Standard Use Case Models

407 Express Toll Route

1. Use Case: RegisterAccount

Scope: 407 ETR System

Level: User Goal

Intention in Context: To register an account for an individual or a company for toll payment.

Multiplicity: One account per individual or company.

Primary Actor: Motorist or Company Representative

Secondary Actors: System (processes registration)

Main Success Scenario:

1. User provides billing information and personal/company details.
2. System creates an account and confirms registration to the user.

Extensions & Exceptions:

1a. If the user provides incomplete or invalid information, the system prompts for correct details.

2. Use Case: RegisterVehicle

Scope: 407 ETR System

Level: User Goal

Intention in Context: To register a vehicle with the account and issue a transponder.

Multiplicity: Multiple vehicles per account.

Primary Actor: Account Holder

Secondary Actors: System (registers vehicles and issues transponders)

Main Success Scenario:

1. Account Holder provides vehicle details and requests a transponder.
2. System registers the vehicle, leases a transponder, and links it to the account.

Extensions & Exceptions:

1a. If the vehicle is already registered or details are incorrect, the system notifies the Account Holder.

3. Use Case: RecordTrip

Scope: 407 ETR System

Level: User Goal

Intention in Context: To automatically record entry and exit of registered vehicles and calculate tolls.

Multiplicity: One record per trip.

Primary Actor: System

Secondary Actors: Transponder (provides vehicle identification), Gantries (record entry/exit)

Main Success Scenario:

1. Vehicle passes under entry gantry; system reads transponder, records entry time, date, and location.

2. Vehicle passes under exit gantry; system records exit and matches with entry record.

3. System calculates toll based on distance, time, and vehicle class, and charges the account holder's account.

Extensions & Exceptions:

1a. If the transponder is not read, classify the vehicle as unregistered and initiate unregistered vehicle process.

4. Use Case: ProcessUnregisteredVehicle

Scope: 407 ETR System

Level: User Goal

Intention in Context: To identify unregistered vehicles, capture license plates, classify vehicles, and issue toll charges.

Multiplicity: One process per unregistered vehicle.

Primary Actor: System

Secondary Actors: Cameras (capture license plate), Lights (ensure visibility), Government Records (identify owner)

Main Success Scenario:

1. Cameras capture the rear number plate images as the vehicle passes under the gantry.

2. System classifies the vehicle and retrieves owner information from government records.

3. System issues a toll invoice to the vehicle owner, including an additional video toll charge.

Extensions & Exceptions:

1a. If image capture or processing fails, a human operator reviews the images to determine the license plate.

5. Use Case: IssueInvoice

Scope: 407 ETR System

Level: User Goal

Intention in Context: To issue monthly invoices to registered account holders and unregistered vehicle owners.

Multiplicity: One invoice per account or vehicle per month.

Primary Actor: System

Secondary Actors: Account Holder, Vehicle Owner

Main Success Scenario:

1. System compiles trip records for the month.
2. System calculates total toll charges, including any video toll charges.
3. System sends invoice to the account holder or vehicle owner.

Extensions & Exceptions:

3a. If the account holder disputes charges, the system processes the dispute and adjusts the invoice if necessary.

6. Use Case: EnforceNonPayment

Scope: 407 ETR System

Level: User Goal

Intention in Context: To enforce penalties on cheating motorists and handle non-payment of invoices or fines.

Multiplicity: Varies based on non-compliance cases.

Primary Actor: System

Secondary Actors: Government (enforces plate denial)

Main Success Scenario:

1. System identifies unpaid invoices or fines after the due date.
2. System flags the account/vehicle for enforcement.
3. System coordinates with government authorities for plate denial until dues are cleared.

Extensions & Exceptions:

1a. If payment (**NOTE: ChatGPT stopped generating here and only provided the rest after being asked to continue)** is made after the flagging, update the account/vehicle status and inform government authorities to lift the enforcement.

1b. If the motorist contests the fine, the system processes the dispute and takes necessary action based on the outcome.

Online Cafeteria System

1. Use Case: BrowseMenus

Scope: Online Cafeteria System

Level: User Goal

Intention in Context: The Employee wants to view available lunch menus several days in advance, including ingredients, price, and configuration options.

Multiplicity: Many Employees can browse menus simultaneously.

Primary Actor: Employee

Secondary Actors: Caterer (provides menu information)

Main Success Scenario:

1. Employee accesses the Online Cafeteria System.
2. System displays available menus with details.

Extensions & Exceptions:

2a. System cannot retrieve menu data: Show error and suggest retrying later.

