PES UNIVERSITY

Hostel-Management-System

Database Management System
Assignment-004
(UE 19CS302)

Section: G **Date:07-12-2021**

Name:	**SRN:**
Traimbak Mahabal Bhat	PES2UG19CS434
Vyshak R	PES2UG19CS463
Suhan B Revankar	PES2UG19CS412

Problem Statement:

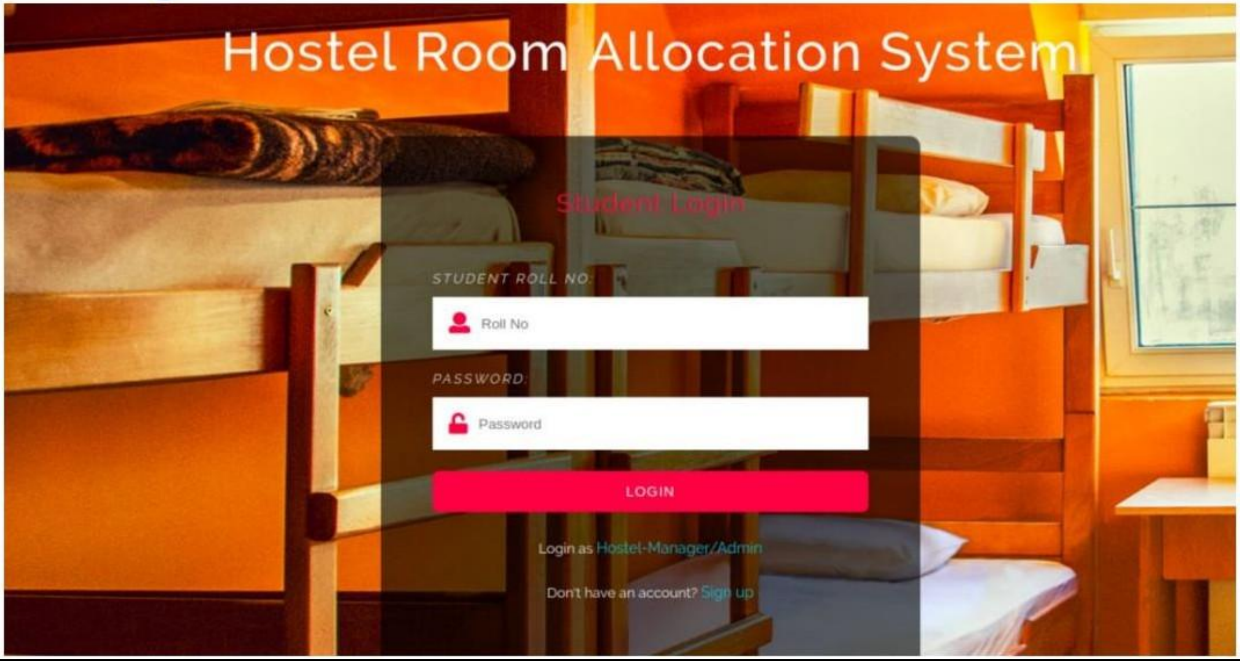
We wish to create a very efficient and interaction-friendly database management system to manage an institution’s data. The initial problem faced by many institutions is that there are many students whose information needs to be stored and managed, and in addition, they also need to manage the data of all the staff working there, keep track of all the courses offered to students, transportation provided to students and need to maintainfinancial management of the canteen.

Justification of languages used for front end
HTML describes and defines the content and basic structure of the website.

CSS is responsible for outlining the colours, font and positioning of the content on a website.

JavaScript controls the behaviour of the website, it was designed to manipulate web pages and it is usedto create interactive functionality.

Login Page

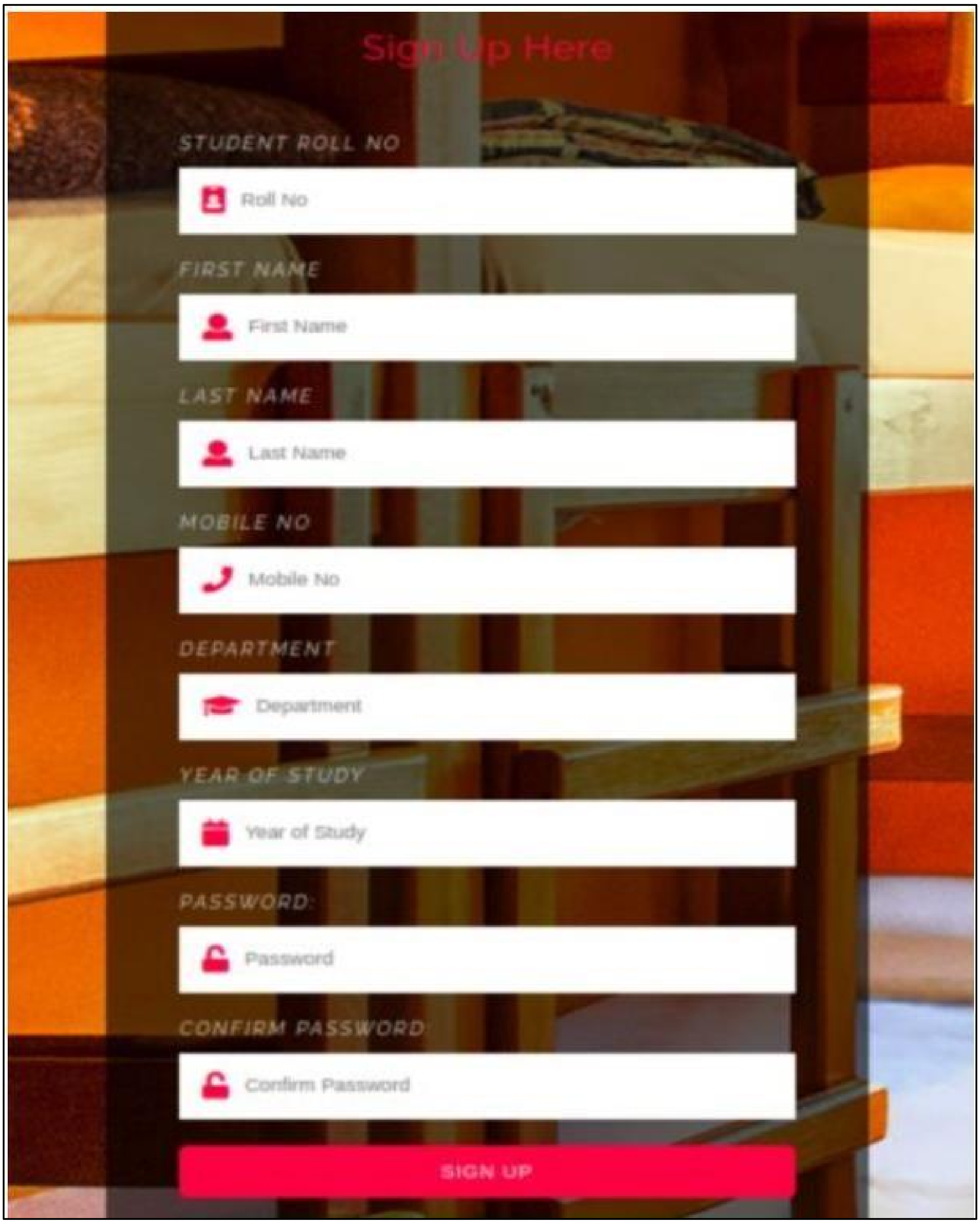


Hostel Room Allocation System

Student Login

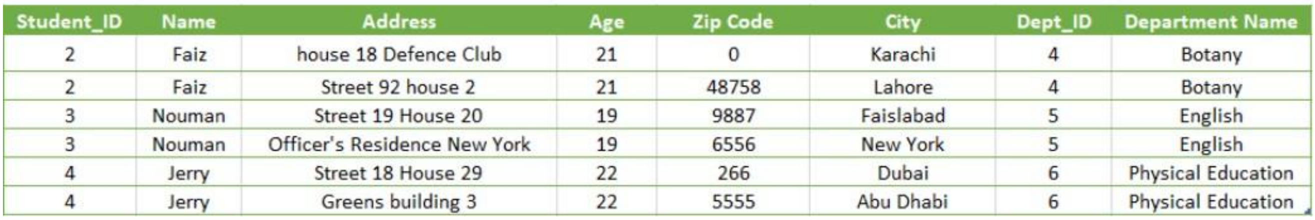
Login as Hostel-Manager/Admin

Don't have an account? Sign up



Additional Queries (Normalisation)

