

GOVERNMENT COLLEGE OF ENGINEERING AND CERAMIC TECHNOLOGY



Project Report

On

HOSTEL MANAGEMENT

SYSTEM

Submitted by

SUBHAJIT BOKSHI (*GCECTB-R18-3031*)
OGNEEV BHADRA (*GCECTB-R18-3017*)
PUSPITA POLLEY (*GCECTB-L19-3005*)
TARUN ROUT (*GCECTB-R17-3038*)

DEBSMITA BANERJEE (*GCECTB-R18-3008*)
SOUHARDYA MANDAL (*GCECTB-R18-3038*)
JACQUES NARJINARY (*GCECTB-R18-3012*)

Under the Esteemed Guidance of
DR. KALPANA SAHA
Asst. Professor & Head of the Department
Government College of Engineering and Ceramic Technology

ACKNOWLEDGEMENT

The project work in this report is an outcome of continual work and intellectual support from various sources. It is almost impossible to express adequately the debts owed to many persons who have been instrumental in imparting this work a successful status. It is however a matter of great pleasure to express my gratitude and appreciation to all those people who had helped in completion of this project.

*We would like to take this opportunity to thank **Dr. Kalpana Saha** ma'am for giving us an opportunity to work on this project, which not only has increased our awareness but also has taught the importance of teamwork, it is because of her guidance, constant encouragement and inspiration that we have been able to accomplish the task of completing this project.*

We express sincere thanks to our project mentor for her invaluable guidance and frequent suggestions during the course of the project. Their suggestions helped us to maintain a good quality of work. We express our deep gratitude to her.

Souhardya Mandal (GCECTB-R18-3038)
Debsmita Banerjee (GCECTB-R18-3008)
Ogneev Bhadra (GCECTB-R18-3017)
Subhajit Bokshi (GCECTB-R18-3031)
Puspita Polley (GCECTB-L19-3005)
Tarun Rout (GCECTB-R17-3038)
Jacques Narjinary (GCECTB-R18-3012)

➤ TABLE OF CONTENTS

1. INTRODUCTION

1.1. Purpose

1.2. Observation

1.3. Scope

1.4. Overview

2. SYSTEM REQUIREMENTS SPECIFICATION

2.1. Methodology Development Model

2.2. Hardware configuration

2.3. Software configuration

2.4. Tools and techniques used

2.5. Feasibility study

3. SYSTEM DESIGN

3.1. Process design

3.2. Database design

4. SYSTEM IMPLEMENTATION & SCREENSHOTS

4.1. Implementation

4.2. Screenshots

5. SYSTEM TESTING

5.1. Testing procedures

5.2. Objectives of system testing

5.3. Test cases

6. CONCLUSION

INTRODUCTION

Hostel Management System is a Software development for managing various activities of the hostel. The Software help user in case of Hostel Information, registration, room and Searching Hostel Account. Aim of the Hostel Management System is to manage the records and make user friendly software. It helps admin to manage student records, Staff Records and many more. Due to the increase in educational institution Hostel are also increasing for the accommodation.

Hostel management System is online application developed for managing various activities of the hostel. Deals with the problem on managing the hostel and avoided the problem which occur when carried out manually.

1.1. PURPOSE

The purpose is to make an automated system to carry out the various operation of a Hostel. The system will provide the ease of use to the staff of the hostel by performing all the work on a computer system rather than following a paper pen approach. This approach helps improving the reliability of the data maintained and provides a fast and efficient interface for the users of the software.

1.2. OBJECTIVE

- Provide quick and efficient means for gathering the student information along with their rooms, course, contact etc.
- Maintaining Student's Records.
- Secure all the data of the Hosteller

1.3. SCOPE

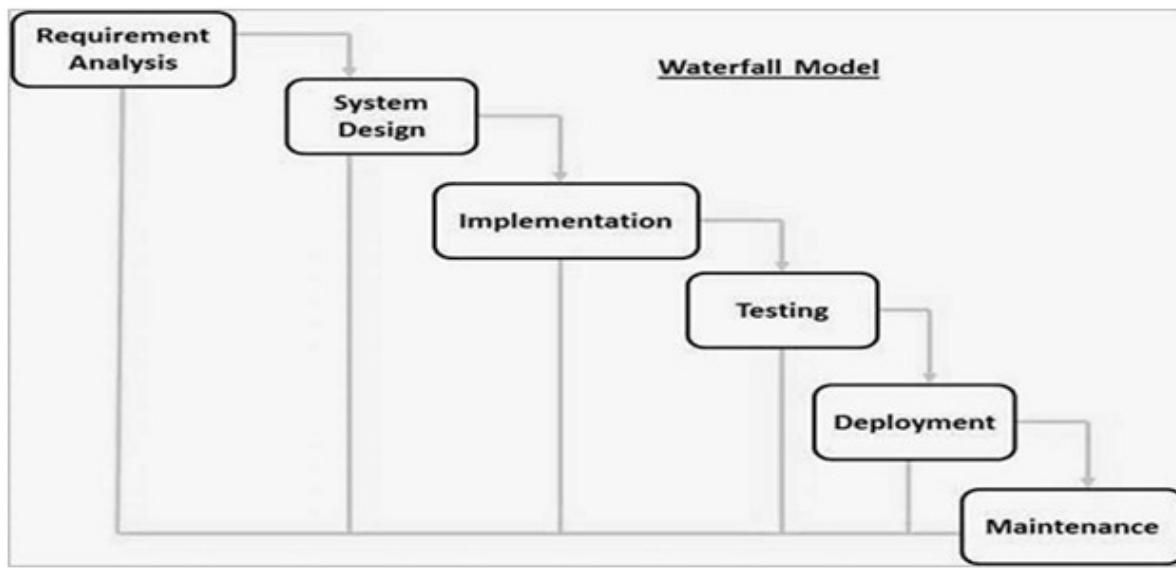
The software product “Hostel Management System” will be an application that will be used for maintaining the records in an organized manner and to replace old paper work system. This project aims at automating the hostel management for smooth working of the hostel by automating almost all the activities. Updations and modifications will be easily achievable and all the calculations and accounting work would be accurate.

1.4 OVERVIEW OF THE PROJECT

Hostel Room Allocation System is a web application which aims at computerization of the current procedure of allocating hostel rooms. Currently, the process involves students filling up the forms and submitting them in respective hostel offices which involves a lot of paperwork, hence less efficient. The aim of the Hostel Management System is to carry out the activities of Hostel in an efficient way. It will take the operations of Hostel to an upper level by providing faster access to data and allowing addition, upgradation, modification, and deletion of data in a very systematic and reliable manner.

SYSTEM REQUIREMENTS SPECIFICATION

2.1. METHODOLOGY DEVELOPMENT MODEL:



The sequential phases in Waterfall model are –

- **Requirement Gathering and analysis** – All possible requirements of the system to be developed are captured in this phase and documented in a requirement specification document
- **System Design** – The requirement specifications from first phase are studied in this phase and the system design is prepared. This system design helps in specifying hardware and system requirements and helps in defining the overall system architecture.
- **Implementation** – With inputs from the system design, the system is first developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality, which is referred to as Unit Testing.
- **Integration and Testing** – All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the

entire system is tested for any faults and failures.

- **Deployment of system** – Once the functional and non-functional testing is done; the product is deployed in the customer environment or released into the market.
- **Maintenance** – There are some issues which come up in the client environment. To fix those issues, patches are released. Also to enhance the product some better versions are released. Maintenance is done to deliver these changes in the customer environment.

2.2. HARDWARE CONFIGURATIONS:

1. Minimum 128 MB RAM
2. Pentium III or above processor
3. Minimum 512 MB of Hard Disk space

2.3. SOFTWARE CONFIGURATIONS:

This system will work under Windows 98 or above OS

2.4. TOOLS & TECHNIQUES USED:

1. FLASK
2. HTML
3. CSS
4. JAVASCRIPT
5. BOOTSTRAP
6. SQLAlchemy
7. RAZORPAY
8. VISUAL STUDIO CODE

2.5. FEASIBILITY STUDY:

After doing the project Hostel Management System, study and analyzing all the existing or required functionalities of the system, the next task is to do the feasibility study for the project. All projects are feasible given unlimited resources and infinite time.

Feasibility study includes consideration of all the possible ways to provide a solution to the given problem. The proposed solution should satisfy all the user requirements and should be flexible enough so that future changes can be easily done based on the future upcoming requirements.

A. Economical Feasibility

This is a very important aspect to be considered while developing a project. We decided the technology based on minimum possible cost factor.

- All hardware and software cost has to be borne by the organization.
- Overall we have estimated that the benefits the organization is going to receive from the proposed system will surely overcome the initial costs and the later on running cost for system.

B. Technical Feasibility

This included the study of function, performance and constraints that may affect the ability to achieve an acceptable system. For this feasibility study, we studied complete functionality to be provided in the system, as described in the System Requirement Specification (SRS), and checked if everything was possible using different type of frontend and backend platforms.

C. Operational Feasibility

No doubt the proposed system is fully GUI based that is very user friendly and all inputs to be taken are self-explanatory even to a layman. Besides, a proper training has been conducted to let know the essence of the system to the users so that they feel comfortable with new system. As far our study is concerned the clients are comfortable and happy as the system has cut down their loads and doing,

SYSTEM DESIGN

3.1. PROCESS DESIGN

Process design plays an important role in project development. In order to understand the working procedure, process design is necessary. Data Flow Diagram and Use case diagrams are the often used for process design.

A **use case diagram** is a graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of users the system has and will often be accompanied by other types of diagrams as well. The use cases are represented by either circles or ellipses. The actors are often shown as stick figures.

Data Flow Diagram is the logical representation of the data flow of the project. The DFD is drawn using various symbols. It has a source and a destination. The process is represented using circles and source and destination are represented using squares. The data flow is represented using arrows. One reader can easily get the idea about the project through Data Flow diagrams.

