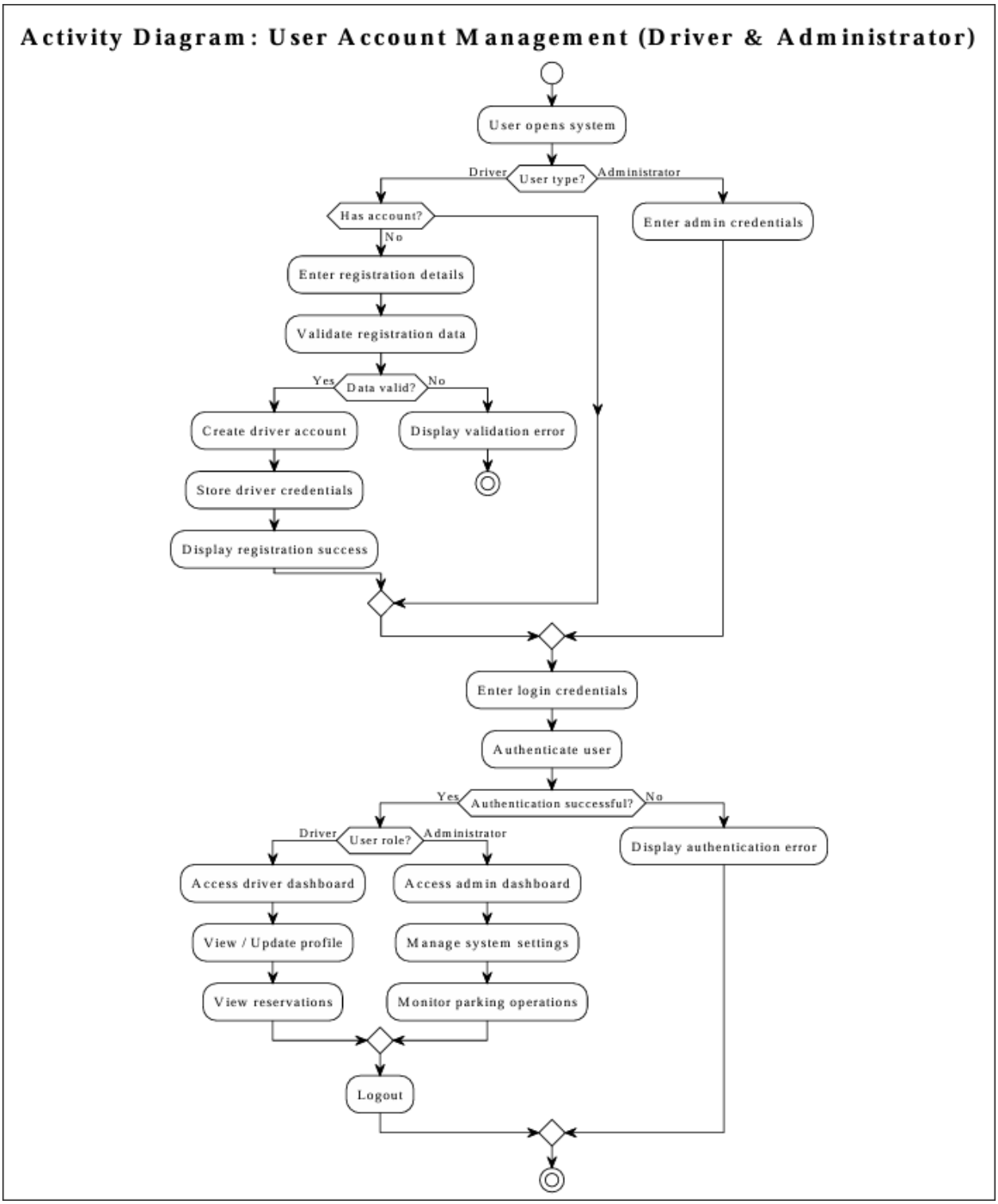
Diagram Explanations

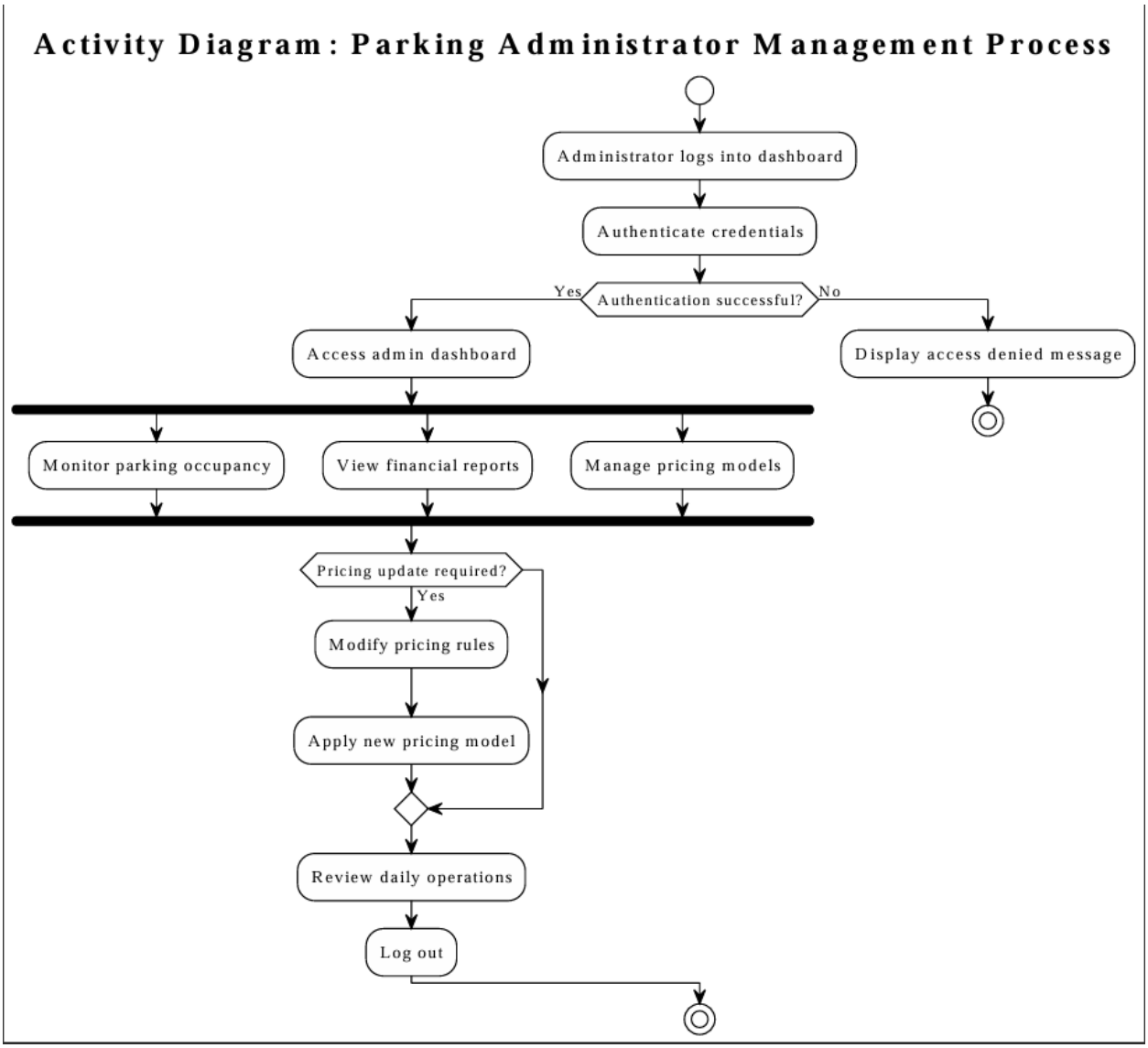
1. Activity Diagram: User Account Management (Driver & Administrator)



This diagram illustrates the authentication and onboarding workflow for the two primary system actors: Drivers and Administrators.

