Hospital 360: Intelligent Resource Utilization & Patient Outcomes Dashboard

Problem Statement:

Hospital operations teams often struggle with invisible bottlenecks, unpredictable patient influx, and inefficient resource allocation. Without real-time data, decision-makers cannot optimize bed usage, staffing levels, or patient flow, leading to longer wait times and lost revenue.

The Solution:

I built **Hospital 360**, an end-to-end analytics platform that ingests raw operational data and converts it into strategic insights. The system uses a robust **Python (FastAPI) and MySQL** backend to manage complex relationships between patients, admissions, and billing. The frontend is an interactive **Power BI dashboard** that empowers non-technical staff to monitor KPIs and drill down into department-level performance.

Key Features & Innovations:

- 🤖 **Predictive Analytics:**

  Implemented AI forecasting models to predict patient admission trends for the upcoming weeks, enabling proactive resource planning.

- 🚨 **Bottleneck Detection:**

  A clustered analysis of "Average Length of Stay" (ALOS) by department identifies exactly where patient flow is stalling.

- 💰 **Revenue Intelligence:**

  Visualizes revenue breakdown by insurance type and department, highlighting the most profitable service lines.

- ⚡ **Interactive Exploration:**

  Dynamic slicers allow users to filter the entire dashboard by Department (e.g., Cardiology) or Admission Status instantly.

- 👷 **Staff Efficiency:**

  Automated leaderboards track doctor utilization rates and patient outcomes to balance workload.

How it Solves the Problem:

1. **Optimizes Capacity:**

   The "Occupancy Rate" and "Bed Utilization" metrics allow for real-time bed management.

2. **Improves Staffing:**

   The "Peak Traffic" and "Prediction" charts help HR schedule more nurses during predicted surge periods.

3. **Reduces Costs:**

   By identifying departments with abnormal ALOS, management can investigate and fix inefficiencies that drive up costs.

Tech Stack:

- **Backend:** Python (FastAPI), SQLAlchemy, Pandas
- **Database:** MySQL 8.0
- **Analytics:** Microsoft Power BI (DAX Measures, Forecasting)