2. Use Case: PlaceOrder

Scope: Online Cafeteria System

Level: User Goal

Intention in Context: The Employee wants to place an order for a meal, specifying quantity and meal customizations if available.

Multiplicity: One order per Employee at a time.

Primary Actor: Employee

Secondary Actors: System (processes order), Caterer (receives order)

Main Success Scenario:

1. Employee selects meal and specifies quantity and customizations.
2. System confirms availability and records the order.
3. System sends order details to the Caterer.

Extensions & Exceptions:

2a. Desired meal is not available: Notify Employee and suggest alternatives.

2b. Employee is not authenticated: Prompt for login.

3. Use Case: ModifyOrCancelOrder

Scope: Online Cafeteria System

Level: User Goal

Intention in Context: The Employee wants to modify or cancel their order up to midnight on the previous day.

Multiplicity: One modification or cancellation per order.

Primary Actor: Employee

Secondary Actors: System (processes changes), Caterer (updates order)

Main Success Scenario:

1. Employee requests to modify or cancel an existing order.
2. System checks if the modification/cancellation window is still open.
3. System updates the order as per the Employee's request and notifies the Caterer.

Extensions & Exceptions:

2a. Modification window has passed: Notify Employee and deny changes.

3a. Modified order cannot be fulfilled: Notify Employee and suggest alternatives.

4. Use Case: ProcessDelivery

Scope: Online Cafeteria System

Level: User Goal

Intention in Context: The Caterer prepares and delivers meals to the company's cafeteria, labeled with barcodes, meal price, and Employee name.

Multiplicity: Multiple meals delivered each day.

Primary Actor: Caterer

Secondary Actors: System (provides order details), Employee (receives meal)

Main Success Scenario:

1. Caterer prepares meals based on received orders.
2. Meals are packaged and labeled with necessary details.
3. Caterer delivers meals to the company's cafeteria.

Extensions & Exceptions:

3a. Delivery issues (e.g., delays, damaged goods): Notify System and seek resolution.

5. Use Case: CompleteTransaction

Scope: Online Cafeteria System

Level: User Goal

Intention in Context: The Employee wants to pick up their meal and complete the transaction, using cash, credit card, or charging to their account.

Multiplicity: One transaction per Employee per meal.

Primary Actor: Employee

Secondary Actors: Cashier (handles transaction), System (records transaction)

Main Success Scenario:

1. Employee presents meal barcode at the cafeteria.
2. Cashier scans the barcode and displays the price.
3. Employee chooses payment method and completes the transaction.

Extensions & Exceptions:

3a. Payment failure: Notify Employee and request alternative payment method.

6. Use Case: ManageUnpickedOrders

Scope: Online Cafeteria System

Level: User Goal

Intention in Context: The System manages orders that are not picked up by 2:30pm, offering them at a discounted price or charging the original orderer.

Multiplicity: Varies based on unpicked orders.

Primary Actor: System

Secondary Actors: Employee (potential buyers of discounted meals or charged for unpicked meals), Cashier (handles transactions)

Main Success Scenario:

1. System identifies orders not picked up by the cutoff time.
2. System offers unpicked meals at a discounted price.
3. For meals not sold, System charges the cost to the original orderer

Smart Store System

Summary Level Use Case: OperateSmartStore
- **Scope:** SmartStoreSystem
- **Level:** Summary
- **Intention in Context:** "To facilitate all operations within the Smart Store system."

- **Multiplicity:** Multiple customers and staff operating concurrently.
- **Primary Actor:** System
- **Main Success Scenario:**
  - Customer Registration
  - Customer Entry
  - Item Recognition
  - Customer Exit and Payment
  - Staff Maintenance

User-Goal Level Use Case: Customer Registration
- **Level:** User-Goal
- **Intention in Context:** "Customers register to use the Smart Store services."
- **Primary Actor:** Customer
- **Secondary Actors:** Mobile App, Registration System
- **Main Success Scenario:**
  1. Customer downloads and opens the Smart Store app.
  2. Customer inputs personal and payment information.
  3. System verifies and creates customer account.
- **Extensions:**
  - If registration fails, System prompts Customer to retry.

User-Goal Level Use Case: Customer Entry
- **Level:** User-Goal
- **Intention in Context:** "Customers enter the store for shopping."
- **Primary Actor:** Customer
- **Secondary Actors:** Entry Gate, Identification System
- **Main Success Scenario:**
  1. Customer scans registered device at Entry Gate.
  2. System validates Customer and grants entry.
- **Extensions:**
  - If authentication fails, System denies entry and requests re-scan.