- **Process Flow:** The process begins when a user opens the system. The system first checks the user type.
- **Driver Path:** If the user is a driver, they can choose to register if they do not have an account. The system validates their registration data; if valid, a new account is created. Once registered, the driver logs in to access features like viewing their profile and checking reservations.
- **Administrator Path:** Administrators skip registration and proceed directly to login using pre-assigned credentials to access the admin dashboard, where they can manage settings and monitor operations.
- **Outcome:** The process ends when the user logs out or if authentication fails.

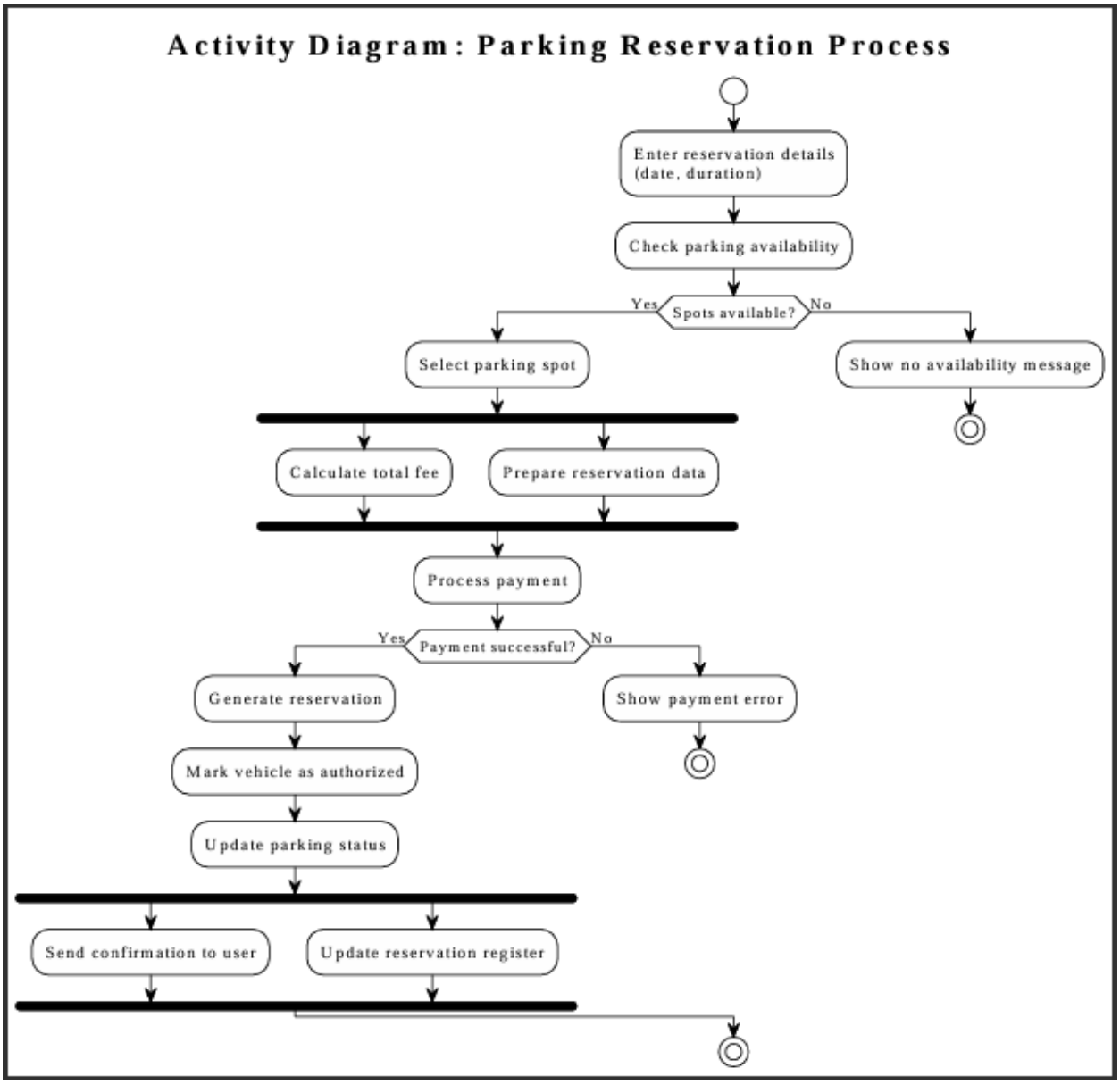
2. Activity Diagram: Parking Administrator Management Process



This diagram details the operational workflows available to the Parking Administrator after logging into the dashboard.

- **Process Flow:** Upon successful authentication, the administrator enters a parallel processing node (fork), allowing them to perform three simultaneous tasks: monitoring parking occupancy, viewing financial reports, and managing pricing models.
- **Key Decisions:** A specific decision point exists for pricing: if an update is required, the administrator modifies the rules and applies the new model.
- **Outcome:** The workflow concludes after the administrator reviews daily operations and logs out of the system.

