NATIONAL INSTITUTE OF TECHNOLOGY ANDHRA PRADESH

Department of Computer Science and Engineering



HOSTEL MANAGEMENT SYSTEM

SUBMITTED

BY

Nadipalli Jaswanth 422224

Vanguri Michael 422268

Kushal C 422204

2023-24

NATIONAL INSTITUTE OF TECHNOLOGY ANDHRA PRADESH

Department of Computer Science and Engineering



Certificate

This is to certify that the Mini project entitled, "HOSTEL MANAGEMENT SYSTEM" has been done by NADIPALLI JASWANTH (422224), VANGURI MICHAEL (422268) and KUSHAL C (422204) of Bachelor of Techonolgy, Computer Science and Engineering during semester IV from National Institute of Technology Andhra Pradesh towards DBMS Lab.

Mr.D.Prasad

Faculty

CSE Department

NIT,AndhraPradesh

INDEX PAGE

|  |  |  |
| --- | --- | --- |
| S.No | Topic | Page No |
| 1 | ABSTRACT | 4 |
| 2 | INTRODUCTION | 5 |
| 3 | EXISTING SYSTEM AND DISADVANTAGES | 6 |
| 4 | PROPOSED SYSTEM AND ADVANTAGES | 7 |
| 5 | E-R MODEL | 8 |
| 6 | RELATIONAL SCHEMA | 9 |
| 7 | SCHEMA REFINEMENT(NORMALIZTION) | 10 |
| 8 | SQL | 11 |
| 9 | INTEGRITY CONSTRAINTS | 15 |
| 10 | OUTPUT SCREENSHOTS | 16 |
| 11 | CONCLUSION | 18 |

**ABSTRACT**

Welcome to the presentation of our Database Management System (DBMS) project, focusing on the development of Hostel Registration and Management System. Our project aims to streamline the management of hostel accommodations within our institute, offering a centralized platform for student enrollment, allocation of accommodations, and facility management. By integrating essential functionalities such as student registration, room allocation, and facility tracking, our system enhances efficiency and transparency in hostel operations. Leveraging SQL for database management, we ensure a user-friendly experience for both administrators and students. Through this project, we showcase the power of DBMS in optimizing hostel management processes, improving student satisfaction, and facilitating administrative operations.

**INTRODUCTION**

Welcome to the presentation of our Hostel Registration Management System. Our system is designed to revolutionize the way hostel accommodations are managed for students within our institute. By providing a centralized platform for handling student enrollment details, accommodation allocation, and facility management, we aim to optimize the efficiency, accuracy, and transparency of the entire process. Through the implementation of robust database management techniques utilizing SQL , our system ensures seamless integration and user-friendly interaction for both administrators and students alike. Furthermore, our project addresses the evolving needs of our institute's student population and administrative staff, contributing to a more efficient and effective hostel management system. We'll explore the key functionalities, significance, and technological aspects of our Hostel Registration Management System throughout this presentation, highlighting its potential to enhance the overall administrative operations.

**EXISTING SYSTEM AND DISADVANTAGES**

**In the existing hostel management system, various tasks such as student registration, room allocation, and hostel facility management are often managed manually or using fragmented databases. Each hostel or department may maintain its own records, leading to data redundancy and inconsistency. Access to these databases may be limited, and administrative tasks may require significant manual effort.**