Sub-Function Level Use Case: Item Recognition
- **Level:** Sub-Function
- **Intention in Context:** "Identify items picked by Customers."
- **Primary Actor:** Sensor::ItemSensor
- **Secondary Actors:** Tag Reader, Pressure Sensor, Weight Sensor
- **Main Success Scenario:**
  1. Sensor detects item removal from shelf.
  2. System updates Customer's virtual cart.
- **Extensions:**
  - If Sensor fails, System logs error and alerts staff.

User-Goal Level Use Case: Customer Exit and Payment

- **Level:** User-Goal
- **Intention in Context:** "Process Customer's payment upon exit."
- **Primary Actor:** Customer
- **Secondary Actors:** Payment System, Exit Gate
- **Main Success Scenario:**
  1. Customer approaches Exit Gate and scans device.
  2. System calculates bill and processes payment.
  3. Exit Gate opens post successful payment.
- **Extensions:**
  - If payment fails, System prompts for an alternative method.

User-Goal Level Use Case: Staff Maintenance
- **Level:** User-Goal
- **Intention in Context:** "Staff perform store maintenance and restocking."
- **Primary Actor:** Staff
- **Secondary Actors:** Inventory System
- **Main Success Scenario:**
  1. Staff restocks items and performs maintenance tasks.
  2. System updates inventory and logs maintenance activities.
- **Extensions:**
  - If System detects stocking errors, alerts staff for correction.

Smart Fire Alarm

Use Case Name: DetectAndAlertFire

- Scope: Smart Fire Alarm System
- Level: User goal
- Intention in Context: To detect a fire quickly and alert the User and local fire department within 30 seconds.
- Multiplicity: One per fire event detected.
- Primary Actor: Fire Sensor (Smoke, Heat, Carbon Monoxide)
- Secondary Actors: User, Local Fire Department, Emergency Contact
- Main Success Scenario:
    - Fire Sensor detects smoke, heat, or carbon monoxide.

        - System verifies the fire event to reduce false alarms.
        - Alarm is sounded.
        - System sends an alert to the User and attempts to contact the local fire department.
        - If the system fails to contact the fire department, it notifies the User of the failure.
        - If the User does not acknowledge the notification within a specified time, the system escalates the alert to an Emergency Contact.
- Extensions & Exceptions:
    - 1a. False alarm: System allows the User to manually deactivate the alarm.
    - 4a. Communication failure: If unable to reach the fire department, the system immediately informs the User.
    - 6a. No acknowledgment from the User: Alert is escalated to the Emergency Contact.

Use Case Name: ActivateSprinklerResponse

- Scope: Smart Fire Alarm System
- Level: Subfunction
- Intention in Context: To lower the temperature threshold of the sprinkler system for an early response to fire.
- Multiplicity: Triggered upon fire detection.
- Primary Actor: Sprinkler System
- Secondary Actors: Fire Sensor
- Main Success Scenario:
    - Fire detected by the Fire Sensor.
    - System notifies the Sprinkler System of the fire event.
    - Sprinkler System lowers its temperature threshold based on the detected fire's location.

        - Sprinklers activate to control or extinguish the fire early.
- Extensions & Exceptions:
    - 2a. Sprinkler system fails to activate: System notifies the User and advises manual intervention.
    - 4a. Unnecessary sprinkler activation: User has the capability to adjust or deactivate the sprinkler system.

Use Case Name: SystemHealthCheck

- Scope: Smart Fire Alarm System
- Level: System function
- Intention in Context: To ensure all sensors and the battery are functioning correctly to maintain system reliability.
- Multiplicity: Continuous or periodic checks.
- Primary Actor: System Health Monitor
- Secondary Actors: User
- Main Success Scenario:
    - System performs regular checks on the smoke, heat, and carbon monoxide sensors, as well as battery levels.
    - If a low battery or sensor failure is detected, the system notifies the User.
    - User receives instructions for addressing the issue.
- Extensions & Exceptions:
    - 2a. User does not respond to maintenance alerts: System sends repeated notifications until the issue is addressed.

UCM4IoT Use Case Models

Summary Level Use Case: ManageSmartStoreOperations

- Scope: SmartStoreSystem
- Level: Summary
- Intention: "Coordinate all operations within the Smart Store."
- Multiplicity: Multiple customers and staff interact concurrently.
- Primary Actor: SYSTEM
- Main Success Scenario:
    - [CustomerRegistration]
    - [CustomerEntryProcessing]
    - [ItemIdentification]
    - [CustomerExitAndPayment]
    - [StaffActivityManagement]
- Extensions and Exceptions:
    - Network issues, sensor malfunctions, authentication failures, payment transaction failures, inventory mismatches.

