HMS

HOSTEL MANAGEMENT SYSTEM

A project report submitted by

KEERTHY K R (GTAUSCS021)

Under the guidance of

Mrs.Shamly K

Assistant Professor, Department of Computer Science

Sri. C Achutha Menon Government College, Thrissur

For the award of the Degree of

BACHELOR OF SCIENCE IN COMPUTER SCIENCE

(Calicut University)

2020-2023

Submitted to

Department of Computer Science

Sri. C. AchuthaMenon Government College, Thrissur

Kuttanellur, Kerala

CERTIFICATE

This is to certify that the project titled **HMS Hostel Management System** submitted by, **KEERTHY K R(GTAUSCS021)** is a bonafide Record of the project work done at the Department of Computer Science, Sri C Achutha Menon Government College, Thrissur in partial fulfilment of the requirement of **Bachelor of Computer Science of the University of Calicut.**

Submitted for the Viva Voce Examination held on 11/04/2023

Internal Examiner

External Examiner

Shamly K
Internal Project Guide

Sreevidya S
Head of the Department

DECLARATION

We hereby declare that this submission is our own work and that, to the best of our knowledge and belief, it contains no material previously published or written by another person nor material which has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgment has been made in the text.

ARATHY E R (GTAUSCS002)

AJITH P S (GTAUSCS024)

KEERTHY K R (GTAUSCS021)

VIDYASAGAR A K (GTAUSCS016)

Place: Kuttanellur
Date :

ACKNOWLEDGEMENT

Every work accomplished is a pleasure, a sense of satisfaction. However, a number of people always motivate, criticize and appreciate a work with their objective ideas and options. Hence, we would like to use this opportunity to thank all who have directly or indirectly helped us to accomplish this project.

First and foremost, we give all glory, honor and praise to the Lord who gave us wisdom, provided us with everything we required and enabled us to complete this project successfully. We express our heartfelt thanks to our parents for their encouragement. We would also like to express our gratitude to Sri. A K Suresh, Principal of Sri. C AchuthaMenon Government College Kuttanellur, for the help given by her, and we would also like to express our sincere thanks to Dr. Sri. Sreevidya S, Head of the department of Computer Science, for her encouragement rendered to us during the course of our project and for supporting us with resources.

We owe a particular debt of gratitude to our guide concerned, Sri. Shamly K, assistant professor of Department of Computer Science of Sri. C AchuthaMenon Government College Kuttanellur, for her valuable time for giving technical assistance and support extended to us. We also take this opportunity to thank our senior, Neeraj for his support and guidance given by him in this project. Finally, we wish to express our sincere thanks to all the teachers, non-teaching staff as well as our friends for their suggestions and support.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Hostel Management Systems (HMS) are essential for colleges to manage their student accommodation facilities. In the existing system, students apply for rooms by filling out paper applications, and room allotment is based on eligibility criteria. The process of maintaining student records, attendance, leave applications, and meal plans is also manual. This can lead to inefficiencies, errors, and delays in the process, making it difficult for students to access timely and accurate information.

To address these challenges, we propose a digital Hostel Management System named HMS that streamlines the process of managing student accommodation and related data. The proposed system is designed to be easy-to-use, efficient, and scalable, and can be customized to meet the specific needs of different educational institutions.

The proposed Hostel Management System is significant for several reasons. First, it leverages digital technologies to automate and streamline the hostel management process, reducing the time and effort required to manage student accommodation facilities. This can save valuable resources and improve operational efficiency, which can ultimately benefit educational institutions, students, and other stakeholders.

Second, the proposed system could improve the accuracy and completeness of student records, attendance, leave applications, and meal plans. This can reduce errors and inconsistencies, which can ultimately benefit students and educational institutions alike.

Finally, the proposed system could enhance the overall student experience and satisfaction. By providing timely and accurate information to students, hostel wardens, and administrators, the system can help create a more transparent and responsive hostel management process, which can ultimately lead to higher levels of student satisfaction and retention.

In summary, the proposed Hostel Management System is an innovative and impact solution that addresses the challenges of managing student accommodation facilities in educational institutions. The system has the potential to improve operational efficiency, reduce errors and inconsistencies, and enhance the overall student experience and satisfaction.

1.1 Hostel Management System

HMS is an online hostel management system that provides a digital interface for hostel admission and management processes for educational institutions. The system is designed to automate the hostel management process, eliminating most of the paper works and reducing human effort. HMS uses PHP for the back end and HTML, CSS, JavaScript and Ajax for the front-end, with the Bootstrap framework, making it a robust and scalable hostel management system.

1.2 Features of Existing Systems

The existing hostel management system has a number of features that enable students to apply for hostel rooms, get allotted based on eligibility, and manage their accommodation requirements. These features include an application process, room allotment based on eligibility, and manual tracking of student attendance, leave applications, and fee payments by the hostel admin and warden. However,
the system has several limitations that can impact on the efficient management of student accommodation facilities.

1.3 Limitations of Existing Systems

The limitations of existing attendance management systems include manual processes, lack of transparency, limited accessibility, data security risks, and inadequate record-keeping. The reliance on manual processes can lead to inefficiencies, delays, and errors in managing hostel applications, room allotment, fee collection, attendance tracking, and leave applications. The lack of transparency can cause difficulties for students in understanding the status of their applications, room allotment, attendance, or leave applications in real-time. Additionally, the system may not be accessible to students who are unable to physically visit the office to obtain an application form or make fee payments. Data security risks may also arise, as student records and personal information are stored in physical files and may be susceptible to loss or theft. Lastly, the system may not maintain accurate and up-to-date records of student attendance, fee payments, or leave applications.

1.4 Area and Category of the Project Work

The project work on hostel management system is a software engineering project that focuses on creating a web-based application to automate and improve the processes involved in

managing student accommodation facilities. It involves developing a user-friendly system that facilitates activities such as student applications, room allotment, fee payments, attendance tracking, leave applications, and meal management. The project incorporates the latest web technologies such as

HTML, CSS, JavaScript, PHP, Ajax and MySQL to create a robust and scalable application. The project work falls under the category of computer science and software engineering, and it has potential applications in educational institutions worldwide.

Chapter 2

Problem Definition & Methodology

2.1 Introduction

Hostels are an essential part of the education system, providing accommodation for students who are studying away from home. Managing a hostel can be a daunting task, with numerous students to manage, room allocation, and payment management. HMS Hostel management systems provide a solution to these problems by automating the processes of managing a hostel, providing a centralized system for managing student records, room allocation, and payment management. The system also provides a means for administrators to monitor the hotel's operations and make informed decisions.

2.2 Problem Definition

Hostel management can be a complex and time-consuming task. College hostel with multiple rooms and students. Hostel admin and warden often face challenges in managing reservations, room assignments, check-in and check-out procedures, payment details, attendance taking and inventory management. These tasks are typically carried out manually or with the use of outdated systems, which can lead to errors, inefficiencies, and delays.In addition, managing a hostel requires constant monitoring of room occupancy and expenses, which can be overwhelming without the help of a robust hostel management system. Without accurate data and analytic, hostel admin may find it challenging to make informed decisions about their operations, leading to missed opportunities and decreased profitability.

Therefore, the problem that an HMS hostel management system aims to solve is to provide a comprehensive, automated, and user-friendly solution for managing all aspects of hostile operations. The system should streamline the process of managing.

reservations, room assignments, check-in and check-out procedures, payment details, attendance taking and inventory management. It should also provide real-time data

and analytic to help admin and warden to make informed decisions about their operations, leading to improved efficiency, profitability, and student satisfaction.

2.3 Objectives

The primary objective of our project is to develop a modern Hostel Management System that improves the efficiency and effectiveness of managing student accommodation facilities. By leveraging modern technologies, the system aims to reduce the time and effort required to manage student accommodation facilities. Additionally, it seeks to enhance the accuracy and completeness of student records, attendance, leave applications, and meal plans. The system
also provides timely and accurate information to students, hostel wardens, and administrators. This, in turn, increases transparency and accountability in the hostel management process, improving the overall student experience and satisfaction. Ultimately, the HMS Hostel Management System aims to transform the way student accommodation facilities are managed, offering a more modern and effective solution to the existing manual systems used in educational institutions.

2.4 Motivation

The motivation behind developing a hostel management system is to improve the efficiency and effectiveness of managing hostel operations. The system aims to streamline and automate the various tasks involved in managing a hostel, which are often time-consuming and labor-intensive. By automating these tasks, hostel admin can focus on providing a better experience for students and managing the day-to-day operations of the hostel more effectively.
Moreover, with the increasing use of technology in the hospitality industry, students expect hostels to provide a seamless and easy-to-use booking and management experience. A hostel management system can provide students with a convenient way to make reservations, check-in and check-out, and pay for services, improving their overall experience and satisfaction.
Overall, the motivation behind a hostel management system is to improve the efficiency, profitability, and student satisfaction of hostel operations while providing a seamless and easy-to-use experience for both staff and students.

2.5 Methodology

The methodology adopted in the development of HMS is an object-oriented approach. The system is designed to be scalable, modular, and maintainable, with a focus on usability and user experience. The system uses PHP for the back end and HTML, CSS, Ajax and JavaScript for the front end, with the Bootstrap framework. The development process includes requirements gathering, design,implementation, testing, and deployment. The system is designed with security in mind, using encryption and authentication techniques to protect sensitive data. The development process begins with requirements gathering, where the team identifies the requirements and needs of the target users of the system. The team then moves on to the design phase, where they create a detailed design of the system, including the user interface, database schema, and algorithms. The team then moves on to the implementation phase, where they write the code and build the system. The team conducts rigorous testing to ensure that the system is error-free, and then moves on to deployment, where they install the system on the server and make it available to users.

2.6 Scope

The HMS Hostel Management System project has a broad scope and is intended to provide a comprehensive solution to the management of student accommodation facilities. The system is designed to be used by students, hostel wardens, and administrators to manage various aspects of hostel management, including room allocation, fee collection, attendance tracking, leave applications, meal planning, and more.

The system is expected to be scalable and flexible, allowing it to be used in hostels of different sizes and with varying requirements. It is also designed to be customization, with the ability to add or modify features based on specific needs.

The system is also intended to be user-friendly and accessible, with a simple and intuitive user interface that is easy to navigate and use. This will enable students, hostel wardens, and administrators to use the system without any significant training or technical expertise.

In summary, the Hostel Management System project has a broad scope and is intended to provide a comprehensive solution to the management of student accommodation facilities. It is designed to be scalable, flexible, customization, user-friendly, and accessible, making it suitable for use in hostels of different sizes and with varying requirements.

Chapter 3

Analysis

3.1 Requirement Analysis

Requirement analysis is a critical phase in the development of a hostel management system, where the requirements and expectations of the stakeholders are analysed and documented. The following are some key steps involved in requirement analysis for a hostel management system: Identifying Stakeholders: Identify the stakeholders who will be using or affected by the hostel management system, such as hostel administrators, staff, students.

Gathering Requirements: Gather requirements from the stakeholders through various means such as interviews and surveys. The requirements should cover various aspects, such as room management, student management, inventory management, attendance management, and reporting.

Prioritizing Requirements: Prioritize the requirements based on their importance and relevance to the stakeholders. The prioritization should be based on factors such as impact, feasibility, and urgency.

Documenting Requirements: Document the requirements in a detailed and structured manner using techniques such as DFD, ER diagram. The documentation should clearly define the functionalities and features of the hostel management system.

Validating Requirements: Validate the requirements with the stakeholders to ensure that they are accurate, complete, and meet their expectations.

Managing Changes: Manage changes to the requirements throughout the development cycle by implementing a change management process. The change management process should ensure that the changes are properly documented, reviewed, approved, and implemented.

By following these steps, the requirement analysis phase can help ensure that the hostel management system meets the needs of the stakeholders and is aligned with the objectives of the organization.

3.2 Existing System

The existing hostel management system at our college involves a manual process for managing student accommodation facilities. Students interested in applying for a room must obtain an application form from the office and fill it out. The room allotment process is based on eligibility criteria, which are determined by reviewing the applications received.

Student records, fee payments, attendance, and leave applications are all maintained manually by the hostel warden. There is no centralized system to manage these records, which can lead to errors and inconsistencies. Moreover, there is no system for students to view their attendance records, meal plans, or leave applications.The existing system also lacks transparency and accountability, as there is no mechanism for students to track the status of their room applications or for administrators to monitor the hostel warden's performance. Overall, the existing system is time-consuming, error-prone, and lacks the features required to effectively manage student accommodation facilities.

3.3 Proposed System

Our proposed HMS hostel management system aims to leverage modern technologies to streamline the process of managing student accommodation facilities. The system will provide a user-friendly interface that allows administrators, hostel wardens, and students to manage their respective roles more efficiently.The proposed system will include an admin module that only authorized personnel can access. This module will allow admins to add and edit rooms, create accounts for students, and book rooms based on eligibility criteria. Admins will also be able to view and edit student information, manage payment details, and generate attendance reports. The system will also include a warden module, which will allow wardens to view and manage student information, record attendance, generate attendance reports, and verify leave applications. Additionally, wardens will be able to upload meal menus, allowing students to view their meal plans online.Finally, the student module will allow students to create accounts, apply for hostel rooms, view their room details and fees, and track their attendance and meal plans. Students will also be able to submit leave applications and track their status.

The proposed system will significantly improve the efficiency and effectiveness of managing student accommodation facilities. It will provide real-time access to information, streamline the process of managing student records, and increase transparency and accountability in the hostel management process. Overall, the proposed system will enhance the student experience and satisfaction.

3.4 Requirement Specification

Requirement gathering is one of the most important phases of the software development life cycle. It is the phase that tells us what the system is supposed to do and drives the other phases in the life cycle.

The Software requirements specification establishes the basis for an agreement between customers and contractors or suppliers on how the software product should function (in a market-driven project, these roles may be played by the marketing and development divisions). The Software requirements specification is a rigorous assessment of requirements before the more specific system design stages, and its goal is to reduce later redesign. It should also provide a realistic basis for estimating product costs, risks, and schedules. Used appropriately, software requirements specifications can help prevent software project failure. The software requirements specification document lists sufficient and necessary requirements for the project development. To derive the requirements, the developer needs to have a clear and thorough understanding of the products under development. This is achieved through detailed and continuous communications with the project team and customer throughout the software development process. The requirement specifications may include both functional requirements and non-functional requirements.

3.4.1 Functional Requirements

Implementation is the stage of a project when the theoretical design is turned into a working system. If the implementation stage is not properly planned and controlled, it can cause chaos. Thus, it can be considered to be the most crucial stage in achieving a successful new system and in giving the users confidence that the new system will work and be effective. Normally, this stage involves setting up a co-ordination committee, which will act as a sounding board of ideas, complaints and problems. The first task is implementation planning, i.e.; decision of the methods and timescale to be adopted. Apart from planning, the two major tasks of repairing for implementation are education and training of administrators and testing of the system.After the implementation phase is completed and the user staff adjusted to the changes created by the candidate system, evaluation and maintenance is continued to bring the new system standards. The activities of the implementation phase can be summarized as;

- Implementation planning
- Education and training
- System training System implementation is the final stage that puts the utility into action.

Implementation is the state in the project where the theoretical design is turned into a working system.

3.4.2 Non-functional Requirements

Non-functional requirements are an essential aspect of any software development project. Unlike functional requirements, non-functional requirements focus on the system's performance, usability, reliability, and security. These requirements may not be directly relatedto the software's features, but they play a crucial role in determining how well the software performs in real-world scenarios.

Performance

System is the most important quality in non-functional requirements and affects almost all the other preceding ones. Furthermore, reliability, availability, and maintainability (RAM) features fall exclusively under these requirements. System performance defines how fast a system can respond to a particular user's action under a certain workload.

Reliability

Reliability is the probability and percentage of the software performing without failure for a specific number of uses or amount of time.

Availability

This feature defines the amount of time the system is running, the time it takes to repair a fault, and the time between lapses.

Security

Security measures ensure your software's safety against espionage or sabotage. These features are necessary even for stand-alone systems; you don't want anyone to have access to your sensitive data.

Maintainability

This feature indicates the average time and ease and rapidity with which a system can be restored after a failure.

Portability

Portability in high-level computer programming is the usability of the same software in different environments. The primary requirement for portability is the generalized abstraction between the application logic and system interfaces. When software with the same functionality is produced for several computing platforms, portability is the key issue for development cost reduction.

3.4.3 Hardware Requirements

The selection of hardware configuration is a very important task related to software development. The processor should be powerful to handle entire operations.

Processor : Intel® Core™ i3 or above

RAM : 4 GB RAM minimum, 8 GB recommended

Hard Disk : 150 GB

Keyboard : Standard

Mouse : Normal

Monitor : Plug and play monitor

3.4.4 Software Requirements

The major element in building a system is the selection of compatible software. The software requirement of the system on which the project was developed is as follows:

Operating System : Microsoft Windows 7 or above

Front End : HTML, CSS, JavaScript,BOOTSTRAP, AJAX

Back End : PHP, XAMPP, MySQL

IDE : Visual Studio Code

Web Browser :Google Chrome

3.4.5 Other Requirements

OPERATING SYSTEM

An operating system (OS) is system software that manages computer hardware and software resources and provides common services for computer programs. The operating system used in this project is Windows OS.

HTML (FRONT END)

The Hyper Text Markup Language or HTML is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript.Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.HTML elements are the building blocks of HTML pages.

With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by tags, written using angle brackets.

CSS (FRONT END)

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language such as HTML.CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.CSS is designed to enable the separation of presentation and content, including layout, colours, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate.CSS file, which reduces complexity and repetition in the structural content as well as enables the CSS files to be cached to improve the page load speed between the pages that share the file and its formatting.

BOOTSTRAP(FRONT END)

Bootstrap is an HTML, CSS and JS library that focuses on simplifying the development of informative web pages (as opposed to web applications). The primary purpose of adding it to

a web project is to apply Bootstrap's choices of colour, size, font and layout to that project. As such, the primary factor is whether the developers in charge find those choices to their liking. Once added to a project, Bootstrap provides basic style definitions for all HTML elements. The result is a uniform appearance for prose, tables and form elements across web browsers. In addition, developers can take advantage of CSS classes defined in Bootstrap to further customize the appearance of their contents. For example, Bootstrap has provisions for light- and dark-coloured tables, page headings, more prominent pull quotes, and text with a highlight.

AJAX (FRONT END)

is a set of web development techniques that uses various web technologies on the client-side to create asynchronous web applications. With Ajax, web applications can send and retrieve data from a server asynchronously (in the background) without interfering with the display and behaviour of the existing page.

PHP (BACK END)

PHP is a general-purpose scripting language geared toward web development. PHP was originally an abbreviation for *Personal Home Page*, but it now stands for the recursive initialism PHP: Hypertext Pre processor.PHP code is usually processed on a web server by a PHP interpreter implemented as a module, a daemon or as a Common Gateway Interface (CGI) executable. On a web server, the result of the interpreted and executed PHP code – which may be any type of data, such as generated HTML or binary image data – would form the whole or part of an HTTP response. Various web template systems, web content management systems, and web frameworks exist which can be employed to orchestrate or facilitate the generation of that response. Additionally, PHP can be used for many programming tasks outside the web context, such as standalone graphical applications] and robotic drone control. PHP code can also be directly executed from the command line.

XAMPP (BACK END)

XAMPP is a free and open-source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in PHP and Perl programming languages. Since most actual web server deployments use the same components as XAMPP, it makes transitioning from a local test server to a live server possible.

MY SQL (BACK END)

MySQL is an open-source relational database management system (RDBMS). Its name is a combination of "My", the name of co-founder Michael Widenius's daughter My, and "SQL", the acronym for Structured Query Language. A relational database organizes data into one or more data tables, in which data may be related to each other; these relations help structure the data. SQL is a language programmers use to create, modify and extract data from the relational database, as well as control user access to the database. In addition to relational databases and SQL, an RDBMS like MySQL works with an operating system to implement a relational database in a computer's storage system, manages users, allows for network access and facilitates testing database integrity and creation of backups.

MySQL is free and open-source software under the terms of the GNU Public Licence, and is also available under a variety of proprietary licences. MySQL was owned and sponsored by the Swedish company MySQL AB, which was bought by Sun Microsystems (now Oracle Corporation).In 2010, when Oracle acquired Sun, Widenius forked the open-source MySQL project to create MariaDB.

Visual Studio Code

Visual Studio Code is a source-code editor that can be used with a variety of programming languages, including Java, JavaScript, Go, Node.js, Python and C++. It is based on the Electron framework, which is used to develop Node.js Web applications that run on the Blink layout engine. Visual Studio Code employs the same editor component (code named "Monaco") used in Azure DevOps (formerly called Visual Studio Online and Visual Studio Team Services). Instead of a project system, it allows users to open one or more directories, which can then be saved in their workspace for future reuse. This allows it to operate as a language- agnostic code editor for any language. It supports a number of programming languages and a set of features that differ per language. Unwanted files and folders can be excluded from the project tree via the settings. Many Visual Studio Code features are not exposed through menus or the user interface, but can be accessed via the command palette.Visual Studio Code can be extended via extensions, available through a central repository. This includes additions to the editor and language support. A notable feature is the ability to create extensions that add support for new languages, themes, and debuggers, perform static code analysis, and add code litres using the Language Server Protocol. Visual Studio Code includes multiple extensions for FTP, allowing the software to be used as a free

alternative for web development. Code can be synced between the editor and the server, without downloading any extra software.

3.5 Feasibility Study

Feasibility study is a process that identifies, describes and evaluates proposed systems and selects the best system. During the study, the problem definition is solved and all aspects of the problem to be included in the system are determined. The size of the project and cost of benefits is also estimated with greater accuracy. A good feasibility study will show the strength and defects before the project is planned or budgeted.It evaluates the project's potential success, because it rationally uncovers strengths and weaknesses of a proposed project.