3. Activity Diagram: Parking Reservation Process

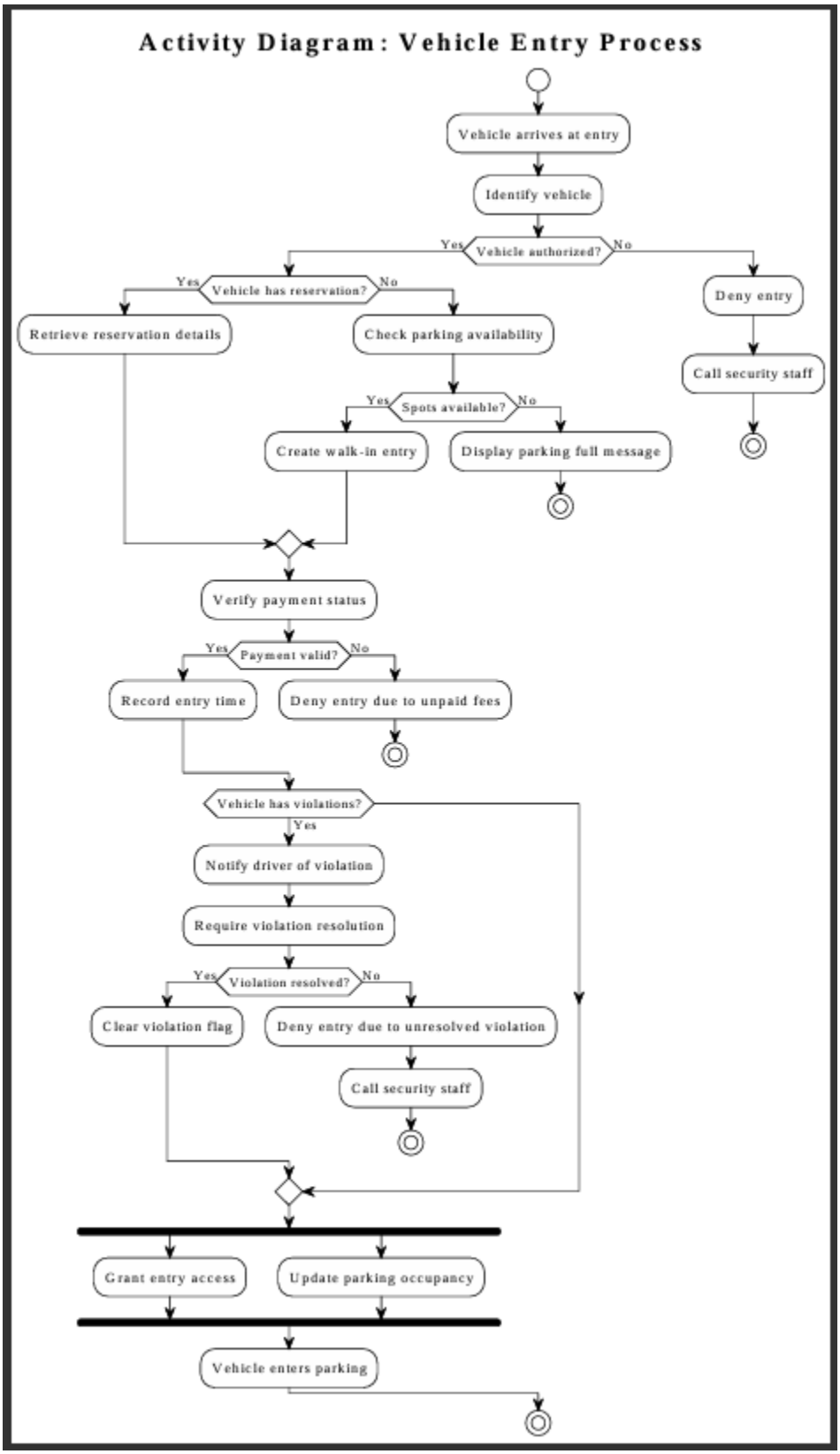


This diagram maps out the steps a driver takes to book a parking spot in advance via the mobile app.

- **Process Flow:** The driver enters reservation details (date, duration), and the system checks database availability. If no spots are free, the process ends with a message.
- **Booking Logic:** If a spot is available, the user selects it. The system then splits into parallel tasks to calculate fees and prepare reservation data.

- **Payment & Confirmation:** The driver processes the payment. If successful, the system generates the reservation, marks the vehicle as authorized, and simultaneously sends a confirmation to the user while updating the internal reservation register.

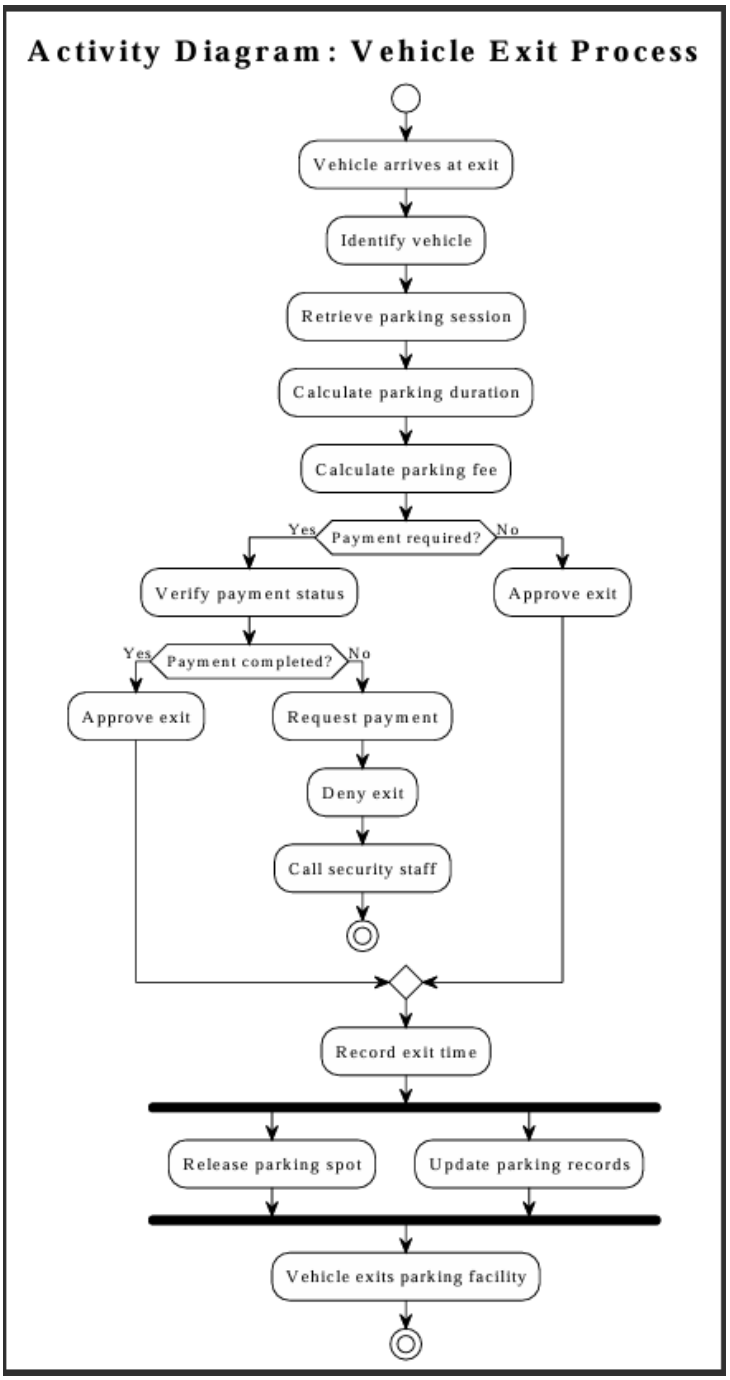
4. Activity Diagram: Vehicle Entry Process



This diagram captures the complex logic executed by the system when a vehicle arrives at the entry gate, handling both reserved and walk-in users.

- **Identification & Authorization:** The system identifies the vehicle (via ANPR). It first checks if the vehicle is authorized. If not, entry is denied, and security is notified.
- **Reservation vs. Walk-in:** If the vehicle is authorized, the system checks for an existing reservation. If none exists, it checks for walk-in availability. If the lot is full, the vehicle is turned away.
- **Validation:** The system performs checks for valid payment status and unresolved violations. If a violation is found, the driver must resolve it immediately, or entry is denied.
- **Outcome:** If all checks pass, the system performs a parallel action to grant access (open gate) and update the occupancy count.

