# P1B1 S2G1

Authors: Robert Kemp, Matthew Nguyen, Tiehang Zheng, Rishi Jeswani

## Comparison:

Between ChatGPT-4 and Claude Sonnet 4, we employed an identical pipeline for document analysis—utilizing the same Python script for chunking and indexing, Facebook AI's Sentence Transformation for semantic embedding, and consistent chunk passing. The only variation lay in the LLM-specific libraries: OpenAI for ChatGPT and Anthropic for Claude. This ensured that both models received the same input and prompt structure. Notably, ChatGPT-4 generated 20 distinct use cases from the provided text, each closely mirroring the original source. Its responses were precise, with citations directly linked to their origins. In contrast, Claude Sonnet 4 synthesized and expanded upon the material, merging related examples into broader, more nuanced use cases. Claude's output was more detailed, featuring structured elements like *Actors*, *Subflows*, and *Alternative Flows*, though it included fewer citations overall.

In summary, ChatGPT-4 excelled in **accuracy** and **direct information retrieval**, making it well-suited for tasks requiring strict fidelity to source material. Claude Sonnet 4, however, demonstrated a stronger ability to **synthesize**, **infer**, and **conceptually organize** information—producing responses that resembled the work of a creative professional refining a set of requirements. Each model brought distinct strengths to the table, with ChatGPT acting as a meticulous researcher and Claude as a systems-level designer.

## Cost:

ChatGPT-4: $1.30
Claude Sonnet 4: $0.18

ChatGPT-4 was used for initial testing and set up of the Python Script for RAGs, so its significantly higher price is not a good measurement of a cost to output analysis as it's also the result of 12 total requests versus 4 total requests between ChatGPT and Claude respectively.

---

# 📑 Use Case Specifications — Food Delivery System

---

## UC-01: Browse and Search Restaurants

**Actors:** Customer
**Description:** Customer searches and browses available restaurants with dynamic filtering capabilities.

**Preconditions:**

- Customer is logged into the platform
- Internet connection available

**Main Flow:**

1. Customer opens the app/web platform
2. System displays dynamic list of nearby restaurants based on availability
3. Customer applies filters (cuisine, rating, delivery time, price range)
4. System shows updated list with interactive search suggestions
5. Customer views restaurant profiles with real-time availability

**Subflows:**

- **Interactive Search:** Customer types restaurant/dish name → system provides suggestions even with typos
- **Apply Multiple Filters:** Customer combines cuisine, price, rating filters → system updates results
- **Location-Based Sorting:** System sorts by distance using GPS data
- **Availability Check:** System shows only open restaurants with real-time status updates

**Alternate Flows:**

- **A1:** No restaurants available → system suggests nearby areas or schedule for later
- **A2:** Search yields no results → system suggests alternative dishes/restaurants
- **A3:** Location services disabled → system prompts for manual location entry

**Postconditions:**

- Customer views available restaurant options with accurate information

---

# UC-02: Place Food Order with Customization

**Actors:** Customer, Restaurant, Payment Provider
**Description:** Customer selects items, customizes order, and completes checkout process.

**Preconditions:**

- Customer has active account

- Selected restaurant is open
- Valid payment method available

**Main Flow:**

1. Customer adds items to cart with customization options
2. Customer reviews cart and modifies quantities/customizations
3. Customer selects delivery address from saved addresses or enters new one
4. Customer chooses payment method and applies any promo codes
5. System validates order against restaurant availability
6. Customer confirms order and payment is processed
7. Order is sent to restaurant with customer and delivery details

**Subflows:**

- **Customize Items:** Customer adds special instructions → system validates with restaurant capabilities
- **Apply Promotions:** Customer enters promo code → system validates and recalculates total
- **Address Validation:** System checks delivery zone coverage and estimates delivery time
- **Payment Processing:** System integrates with payment provider for secure transaction

**Alternate Flows:**

- **A1:** Payment fails → system offers alternative payment methods
- **A2:** Item becomes unavailable → restaurant suggests substitution
- **A3:** Address outside delivery zone → system suggests pickup option

**Postconditions:**

- Order confirmed
- Payment processed
- Restaurant notified

---

# UC-03: Track Order in Real Time

**Actors:** Customer, Driver, Restaurant
**Description:** Customer tracks their order through preparation, dispatch, and delivery stages.