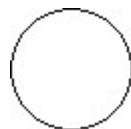
SYMBOLS USED IN DATA FLOW DIAGRAM



- *Source rectangle, which defines or destination*



- *Arrow, which shows dataflow.*



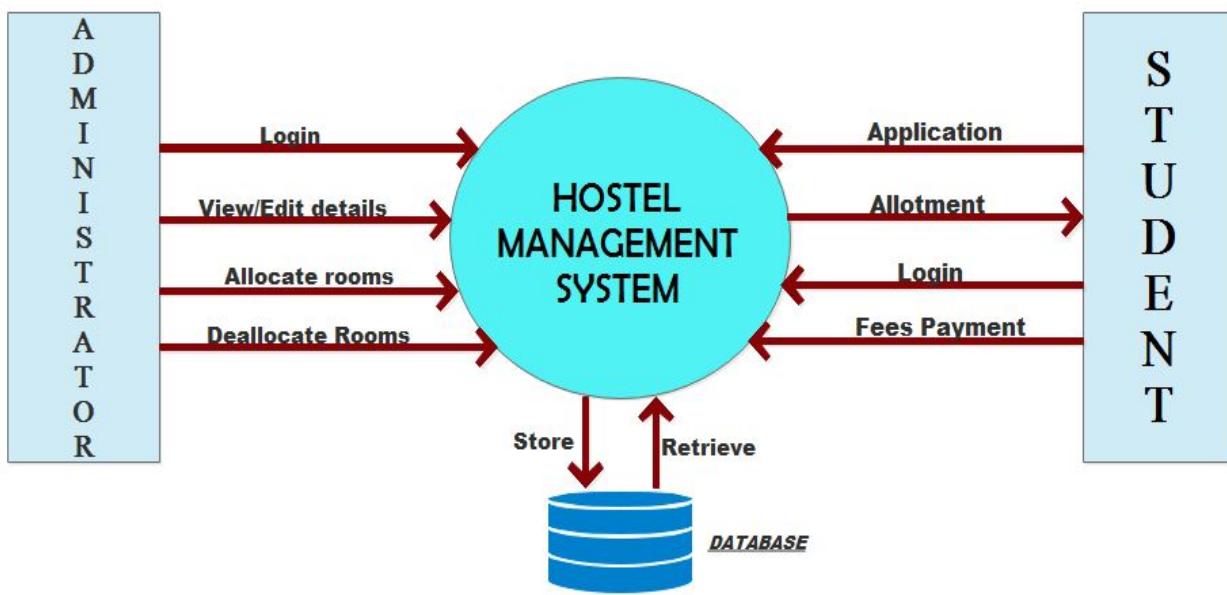
- *Circle, which represents a process that transforms incoming data into outgoing flow.*



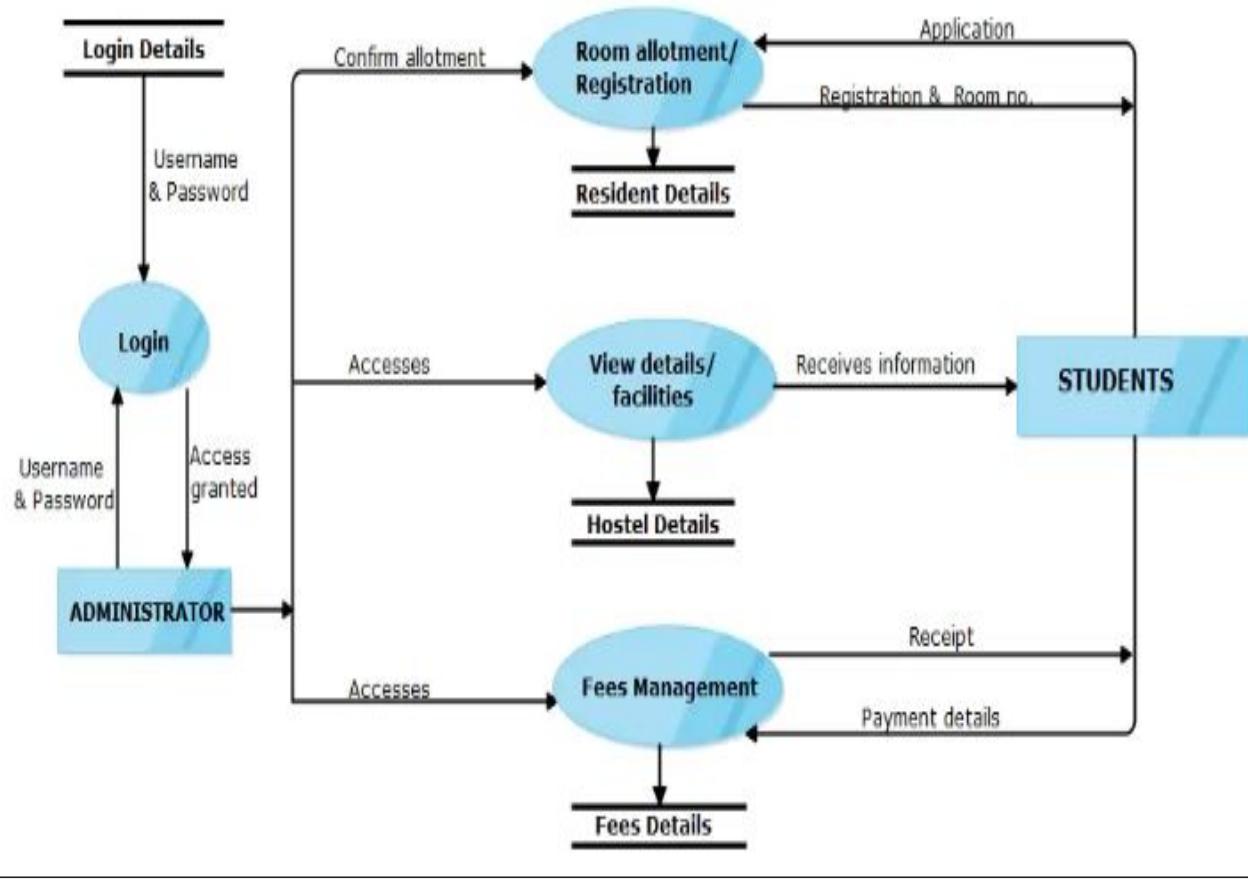
- *Open rectangle, which shows a data store.*

