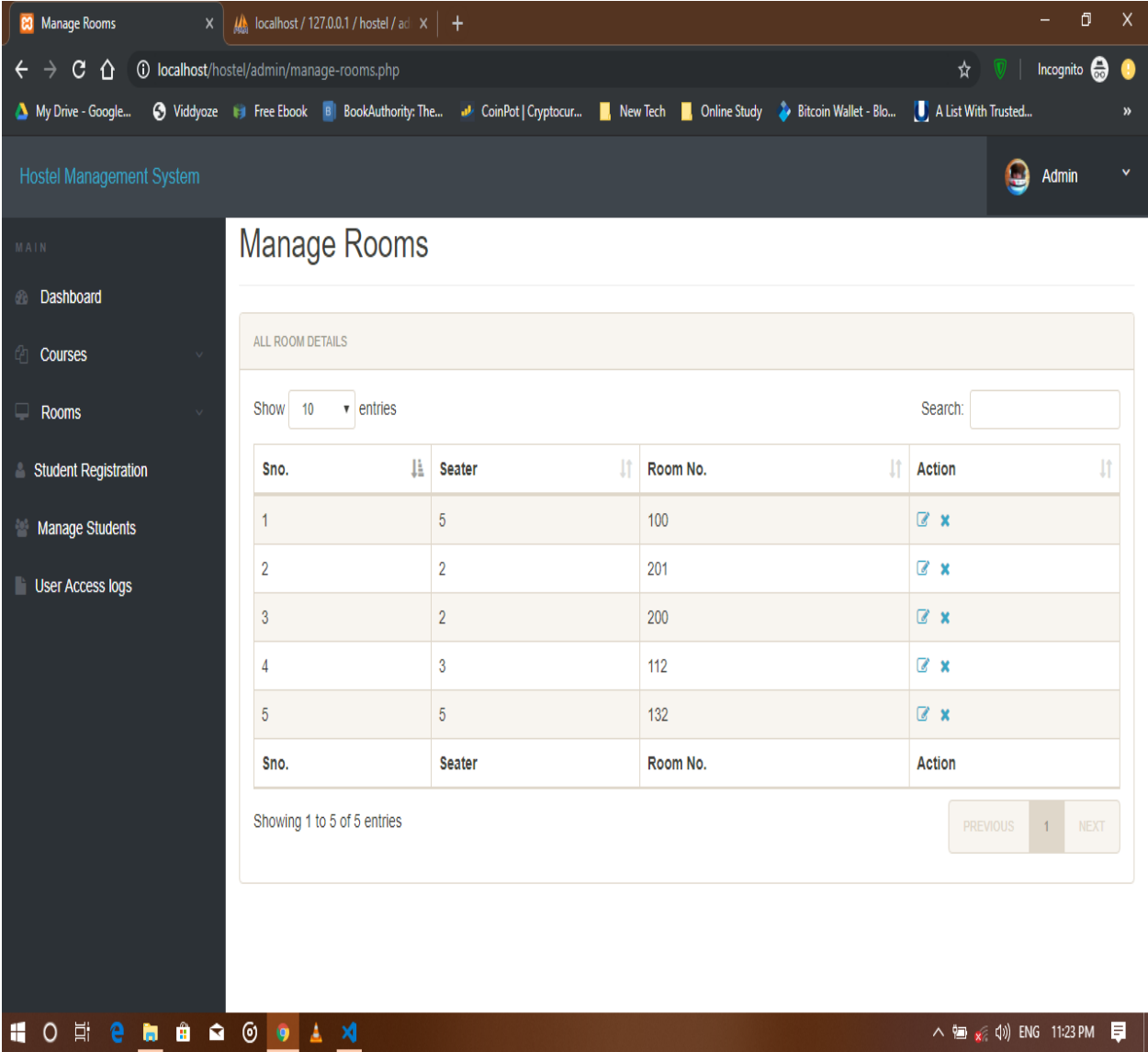


Figure 4.1.1 Showing the Manage rooms interface

k. Registered students

This interface shows the students that have booked and registered for hostel, showing their details, that is, room number, matric number and so on.