User-Goal Level Use Case: CustomerRegistration

- Level: User-Goal
- Intention: "Enable customers to register for the Smart Store service."
- Primary Actor: HUMAN::Customer
- Secondary Actors: SOFTWARE::RegistrationApp, DATABASE::CustomerDatabase
- Main Success Scenario:
    - Customer downloads and registers on the App.
    - The System verifies and stores Customer data.
- Extensions:

    - 1a. {NETWORK_EXCEPTION::ConnectivityIssue} - Invoke "NetworkIssueHandler."

User-Goal Level Use Case: CustomerEntryProcessing

- Level: User-Goal
- Intention: "Process customer entry through identification at the gate."
- Primary Actor: HUMAN::Customer
- Secondary Actors: SENSOR::EntryGateScanner, SOFTWARE::IdentityVerification
- Main Success Scenario:
    - Customer scans their device at the Entry Gate.
    - The System authenticates and allows entry.
- Extensions:
    - 2a. {IDENTIFICATION_EXCEPTION::AuthenticationFailure} - Invoke "AuthenticationIssueHandler."

Sub-Function Level Use Case: ItemIdentification

- Level: Sub-Function
- Intention: "Identify items picked up by customers."
- Primary Actor: SENSOR::ItemSensor
- Secondary Actors: DATABASE::InventoryDatabase
- Main Success Scenario:
    - Sensors detect item interaction.
    - The System updates the Customer's virtual cart.
- Extensions:
    - 1a. {SENSOR_EXCEPTION::Malfunction} - Invoke "SensorMalfunctionHandler."

User-Goal Level Use Case: CustomerExitAndPayment

- Level: User-Goal
- Intention: "Manage customer checkout and payment process."

- Primary Actor: HUMAN::Customer
- Secondary Actors: SOFTWARE::PaymentProcessor, SENSOR::ExitGateScanner
- Main Success Scenario:
    - Customer approaches Exit Gate for checkout.
    - The System processes payment and opens the gate.
- Extensions:
    - 2a. {PAYMENT_EXCEPTION::TransactionFailure} - Invoke "PaymentIssueHandler."

User-Goal Level Use Case: StaffActivityManagement

- Level: User-Goal
- Intention: "Coordinate staff activities for store maintenance and assistance."
- Primary Actor: HUMAN::Staff
- Secondary Actors: SOFTWARE::MaintenanceScheduler, DATABASE::StaffDatabase
- Main Success Scenario:
    - Staff performs restocking and maintenance tasks.
    - The System logs activities and updates inventory.
- Extensions:
    - 2a. {INVENTORY_EXCEPTION::Mismatch} - Invoke "InventoryMismatchHandler."

Handler Use Cases

Handler Use Case: NetworkIssueHandler

- Main Success Scenario:

    The System attempts to re-establish network connectivity.

    On reconnection, resumes pending processes.

Handler Use Case: AuthenticationIssueHandler

- Main Success Scenario:

    The System prompts the Customer to re-authenticate.

    On successful authentication, grants access.

Handler Use Case: SensorMalfunctionHandler

- Main Success Scenario:

    The System alerts staff to check and fix the sensor.

    Resumes item identification once fixed.

Handler Use Case: PaymentIssueHandler

- Main Success Scenario:

    The System requests an alternative payment method.

    Processes the payment upon successful transaction.

Handler Use Case: InventoryMismatchHandler

- Main Success Scenario:

    The System alerts staff for inventory verification.

    Staff corrects inventory data in the System.

Smart Fire Alarm

Use Case Name: FireDetectionAndNotification

- Scope: Smart Fire Alarm System
- Level: SUMMARY
- Intention: "Detect fire using advanced sensors and immediately notify the User, fire department, and emergency contacts."
- Multiplicity: "Activated for each detected fire incident."
- Primary Actor: SENSOR::FireDetectionSensors::1..*

- Secondary Actor: HUMAN::User::1.., *HUMAN::FireDepartment::1..1, SOFTWARE::NotificationSystem::..**
- Main Success Scenario:
    - "FireDetectionSensors detect signs of fire, including smoke, heat, or carbon monoxide."
    - "The System analyzes sensor data to confirm fire presence."
    - "Upon confirmation, the System activates the alarm and sends an alert to the User's mobile app."
    - "Simultaneously, the System attempts to notify the local Fire Department."
    - "If the System fails to reach the Fire Department, it informs the User and tries to contact the Emergency Contact."
    - "The System monitors User acknowledgment; lack thereof triggers notification to Emergency Contact."
- Extensions:

alternative for 4:

    - 4a. "The System uses an alternative communication channel if the primary fails."

exception for (1-6):

    - (1-6)a.^ timeout:30s "{NETWORK_EXCEPTION::WiFiUnreachable} In case of network failure, attempt to reconnect."