5. Activity Diagram: Vehicle Exit Process

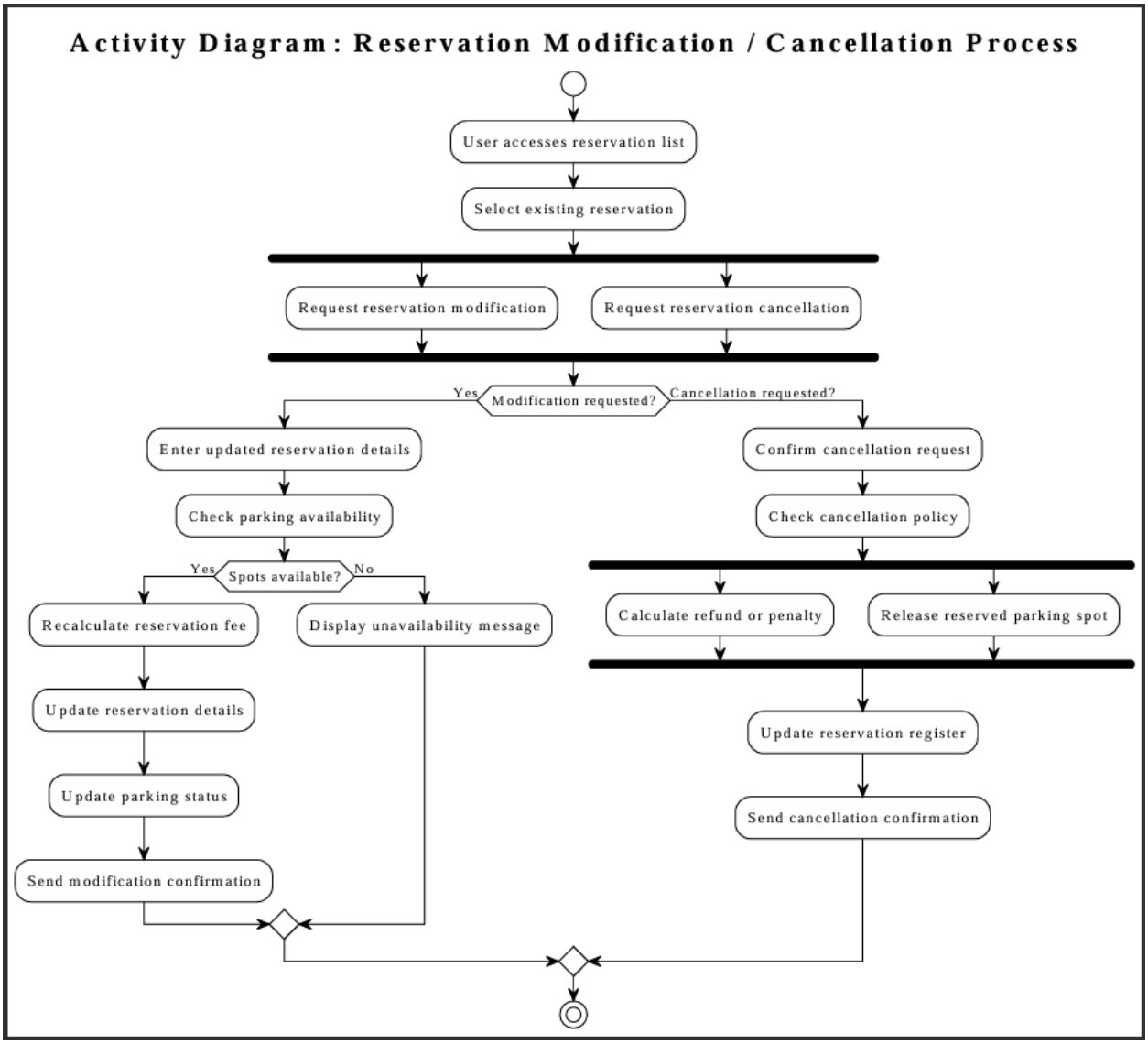


This diagram defines the logic for a vehicle leaving the facility, ensuring all fees are settled before the gate opens.

- **Process Flow:** When a vehicle arrives at the exit, the system identifies it and retrieves the active parking session to calculate the duration and total fee.

- **Payment Check:** The system determines if payment is required. If the user has already paid (or has a pre-paid reservation), exit is approved. If payment is pending, the system requests it; failure to pay results in a denied exit and a security alert.
- **Outcome:** Upon approval, the system records the exit time and performs parallel updates to release the parking spot in the database and archive the parking record.

6. Activity Diagram: Reservation Modification & Cancellation Process

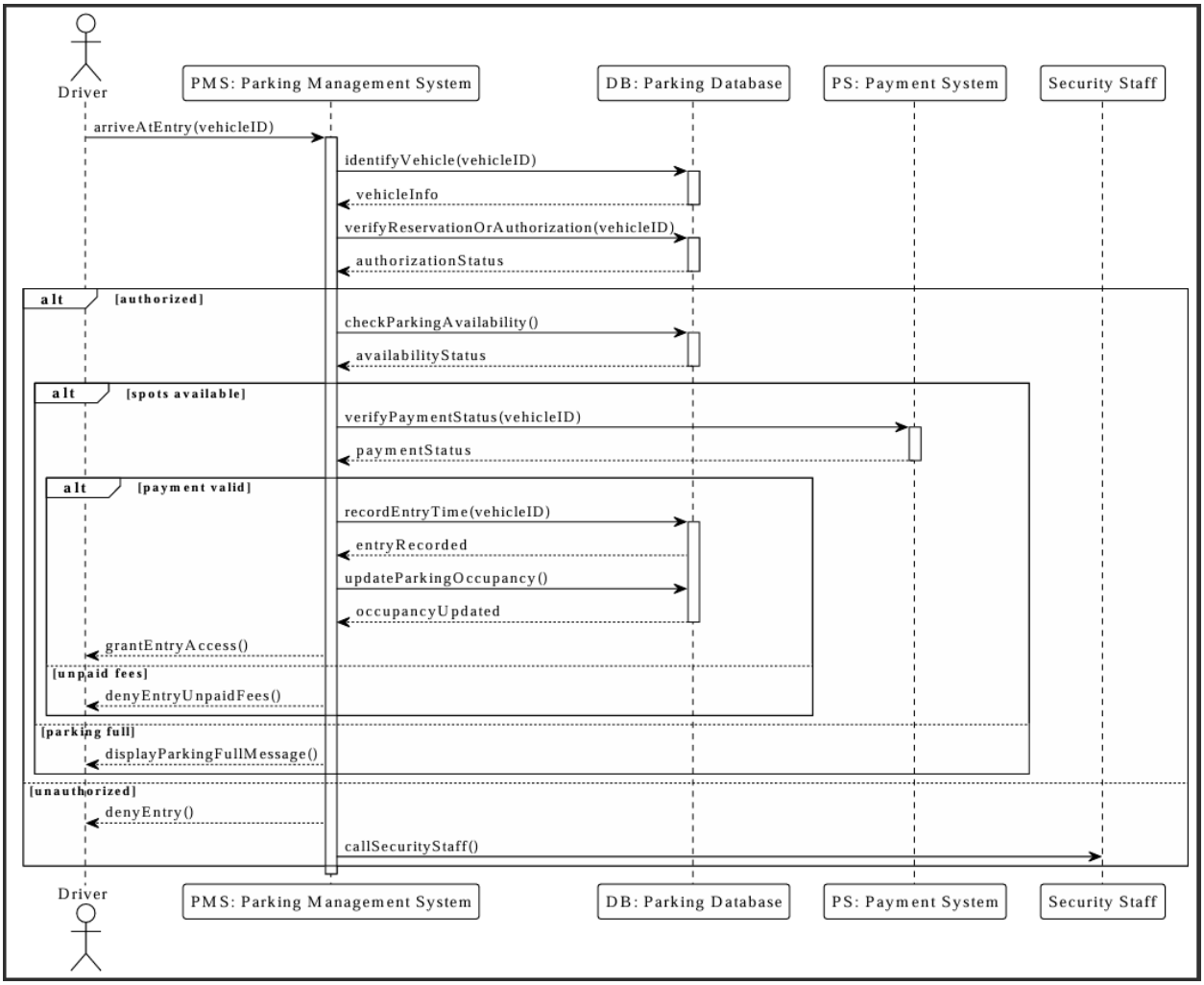


This diagram shows how users manage existing reservations, offering two distinct paths: modification or cancellation.