The screenshot displays a web application interface for managing a hostel. The main content area is titled "REGISTERED STUDENTS" and contains a table listing student registration details. The table has columns for S/N, Student Name, Matric no, Room number, Bed Space, and Action. There are 4 entries listed. The interface also includes a sidebar with navigation options like Dashboard, Courses, Rooms, Student Registration, Manage Students, and User Access logs. The browser address bar shows "localhost/hostel/admin/manage-students.php".

S/N	Student Name	Matric no	Room number	Bed Space	Action
1	Anujkumar	10806121	100	5	
2	IssacJonahAbayomi	108061211	100	5	
3	rahulkumarSingh	102355	112	3	
4	Ajaykumar	586952	132	5	

Showing 1 to 4 of 4 entries

PREVIOUS 1 NEXT

Figure 4.1.2 Showing the registered students

k) Access log

This interface shows access logs, that is the times a user logs in and log out of the system.

Hostel Management System Admin

Access Log

ALL COURSES DETAILS

Show 10 entries Search:

Sno.	User Id	User Email	IP	City	Country	Login Time
1	10	test@gmail.com				2016-06-22 07:16:42
2	10	test@gmail.com				2016-06-24 12:20:28
3	10	test@gmail.com	::1			2016-06-24 12:22:47
4	10	test@gmail.com	::1			2016-06-26 16:37:40
5	20	ajay@gmail.com	::1			2016-06-26 17:40:57
6	10	test@gmail.com	::1			2019-06-10 06:02:51
7	10	test@gmail.com	::1			2019-06-10 06:49:42
8	10	test@gmail.com	::1			2019-06-10 08:17:32
9	10	test@gmail.com	::1			2019-06-10 09:08:59
10	10	test@gmail.com	::1			2019-06-23 22:54:29

Sno. User Id User Email IP City Country Login Time

Figure 4.1.3 Showing the access log

CHAPTER 5

SUMMARY, CONCLUSIONS AND RECOMMENDATIONS

5.0 INTRODUCTION

This chapter gives a summary about what has been discussed so far, starting from the Chapter 1 of the project to the Chapter 4 of the project, and also it gives conclusions about the entire projects, what has been found out during the research work and the results gotten so far from the research work.

5.1 SUMMARY

This project work titled “AUTOMATED HOSTEL MANAGEMENT SYSTEM” developed, is based on the user requirement specification and the analysis of the existing system, acknowledging the issues with the existing system leads to the new design of a computerized system which would be compatible with the existing system, and it is more user friendly and more Graphic user interface oriented. This research project was done on able to build an online hostel application and allocation platform for Mountain Top University, having a large and robust database, it has to store every information relating to the hostel management system, it also allows updating of information, as it applies to file processing and to the characteristics of databases. So also the online application of hostels by the students is made possible

The system developed allows:

1. Rooms to be allocated automatically
2. Hostel allocation to be done on a computer device through the internet
3. Human error would reduce
4. A layman would be able to use the system without issues
5. The students would be more familiar with the use of the hostel application
6. The student’s data and information is kept safe and highly maintained

Chapter one of this project simply talks about Background to the study, the statement of problem, the aim and objectives, the scope of study, the significance of study and the definition of terms. Chapter two talks about the literature review, the existing system, etc. Chapter 3 talks about the methodology, the system analysis and design, etc. Chapter four simply talks about the

implementation of the system, and here we have chapter five which brings us to the conclusion, summary and recommendation of this project

