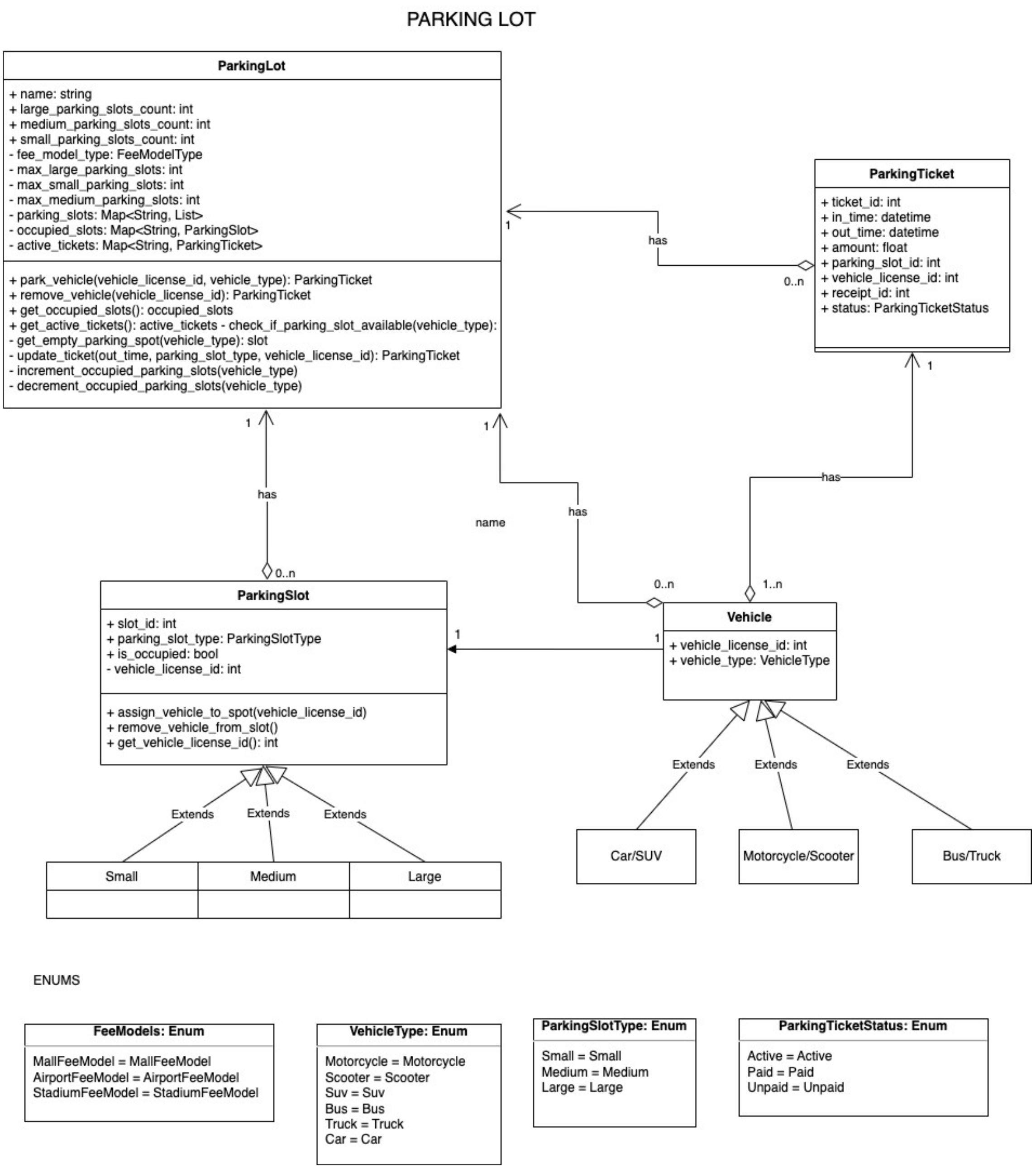
Parking Lot

This project is for coming up with a generic parking lot which can be used for different areas, buildings and utilities like Malls, Airports, Stadiums etc.