- **Modification Path:** If a user chooses to modify, the system accepts new details, checks availability for the new slot, recalculates fees, and updates the booking if successful.

- **Cancellation Path:** If cancellation is chosen, the system checks the policy. It then executes parallel actions to calculate any necessary refunds (or penalties) and release the reserved spot back to the pool of available parking.

7. Sequence Diagram: Vehicle Entry

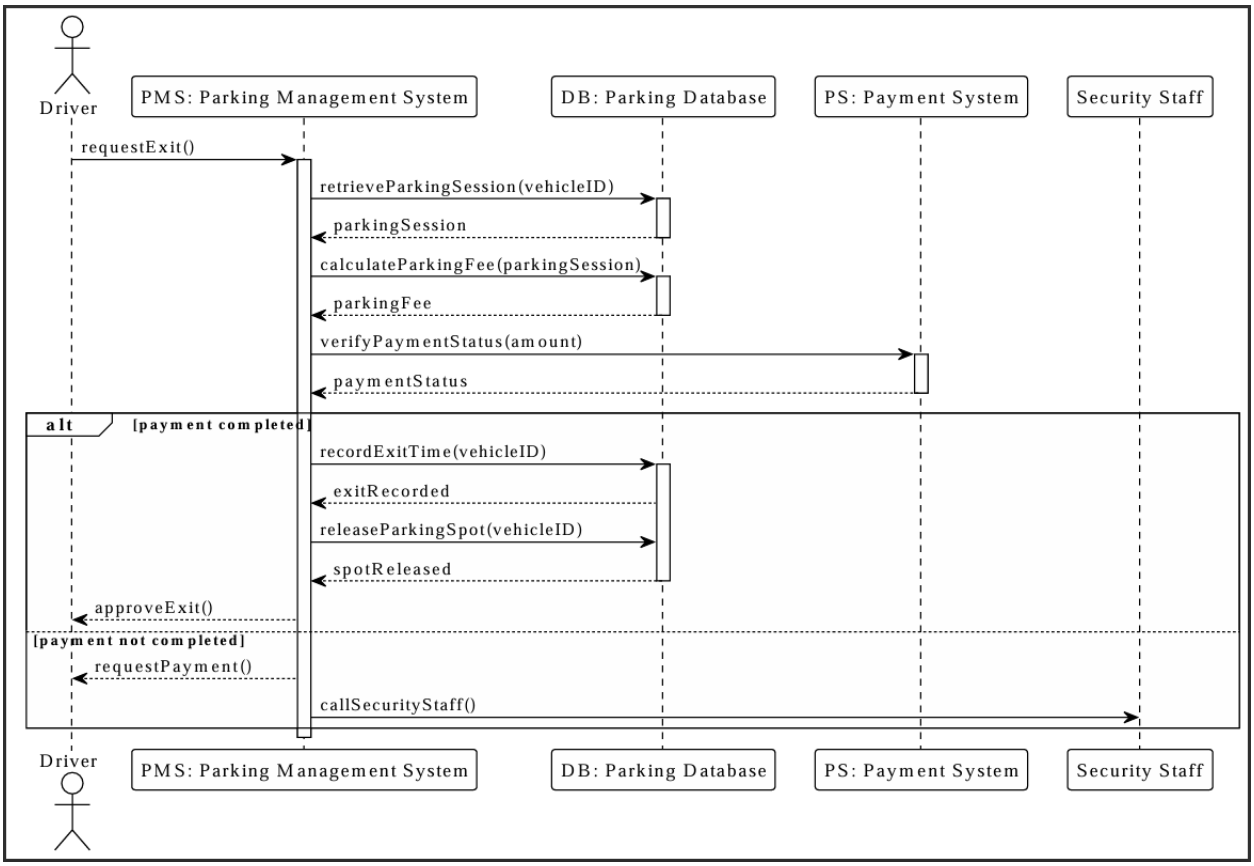


This diagram visualizes the real-time interactions between the Driver, Parking Management System (PMS), Database (DB), Payment System (PS), and Security Staff during entry.

- **Interaction:** The Driver triggers the flow by arriving. The PMS queries the DB to identify the vehicle and verify authorization.
- **Logic:** The PMS checks availability and then queries the PS to verify payment status.
- **Alt Frames:** The diagram uses "alt" blocks to show different scenarios:
    - **Authorized/Paid:** The PMS records entry in the DB, updates occupancy, and grants access.

- **Unpaid/Full:** The PMS returns a denial message to the driver.
- **Unauthorized:** The PMS denies entry and sends a message to the Security Staff.

