Use Cases For A Parking System: Part One & Two

ICT 4305: Object Oriented Methods and Programming

for

Master of Science

Information Technology

Kalika Browder

University of Denver College of Professional Studies

September 17, 2025

Faculty: Nirav Shah, M.S , MBA
Director: Cathie Wilson, M.S.
Dean: Michael J. McGuire, MLS

Part One: Use Case Interactions + Diagram

Key Actor(s): Primary

- Customer (Registered user with the parking office)
- Parking Office (Manages registration, permits, and billing)
- Parking Lot System (Gate scanners, entry/exit handling)
- Billing System (Monthly account charges)

Key Actor(s): Secondary

- Vehicle (Could be it's own entity. Going to be treated as part of the customer going forward for ease)
- University Admin (Could manage parking policies, lots, and various fees but not going to include due to lack of detail and for future ease).

Key Interactions (# = Identifier)

1. **Customer Registration (Customer <-> Parking Office: #UC-01**
    a. Preconditions: Customers must be affiliated with the University as a student or as faculty.
    b. Success Condition: Customer is added to the parking system with an active profile.

c. Failure Condition: Registration fails if required information is missing or invalid.

d. Normal Flow:

    i. Customer provides personal and vehicle info to the Parking Office.

    ii. The Parking Office verifies eligibility.

    iii. The Parking Office creates a customer account.

e. Alternative Flow:

    i. If a customer is ineligible to create an account, the system rejects registration and then proceeds to notify the customer.

2. Purchase Parking Permit (Customer <-> Parking Office: #UC-02

a. See Part Two in depth description + Use Case Elements.

3. Drive Into Lot (Customer <-> Parking Lot System): #UC-03

a. Preconditions: Customer must have a valid permit registered with the Parking Office.

b. Success Condition: Gate opens and vehicle enters the parking lot.

c. Failure Condition: Entry is denied to customer/vehicle if permit is invalid or if the lot is full.

d. Normal Flow:

    i. Customer approaches the entry gate

    ii. Permit is scanned.

    iii. The system verifies the validity of the permit.

    iv. The gate opens.

e. Alternative Flow:

i. If a parking lot is at capacity, the system denies entry and displays an error message to the customer.

4. **Entry Scan Permit (Parking Lot System <-> Customer): #UC-04**
    a. Preconditions: Permit must be valid and active.
    b. Success Conditions: System records entry timestamp.
    c. Failure Conditions: Entry to the parking lot is rejected if the permit is expired or invalid.
    d. Normal Flow:
        i. Customer presents permit at scanner.
        ii. The system checks permit validity.
        iii. The system logs the customer/vehicle's entry time.
    e. Alternative Flow:
        i. If the scanner cannot read the permit, the customer needs to retry entry or request assistance from Parking Admin.
5. **Scan Permit To Exit, If Applicable (Parking Lot System <-> Billing System): #UC-05**
    a. Preconditions: Vehicle previously entered with a valid permit.
    b. Success Condition: System records exit time and calculates the charges for the customer.
    c. Failure Condition: Exit is delayed if the permit cannot be verified.
    d. Normal Flow:
        i. Customer presents permit at exit gate.
        ii. System retrieves entry record.

iii. The system logs the exit time.

e. Alternative Flow:

i. If no matching entry record exists for the permit, the system alerts Parking Admin or attendant.

6. Calculate Parking Fee (Parking Lot System <-> Billing System): #UC-06

a. Preconditions: Entry and/or exit event must be recorded.

b. Success Condition: Correct fee is calculated for the transaction.

c. Failure Condition: Fee cannot be computed if the data is missing or is corrupted in some manner.

d. Normal Flow:

i. System retrieves entry and if applicable, the exit timestamps.

ii. The system determines a lot's pricing model. This could be a flat fee, hourly, overnight, or by business hours.

iii. The fee is calculated.

e. Alternative Flow:

i. If timestamps are incomplete, the system applies the default daily charge for the specific lot.

7. Apply Compact Car Discount (System <-> Vehicle Type): #UC-07

a. Preconditions: Vehicle must be registered as a compact car.

b. Success Condition: The 20% discount is applied to the calculated fee.

c. Failure Condition: No discount is applied if the vehicle type is not compact.

d. Normal Flow:

i. The system checks vehicle type associated with the permit.

ii. If the car is compact, the system reduces the parking lot fee by 20%.

e. Alternative Flow:

i. There isn't one. The discount is only applicable when conditions are met.

