GENERAL PROBLEM STATEMENTS

1. AI-Powered Resume Screener

Problem: Hiring managers receive hundreds of resumes. Manually reviewing them is time-consuming.
Solution: Build an AI tool that scans resumes and ranks candidates based on job descriptions.
Tech Stack: Python, OpenAI API, Firebase, HTML/CSS
Resources:

- OpenAI API Docs – For AI-based text processing
- How Resume Screening Works – Understanding ATS systems

---

2. Fake News Detector

Problem: Misinformation spreads quickly online, making it hard to identify real vs. fake news.
Solution: Develop a tool that scans news articles or social media posts and predicts their credibility.
Tech Stack: Python, TensorFlow, NLP models
Resources:

- Fake News Dataset – For training AI models
- Hugging Face Transformers – For NLP model implementation

---

3. Disaster Relief Coordination Platform

Problem: In emergencies, people struggle to find help, and volunteers don't know where to assist.
Solution: Build an app that connects people in need with NGOs and volunteers in real-time.
Tech Stack: React, Firebase, Google Maps API
Resources:

- Google Maps API – For mapping locations
- Firebase Realtime Database – For live updates

---

4. Smart Traffic Management System

Problem: Traffic congestion increases pollution and delays.
Solution: Develop a system that analyzes live traffic data and optimizes signals accordingly.
Tech Stack: Python, OpenCV, IoT Sensors (optional)
Resources:

- Traffic Dataset – For training AI models
- OpenCV Object Detection – For detecting vehicles

---

5. AI-Based Mental Health Assistant

Problem: Many people struggle with mental health but don't have access to immediate help.
Solution: Build an AI chatbot that provides emotional support and suggests resources.
Tech Stack: Python, OpenAI API, Twilio (for SMS support)
Resources:

- Mental Health Dataset – For model training
- Twilio SMS API – For message-based support

---

6. Food Waste Reduction App

Problem: Tons of food is wasted daily while many people remain hungry.
Solution: An app that connects restaurants with NGOs and people who need food.
Tech Stack: React, Firebase, Google Maps API
Resources:

- Food Rescue Organizations – Example initiatives
- Firebase Firestore – For tracking food donations

---

7. AI-Powered Sign Language Translator

Problem: Many deaf people struggle to communicate in public places.
Solution: Build an AI tool that converts sign language to text in real time.
Tech Stack: Python, TensorFlow, OpenCV
Resources:

- Sign Language Dataset – For training models

- OpenCV Hand Detection – For recognizing gestures

---

8. Cloud-Based AI for Diagnosing Skin Diseases

Problem: Many people cannot access dermatologists for basic skin check-ups.
Solution: An AI model that analyzes skin images and suggests potential conditions.
Tech Stack: Python, TensorFlow, Google Cloud Storage
Resources:

- Dermatology Dataset – For AI training
- Google Cloud Vision API – For image analysis

---

9. AI-Powered Meeting Summarizer

Problem: People spend hours in meetings but struggle to remember key points.
Solution: A tool that listens to meetings and generates bullet-point summaries.
Tech Stack: Python, Whisper AI, Google Drive API
Resources:

- OpenAI Whisper – For speech recognition
- Google Drive API – For saving summaries

---

10. AI-Based Tutor for Kids

Problem: Some kids struggle with learning and need personalized teaching.
Solution: An AI tutor that adapts to a child's learning pace and provides exercises.
Tech Stack: Python, OpenAI API, Firebase
Resources:

- Duolingo AI – Example of AI in education
- Khan Academy API – For educational content

---

11. Cloud Cost Optimization Tool

Problem: Companies often overspend on cloud services without realizing it.
Solution: A tool that analyzes cloud usage and suggests cost-saving measures.
Tech Stack: AWS Cost Explorer, Python, Firebase
Resources:

- AWS Cost Management – To track cloud spending

---

12. AI-Based Home Security System

Problem: Many home security cameras don't detect threats intelligently.
Solution: A smart security camera system that detects intruders and alerts homeowners.
Tech Stack: Raspberry Pi, OpenCV, Firebase
Resources:

- OpenCV Motion Detection

---

13. Real-Time Air Quality Monitoring App

Problem: Many cities suffer from pollution, but people don't have real-time air quality data.
Solution: An app that uses IoT sensors or API data to show live air quality levels.
Tech Stack: React, Firebase, IoT Sensors (optional)
Resources:

- Air Quality API – Real-time pollution data

---

14. AI-Powered Resume Writer

Problem: Many job seekers struggle to create well-structured resumes.
Solution: An AI tool that helps users generate professional resumes.
Tech Stack: OpenAI API, React, Firebase
Resources:

- Resume Parsing AI

---

15. Blockchain-Based Digital Identity System

Problem: Many people lack secure digital identities for online services.
Solution: A blockchain-powered identity verification system.
Tech Stack: Ethereum, Solidity, React
Resources:

- Blockchain Identity Management

PROBLEM STATEMENT RELATED TO UNIVERSITY

Problem Statement 16: The UMS Information Black Hole

Problem: The University Management System (UMS) is a chaotic flood of information. Critical updates like exam schedules, fee deadlines, and event registrations are buried under dozens of non-urgent circulars. Students constantly live in fear of missing a crucial deadline because the system lacks intelligent prioritization, leading to unnecessary stress and last-minute panic.

Challenge: Design a unified dashboard, mobile app, or browser extension that intelligently scrapes, categorizes, and prioritizes all UMS notifications. Your solution must provide a clean, scannable interface with personalized alerts for deadlines and truly important announcements.

Problem Statement 17: The Mess Queue Time Sink

Problem: Students waste a significant portion of their day—up to 30-40 minutes per meal—simply standing in unpredictable queues at the university messes. This inefficient system forces students to choose between attending a class on time and having a proper meal, directly impacting their well-being and time management.

Challenge: Develop a real-time crowd monitoring system for all university messes. Your solution should use live data (via sensors, user check-ins, or camera feeds) to provide an accurate 'wait time' and 'crowd level' for each mess, allowing students to choose the least crowded option and plan their schedule effectively.

Problem Statement 18: The Campus Navigation Nightmare

Problem: The 600-acre campus is a maze of identically designed blocks. Finding a specific lecture hall, lab, or faculty cabin for the first time is a frustrating and time-consuming ordeal. Static maps are useless for indoor navigation, making thousands of students, especially newcomers, late for their classes, meetings, and vivas every single day.

Challenge: Create a comprehensive indoor and outdoor navigation app for the LPU campus. The app must provide turn-by-turn directions from a user's current location to any room on campus, be it a classroom in Block 34 or an office in the administration building.

Problem Statement 19: The Dead-End Grievance System

Problem: When students face non-academic issues like a broken fan in their hostel room, a Wi-Fi outage, or a lost ID card, the process for reporting it and tracking its status is broken. Students submit complaints into a black box, with no visibility on who is responsible, what the status is, or when to expect a resolution, leading to unresolved issues and immense frustration.

Challenge: Build a transparent and streamlined grievance redressal platform. The system must allow students to submit a complaint with photo evidence, automatically route it to the correct department (e.g., Hostel Maintenance, IT Services), and provide real-time status updates from 'Ticket Raised' to 'In Progress' to 'Resolved.'

Problem Statement 20: The Study Space Scramble

Problem: During mid-terms and final exams, finding an empty seat in the Central Library or any designated study area is a stressful gamble. Students waste valuable study time wandering through floors and buildings, only to find every spot occupied. The lack of a centralized system to track space availability creates unnecessary competition and anxiety.

Challenge: Develop a smart occupancy monitoring system for all university study spaces. Your solution should display the real-time availability of seats in the library, reading rooms, and common study areas on a live map, allowing students to find and reserve a spot without the frustrating search.