**Preconditions:**

- Customer has placed an order
- Driver has been assigned

**Main Flow:**

1. System shows order confirmation with estimated preparation and delivery time
2. Customer receives updates at key stages (accepted, preparing, ready, en route)
3. System provides live GPS tracking of driver once order is picked up
4. ETA dynamically updates based on driver's location

**Subflows:**

- **Preparation Updates:** Restaurant updates status → system notifies customer
- **Driver Assignment:** System notifies customer of driver's identity and vehicle details
- **Live GPS Tracking:** Customer views map with driver location and ETA
- **Delay Alerts:** System recalculates ETA if traffic or delays occur

**Alternate Flows:**

- **A1:** No driver available → system informs customer and offers cancellation/refund
- **A2:** Driver cancels → system reassigns driver and updates customer
- **A3:** System error in tracking → fallback ETA is provided without GPS

**Postconditions:**

- Customer successfully monitors order status until delivery completion

---

# UC-04: Accept and Fulfill Order

**Actors:** Restaurant Staff, Customer
**Description:** Restaurant receives and processes an order to prepare it for delivery.

**Preconditions:**

- Restaurant is logged into the platform
- Customer order received

**Main Flow:**

1. Restaurant receives notification of new order
2. Staff reviews order details, customizations, and notes
3. Staff confirms acceptance of order
4. Food preparation begins and progresses to completion
5. Restaurant marks order as "Ready for Pickup"

**Subflows:**

- **Queue Management:** Orders displayed in preparation queue in chronological order
- **Substitutions:** If ingredient unavailable, staff suggests alternatives → customer confirms/declines
- **Batch Orders:** Restaurant groups orders with similar prep time for efficiency

**Alternate Flows:**

- **A1:** Restaurant rejects order → system informs customer and issues refund
- **A2:** Staff accidentally accepts wrong order → admin support required for correction

**Postconditions:**

- Food is prepared and ready for delivery driver pickup

---

# UC-05: Assign and Dispatch Driver

**Actors:** Driver, System, Restaurant
**Description:** System assigns drivers to orders based on location and availability.

**Preconditions:**

- Order has been accepted by restaurant
- Drivers are active on the platform

**Main Flow:**

1. System searches for available drivers near restaurant
2. System sends job request to nearest driver
3. Driver accepts assignment
4. Driver navigates to restaurant for pickup

**Subflows:**

- **Driver Matching:** System evaluates distance, availability, and driver rating
- **Batch Dispatch:** System may assign multiple orders in same route to one driver
- **Auto-Reassignment:** If driver rejects, system reassigns to next available driver

**Alternate Flows:**

- **A1:** No drivers accept → system alerts customer and extends delivery time
- **A2:** Driver cancels after acceptance → new driver reassigned

**Postconditions:**

- Driver assigned and notified to pick up food

---

# UC-06: Pickup and Deliver Food

**Actors:** Driver, Customer, Restaurant
**Description:** Driver collects the prepared order from the restaurant and delivers it to the customer.

**Preconditions:**

- Driver has been assigned to the order
- Restaurant has marked the order as “Ready for Pickup”

**Main Flow:**

1. Driver arrives at the restaurant
2. Driver verifies order ID with restaurant staff
3. Restaurant staff hands food to driver
4. Driver confirms pickup in system
5. Driver navigates to customer’s address
6. Driver hands food to customer and marks delivery as complete

**Subflows:**

- **Restaurant Handoff:** Driver shows order code → restaurant validates before handoff
- **Customer Contact:** Driver calls/messages customer if clarification is needed
- **Proof of Delivery:** Driver captures photo/signature for delivery confirmation
- **Tip Collection:** Customer adds tip at or after delivery

**Alternate Flows:**

- **A1:** Customer unavailable → driver retries or contacts support
- **A2:** Incorrect order at pickup → driver/restaurant escalate to admin
- **A3:** Delivery address unclear → driver contacts customer for clarification

**Postconditions:**

- Order is successfully delivered
- System marks order as completed

---

# UC-07: Handle Payments and Refunds

**Actors:** Customer, Payment Provider, Admin
**Description:** System manages payment authorization, processing, and potential refunds.