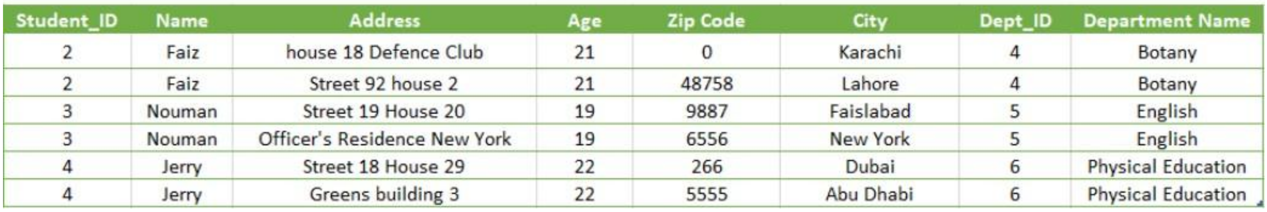
1NF Example :



Student_ID	Name	Address	Age	Zip Code	City	Dept_ID	Department Name
2	Faiz	house 18 Defence Club	21	0	Karachi	4	Botany
2	Faiz	Street 92 house 2	21	48758	Lahore	4	Botany
3	Nouman	Street 19 House 20	19	9887	Faislabad	5	English
3	Nouman	Officer's Residence New York	19	6556	New York	5	English
4	Jerry	Street 18 House 29	22	266	Dubai	6	Physical Education
4	Jerry	Greens building 3	22	5555	Abu Dhabi	6	Physical Education

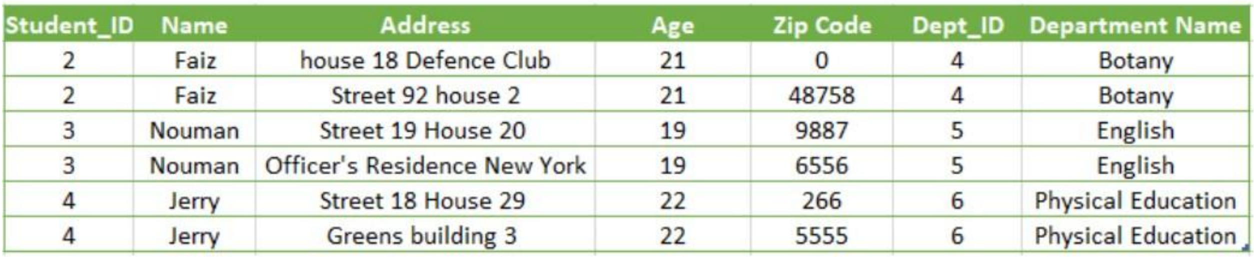
SECOND NORMAL FORM 2NF

Table is 2NF only and only if it already has 1NF and then it must be absolute, that there should not be partial dependency of any column on primary key. This part can be confusing , so let’s try to spend some more time here.

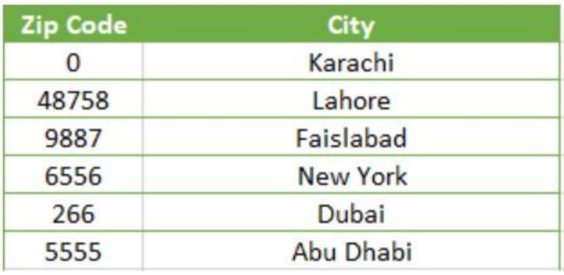


Student_ID	Name	Address	Age	Zip Code	City	Dept_ID	Department Name
2	Faiz	house 18 Defence Club	21	0	Karachi	4	Botany
2	Faiz	Street 92 house 2	21	48758	Lahore	4	Botany
3	Nouman	Street 19 House 20	19	9887	Faislabad	5	English
3	Nouman	Officer's Residence New York	19	6556	New York	5	English
4	Jerry	Street 18 House 29	22	266	Dubai	6	Physical Education
4	Jerry	Greens building 3	22	5555	Abu Dhabi	6	Physical Education

2NF Example

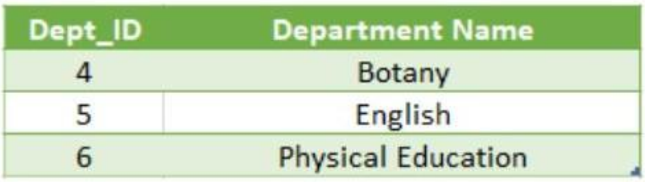


Student_ID	Name	Address	Age	Zip Code	Dept_ID	Department Name
2	Faiz	house 18 Defence Club	21	0	4	Botany
2	Faiz	Street 92 house 2	21	48758	4	Botany
3	Nouman	Street 19 House 20	19	9887	5	English
3	Nouman	Officer's Residence New York	19	6556	5	English
4	Jerry	Street 18 House 29	22	266	6	Physical Education
4	Jerry	Greens building 3	22	5555	6	Physical Education

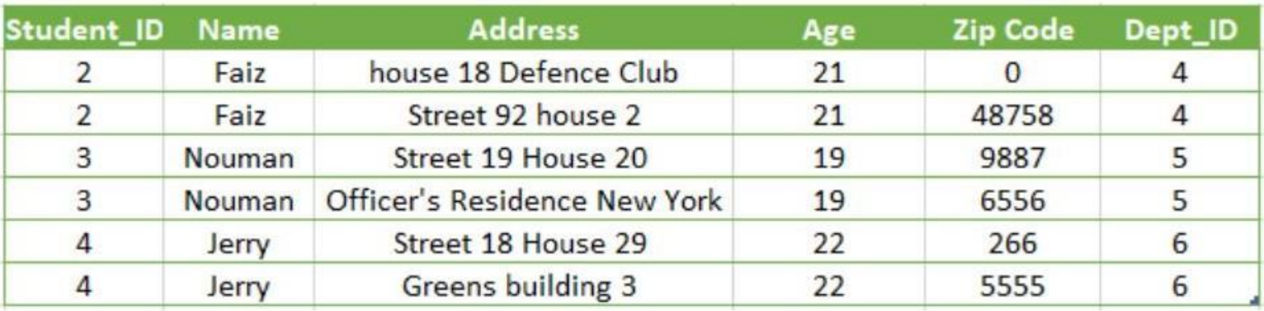


Zip Code	City
0	Karachi
48758	Lahore
9887	Faislabad
6556	New York
266	Dubai
5555	Abu Dhabi

3NF Example

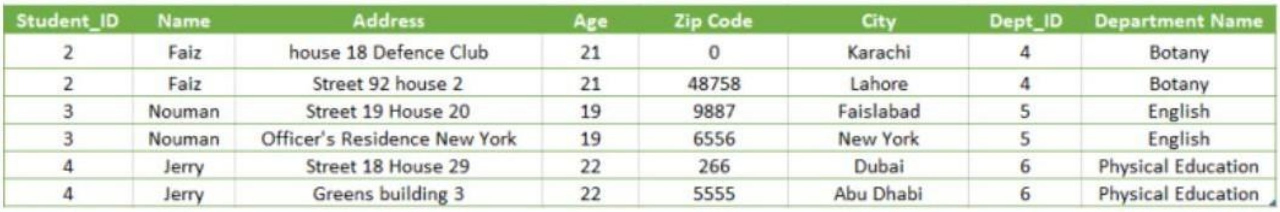


Dept_ID	Department Name
4	Botany
5	English
6	Physical Education



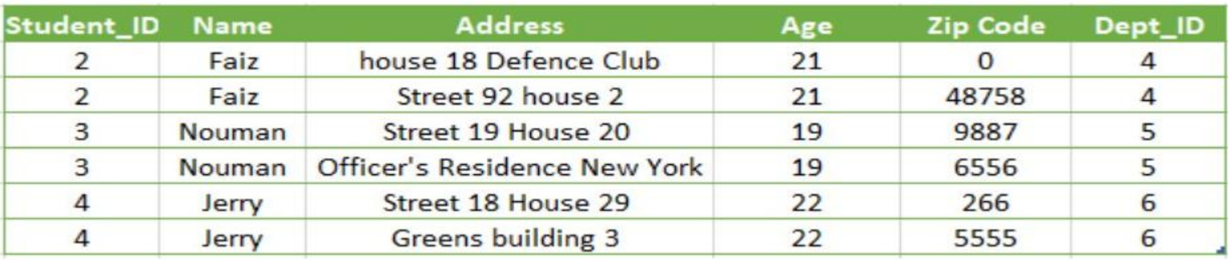
Student_ID	Name	Address	Age	Zip Code	Dept_ID
2	Faiz	house 18 Defence Club	21	0	4
2	Faiz	Street 92 house 2	21	48758	4
3	Nouman	Street 19 House 20	19	9887	5
3	Nouman	Officer's Residence New York	19	6556	5
4	Jerry	Street 18 House 29	22	266	6
4	Jerry	Greens building 3	22	5555	6