5.2 CONCLUSIONS

Hostel Accommodation Management System is a user-friendly computer-based system for managing hostel facilities in institutions. It has been designed to automate, manage and look after the overall processing of records of students residing in a large hostel. It is capable of managing Enquiry details, Student Details, etc. The developed system provides solution to manual hostel management problems and also provides information such as hostel information, hostel room information, and hostel accounts information. The software offers stability, cost-effectiveness and usability. It provides the most flexible and adaptable standards management system solutions for hostel. In short about the framework, the project was created utilizing HTML, PHP, JavaScript and MySQL is in light of the prerequisite detail of the user and the research of the current framework, with adaptability for future improvement. The functionality of the present programming requires a proper approach towards programming advancement. This Hostel administration website is intended for individuals who need to handle different activities in the hostel. As from last few years' numbers of colleges/universities are increasing and so is students number which require an automated system which can reduce human effort and make administration easy and in more technological way.

5.3 RECOMMENDATIONS

I would like to recommend that other people and associations to make further research and look into this problem very well, because as the day goes by, new problems arises, therefore we need new solutions to tackle these problems that are likely to come up, I would also recommend that people should try using other methods to solve this problem so that there can be alternative ways in which the problem can be solved without much stress.

5.4 CONTRIBUTION TO KNOWLEDGE

This project has contributed to my knowledge, by having to pick up new programming languages to solve the problem of the HOSTEL MANAGEMENT SYSTEM, it has also taught about keeping the database and having to keep and organize files for record keeping in managing the system.

5.5 LIMITATION OF THE STUDY

The limitation to this study includes

- i. A computer is needed for this system to function
- ii. Without Electricity this system cannot be used
- iii. Without the internet, this system cannot be used
- iv. A computer illiterate would not be able to use this system

REFERENCES

- Ala' Abbadi, Al-Hammadi Y., Abdul-Rahman A. D. and Mior N. Hostel, Business Model, and IIUM Hostel Room Management System: A Conceptual Transformation Proposal Program. (*Faculty of Communication and Information Technology International Islamic University Malaysia*), Gombak, 53100, Kuala Lumpur, Malaysia (2017)
- Automated resource management system for hostel allocation (<https://iproject.com.ng/computer-science/automated-resource-management-system-for-hostel-allocation>) (Accessed May 22, 2019)
- College Hostel Management Software by Initio (2010)
- Design and implementation of hostel allocation system(<https://www.projecttopics.org/hostel-allocation-system.html>) (Accessed May 22, 2019)
- Felke-Morris. Basics of Web Design: HTML5 & CSS3, 2nd Edition, Addison-Wesley, 2013.
- Hampton, David R. 1977. Contemporary Management, McGraw-Hill Series in Management.
- Hostel manager by Laisikwa W.O. (2015)
- <https://searchoracle.techtarget.com/definition/MySQL> (Accessed June 23, 2019)
- <https://www.php.net/manual/en/intro-what-is.php> (Accessed June 23, 2019)
- <https://www.vocabulary.com/dictionary/hostel> (accessed 24, july 2019)
- https://www.w3schools.com/whatis/whatis_css.asp (Accessed June 23, 2019)
- https://www.w3schools.com/whatis/whatis_html.asp (Accessed June 23, 2019)
- <https://www.quora.com/What-is-the-use-of-XAMPP> (Accessed June 23, 2019)
- Indocon Hostel Management Software by Indocon Micro Engineers Limited.
- K.A. Muhammed Shaheer, A. Muhammed Shiras, R. Vinod Raj, G.V Prashobh (2009).
Project Report on Hostel Management System.