3.5.1 Technical feasibility

It determines whether the technology needed for the proposed system is available and how it can be integrated with the government. Technical evaluation must assess whether the user has technical expertise to understand and use the new system. This assessment is based on an outline design of system requirements, to determine whether the proposal is technically and legally feasible. It is the evaluation of the hardware and software and how it meets the needs of the proposed system.

3.5.2 Economical feasibility

The purpose of the economic feasibility assessment is to determine the positive economic benefits to the organization that the proposed system will provide. It includes quantification and identification. It identifies the financial benefits and costs associated with the development of the system. Economic feasibility is often known as the cost benefit analysis. To carry out an economic feasibility study, it is necessary to estimate actual money value against activities needed for implementing the system. While implementing our system, we can ensure that the cost of prospective new ventures will be ultimately be profitable for the people. So, we can say it is financially feasible. The technology
used to be developed with the current equipment and has the technical capacity to hold data by the old system required.

- This technology supports the modern trends of technology.

- Easily accessible, more secure technologies.

3.5.3 Operational feasibility

Operational feasibility is the measure of how well a proposed system solves the problems, and takes advantage of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirement analysis phase of system development. It focuses on the degree to which the proposed development project fits in with the existing environment and the objectives in regard to development schedule. Operational feasibility focuses on human, organizational and political. The proposed system can easily be implemented, as it is based on PHP coding, HTML. The database was created with a Tomcat server, which is more secure and easy to handle. The resources that are required to implement/install these are available. So the project is operationally feasible.

Chapter 4

Design

4.1 Introduction

System design is the process of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements. System development is the process of creating or altering systems, along with the processes, practices, models, and methodologies used to develop them. The different interfaces of those components and the data that goes through that system. It is meant to satisfy specific needs and requirements of a business or organization through the engineering of a coherent and well-running system.Systems design implies a systematic approach to the design of a system. It may take a bottom-up or top-down approach, but either way, the process is systematic wherein it takes into account all related variables of the system that needs to be created, from the architecture, to the required hardware and software, right down to the data and how it travels and transforms throughout its travel through the system. Systems design then overlaps with systems analysis, systems engineering and systems architecture

4.2 Modularity Criteria

The concept of modularity is used primarily to reduce complexity by breaking a system into varying degrees of independence and independence across, and hiding the complexity of each part behind an abstraction and interface. Effective modular design can be achieved if the partitioned modules are separately solvable, modifiable as well as compatible.

4.3 Architecture Diagrams/DFD

A data-flow diagram (DFD) is a way of representing the flow of a data of a process or a system. The DFD also provides information about the outputs and inputs of each entity and the process

itself. A data-flow diagram has no control flow, there are no decision rules and no loops.

Specific operations based on the data can be represented. There are several notations for displaying data flow

diagrams. The notation presented above was described in 1979 by Tom DE Marco as part of Structured Analysis. For each data flow, at least one of the endpoints must exist in a process. The refined representation of a process can be done in another data-flow diagram, which subdivides this process into sub-processes. The data-flow diagram is part of the structured-analysis modelling tools. When using UML, the activity diagram typically takes over the role of the data flow diagram. A special form of data-flow plan is a site-oriented data-flow plan. Data-flow diagrams can be regarded as inverted Petri nets, because places in such networks correspond to the semantics of data memories. Analogously, the semantics of transitions from Petri nets and data flows and functions from data-flow diagrams should be considered equivalent.

Terminologies used in DFD diagrams are:

Process

The process is part of a system that transforms inputs into outputs. The symbol of a process is a circle, an oval, a rectangle or a rectangle with rounded corners. The process is named on one word, a short sentence, or a phrase that is clear to express its essence.

Data Store

Most information systems capture data for later use. The data is kept in a data store. It is represented by the open-ended box. At the left end is a small box used to number the data store and inside the main part of the rectangle is a meaningful label for the data store.

Data Flow

Data flow shows the transfer of information from one part of the system to another. The symbol of the flow is the arrow. The flow should have a name that determines what information (or what material) is being moved. Exceptions are flows where it is clear what information is transferred through the entities that are linked to these flows. Material shifts are modeled in systems that are not merely informative. Flow should only transmit one type of information (material). The arrow shows the flow direction. Flows link processes, warehouses and terminators.

Warehouse

The warehouse (data store, data store, file, and database) is used to store data for later use. The symbol of the store is two horizontal lines; the other way of view is shown in the DFD

Notation. The name of the warehouse is a plural noun (e.g., orders) - it derived from the input and output streams of the warehouse. The warehouse does not have to be just a data file, for example, a folder with documents, a filing cabinet, and optical discs. Therefore, viewing the warehouse in DFD is independent of implementation. The flow from the warehouse usually represents the reading of the data stored in the warehouse, and the flow to the warehouse usually expresses data entry or updating (sometimes also delete data). The warehouse is represented by two parallel lines, between which the memory name is located.

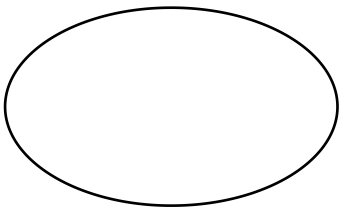
Source/Sink

A source/sink is the origin and/or destination of the data. Source/sinks are sometimes referred to as external entities because they are outside the system and they define boundaries of the system. Data must originate outside a system from one or more sources, and the system must produce information to one or more sinks.

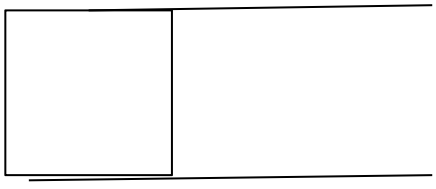
Terminator

The Terminator is an external entity that communicates with the system and stands outside the system. It can be, for example, various organizations, groups of people (e.g., customers), authorities or a department (e.g., a human-resources department) of the same organization, which does not belong to the model system. The terminator may be another system with which the modelled system communicates.