So we have converted this long table

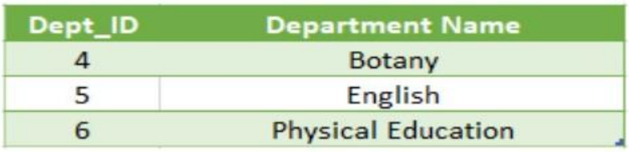


Student_ID	Name	Address	Age	Zip Code	City	Dept_ID	Department Name
2	Faiz	house 18 Defence Club	21	0	Karachi	4	Botany
2	Faiz	Street 92 house 2	21	48758	Lahore	4	Botany
3	Nouman	Street 19 House 20	19	9887	Faislabad	5	English
3	Nouman	Officer's Residence New York	19	6556	New York	5	English
4	Jerry	Street 18 House 29	22	266	Dubai	6	Physical Education
4	Jerry	Greens building 3	22	5555	Abu Dhabi	6	Physical Education

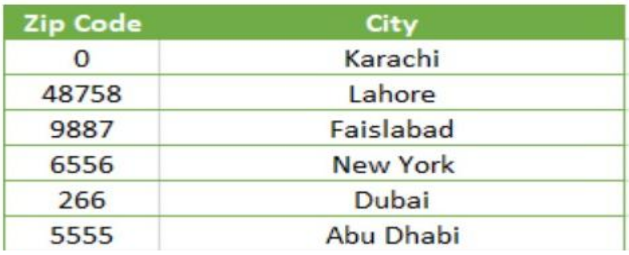
Into these well-designed tables



Student_ID	Name	Address	Age	Zip Code	Dept_ID
2	Faiz	house 18 Defence Club	21	0	4
2	Faiz	Street 92 house 2	21	48758	4
3	Nouman	Street 19 House 20	19	9887	5
3	Nouman	Officer's Residence New York	19	6556	5
4	Jerry	Street 18 House 29	22	266	6
4	Jerry	Greens building 3	22	5555	6



Dept_ID	Department Name
4	Botany
5	English
6	Physical Education



Zip Code	City
0	Karachi
48758	Lahore
9887	Faislabad
6556	New York
266	Dubai
5555	Abu Dhabi

Database Migration and Support

There are some tools to help the migration, but in the end, MySQL is a relational databasewhich has a completely different structure from noSQL databases.

In the end, you will almost always have to dothese four steps

1. Get to know MongoDB. Download it, read thetutorials, try some toy projects.

2. Think about how to represent your model inits document store.

3. Migrate the data from the database to MongoDB, probably simply by writing a bunch ofSELECT * FROM statements against the database and then loading the data into your MongoDB model using the language of your choice.

4. Rewrite your application code to query MongoDB through statements such as insert() orfind().

Dependencies for database connectivity:

We made use of Sql module to create the frontend using python and further connect to the postgresql database.

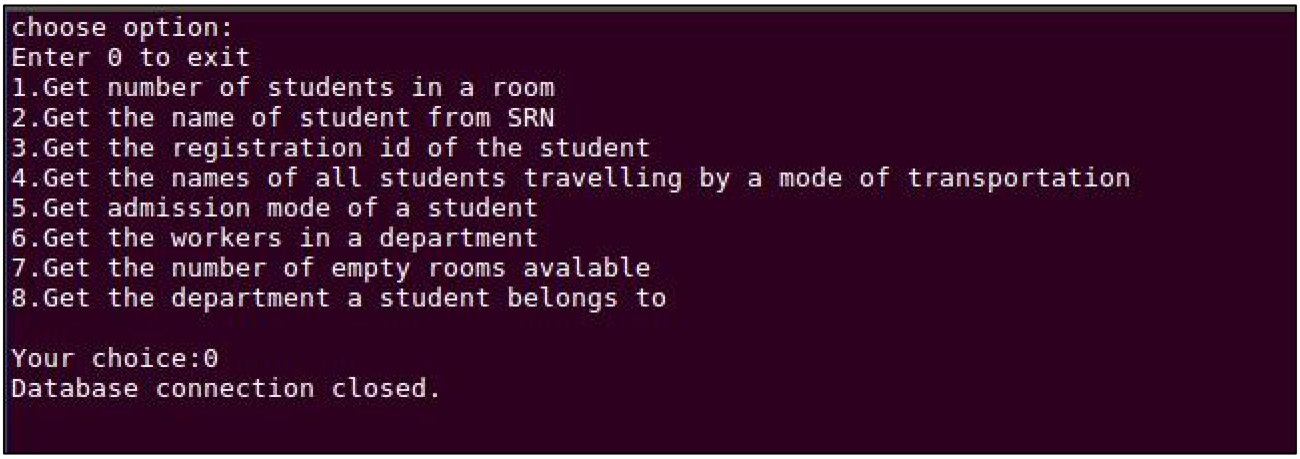
During the process, wecreated 3 files: database.ini, config.py and connect.py

The database.ini file contains information about the host, database, username and password that help gain access to the database in postgresql.

The config.py file contains a function config() that returns connection parameters after reading database.ini.

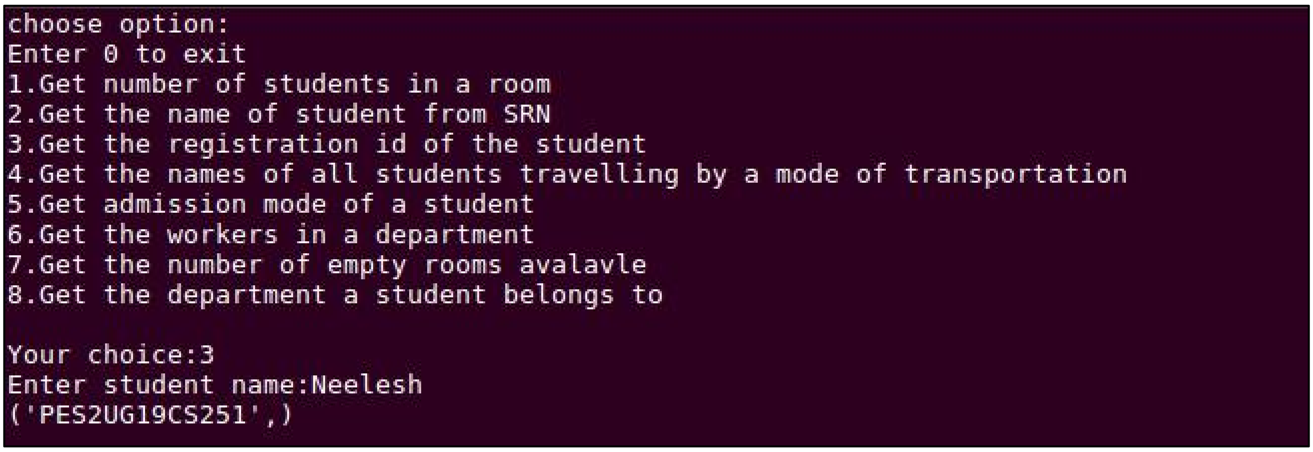
Finally, the function connect() in connect.py connects to the college database and prints out the results of the queries executed on the database.

Screenshot for front end:



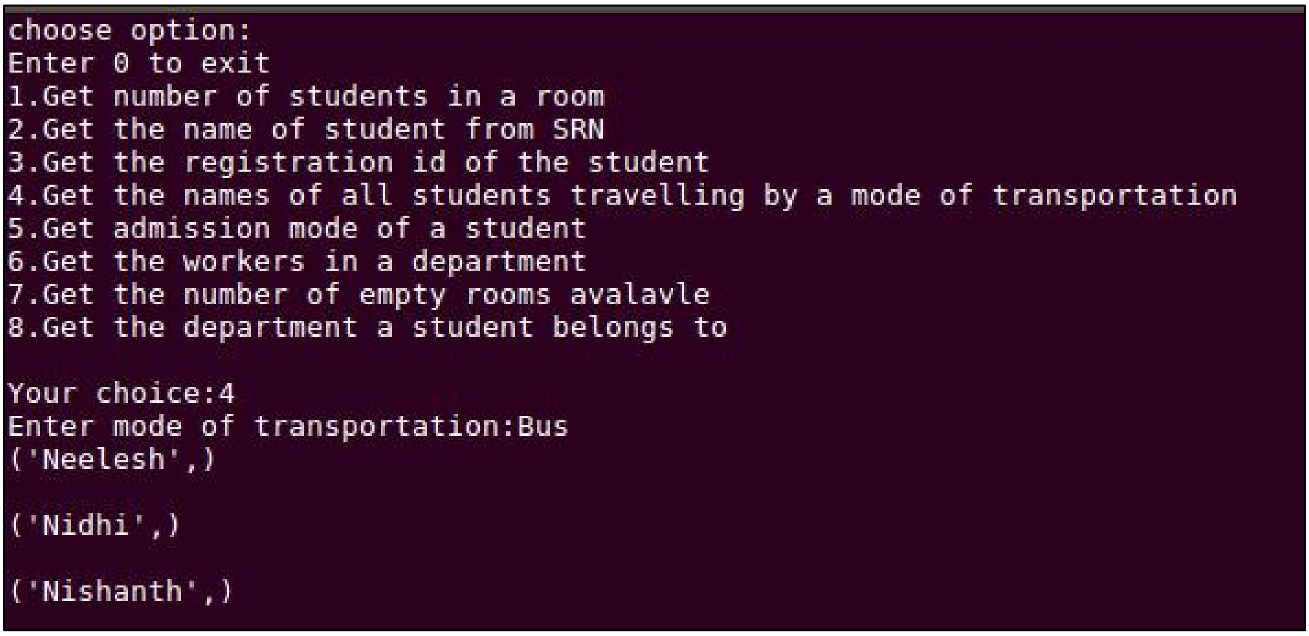
```
choose option:
Enter 0 to exit
1.Get number of students in a room
2.Get the name of student from SRN
3.Get the registration id of the student
4.Get the names of all students travelling by a mode of transportation
5.Get admission mode of a student
6.Get the workers in a department
7.Get the number of empty rooms avalable
8.Get the department a student belongs to

Your choice:0
Database connection closed.
```



```
choose option:
Enter 0 to exit
1.Get number of students in a room
2.Get the name of student from SRN
3.Get the registration id of the student
4.Get the names of all students travelling by a mode of transportation
5.Get admission mode of a student
6.Get the workers in a department
7.Get the number of empty rooms avalavle
8.Get the department a student belongs to

Your choice:3
Enter student name:Neelesh
('PES2UG19CS251',)
```



```
choose option:
Enter 0 to exit
1.Get number of students in a room
2.Get the name of student from SRN
3.Get the registration id of the student
4.Get the names of all students travelling by a mode of transportation
5.Get admission mode of a student
6.Get the workers in a department
7.Get the number of empty rooms avalavle
8.Get the department a student belongs to

Your choice:4
Enter mode of transportation:Bus
('Neelesh',)

('Nidhi',)

('Nishanth',)
```

Change in application that leads to schema and constraint change:

A schema is a collection of database objectslike tables, triggers, stored procedures, etc. A schema relates to a user which is known as the schema owner. Database may haveone or more schema.

Constraints are a set of rules implemented ontables in relational databases to dictate what data can be inserted, updated, or deleted inits tables.

There are multiple reasons which might lead tothe change of schema and constraints in the college database.

We have specified a few reasons forchanging Schema and Constraint:

1. In the Students relation, we might want to remove the Mode_of_Transportation attribute as we can make a small furtherchange to the Bus_Num attribute (shown below) so we can reduce one attribute inthe relation.

2. In the Students relation again, we see thatthere is an attribute that has many null values, and NULL values are something that should be avoided in relation as it could lead to further confusion. So, with this inmind and with us previously removing the Mode_of_Transportation attribute we can nowdefault the value entered this attribute as 0 . Which will represent that if a student has Bus_Num as 0 he doesn't use a collegebus.

3. In the Worker relation, we see that there is an attribute called Salary whichdoesn't give us a clear image of whether it is weekly salary monthly salary, oryearly salary so for better understanding we rename the column to Monthly_Salary.

4. Every year in college a set of students graduate and are no longer affiliated with the institution so after they graduate if we want to be space conscious, we can remove the students who have graduated thisyear (who were in 8th sem.)
   For this we first need to disable all triggers in Student relations then we can delete the students.

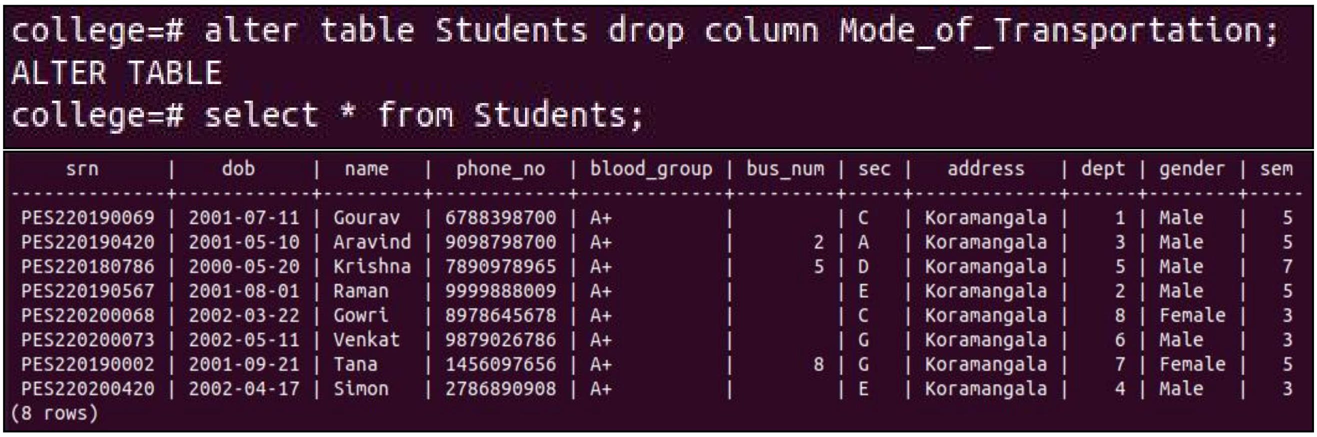
5.In the Worker relation, all the workers work for a fixed durationfrom 6-5 which isa total of 11 hours so instead of setting 11 hours for every new worker explicitly,we can make 11 hours default.

6.In bus relation, we add a constraint called unique_red makingreg_no unique astwo buses cannot have the same registrationnumber.

7.In bus relation again we alter column Insurance_no to make itset to not null as it we need a vehicle to have an activevalid license or else it is illegal to drive it in.

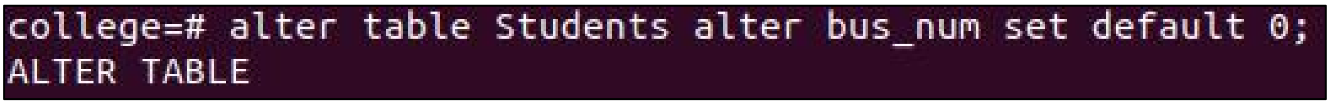
Screenshot for schema and constraint change statements:

1.



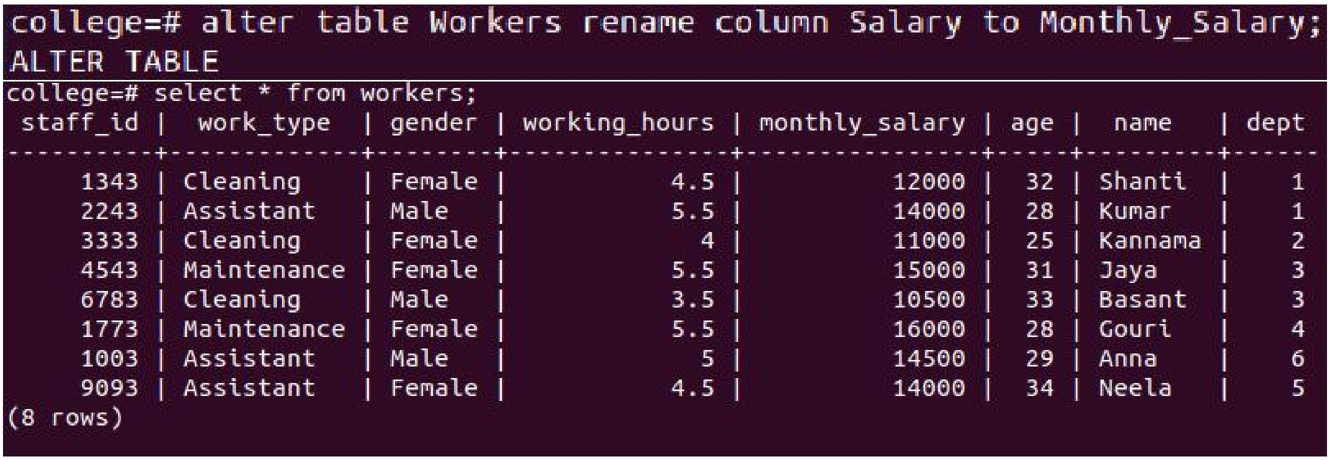
```
college=# alter table Students drop column Mode_of_Transportation;
ALTER TABLE
college=# select * from Students;
     srn      |    dob     |  name   |  phone_no  | blood_group | bus_num | sec |   address   | dept | gender | sem
--------------+------------+---------+------------+-------------+---------+-----+-------------+------+--------+-----
 PES220190069 | 2001-07-11 | Gourav  | 6788398700 | A+          |         | C   | Koramangala |    1 | Male   |   5
 PES220190420 | 2001-05-10 | Aravind | 9098798700 | A+          |       2 | A   | Koramangala |    3 | Male   |   5
 PES220180786 | 2000-05-20 | Krishna | 7890978965 | A+          |       5 | D   | Koramangala |    5 | Male   |   7
 PES220190567 | 2001-08-01 | Raman   | 9999888009 | A+          |         | E   | Koramangala |    2 | Male   |   5
 PES220200068 | 2002-03-22 | Gowri   | 8978645678 | A+          |         | C   | Koramangala |    8 | Female |   3
 PES220200073 | 2002-05-11 | Venkat  | 9879026786 | A+          |         | G   | Koramangala |    6 | Male   |   3
 PES220190002 | 2001-09-21 | Tana    | 1456097656 | A+          |       8 | G   | Koramangala |    7 | Female |   5
 PES220200420 | 2002-04-17 | Simon   | 2786890908 | A+          |         | E   | Koramangala |    4 | Male   |   3
(8 rows)
```

2.