- Kola, A., Shoewu, O., Segun, O. O., Olusegun, O., Damilola, D. B. (2014). Development of an automated hostel facility management system. *Journal of Science and Engineering* , 5(1), 1-10.
- Kozaei,F., Ayub, N., Hassan, A.S., &Kozaei, Z. (2010). The factors predicting students satisfaction with university hostels, case study, university Sains Malaysia. *Asian Culture and History*,2(2), 148-158.
- Loventis Booking System by Loventis Systems (2005).
- Mishra,A. N. (1994). *Students and Hostel Life*.New Delhi: Mittal publications.
- R. Radhakrishhan, P.A Rinsha, R Roopersree (2014). *Online Hostel Management System: Mini Project*.
- Radhakrishnan, R. (2014). *Online hostel management system (doctoral dissertation, cochin university of science and technology)*
- S.R Ahmad, A.K Ghalib and S.A. Mahmood (2013). *Pakistan Journal of Science (Vol 65 No. 1) : GIS Based Hostel Management System for Punjab University*. W. N. H. B. W. JA'AFAR (2012). *Hostel Management system*.
- Thakkar, D. (2012).*Diary of a fresher-2*.DhirubhaiAmbani Institute of information and Communication, Gundhinagar.<http://www.coolage.in/2012/06/02/diary-of-a-fresher-2/>.
- Whitten, Bentley, and Dittman. 2004. *System Analysis and Design Methods (5th edition)*. McGrawHill: New York, NY

APPENDIX

Home Page

```
<?php
```

```
session_start();
```

```
include('includes/config.php');
```

```
if(isset($_POST['login']))
```

```
{
```

```
    $matricno=$_POST['matricno'];
```

```
    $password=$_POST['password'];
```

```
    $stmt=$mysqli->prepare("SELECT matricno, password FROM registration WHERE  
matricno=? AND password=? ");
```

```
        $stmt->bind_param('ss',$matricno,$password);
```

```
        $stmt->execute();
```

```
        $stmt -> bind_result($matricno,$password);
```

```
        $rs=$stmt->fetch();
```

```
        $stmt->close();
```

```
        $_SESSION['matricno']=$matricno;
```

```
        $uip=$_SERVER['REMOTE_ADDR'];
```

```
        $ldate=date('d/m/Y h:i:s', time());
```

```
        if($rs)
```

```
        {
```

```
            $matricno = $_SESSION['matricno'];
```

```
            $ip=$_SERVER['REMOTE_ADDR'];
```

```
            $geopluginURL='http://www.geoplugin.net/php.gp?ip='.$ip;
```

```
            $addrDetailsArr = unserialize(file_get_contents($geopluginURL));
```

```
            $city = $addrDetailsArr['geoplugin_city'];
```

```
            $country = $addrDetailsArr['geoplugin_countryName'];
```



```

        $log="insert into userlog(matricno, userIp, city, country)
values('$matricno','$ip','$city','$country)";

        $mysqli->query($log);
if($log)
{
header("location:dashboard.php");

        }
}

        else
        {

                echo "<script>alert('Invalid Username/Email or
password');</script>";

        }

}

?>

```

```

<!doctype html>
<html lang="en" class="no-js">
<head>
        <meta charset="UTF-8">
        <meta http-equiv="X-UA-Compatible" content="IE=edge">
        <meta name="viewport" content="width=device-width, initial-scale=1, minimum-
scale=1, maximum-scale=1">
        <meta name="description" content="">
        <meta name="author" content="">
        <meta name="theme-color" content="#3e454c">
        <title>Student Hostel Registration</title>
        <link rel="stylesheet" href="css/font-awesome.min.css">
        <link rel="stylesheet" href="css/bootstrap.min.css">

```

```

<link rel="stylesheet" href="css/dataTables.bootstrap.min.css">>
<link rel="stylesheet" href="css/bootstrap-social.css">
<link rel="stylesheet" href="css/bootstrap-select.css">
<link rel="stylesheet" href="css/fileinput.min.css">
<link rel="stylesheet" href="css/awesome-bootstrap-checkbox.css">
<link rel="stylesheet" href="css/style.css">
<script type="text/javascript" src="js/jquery-1.11.3-jquery.min.js"></script>
<script type="text/javascript" src="js/validation.min.js"></script>
<script type="text/javascript" src="http://code.jquery.com/jquery.min.js"></script>
<script type="text/javascript">
function valid()
{
if(document.registration.password.value!= document.registration.cpassword.value)
{
alert("Password and Re-Type Password Field do not match !!");
document.registration.cpassword.focus();
return false;
}
return true;
}
</script>
</head>
<body>
<?php include('includes/header.php');?>
<div class="ts-main-content">
<?php include('includes/sidebar.php');?>
<div class="content-wrapper">
<div class="container-fluid">