3.2.1 Data Flow Diagram

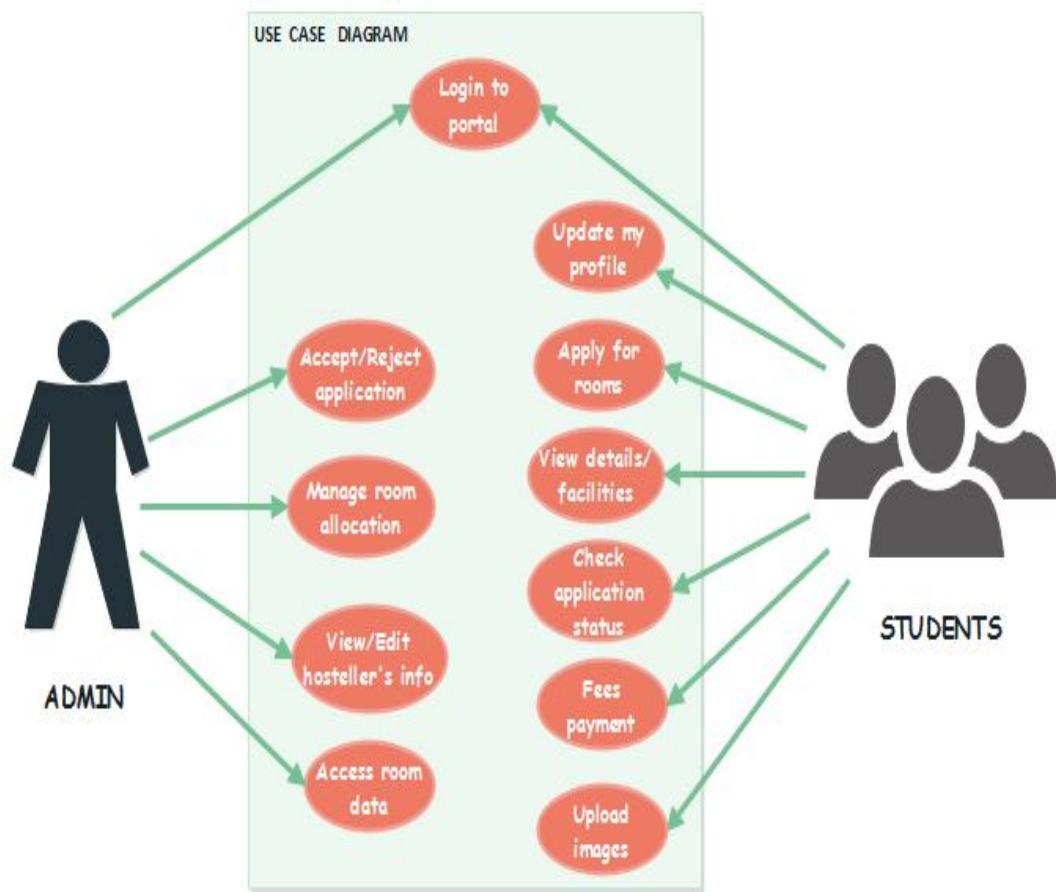
Context level DFD-



Level-1 DFD-



3.2.2 Use Case Diagram



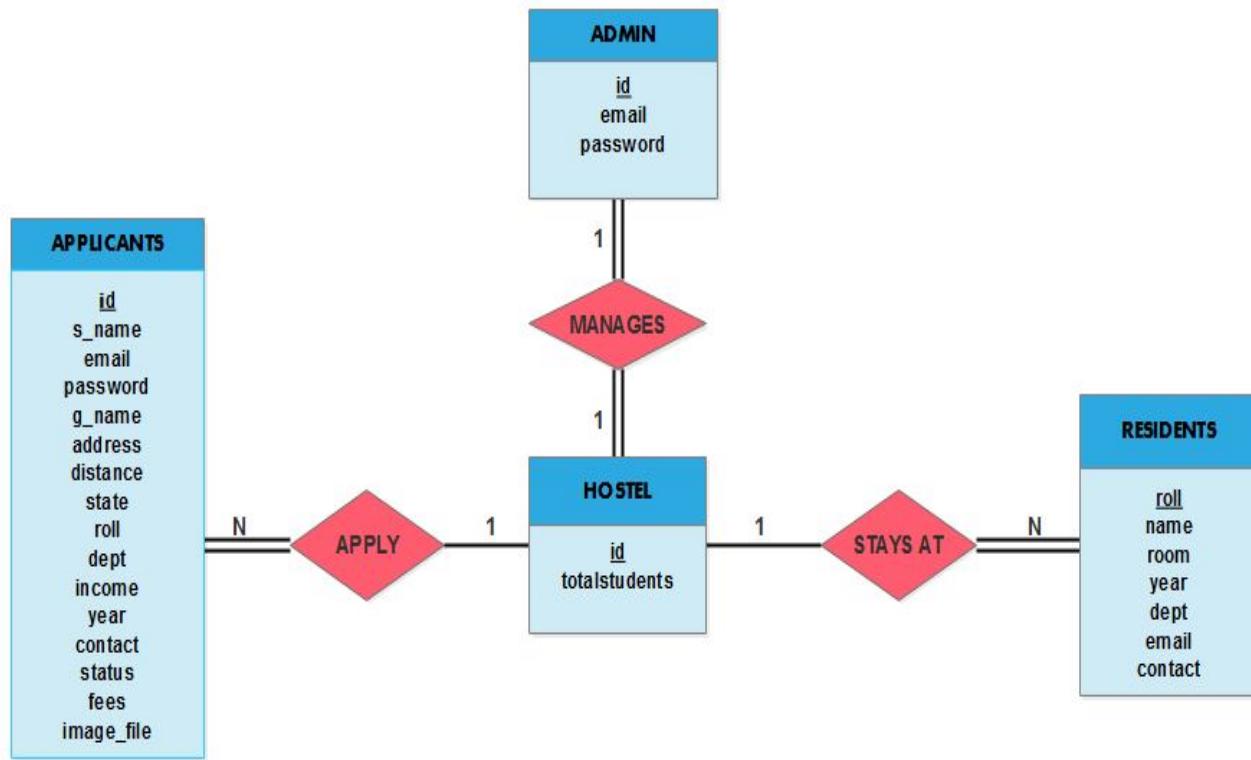
3.2. DATABASE DESIGN

The data in the system has to be stored and retrieved from database. Designing the database is part of system design. Data elements and data structures to be stored have been identified at analysis stage. They are structured and put together to design the data storage and retrieval system.

A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently. The general objective is to make database access easy, quick, inexpensive and flexible for the user. Relationships are established between the data items and unnecessary data items are removed. Normalization is done to get an internal consistency of data and to have minimum redundancy and maximum stability. This ensures minimizing data storage required, minimizing chances of data inconsistencies and optimizing for updates. The **SQLAlchemy** has been chosen for developing the relevant databases.

. An entity relationship diagram (ERD) shows the relationships of entity sets stored in a database. An entity in this context is an object, a component of data. An entity set is a collection of similar entities. These entities can have attributes that define its properties. By defining the entities, their attributes, and showing the relationships between them, an ER diagram illustrates the logical structure of databases

3.2.3 Entity-Relationship Diagram



The following are the tables that are involved in this system

1. ADMIN

FIELD NAME	DATA TYPE	DESCRIPTION
<u>id</u>	<i>Integer</i>	<i>Id of administrator</i>
<i>email</i>	<i>String</i>	<i>Email id of administrator</i>
<i>password</i>	<i>String</i>	<i>Password of administrator</i>

2. APPLICANTS

FIELD NAME	DATA TYPE	DESCRIPTION
<i>id</i>	<i>integer</i>	<i>Id of applicant</i>
<i>S_name</i>	<i>String</i>	<i>Name of applicant</i>
<i>Email</i>	<i>String</i>	<i>Email id of applicant</i>
<i>Password</i>	<i>String</i>	<i>Password of applicant</i>
<i>G_name</i>	<i>String</i>	<i>Guardian name</i>
<i>Address</i>	<i>string</i>	<i>Address of applicant</i>
<i>Distance</i>	<i>integer</i>	<i>Distance from home</i>
<i>State</i>	<i>String</i>	<i>State living in</i>
<i>Roll</i>	<i>String</i>	<i>Roll no. of the applicant</i>
<i>Dept</i>	<i>String</i>	<i>Department of the applicant</i>
<i>Income</i>	<i>integer</i>	<i>Family income</i>
<i>Year</i>	<i>String</i>	<i>Year of the course</i>
<i>Contact</i>	<i>String</i>	<i>Contact of the applicant</i>
<i>Status</i>	<i>String</i>	<i>Application form status</i>
<i>Fees_status</i>	<i>String</i>	<i>Fees payment status</i>
<i>Image_file</i>	<i>String</i>	<i>Photo of the applicant</i>

3. HOSTEL

FIELD NAME	DATA TYPE	DESCRIPTION
<i>id</i>	<i>integer</i>	<i>Hostel id</i>
<i>Total_students</i>	<i>integer</i>	<i>Total no. of students</i>

4. RESIDENTS

FIELD NAME	DATA TYPE	DESCRIPTION
<i>Roll</i>	<i>String</i>	<i>Roll no. of the student</i>
<i>Name</i>	<i>String</i>	<i>Name of the student</i>
<i>Room</i>	<i>Integer</i>	<i>Room no. of the student</i>
<i>Year</i>	<i>Integer</i>	<i>Couse year of the student</i>
<i>Dept</i>	<i>String</i>	<i>Department of the student</i>
<i>Email</i>	<i>String</i>	<i>Email id of the student</i>
<i>Contact</i>	<i>String</i>	<i>Contact details of the student</i>

SYSTEM IMPLEMENTATION & SCREENSHOTS

4.1. IMPLEMENTATION

For the past few years the number of educational institutions has been increasing rapidly. Thereby the number of hostels are also increasing for the accommodation of the students studying in this institution. And hence there is a lot of strain on the person who is running the hostel and software's are not usually used in this context. This particular project deals with the problems on managing a hostel and avoids the problems which occur when carried manually.

Identification of the drawbacks of the existing system leads to the designing of a computerized system that will be compatible with the existing system with the system which is more user friendly and more GUI oriented. We can improve the efficiency of the system, thus overcoming the following drawbacks of the existing system.

- more human error
- more strength and strain of manual labour needed
- Repetition of the same procedures
- low security
- Data redundancy
- difficult to handle
- difficult to update data
- record keeping is difficult
- Backup data can be easily generated

4.2. SCREENSHOTS

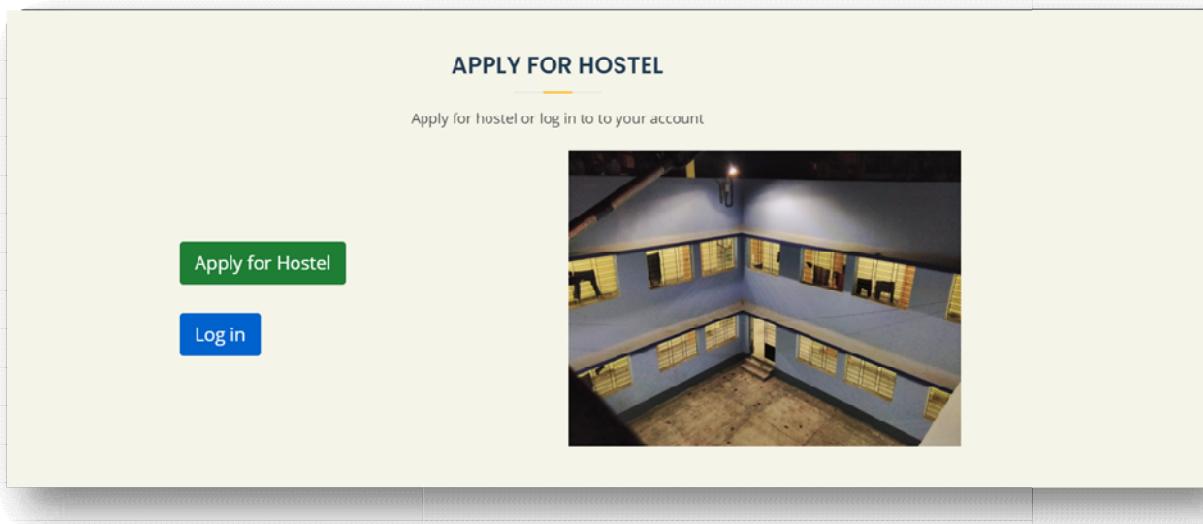
4.1.1 - Homepage

This is the **Home** Section of the Homepage. This section contains a navigation bar containing the links of different sections of the page, the gallery section, main college website and the links of login page. This section also contains a button containing the link to the application page.