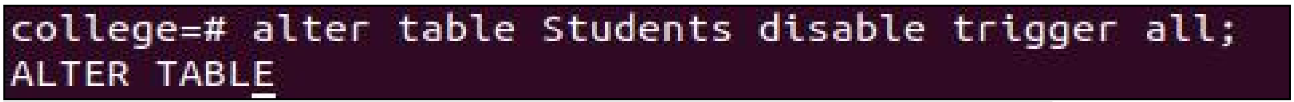
```
college=# alter table Students alter bus_num set default 0;
ALTER TABLE
```

3.

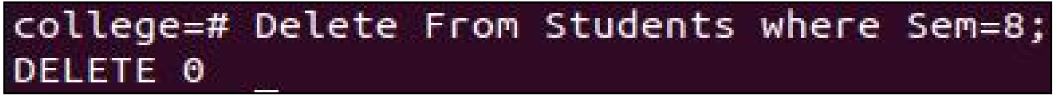


```
college=# alter table Workers rename column Salary to Monthly_Salary;
ALTER TABLE
college=# select * from workers;
 staff_id |  work_type  | gender | working_hours | monthly_salary | age |  name   | dept
----------+-------------+--------+---------------+----------------+-----+---------+------
     1343 | Cleaning    | Female |           4.5 |          12000 |  32 | Shanti  |    1
     2243 | Assistant   | Male   |           5.5 |          14000 |  28 | Kumar   |    1
     3333 | Cleaning    | Female |             4 |          11000 |  25 | Kannama |    2
     4543 | Maintenance | Female |           5.5 |          15000 |  31 | Jaya    |    3
     6783 | Cleaning    | Male   |           3.5 |          10500 |  33 | Basant  |    3
     1773 | Maintenance | Female |           5.5 |          16000 |  28 | Gouri   |    4
     1003 | Assistant   | Male   |             5 |          14500 |  29 | Anna    |    6
     9093 | Assistant   | Female |           4.5 |          14000 |  34 | Neela   |    5
(8 rows)
```

4.

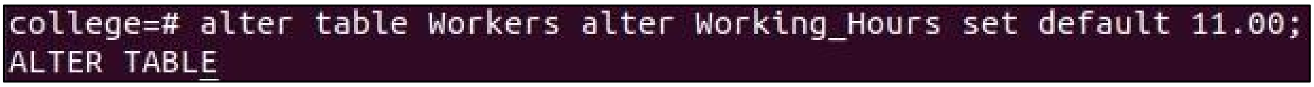


```
college=# alter table Students disable trigger all;
ALTER TABLE
```



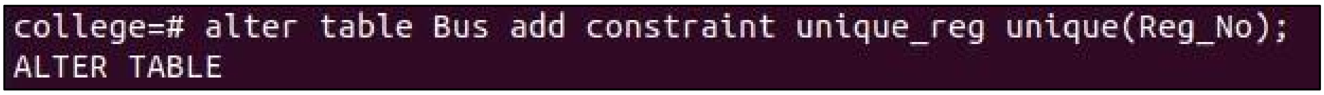
```
college=# Delete From Students where Sem=8;
DELETE 0
```

5.



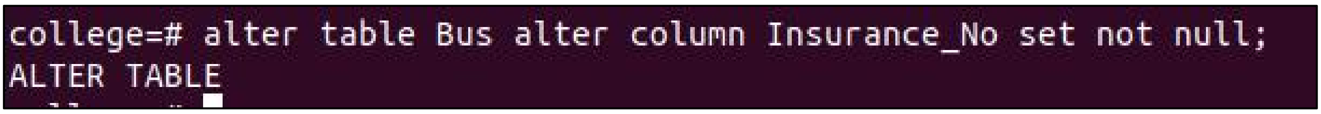
```
college=# alter table Workers alter Working_Hours set default 11.00;
ALTER TABLE
```

6.



```
college=# alter table Bus add constraint unique_reg unique(Reg_No);
ALTER TABLE
```

7.



```
college=# alter table Bus alter column Insurance_No set not null;
ALTER TABLE
```

Alternative Database migration:

Relational databases are designed for staticschemas and consistent, reliable transaction.They are not designed to handle booming data volumes at top speed. NoSQL approach is towards: -

1.Real time data collection.
2.Big data storage.
3.Agile schemas.
4.Searchable catalogues of information.

Most universities store their consumption data in relational databases; the amount of data is huge, but the databases performance only enablesit to show the student consumption records from recent days.

The delay when performing big data analysis inrelational databases is very large.

This might be the big reason we need to shift fromSQL to NoSQL.

But after we have made the decision of change,we need to figure out which type of NoSQL we want as they have four main divisions with eachof their main examples mentioned below: -

1. Key Value
    - Example: Riak, Tokyo Cabinet,Redis Server, Memcached, Scalaris.

2. Document-Based
    - Example: MongoDB, CouchCB, OrientDB, RavenDB.

3. Column-Based
    - Example: BigTable, Cassandra, Hbase,Hypertable.

4. Graph-Based
    - Example: Neo4J, InfoGrid, Infinite Graph,Flock DB.

Each of this category has its unique attributes andlimitations. None ofthe above database is better to solve any problem.

In our case if we had to transition, we would useMongoDB as it’s the best fit for cases when we integrate hundreds of different data sources, with great read and write operations speed.

So, we will be using Document-Based NoSQL database.

These will be the different strategies we coulduse to change from SQL to NoSQL:

1. Rewrite: Writing the whole thing over

2. Redesign Schema: keep your business logic,rewrite your data layers, and totally redesign your schema with a NoSQL model.