Symbols used in DFD



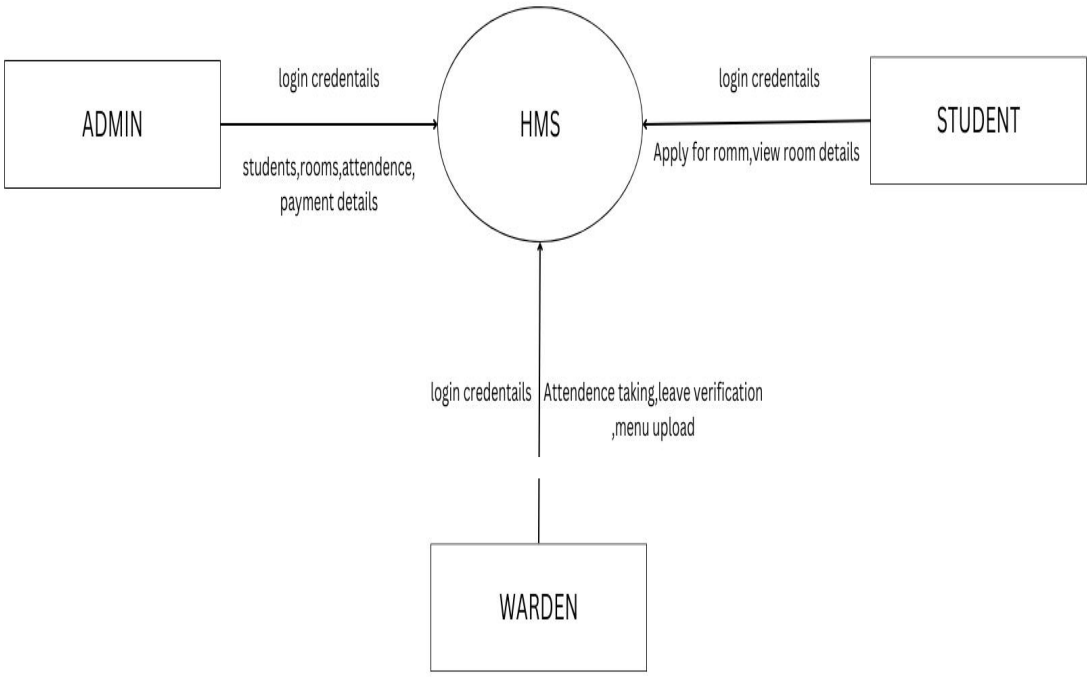
Process



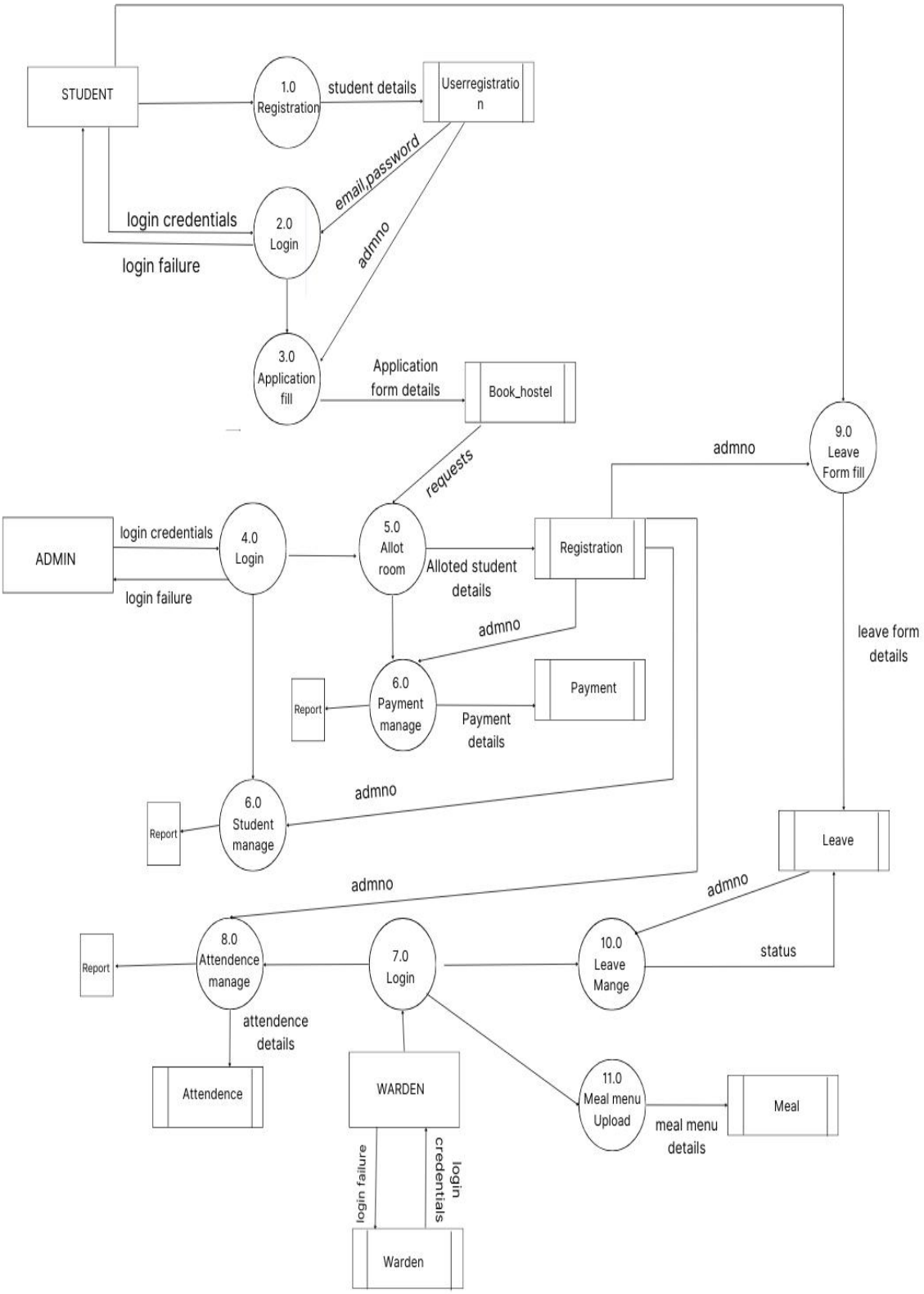
Data store

Data Flow

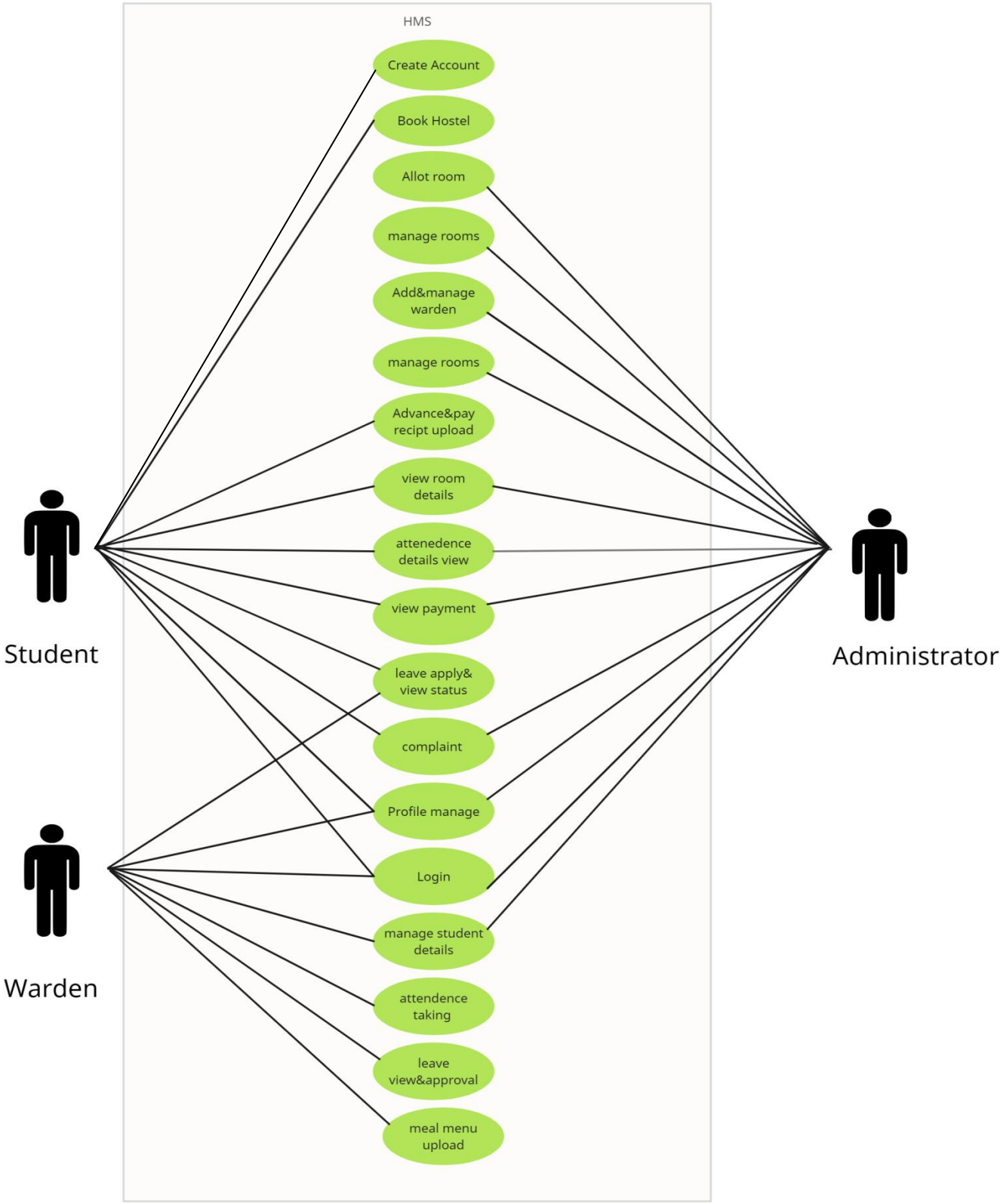
4.3.1 DFD Level 0



4.3.2 DFD Level 1

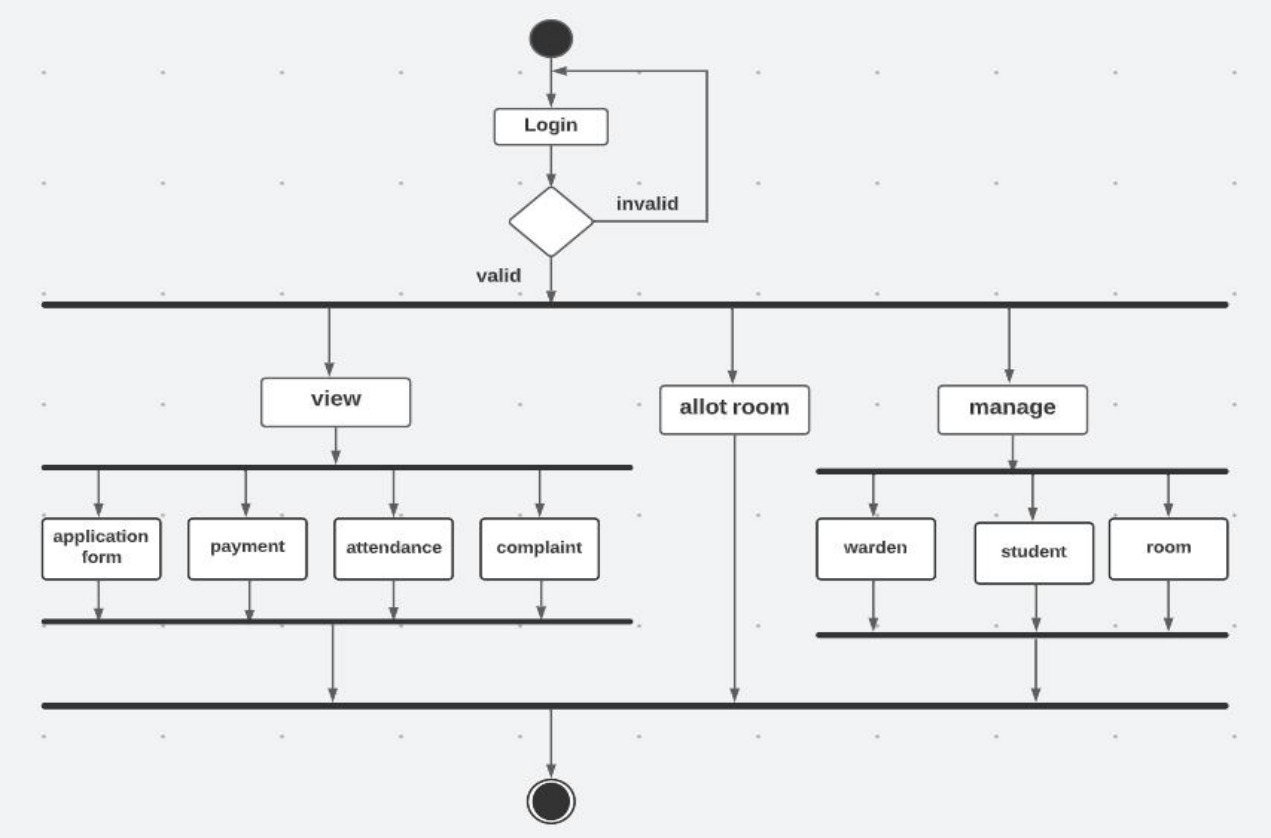


4.4 Use Case Diagrams

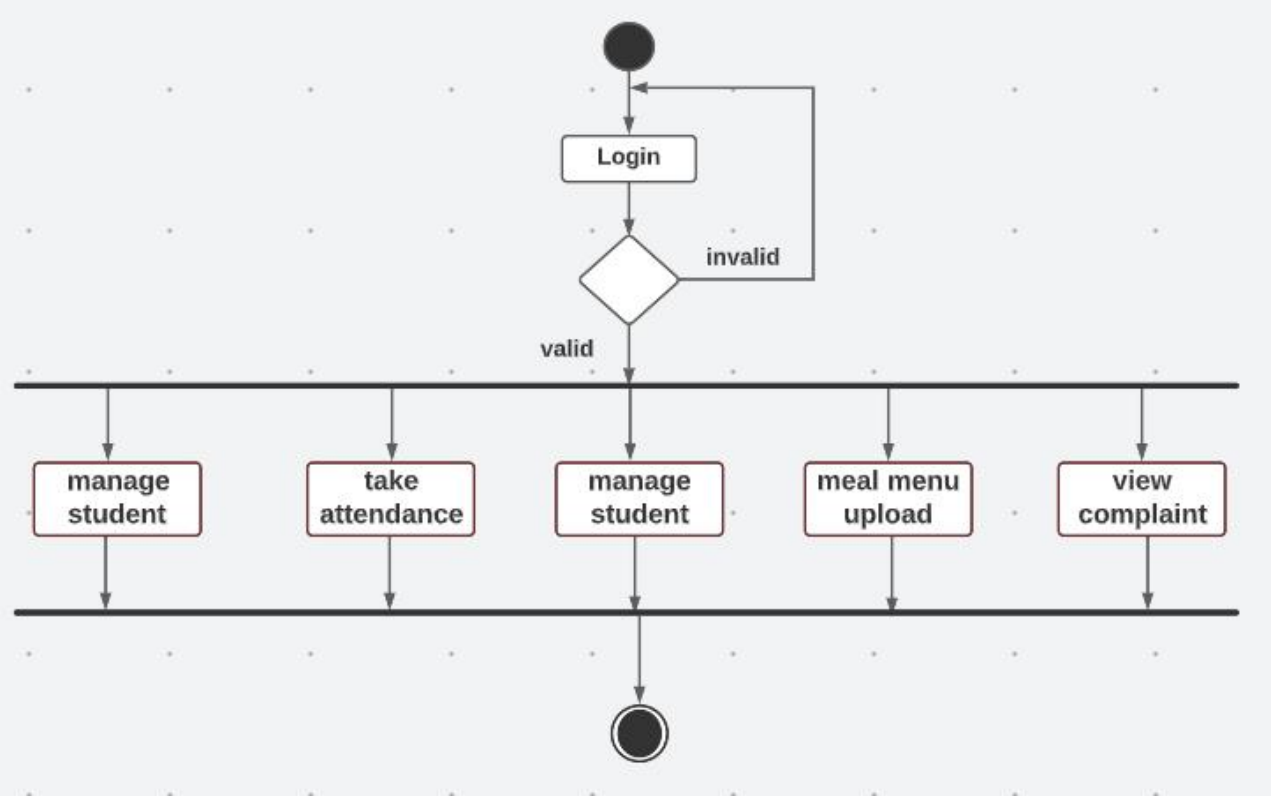


4.5 Activity Diagrams

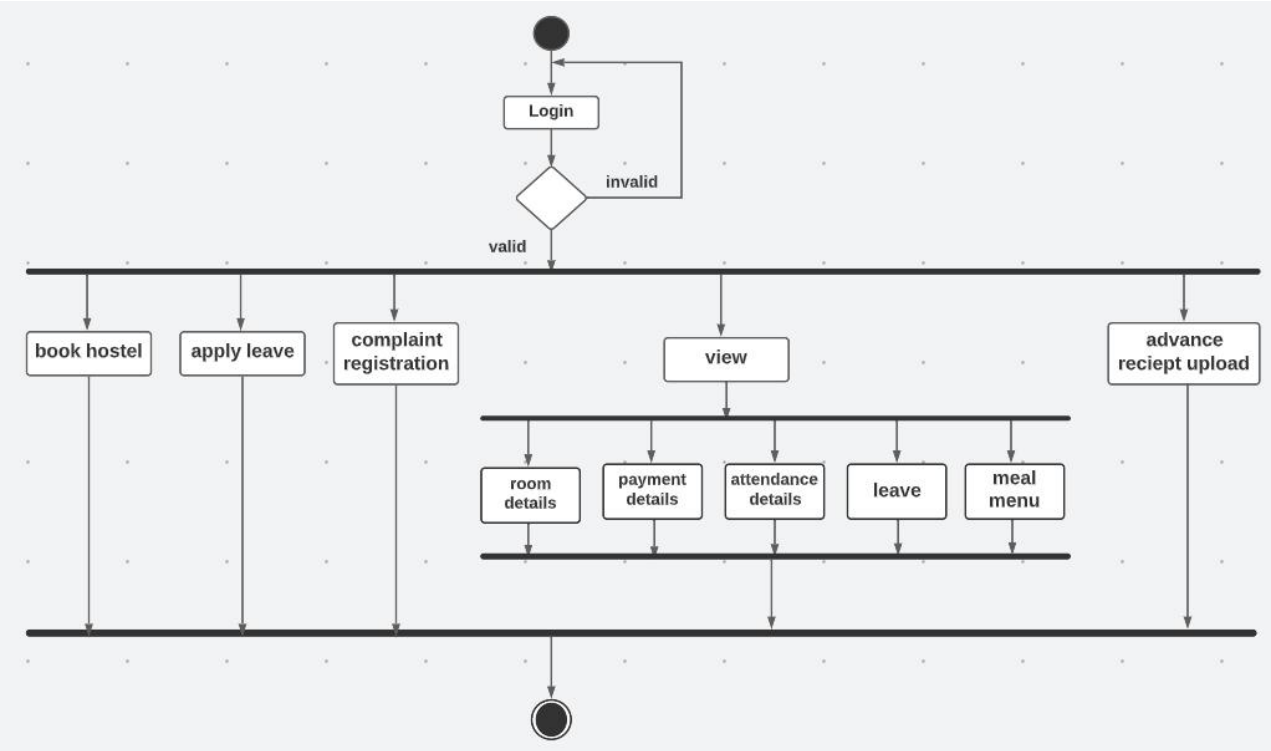
ADMIN



WARDEN

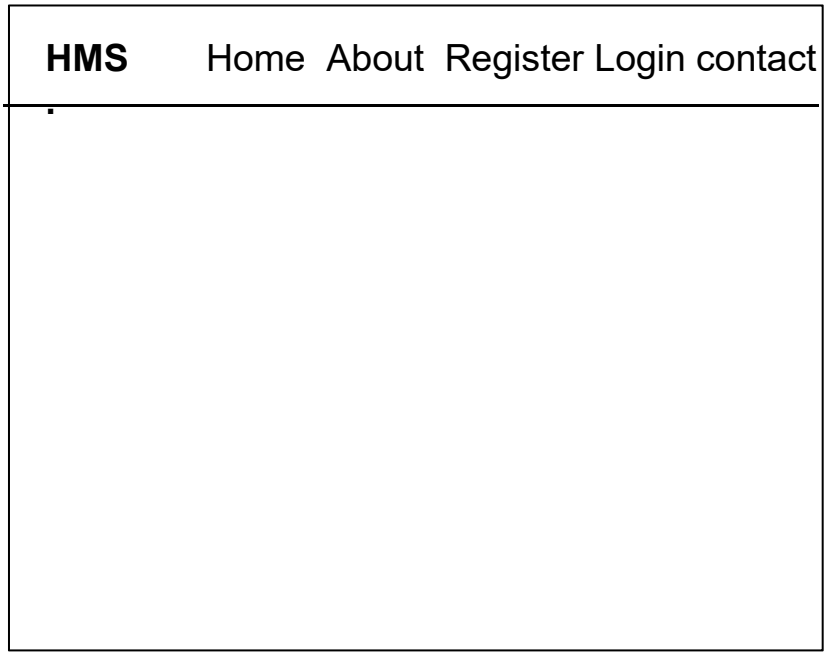


STUDENT

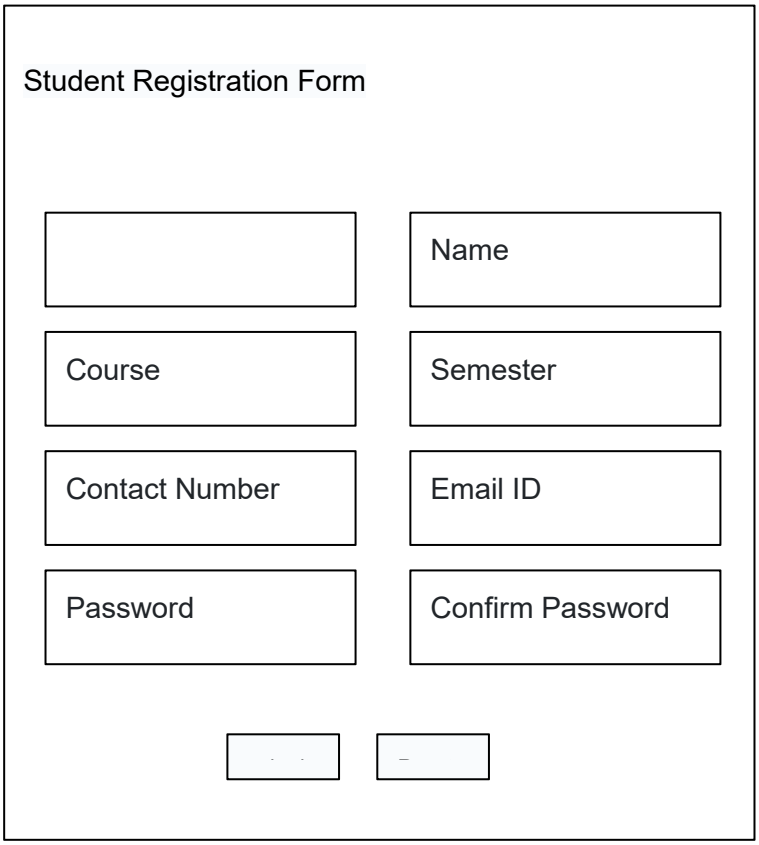


4.6 User Interface Layout

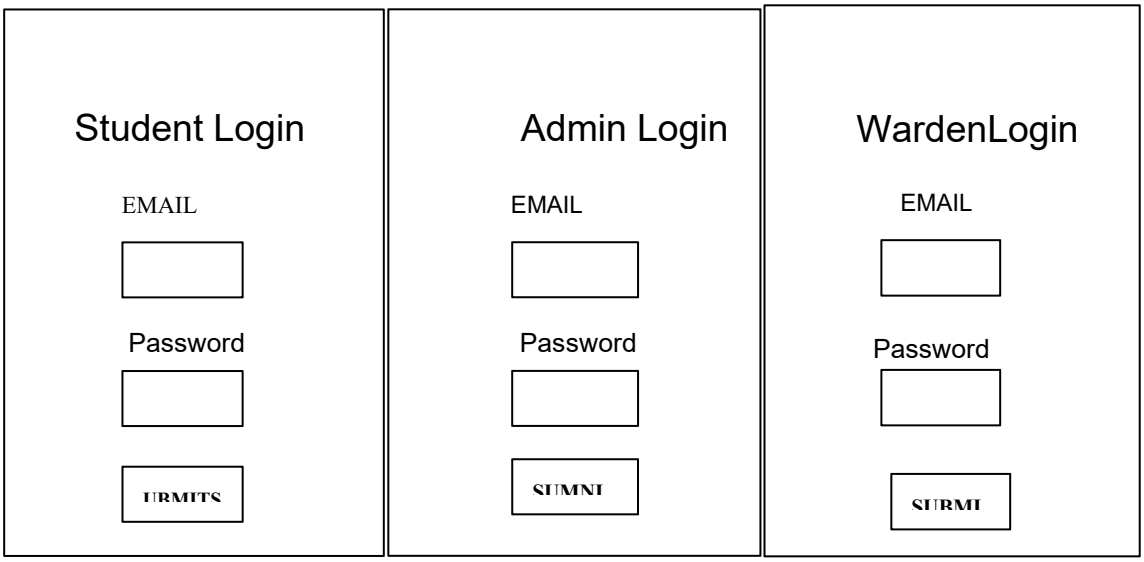
HOME PAGE



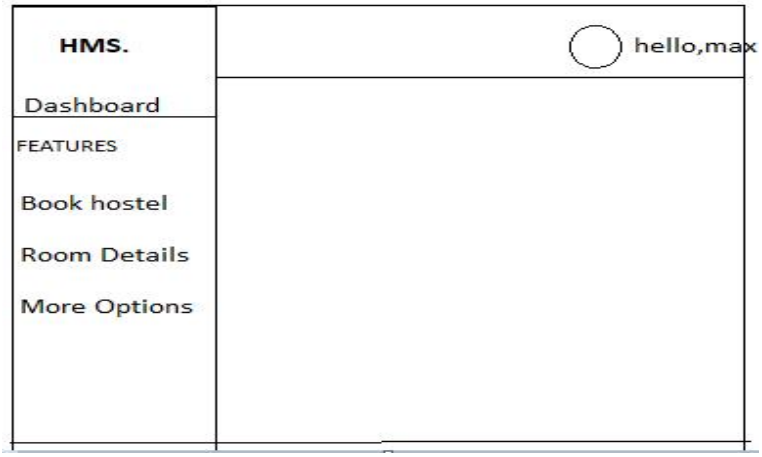
Student registration page



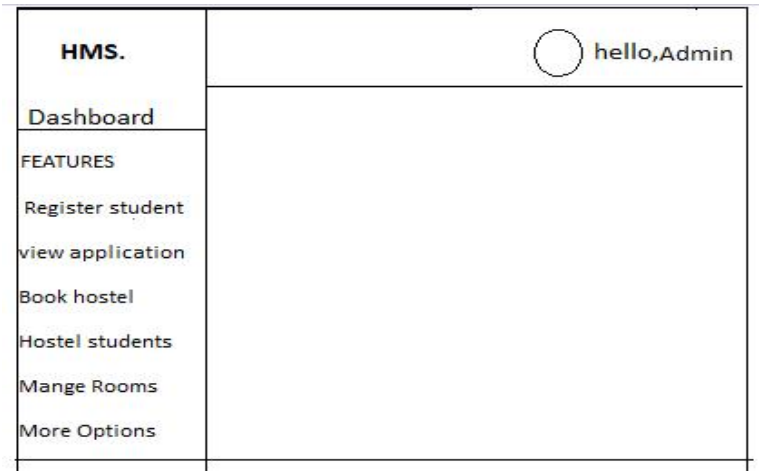
Login Pages


Student Login
EMAIL
Password
Admin Login
EMAIL
Password
WardenLogin
EMAIL
Password

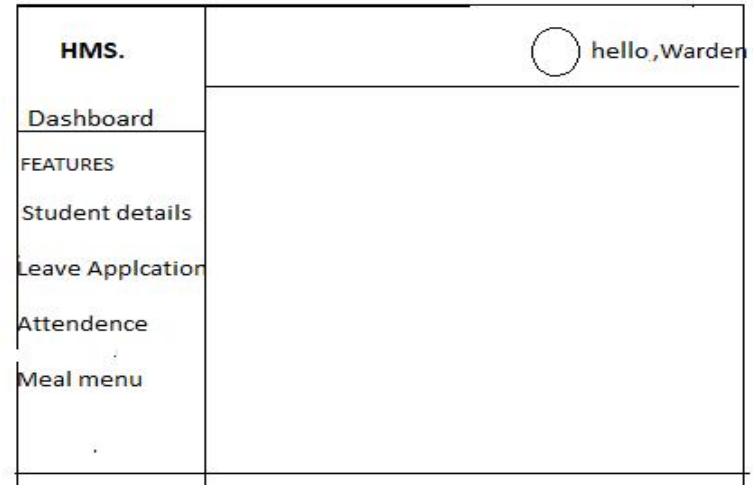
Student Dashboard



Admin Dashboard



Warden Dashboard



4.7 Structure of Reports Being Created

In the Hostel Management System, users have the ability to generate attendance reportsand allow seamless printing of the reports. The fields in the report are

- Admno
- Name
- Roomno
- date

4.8 Database Design

A database is a data structure that stores organized information. Most databases contain multiple tables, which may each include several different fields. For example, a company database may include tables for products, employees, and financial records. Each of these tables would have different fields that are relevant to the information stored in the table. Nearly all e-commerce sites use databases to store product inventory and customer information. These sites use a database management system (or DBMS), such as Microsoft Access, File Maker Pro, or MySQL as the "back end" to the website. By storing website data in a database, the data can be easily searched, sorted, and updated. This flexibility is important for e-commerce sites and other types of dynamic websites. Early databases were relatively "flat," which means they were limited to simple rows and columns, like a spreadsheet. (See also "flat file database"). However, today's relational databases allow users to access, update, and search information based on the relationship of data stored in different tables. Relational databases can also run queries that involve multiple databases. While early databases could only store text or numeric data, modern databases also let users store other data types such as sound clips, pictures, and videos.

The database design has several specific objectives:

- Control redundancy
- Ease of Learning and use
- Data independence
- More information at low cost
- Accuracy and integrity

- Recovery from failure
- Privacy and security
- Performance

Normalization

Normalization is the process of organizing data in a database. This includes creating tables and establishing relationships between those tables according to rules designed both to protect the data and to make the database more flexible by eliminating redundancy and inconsistent dependency. Redundant data wastes disk space and creates maintenance problems. If data that exists in more than one place must be changed, the data must be changed in exactly the same way in all locations. A customer address change is much easier to implement if that data is stored only in the Customers table and nowhere else in the database. There are a few rules for database normalization. Each rule is called a "normal form." If the first rule is observed, the database is said to be in "first normal form." If the first three rules are observed, the database is considered to be in "third normal form." Although other levels of normalization are possible, the third normal form is considered the highest level necessary for most applications. As with many formal rules and specifications, real world scenarios do not always allow for perfect compliance. In general, normalization requires additional tables and some customers find this cumbersome. If you decide to violate one of the first three rules of normalization, make sure that your application anticipates any problems that could occur, such as redundant data and inconsistent dependencies.

The following descriptions include examples:

First Normal Form

- Eliminate repeating groups in individual tables.
- Create a separate table for each set of related data.
- Identify each set of related data with a primary

Do not use multiple fields in a single table to store similar data. For example, to track an inventory item that may come from two possible sources, an inventory record may contain fields for Vendor Code 1 and Vendor Code 2. What happens when you add a third vendor? Adding a field is not the answer; it requires program and table modifications and does not

smoothly accommodate a dynamic number of vendors. Instead, place all vendor information in a separate table called Vendors, then link inventory to vendors with an item number key, or vendors to inventory with a vendor code key.

Second Normal Form

- Create separate tables for sets of values that apply to multiple records.
- Relate these tables with a foreign key.

Records should not depend on anything other than a table's primary key (a compound key, if necessary). For example, consider a customer's address in an accounting system. The address is needed by the Customers table, but also by the Orders, Shipping, Invoices, Accounts Receivable, and Collections tables. Instead of storing the customer's address as a separate entry in each of these tables, store it in one place, either in the Customers table or in a separate addresses table.

Third Normal Form

- Eliminate fields that do not

Values in a record that are not part of that record's key do not belong on the table. In general, any time the contents of a group of fields may apply to more than a single record in the table, consider placing those fields in a separate table. For example, in an Employee Recruitment table, a candidate's university name and address may be

included. But you need a complete list of universities for group mailings. If university information is stored in the candidates table, there is no way to list universities with no current candidates. Create a separate Universities table and link it to the candidates' table with a university code key.

EXCEPTION:

Adhering to the third normal form, while theoretically desirable, is not always practical. If you have a customer table, and you want to eliminate all possible interfiled dependencies, you must

create separate tables for cities, ZIP codes, sales representatives, customer classes, and any other factor that may be duplicated in multiple records. In theory, normalization is worth pursuing. However, many small tables may degrade performance or exceed open file and

memory capacities. It may be more feasible to apply the third normal form only to data that changes frequently. If some dependent fields remain, design your application to require the user to verify all related fields when anyone is changed.

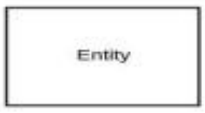
Other Normalization Forms

The fourth normal form, also called Boyce Codd Normal Form (BCNF), and fifth normal form do exist, but are rarely considered in practical design. Disregarding these rules may result in less than perfect database design, but should not affect functionality.

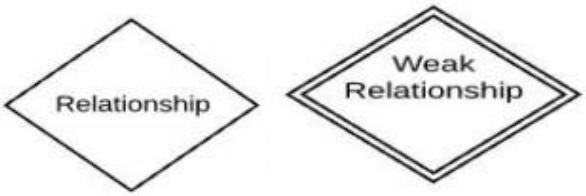
4.8.1 List of Entities and Attributes

Entities are objects or concepts that represent important data. Entities are typically nouning such as product, customer, location, or promotion. There are three types of entities commonly used in entity relationship diagrams.

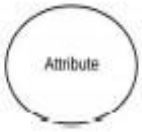
Entities are objects or concepts that represent important data. Entities are typically nouning such as product, customer, location, or promotion. There are three types of entities commonly used in entity relationship diagrams.

	Strong entity	These shapes are independent from other entities, and are often called parent entities, since they will often have weak entities that depend on them. They will also have a primary key, distinguishing each occurrence of the entity.
	Weak entity	Weak entities depend on some other entity type. They don't have primary keys, and have no meaning in the diagram without their parent entity.
	Associative entity	Associative entities relate the instances of several entity types. They also contain attributes specific to the relationship between those entity instances.

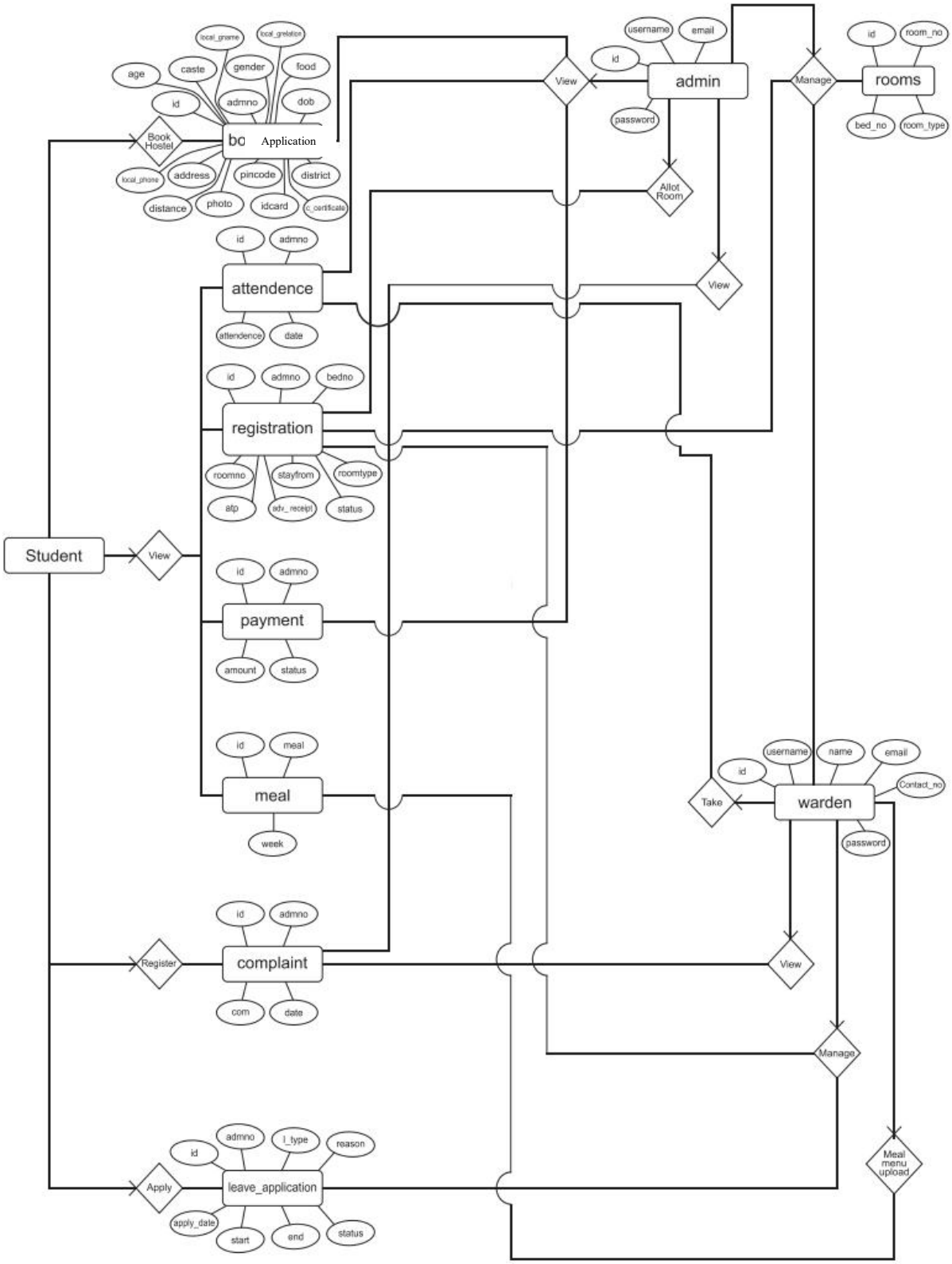
Relationships are meaningful associations between or among entities. They are usually verbs, example-assign, associate or track. A relationship provides useful information that couldn't be discerned with just entity types. Weak relationships, or identifying relationships, are connections that exist between a weak entity type and its owner.



