HOSTEL MANAGEMENT SYSTEM

Database & Application Architecture

Project Documentation

Source code available on

Contents

Abstract

The Hostel Management System is a project developed to simplify and streamline the tasks involved in managing a hostel. Built with a user-friendly interface using Python and supported by a PostgreSQL database, this system enables hostel administrators to handle student registrations, assign rooms, track fee payments, and generate useful reports.

The goal of the project is to provide a practical solution that replaces manual record-keeping with a digital system that is efficient, secure, and easy to use. Key features include secure login, organized student and room management, and an overview of payments, all designed to help administrators manage hostel operations smoothly.

System Requirements

Component	Specification
Hardware	Standard desktop or laptop (Min: 4GB RAM, 5GB Free Disk Space)
Operating System	Windows 10 or 11 / Ubuntu 22.04
Database	PostgreSQL (v16+)
Language	Python (v3.12.10)
Libraries	`tkinter`, `datetime`, `psycopg2`

Table 1: Hardware and Software Requirements

Database Design

The PostgreSQL database is structured with several tables, including a Student table for storing personal details and assigned room numbers, a Room table that maintains room availability and capacity, and a Payment table for tracking fee payments. An additional User table stores credentials for role-based access to ensure secure login. This relational database design supports complex data operations while maintaining data integrity and accessibility.
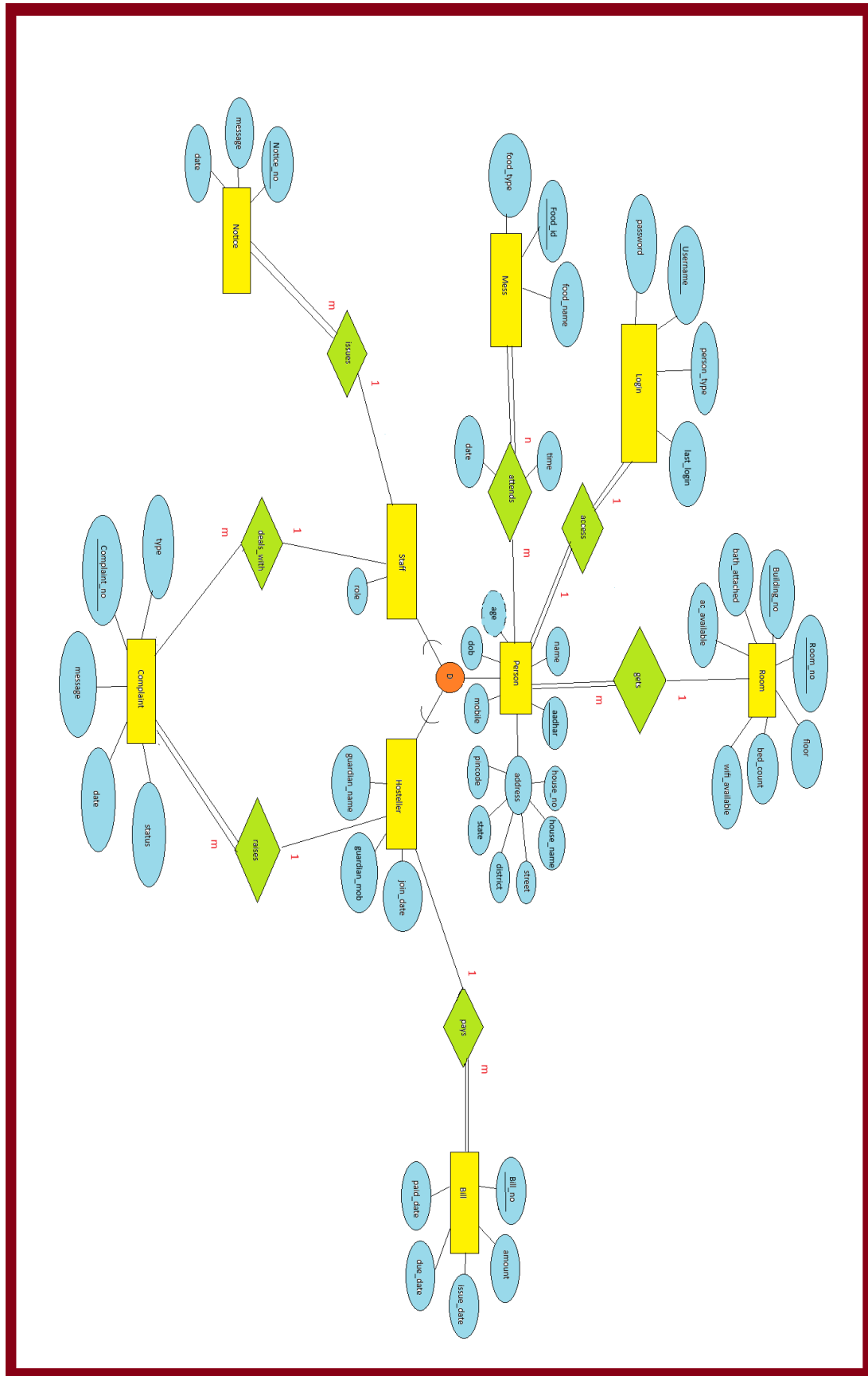
Figure 1: Entity Relationship Diagram (ERD)