Use Case Name: SprinklerSystemActivation

- Scope: Smart Fire Alarm System
- Level: SUMMARY
- Intention: "To activate the sprinkler system upon fire detection to control or extinguish the fire."

- Multiplicity: "Triggered by fire detection."
- Primary Actor: ACTUATOR::SprinklerSystem::1..1
- Secondary Actor: SENSOR::FireDetectionSensors::1..*
- Main Success Scenario:
    - "FireDetectionSensors detect fire and send a signal to the Sprinkler System."
    - "The System lowers the Sprinkler System's temperature threshold for early activation."
    - "Sprinkler System activates to control or extinguish the fire."
- Extensions:

alternative for 2:

    - 2a. "If temperature adjustment fails, manually activate Sprinkler System."

exception for 1:

    - 1a. "{HARDWARE_EXCEPTION::SprinklerSystemFailure} If the Sprinkler System fails to activate, notify User for manual intervention."

Use Case Name: SystemMaintenanceAlert

- Scope: Smart Fire Alarm System
- Level: SUMMARY
- Intention: "To ensure the system and sensors are functioning correctly by alerting the User of maintenance needs."
- Multiplicity: "Regular checks ensure continuous protection."
- Primary Actor: SOFTWARE::MaintenanceMonitor::**..**
- Secondary Actor: HUMAN::User::1..*
- Main Success Scenario:

    - "MaintenanceMonitor conducts regular diagnostic checks on sensors and system battery."
    - "If a low battery or sensor failure is detected, the System alerts the User."
    - "The User receives actionable advice for resolving the detected issue."
- Extensions:

alternative for 2:

    - 2a. "User schedules a maintenance check through the app if unable to resolve."

exception for 1:

    - 1a. "{HARDWARE_EXCEPTION::SensorFailure} Immediate notification to the User for critical sensor failure."

Handler Use Case Name: NetworkRecoveryHandler

- Scope: Smart Fire Alarm System
- Level: SUMMARY
- Intention: "To restore network connectivity when alerts cannot be sent due to network issues."
- Multiplicity: "Applicable to all network-related exceptions."
- Primary Actor: None
- Contexts and Exceptions: FireDetectionAndNotification {NETWORK_EXCEPTION::WiFiUnreachable}, SprinklerSystemActivation {NETWORK_EXCEPTION::WiFiUnreachable}
- Main Success Scenario:
    - "The System detects network connectivity issues preventing alert notifications."
    - "Attempts to reconnect to the WiFi/network."

        - "Upon re-establishment of network connection, resumes pending alert notifications."
- Extensions:

alternative for 2:

        - 2a. "If unable to reconnect within a predefined timeout, switch to an alternative communication method (e.g., cellular)."

Handler Use Case Name: SprinklerFailureHandler

- Scope: Smart Fire Alarm System
- Level: SUMMARY
- Intention: "To manage situations where the Sprinkler System fails to activate."
- Multiplicity: "Activated upon detection of sprinkler system failure."
- Primary Actor: None
- Contexts and Exceptions: SprinklerSystemActivation {HARDWARE_EXCEPTION::SprinklerSystemFailure}
- Main Success Scenario:
    - "System identifies failure in Sprinkler System activation."
    - "Notifies the User of the failure and advises manual intervention."
- Extensions:

alternative for 1:

        - 1a. "System triggers a secondary safety measure, such as activating fire retardant dispensers."

Handler Use Case Name: SensorMaintenanceHandler

- Scope: Smart Fire Alarm System

- Level: SUMMARY
- Intention: "To prompt immediate action when a critical sensor failure is detected."
- Multiplicity: "Initiated upon sensor check revealing issues."
- Primary Actor: None
- Contexts and Exceptions: SystemMaintenanceAlert {HARDWARE_EXCEPTION::SensorFailure}
- Main Success Scenario:
    - "Maintenance Monitor detects a critical sensor failure."
    - "System immediately alerts the User, detailing the nature of the failure and steps for resolution."
- Extensions:

alternative for 2:

    - 2a. "If the User does not acknowledge the alert within a set timeframe, escalate to a registered technician or emergency contact."