ERD attributes are characteristics of the entity that help users to better understand the database. Attributes are included to include details of the various entities that are highlighted in a conceptual ER diagram

Attribute Symbol	Name	Description
	Attribute	Attributes are characteristics of an entity, a many-to-many relationship, or a one-to-one relationship.
	Multivalued attribute	Multivalued attributes are those that are can take on more than one value.

4.8.2 E R Diagram



4.8.3 Structure of Tables

Database design is the process of producing a detailed data model of a database. This data model contains all the needed logical and physical design choices and physical storage parameters needed to generate a design in a data definition language, which can then be used to create a database. A fully attributed data model contains detailed attributes for each entity.
The term database design can be used to describe many different parts of the design of an overall database system. Principally, and most correctly, it can be thought of as the logical design of the base data structures used to store the data. In the relational model, these are the tables and views. In an object database, the entities and relationships map directly to object classes and named relationships. However, the term database design could also be used to apply to the overall process of designing, not just the base data structures, but also the forms and queries used as part of the overall database application within the database system (DBMS).

- Determine the relationships between the different data elements.
- Determine the data to be stored in the database.

Superimpose a logical structure upon the data on the basis of these relationships

Table no:1

Admin Details

field name	type	constraints
id	int(11)	primary key
username	varchar(200)	not null
email	varchar(200)	not null
password	varchar(200)	not null
reg_date	timestamp	not null
updation_date	date	not null

Warden Details

field name	type	constraints
id	int(11)	primary key
username	varchar(200)	not null
name	varchar(200)	not null
email	varchar(200)	not null
password	varchar(200)	not null
reg_date	timestamp	not null
updation_date	date	not null

User Registration Details

field name	type	constraints
id	int(11)	primary key
admNo	varchar(200)	not null
name	varchar(200)	not null
course	varchar(200)	not null
semester	varchar(20)	not null
contactNo	varchar(200)	not null
email	varchar(200)	not null

password	varchar(200)	not null
regDate	date	not null

Book_hostel Details

field name	type	constraints
id	int(11)	primary key
admno	varchar(200)	not null
sex	varchar(20)	not null
dob	date	not null
age	varchar(20)	not null
caste	varchar(20)	not null
food	varchar(20)	not null
distance	varchar(20)	not null
Local_gname	varchar(200)	not null
Local_grelation	varchar(200)	not null
Local_gcontact	varchar(200)	not null
address	varchar(200)	not null
district	varchar(200)	not null
pincode	varchar(200)	not null
photo	varchar(200)	not null
idcard	varchar(200)	not null

cc	varchar(200)	not null
status	varchar(200)	not null

Registration Details

id	int(50)	Primary key
admno	varchar(50)	not null
bedno	varchar(50)	not null
roomno	varchar(50)	not null
stayfrom	varchar(50)	not null
atp	timestamp	not null
Adv_recipt	varchar(500)	not null
status	varchar(50)	not null

Rooms Details

field name	**type**	**constraints**
id	int(11)	primary key
room_no	varchar(50)	not null
bed_no	varchar(50)	not null
room_type	varchar(50)	not null
posting_date	timestamp	not null

Payment Details

field name	type	constraints
id	int(11)	primary key
admno	varchar(50)	not null
amount	varchar(50)	not null
status	varchar(50)	not null
posted	date	not null
date	varchar(50)	not null
recipt	varchar(500)	not null

Attendance Details

field name	type	constraints
id	int(11)	primary key
admno	varchar(50)	not null
amount	varchar(50)	not null
remark	varchar(50)	not null
date	varchar(50)	not null

Leave Details

field name	type	constraints
id	int(11)	primary key

admno	varchar(50)	not null
l_type	varchar(50)	not null
reason	varchar(50)	not null
apply_date	date	not null
start	varchar(50)	not null
end	varchar(50)	not null
remark	varchar(50)	null
status	varchar(50)	null

Meal Details

field name	type	constraints
id	int(11)	primary key
meal	varchar(500)	not null
week	date	not null

Complaint Details

field name	type	constraints
id	int(11)	primary key
admno	varchar(50)	not null
com	varchar(50)	not null
date	date	not null

Chapter 5

Implementation

5.1 Introduction

Systems implementation is a set of procedures performed to complete the design contained in the approved systems design document and to test, install, and begin to use the new or revised Information System. It is the fifth major step in the development of an Information System. In software development, an implementation is the realization of a technical specification or algorithm as a program, software component, or other computer system through computer programming and deployment.

5.2 Tools/Scripts for Implementation

Project planning tools include charts and graphs designed to track progress, repetition-based approaches to testing and adjusting everyday processes, and other actions that allow organizations to manage and improve important projects.Tools for project implementation are defined as the series of systems and methodologies designed to ensure teams are able to accomplish both short and long-term projects.In our project, we used flow charts, Data Flow Diagrams, Entity Relationship diagrams as tools for project implementation.

5.3 Process Logic

Project logic provides the basis for planning and implementing monitoring and evaluation at project level. Project logic is defined as a conceptual framework of how a program or project is understood, or intended, to contribute to its specified outcomes. It focuses on outcomes rather than process.Process logic is a cause-and-effect explanation of a process. It expresses all the principal definitions and arguments that
appear to be true for the process and its events, causes and circumstances.

In our project, we used three different modules as the basis of process logic. They are the admin module,warden module and student module.

5.3.1 Module 1:Admin Module

The admin is the most important module in HMS.Only the admin is authorized to access the database. Admin can add and edit rooms.Admin can create accounts for students and book rooms.The Admin can view the applications received for the hostel room and from there allot the room based on the eligibility.You can view and edit information from hostlers.Payment details of holsters can be managed and uploaded.The attendance report can be viewed and verified.Admin can manage hostel warden.

5.3.2 Module 2 :Student module

Students can create accounts in HMS.They can apply for a hostel room for that account
If the hostel room is allotted, they will know their room details and fees from HMS.
Students can see their attendance status and meal menu. They can also apply for leave online and get information.

5.3.3 Module 3:Warden module

The warden can view the information from the holsters.Attendance of students can be marked, and a report generated. Warden can view and verify the leave applications of the hostel residents and also
meal menu can be uploaded

5.4 Coding

Admin module

index.php

```
<?php
   session_start();
   include('../includes/dbconn.php');
   if(isset($_POST['login'])){
   $username=$_POST['username'];
   $password=$_POST['password'];
   $password = md5($password);
```

```
    $stmt=$mysqli->prepare("SELECT username, email,password,id FROM admin WHERE (userName=?|| email=?) and password=? ");
        $stmt->bind_param('sss',$username,$username,$password);
        $stmt->execute();
        $stmt -> bind_result($username,$username,$password,$id);
        $rs=$stmt->fetch();
        $_SESSION['id']=$id;
        $uip=$_SERVER['REMOTE_ADDR'];
        $ldate=date('d/m/Y h:i:s', time());
        if($rs){
            header("location:dashboard.php");
                } else {
                    echo "<script>alert('Invalid Username/Email or password');</script>";
                }
    }
?>

<!DOCTYPE html>
<html dir="ltr">

<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta name="description" content="">
    <meta name="author" content="">
    <link rel="icon" type="image/png" sizes="16x16" href="../assets/images/favicon.png">
    <title>Hostel Management System</title>
    <link href="../dist/css/style.min.css" rel="stylesheet">
    <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>
    <div class="main-wrapper">
          <div class="preloader">
          <div class="lds-ripple">
             <div class="lds-pos"></div>
             <div class="lds-pos"></div>
          </div>
       </div>
        <div class="auth-wrapper d-flex no-block justify-content-center align-items-center position-relative"
          style="background:url(../assets/images/big/auth-bg.jpg) no-repeat center center;">
          <div class="auth-box row">
             <div class="col-lg-7 col-md-5 modal-bg-img" style="background-image: url(../assets/images/adimg.jpg);">
             </div>
             <div class="col-lg-5 col-md-7 bg-white">
                <div class="p-3">
                   <div class="text-center">
                      <img src="../assets/images/big/login.png" alt="wrapkit">
                   </div>
                   <h2 class="mt-3 text-center">Admin Login</h2>
                   <form class="mt-4" method="POST">
                      <div class="row">
                         <div class="col-lg-12">
                            <div class="form-group">
                               <label class="text-dark" for="uname">Email or Username</label>
                               <input class="form-control" name="username" id="uname" type="text"
```

```
                        placeholder="Email or Username" required>
                    </div>
                  </div>
                  <div class="col-lg-12">
                    <div class="form-group">
                      <label class="text-dark" for="pwd">Password</label>
                      <input class="form-control" name="password" id="pwd"
type="password"
                        placeholder="Enter your password" required>
                    </div>
                  </div>
                  <div class="col-lg-12 text-center">
                    <button type="submit" name="login" class="btn btn-block btn-
danger">LOGIN</button>
                  </div>
                  <div class="col-lg-12 text-center mt-5">
                    <a href="../index.html" class="text-danger">Go Back</a>
                  </div>
                </div>
              </form>
            </div>
          </div>
        </div>
      </div>
    <script src="../assets/libs/jquery/dist/jquery.min.js "></script>
  <script src="../assets/libs/popper.js/dist/umd/popper.min.js "></script>
  <script src="../assets/libs/bootstrap/dist/js/bootstrap.min.js "></script>
    <script>
     $(".preloader ").fadeOut();
  </script>
</body>

</html>
```

Dashboard.php

```
<?php
   session_start();
   include('../includes/dbconn.php');
   include('../includes/check-login.php');
   check_login();
?>

<!DOCTYPE html>
<html dir="ltr" lang="en">
<head>
   <meta charset="utf-8">
   <meta http-equiv="X-UA-Compatible" content="IE=edge">
   <!-- Tell the browser to be responsive to screen width -->
   <meta name="viewport" content="width=device-width, initial-scale=1">
   <meta name="description" content="">
   <meta name="author" content="">
   <!-- Favicon icon -->
   <link rel="icon" type="image/png" sizes="16x16" href="../assets/images/favicon.png">
   <title>Hostel Management System</title>
   <!-- Custom CSS -->
   <link href="../assets/extra-libs/c3/c3.min.css" rel="stylesheet">
   <link href="../assets/libs/chartist/dist/chartist.min.css" rel="stylesheet">
   <link                href="../assets/extra-libs/datatables.net-bs4/css/dataTables.bootstrap4.css"
rel="stylesheet">
   <!-- Custom CSS -->
   <link href="../dist/css/style.min.css" rel="stylesheet">
</head>
<body>
     <div class="preloader">
       <div class="lds-ripple">
          <div class="lds-pos"></div>
          <div class="lds-pos"></div>
```

```
      </div>
   </div>
    <div id="main-wrapper" data-theme="light" data-layout="vertical" data-navbarbg="skin6" data-sidebartype="full"
            <header class="topbar" data-navbarbg="skin6">
         <?php include 'includes/navigation.php'?>
      </header>
       <aside class="left-sidebar" data-sidebarbg="skin6">
         <div class="scroll-sidebar" data-sidebarbg="skin6">
            <?php include 'includes/sidebar.php'?>
         </div>
             </aside>
              <div class="page-breadcrumb">
            <div class="row">
               <div class="col-7 align-self-center">
                  <?php include 'includes/greetings-a.php'?>
                   <div class="d-flex align-items-center">
                   </div>
               </div>
            </div>
         </div>
                 <div class="card-group">
               <div class="card border-right">
                  <div class="card-body">
                     <div class="d-flex d-lg-flex d-md-block align-items-center">
                        <div>
                           <div class="d-inline-flex align-items-center">
                              <h2 class="text-dark mb-1 font-weight-medium"><?php include 'counters/student-count.php'?></h2>
                           </div>
                           <h6 class="text-muted font-weight-normal mb-0 w-100 text-truncate">Registered Students</h6>
                        </div>
                        <div class="ml-auto mt-md-3 mt-lg-0">
```

```
                        <span class="opacity-7 text-muted"><i data-feather="user-plus"></i></span>
                    </div>
                </div>
            </div>
            <div class="card border-right">
                <div class="card-body">
                    <div class="d-flex d-lg-flex d-md-block align-items-center">
                        <div>
                            <h2 class="text-dark mb-1 w-100 text-truncate font-weight-medium"><?php include 'counters/room-count.php'?></h2>
                            <h6 class="text-muted font-weight-normal mb-0 w-100 text-truncate">Total Rooms
                            </h6>
                        </div>
                        <div class="ml-auto mt-md-3 mt-lg-0">
                            <span class="opacity-7 text-muted"><i data-feather="grid"></i></span>
                        </div>
                    </div>
                </div>
            </div>
            <div class="card border-right">
                <div class="card-body">
                    <div class="d-flex d-lg-flex d-md-block align-items-center">
                        <div>
                            <div class="d-inline-flex align-items-center">
                                <h2 class="text-dark mb-1 font-weight-medium"><?php include 'counters/booked-count.php'?></h2>
                            </div>
                            <h6 class="text-muted font-weight-normal mb-0 w-100 text-truncate">Booked Rooms</h6>
                        </div>
```

```
                    <div class="ml-auto mt-md-3 mt-lg-0">
                      <span class="opacity-7 text-muted"><i data-feather="book-open"></i></span>
                    </div>
                  </div>
                </div>
              </div>
              <div class="card">
                <div class="card-body">
                  <div class="d-flex d-lg-flex d-md-block align-items-center">
                    <div>
                      <h2 class="text-dark mb-1 font-weight-medium"><?php include 'counters/course-count.php'?></h2>
                      <h6 class="text-muted font-weight-normal mb-0 w-100 text-truncate">Featured Courses</h6>
                    </div>
                    <div class="ml-auto mt-md-3 mt-lg-0">
                      <span class="opacity-7 text-muted"><i data-feather="globe"></i></span>
                    </div>
                  </div>
                </div>
              </div>
            </div>
            <div class="col-12">
                <div class="card"><div class="card-body">
                  <div class="table-responsive">
                  <table id="zero_config" class="table table-striped table-bordered no-wrap">
                      <thead>
                        <tr>
                          <th scope="col">#</th>
                          <!-- <th scope="col">User ID</th> -->
                          <th scope="col">Student's Email</th>
```

```
                        <th scope="col">Last Activity</th>
                      </tr>
                    </thead>
                    <tbody>
                    <?php
                      $aid=$_SESSION['id'];
                      $ret="SELECT * from userlog ORDER BY loginTime DESC";
                      $stmt= $mysqli->prepare($ret) ;
                      $stmt->execute() ;
                      $res=$stmt->get_result();
                      $cnt=1;
                      while($row=$res->fetch_object()) {
                          ?>
                      <tr><td><?php echo $cnt;;?></td>
                      <!-- <td><?php echo $row->userId;?></td> -->
                      <td><?php echo $row->userEmail;?></td>
                      <td><?php echo $row->loginTime;?></td>
                        </tr>
                        <?php
                      $cnt=$cnt+1;
                        } ?>
                    </tbody>
                  </table>
                </div>
                </div>
              </div>
            </div></div>
              <?php include '../includes/footer.php' ?>
          </div>
</div>
  <script src="../assets/libs/jquery/dist/jquery.min.js"></script>
<script src="../assets/libs/popper.js/dist/umd/popper.min.js"></script>
<script src="../assets/libs/bootstrap/dist/js/bootstrap.min.js"></script>
<script src="../dist/js/app-style-switcher.js"></script>
```

```
    <script src="../dist/js/feather.min.js"></script>
    <script src="../assets/libs/perfect-scrollbar/dist/perfect-scrollbar.jquery.min.js"></script>
    <script src="../dist/js/sidebarmenu.js"></script>
    <!--Custom JavaScript -->
    <script src="../dist/js/custom.min.js"></script>

    <script src="../assets/extra-libs/c3/d3.min.js"></script>
    <script src="../assets/extra-libs/c3/c3.min.js"></script>
    <script src="../assets/libs/chartist/dist/chartist.min.js"></script>
    <script src="../assets/libs/chartist-plugin-tooltips/dist/chartist-plugin-
tooltip.min.js"></script>
    <script src="../dist/js/pages/dashboards/dashboard1.min.js"></script>
    <script src="../assets/extra-libs/datatables.net/js/jquery.dataTables.min.js"></script>
    <script src="../dist/js/pages/datatable/datatable-basic.init.js"></script>
</body>
</html>
```

manage-room.php

```
<?php
    session_start();
    include('../includes/dbconn.php');
    include('../includes/check-login.php');
    check_login();
    if(isset($_GET['del']))
    {
        $id=intval($_GET['del']);
        $adn="DELETE from rooms where id=?";
            $stmt= $mysqli->prepare($adn);
            $stmt->bind_param('i',$id);
            $stmt->execute();
            $stmt->close();
            echo "<script>alert('Record has been deleted');</script>" ;
    }
?>
<!DOCTYPE html>
```

```
<html dir="ltr" lang="en">
<head>
   <meta charset="utf-8">
   <meta http-equiv="X-UA-Compatible" content="IE=edge">
   <meta name="viewport" content="width=device-width, initial-scale=1">
   <meta name="description" content="">
   <meta name="author" content="">
   <link rel="icon" type="image/png" sizes="16x16" href="../assets/images/favicon.png">
   <title>Hostel Management System</title>
   <link href="../assets/extra-libs/c3/c3.min.css" rel="stylesheet">
   <link href="../assets/libs/chartist/dist/chartist.min.css" rel="                <div class="lds-
pos"></div>
         <div class="lds-pos"></div>
      </div>
   </div>
 <div id="main-wrapper" data-theme="light" data-layout="vertical" data-navbarbg="skin6"
data-sidebartype="full"
      data-sidebar-position="fixed" data-header-position="fixed"
      <header class="topbar" data-navbarbg="skin6">
         <?php include 'includes/navigation.php'?>
      </header>
      <aside class="left-sidebar" data-sidebarbg="skin6">

         <div class="scroll-sidebar" data-sidebarbg="skin6">
            <?php include 'includes/sidebar.php'?>
         </div>
            <div class="page-wrapper">
         <div class="page-breadcrumb">
            <div class="row">
               <div class="col-7 align-self-center">
               <h4 class="page-title text-truncate text-dark font-weight-medium mb-1">Rooms
Management</h4>
                  <div class="d-flex align-items-center">
                  </div>
```

```
            </div>
          </div>
        <div class="container-fluid">
             <div class="row">
            <div class="col-12">
              <div class="card">
                <div class="card-body">
                <a href="add-rooms.php"><button type="button" class="btn btn-block btn-md btn-success">Add New Room Details</button></a>
                <hr>
                  <div class="table-responsive">
                    <table id="zero_config" class="table table-striped table-hover table-bordered no-wrap">
                    <thead class="thead-dark">
                        <tr>
                          <th>#</th>
                          <th>Room No.</th>
                          <th>Bed No</th>
                          <th>Room Type</th>
                          <th>Published On</th>
                          <th>Actions</th>
                        </tr>
                      </thead>
                      <tbody>
                      <?php
                        $aid=$_SESSION['id'];
                        $ret="SELECT * from rooms";
                        $stmt= $mysqli->prepare($ret) ;
                        //$stmt->bind_param('i',$aid);
                        $stmt->execute() ;//ok
                        $res=$stmt->get_result();
                        $cnt=1;
                        while($row=$res->fetch_object())
                          {
```

```
                                    ?>
                                <tr><td><?php echo $cnt;;?></td>
                                <td><?php echo $row->room_no;?></td>
                                <td><?php echo $row->bed_no;?></td>
                                <td>$<?php echo $row->room_type;?></td>
                                <td><?php echo $row->posting_date;?></td>
                                <td><a href="edit-room.php?id=<?php echo $row->id;?>"
title="Edit"><i class="icon-note"></i></a>  
                                <a href="manage-rooms.php?del=<?php echo $row->id;?>"
title="Delete" onclick="return confirm("Do you want to delete");"><i class="icon-close"
style="color:red;"></i></a></td>
                                </tr>
                                  <?php
                                    $cnt=$cnt+1;
                                  } ?>
                                </tbody>
                              </table>
                            </div>
                          </div>
      </div>
    </div>
    <script src="../assets/libs/jquery/dist/jquery.min.js"></script>
    <script src="../assets/libs/popper.js/dist/umd/popper.min.js"></script>
    <script src="../assets/libs/bootstrap/dist/js/bootstrap.min.js"></script>
    <script src="../dist/js/app-style-switcher.js"></script>
    <script src="../dist/js/feather.min.js"></script>
    <script src="../assets/libs/perfect-scrollbar/dist/perfect-scrollbar.jquery.min.js"></script>
    <script src="../dist/js/sidebarmenu.js"></script>
    <script src="../dist/js/custom.min.js"></script>
    <script src="../assets/extra-libs/c3/d3.min.js"></script>
    <script src="../assets/extra-libs/c3/c3.min.js"></script>
    <script src="../assets/libs/chartist/dist/chartist.min.js"></script>
    <script src="../assets/libs/chartist-plugin-tooltips/dist/chartist-plugin-
tooltip.min.js"></script>
```

```
    <script src="../dist/js/pages/dashboards/dashboard1.min.js"></script>
    <script src="../assets/extra-libs/datatables.net/js/jquery.dataTables.min.js"></script>
    <script src="../dist/js/pages/datatable/datatable-basic.init.js"></script>
</body>
</html>
```