8. Charge Customer Account (Billing System <-> Customer): #UC-08

a. Preconditions: The fee must be calculated.

b. Success Condition: Transaction amount is added to the customer's monthly account charge.

c. Failure Condition: Transaction is not recorded if the account cannot be updated.

d. Normal Flow:

i. Billing System receives charge request from Parking Lot System.

ii. The billing system adds charge to the customer's account balance.

e. Alternative Flow:

i. If the billing system is offline, charge is queued for later processing and updating.

9. Generate Monthly Bill (Billing System <-> Customer): #UC-09

a. Preconditions: Customer account must have at least one transaction in a billing period.

b. Success Condition: Monthly bill is generated and delivered to the customer.

c. Failure Condition: The bill is delayed or missing if the report generation fails.

d. Normal Flow:

i. At the end of the month, the Billing System clusters all of the charges.

ii. The system generates a statement for each customer.

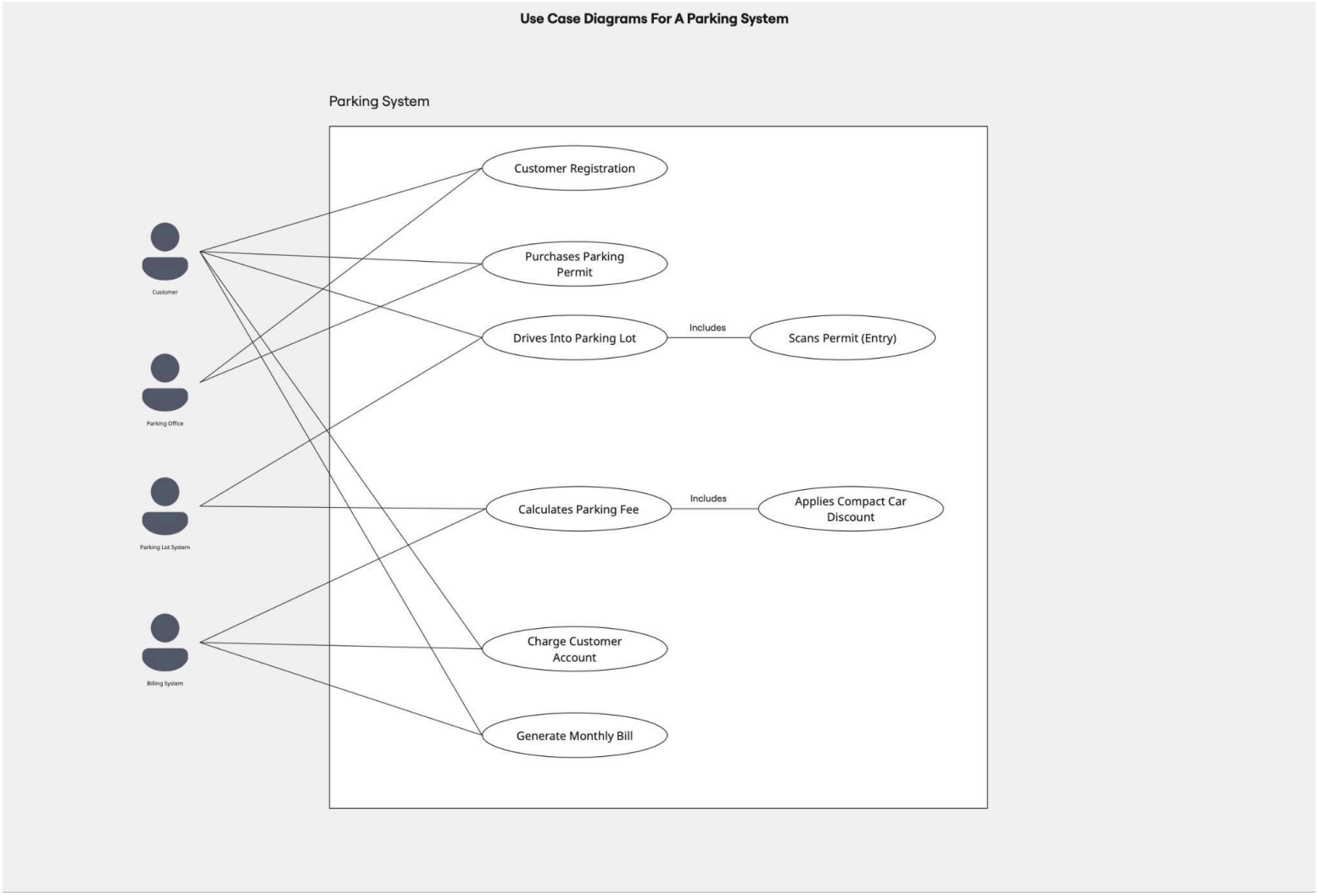
iii. The bill is sent to the customer via email or mail.

