**Title:** 15 Real-Time Use Cases with Low-Level Design (LLD)

### 1. EV Charging Slot Booking System ⚡

**Domain:** Smart Mobility / IoT

**Functional Requirements:** - Users can view EV stations - Each station has available time slots - Users can book or cancel a slot - Prevent booking overlap for the same slot

**Low-Level Design:** - Station: station\_id, location, list of Slot - Slot: slot\_id, time\_window, is\_booked - EVBookingSystem: stores station dictionary {station\_id: Station}, methods for booking/canceling

### 2. Freelancer Availability Tracker 👨‍💼

**Domain:** Gig Economy / HR

**Functional Requirements:** - Add freelancers with skill tags - Book freelancers - Filter by skill or availability

**Low-Level Design:** - Freelancer: id, name, skills, is\_available - FreelancePortal: stores {id: Freelancer}, methods to filter, book

### 3. Blood Donation Matching System 🩸

**Domain:** Healthcare / NGO

**Functional Requirements:** - Register donors with blood group and location - Emergency request for blood - Show matching donors

**Low-Level Design:** - Donor: id, name, blood\_group, city, is\_available - BloodBank: stores {id: Donor}, methods to match and book donor

### 4. Green Points Carbon Tracker 🌱

**Domain:** ESG / Sustainability

**Functional Requirements:** - Log activities (e.g., cycling, recycling) - Each activity has point value - Total points per user - Leaderboard view

**Low-Level Design:** - User: id, name, green\_points - Activity: name, point\_value - GreenApp: stores users and activities in dictionaries

### 5. Remote Interview Scheduler 🗓️

**Domain:** HR / EdTech

**Functional Requirements:** - Add candidates with available slots - Schedule without time conflicts - Reschedule and cancel

**Low-Level Design:** - Candidate: id, name, list of available slots - Interview: candidate\_id, time\_slot - InterviewScheduler: methods to check conflict and assign

### 6. AI Prompt Manager 🤖

**Domain:** Developer Tools / GenAI

**Functional Requirements:** - Save prompts with tags and tool (ChatGPT, DALL-E, etc.) - Search, filter, edit, favorite

**Low-Level Design:** - Prompt: id, text, tags, tool, is\_favorite - PromptManager: stores {id: Prompt}, filtering, CRUD methods

### 7. Personal Finance Tracker 💰

**Domain:** FinTech

**Functional Requirements:** - Log income/expense - Filter by category/date - View monthly balance

**Low-Level Design:** - Transaction: id, amount, category, date, type - FinanceTracker: stores list of transactions per user

### 8. Campus Bus Tracker 🚌

**Domain:** Education / Logistics

**Functional Requirements:** - Add routes and timings - Assign students to routes - View availability and status

**Low-Level Design:** - Bus: id, route, timing - Student: id, name, route\_id - CampusBusApp: maps students to routes

### 9. Digital Waste Collection Tracker ♻️

**Domain:** Civic Tech / Sustainability

**Functional Requirements:** - Add pickup request - Assign waste collector - Track collection status

**Low-Level Design:** - Request: id, user\_id, location, status - Collector: id, name - WasteTracker: manages requests and assignments

### 10. Vaccination Slot Notification System 💉

**Domain:** Health / Govt

**Functional Requirements:** - Register citizens - Notify based on eligibility - Book slots and track status

**Low-Level Design:** - Citizen: id, name, age, dose1\_done, dose2\_done - Slot: date, location, vaccine - VaccinationSystem: manages bookings and notifications

### 11. Lost & Found Portal for Campus 🎒

**Domain:** Campus Utility

**Functional Requirements:** - Report lost/found item - Match items - View all items by category

**Low-Level Design:** - Item: id, description, category, status - LostFoundManager: matches based on keywords/category

### 12. Custom PC Build Configurator 🖥️

**Domain:** E-commerce / Tech

**Functional Requirements:** - Select CPU, GPU, RAM etc. - Check compatibility - Save and share builds

**Low-Level Design:** - Component: id, name, type (e.g. CPU), specs - Build: id, components\_list - PCBuilder: validates and saves builds

### 13. Pet Care Appointment Scheduler 🐶

**Domain:** Veterinary / Booking

**Functional Requirements:** - Add pet profiles - View doctor slots and book - Track appointment history

**Low-Level Design:** - Pet: id, name, breed, owner - Appointment: pet\_id, time\_slot, doctor - PetCareApp: manages appointments

### 14. Hackathon Registration System 🧠

**Domain:** EdTech / Events

**Functional Requirements:** - Register teams - Assign to tracks - Lock registration post-deadline

**Low-Level Design:** - Team: id, name, members, track - HackathonSystem: stores teams, filters by track

### 15. Apartment Complaint Tracker 🏢

**Domain:** Property Management

**Functional Requirements:** - Raise maintenance complaints - Assign technician - Mark as resolved or pending

**Low-Level Design:** - Complaint: id, category, status, resident\_id - Technician: id, name - ComplaintSystem: tracks and assigns complaints

**Prepared by:** ChatGPT for practical student project planning. **Target Audience:** B.E / B.Tech (CSE, IT, AI-DS) 2nd-3rd year. **Implementation Base:** Python OOP + Dictionary