manage-students.php

```
<?php
    session_start();
    include('../includes/dbconn.php');
    include('../includes/check-login.php');
    check_login();
    if(isset($_GET['del'])) {
        $id=intval($_GET['del']);
        $adn="DELETE from registration where id=?";
            $stmt= $mysqli->prepare($adn);
            $stmt->bind_param('i',$id);
            $stmt->execute();
            $stmt->close();
            echo "<script>alert('Record has been deleted');</script>" ;
    }
?>
<!DOCTYPE html>
<html dir="ltr" lang="en">
<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <!-- Tell the browser to be responsive to screen width -->
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta name="description" content="">
    <meta name="author" content="">
    <link rel="icon" type="image/png" sizes="16x16" href="../assets/images/favicon.png">
    <title>Hostel Management System</title>
    <link href="../assets/extra-libs/c3/c3.min.css" rel="stylesheet">
```

```
    <link href="../assets/libs/chartist/dist/chartist.min.css" rel="stylesheet">
    <link href="../assets/extra-libs/datatables.net-bs4/css/dataTables.bootstrap4.css"
rel="stylesheet">
    <link href="../dist/css/style.min.css" rel="stylesheet">
    <script language="javascript" type="text/javascript">
    var popUpWin=0;
    function popUpWindow(URLStr, left, top, width, height){
      if(popUpWin) {
       if(!popUpWin.closed) popUpWin.close();
        }
        popUpWin = open(URLStr,'popUpWin',
'toolbar=no,location=no,directories=no,status=no,menubar=no,scrollbars=yes,resizable=no,c
opyhistory=yes,width='+510+',height='+430+',left='+left+',
top='+top+',screenX='+left+',screenY='+top+");
      }
    </script>
</head>
<body>
    <div class="preloader">
      <div class="lds-ripple">
        <div class="lds-pos"></div>
        <div class="lds-pos"></div>
      </div>
    </div>
    <div id="main-wrapper" data-theme="light" data-layout="vertical" data-navbarbg="skin6"
data-sidebartype="full"
      data-sidebar-position="fixed" data-header-position="fixed" data-boxed-layout="full">
      <header class="topbar" data-navbarbg="skin6">
        <?php include 'includes/navigation.php'?>
      </header>
      <aside class="left-sidebar" data-sidebarbg="skin6">
        <div class="scroll-sidebar" data-sidebarbg="skin6">
          <?php include 'includes/sidebar.php'?>
        </div>
```

```
</aside>
<div class="page-wrapper">
  <div class="page-breadcrumb">
    <div class="row">
      <div class="col-7 align-self-center">
      <h4 class="page-title text-truncate text-dark font-weight-medium mb-1">Hostel Student Management</h4>
        <div class="d-flex align-items-center">
        </div>
      </div>

    </div>
  </div>
  <div class="container-fluid">

    <!-- Table Starts -->
    <div class="row">
      <div class="col-12">
        <div class="card">
          <div class="card-body">
            <h6 class="card-subtitle">Displaying all the registered students list.</h6>
            <div class="table-responsive">
              <table id="zero_config" class="table table-striped table-hover table-bordered no-wrap">
                <thead class="thead-dark">
                    <tr>
                      <th>#</th>
                      <th>Reg. No.</th>
                      <th>Student's Name</th>
                      <th>Room No.</th>
                      <th>Seater</th>
                      <th>Staying From</th>
                      <th>Contact</th>
```

```
                    <th>Actions</th>
                  </tr>
                </thead>
                <tbody>
                <?php
                $aid=$_SESSION['id'];
                 $ret = "SELECT ur.admno, ur.name, r.bedno, r.roomno, r.stayfrom,
ur.contactNo
                FROM userregistration ur
                JOIN registration r
                ON ur.admno = r.admno WHERE r.status='approved'";
                $stmt= $mysqli->prepare($ret) ;
                $stmt->execute() ;//ok
                $res=$stmt->get_result();
                $cnt=1;
                while($row=$res->fetch_object())
                  {
                    ?>
                <tr><td><?php echo $cnt;;?></td>
                <td><?php echo $row->admno;?></td>
                <td><?php echo $row->name;?> </td>
                <td><?php echo $row->roomno;?></td>
                <td><?php echo $row->bedno;?></td>
                <td><?php echo $row->stayfrom;?></td>
                <td><?php echo $row->contactNo;?></td>
                <td>
                            <a href="students-profile.php?admno=<?php echo $row-
>admno;?>" title="View Full Details"><i class="icon-size-
fullscreen"></i></a>  
                    <a href="manage-students.php?del=<?php echo $row->admno;?>"
title="Delete Record" onclick="return confirm("Do you want to delete");"><i class="icon-
close" style="color:red;"></i></a></td>
                </tr>
                  <?php
```

```
                                $cnt=$cnt+1;
                                } ?>
                                </tbody>
                            </table>
                        </div>
                    </div>
                </div>
            </div>
        </div>

        <?php include '../includes/footer.php' ?>
    </div>
</div>
<script src="../assets/libs/jquery/dist/jquery.min.js"></script>
<script src="../assets/libs/popper.js/dist/umd/popper.min.js"></script>
<script src="../assets/libs/bootstrap/dist/js/bootstrap.min.js"></script>
<script src="../dist/js/app-style-switcher.js"></script>
<script src="../dist/js/feather.min.js"></script>
<script src="../assets/libs/perfect-scrollbar/dist/perfect-scrollbar.jquery.min.js"></script>
<script src="../dist/js/sidebarmenu.js"></script>
<script src="../dist/js/custom.min.js"></script>
<script src="../assets/extra-libs/c3/d3.min.js"></script>
<script src="../assets/extra-libs/c3/c3.min.js"></script>
<script src="../assets/libs/chartist/dist/chartist.min.js"></script>
<script src="../assets/libs/chartist-plugin-tooltips/dist/chartist-plugin-
tooltip.min.js"></script>
<script src="../dist/js/pages/dashboards/dashboard1.min.js"></script>
<script src="../assets/extra-libs/datatables.net/js/jquery.dataTables.min.js"></script>
<script src="../dist/js/pages/datatable/datatable-basic.init.js"></script>

</body>
</html>
```

Warden module

index.php

```
<?php
   session_start();
   include('../includes/dbconn.php');
   if(isset($_POST['login'])){
   $username=$_POST['username'];
   $password=$_POST['password'];
   $stmt=$mysqli->prepare("SELECT username,email,password,id FROM warden WHERE (userName=?|| email=?) and password=? ");
      $stmt->bind_param('sss',$username,$username,$password);
      $stmt->execute();
      $stmt -> bind_result($username,$username,$password,$id);
      $rs=$stmt->fetch();
      $_SESSION['id']=$id;
      $uip=$_SERVER['REMOTE_ADDR'];
      $ldate=date('d/m/Y h:i:s', time());
      if($rs){
         header("location:dashboard.php");
            } else {
               echo "<script>alert('Invalid Username/Email or password');</script>";
            }
   }
?>
<!DOCTYPE html>
<html dir="ltr">
<head>
   <meta charset="utf-8">
   <meta http-equiv="X-UA-Compatible" content="IE=edge">
   <meta name="viewport" content="width=device-width, initial-scale=1">
   <meta name="description" content="">
   <link rel="icon" type="image/png" sizes="16x16" href="../assets/images/favicon.png"><link href="../dist/css/style.min.css" rel="stylesheet">
```

```
<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>
<div class="main-wrapper">
   <div class="preloader">
      <div class="lds-ripple">
         <div class="lds-pos"></div>
         <div class="lds-pos"></div>
      </div>
   </div>
      <div class="auth-wrapper d-flex no-block justify-content-center align-items-center position-relative"
      style="background:url(../assets/images/big/auth-bg.jpg) no-repeat center center;">
      <div class="auth-box row">
         <div class="col-lg-7 col-md-5 modal-bg-img" style="background-image: url(../assets/images/adimg.jpg);">
         </div>
         <div class="col-lg-5 col-md-7 bg-white">
            <div class="p-3">
               <div class="text-center">
                  <img src="../assets/images/big/login.png" alt="wrapkit">
               </div>
               <h2 class="mt-3 text-center">Warden Login</h2>

               <form class="mt-4" method="POST">
```

```
                    <div class="row">
                      <div class="col-lg-12">
                        <div class="form-group">
                          <label class="text-dark" for="uname">Email or Username</label>
                          <input class="form-control" name="username" id="uname"
type="text"
                            placeholder="Email or Username" required>
                        </div>
                      </div>
                      <div class="col-lg-12">
                        <div class="form-group">
                          <label class="text-dark" for="pwd">Password</label>
                          <input class="form-control" name="password" id="pwd"
type="password"
                            placeholder="Enter your password" required>
                        </div>
                      </div>
                      <div class="col-lg-12 text-center">
                        <button type="submit" name="login" class="btn btn-block btn-
danger">LOGIN</button>
                      </div>
                      <div class="col-lg-12 text-center mt-5">
                        <a href="../index.html" class="text-danger">Go Back</a>
                      </div>
                    </div>
                  </form>
                </div>
              </div>
            </div>
          </div>
    <script src="../assets/libs/jquery/dist/jquery.min.js "></script>
    <!-- Bootstrap tether Core JavaScript -->
    <script src="../assets/libs/popper.js/dist/umd/popper.min.js "></script>
    <script src="../assets/libs/bootstrap/dist/js/bootstrap.min.js "></script>
```

```
    <script>
        $(".preloader ").fadeOut();
    </script>
</body>

</html>
```

dashboard.php

```
<?php
    session_start();
    include('../includes/dbconn.php');
    include('../includes/check-login.php');
    check_login();

?>

<!DOCTYPE html>
<html dir="ltr" lang="en">
<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <!-- Tell the browser to be responsive to screen width -->
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta name="description" content="">
    <meta name="author" content="">
    <!-- Favicon icon -->
    <link rel="icon" type="image/png" sizes="16x16" href="../assets/images/favicon.png">
    <title>Hostel Management System</title>
    <!-- Custom CSS -->
    <link href="../assets/extra-libs/c3/c3.min.css" rel="stylesheet">
    <link href="../assets/libs/chartist/dist/chartist.min.css" rel="stylesheet">
    <link                href="../assets/extra-libs/datatables.net-bs4/css/dataTables.bootstrap4.css"
rel="stylesheet">
    <!-- Custom CSS -->
```

```
    <link href="../dist/css/style.min.css" rel="stylesheet">
</head>
<body>
    <div class="preloader">
        <div class="lds-ripple">
            <div class="lds-pos"></div>
            <div class="lds-pos"></div>
        </div>
    </div>
    <div id="main-wrapper" data-theme="light" data-layout="vertical" data-navbarbg="skin6"
data-sidebartype="full"
        data-sidebar-position="fixed" data-header-position="fixed" data-boxed-layout="full">
            <?php include 'includes/navigation.php'?>
        </header>
        <aside class="left-sidebar" data-sidebarbg="skin6">
            <!-- Sidebar scroll-->
            <div class="scroll-sidebar" data-sidebarbg="skin6">
                <?php include 'includes/sidebar.php'?>
            </div>
            <!-- End Sidebar scroll-->
        </aside>
        <!-- End Left Sidebar - style you can find in sidebar.scss  -->
            <div class="page-breadcrumb">
                <div class="row">
                    <div class="col-7 align-self-center">
                        <?php include 'includes/greetings-a.php'?>
                    </div>
                </div>
                <div class="card-group">
                    <div class="card border-right">
                        <div class="card-body">
                            <div class="d-flex d-lg-flex d-md-block align-items-center">
                                <div>
                                    <div class="d-inline-flex align-items-center">
```

```
                        <h2 class="text-dark mb-1 font-weight-medium"><?php include 'counters/student-count.php'?></h2>
                        </div>
                        <h6 class="text-muted font-weight-normal mb-0 w-100 text-truncate">Registered Students</h6>
                    </div>
                    <div class="ml-auto mt-md-3 mt-lg-0">
                        <span class="opacity-7 text-muted"><i data-feather="user-plus"></i></span>
                    </div>
                </div>
            </div>
            <div class="card border-right">
                <div class="card-body">
                    <div class="d-flex d-lg-flex d-md-block align-items-center">
                        <div>
                            <h2 class="text-dark mb-1 w-100 text-truncate font-weight-medium"><?php include 'counters/room-count.php'?></h2>
                            <h6 class="text-muted font-weight-normal mb-0 w-100 text-truncate">Total Rooms </h6>
                        </div>
                        <div class="ml-auto mt-md-3 mt-lg-0">
                            <span class="opacity-7 text-muted"><i data-feather="grid"></i></span>
                        </div>
                    </div>
                </div>
            </div>
            <div class="card border-right">
                <div class="card-body">
                    <div class="d-flex d-lg-flex d-md-block align-items-center">
                        <div>
                            <div class="d-inline-flex align-items-center">
```

```
                        <h2 class="text-dark mb-1 font-weight-medium"><?php include 'counters/booked-count.php'?></h2>
                        </div>
                        <h6 class="text-muted font-weight-normal mb-0 w-100 text-truncate">Booked Rooms</h6>
                    </div>
                    <div class="ml-auto mt-md-3 mt-lg-0">
                        <span class="opacity-7 text-muted"><i data-feather="book-open"></i></span>
                    </div>
                </div>
            </div>
            <div class="card">
                <div class="card-body">
                    <div class="d-flex d-lg-flex d-md-block align-items-center">
                        <div>
                            <h2 class="text-dark mb-1 font-weight-medium"><?php include 'counters/course-count.php'?></h2>
                            <h6 class="text-muted font-weight-normal mb-0 w-100 text-truncate">Featured Courses</h6>
                        </div>
                        <div class="ml-auto mt-md-3 mt-lg-0">
                            <span class="opacity-7 text-muted"><i data-feather="globe"></i></span>
                        </div>
                    </div>
                </div>
            </div>
        </div
        <div class="col-12">
            <div class="card">
                <div class="card-body">
                <div class="table-responsive">
```

```
<table id="zero_config" class="table table-striped table-bordered no-wrap">
                    <thead>
                      <tr>
                        <th scope="col">#</th>
                        <!-- <th scope="col">User ID</th> -->
                        <th scope="col">Student's Email</th>
                        <th scope="col">Last Activity</th>
                      </tr>
                    </thead>
                    <tbody>
                    <?php
                      $aid=$_SESSION['id'];
                      $ret="SELECT * from userlog ORDER BY loginTime DESC";
                      $stmt= $mysqli->prepare($ret) ;
                      $stmt->execute() ;
                      $res=$stmt->get_result();
                      $cnt=1;
                      while($row=$res->fetch_object()) {
                            ?>
                      <tr><td><?php echo $cnt;;?></td>
                      <!-- <td><?php echo $row->userId;?></td> -->
                      <td><?php echo $row->userEmail;?></td>
                      <td><?php echo $row->loginTime;?></td>
                        </tr>
                        <?php
                      $cnt=$cnt+1;
                        } ?>
                    </tbody>
                  </table>
                </div>
                </div>
              </div>
            </div>
```

```
        </div>
        <?php include '../includes/footer.php' ?>
      </div>
   </div>
   <script src="../assets/libs/jquery/dist/jquery.min.js"></script>
   <script src="../assets/libs/popper.js/dist/umd/popper.min.js"></script>
   <script src="../assets/libs/bootstrap/dist/js/bootstrap.min.js"></script>
   <script src="../dist/js/app-style-switcher.js"></script>
   <script src="../dist/js/feather.min.js"></script>
   <script src="../assets/libs/perfect-scrollbar/dist/perfect-scrollbar.jquery.min.js"></script>
   <script src="../dist/js/sidebarmenu.js"></script>
    <script src="../dist/js/custom.min.js"></script>
     <script src="../assets/extra-libs/c3/d3.min.js"></script>
   <script src="../assets/extra-libs/c3/c3.min.js"></script>
   <script src="../assets/libs/chartist/dist/chartist.min.js"></script>
   <script                    src="../assets/libs/chartist-plugin-tooltips/dist/chartist-plugin-
tooltip.min.js"></script>
   <script src="../dist/js/pages/dashboards/dashboard1.min.js"></script>
   <script src="../assets/extra-libs/datatables.net/js/jquery.dataTables.min.js"></script>
   <script src="../dist/js/pages/datatable/datatable-basic.init.js"></script>
</body>


</html>
```

attendence.php

```
<?php
   session_start();
   include('../includes/dbconn.php');
   include('../includes/check-login.php');
   check_login();
   if(isset($_POST['submit'])){
      $q=$_POST['adate'];
      $a=$_POST['admno'];
      $e=$_POST['att'];
      $f=$_POST['remark'];
```

```
    foreach($a as $index => $ae)
  {
    $qq=$q;
    $aa = $ae;
    $ee = $e[$index];
    $ff = $f[$index];

                    $query="INSERT into attendence(admno,attendence,remark,date)
values('$aa','$ee','$ff','$qq')";
    $result=mysqli_query($mysqli,$query);
  }
        echo"<script>alert('attendence recorded');
    window.location.href='dashboard.php';
    </script>";
    }
?>
<!DOCTYPE html>
<html dir="ltr" lang="en">
<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <!-- Tell the browser to be responsive to screen width -->
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <meta name="description" content="">
  <meta name="author" content="">
  <link rel="icon" type="image/png" sizes="16x16" href="../assets/images/favicon.png">
  <link href="../assets/extra-libs/c3/c3.min.css" rel="stylesheet">
  <link href="../assets/libs/chartist/dist/chartist.min.css" rel="stylesheet">
        <link href="../assets/extra-libs/datatables.net-bs4/css/dataTables.bootstrap4.css"
rel="stylesheet">
   <link href="../dist/css/style.min.css" rel="stylesheet">
  <script language="javascript" type="text/javascript">
  var popUpWin=0;
  function popUpWindow(URLStr, left, top, width, height){
```

```
        if(popUpWin) {
         if(!popUpWin.closed) popUpWin.close();
            }
                                                    popUpWin     =     open(URLStr,'popUpWin',
'toolbar=no,location=no,directories=no,status=no,menubar=no,scrollbars=yes,resizable=no,c
opyhistory=yes,width='+510+',height='+430+',left='+left+',
top='+top+',screenX='+left+',screenY='+top+");
        }
    </script>

</head>

<body>
    <div class="preloader">
        <div class="lds-ripple">
            <div class="lds-pos"></div>
            <div class="lds-pos"></div>
        </div>
    </div>
    <div id="main-wrapper" data-theme="light" data-layout="vertical" data-navbarbg="skin6"
data-sidebartype="full"
        data-sidebar-position="fixed" data-header-position="fixed" data-boxed-layout="full">
            <header class="topbar" data-navbarbg="skin6">
            <?php include 'includes/navigation.php'?>
        </header>
     <aside class="left-sidebar" data-sidebarbg="skin6">
                    <div class="scroll-sidebar" data-sidebarbg="skin6">
              <?php include 'includes/sidebar.php'?>
            </div>
                  </aside>
              <div class="page-breadcrumb">
              <div class="row">
                  <div class="col-7 align-self-center">
```

```
                    <h4 class="page-title text-truncate text-dark font-weight-medium mb-1">Attendence </h4>
        <div class="container-fluid">
        <form method="POST">
          <!-- Table Starts -->
          <div class="row">
            <div class="col-12">
              <div class="card">
                <div class="card-body">
                  <h4 class="card-title"> Date</h4>
                    <div class="form-group">
                        <input type="date" name="adate" id="adate" class="form-control" required>
                    </div>
                  <div class="table-responsive">
                      <table id="zero_config" class="table table-striped table-hover table-bordered no-wrap">                                    <thead class="thead-dark">
                        <tr>
                          <th>#</th>
                          <th>Adm. No.</th>
                          <th>Student's Name</th>
                          <th>Bed No</th>
                          <th>Room No</th>
                          <th>Attendence</th>
                          <th>Remark</th>

                          <th></th>
                        </tr>
                      </thead>
                      <tbody>
                      <?php
                      $aid=$_SESSION['id'];
                      $ret = "SELECT u.id, u.admno, u.name, r.roomno, r.bedno
                      FROM userregistration u
```

```
                    JOIN registration r ON u.admno = r.admno
                    WHERE r.status = 'approved'";
                    $stmt= $mysqli->prepare($ret) ;
                    $stmt->execute() ;//ok
                    $res=$stmt->get_result();
                    $cnt=1;
                    while($row=$res->fetch_object())
                      {
                        ?>
                    <tr>
                            <td ><input type="text" name="slno"  style="width:50px;
background: transparent; border: none;" value="<?php echo $cnt;;?>" readonly></td>
                          <td ><input type="text" name="admno[]"  style="width:80px;
background: transparent; border: none;" value="<?php echo $row->admno;?>"
readonly></td>
                          <td ><input type="text" name="name[]"  style="width:80px;
background: transparent; border: none;" value="<?php echo $row->name;?>" readonly></td>
                          <td ><input type="text" name="bed[]"  style="width:80px;
background: transparent; border: none;" value="<?php echo $row->bedno;?>"
readonly></td>
                          <td ><input type="text" name="room[]"  style="width:80px;
background: transparent; border: none;" value="<?php echo $row->roomno;?>"
readonly></td>

                          <td><input type="checkbox" name="att[]" value="present"
id="">Present<br>
                              <input type="checkbox" name="att[]" value="absent"
id="">Absent</td>
                    <td><input type="text" name="remark[]" id="remark"></td>
                     </tr>
                      <?php
                    $cnt=$cnt+1;
                      } ?>
                          </tbody>
```

```
                    </table>
                  </div>
                </div>
              </div>
            </div>
          </div>
          <div class="form-actions">
    <div class="text-center">
      <button type="submit" name="submit" class="btn btn-success">Submit</button>
      <button type="reset" class="btn btn-dark">Reset</button>
    </div>
  </div>
          </form>
        </div>

        <?php include '../includes/footer.php' ?>
        </div>
  <script src="../assets/libs/jquery/dist/jquery.min.js"></script>
  <script src="../assets/libs/popper.js/dist/umd/popper.min.js"></script>
  <script src="../assets/libs/bootstrap/dist/js/bootstrap.min.js"></script>
  <!-- apps -->
  <!-- apps -->
  <script src="../dist/js/app-style-switcher.js"></script>
  <script src="../dist/js/feather.min.js"></script>
  <script src="../assets/libs/perfect-scrollbar/dist/perfect-scrollbar.jquery.min.js"></script>
  <script src="../dist/js/sidebarmenu.js"></script>
  <!--Custom JavaScript -->
  <script src="../dist/js/custom.min.js"></script>
  <script src="../assets/extra-libs/c3/c3.min.js"></script>
  <script src="../assets/libs/chartist/dist/chartist.min.js"></script>
                    <script       src="../assets/libs/chartist-plugin-tooltips/dist/chartist-plugin-
tooltip.min.js"></script>
  <script src="../dist/js/pages/dashboards/dashboard1.min.js"></script>
  <script src="../assets/extra-libs/datatables.net/js/jquery.dataTables.min.js"></script>
```

```
    <script src="../dist/js/pages/datatable/datatable-basic.init.js"></script>
</body>
</html>
```