Normalized Schemas

The database is normalized to ensure data integrity. The schema definitions are as follows:

1. **Person** (aadhar, name, age, dob, mobile, house_no, house_name, street, pincode)

2. **Pincode** (pincode, district)

3. **District** (district, state)

4. **State** (pincode, state)

5. **Hosteller** (aadhar, guardian_mob, join_date)

6. **Guardian** (guardian_mob, guardian_name)

7. **Staff** (aadhar, role)

8. **Room** (Room_no, building_no, floor, bed_count, wifi_available, bath_attached, ac_available)

9. **Gets** (aadhar, room_no, building_no)

10. **Bill** (Bill_no, amount, issue_date, paid_date)

11. **Bill_Issue** (issue_date, due_date)

12. **Pays** (Bill_no, aadhar)

13. **Complaint** (Complaint_no, Date, Status, Message, type)

14. **Raises** (Complaint_no, aadhar)

15. **Deals_with** (complaint_no, aadhar)

16. **Mess** (Food_id, Food_name)

17. **Food** (Food_name, Food_type)

18. **Attends** (aadhar, Food_id, Date, time)

19. **Notice** (notice_no, message, date)

20. **Issues** (notice_no, aadhar)

21. **Login** (username, person_type, password, last_login)

22. **Access** (aadhar, username)

GUI Design

The Tkinter-based GUI is organized into different windows for various operations. A login screen appears first, guiding users to the dashboard after successful authentication. The main dashboard provides options for managing students, rooms, payments, and reports.