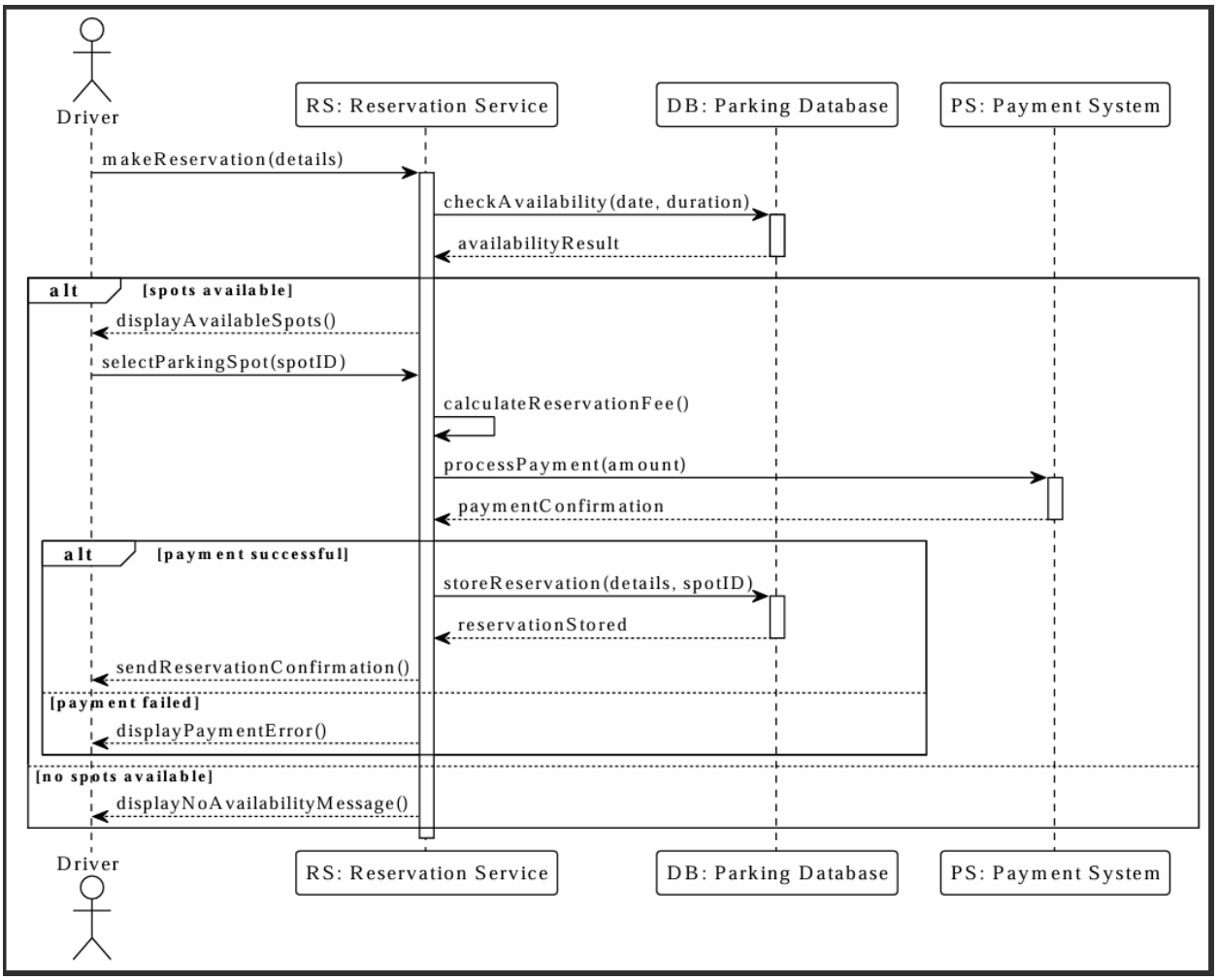
8. Sequence Diagram: Vehicle Exit



This diagram details the object interactions required to process a vehicle leaving the lot.

- **Interaction:** The Driver requests an exit. The PMS retrieves the session from the DB and calculates the fee.
- **Payment Verification:** The PMS contacts the Payment System (PS) to verify if the fee is settled.
- **Alt Frames:**
  - **Payment Completed:** The PMS tells the DB to record the exit time and release the spot, then approves the exit for the Driver.
  - **Payment Not Completed:** The PMS prompts the Driver to pay and simultaneously alerts Security Staff.

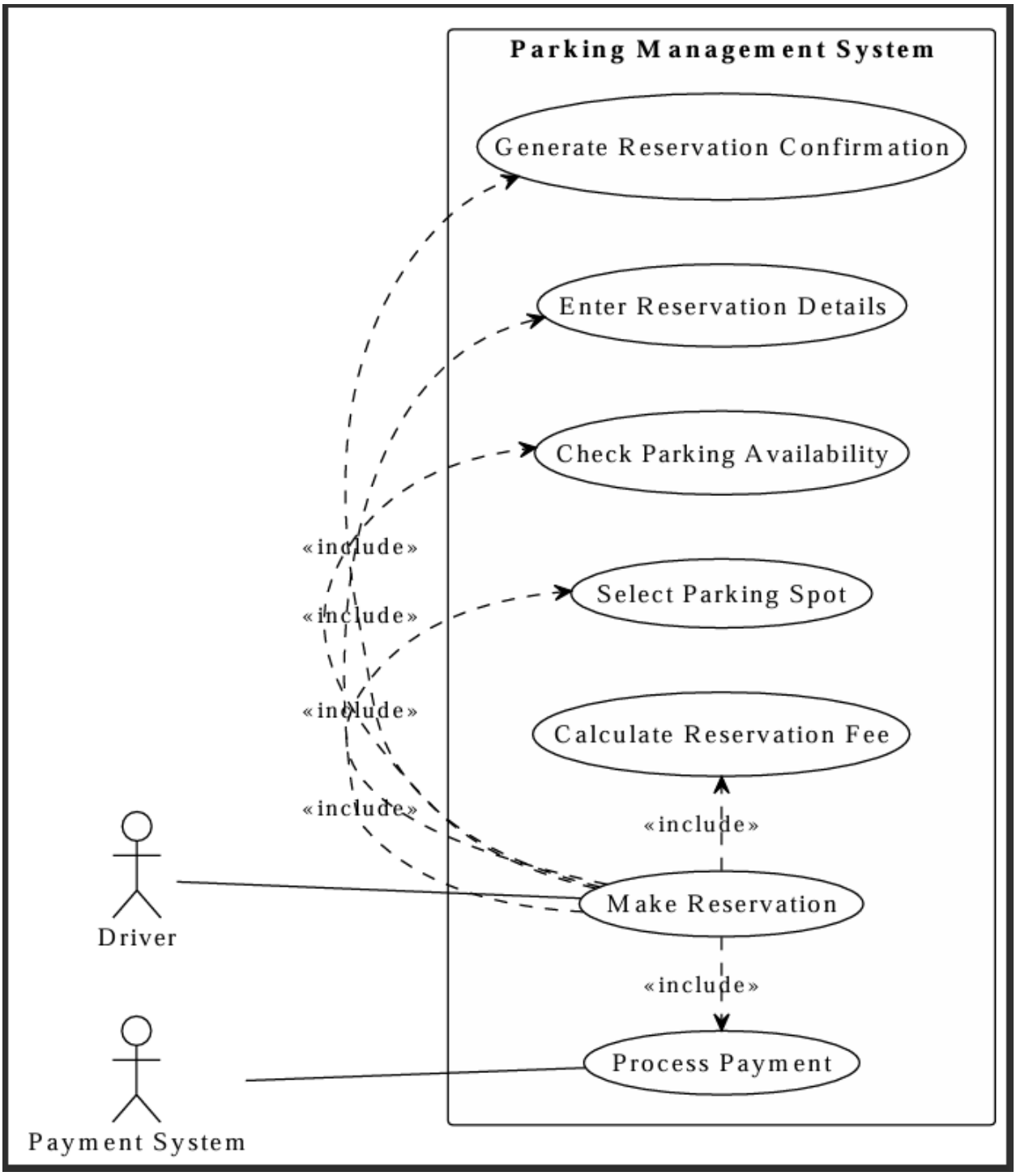
9. Sequence Diagram: Make Reservation



This diagram shows the technical message exchange required to create a new reservation.

- **Interaction:** The Driver sends details to the Reservation Service (RS). The RS queries the DB for availability.
- **Flow:** If spots are available, the RS displays them. The driver selects a spot, and the RS calculates the fee.
- **Payment:** The RS sends a payment request to the Payment System (PS).
- **Alt Frames:**
    - **Success:** If the PS confirms payment, the RS stores the reservation in the DB and sends a confirmation to the Driver.
    - **Failure:** If payment fails or no spots are available, the RS returns an error message to the Driver.