3. Refactor First: Keep everything but refactoryour data logic and RDBMS schema into a NoSQLmodel.

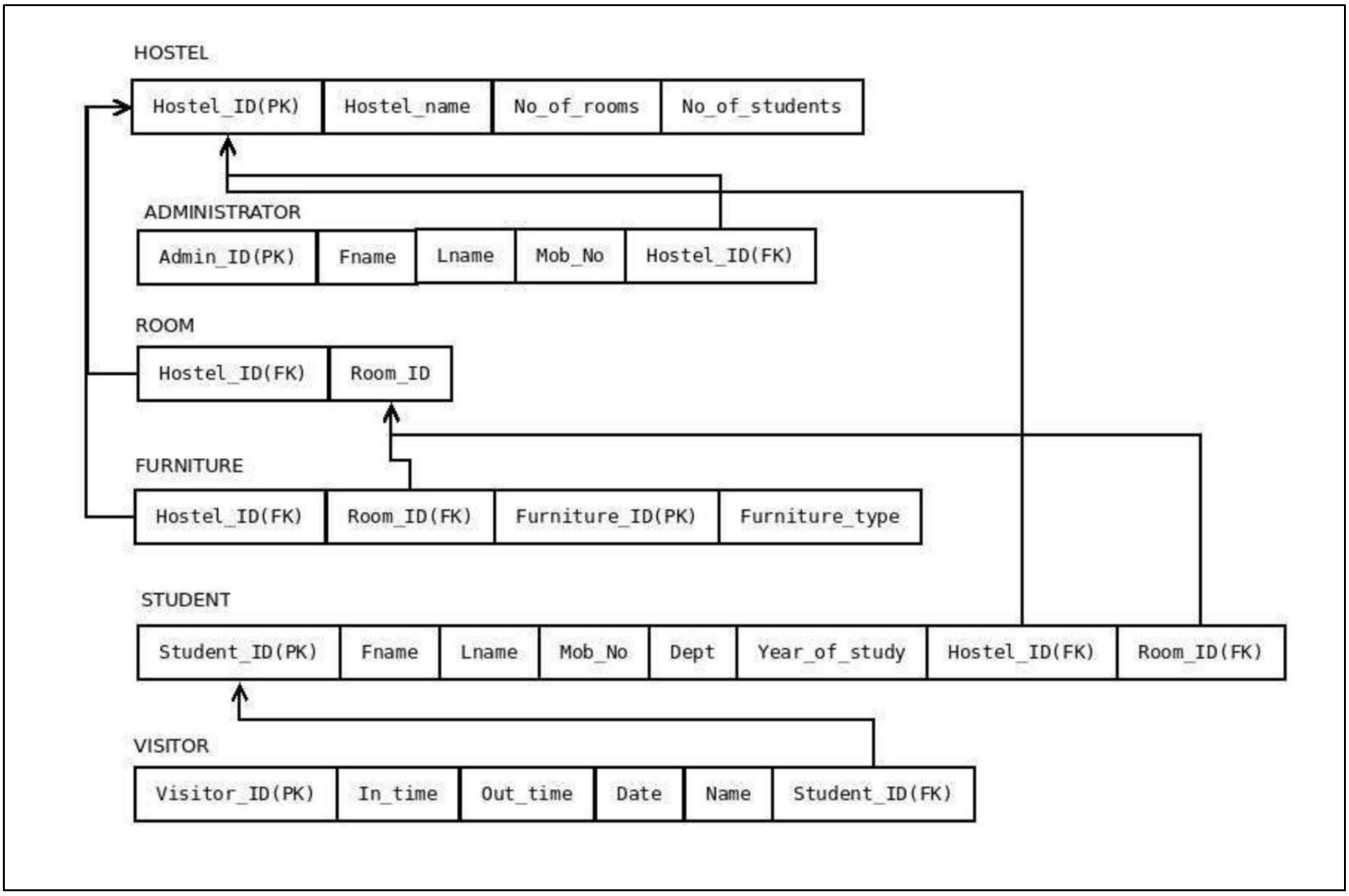
4. Optimize Later: Host your schema with asfew changes as possible.

Indicate contribution of each member clearly and approximate time spent in hours for the respectiveactivity.

Name	Contributions	Time spent
Traimbak Mahabal Bhat	Database migration and support	3 Hours
Vyshak R	Simple User interface design for front end, Additional queries	4 Hours
Suhan B Revankar	Report/Writeup	3 Hours

****

As a part of assignment 001 the relational schema was not submitted . Therefore , submitting here … please consider



HOSTEL

Hostel_ID(PK)	Hostel_name	No_of_rooms	No_of_students

ADMINISTRATOR

Admin_ID(PK)	Fname	Lname	Mob_No	Hostel_ID(FK)

ROOM

Hostel_ID(FK)	Room_ID

FURNITURE

Hostel_ID(FK)	Room_ID(FK)	Furniture_ID(PK)	Furniture_type

STUDENT

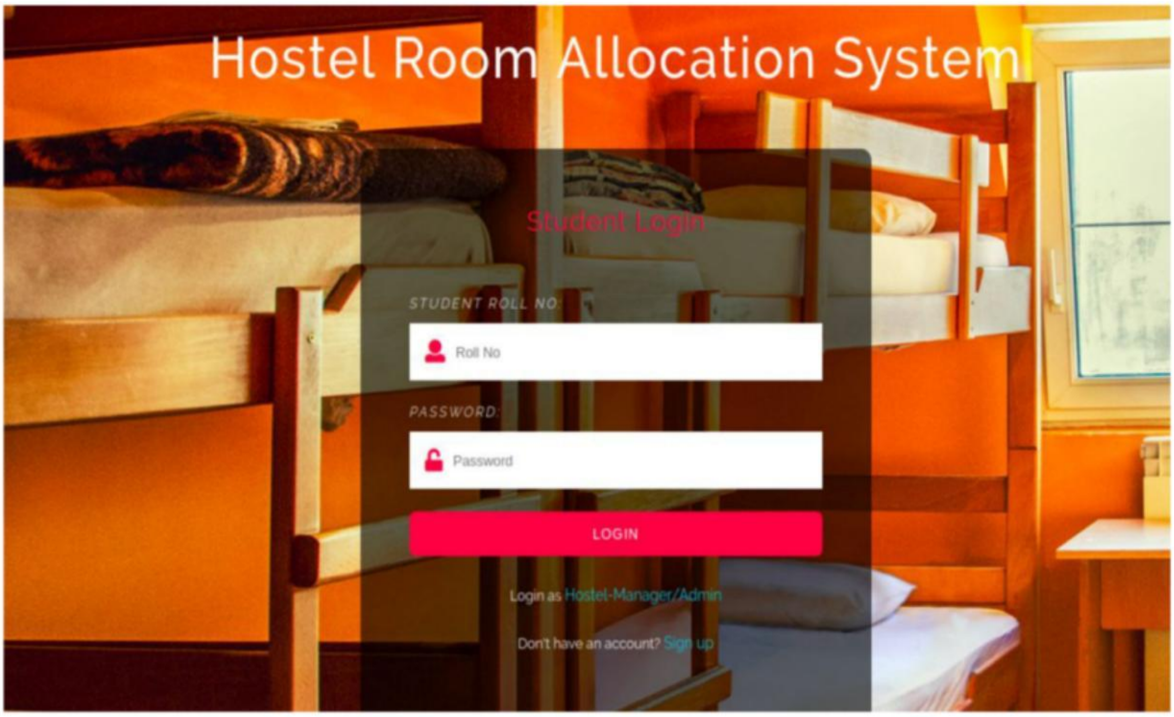
Student_ID(PK)	Fname	Lname	Mob_No	Dept	Year_of_study	Hostel_ID(FK)	Room_ID(FK)

VISITOR

Visitor_ID(PK)	In_time	Out_time	Date	Name	Student_ID(FK)

Additional screenshots of the front end made using php

1.Login Page


Hostel Room Allocation System
Student Login
STUDENT ROLL NO:
Roll No
PASSWORD:
Password
LOGIN
Login as Hostel-Manager/Admin
Don't have an account? Sign up

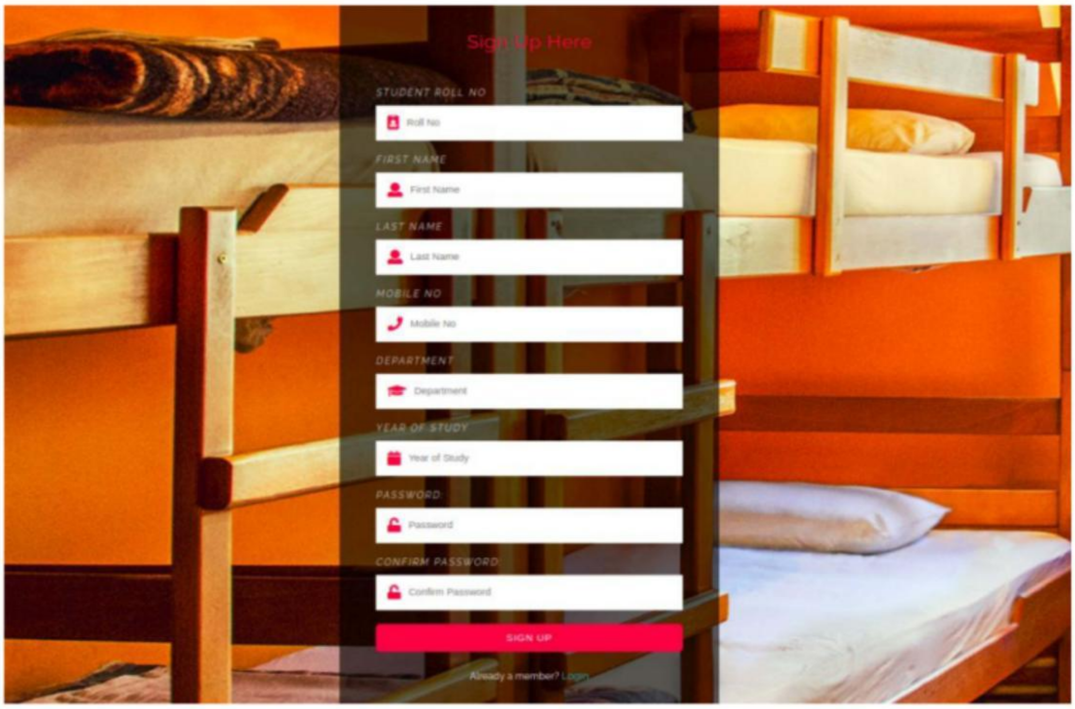
Users who have previously registered for the Hostel Room Allocation System Web Application must login by:

