# 

# 

# 

**Department of Computer Science and Engineering**

**University of Dhaka**

**Project Report**

CSE2211 - Database Management Systems-I

2nd Year 2nd Semester - 2018

**Project Title**

**Database for Hall Residents**

**Submitted By**

Name: Sree Sowmik Kumar Sarker

Registration No: 2016-314-445

Roll No: JN-054

[**Introduction**](#_4e4sl27pvhya) **3**

[**Descriptions**](#_kqtcinouo5du) **4**

[**Design Diagrams**](#_yhwjhq8tsdmn) **5**

[**Environment of Implementation**](#_ypw1ny5y80gz) **8**

[**Application of the Database**](#_in0rdu3rq18u) **9**

[**Conclusions and Discussions**](#_98sf1zpk74sf) **15**

# 

# **Introduction :**

In a Hall it contains enormous data. Every year a lot of students go out of the hall and are added to the hall. So it is very difficult to keep these accounts in the pen. Besides I see from different online source that there are huge amounts of data and that are easy to maintain. From that concept I wanted to do something like that.

The "Database for Hall Residents system" is an improved hall information service, increase information sharing and providing hall house teacher facilities. It can handle all details about a student and hall building, canteen,mess,library and computer lab. The details include student's all information such as students name,department name, contact information , seat allocation details,hometown etc and each buildings and room details ,mess,canteen and computer lab details and also their director details.

This whole database system is managed by each buildings house tutor. It is the job of the house tutor to insert,update and monitor the whole process.

So this database system will serve to reduce times for insertion or updation any data and can faster keep track of all the information of the hall.

I have learnt relational database management system(RDBMS). Which helps me to create relation between tables. RDBMS is very helpful to build a database on hall residents. So I planned to do it by RDBMS. Because student, teacher, building, room ,director etc. can be different tables which has relations between them like a student is allocated in a room and a room is in a building and each building has teacher or house tutor.

# **Descriptions :**

Objectives and Motivation:

Web applications are becoming so widespread is because it offers a user friendly interface for online access from everywhere of the human race. In addition, it helps to amplify users sustain rate. Also it is very difficult to keep all the information of hall in the pen. So this database will be a good source of information for hall residents.

Main Features:

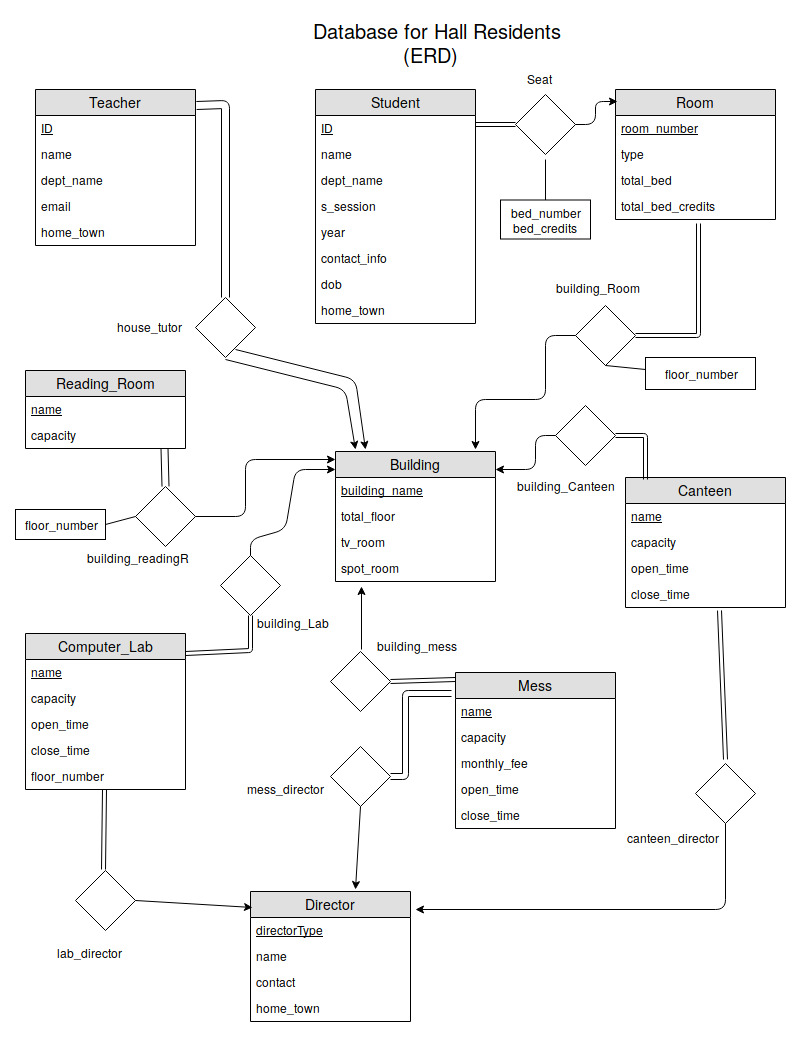
The main objective of the proposed of this database system is to computerize the existing system and reduce manpower and time consumption. It provides the following features.