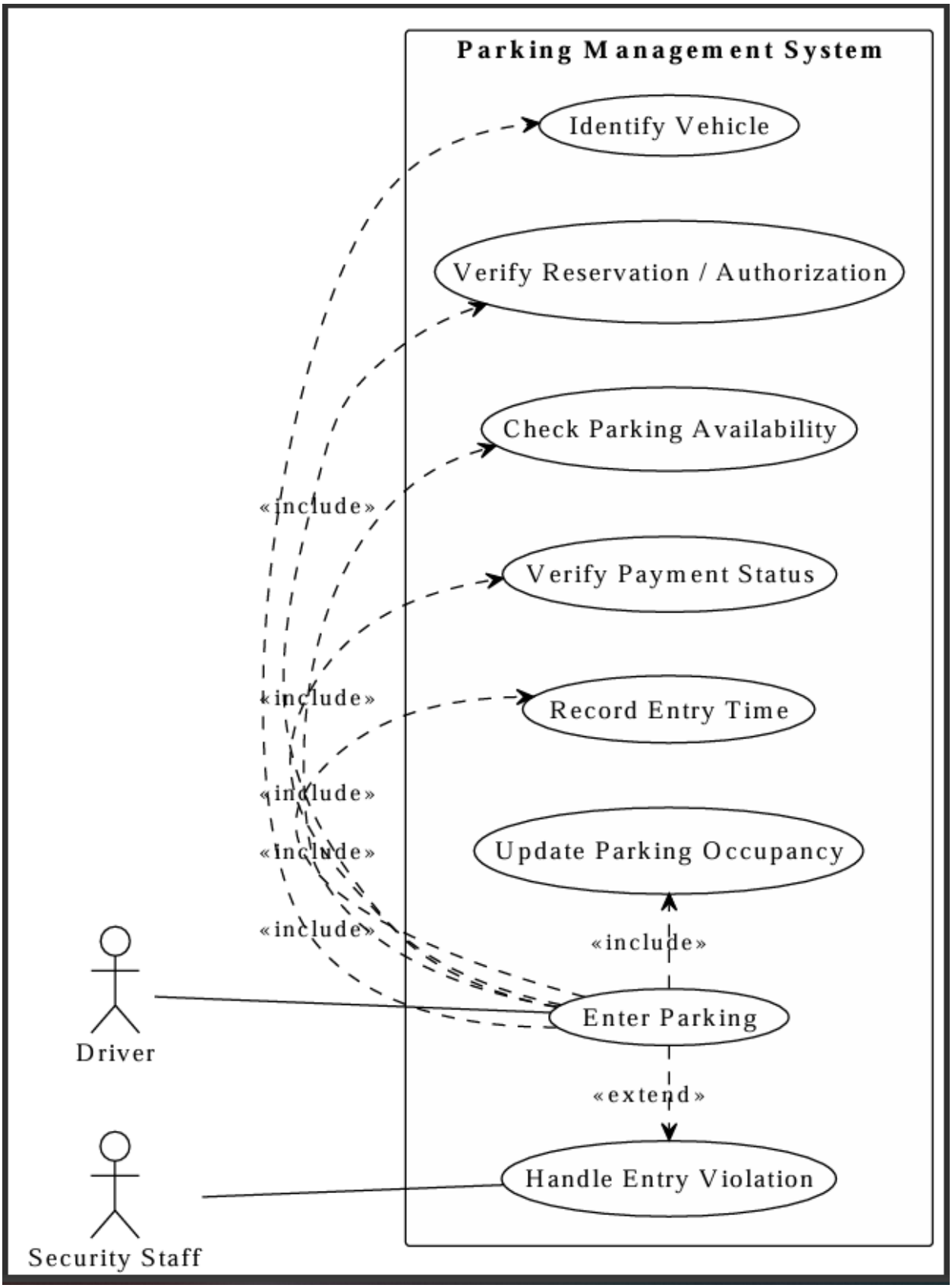
Use Case Description 1: Make Reservation



Item	Description
Use Case Name	Make Reservation

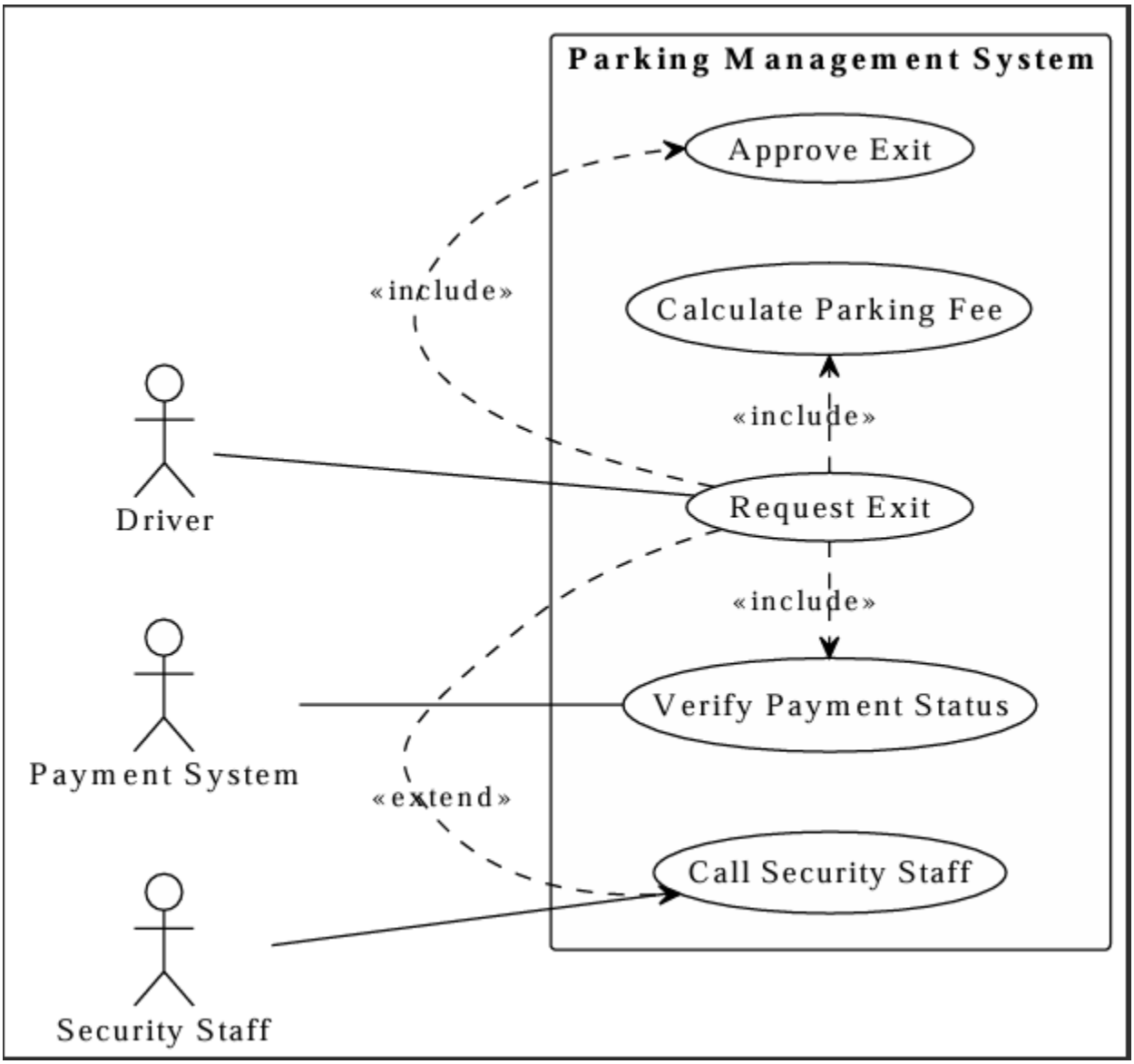
Item	Description
Primary Actor	Driver
Secondary Actor(s)	Payment System
Goal	To allow the driver to reserve a parking spot for a specific date and duration
Preconditions	- Driver is registered and logged in - Parking system is available
Postconditions	- Reservation is created and stored - Parking spot is reserved - Confirmation is sent to the driver
Main Scenario	1. Driver selects “Make Reservation” 2. Driver enters reservation details (date, duration) 3. System checks parking availability 4. Driver selects an available parking spot 5. System calculates the reservation fee 6. System processes payment via Payment System 7. System generates reservation confirmation
Alternative Scenarios	- **3a. No parking spots available:** system displays unavailability message - **6a. Payment fails:** system displays payment error and reservation is not created