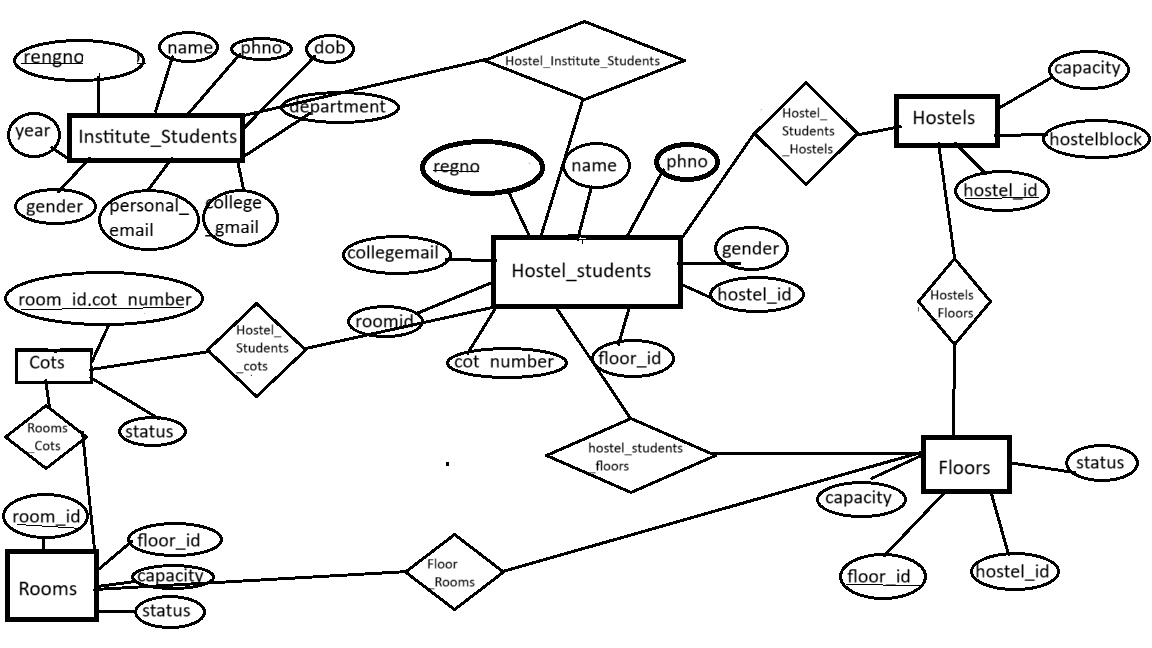
1. **Fragmented Data Management:** Hostel data may be stored in separate databases or spreadsheets, leading to fragmentation and difficulty in maintaining a unified view of student accommodation.
2. **Limited Accessibility:** Access to hostel data may be restricted to specific staff members, hindering collaboration and timely decision-making.
3. **Manual Processes:** Tasks such as student room allocation, fee collection, and inventory management may rely on manual processes, increasing the likelihood of errors and delays.
4. **Lack of Integration:** Hostel management systems may operate independently of other institutional systems, such as student information systems or accounting software, resulting in data silos and inefficiencies.

**PROPOSED SYSTEM AND ADVANTAGES**

**In the existing hostel management system, various tasks such as student registration, room allocation, and hostel facility management are often managed manually or using fragmented databases. Each hostel or department may maintain its own records, leading to data redundancy and inconsistency. Access to these databases may be limited, and administrative tasks may require significant manual effort.**

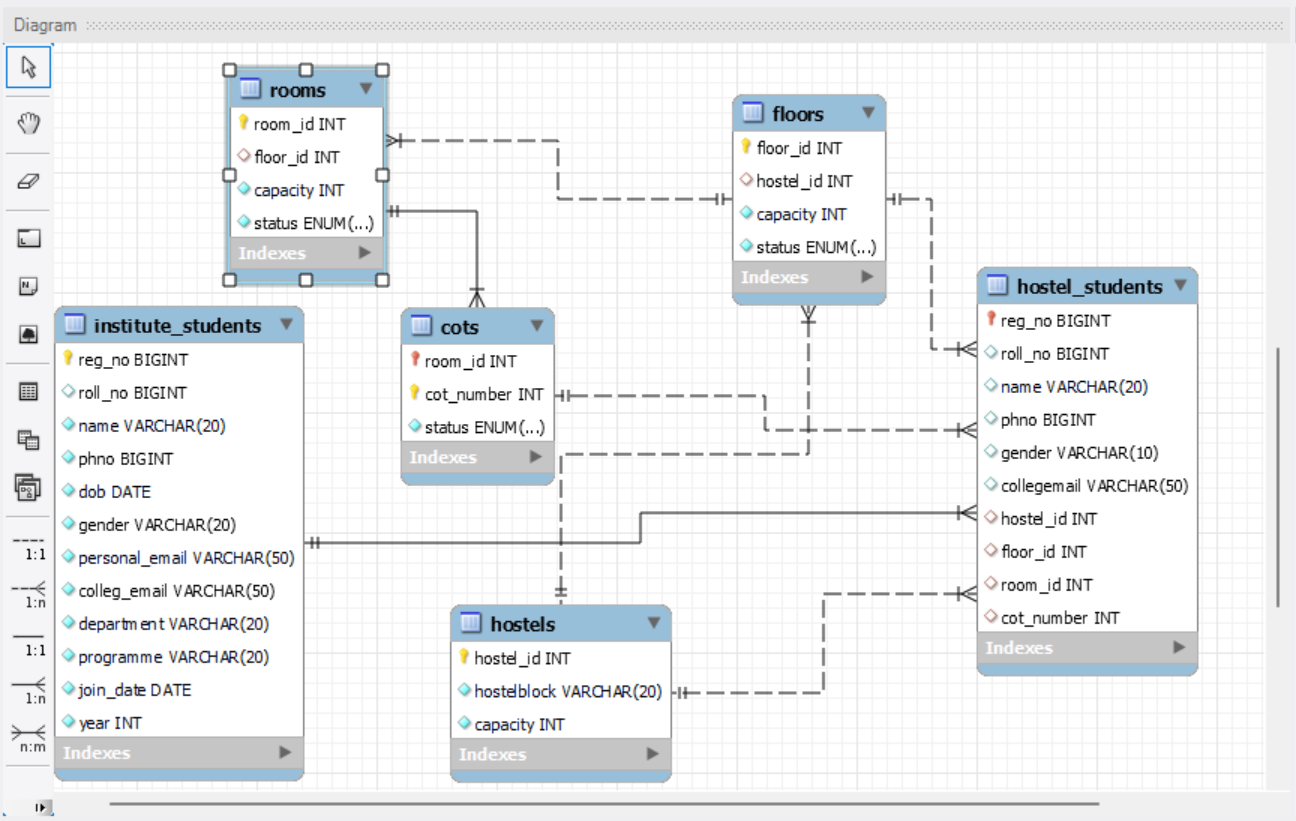
1. **Centralized Hostel Management:** The proposed system centralizes hostel management operations, consolidating all hostel-related data into a single, cohesive platform. This eliminates data fragmentation and ensures consistent and accurate information across the hostel.
2. **Enhanced Accessibility for Stakeholders:** The system enhances accessibility by providing authorized stakeholders, including hostel administrators, staff, students, and parents, with secure access to hostel-related information. Through web-based or mobile interfaces, stakeholders can conveniently access and update data from anywhere, at any time, promoting transparency and collaboration.
3. **Automation of Hostel Processes**: Leveraging automation, the system streamlines hostel processes such as room allocation, attendance tracking, and maintenance requests. By automating routine tasks, administrative burdens are reduced, and operational efficiency is improved, leading to faster response times and enhanced service delivery.
4. Integration with Institutional Systems: The system offers integration capabilities with existing institutional systems, such as student information systems and accounting software.