```

```

<div class="row">
  <div class="col-md-12">
    <h2 class="page-title">Welcome to Mountain Top
University Hostel Management System. </h2>
    <div class="row">
      <div class="col-md-6 col-md-offset-3">
        <!-- <bk-light> -->
        <div class=" panel panel-primary well row pt-2x
pb-3x">
          <div class="panel-heading">
            <h2 class="text-center">User
Login</h2>
          </div>
          <div class="panel-body col-md-8 col-md-
offset-2">
            <form action="" class="mt"
method="post">
              <label for="" class="text-
uppercase text-sm">Matric No</label>
              <input type="text"
placeholder="Matric No" name="matricno" id="matricno" class="form-control mb">
              <label for="" class="text-
uppercase text-sm">Password</label>
              <input type="password"
placeholder="Password" name="password" id="matricno" class="form-control mb">
              <input type="submit"
name="login" class="btn btn-primary btn-block text-lg" value="Login" >

```



```
</html>
```

REGISTRATION

```
<?php
session_start();
include('includes/config.php');
if(isset($_POST['submit']))
{
$matricno=$_POST['matricno'];
$fname=$_POST['fname'];
$mname=$_POST['mname'];
$lname=$_POST['lname'];
$gender=$_POST['gender'];
$college=$_POST['college'];
$dept=$_POST['dept'];
$programme=$_POST['programme'];
$level = $_POST['level'];
$password=$_POST['password'];
$query="insert into
registration(matricno,firstName,middleName,lastName,gender,college,dept,programme,year,password) values (?,?,?,?,?,?,?,?,?,?)";
$stmt = $mysqli->prepare($query);
$rc=$stmt-
>bind_param('sssssssss',$matricno,$fname,$mname,$lname,$gender,$college,$dept,$programme,$level,$password);
$stmt->execute();
echo"<script>alert('Student Succssfully Registered');</script>";
}
?>
```

```

<!doctype html>
<html lang="en" class="no-js">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1, minimum-
scale=1, maximum-scale=1">
    <meta name="description" content="">
    <meta name="author" content="">
    <meta name="theme-color" content="#3e454c">
    <title>User Registration</title>
    <link rel="stylesheet" href="css/font-awesome.min.css">
    <link rel="stylesheet" href="css/bootstrap.min.css">
    <link rel="stylesheet" href="css/dataTables.bootstrap.min.css">>
    <link rel="stylesheet" href="css/bootstrap-social.css">
    <link rel="stylesheet" href="css/bootstrap-select.css">
    <link rel="stylesheet" href="css/fileinput.min.css">
    <link rel="stylesheet" href="css/awesome-bootstrap-checkbox.css">
    <link rel="stylesheet" href="css/style.css">
<script type="text/javascript" src="js/jquery-1.11.3-jquery.min.js"></script>
<script type="text/javascript" src="js/validation.min.js"></script>
<script type="text/javascript" src="http://code.jquery.com/jquery.min.js"></script>
<script type="text/javascript">
function valid()
{
if(document.registration.password.value!= document.registration.cpassword.value)
{
alert("Password and Re-Type Password Field do not match !!");

```

```

document.registration.cpassword.focus();
return false;
}
return true;
}
</script>
</head>
<body>
    <?php include('includes/header.php');?>
    <div class="ts-main-content">
        <?php include('includes/sidebar.php');?>
        <div class="content-wrapper">
            <div class="container-fluid">

                <div class="row">
                    <div class="col-md-12">

                        <h2 class="page-title">Student Registration </h2>

                        <div class="row">
                            <div class="col-md-12">
                                <div class="panel panel-primary">
                                    <div class="panel-
heading"><b>Please fill all fields</b></div>

                                    <div class="panel-body">

                                        <form method="post" action="" name="registration" class="form-
horizontal" onSubmit="return valid();">