Student Module

dashboard.php

```
<?php
    session_start();
    include('../includes/dbconn.php');
    include('../includes/check-login.php');
    check_login();
?>
<!DOCTYPE html>
<html dir="ltr" lang="en">
<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
      <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta name="description" content="">
    <meta name="author" content="">
       <link rel="icon" type="image/png" sizes="16x16" href="../assets/images/favicon.png">
    <title>Hostel Management System</title>
       <link href="../assets/extra-libs/c3/c3.min.css" rel="stylesheet">
    <link href="../assets/libs/chartist/dist/chartist.min.css" rel="stylesheet">
      <link href="../dist/css/style.min.css" rel="stylesheet">
</head>

<body>
       <div class="preloader">
       <div class="lds-ripple">
          <div class="lds-pos"></div>
          <div class="lds-pos"></div>
```

```
      </div>
   </div>
     <div id="main-wrapper" data-theme="light" data-layout="vertical" data-navbarbg="skin6" data-sidebartype="full"
      data-sidebar-position="fixed" data-header-position="fixed" data-boxed-layout="full">
          <header class="topbar" data-navbarbg="skin6">
        <?php include '../includes/student-navigation.php'?>
      </header>
            <aside class="left-sidebar" data-sidebarbg="skin6">
        <!-- Sidebar scroll-->
        <div class="scroll-sidebar" data-sidebarbg="skin6">
          <?php include '../includes/student-sidebar.php'?>
        </div>
        <!-- End Sidebar scroll-->
      </aside>
        <div class="page-breadcrumb">
          <div class="row">
            <div class="col-7 align-self-center">
              <?php include '../includes/greetings.php'?>
               <div class="d-flex align-items-center">
          </div>
        </div>
                 <div class="card-group">
            <div class="card border-right">
              <div class="card-body">
                <div class="d-flex d-lg-flex d-md-block align-items-center">
                  <div>
                    <div class="d-inline-flex align-items-center">
                      <h2 class="text-dark mb-1 font-weight-medium"><?php include 'counters/student-count.php'?></h2>
                    </div>
                    <h6 class="text-muted font-weight-normal mb-0 w-100 text-truncate">Registered Students</h6>
                  </div>
```

```
                    <div class="ml-auto mt-md-3 mt-lg-0">
                      <span class="opacity-7 text-muted"><i data-feather="user-plus"></i></span>
                    </div>
                  </div>
                </div>

                <div class="card border-right">
                  <div class="card-body">
                    <div class="d-flex d-lg-flex d-md-block align-items-center">
                      <div>
                        <h2 class="text-dark mb-1 w-100 text-truncate font-weight-medium"><?php include 'counters/room-count.php'?></h2>
                        <h6 class="text-muted font-weight-normal mb-0 w-100 text-truncate">Total Rooms
                        </h6>
                      </div>
                      <div class="ml-auto mt-md-3 mt-lg-0">
                        <span class="opacity-7 text-muted"><i data-feather="grid"></i></span>
                      </div>
                    </div>
                  </div>
                </div>
                <div class="card border-right">
                  <div class="card-body">
                    <div class="d-flex d-lg-flex d-md-block align-items-center">
                      <div>
                        <div class="d-inline-flex align-items-center">
                          <h2 class="text-dark mb-1 font-weight-medium"><?php include 'counters/booked-count.php'?></h2>
                        </div>
                        <h6 class="text-muted font-weight-normal mb-0 w-100 text-
```

```
truncate">Booked Rooms</h6>
                    </div>
                    <div class="ml-auto mt-md-3 mt-lg-0">
                      <span class="opacity-7 text-muted"><i data-feather="book-open"></i></span>
                    </div>
                  </div>
                </div>
              </div>
              <div class="card">
                <div class="card-body">
                  <div class="d-flex d-lg-flex d-md-block align-items-center">
                    <div>
                      <h2 class="text-dark mb-1 font-weight-medium"><?php include 'counters/course-count.php'?></h2>
                      <h6 class="text-muted font-weight-normal mb-0 w-100 text-truncate">Featured Courses</h6>
                    </div>
                    <div class="ml-auto mt-md-3 mt-lg-0">
                      <span class="opacity-7 text-muted"><i data-feather="globe"></i></span>
                    </div>
                  </div>
                </div>
              </div>
            </div>

        </div>
        <script src="../assets/libs/jquery/dist/jquery.min.js"></script>
    <script src="../assets/libs/popper.js/dist/umd/popper.min.js"></script>
    <script src="../assets/libs/bootstrap/dist/js/bootstrap.min.js"></script>
    <!-- apps -->
    <script src="../dist/js/app-style-switcher.js"></script>
    <script src="../dist/js/feather.min.js"></script>
```

```
    <script src="../assets/libs/perfect-scrollbar/dist/perfect-scrollbar.jquery.min.js"></script>
    <script src="../dist/js/sidebarmenu.js"></script>
    <!--Custom JavaScript -->
    <script src="../dist/js/custom.min.js"></script>
    <!--This page JavaScript -->
    <script src="../assets/extra-libs/c3/d3.min.js"></script>
    <script src="../assets/extra-libs/c3/c3.min.js"></script>
    <script src="../assets/libs/chartist/dist/chartist.min.js"></script>
    <script src="../assets/libs/chartist-plugin-tooltips/dist/chartist-plugin-
tooltip.min.js"></script>
    <script src="../dist/js/pages/dashboards/dashboard1.min.js"></script>
</body>

</html>
```

book-hostel.php

```
<?php
    session_start();
    include('../includes/dbconn.php');
    include('../includes/check-login.php');
    check_login();
    if(isset($_POST['submit'])){
        $ano=$_POST['admno'];
        $phone=$_POST['contact'];
        $dob=$_POST['dob'];
        $age=$_POST['age'];
        $caste=$_POST['caste'];
        $gender=$_POST['sex'];
        $food=$_POST['food'];
        $gname=$_POST['gname'];
        $grel=$_POST['grelation'];
        $gcon=$_POST['gcontact'];
        $addr=$_POST['address'];
```

```
        $pin=$_POST['pincode'];
        $district=$_POST['district'];
        $dist=$_POST['distance'];
        $image = $_FILES["photo"]["name"];
        $image_tmp = $_FILES["photo"]["tmp_name"];
        $folder="../photo/".$image;
        move_uploaded_file($image_tmp,$folder);
        $idcard = $_FILES["idcard"]["name"];
        $id_tmp = $_FILES["idcard"]["tmp_name"];
        $folder1="../idcard/".$idcard;
        move_uploaded_file($id_tmp,$folder);
        $cc = $_FILES["cc"]["name"];
        $cc_tmp = $_FILES["cc"]["tmp_name"];
        $folder2="../caste/".$cc;

                                                $query="insert into
book_hostel(admno,dob,age,caste,gender,food,local_gname,local_grelation,local_phone,addr
ess,pincode,district,distance,photo,idcard,c_certificate)
values('$ano','$dob','$age','$caste','$gender','$food','$gname','$grel','$gcon','$addr','$pin','$dist
rict','$dist','$folder','$folder1','$folder2')";
    $result=mysqli_query($mysqli,$query);
  if($result)
  {
    echo "<script>alert('Hostel Requested!')</script>";
    // echo "<script>window.location.href='dashboard.php'</script>";
  }else{
    echo "<script>alert('Hostel Request failed!')</script>";
    echo "<script>window.location.href='dashboard.php'</script>";
  }



    }
?>
```

```
<!DOCTYPE html>
<html dir="ltr" lang="en">
<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta name="description" content="">
    <meta name="author" content="">
    <link rel="icon" type="image/png" sizes="16x16" href="../assets/images/favicon.png">
    <title>Hostel Management System</title>
    <link href="../assets/extra-libs/c3/c3.min.css" rel="stylesheet">
    <link href="../assets/libs/chartist/dist/chartist.min.css" rel="stylesheet">
    <link href="../dist/css/style.min.css" rel="stylesheet">
    function getSeater(val) {
        $.ajax({
        type: "POST",
        url: "get-seater.php",
        data:'roomid='+val,
        success: function(data){
        //alert(data);
        $('#seater').val(data);
        }
        });
        $.ajax({
        type: "POST",
        url: "get-seater.php",
        data:'rid='+val,
        success: function(data){
        //alert(data);
        $('#fpm').val(data);
        }
        });
    }
```

```
   </script> -->
</head>
<body>
     <div class="preloader">
       <div class="lds-ripple">
          <div class="lds-pos"></div>
          <div class="lds-pos"></div>
       </div>
   </div>
    <div id="main-wrapper" data-theme="light" data-layout="vertical" data-navbarbg="skin6" data-sidebartype="full"
       data-sidebar-position="fixed" data-header-position="fixed" data-boxed-layout="full">
            <header class="topbar" data-navbarbg="skin6">
          <?php include '../includes/student-navigation.php'?>
       </header>
            <aside class="left-sidebar" data-sidebarbg="skin6">
          <!-- Sidebar scroll-->
          <div class="scroll-sidebar" data-sidebarbg="skin6">
             <?php include '../includes/student-sidebar.php'?>
          </div>
          <!-- End Sidebar scroll-->
       </aside>
              <?php
                 $uid=$_SESSION['login'];
                 $stmt=$mysqli->prepare("SELECT email FROM book_hostel WHERE email=? ");
                 $stmt->bind_param('s',$uid);
                 $stmt->execute();
                 $stmt -> bind_result($email);
                 $rs=$stmt->fetch();
                 $stmt->close();
                 if($rs){ ?>
                 <div class="alert alert-primary alert-dismissible bg-danger text-white border-0 fade show"
```

```
                role="alert">
                <button type="button" class="close" data-dismiss="alert" aria-label="Close">
                    <span aria-hidden="true">&times;</span>
                </button>
                    <strong>Info: </strong> You have already booked a hostel!
            </div>
            <?php }
            else{
                echo "";
            }
        ?>
        <div class="col-7 align-self-center">
            <h4 class="page-title text-truncate text-dark font-weight-medium mb-1">Hostel Bookings</h4>
            </div>
        <div class="row">
        </div>
        <h4 class="card-title mt-5">Student's Personal Information</h4>
<div class="row">
        <?php
        $aid=$_SESSION['id'];
            $ret="select * from userregistration where id=?";
                $stmt= $mysqli->prepare($ret) ;
            $stmt->bind_param('i',$aid);
            $stmt->execute();
            $res=$stmt->get_result();
            while($row=$res->fetch_object())
            {
                ?>
            <div class="col-sm-12 col-md-6 col-lg-4">
                <div class="card">
                    <div class="card-body">
                        <h4 class="card-title">Admission Number</h4>
                            <div class="form-group">
```

```
                        <input type="text" name="admno" id="admno" value="<?php
echo $row->admNo;?>" class="form-control" readonly>
                    </div>
                </div>
            </div>
        </div>
        <div class="col-sm-12 col-md-6 col-lg-4">
            <div class="card">
                <div class="card-body">
                    <h4 class="card-title">Name</h4>
                        <div class="form-group">
                            <input type="text" name="name" id="name" value="<?php echo
$row->name;?>" class="form-control" readonly>
                        </div>
                </div>
            </div>
        <div class="col-sm-12 col-md-6 col-lg-4">
            <div class="card">
                <div class="card-body">
                    <h4 class="card-title">Course</h4>
                    <div class="form-group">
                        <input type="text" name="course" id="course" value="<?php echo
$row->course;?>" class="form-control" readonly>
                    </div>
                </div>
            </div>
        </div>
        <div class="col-sm-12 col-md-6 col-lg-4">
            <div class="card">
                <div class="card-body">
                    <h4 class="card-title">Semester</h4>
                    <div class="form-group">
                        <input type="text" name="sem" id="sem" value="<?php echo $row-
>semester;?>" class="form-control" readonly>
```

```
                    </div>
                </div>
            </div>
        </div>
        <div class="col-sm-12 col-md-6 col-lg-4">
            <div class="card">
                <div class="card-body">
                    <h4 class="card-title">Email</h4>
                        <div class="form-group">
                            <input type="email" name="email" id="email" value="<?php echo
$row->email;?>" class="form-control" readonly>
                        </div>
                </div>
            </div>
        </div>
        <div class="col-sm-12 col-md-6 col-lg-4">
            <div class="card">
                <div class="card-body">
                    <h4 class="card-title">Sex</h4>
                    <div class="form-group mb-4">
                        <select class="custom-select mr-sm-2" id="sex" name="sex">
                            <option selected>Please Select...</option>
                            <option >Male</option>
                            <option>Female</option>
                        </select>
                    </div>
                </div>
            </div>
        </div>
        <div class="col-sm-12 col-md-6 col-lg-4">
            <div class="card">
                <div class="card-body">
                    <h4 class="card-title">Contact Number</h4>
                        <div class="form-group">
```

```
                        <input type="text" name="contact" id="contact" value="<?php echo
$row->contactNo;?>" class="form-control" readonly>
                    </div>
                </div>
            </div>
            <?php }?>

            <div class="col-sm-12 col-md-6 col-lg-4">
                <div class="card">
                    <div class="card-body">
                        <h4 class="card-title">Birth Date</h4>
                            <div class="form-group">
                                <input type="date" name="dob" id="dob" class="form-control"
required>
                            </div>
                    </div>
                </div>
            </div>
            <div class="col-sm-12 col-md-6 col-lg-4">
                <div class="card">
                    <div class="card-body">
                        <h4 class="card-title">Age</h4>
                            <div class="form-group">
                                <input type="text" name="age" id="age" class="form-control"
required>
                            </div>
                    </div>
                </div>
            </div>
            <div class="col-sm-12 col-md-6 col-lg-4">
                <div class="card">
                    <div class="card-body">
                        <h4 class="card-title">Caste</h4>
```

```
            <div class="form-group mb-4">
                <select class="custom-select mr-sm-2" id="caste" name="caste">
                  <option selected>Please Select...</option>
                  <option >SC</option>
                  <option>ST</option>
                  <option>OEC</option>
                  <option>OBC</option>
                  <option>GENERAL</option>
                  <option>OTHERES</option>

                </select>
              </div>
          </div>
      </div>
    </div>
    <div class="col-sm-12 col-md-6 col-lg-4">
      <div class="card">
        <div class="card-body">
          <h4 class="card-title">Food</h4>
          <div class="form-group mb-4">
                <select class="custom-select mr-sm-2" id="food" name="food">
                  <option selected>Please Select...</option>
                  <option >Veg</option>
                  <option>Non Veg</option>
                </select>
              </div>
        </div>
      </div>
    </div>
    <div class="col-sm-12 col-md-6 col-lg-4">
      <div class="card">
        <div class="card-body">
          <h4 class="card-title">Distance To College</h4>
            <div class="form-group">
```

```
                        <input type="text" name="distance" id="distance" class="form-
control" required>
                    </div>
                </div>
            </div>
            <div class="col-sm-12 col-md-6 col-lg-4">
                <div class="card">
                    <div class="card-body">
                        <h4 class="card-title">Photo</h4>
                            <div class="form-group">
                                <input type="file" name="photo" id="photo" class="form-control"
required>
                            </div>
                    </div>
                </div>
            </div>
            <div class="col-sm-12 col-md-6 col-lg-4">
                <div class="card">
                    <div class="card-body">
                        <h4 class="card-title">College Id</h4>
                            <div class="form-group">
                                <input type="file" name="idcard" id="idcard" class="form-control"
required>
                            </div>
                    </div>
                </div>
            </div>
            <div class="col-sm-12 col-md-6 col-lg-4">
                <div class="card">
                    <div class="card-body">
                        <h4 class="card-title">Caste Certificate</h4>
                            <div class="form-group">
                                <input type="file" name="cc" id="cc" class="form-control"
```

```
required>
                        </div>
                    </div>
                </div>
            </div>
              </div>
          <h4 class="card-title mt-5">Guardian's Information</h4>
            <div class="row">
              <div class="col-sm-12 col-md-6 col-lg-4">
                <div class="card">
                  <div class="card-body">
                    <h4 class="card-title">Guardian Name</h4>
                      <div class="form-group">
                        <input type="text" name="gname" id="gname" class="form-
control" placeholder="Enter Guardian's Name" required>
                      </div>
                  </div>
                </div>
              </div>
              <div class="col-sm-12 col-md-6 col-lg-4">
                <div class="card">
                  <div class="card-body">
                    <h4 class="card-title">Relation</h4>
                      <div class="form-group">
                        <input type="text" name="grelation" id="grelation" required
class="form-control" placeholder="Student's Relation with Guardian">
                      </div>
                  </div>
                </div>
              </div>
              <div class="col-sm-12 col-md-6 col-lg-4">
                <div class="card">
                  <div class="card-body">
                    <h4 class="card-title">Contact Number</h4>
```

```
                        <div class="form-group">
                            <input type="text" name="gcontact" id="gcontact" required class="form-control" placeholder="Enter Guardian's Contact No.">
                        </div>
                    </div>
                </div>
            </div>
            <h4 class="card-title mt-5">Current Address Information</h4>
            <div class="row">
                <div class="col-sm-12 col-md-6 col-lg-4">
                    <div class="card">
                        <div class="card-body">
                            <h4 class="card-title">Address</h4>
                            <div class="form-group">
                                <input type="text" name="address" id="address" class="form-control" placeholder="Enter Address" required>
                            </div>
                        </div>
                    </div>
                </div>
                <div class="col-sm-12 col-md-6 col-lg-4">
                    <div class="card">
                        <div class="card-body">
                            <h4 class="card-title">District</h4>
                            <div class="form-group">
                                <input type="text" name="district" id="district" class="form-control" placeholder="Enter City Name" required>
                            </div>
                        </div>
                    </div>
                </div>
                <div class="col-sm-12 col-md-6 col-lg-4">
                    <div class="card">
```

```
                    <div class="card-body">
                      <h4 class="card-title">Postal Code</h4>
                        <div class="form-group">
                          <input type="text" name="pincode" id="pincode" class="form-
control" placeholder="Enter Postal Code" required>
                        </div>
                    </div>
                  </div>
                </div>
                </div>
                <div class="col-sm-12 col-md-6 col-lg-12">
                <div class="column">
                  <div class="input-box">
                    <label><b>Declaration</b></label><br>
                    <span>i declare that i have read the rules of the hostel and do hereby promise
to abide by them and rules as may be promulgated from time to time.</span> 
                    <input type="checkbox" name="declare"  required/>
                  </div>
                </div>
                          <div class="form-actions">
                    <div class="text-center">
                      <button type="submit" name="submit" class="btn btn-
success">Submit</button>
                      <button type="reset" class="btn btn-dark">Reset</button>
                    </div>
                </div>
              </form>
            </div>
                 <?php include '../includes/footer.php' ?>
              </div>
        </div>
        <script src="../assets/libs/jquery/dist/jquery.min.js"></script>
      <script src="../assets/libs/popper.js/dist/umd/popper.min.js"></script>
```

```
    <script src="../assets/libs/bootstrap/dist/js/bootstrap.min.js"></script>
    <script src="../dist/js/app-style-switcher.js"></script>
    <script src="../dist/js/feather.min.js"></script>
    <script src="../assets/libs/perfect-scrollbar/dist/perfect-scrollbar.jquery.min.js"></script>

    <script src="../assets/extra-libs/c3/d3.min.js"></script>
    <script src="../assets/extra-libs/c3/c3.min.js"></script>
    <script src="../assets/libs/chartist/dist/chartist.min.js"></script>
    <script src="../assets/libs/chartist-plugin-tooltips/dist/chartist-plugin-
tooltip.min.js"></script>
    <script src="../dist/js/pages/dashboards/dashboard1.min.js"></script>
</body>
<script type="text/javascript">
    $(document).ready(function(){
        $('input[type="checkbox"]').click(function(){
            if($(this).prop("checked") == true){
                $('#paddress').val( $('#address').val() );
                $('#pcity').val( $('#city').val() );
                $('#ppincode').val( $('#pincode').val() );
            }

        });
    });
    </script>
    <script>
        function checkAvailability() {
        $("#loaderIcon").show();
        jQuery.ajax({
        url: "check-availability.php",
        data:'roomno='+$("#room").val(),
        type: "POST",
        success:function(data){
            $("#room-availability-status").html(data);
            $("#loaderIcon").hide();
```

```
        },
            error:function (){}
            });
        }
    </script>


    <script type="text/javascript">
    $(document).ready(function() {
        $('#duration').keyup(function(){
            var fetch_dbid = $(this).val();
            $.ajax({
            type:'POST',
            url :"ins-amt.php?action=userid",
            data :{userinfo:fetch_dbid},
            success:function(data){
            $('.result').val(data);
            }
            });
    })});
    </script>
</html>
```

room-details.php

```
<?php
    session_start();
    include('../includes/dbconn.php');
    include('../includes/check-login.php');
    check_login();
?>
<!DOCTYPE html>
<html dir="ltr" lang="en">
<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
```

```
    <!-- Tell the browser to be responsive to screen width -->
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta name="description" content="">
    <meta name="author" content="">

    <link href="../assets/extra-libs/c3/c3.min.css" rel="stylesheet">
    <link href="../assets/libs/chartist/dist/chartist.min.css" rel="stylesheet">
    <link href="../dist/css/style.min.css" rel="stylesheet">
</head>
<body>
    <div class="preloader">
        <div class="lds-ripple">
            <div class="lds-pos"></div>
            <div class="lds-pos"></div>
        </div>
    </div>
    <div id="main-wrapper" data-theme="light" data-layout="vertical" data-navbarbg="skin6"
data-sidebartype="full"
        data-sidebar-position="fixed" data-header-position="fixed" data-boxed-layout="full">
        <header class="topbar" data-navbarbg="skin6">
            <?php include '../includes/student-navigation.php'?>
        </header>
        <aside class="left-sidebar" data-sidebarbg="skin6">
            <div class="scroll-sidebar" data-sidebarbg="skin6">
                <?php include '../includes/student-sidebar.php'?>
            </div>
        </aside>
            <div class="container-fluid">
                <div class="col-7 align-self-center">
                        <h4 class="page-title text-truncate text-dark font-weight-medium mb-
1">Details About My Booked Room</h4>
                </div>
                 <div class="card">
                   <div class="card-body">
```