E-R MODEL



**RELATIONAL SCHEMA**

|  |  |
| --- | --- |
| Entity | Attributes |
| Institute\_Students | reg\_no,roll\_no,name,phno,dob,gender,personal\_email,college\_email,department,programme,  year,join\_date |
| Hostel\_Students | reg\_no,roll\_no,name,phno, gender, collegeemail,hostel\_id,floor\_id,room\_id,cot\_number |
| Hostels | hostel\_id,hostelblock,capacity,gender |
| Floors | hostel\_id,floor\_id ,capacity,status |
| Rooms | room\_id,floor\_id ,capacity,gender |
| Cots | room\_id,cot\_number ,capacity,gender |



**SCHEMA REFINEMENT (NORMALIZATION)**

* **1. Institute\_Students: It has a primary key (reg\_no) and there are no partial dependencies or transitive dependencies, so it satisfies 1NF , 2NF and 3NF.**
* **2. Hostel\_Students : It has a primary key (reg\_no) and there are no partial dependencies or transitive dependencies, so it satisfies 1NF , 2NF and 3NF.**
* **3. Hostels: : It has a primary key (hostel\_id) and there are no partial dependencies or transitive dependencies, so it satisfies 1NF , 2NF and 3NF.**
* **4. Floors: It has a primary key (floor\_id) and there are no partial dependencies or transitive dependencies, so it satisfies 1NF , 2NF and 3NF.**
* **5. Rooms: It has a primary key (room\_id) and there are no partial dependencies or transitive dependencies, so it satisfies 1NF , 2NF and 3NF..**
* **6. Cots: It has a primary key room\_id, cot\_number) and there are no partial dependencies or transitive dependencies, so it satisfies 1NF , 2NF and 3NF..**
* **All tables (Institute\_Students, Hostel\_Students, Hostels, Floors, Rooms, Cots) satisfy 1NF, 2NF, and 3NF.**
* **Data Redundancy : Maintained data redundancy by removing repetitive columns in cots table and making more tables**

**IMPLEMENTATION OF DATABASE (SQL):**

**Table Creation:**

Institute\_Students Table:

create table Institute\_Students(

reg\_no bigint PRIMARY KEY,

roll\_no bigint,

name varchar(20) not null,

phno bigint not null,

dob date not null,

gender varchar(20) not null,

personal\_email varchar(50) not null,

colleg\_email varchar(50) not null,

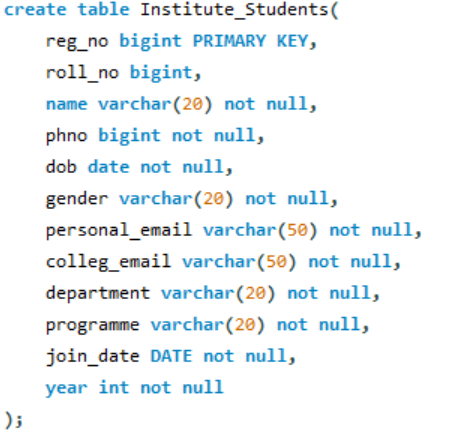
department varchar(20) not null,

programme varchar(20) not null,

join\_date DATE not null,

year int not null

);