Use Case Description 2: Enter Parking (Vehicle Entry)


Parking Management System
Identify Vehicle
Verify Reservation / Authorization
Check Parking Availability
Verify Payment Status
Record Entry Time
Update Parking Occupancy
Enter Parking
Handle Entry Violation
«include»
«include»
«include»
«include»
«include»
«include»
«extend»
Driver
Security Staff

Item	Description
Use Case Name	Enter Parking
Primary Actor	Driver
Secondary Actor(s)	Security Staff
Goal	To allow an authorized vehicle to enter the parking facility
Preconditions	- Driver has a valid reservation or authorization - Parking system is operational
Postconditions	- Entry time is recorded - Parking occupancy is updated - Vehicle is granted access
Main Scenario	1. Driver arrives at parking entry 2. System identifies the vehicle 3. System verifies reservation or authorization 4. System checks parking availability 5. System verifies payment status 6. System records entry time 7. System updates parking occupancy 8. System grants entry access
Alternative Scenarios	- **3a. Unauthorized vehicle:** entry is denied and call security staff - **5a. Unpaid fees:** entry is denied and call security staff - **Violation detected:** system calls security staff

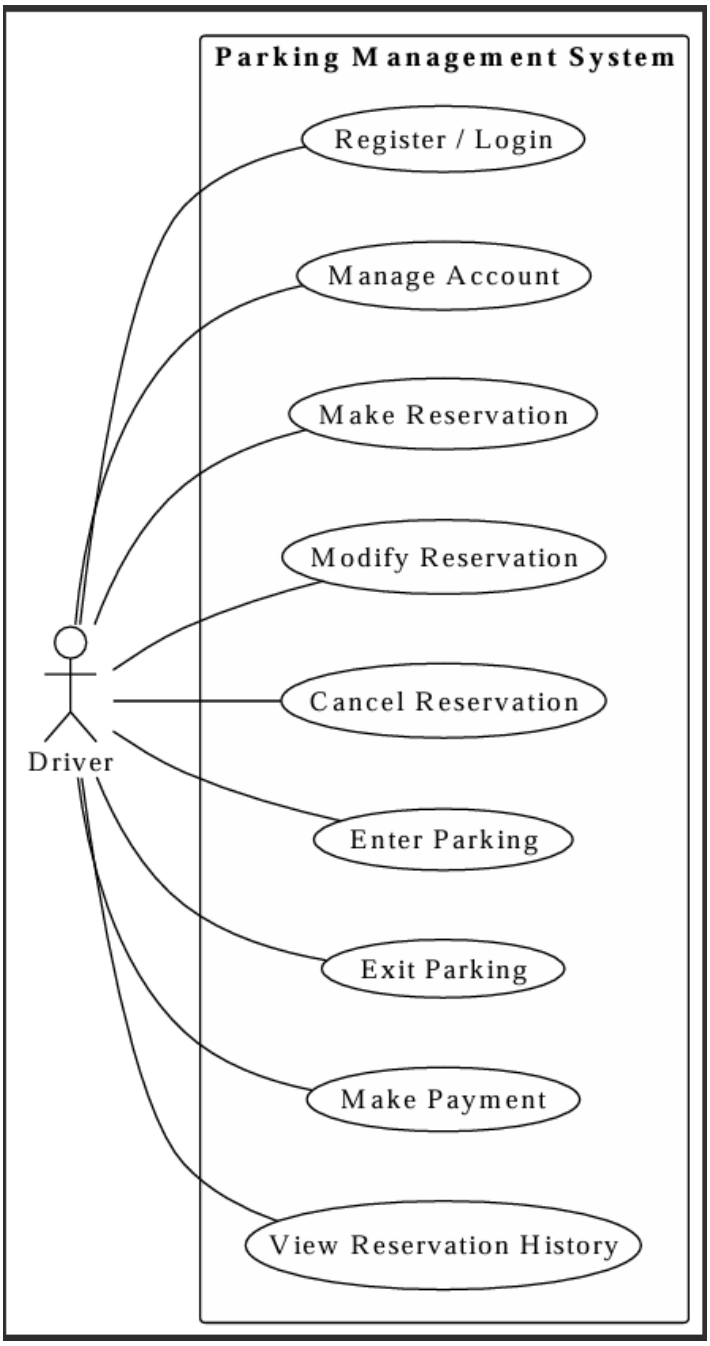
Use Case Description: Exit Parking



Item	Description
Use Case Name	Exit Parking
Primary Actor	Driver
Secondary Actor(s)	Payment System, Security Staff
Goal	To allow a parked vehicle to exit the parking facility after settling any required fees

Item	Description
Preconditions	- Vehicle is currently parked in the facility - Parking system is operational
Postconditions	- Exit time is recorded - Parking fee is settled (if required) - Parking spot is released - Vehicle exits the parking facility
Main Scenario	1. Driver arrives at the parking exit 2. Driver requests exit 3. System calculates parking fee 4. System verifies payment status via the Payment System 5. System approves exit 6. System records exit time 7. System releases the parking spot 8. Vehicle exits the parking facility
Alternative Scenarios	- **4a. Payment not completed:** system requests payment and denies exit until payment is completed and call security staff - **Payment failure:** system denies exit and call security staff - **Violation or issue detected:** system calls security staff

Use Case Description: Driver System Interaction (Composite Use Case)



Item	Description
Use Case Name	Driver System Interaction
Primary Actor	Driver

Item	Description
Secondary Actor(s)	Payment System, Security Staff
Goal	To allow the driver to interact with the parking management system to manage reservations, parking entry and exit, payments, and account information
Preconditions	- Driver has access to the parking management system - System is operational
Postconditions	- Driver actions are completed successfully (reservation, entry, exit, or account updates) - System records are updated accordingly
Main Scenario	1. Driver registers or logs into the system 2. Driver manages account information 3. Driver makes, modifies, or cancels a reservation 4. Driver enters the parking facility 5. Driver makes required payments 6. Driver exits the parking facility 7. Driver views reservation history
Alternative Scenarios	- Payment failure during reservation or exit - Reservation modification or cancellation request - Entry or exit denied due to unpaid fees or violations