## Project Division: Parking Lot Management System

**Member 1: Structural & Data Module**

- **Focus:** Requirement 1 (Parking Lot Structure).
- **Responsibilities:**
    - Create the `ParkingLot`, `Floor`, and `ParkingSpot` classes.
    - Implement the logic for **Spot ID generation** (e.g., "F1-R1-S1").
    - Manage the **Status** (Available vs. Occupied) and the hourly rates for different spot types (Compact, Regular, Handicapped, Reserved) .

**Member 2: Vehicle & Logic Module**

- **Focus:** Requirement 2 (Vehicle Management).
- **Responsibilities:**
    - Create the `Vehicle` inheritance tree: `Motorcycle`, `Car`, `SUV/Truck`, and `HandicappedVehicle` .
    - Implement the **Validation Engine**: Logic that ensures a vehicle only parks in a valid spot (e.g., SUVs only in Regular spots, Motorcycles in Compact) .
    - Handle the logic for **Handicapped discounts**.
    - **UML:** Responsible for **Sequence Diagram 1 (Entry Process)**.

**Member 3: Entry/Exit & Ticket Module**

- **Focus:** Requirement 3 (Entry/Exit System).
- **Responsibilities:**
    - Develop the `Ticket` class (using the format T-PLATE-TIMESTAMP).
    - Manage the **Entry Process**: Recording entry time and marking spots as occupied.
    - Manage the **Exit Process**: Locating the vehicle by license plate and marking spots as available after payment.
    - **UML:** Responsible for **Sequence Diagram 2 (Exit/Payment)**.

**Member 4: Finance & Fine Module (Suraj)**

- **Focus:** Requirements 4 & 5 (Fine & Payment Management).
- **Responsibilities:**
    - Implement the **Duration Calculator**: Rounding hours up to the nearest hour (Ceiling rounding).
    - Create the **Fine Engine**: Implement the 3 Fine Schemes (Fixed, Progressive, and Hourly) so an admin can switch between them .
    - Handle **Fine History**: Ensuring unpaid fines are linked to the license plate and added to future totals.
    - **UML:** Responsible for **Sequence Diagram 3 (Unpaid Fines Scenario)**.

**Member 5: GUI & Reporting Module**

- **Focus:** Requirement 6 (User Interface).

- **Responsibilities:**
    - Build the **Admin Panel**: Occupancy rates, revenue tracking, and fine scheme selection .
    - Build the **Reporting Panel**: List current vehicles and outstanding fines .
    - Develop the **Main Interface**: Using Java Swing to connect the components from Members 1-4 into a single application.
    - **UML:** Responsible for the **Use Case Diagram**.