Hostel\_Students Table:

CREATE TABLE Hostel\_Students (

reg\_no bigint PRIMARY KEY,

roll\_no bigint,

name VARCHAR(20),

phno BIGINT,

gender VARCHAR(10),

collegemail VARCHAR(50),

hostel\_id INT,

floor\_id INT,

room\_id INT,

cot\_number INT,

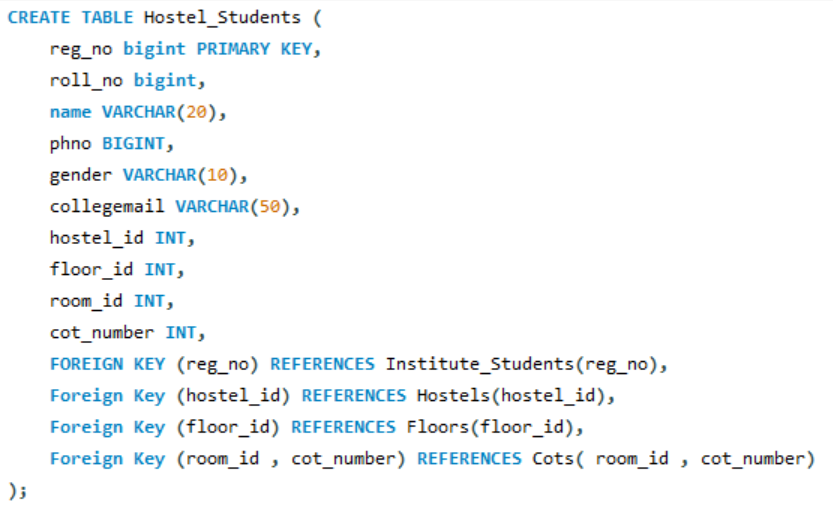
FOREIGN KEY (reg\_no) REFERENCES Institute\_Students(reg\_no),

Foreign Key (hostel\_id) REFERENCES Hostels(hostel\_id),

Foreign Key (floor\_id) REFERENCES Floors(floor\_id),

Foreign Key (room\_id , cot\_number) REFERENCES Cots( room\_id , cot\_number)

);



Hostels Table:

create table Hostels(

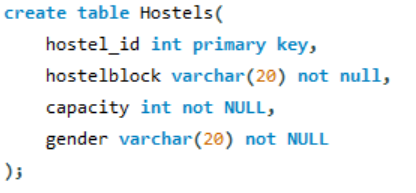
hostel\_id int primary key,

hostelblock varchar(20) not null,

capacity int not NULL,

gender varchar(20) not NULL

);



Floors Table:

CREATE TABLE Floors (

floor\_id INT PRIMARY KEY,

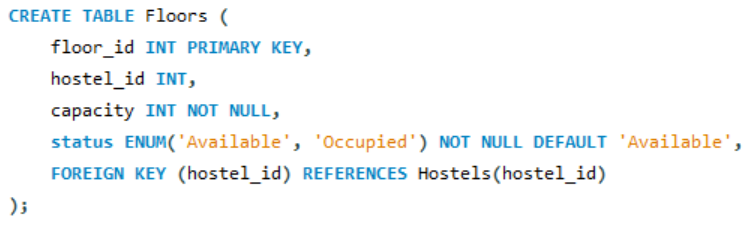
hostel\_id INT,

capacity INT NOT NULL,

status ENUM('Available', 'Occupied') NOT NULL DEFAULT 'Available',

FOREIGN KEY (hostel\_id) REFERENCES Hostels(hostel\_id)

);



Rooms Table:

CREATE TABLE Rooms (

room\_id INT PRIMARY KEY,

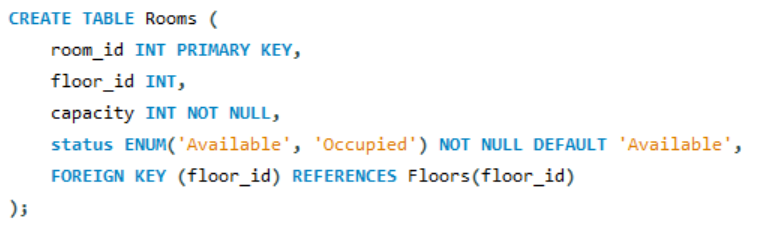
floor\_id INT,

capacity INT NOT NULL,

status ENUM('Available', 'Occupied') NOT NULL DEFAULT 'Available',

FOREIGN KEY (floor\_id) REFERENCES Floors(floor\_id)