Class Diagram of the Parking Lot


PARKING LOT
ParkingLot
+ name: string
+ large_parking_slots_count: int
+ medium_parking_slots_count: int
+ small_parking_slots_count: int
- fee_model_type: FeeModelType
- max_large_parking_slots: int
- max_small_parking_slots: int
- max_medium_parking_slots: int
- parking_slots: Map<String, List>
- occupied_slots: Map<String, ParkingSlot>
- active_tickets: Map<String, ParkingTicket>
+ park_vehicle(vehicle_license_id, vehicle_type): ParkingTicket
+ remove_vehicle(vehicle_license_id): ParkingTicket
+ get_occupied_slots(): occupied_slots
+ get_active_tickets(): active_tickets - check_if_parking_slot_available(vehicle_type):
- get_empty_parking_spot(vehicle_type): slot
- update_ticket(out_time, parking_slot_type, vehicle_license_id): ParkingTicket
- increment_occupied_parking_slots(vehicle_type)
- decrement_occupied_parking_slots(vehicle_type)
ParkingTicket
+ ticket_id: int
+ in_time: datetime
+ out_time: datetime
+ amount: float
+ parking_slot_id: int
+ vehicle_license_id: int
+ receipt_id: int
+ status: ParkingTicketStatus
1
has
0..n
1
1
has
0..n
1
has
name
has
0..n
1..n
ParkingSlot
+ slot_id: int
+ parking_slot_type: ParkingSlotType
+ is_occupied: bool
- vehicle_license_id: int
+ assign_vehicle_to_spot(vehicle_license_id)
+ remove_vehicle_from_slot()
+ get_vehicle_license_id(): int
1
1
Vehicle
+ vehicle_license_id: int
+ vehicle_type: VehicleType
Extends
Extends
Extends
Small
Medium
Large
Extends
Extends
Extends
Car/SUV
Motorcycle/Scooter
Bus/Truck
ENUMS
FeeModels: Enum
MallFeeModel = MallFeeModel
AirportFeeModel = AirportFeeModel
StadiumFeeModel = StadiumFeeModel
VehicleType: Enum
Motorcycle = Motorcycle
Scooter = Scooter
Suv = Suv
Bus = Bus
Truck = Truck
Car = Car
ParkingSlotType: Enum
Small = Small
Medium = Medium
Large = Large
ParkingTicketStatus: Enum
Active = Active
Paid = Paid
Unpaid = Unpaid

Project Structure

The Parking Lot project contains following file structure

- parking_lot

    - parking_lot.py
    - parking_slot.py
    - parking_ticket.py
    - vehicle.py
    - FeeModels
        - Airport Fee Model
        - Mall Fee Model
        - Stadium Fee Model
    - Exceptions
        - NoSlotAvailableException
        - VehicleNotAllowedException
- tests
    - test_scenarios_from_project_requirement_doc
        - Multiple files representing 4 the scenarios in the project description doc
    - test_airport_fee_model.py
    - test_mall_fee_model.py
    - test_parking_lot.py
    - test_parking_slot.py
    - test_stadium_fee_model.py

Running / Testing the parking Lot

All testing of the Parking Lot system is done through Unit tests. In order to run the tests, you can run 'python -m unittest' from inside the ParkinLot directory on the terminal. This would automatically detect all the 30 tests and run them. To run any individual test file, run the command 'python -m unittest path/to/file' Code Coverage focus is on testing all the functionalities of the Parking Lot, and covering corner cases, rather than just focusing on Line Coverage. Code Coverage is 94%.

- Each file under the folder tests/test_scenarios_from_project_requirement_doc/ represents tests for a scenario mentioned in the Problem Statement. i.e Scenario 1,2,3 and 4
- Other files under tests/ are general unit tests, which verify the functionalities of each class files and fell models.

Assumptions:

- A smaller vehicle cannot be parked in a larger slot, even when slots for smaller vehicles are full, and slots for larger vehicles are free.
- There is only one entry and exit to the parking slot, hence multiple vehicles cannot enter or exit at the same time.
- We assume that the ticket charges are paid for when the vehicle is unparked.

Note:

In the test scenarios mentioned in the project writeup, the last scenario for an Airport parking lot, the list of max vehicles allowed says Buses/Trucks: 100 spots. But the Airport fee model doesnot allow any Bus/Truck parking. Here, I assume that Bus/Trucks are not allowed and ignore the 100 spots.