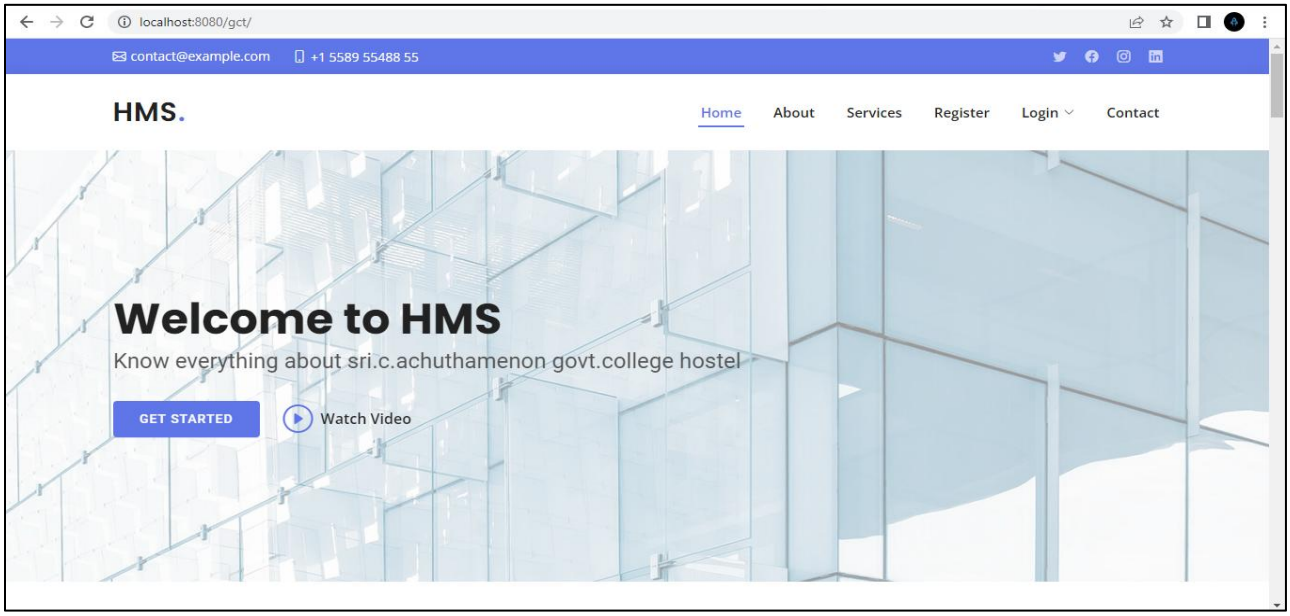
```
<div class="row">
<div class="table-responsive">
        <table id="zctb" class="table table-striped table-bordered no-wrap">
          <tbody>
          <?php
          $aid=$_SESSION['login'];
          $ret="SELECT * from registration where email=?";
          $stmt= $mysqli->prepare($ret) ;
          $stmt->bind_param('s',$aid);
          $stmt->execute() ;
          $res=$stmt->get_result();
          $cnt=1;
          while($row=$res->fetch_object())
            {
              ?>
            <tr>
              <td colspan="3"><b>Date & Time of Registration: </b><?php
echo $row->postingDate;?></td>

            </tr>
            <tr>
            <td><b>Room no :</b></td>
            <td><?php echo $row->roomno;?></td>
            <td><b>Starting Date :</b></td>
            <td><?php echo $row->stayfrom;?></td>
            <td><b>Bed no :</b></td>
            <td><?php echo $row->bedno;?></td>
            </tr>
            <tr>
            <td><b>Food Status:</b></td>
            <td><?php echo $row->food;?>
             </td>
            <td><b>Advance to pay :</b></td>
            <td><?php echo $row->atp;?></td>
```

```
</tr>
<tr>
<td colspan="6"><b>Total Fees</b></td>
</tr>


<tr>
<td><b>Admission Number :</b></td>
<td><?php echo $row->admno;?></td>
<td><b>Full Name :</b></td>
<td><?php echo $row->name;?></td>
<td><b>Email Address:</b></td>
<td><?php echo $row->email;?></td>
</tr>
<tr>
<td><b>Contact Number :</b></td>
<td><?php echo $row->contact;?></td>
<td><b>Gender :</b></td>
<td><?php echo $row->sex;?></td>
<td><b>Selected Course :</b></td>
<td><?php echo $row->course;?></td>
</tr>
<td><b>Guardian Name :</b></td>
<td><?php echo $row->gname;?></td>
<td><b>Guardian Relation :</b></td>
<td><?php echo $row->grelation;?></td>
</tr>
<td><b>Guardian Contact No. :</b></td>
<td colspan="6"><?php echo $row->gcontact;?></td>
</tr>
</td>
<td><b>Permanent Address:</b></td>
<td colspan="2"><?php echo $row->address;?>
<br />
```

```
                    </td>
                    </tr>



                    <?php
                    $cnt=$cnt+1;
                    } ?>

                  </tbody>
                </table>

              </div>
            </div>
          </div>
        </div>

  <script src="../assets/libs/jquery/dist/jquery.min.js"></script>
  <script src="../assets/libs/popper.js/dist/umd/popper.min.js"></script>
    <script src="../assets/extra-libs/c3/d3.min.js"></script>
  <script src="../assets/extra-libs/c3/c3.min.js"></script>
  <script src="../assets/libs/chartist/dist/chartist.min.js"></script>
  <script                    src="../assets/libs/chartist-plugin-tooltips/dist/chartist-plugin-
tooltip.min.js"></script>
  <script src="../dist/js/pages/dashboards/dashboard1.min.js"></script>
</body>

</html>
```