);



Cots Table:

CREATE TABLE Cots (

room\_id INT,

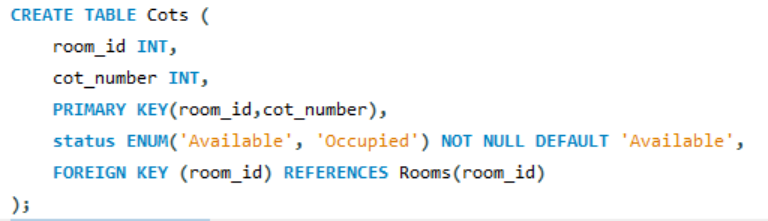
cot\_number INT,

PRIMARY KEY(room\_id,cot\_number),

status ENUM('Available', 'Occupied') NOT NULL DEFAULT 'Available',

FOREIGN KEY (room\_id) REFERENCES Rooms(room\_id)

);



**Integrity constraints:**

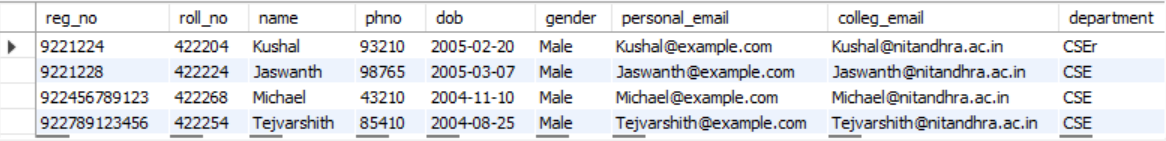
* *Primary Key Constraint*: Ensures that each record in a table has a unique identifier.
  + Example: In Hostels table , Primary Key constraint is defined on the “hostel\_id” column.
* *Foreign Key Constraint:* Enforces referential integrity by ensuring that values in a column (or columns) of one table match values in another table's primary key or unique key.
  + Example :
  + Foreign key constraints in :Hostel\_Students references :
  + ‘reg\_no’ to Institue\_students(‘reg\_no’ ),
    - ‘hostel\_id’ to Hostels(‘hostel\_id’),

‘floor\_id’ to Floors(‘floor\_id’),

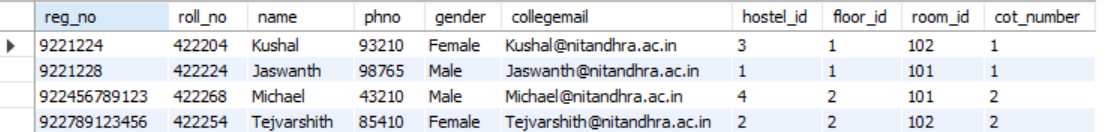
* + ‘room\_id’ and ‘cot\_number’ to Cots(‘room\_id’ , ‘cot\_number’)

**OUTPUT SCREENSHOTS**

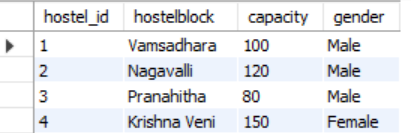
Institute\_Students Table:



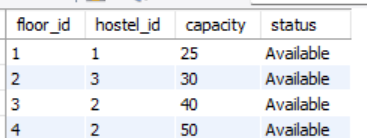
Hostel\_Students Table:



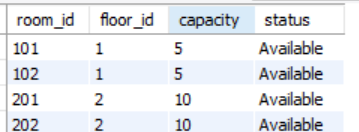
Hostels Table:



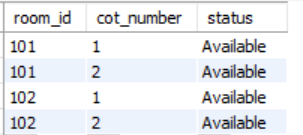
Floors Table:



Rooms Table:



Cots Table:



**CONCLUSION**

In conclusion, the hostel management system marks a transformative step forward in hostel administration and accommodation services. With its intuitive interface and extensive features, the system facilitates smooth interactions and efficient management of hostel facilities. While the system has already delivered substantial advantages, continual refinement and evolution remain imperative. Future enhancements could encompass advanced analytics, seamless integration with external platforms, and scalability to accommodate growing needs. Ultimately, this system stands as a cornerstone for optimizing hostel operations, enhancing resident experiences, and ensuring the seamless functioning of hostel facilities.