e. Alternative Flow:

i. If delivery fails, the system would flag account for a manual review and follow up.

Use Case Diagram:

https://miro.com/app/board/uXjVJJqv44o=/?share_link_id=977856039253



Part Two: Elaboration on Use Cases

Use Case Overview

The customer purchases a parking permit from the parking office (Key Interaction #2).

ID

- UC-02

Name

Purchase Parking Permit

Short Description

A registered customer requests and obtains a parking permit for one of their vehicles. The system then validates the request, confirms eligibility, and issues the permit which is linked to the customer's account.

Goal

The goal of this system interaction is to allow customers to purchase a valid parking permit so that they can legally park in the University parking lots.

Preconditions

- Customers must already be registered in the system.
- Vehicle information must be available.

Success End Condition

- Permit is successfully issued.
- Permit is linked to the customer's account and vehicle.
- The system updates the database with a new permit record.

Fail End Condition

- No permit is issued.
- The customer account remains unchanged.
- The system logs the failed attempt.

Stakeholder

- Primary Actor: Customer
- Secondary Actors: Parking Office staff, Parking System database

Trigger

- Customers request a permit purchase, either online or in person at the Parking Office.

Normal Flow

1. Customer initiates request for a new parking permit.
2. The system then verifies the customer registration status.
3. The customer provides or confirms vehicle details.
4. The system checks for permit eligibility (for example: customers may have a maximum number of permits).
5. The system then issues a new permit.
6. The system updates the customer account and links the permit to the vehicle.
7. Confirmation is provided to the customer.

Alternative Flow

- Invalid Payment: If a payment fails, the system would then notify the customer and would not issue the permit requested.
- Lot Capacity: Lot may reach capacity, meaning no spaces or permits are available for the requested lot. The system would reject the request of the permit, and then perhaps suggest alternatives for the customer.
- Permit Limit Reached: If a customer already has maximum permitted vehicles registered, the system would deny the request.

Includes

- UC-01: Customer Registration
    - Required before purchase.
- UC-08: Charge Customer Account
    - This would be triggered during payment processing.

Frequency of Use

Typically once per semester or academic year, but could occur if there are transfer students throughout the year, or may occur more often if customers continue to add additional vehicles.

Constraints + Special Requirements

- University policy could limit the number of permits per customer.
- The system must integrate with the payment processing system.
- Permit data must sync with parking lot scanners in real time.
- Issuing a permit should not exceed a certain amount (say 5 seconds) of processing time.

Assumptions

- Customers have the internet needed to pay online or they have physical access to the Parking Office.
- Payment systems are online, as well as functionable.
- Vehicle classifications (such as SUV) are already stored in the database.

Notes and Issues

- Future enhancement may allow digital permits, instead of physical tags that go on the car (this would also require a tool to look up tags for Parking staff).
- Confirmation is needed on whether temporary permits would be supported. With the information available, I would say not at this moment.
- No due date. Not assigned.

References

Agile Modeling. n.d.. “UML Use Case Diagrams: An Agile Introduction” *Agile Modeling.* Accessed September 16, 2025. https://agilemodeling.com/artifacts/useCaseDiagram.htm

Geeks For Geeks. 2025. “Unified Modeling Language (UML) Diagrams.” *Geeks For Geeks.* Accessed September 16, 2025. https://www.geeksforgeeks.org/system-design/unified-modeling-language-uml-introduction/

Usability.Gov. n.d. “Use Cases.” *Usability.gov.* Accessed September 11, 2025. https://digital.gov/topics/usability/

- *Site Archived: Obtained Document From Canvas Module 1 Instructional Materials.*