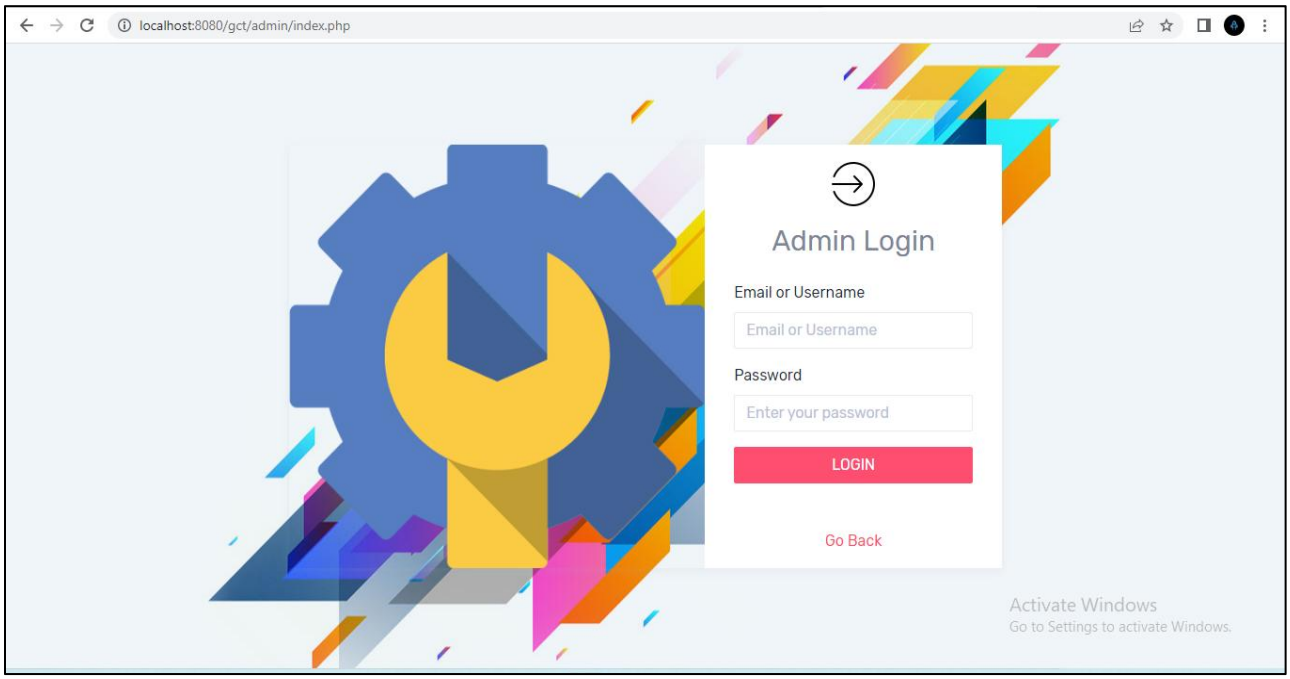
5.5 Screen Shots

5.5.1 Home page

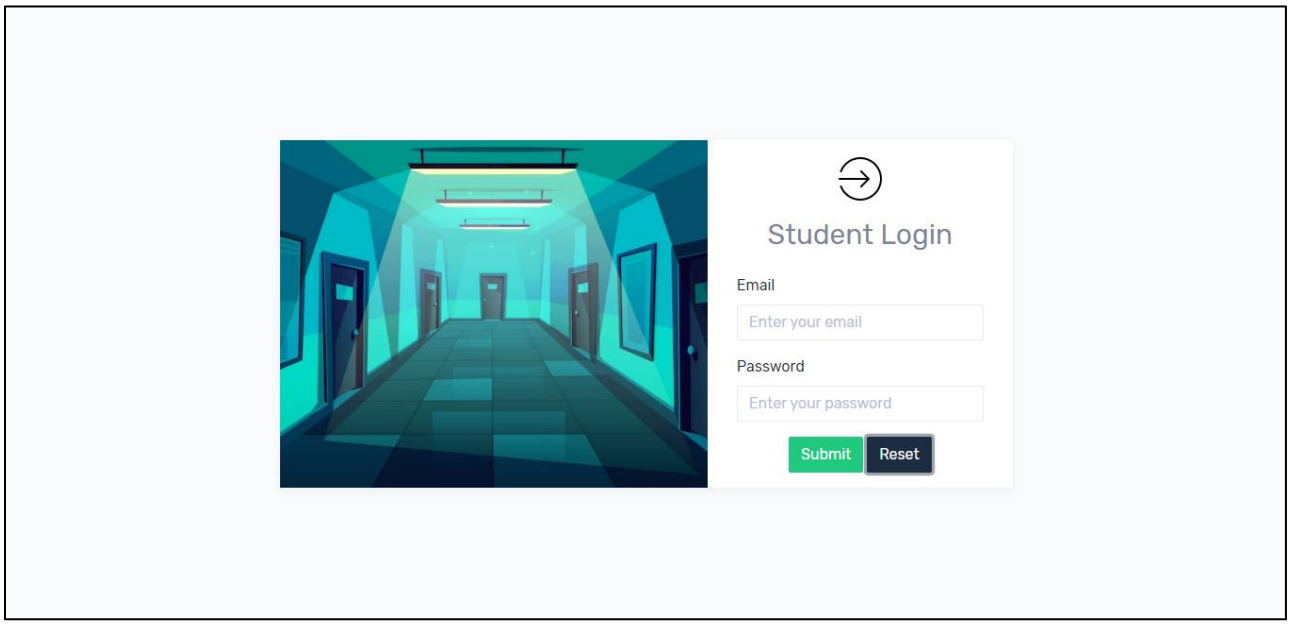


5.5.2 Login page

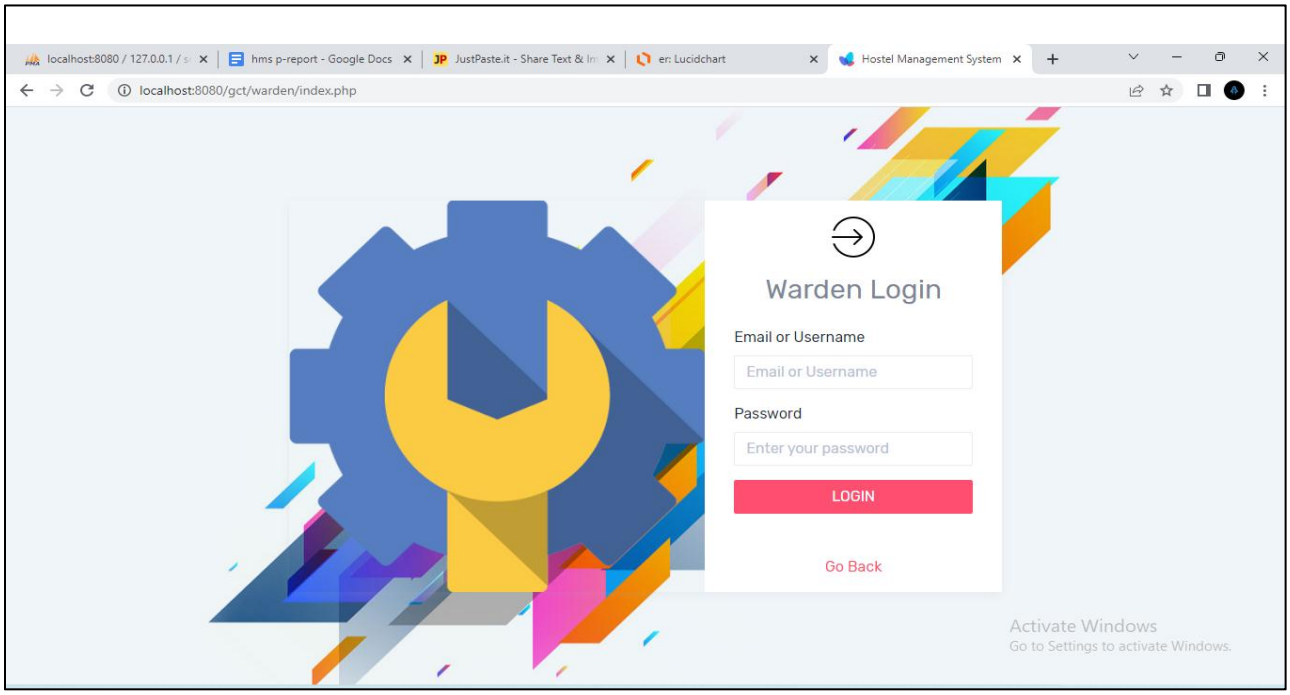
admin login



student login

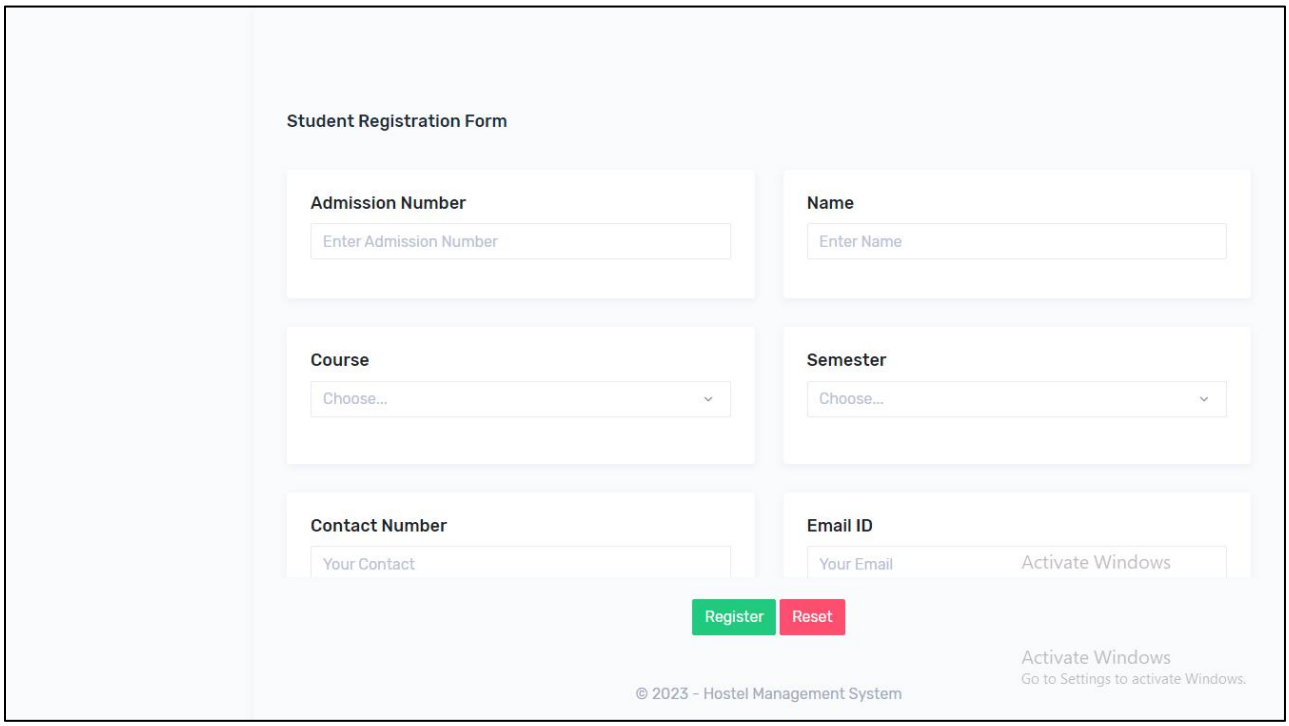


warden login

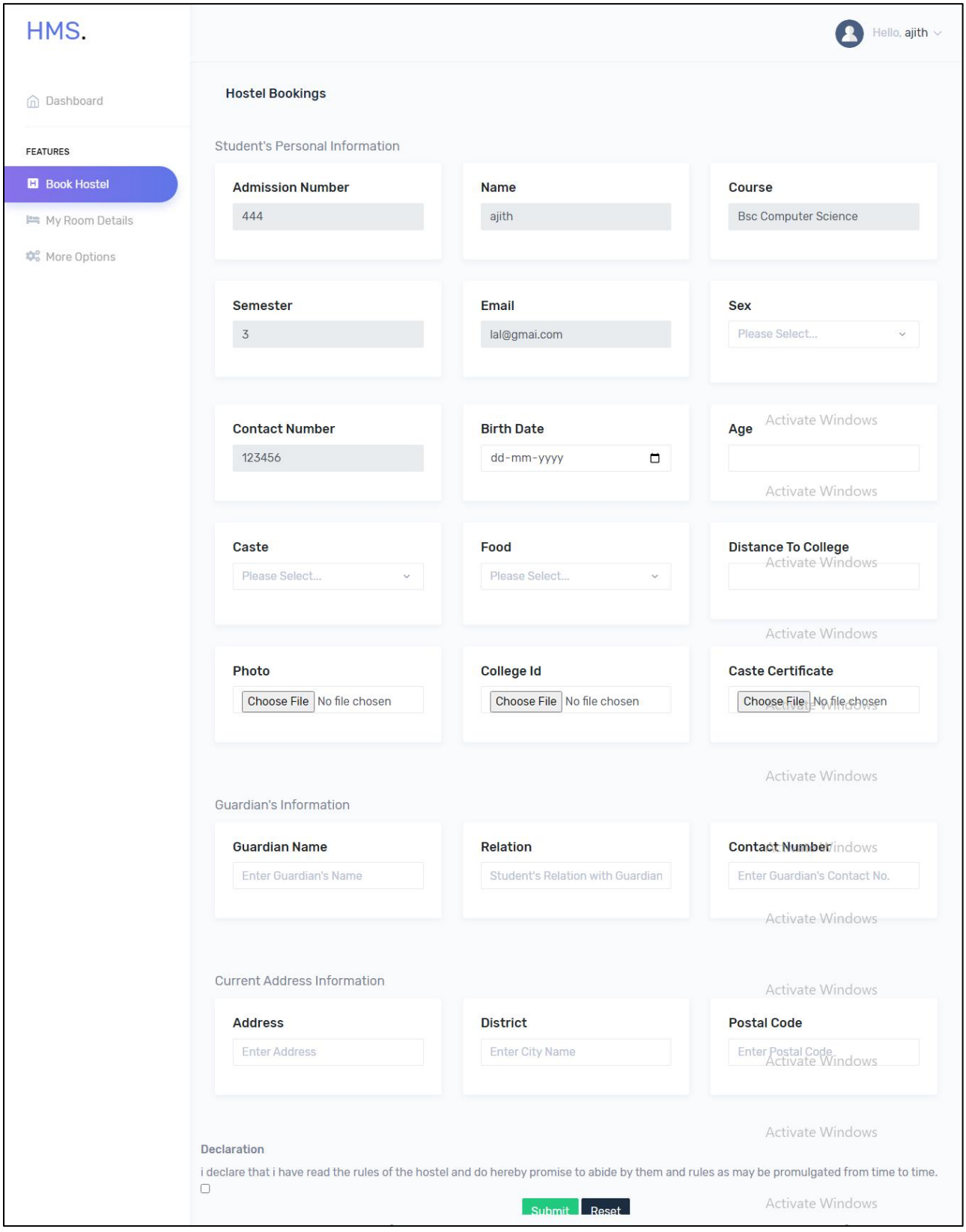


5.5.3 forms page

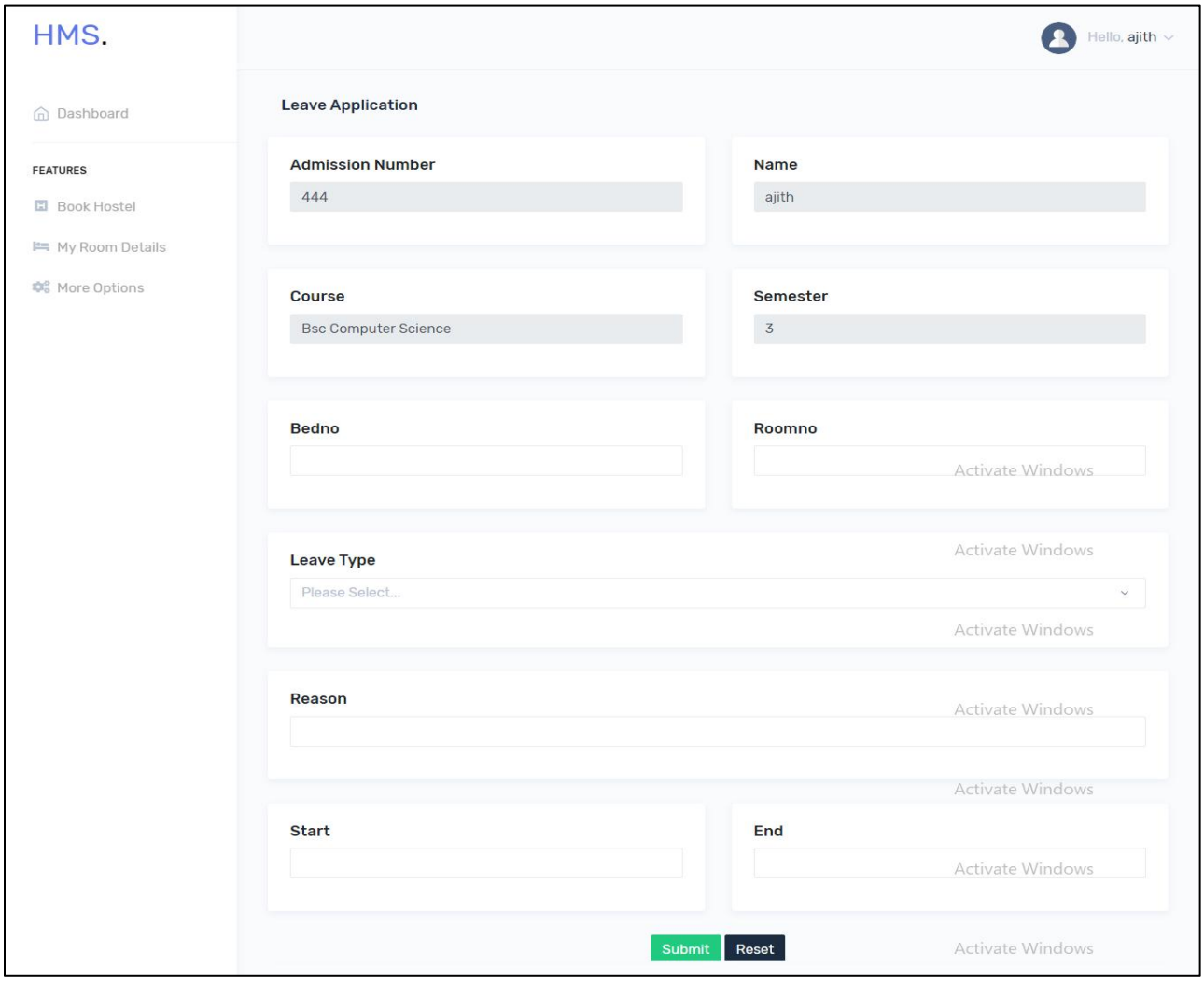
registration page



Book Hostel

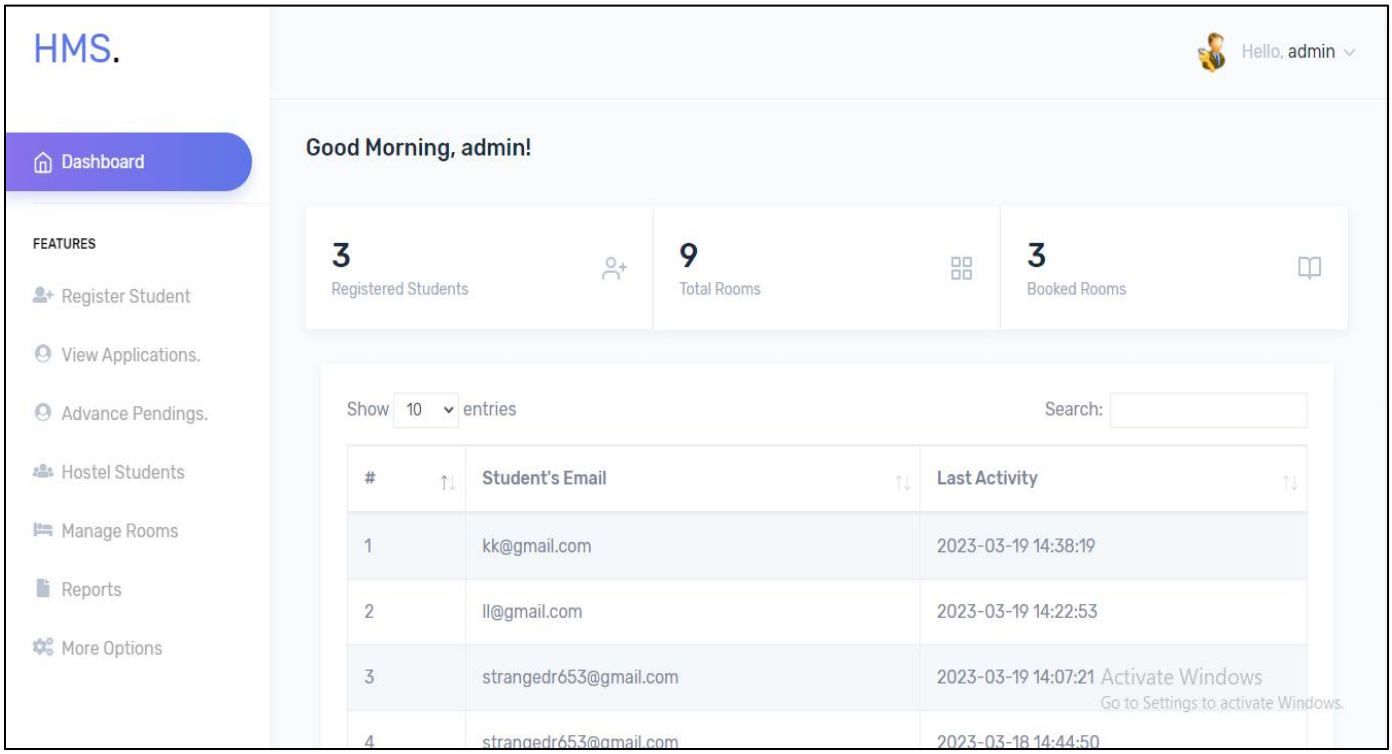

HMS.
Hello, ajith
Dashboard
FEATURES
Book Hostel
My Room Details
More Options
Hostel Bookings
Student's Personal Information
Admission Number
444
Name
ajith
Course
Bsc Computer Science
Semester
3
Email
lal@gmai.com
Sex
Please Select...
Contact Number
123456
Birth Date
dd-mm-yyyy
Age
Caste
Please Select...
Food
Please Select...
Distance To College
Photo
Choose File No file chosen
College Id
Choose File No file chosen
Caste Certificate
Choose File No file chosen
Guardian's Information
Guardian Name
Enter Guardian's Name
Relation
Student's Relation with Guardian
Contact Number
Enter Guardian's Contact No.
Current Address Information
Address
Enter Address
District
Enter City Name
Postal Code
Enter Postal Code
Declaration
i declare that i have read the rules of the hostel and do hereby promise to abide by them and rules as may be promulgated from time to time.
Submit
Reset

Leave application

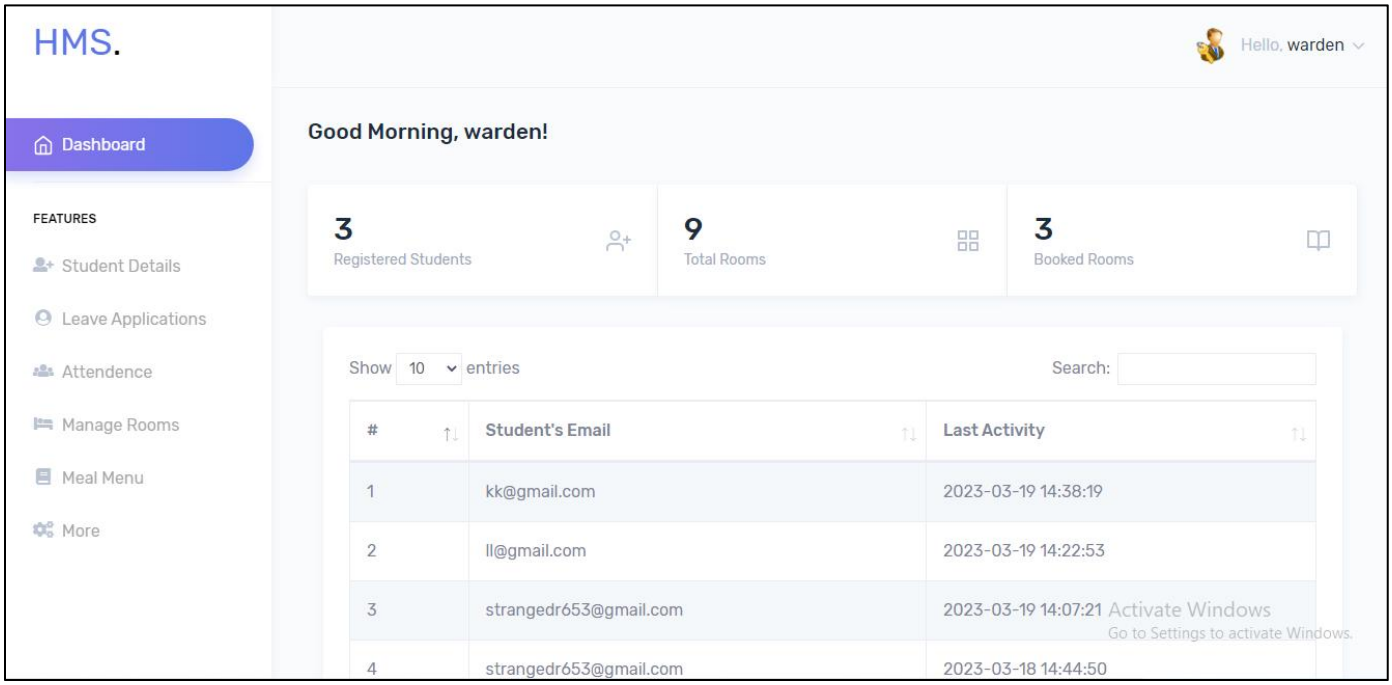


5.5.4 Dashboard page

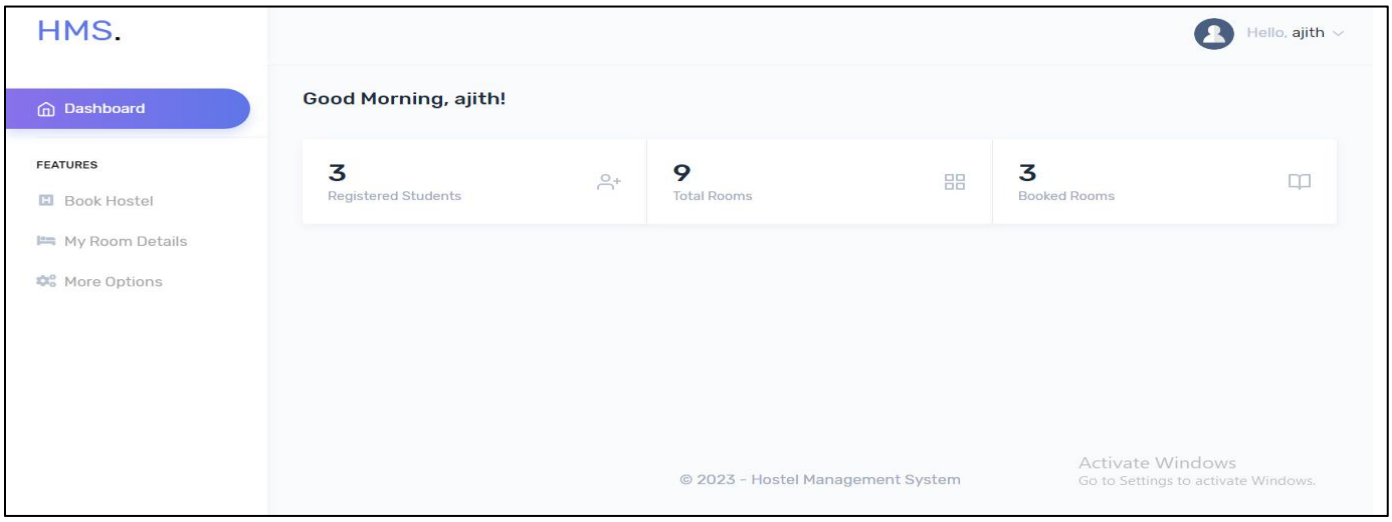
Admin



warden



Student


HMS.
Hello, ajith
Dashboard
Good Morning, ajith!
FEATURES
Book Hostel
My Room Details
More Options
3
Registered Students
9
Total Rooms
3
Booked Rooms
© 2023 - Hostel Management System
Activate Windows
Go to Settings to activate Windows.

Chapter 6

Testing

6.1 Introduction

System testing is a critical aspect of Software Quality Assurance and represents the ultimate review of specification, design, and coding. Testing is the process of executing a program with the intent of finding an error. A good test has a probability of finding an as-yet-undiscovered error. The purpose of testing is to identify and correct bugs in the developed system. Nothing is complete without testing. Testing is vital to the success of the system. The entire testing process can be divided into different phases:

• Unit testing

• Integration testing

6.2 Unit Testing

The first level of testing is unit testing. This test focuses on each module in the system. This is also known as “Module Testing”. To check whether each module in the software is proper, and it gives desired outputs to the given inputs. All validation and conditions are tested in the module in the first unit. The goal is to test the internal logic of the module. In unit testing, each unit is tested during the programming stage itself, in the proposed system we tested the following. In the proposed system, we tested the following: • Response of the system to inputs. • Animation of the selected item.

6.3 Integration Testing

The modules are integrated together to form a complete software package. The purpose of integration is to verify functional, performance, and reliability requirements placed on major design items. This testing is conducted based on modules. The integration testing is performed to detect design errors by focusing on testing the interconnection between modules. The objective is to take the unit-tested modules combined and tested as a whole. In the proposed system, we tested the following:

• Response of the system to inputs.

• Animation of the selected item.

• Linking between all the pages.

6.4 System testing

System testing is a type of software testing that is performed on a complete integrated system to evaluate the compliance of the system with the corresponding requirements. In system testing, integration testing passed components are taken as input. System testing detects defects within both the integrated units and the whole system. System testing the behaviour and design of the system and the expectations of the customer. system testing is black-box testing. System testing is performed on the entire system in the context of either functional requirement specification (FRS) or system requirement specification (SRS) or both. System testing is usually carried out by a team that is independent of the development team to measure the quality of the system unbiased. This are the list of types of system testing: -

• Usability testing
• Load testing
• Regression testing
• Recovery testing
• Functional testing
• Security testing
• Stress testing
• Performance testing
• Deployment testing
• Hardware/software testing

6.5 Test Plan & Test Cases

The primary objectives of test case design methods are to drive a set of tests that have the highest likelihood of uncovering the defects. To accomplish this objective, two categories of test case design techniques are used. Black box testing and white box testing.

Black box testing

Black-box testing is a method of software testing that examines the functionality of an application without peering into its internal structures or workings. This method of test can be applied virtually to every level of software testing such as unit testing, integration testing, system testing, and acceptance testing. It is sometimes referred to as specification-based testing. The tester is aware of what the software is supposed to do, but is not aware of how it does it.

White-box testing

White-box testing is a method of software testing that tests the internal structures or workings of an application, as opposed to its functionality. In white-box testing, an internal perspective of the system, as well as programming skills, are used to design test cases. The tester requires an in-depth knowledge of the source code being tested. The tester chooses inputs to exercise paths through the code and determines the expected outputs. This is analogous to testing nodes in a circuit. White-box testing can be applied at the unit, integration, and system levels of the software testing process.

Test cases

6.5.1 Login Screen

Sl No	Test Case	Expected result	Observed result	Pass/Fail
1	Without entering your username and password, press login button.	It should prompt a message “please fill info”	Message is prompted	Pass
2	If the email id and password do not match.	Alert users that“ Invalid email id and Password” and stay on same page.	Alert is prompted	Pass

6.5.2 Registration Screen

Sl No	Test Case	Expected result	Observed result	Pass/Fail
1	Without entering name, password and other fields and press the register button.	It should prompt a message “please fill info”	Message is prompted	Pass

6.5.3 Book_hostel Screen

Sl No	Test Case	Expected result	Observed result	Pass/Fail
1	Without entering name, admno & other fields and press register button.	It should prompt a message “please fill info”	Message is prompted	Pass
2	When already booked student Again, try to book a hostel	It should display page“you already booked hostel”	A page is displayed	Pass
3	When entering the book_hostel button	It should prompt a message “Hostel Requested”	Message is prompted	Pass

6.5.4 Allot_hostel Screen

Sl No	Test Case	Expected result	Observed result	Pass/Fail
1	Without entering room no, bed no& other fields and press the register button.	It should prompt a message “please fill info”	Message is prompted	Pass

2	When entering the allot button	It should prompt the message “success”	Message is prompted	Pass

6.5.5 Leave_apply Screen

Sl No	Test Case	Expected result	Observed result	Pass/Fail
1	Without entering remarks, approval & other fields, pressed the register button.	It should prompt a message “please fill info”	Message is prompted	Pass
2	When entering the update button	It should prompt the message “leave Request verified”	Message is prompted	Pass

6.5.6 Complaint Screen

Sl No	Test Case	Expected result	Observed result	Pass/Fail
1	Without entering a complaint and pressing the register button.	It should prompt a message “please fill info”	Message is prompted	Pass
3	When entering the register button	It should prompt a message “complaint Registered”	Message is prompted	Pass

Chapter 7

Conclusion

In conclusion, the proposed Hostel Management System aims to improve the efficiency and effectiveness of managing student accommodation facilities by leveraging modern technologies. The system will reduce the time and effort required to manage student accommodation facilities, improve the accuracy and completeness of student records, attendance, leave applications, and meal plans, provide timely and accurate information to students, hostel wardens, and administrators, increase transparency and accountability in the hostel management process, and enhance the overall student experience and satisfaction.

Through the analysis of the existing system, its limitations were identified and taken into consideration while designing the proposed system. The requirement analysis provided a comprehensive list of features to be included in the system, which were further specified in the admin, warden, and student modules. The DFD level 0 ,1 and 2 diagrams illustrated the flow of data in the system, while the structure of the database and table design ensured the efficient storage and retrieval of data.

Overall, the proposed Hostel Management System provides an innovative and modern solution to the challenges faced by hostel management in institutions of higher learning. The successful implementation of the system will not only benefit the hostel management, but also improve the overall experience of students residing in the hostel.

Future Work

In the future, the proposed Hostel Management System could be extended and improved in several ways. One potential area for future work is to incorporate more advanced analytic s and reporting capabilities to provide insights and trends to hostel administrators, wardens, and students. This could include data visualization tools, predictive analytic, and machine learning algorithms to identify patterns and predict future demand for hostel rooms, meals, and other facilities.

Another area for improvement could be to integrate the system with other student management systems, such as course registration, student records, and financial aid. This would provide a more seamless experience for students and administrators and reduce the need for manual data entry and reconciliation.Furthermore, the system could be made more

mobile-friendly, with dedicated mobile applications for students and wardens. This would enable students to access their hostel information and services on-the-go, and wardens to manage hostel activities from their mobile devices.

Finally, the system could be scaled up to support multiple hostels and campuses, with a centralized administration and reporting system. This would enable administrators to manage multiple hostels and campuses from a single system, and provide a more comprehensive view of the student accommodation facilities across the organization.

References

1. James R Groff and Paul N Weinberg, “Complete reference SQL”. Second Edition
2. Agnes Molnar and Adam Granicz, “Net with Visual C# 2010”.
3. CSS files, from W3 schools. http://www. School.com/css file
4. PHP 5 Tutorial, Available at http://www.w3schools.com/PHP/
5. CSS files, from Wikipedia, the free encyclopedia.
   http://www.en.wikipedia.org/wiki/Wireless_security.
6. Jamil, T; Dept. of Electro. & Computer Eng., Sultan Qaboos University, Al Khod, Oman, Automatic attendance recording system using mobile telephone, Telecommunications Forum
   (TELFOR), 2011 19th 1297 – 1299