1. Entering their Username/Roll No.
2. Entering their Password.

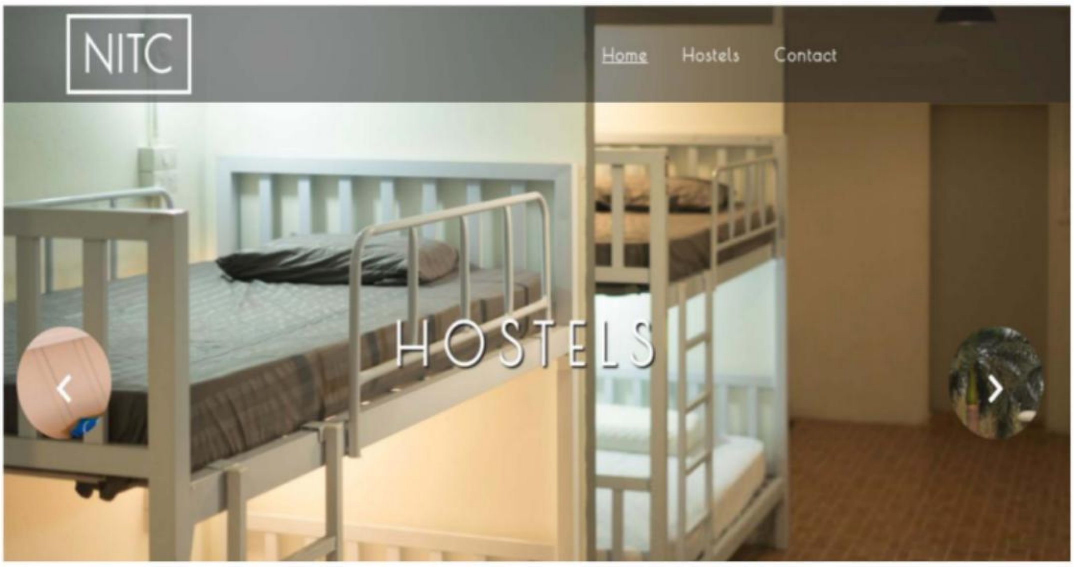
Selecting Login to advance to the next screen and begin using the application.

2. Sign Up Page

Students who have not previously registered for the Hostel Room Allocation System Web Application must select “Register to use the site” to access the “New User Registration” page. There they can enter all the required details and register themselves in the portal.


Sign Up Here
STUDENT ROLL NO
Roll No
FIRST NAME
First Name
LAST NAME
Last Name
MOBILE NO
Mobile No
DEPARTMENT
Department
YEAR OF STUDY
Year of Study
PASSWORD
Password
CONFIRM PASSWORD
Confirm Password
SIGN UP
Already a member? Login

3. Home Page

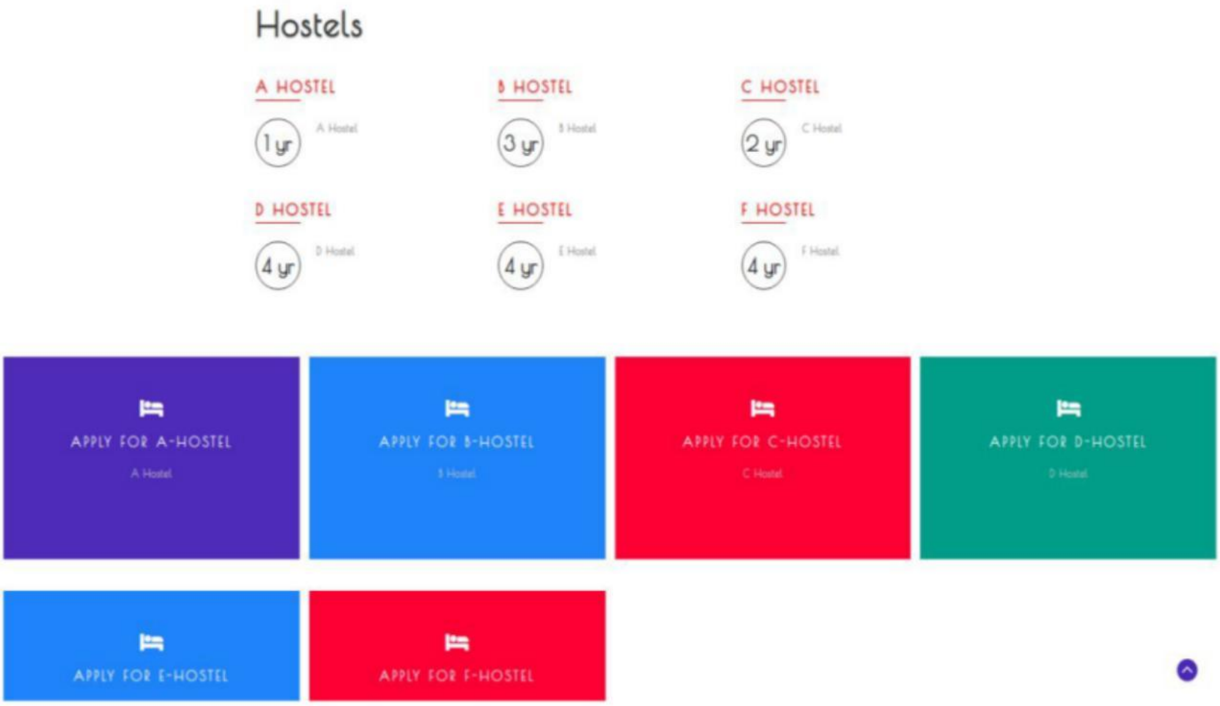

NITC
Home
Hostels
Contact
HOSTELS

There are 3 different home pages

1. User Homepage
2. Hostel Manager Homepage
3. Admin Homepage

The homepage basically cointains a navigation bar which takes the user to different pages to perform different activities.

4. Hostel Page

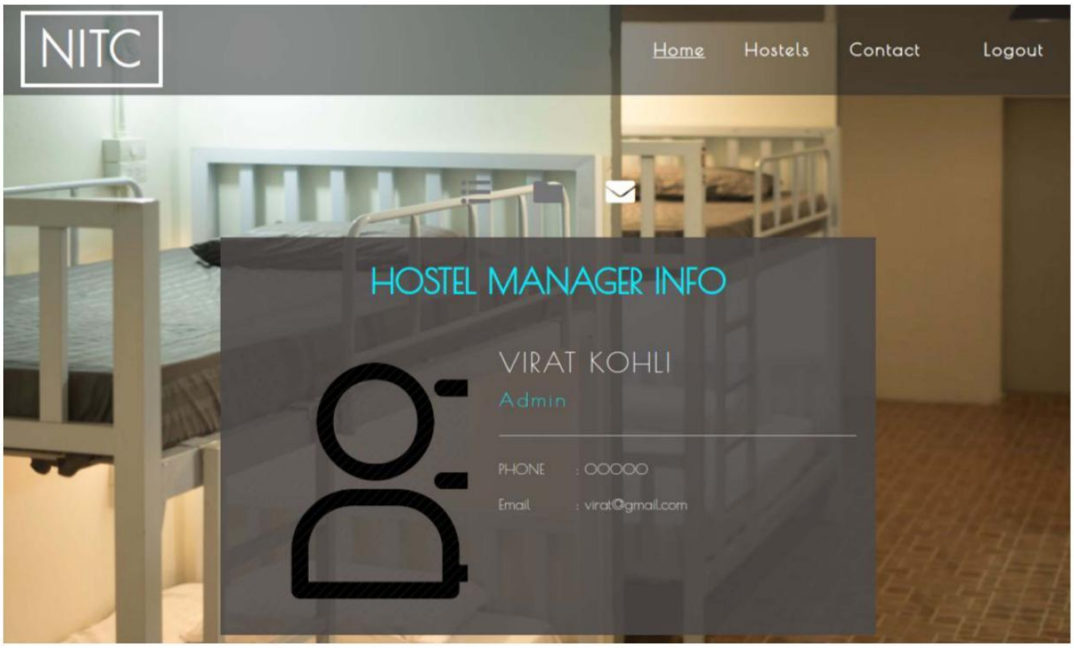


Hostel Page is accessible only to students and they can request for room in any hostel by just clicking on the corresponding hostel link and confirming by entering their password.

