**HOSTEL ROOM ALLOCATION SYSTEM**

**Abstract :**

Managing hostel room allotment manually can be time-consuming and stressful for both students and administrators. Our Hostel Room Allocation System is designed to make the process smooth, fair, and hassle-free. This system allows students to register, view available rooms, and get allocated automatically or manually by the admin. It ensures that rooms are assigned efficiently based on availability and predefined rules, reducing conflicts and confusion. Students can also request room changes and report maintenance issues through the platform. For hostel administrators, the system provides an easy-to-use dashboard to manage room allotments, track vacancies. It eliminates paperwork, minimizes errors, and ensures a transparent process for everyone involved. By automating the hostel allocation process, this project aims to save time, reduce stress, and improve hostel management efficiency.

**Submitted by: Project Guide:**

**Batch – 11 CSE 2-2 Mrs. B. Sangeetha**

1. **G. Dharani 23SS1A0520 Assistant Professor**
2. **P. Komali Kavya 23SS1A0548 CSE Department**
3. **G. Anjali 23SS1A0518**
4. **K. Poojitha 23SS1A0531**

**Algorithms**

First-Come, First-Serve (FCFS) Algorithm

* The first student to register gets allocated a room based on availability.
* Simple and easy to implement but does not consider preferences.

Priority-Based Allocation

* Assigns rooms based on priority factors like seniority (year of study), merit, or special needs.
* Example: Final-year students get single rooms, while freshers are placed in shared rooms.

Randomized Allocation

* If students don’t have specific preferences, rooms are assigned randomly from available ones.
* Ensures fairness but may lead to dissatisfaction.

Preference-Based Matching (Stable Matching Algorithm / Gale-Shapley)

* Ensures students get rooms that best match their choices.

Graph-Based Allocation (Bipartite Matching Algorithm)

* Treats students and rooms as two sets and finds the best allocation using graph theory.
* Useful when there are constraints like room size limits.

Waitlist & Reallocation Algorithm

* If a student cancels their room, the system reallocates it to the next person on the waitlist.

**Software Requirements:**

1. Operating System: Windows / Linux
2. Frontend: HTML, CSS, JavaScript
3. Backend: Node.js / Python (Flask)
4. Database: MySQL / PostgreSQL
5. Authentication: Email Login, Admission Number
6. IDE: VS Code

**Hardware Requirements:**

1. Processor: Minimum Intel Core i3 (or equivalent)
2. RAM: At least 4GB (8GB recommended for smooth performance)
3. Storage: Minimum 20GB free space (for database and application files)
4. Internet Connection: Required for cloud-based hosting and authentication services
5. Server (for hosting): Firebase (if deploying online)