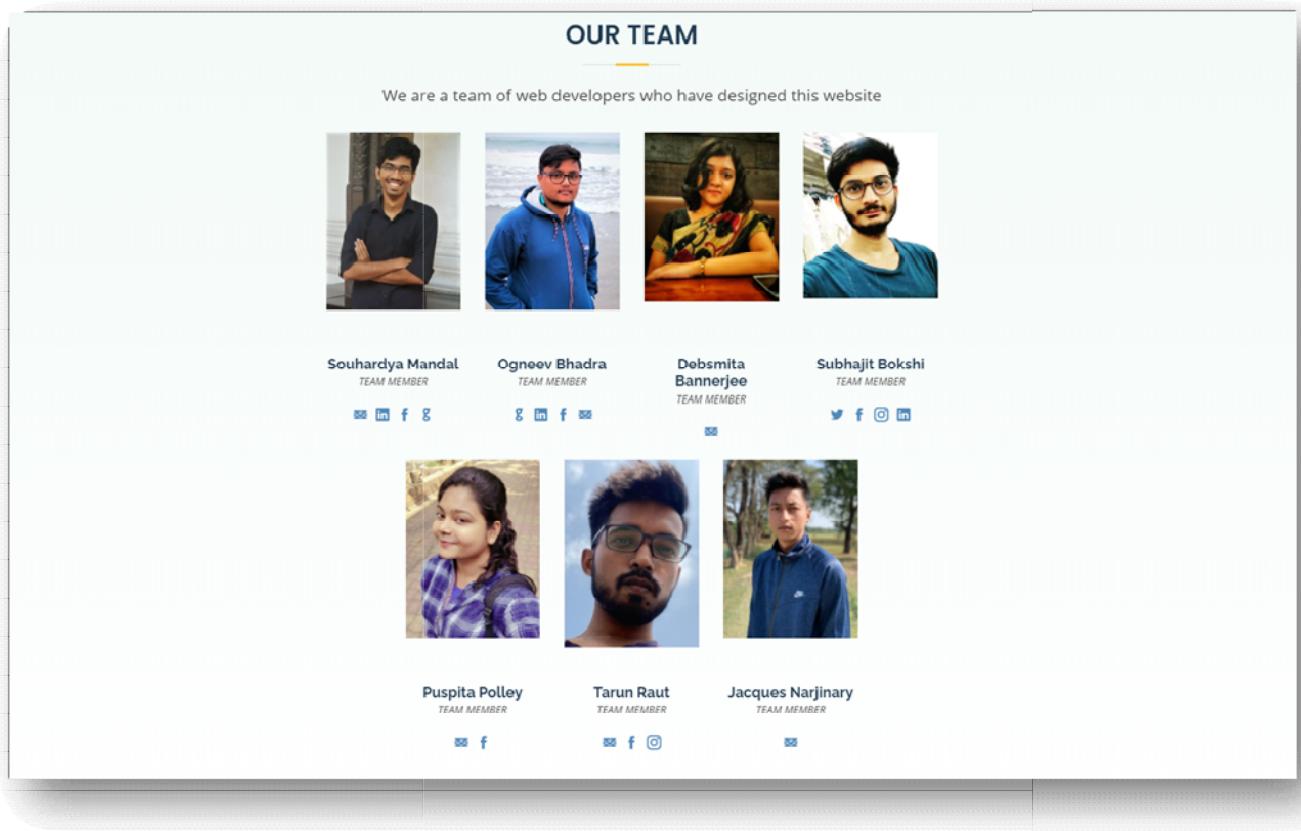


This is the **About** Section of the Homepage. This section contains four parts : **Number of students, Coordinator's Name, Fees and Number of Students**. When we place our mouse pointer on these words, the information is shown in the popover. This information is fetched from the backend.

This is the **Apply** section of the Homepage. This section contains two buttons : **Apply for Hostel** and **Log In** button.



This is the **Team** section of the homepage which contains the details about the developers of this website. Every block contains the name, image and the social links of each developer.



This is the **Contact** section of the Homepage. This section contains two parts, an **Address** part and a **Contact form**. The address part contains the full written address and also a map where the location of the college is marked. We can also zoom in or zoom out to navigate to nearby places. This map is an API of the Google map.

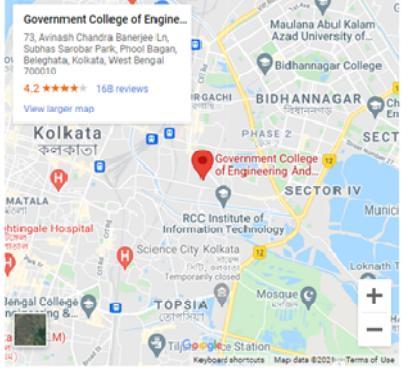
CONTACT

Contact Us and give your valuable feedback.

Address

Government College of Engineering and Ceramic Technology, Kolkata
73, Abinash Chandra Banerjee Lane, Kolkata-700 010
73, Abinash Chandra Banerjee Ln,
Suhash Sarobar Park, Phool Bagh,
Beliaghata, Kolkata, West Bengal
700010

4.2 ★★★★☆ 168 reviews
[View larger map](#)



The map displays the college's location at 73, Abinash Chandra Banerjee Lane, Kolkata-700 010. It shows the surrounding area including Bidhannagar College, Maulana Abul Kalam Azad University of Technology, RCC Institute of Information Technology, Science City Kolkata, and various local landmarks like IITB, IITK, and IITM. The map includes a zoom control and a legend for keyboard shortcuts.

Your Name Your Email
 Subject
 Message

 Email Us
principal@gcct.ac.in
gcctwb@gmail.com

 Call Us
033 2370 1264
033 2363 3674

4.1.2 - Gallery Section



This is a separate webpage that is a gallery section of the hostel. This section contains several images of the college hostel. When we click on any image, it shows the image on fullscreen, and we can navigate to the next and previous images using the arrows on the left and right side respectively.



4.1.3 - Application Form

This is the main application form. A new applicant who wants accommodation in the college hostel has to fill up this form. The applicant has to fill up all the fields. The fields are :

- First Name
- Last Name
- Email
- Password (contains minimum 4 characters)
- Guardian's First name
- Guardian's Last name
- Address
- Distance from college (in km)
- State
- Year
- College Roll Number (in specified format)
- Branch
- Family Income
- Contact Number (in specified format)

The screenshot shows a web-based application for hostel booking. The background image is a photograph of a large, light blue, multi-story building, likely a college or university building, with several windows and a balcony. In the foreground, a yellow rectangular overlay contains the application form.

Apply for college hostel

Form fields:

- First Name: Enter First Name Here..
- Last Name: Enter Last Name Here..
- Email: Email
- Password: Password (4-8 characters)
- Guardian's First Name: Enter Guardian's First Name..
- Guardian's Last Name: Enter Guardian's Last Name..
- Address: Enter Address Here..
- Distance from college: Distance from college (in km) Choose... 1st
- State: Choose... 1st
- Year: Choose... 1st
- College Roll Number: GCECTB-RXX-XXXX
- Branch: Choose... Annual Income
- Contact number: +919876543210

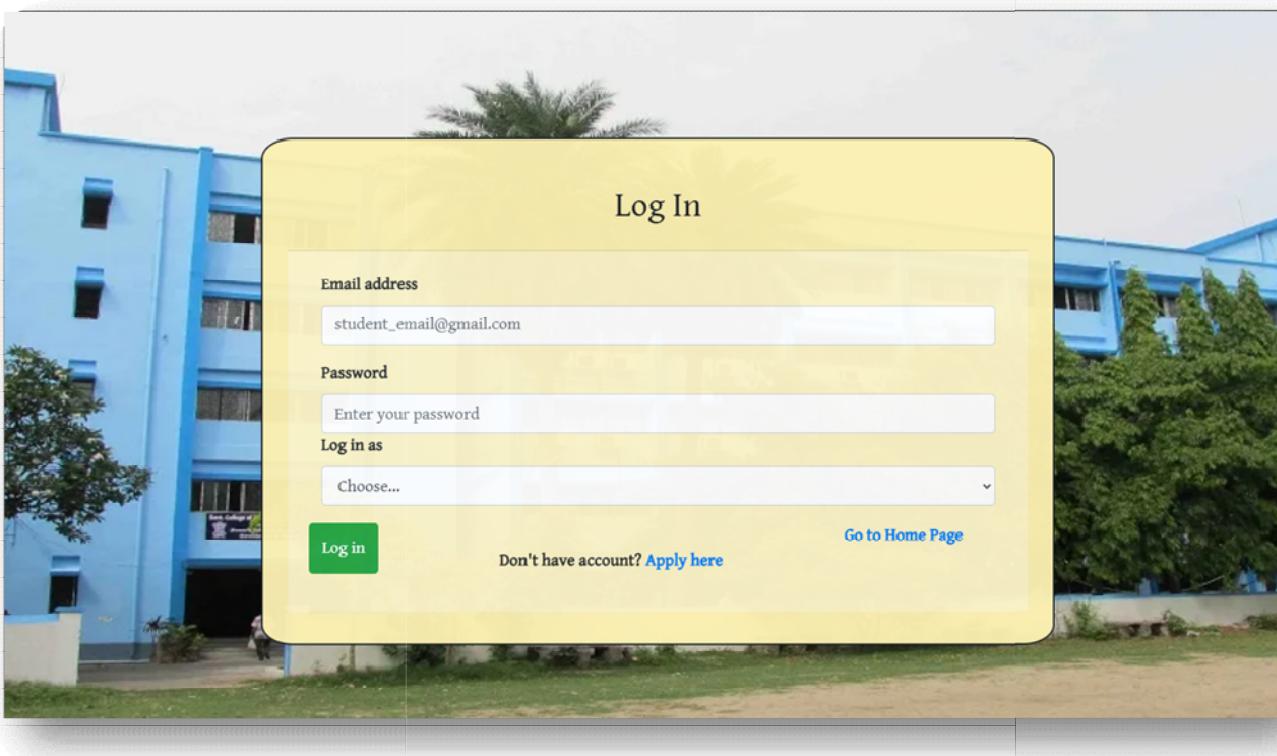
Buttons:

- Apply here (green button)
- Go to Home Page
- Already Applied? Log In here

4.1.4 - Login Form

This is the **login form**. An applicant who has successfully applied for the hostel can log in with the required credentials. There are three fields in this form:

- Email Address,
- Password
- Login type (admin or student)



If the values of all the fields of the application form are not properly given then it **flashes with an error message** which asks the user to fill up all the fields.

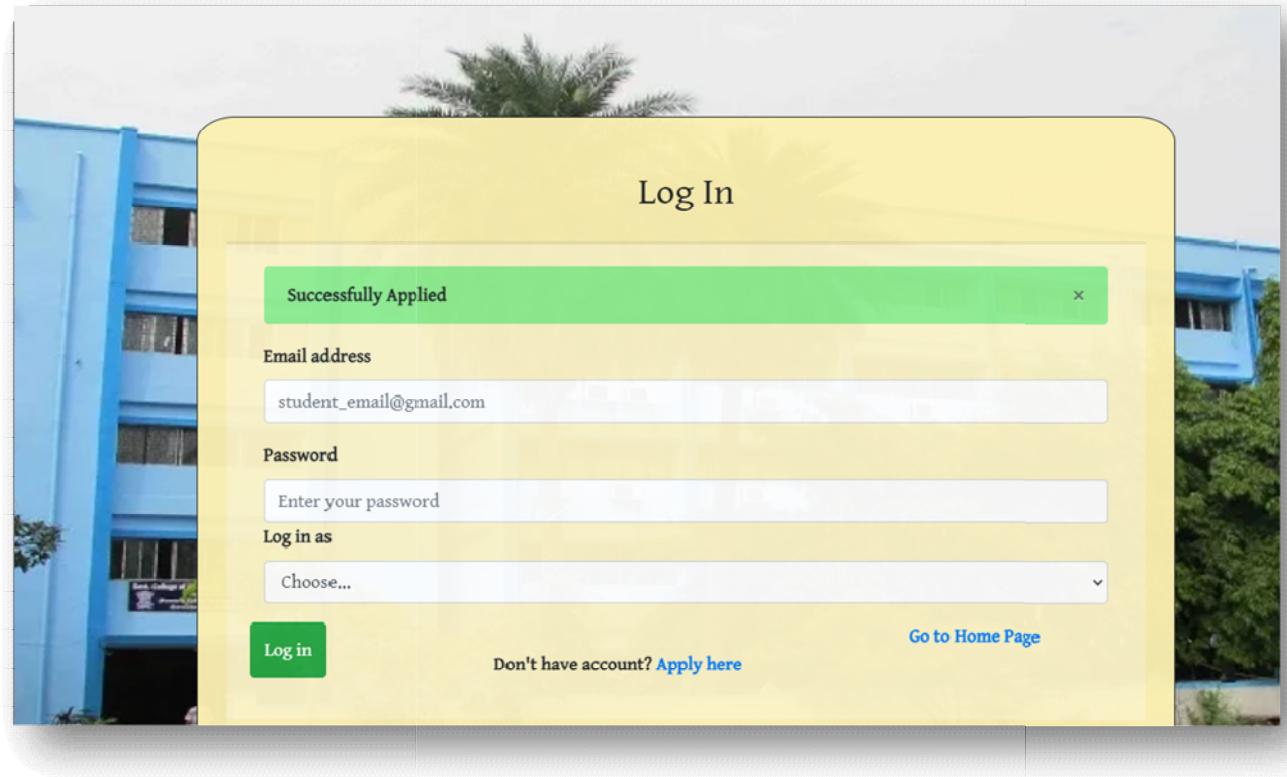
The screenshot shows a web application titled "Apply for college hostel". At the top, there is a pink banner with the text "Please fill up all the fields" and a close button (X). Below the banner, there are four input fields arranged in a 2x2 grid:

First Name Enter First Name Here..	Last Name Enter Last Name Here..
Email Email	Password Password (4-8 characters)

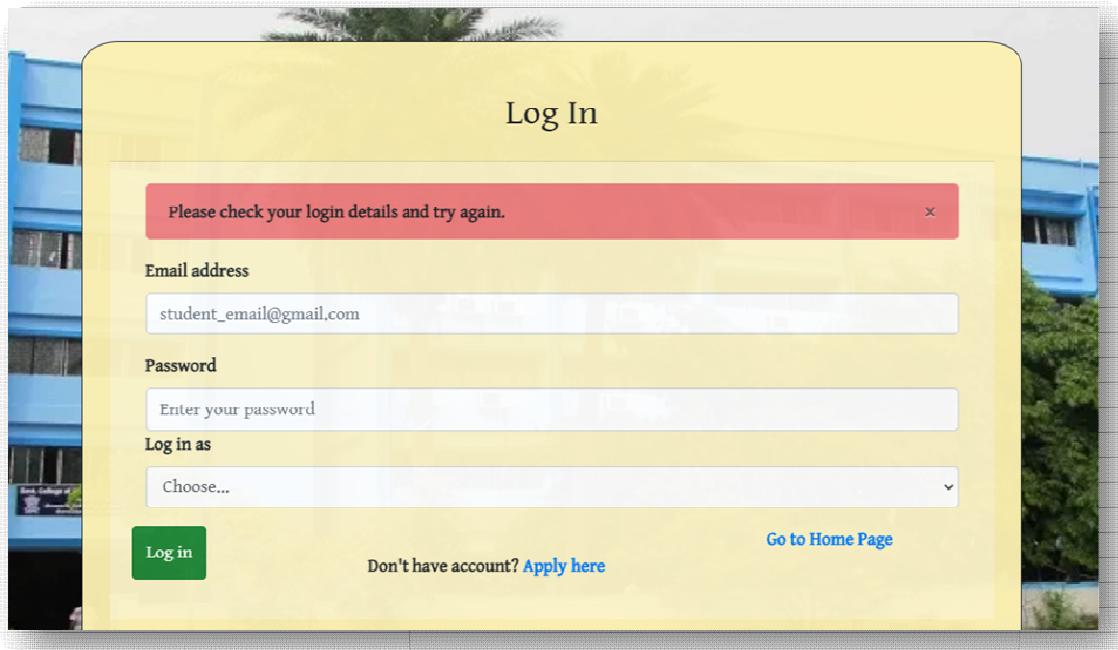
Below these, there are two more input fields:

Guardian's First Name Enter Guardian's First Name..	Guardian's Last Name Enter Guardian's Last Name..
--	--

If the values of all the fields of the login form are properly given , then it **flashes with a success message** which informs the user about successful application.



If the values of all the fields of the login form are not properly given then it **flashes with an error message** which asks the user to fill up all the fields.

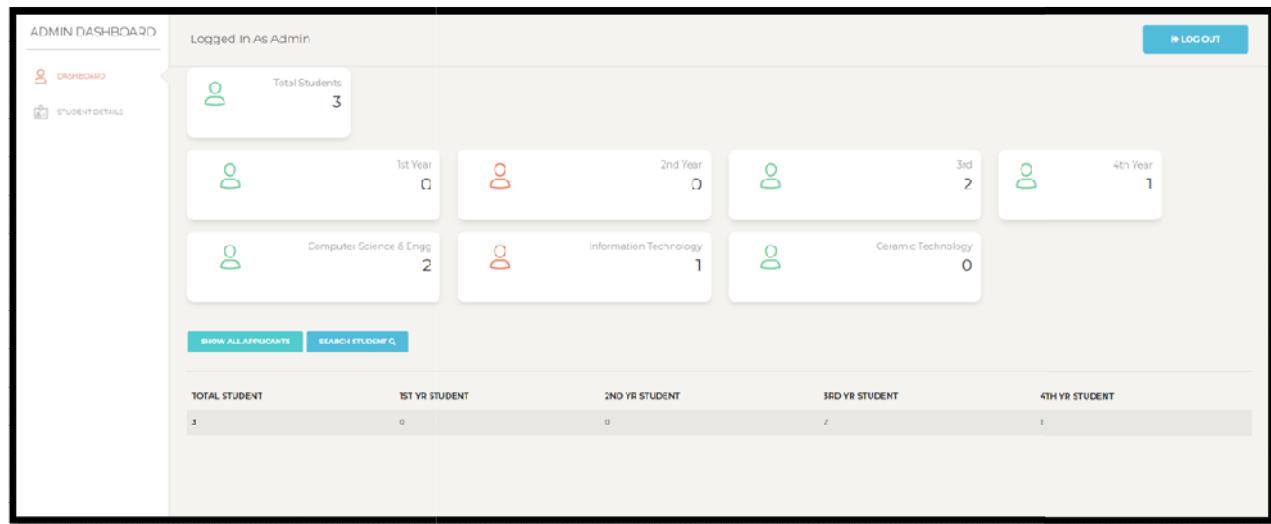


4.1.5 - Admin Dashboard

This is the dashboard view for the admin of this website. On the left side there is a navigation column which contains two sections, the **dashboard** section and **student details** section. This section contains information about the **total number of students** in the hostel. There is information about the **total number of students sorted according to the streams and years**. There are 2 buttons:

- **Show all applicants** : This shows the list of all the students who have applied for hostel.
- **Search student** : This is used to search for a student by his/her name.

On the top right corner there is a **Logout button** which is used to log the user out of the website.



4.1.6 - List of applicants

This section contains the list of all the applicants. The name, year, branch, distance and income are shown .Here the admin has two choices for each student. Admin can perform **Add Student** operation or **Reject** operation. If add student is performed, “**then add new students**” page opens up and the student gets added to the Students database.

NAME	YEAR	BRANCH	DISTANCE(KM)	INCOME(ANNUAL)	ACTIONS
Debmita Banerjee	4th	CSE	40	1234567	ADD STUDENT REJECT
Subhajit Bokhi	3rd	IT	100	123456	ADD STUDENT REJECT

4.1.7 - Add new students

Here the admin can assign room number to a student. The other details are automatically fetched from the application form. These details can not be edited by the admin.

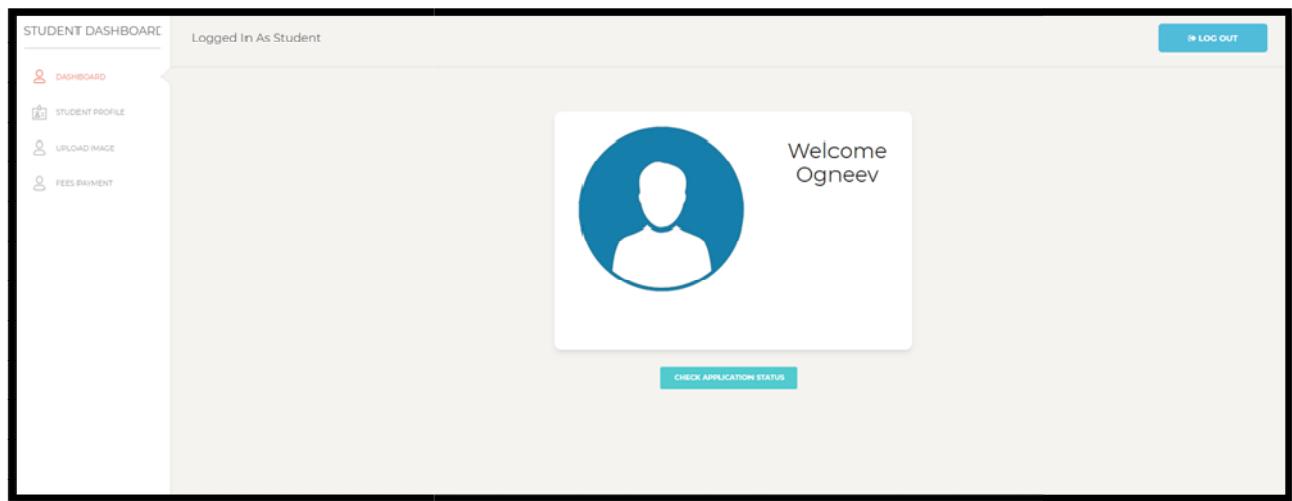
NAME	YEAR	BRANCH	ROOM NO.	ACTIONS
SOUHARDYA MANDAL	3rd	IT	6	EDIT DELETE
Ogneev Bhada	3rd	CSE	1	EDIT DELETE
Debmita Banerjee	4th	CSE	4	EDIT DELETE

4.1.8 - Student Dashboard

This page is the dashboard view of the student which is reached when a student logs in. This page contains a welcome message with the name of the student. A default avatar image is also assigned for each new student. On the left side there is a navigation bar containing four sections:

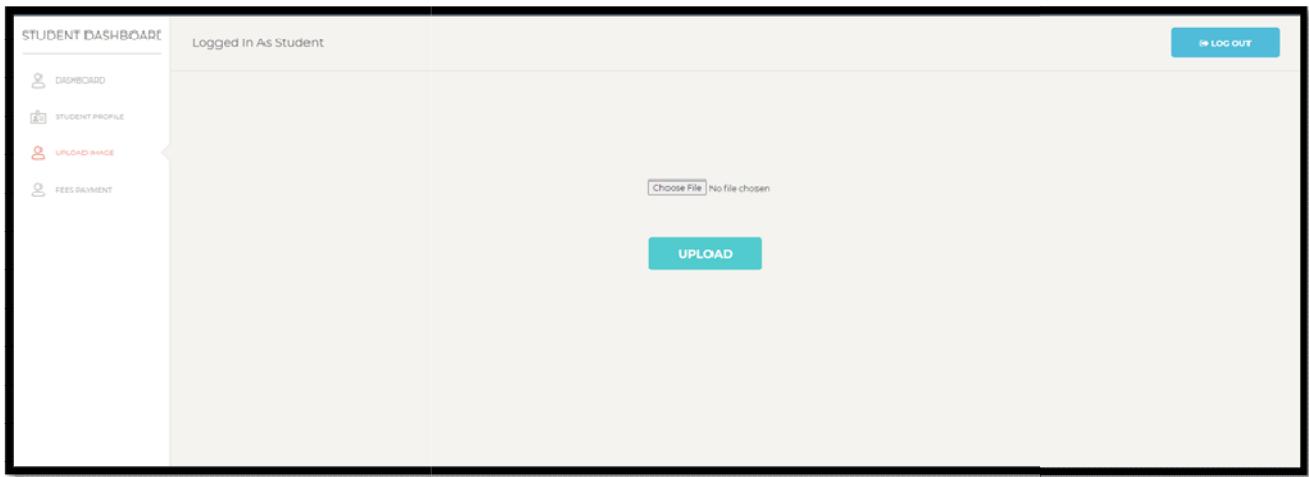
- Dashboard
- Student Profile
- Upload Image
- Fees Payment

There is also a button to **Check Application Status**. On the top right corner there is a **Logout button** which is used to log the user out of the website.

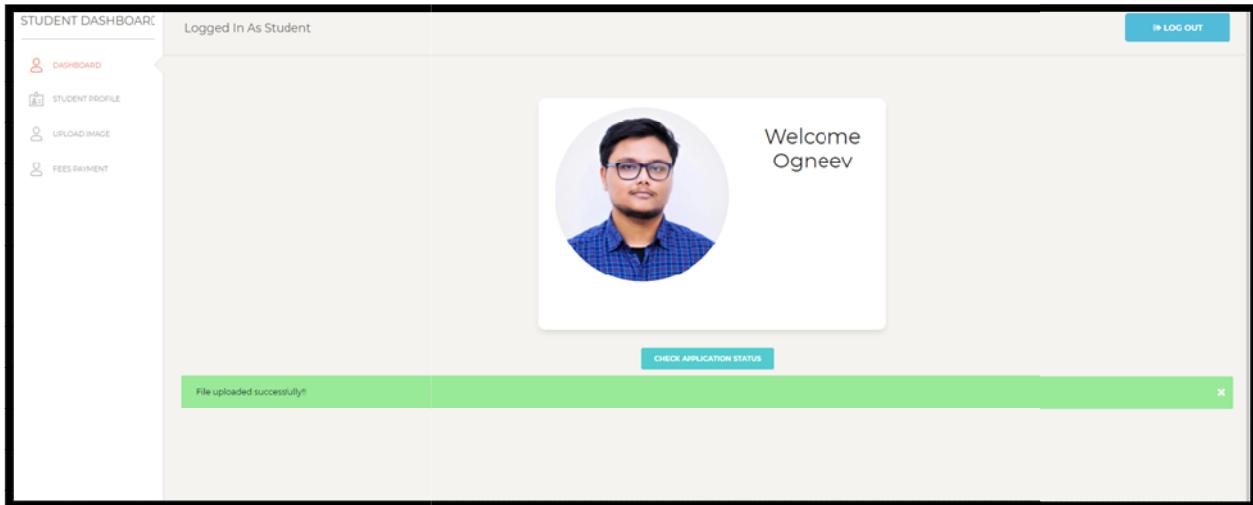


4.1.9 - Upload Image

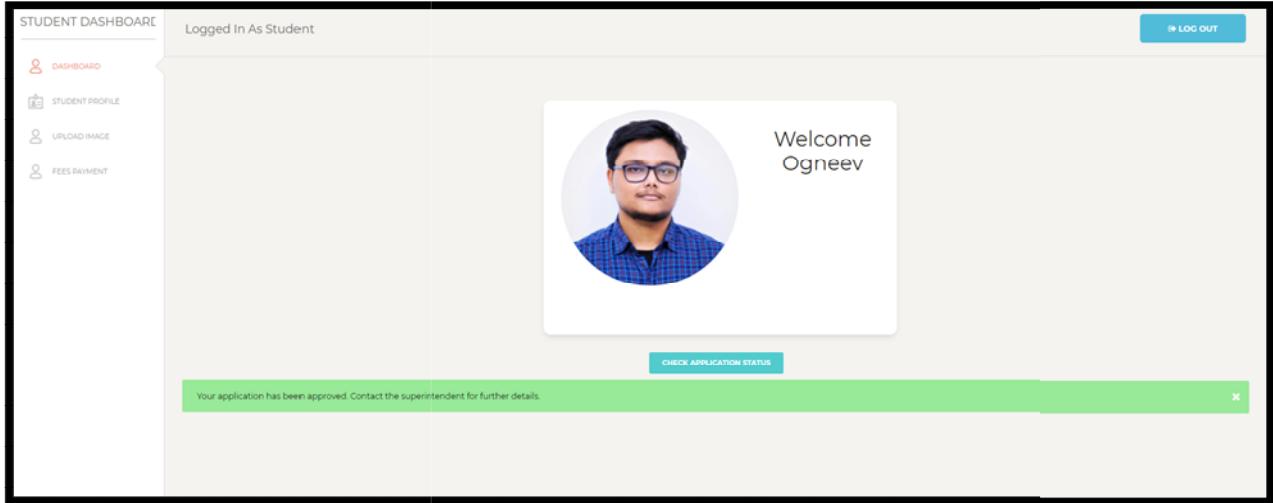
In the upload image section there is a **Choose file** button and a **Submit** button. If an image file is chosen properly and submitted then the image gets successfully uploaded and is set as the profile picture of the user.



If the image is uploaded successfully then it gets redirected to the dashboard section and the profile picture gets updated successfully. A flash message stating successful upload is also shown.



If the **Check Application Status** button on the dashboard is clicked then a message gets flashed. If the student is added to the hostel then it gives a success message otherwise an error message showing rejection of the application of the student.

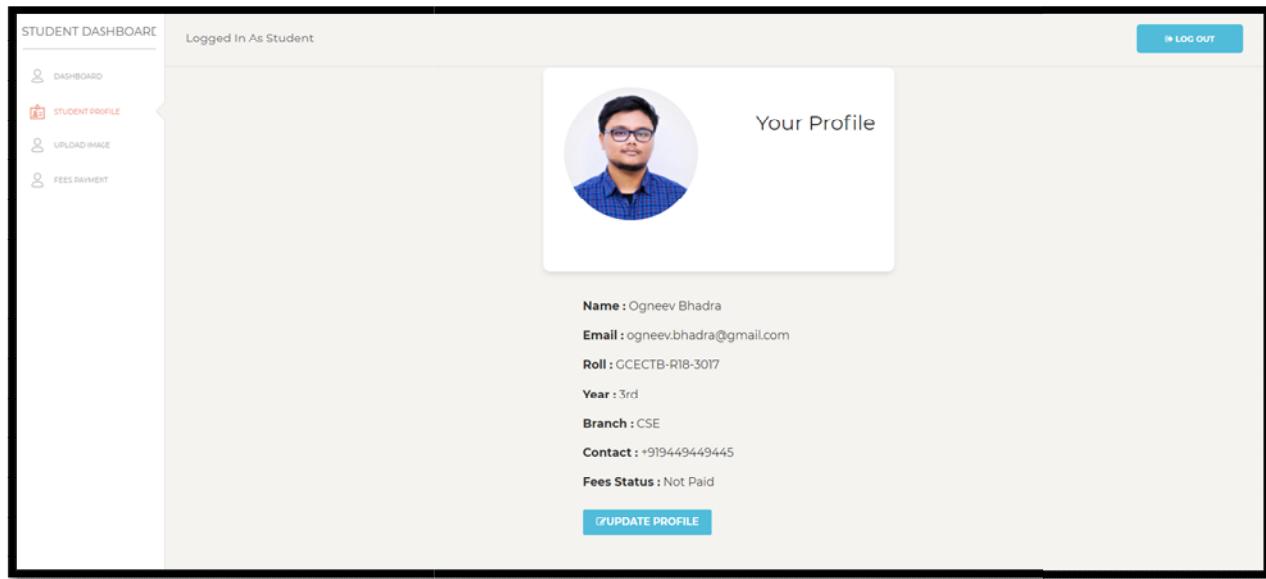


4.1.10 - Student Profile

This is the profile page of the student. It shows the profile picture and the information about the student. The informations shown here are:

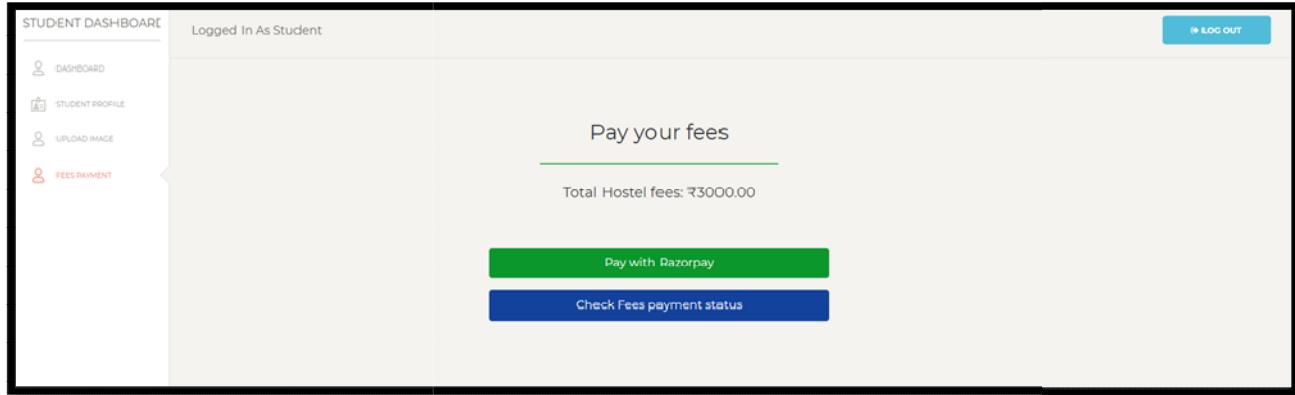
- Name
- Email
- Roll
- Year
- Branch
- Contact
- Fees Status (Paid / Not Paid)

There is another button **Update Profile** which allows the student to update his / her information.

