**📝 Project Proposal: AI-Powered Hostel Manager**

**📌 Project Title:**

**AI-Powered Hostel Manager**

**📖 Project Overview:**

The **AI-Powered Hostel Manager** is a web-based application that automates and enhances hostel management tasks using machine learning. It assists in room allocation, resource management, complaint prioritization, and mess planning while enabling hostel staff to manually oversee and intervene when necessary. The system aims to balance automation (AI features) with manual operations (admin panel, CRUD management) to improve overall efficiency.

**🎯 Project Objectives:**

* Automate repetitive and predictive tasks using ML models.
* Simplify and organize hostel operations (room allotment, complaints, mess).
* Detect anomalies in resource usage to prevent waste.
* Maintain transparency and ease for both students and hostel administrators.
* Reduce food wastage and utility misuse through data-based predictions.

**🔍 Key Features:**

**✅ Manual/Web Features (50% effort):**

1. **Student Registration & Profile Management**
   * New student onboarding with personal and academic details.
   * Upload documents (e.g., ID, fee receipts).
2. **Room Allocation Panel**
   * View available rooms and manually assign or override AI suggestion.
   * Occupancy status per room/floor.
3. **Mess Management**
   * Daily/weekly mess attendance confirmation.
   * Menu planner, feedback submission, and mess bill calculator.
4. **Complaint Management System**
   * Students submit maintenance or hygiene complaints.
   * Staff view, filter, update complaint status.
5. **Admin Dashboard**
   * Monitor students, rooms, complaints, bills, mess attendance.
   * Export reports, filter logs, manually edit records.
6. **Notifications & Alerts**
   * Alerts for overdue fees, high utility usage, pending complaints, etc.

**🤖 AI-Powered Features (50% effort):**

1. **Room Recommendation Model** *(Classification)*
   * Predicts best room for a new student based on preferences, roommate compatibility, and room history using Decision Tree.
2. **Utility Usage Anomaly Detection** *(Anomaly Detection)*
   * Uses Isolation Forest to flag unusual electricity or water usage per room/block.
3. **Complaint Prioritization System** *(Classification)*
   * Classifies and ranks incoming complaints (e.g., urgent, medium, low) using Logistic Regression or Decision Tree.
4. **Mess Attendance Forecasting** *(Regression) (Optional)*
   * Predict number of students eating per day to optimize meal prep using Linear Regression.
5. **Student Stay Duration Predictor** *(Regression) (Optional)*
   * Predict how long a student might stay (1 semester, 2 semesters, etc.) for room planning.

**🧠 Machine Learning Models to be Used:**

| **Task** | **Model** |
| --- | --- |
| Room Recommendation | Decision Tree Classifier |
| Anomaly Detection | Isolation Forest |
| Complaint Priority | Logistic Regression / Decision Tree |
| Mess Attendance Forecast | Linear Regression |
| Stay Duration Estimation | Random Forest Regressor |

**🛠️ Tech Stack:**

**Frontend:**

* HTML, CSS, JavaScript
* React.js or Bootstrap (for responsive UI) *(Optional)*

**Backend:**

* Python (Flask or Django)
* REST API for model integration

**Database:**

* MySQL or PostgreSQL
* Firebase Firestore

**ML Libraries:**

* scikit-learn
* Pandas, NumPy
* matplotlib/seaborn (for visualization)

**📊 Datasets (Real or Simulated):**

* Historical student data (rooms, duration, academic year)
* Resource usage logs (simulated utility meter readings)
* Complaint records and urgency levels
* Daily mess attendance data

**📆 8-Week Project Timeline:**

| **Week** | **Tasks** |
| --- | --- |
| **1** | Requirement Gathering, Design UI mockups, Setup tools |
| **2** | Build database schema, Implement student registration and admin panel |
| **3** | Add room allocation system and manual override features |
| **4** | Integrate mess attendance, menu planning, and complaint module |
| **5** | Train Room Recommendation and Complaint Priority models |
| **6** | Add Mess Attendance Forecasting and Anomaly Detection modules |
| **7** | Connect all AI models to the backend, test API responses |
| **8** | Final testing, UI polish, documentation, and deployment |

**📄 Expected Output:**

* Fully functional web application
* Admin + student interface
* AI assistance integrated with admin control
* Downloadable reports and logs

**✅ Benefits:**

* Reduces workload of hostel managers
* Encourages data-driven decisions
* Prevents misuse of resources
* Improves satisfaction for students with responsive systems