* Detailed information about each student's academic information and seat details .
* Detailed information about each buildings and these buildings house tutor.
* All the information about the canteen , mess, computer lab and reading room and their director.
* House tutor can easily search any student's all information by student's id.
* Can find all the empty room and empty seat in each building easily and can insert or update students in the empty seat easily.
* Can easily find the total number of student's in the hall and total number of first, second, third and fourth year student's .
* Reduce error in data management.
* Easy and user-friendly database system and it reduce paper work and time consumption .

Necessity and Importance:

* The database will provided a huge collection of information in compact a compact environment.
* House tutor can get information doing a few queries only.
* These type of database can promote the student service and save the time, money and manpower.

# **Design Diagrams :**

Entity Relationship Diagram(ERD):

The main object of the hall database system is student, teacher, building,room etc. that are given in the above ER diagram. The relation between Student table and Room table is many to one because more than one student can stay in a room but for a student only one room is allocated . And all student's must have a room.

Same as, Room to Building , Teacher to Building, Reading room to Building , Canteen to Building, Mess to Building , Computer lab to Building and Director to Mess, Computer lab and Canteen relation is many to one.

At the same time, to identify a student uniquely each student's has an unique id. Same as, each teacher has unique id , each building has unique name, each room has unique number, each computer lab, mess, canteen has unique name and each director has unique director type.

In the building table each building has a unique name. Also all the buildings has some little information like if the building has any tv room or spot room if has then the room number and the total floor number in the building. So these are the attributes of the building table.

Also each building can have one or more canteen,mess,reading room and computer lab and each of them has a unique name also has a capacity and open time ,close time and director (except reading room because reading room is all time open in hall) , so these are the attributes of these tables.

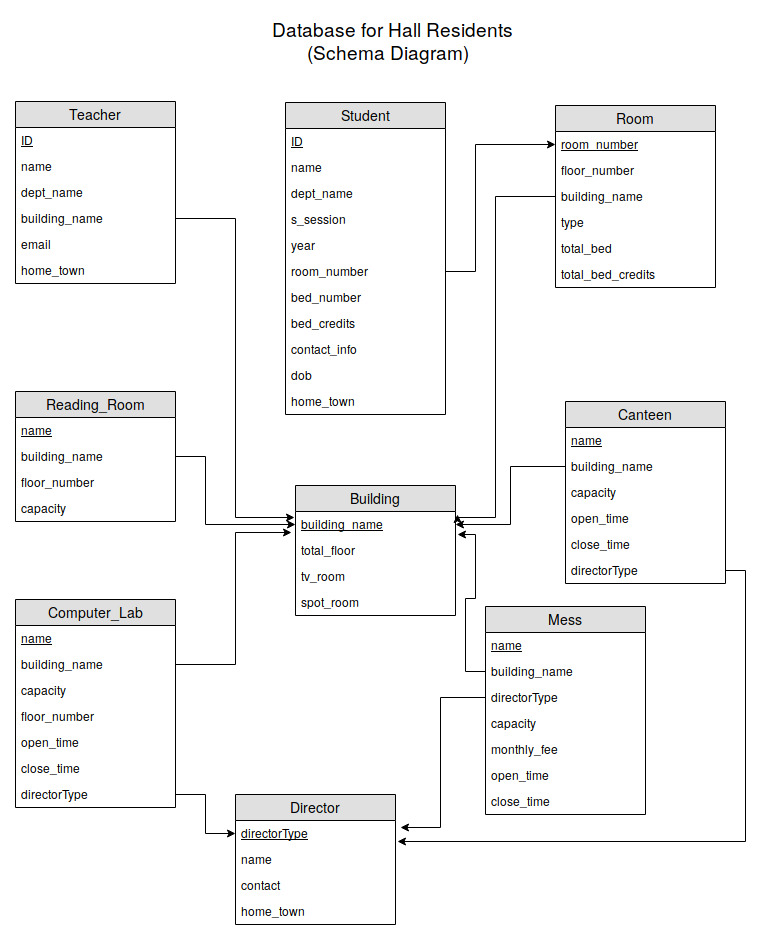
In the director table the attributes are their unique director type and name, contact information and home town.