```

```
<div class="form-group">
<label class="col-sm-2 control-label"> Matric No </label>
<div class="col-sm-8">
<input type="text" name="matricno" id="matricno" class="form-control" onBlur =
"checkAvailability()" required="required" >
<span id="user-availability-status" style="font-size:12px;"></span>
</div>
</div>
```

```
<div class="form-group">
<label class="col-sm-2 control-label">First Name </label>
<div class="col-sm-8">
<input type="text" name="fname" id="fname" class="form-control" required="required" >
</div>
</div>
```

```
<div class="form-group">
<label class="col-sm-2 control-label">Middle Name </label>
<div class="col-sm-8">
<input type="text" name="mname" id="mname" class="form-control">
</div>
</div>
```

```
<div class="form-group">
<label class="col-sm-2 control-label">Last Name </label>
<div class="col-sm-8">
<input type="text" name="lname" id="lname" class="form-control" required="required">
```



```
</div>
```

```
</div>
```

```
<div class="form-group">
```

```
<label class="col-sm-2 control-label">Gender </label>
```

```
<div class="col-sm-8">
```

```
<select name="gender" class="form-control" required="required">
```

```
<option value="">Select Gender</option>
```

```
<option value="male">Male</option>
```

```
<option value="female">Female</option>
```

```
<option value="others">Others</option>
```

```
</select>
```

```
</div>
```

```
</div>
```

```
<div class="form-group">
```

```
<label class="col-sm-2 control-label">College </label>
```

```
<div class="col-sm-8">
```

```
<input type="text" name="college" id="college" class="form-control" required="required">
```

```
</div>
```

```
</div>
```

```
<div class="form-group">
```

```
<label class="col-sm-2 control-label">Department </label>
```

```
<div class="col-sm-8">
```

```
<input type="text" name="dept" id="dept" class="form-control" onBlur=""  
required="required">
```

```
</div>
```

```
</div>
```

```
<div class="form-group">
<label class="col-sm-2 control-label">Programme </label>
<div class="col-sm-8">
<input type="text" name="programme" id="programme" class="form-control" onBlur=""
required="required">
</div>
</div>
```

```
<div class="form-group">
<label class="col-sm-2 control-label">Level </label>
<div class="col-sm-8">
<input type="text" name="level" id="level" class="form-control" onBlur=""
required="required">
</div>
</div>
```

```
<div class="form-group">
<label class="col-sm-2 control-label">Password </label>
<div class="col-sm-8">
<input type="password" name="password" id="password" class="form-control"
required="required">
</div>
</div>
```

```
<div class="form-group">
<label class="col-sm-2 control-label">Confirm Password </label>
<div class="col-sm-8">
```

```
<input type="password" name="cpassword" id="cpassword" class="form-control"
required="required">
```

```
</div>
```

```
</div>
```

```
<div class="col-sm-6 col-sm-offset-4">
```

```
<input type="submit" name="submit" Value="Register" class="btn btn-primary btn-md text-
lg">
```

```
<button class="btn btn-default btn-md text-lg" type="reset">Cancel</button>
```

```
</div>
```

```
</form>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<script src="js/jquery.min.js"></script>
```

```
<script src="js/bootstrap-select.min.js"></script>
```

```

<script src="js/bootstrap.min.js"></script>
<script src="js/jquery.dataTables.min.js"></script>
<script src="js/dataTables.bootstrap.min.js"></script>
<script src="js/Chart.min.js"></script>
<script src="js/fileinput.js"></script>
<script src="js/chartData.js"></script>
<script src="js/main.js"></script>
</body>
<script>
function checkAvailability() {

$("#loaderIcon").show();
jQuery.ajax({
url: "check_availability.php",
data:'matricno='+$("#matricno").val(),
type: "POST",
success:function(data){
$("#user-availability-status").html(data);
$("#loaderIcon").hide();
},
error:function ()
{
event.preventDefault();
alert('error');
}
});
}
</script>