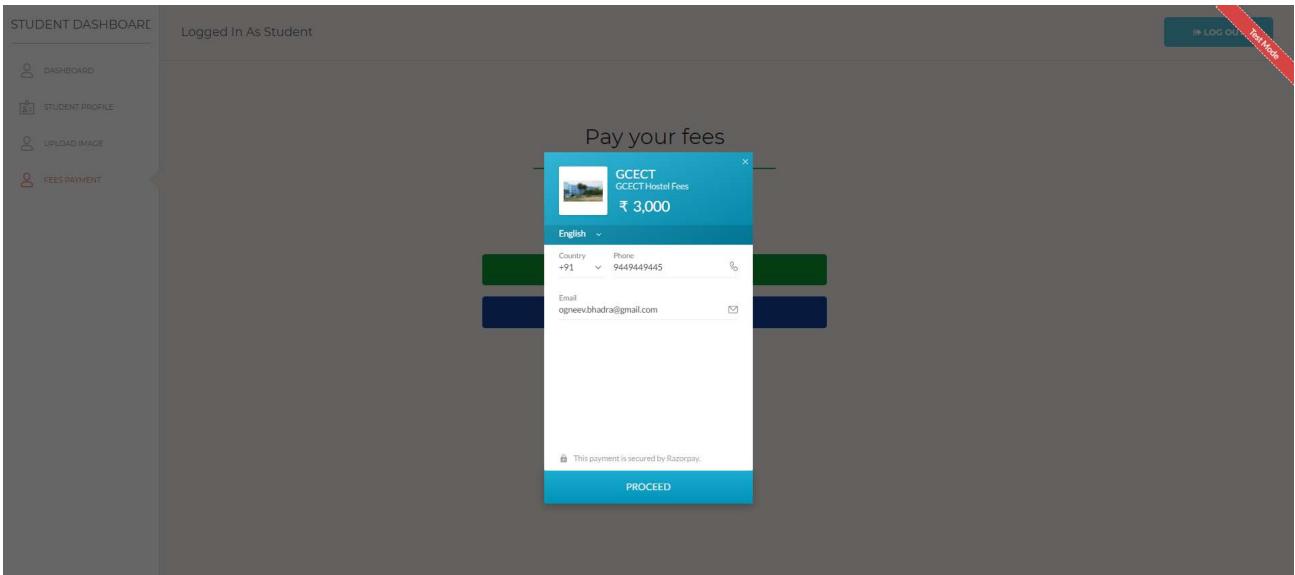


4.1.11 - Fees Payment

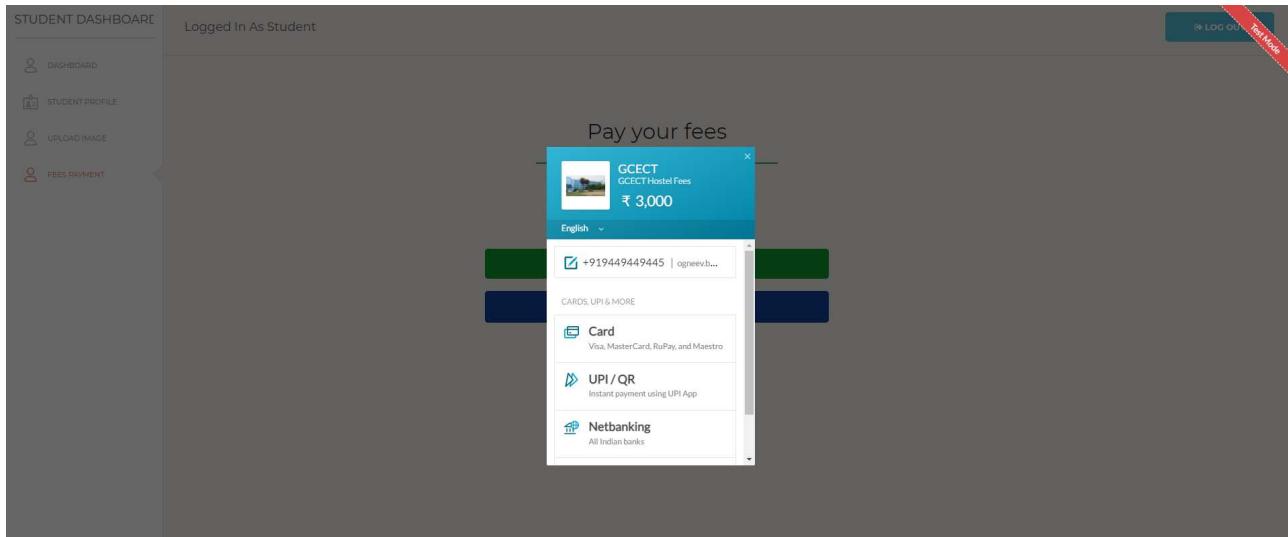
This is the section where a student can pay his / her fees. The student has to click on the button **Pay With Razorpay**. This opens up a payment gateway provided by Razorpay API which is very secure and handles itself. The payment value is fixed at rupees 3000/- . The student can pay using Credit/Debit Card, UPI, QR code and Net Banking.



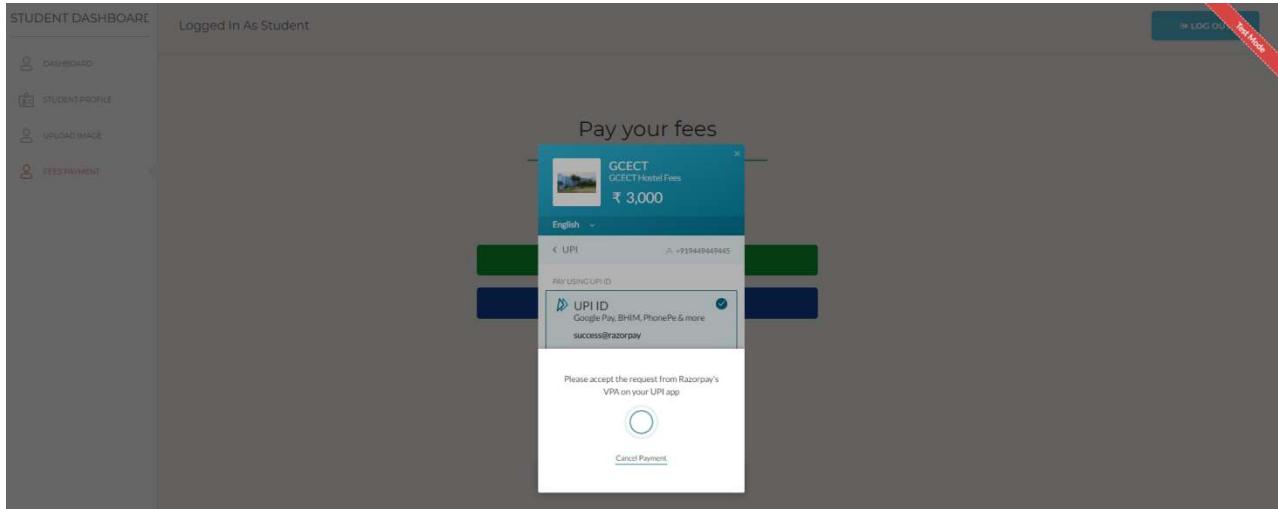
When the **Pay with Razorpay** button is clicked then it opens up the payment interface. The details are automatically fetched and the order details are shown as below.



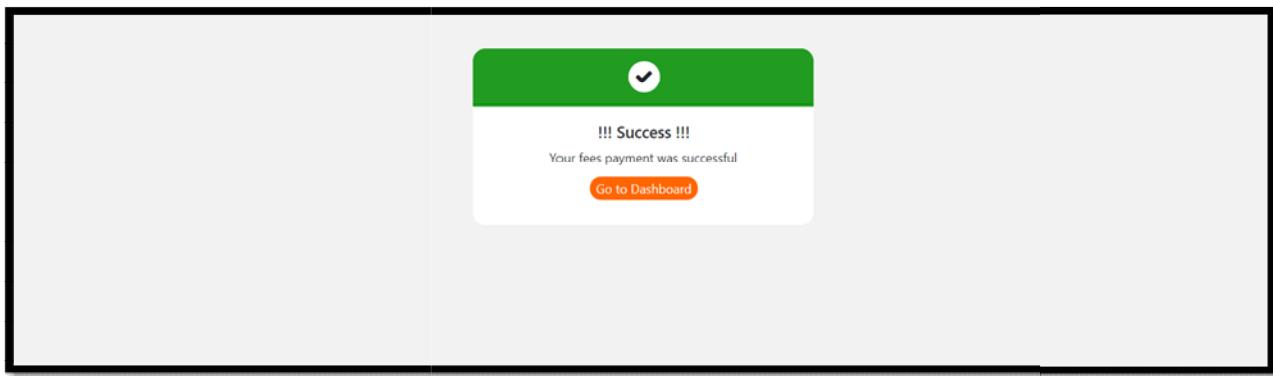
The different payment modes are shown . The student can pay using Credit/Debit Card, UPI, QR code and Net Banking. The user can fill up the payment credentials and advance for payment.