In the teacher table its attributes are id and necessary details like name , department name, contact information , home town and the building name where he works.

In the student's table the attributes are student's unique id, name, department name, session , academic year, date of birth, hometown, contact information , room number , bed number and bed credits. Here bed credits for a 1st year student is 1 and for 2nd , 3rd and 4th year student's is 2,3 and 4 respectively.

For room information each room has a type like single ,double or multiple. The total number of bed in the room and total number of bed credits in the room. So these are the room table attributes.

Schema Diagram :



Most of the assumptions have been described with the ERD part. Now, a few are left to be mentioned. In the student table room number is a foreign key as it indicates the room number in the room table . Same as, the building name is a foreign key of teacher , room, computer lab, mess, canteen , reading room table. The attribute directortype is also a foreign key in mess, computer lab and canteen table.

# **Environment of Implementation :**

Overall I have felt much comfortable using oracle 11g. It’s really easy to use. Besides local hosts helped a lot. Very user friendly environment. SQL is much easier for oracle. I haven’t used other platforms because I don’t know much about those. But It seems oracle is much suitable and SQL is much easier on oracle.

All Oracle databases are backward compatible. This allows businesses to upgrade their systems without a complete overhaul of their database system. This provides efficient and low-cost updates. In addition, new versions of Oracle databases provide new features while keep the popular features from older versions. This ensures that their product is based on the customer's function rather than what is cost-effective for Oracle. But all these advantages are not available in MySQL or PostgreSQL for this reason I have chosen Oracle-11g.

# **Application of the Database :**

Application Scope: Show the room number and total empty seat in this room number.

SQL query:

select room\_number,total\_seat-present\_allocated empty\_seat

from (

select room\_number,count(id) present\_allocated

from student

where room\_number in (select room\_number from room)

group by room\_number

order by room\_number asc

)

natural join

(

select room\_number,case

when type='Single' then '1'

when type='Double' then '2'

when type='Multiple' then '8'

end total\_seat

from room

order by room\_number asc

)

order by empty\_seat desc;

Output:

# 

Application Scope: Find the total number of students of each home town in the hall.

SQL query:

select home\_town,count(id) student\_number

from student

where home\_town in (

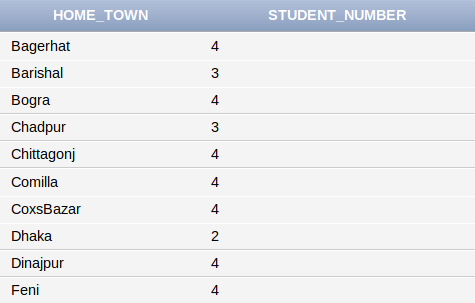
select home\_town from student

)

group by home\_town

order by home\_town asc;

Output:



Application Scope: Show which departments students is max in this hall.