Forms are designed for easy data entry, updating, and deletion, with clear prompts and buttons for navigation. Separate windows for report generation display summarized data in tables and offer options for exporting if needed. Overall, the interface is intuitive, guiding users through tasks with minimal learning curve.
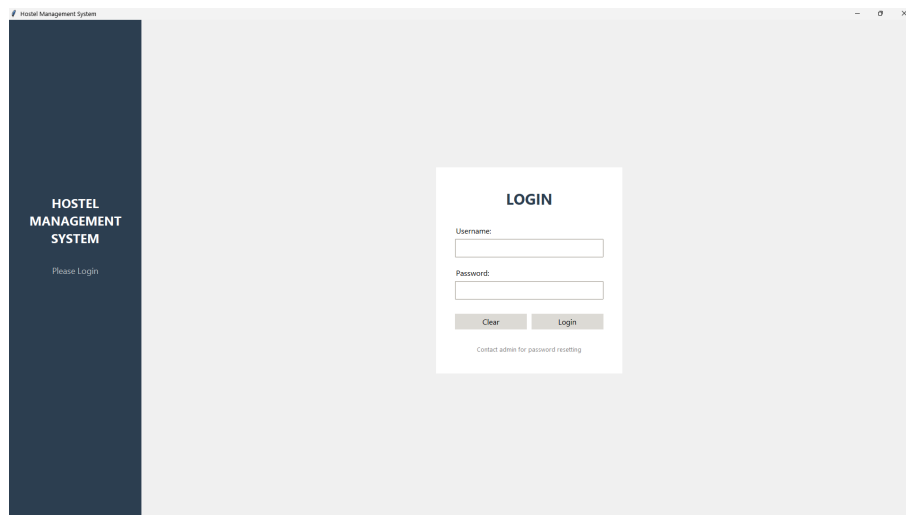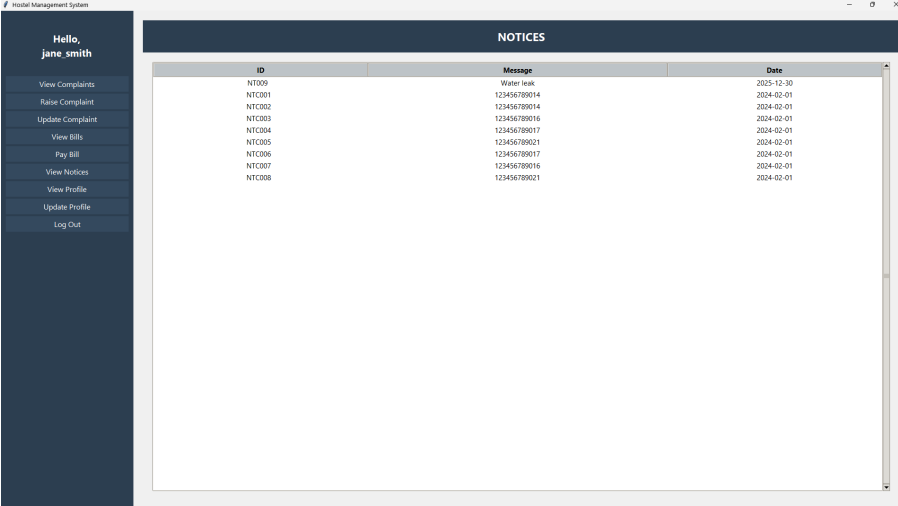

Figure 2: Login Page
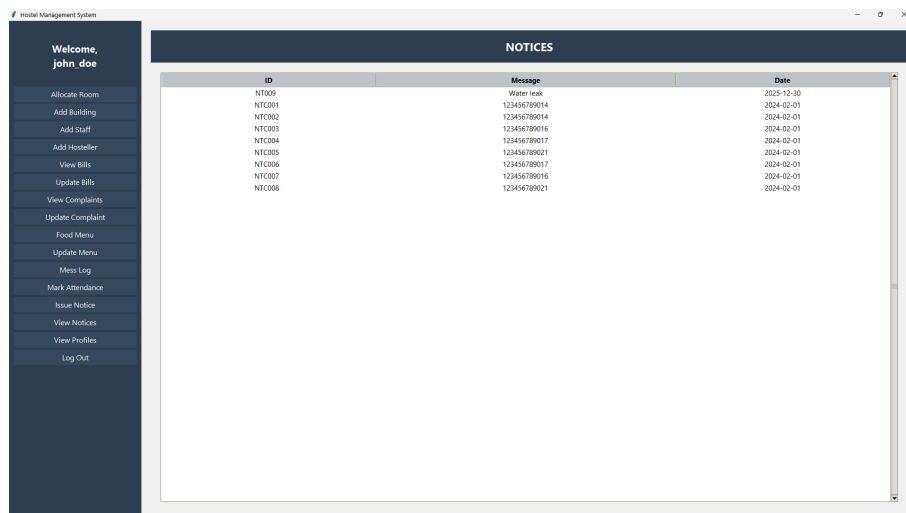
Figure 3: Staff Management Panel

Figure 4: Hosteller Management Panel

Conclusion

In conclusion, this Hostel Management System provides an effective solution for hostel administration by automating core tasks, reducing paperwork, and enhancing data accuracy.

In the future, this system could be enhanced with additional features such as automated email or SMS notifications to remind residents of pending payments and a web-based interface for remote access by administrators and students. Incorporating analytics dashboards to visualize trends in occupancy and revenue could also add value for hostel management, enabling better data-driven decision-making.