```

</html>

LOGIN

<?php

session_start();

include('includes/config.php');

if(isset(\$_POST['login']))

{

\$matricno=\$_POST['matricno'];

\$password=\$_POST['password'];

\$stmt=\$mysqli->prepare("SELECT matricno, password FROM userregistration WHERE matricno=? AND password=? ");

\$stmt->bind_param('ss',\$matricno,\$password);

\$stmt->execute();

\$stmt -> bind_result(\$matricno,\$password);

\$rs=\$stmt->fetch();

\$stmt->close();

\$_SESSION['user']=\$matricno;

\$uip=\$_SERVER['REMOTE_ADDR'];

\$ldate=date('d/m/Y h:i:s', time());

if(\$rs)

{

\$uid=\$_SESSION['id'];

\$uemail=\$_SESSION['login'];

\$ip=\$_SERVER['REMOTE_ADDR'];

\$geopluginURL='http://www.geoplugin.net/php.gp?ip='.\$ip;

\$addrDetailsArr = unserialize(file_get_contents(\$geopluginURL));

```

$city = $addrDetailsArr['geoplugin_city'];
$country = $addrDetailsArr['geoplugin_countryName'];
$log="insert into userLog(matricno, userIp, city, country)
values('$matricno','$ip','$city','$country)";
$mysqli->query($log);
if($log)
{
header("location:dashboard.php");
}
else
{
echo "<script>alert('Invalid Username/Email or
password');</script>";
}
?>

```

```

<!doctype html>
<html lang="en" class="no-js">
<head>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1, minimum-
scale=1, maximum-scale=1">
<meta name="description" content="">
<meta name="author" content="">
<meta name="theme-color" content="#3e454c">
<title>Student Hostel Registration</title>

```

```

<link rel="stylesheet" href="css/font-awesome.min.css">
<link rel="stylesheet" href="css/bootstrap.min.css">
<link rel="stylesheet" href="css/dataTables.bootstrap.min.css">>
<link rel="stylesheet" href="css/bootstrap-social.css">
<link rel="stylesheet" href="css/bootstrap-select.css">
<link rel="stylesheet" href="css/fileinput.min.css">
<link rel="stylesheet" href="css/awesome-bootstrap-checkbox.css">
<link rel="stylesheet" href="css/style.css">
<script type="text/javascript" src="js/jquery-1.11.3-jquery.min.js"></script>
<script type="text/javascript" src="js/validation.min.js"></script>
<script type="text/javascript" src="http://code.jquery.com/jquery.min.js"></script>
<script type="text/javascript">
function valid()
{
if(document.registration.password.value!= document.registration.cpassword.value)
{
alert("Password and Re-Type Password Field do not match !!");
document.registration.cpassword.focus();
return false;
}
return true;
}
</script>
</head>
<body>
<?php include('includes/header.php');?>
<div class="ts-main-content">
<?php include('includes/sidebar.php');?>

```

```

<div class="content-wrapper">
  <div class="container-fluid">

    <div class="row">
      <div class="col-md-12">

        <h2 class="page-title">User Login </h2>

        <div class="row">
          <div class="col-md-6 col-md-offset-3">
            <div class="well row pt-2x pb-3x bk-light">
              <div class="col-md-8 col-md-offset-2">

                <form action="" class="mt"
method="post">

                  <label for="" class="text-
uppercase text-sm">Matric No</label>

                  <input type="text"
placeholder="Matric No" name="matno" class="form-control mb">

                  <label for="" class="text-
uppercase text-sm">Password</label>

                  <input type="password"
placeholder="Password" name="password" class="form-control mb">

                  <input type="submit"
name="login" class="btn btn-primary btn-block" value="login" >

                </form>

              </div>

            </div>

          </div>

        <div class="text-center text-light">

```