SQL query:

select dept\_name,student\_number

from (

select dept\_name,count(id) student\_number

from student

where dept\_name in (

select dept\_name from student

)

group by dept\_name

order by dept\_name asc

)

where student\_number = (

Select max(student\_number)

from (

select dept\_name,count(id) student\_number

from student

where dept\_name in

(

select dept\_name from student

)

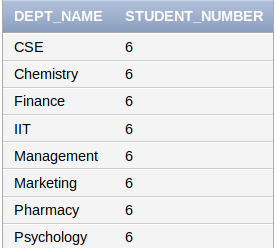
group by dept\_name

order by dept\_name asc

)

)

Output:



Application Scope: Which students stay in the single room.

SQL query:

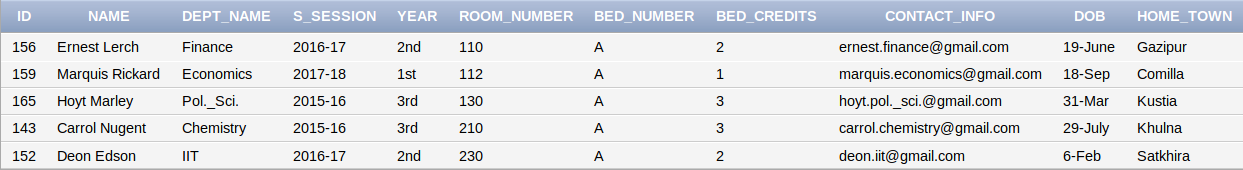
select \*

from student

where room\_number in( select room\_number from room where type = 'Single')

order by home\_town asc;

Output:



Application Scope: How many canteen has in each building.

SQL query:

select building\_name,count(name) total\_canteen

from canteen

where building\_name in

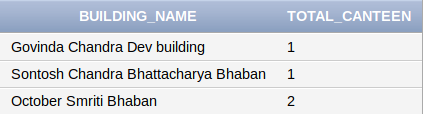
(

select building\_name from building

)

group by building\_name;

Output:



Application Scope: Determine the building name which has no tv room.

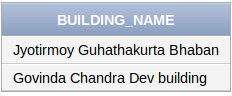
SQL query:

select building\_name

from building

where tv\_room='No';

Output:



Application Scope: Search If a departments house tutor exits or not.

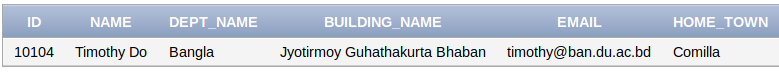
SQL query:

select \*

from teacher

where dept\_name='Bangla';

Output:



Application Scope: Search the building name of a teacher by id.

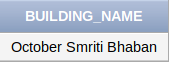
SQL query:

select building\_name

from teacher

where id=10101;

Output:



Application Scope: Search a building if there exits a canteen then determine if there exits tv\_room and spot\_room.

SQL query:

select tv\_room,spot\_room

from building

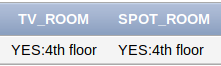
where building\_name =(

select building\_name

from canteen

where name='Sankar Canteen');

Output:



Application Scope: Show all the information about the director by knowing directortype.

SQL query:

select \*

from director

where directorType='C\_lab\_O';

Output:



# **Conclusions and Discussions :**

This project helped me to create real life database system. I have able to learn

creating relations between tables. I selected hall database because its database

is huge which helped me to insert dummy data easily. Besides there are many

things I should have added.

Limitations:

* Can’t store old students data (those who are completed 4 th year) and his seat will be counted as empty seat.
* Rooms / Seats are alocated only for 1 st year to 4 th year students .
* Students credits will not increases every year .
* Teacher can not insert student's more than empty seat.
* Can’t insert student more than two in one bed in a room.

Possibilities:

* If any seat is empty teacher can add as much as student.
* Update , insert or delete any data in this database.

Discussion:

Great skills have been achieved during the development of this project. I have faced a lot of problems firstly. But I managed it successfully. On my course I learned many theoretical comprehensions. Using that knowledge and Observing live operational system. My project is a fundamental approach of these. I develop the project “Database for Hall Residents”.