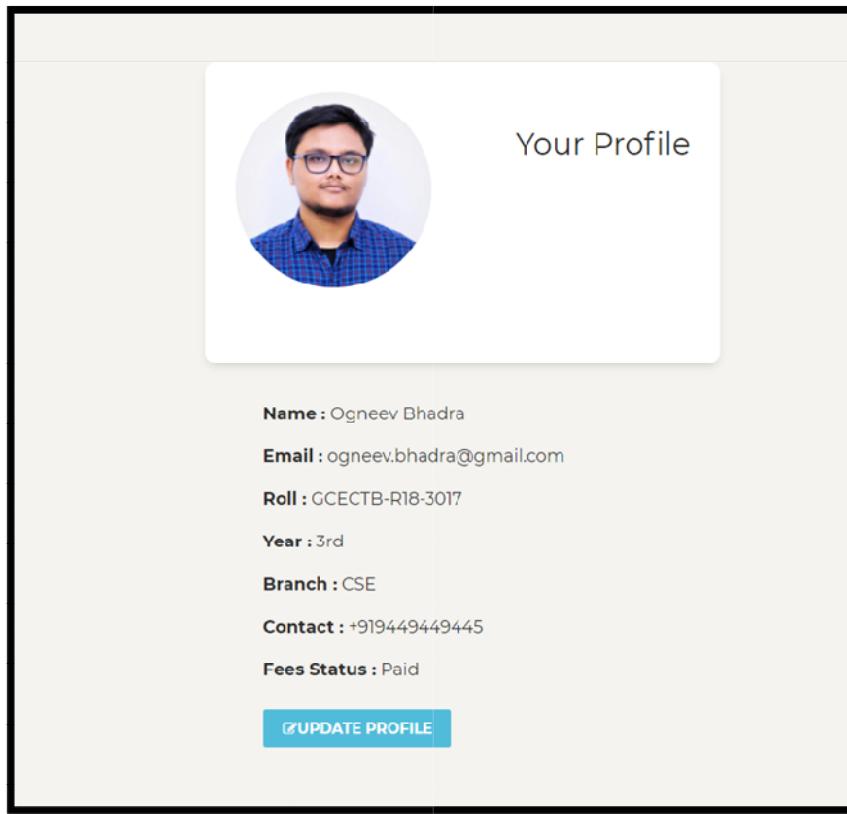
The Razorpay interface handles this payment. On failure, it stays there for the user to retry payment. On successful payment, it redirects to a new page which shows successful payment.



This page shows successful payment and confirms that the payment was successful.



If we visit the **student profile** we can see that the fees status got updated to **Paid**.



4.1.12 - Update Profile

The **Update Student Info** button on the profile page is used to update the information of the student. Only student can perform this task. The student can update :

- Name
- Roll Number
- Year
- Department
- Contact Number

STUDENT DASHBOARD

Logged In As Student

LOG OUT

Update Your Profile

Name: Coneev Bhadra

Roll No.: GCECTB-RIB-3017

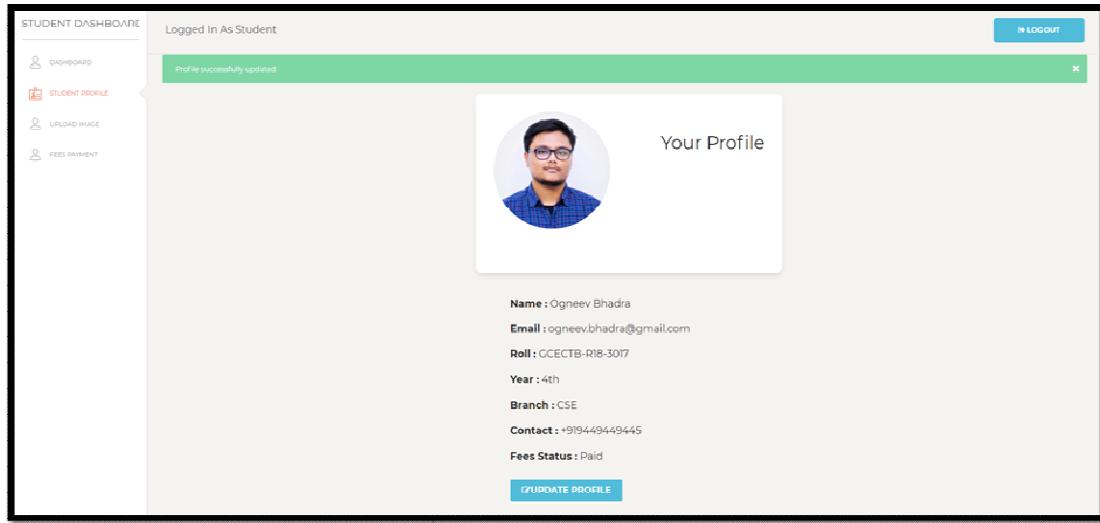
Year: 4th

Department: CSE

Contact No.: +919449449445

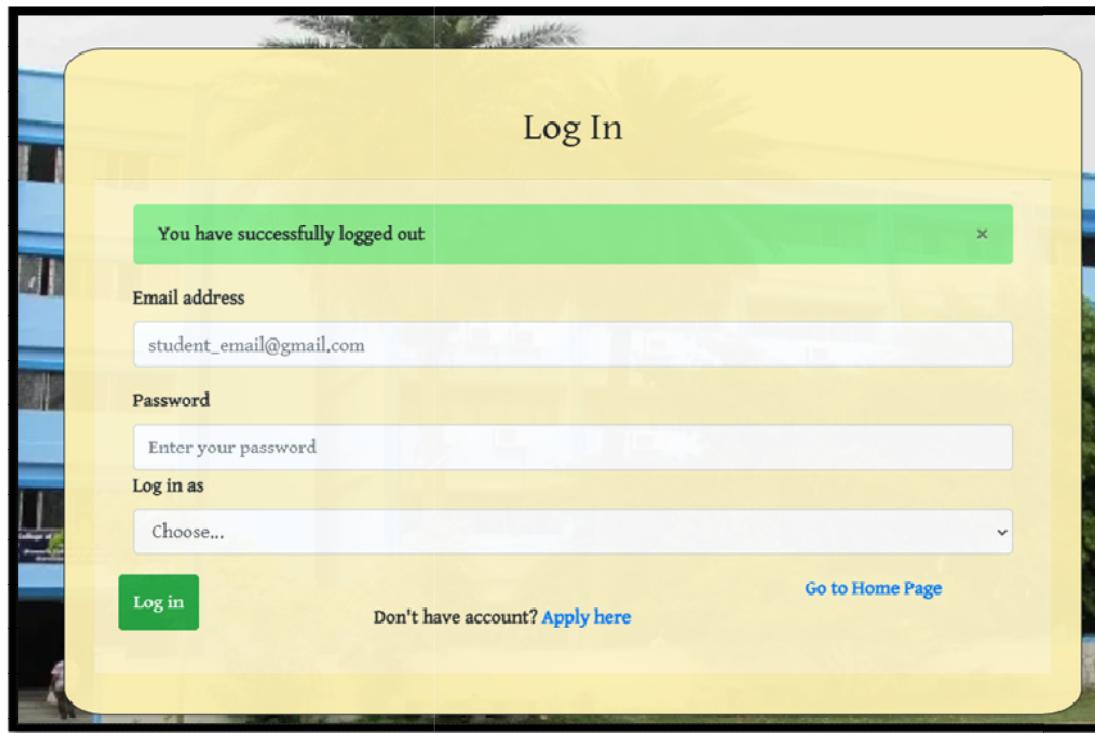
UPDATE STUDENT INFO

If the inputs are properly given and the **update profile** button is clicked , it flashes with a successful message that the profile was updated successfully.



4.1.13 - Log Out

If the log out button is clicked it closes the session for the user (admin or student) and a success message is flashed that the user has successfully logged out.



SYSTEM TESTING

Testing often accounts for more project than any other software engineering activity. If it is conducted haphazardly, time is wasted, unnecessary effort is expended, and even worse, errors sneak through undetected. It would therefore seem reasonable to establish a systematic strategy for testing software. The software testing is a critical step of the software quality assurance and represents the ultimate review of specification, design and code generation.

5.1. TESTING PROCEDURES

- **UNIT TESTING:** This is the testing of an individual module and is usually carried out to ensure the validity of a particular module. NOTE: It makes use of white box testing technique.
- **INTEGRATED TESTING:** Integrated testing is the testing of the system modules. This is done to identify errors, which relate to the interaction of different module, which cannot be found by unit testing but only through an interactive testing. NOTE: It makes use of black box testing technique.
- **SYSTEM TESTING:** System testing is the testing of the system against its initial objectives. It is done either in a simulated environment or in a live environment.
- **VALIDATION TESTING:** Validation testing is the testing where requirement established as part of software requirement analysis are validated against the software that has been constructed. NOTE: It makes use of black box testing technique.

5.2. OBJECTIVES OF SYSTEM TESTING

Once a system has been designed, it is necessary to undergo an exhaustive testing before installing the system. This is important because in some cases a small system error, not detect and corrected early before installation, may explode into a much larger problem later on. Testing is performed when user is asked to assist in identifying all possible situations. That might arise as regards the factors that effort was put to tackle the problem under consideration.

TEST CASES:

SL. NO.	FUNCTION	ACTION	RESULT	REDIRECT TO
1.	LOGIN	Entered valid details	Login successful	Student Dashboard/admin dashboard
2.	LOGIN	Entered wrong details	Error	Login page
3.	LOGIN	Kept field blank	Error	Login page
4.	APPLY	No field kept blank	Successfully Applied	Login page
5.	APPLY	Kept field blank	Error	Apply page
6.	UPDATE PROFILE	Kept field blank	Error	Update Profile page
7.	EDIT STUDENT	Kept field blank	Error	Edit student page

CONCLUSION

To conclude the description about the project: The project is based on the requirement specification of the user and the analysis of the existing system, with flexibility for future enhancement.

The expanded functionality of today's software requires an appropriate approach towards software development. This hostel management software is designed for people who want to manage various activities in the hostel. For the past few years the numbers of educational institutions are increasing rapidly. Thereby the numbers of hostels are also increasing for the accommodation of the students studying in this institution. And hence there is a lot of strain on the person who are running the hostel and software's are not usually used in this context. This particular project deals with the problems on managing a hostel and avoids the problems which occur when carried manually.

Identification of the drawbacks of the existing system leads to the designing of computerized system that will be compatible with the existing system and more user friendly and GUI oriented.