5. User Profile

This is again of 3 types

1. Student user profile
2. Hostel manager user profile
3. Admin user profile



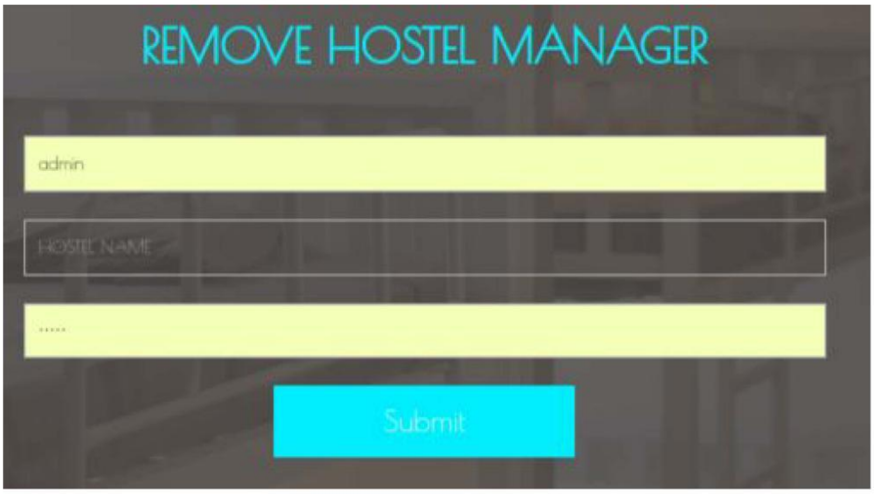
HOSTEL MANAGER INFO

VIRAT KOHLI

Admin

PHONE : 00000

Email : virat@gmail.com



REMOVE HOSTEL MANAGER

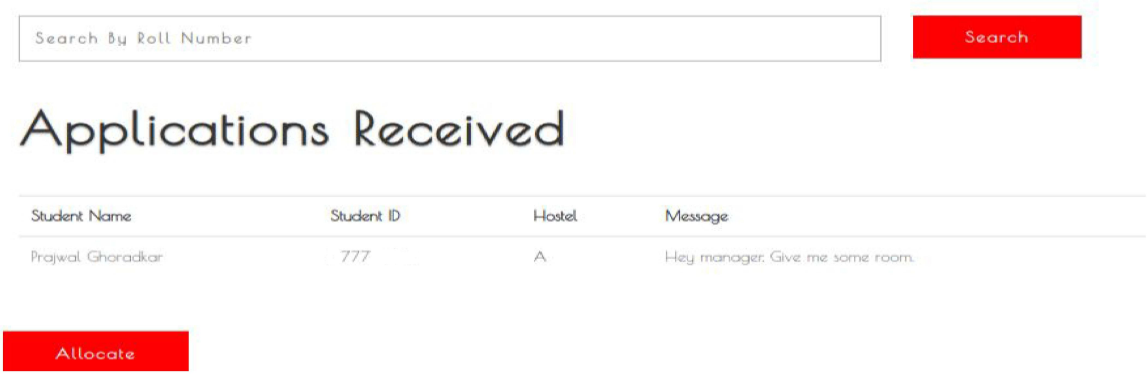
admin

HOSTEL NAME

·····

This page can only be accessed by hostel admin. He can allocate a new hostel admin or remove an existing hostel manager.

6. Applications Received

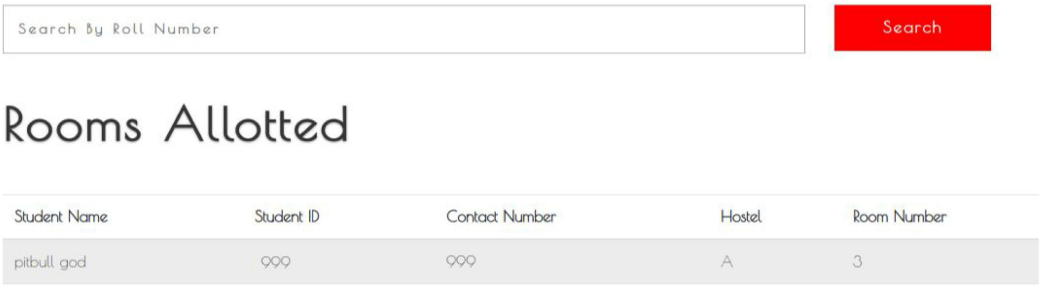


Applications Received

Student Name	Student ID	Hostel	Message
Prajwal Ghoradkar	777	A	Hey manager. Give me some room.

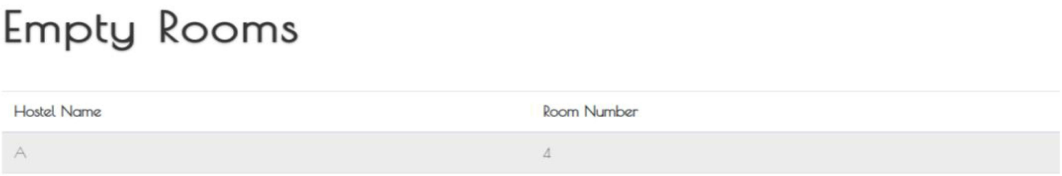
This tab can only be accessed by hostel manager. He can view all the applications his hostel received and allocate rooms accordingly.

7. Allocated and Empty Rooms



Rooms Allotted

Student Name	Student ID	Contact Number	Hostel	Room Number
pitbull god	999	999	A	3

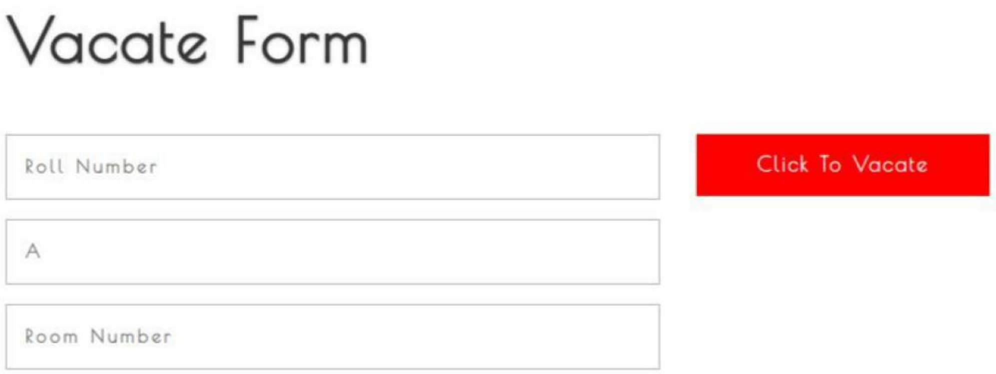


Empty Rooms

Hostel Name	Room Number
A	4

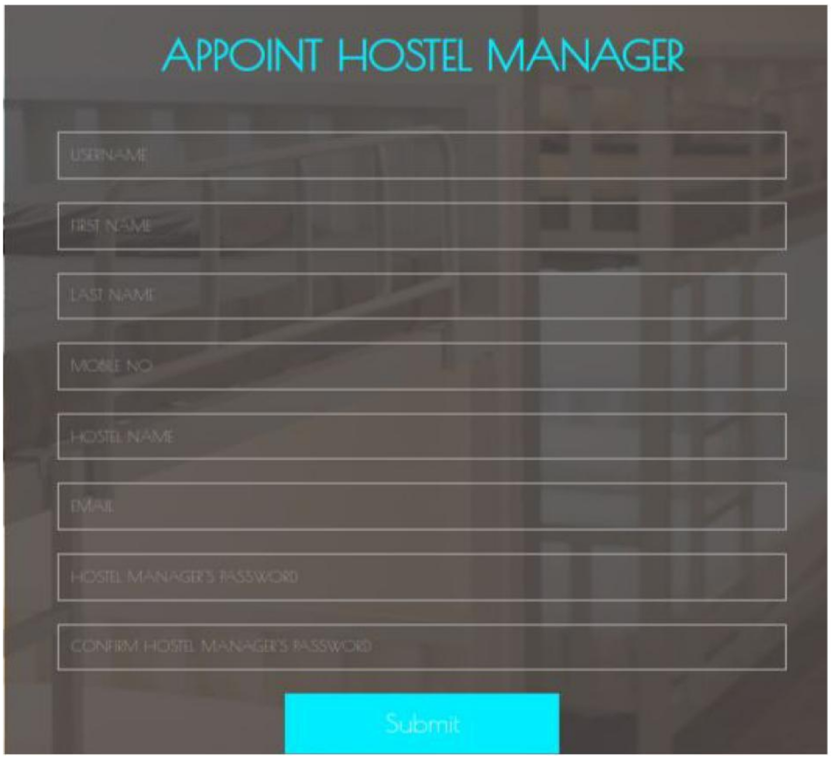
Each hostel manager can look at the list of allocated rooms and empty rooms in his/her hostel by going to this tab.

8. Vacate Rooms



Hostel Manager can vacate an allocated room by filling the student and room details in this form and clicking on “click to vacate” button.

9. Appoint/Remove Hostel Manager


APPOINT HOSTEL MANAGER
USERNAME
FIRST NAME
LAST NAME
MOBILE NO
HOSTEL NAME
EMAIL
HOSTEL MANAGER'S PASSWORD
CONFIRM HOSTEL MANAGER'S PASSWORD
Submit