**Preconditions:**

- Customer has valid payment method on file
- Payment provider services are operational

**Main Flow:**

1. Customer selects payment method during checkout
2. System sends transaction request to payment provider
3. Payment provider validates and authorizes funds
4. System confirms successful transaction and records it
5. Admin can later issue refunds when required

**Subflows:**

- **Transaction Authorization:** Provider checks card validity and funds
- **Split Payments:** System processes promo codes, gift cards, or multiple payment methods
- **Refund Request:** Customer requests refund → admin reviews
- **Partial Refunds:** Admin approves refund for missing/incorrect items

**Alternate Flows:**

- **A1:** Payment fails → customer prompted to use alternate method
- **A2:** Chargeback raised → admin investigates with evidence
- **A3:** Payment provider service outage → order cannot be completed

**Postconditions:**

- Successful transactions recorded
- Refunds processed when necessary

---

# UC-08: Rate and Review Experience

**Actors:** Customer, Restaurant, Driver, Admin
**Description:** Customer rates and reviews both the restaurant and driver after delivery.

**Preconditions:**

- Order completed
- Customer logged in

**Main Flow:**

1. System prompts customer to leave feedback after delivery
2. Customer rates driver (1–5 stars)
3. Customer rates restaurant for food quality and packaging
4. Customer may add written comments
5. System stores feedback and updates average ratings

**Subflows:**

- **Driver Rating:** Customer rates driver performance → visible to driver
- **Restaurant Review:** Customer comments on food/service quality
- **Reply to Review:** Restaurant or driver can respond to reviews
- **Review Moderation:** Inappropriate content flagged for admin review

**Alternate Flows:**

- **A1:** Customer skips feedback → system closes review process
- **A2:** Customer leaves abusive review → flagged for moderation
- **A3:** Review disputed → admin intervenes and resolves

**Postconditions:**

- Ratings and reviews updated in system
- Feedback available for future customers

---

# UC-09: Manage Menus and Promotions

**Actors:** Restaurant Staff, Admin
**Description:** Restaurants update menu items and create promotional offers.

**Preconditions:**

- Restaurant has active account
- Staff logged into system

**Main Flow:**

1. Restaurant staff log into portal
2. Staff add, edit, or remove menu items
3. Staff upload descriptions, prices, and photos
4. Staff set availability (in-stock, out-of-stock, scheduled)
5. Staff create promotions or discounts

**Subflows:**

- **Add/Edit Menu Items:** Restaurant enters details → system validates fields
- **Set Availability:** Items marked available/unavailable in real time
- **Create Promotion:** Staff set discount terms → system validates rules
- **Approval Workflow:** Certain promotions require admin review

**Alternate Flows:**

- **A1:** Invalid menu entry → system rejects update
- **A2:** Duplicate promotion code → system requests different code
- **A3:** Admin disapproves promotion → restaurant notified

**Postconditions:**

- Menu and promotions updated and visible to customers

---

# UC-10: Administer and Monitor Platform

**Actors:** Admin, Customers, Drivers, Restaurants
**Description:** Admin manages accounts, monitors platform activity, and ensures compliance.

**Preconditions:**

- Admin logged into system with appropriate privileges

**Main Flow:**

1. Admin logs into the system dashboard
2. Admin views reports and analytics on platform usage
3. Admin manages user accounts (add, suspend, deactivate)
4. Admin reviews and resolves disputes
5. Admin monitors compliance with regulations

**Subflows:**

- **User Management:** Admin edits/suspends customer, driver, or restaurant accounts
- **Fraud Detection:** Suspicious activity flagged for admin investigation
- **Dispute Resolution:** Admin mediates between parties and issues refunds if needed
- **Audit Logging:** All admin actions recorded for accountability
- **Compliance Reporting:** Admin generates tax/food safety/data privacy reports

**Alternate Flows:**

- **A1:** Unauthorized admin access → system denies entry and logs attempt
- **A2:** Dispute unresolved → escalated to higher authority
- **A3:** Data access request denied due to insufficient privileges

**Postconditions:**

- Platform remains secure, compliant, and well-administered

---

# UC-11: Register New User Account

**Actors:** Customer, Driver, Restaurant Staff
**Description:** A new user creates an account to access the platform.

**Preconditions:**

- User has a valid email/phone number
- System services available

**Main Flow:**

1. User opens the app or website
2. User selects “Create Account” option
3. User provides required details (name, email, phone, password)
4. System sends verification code to email/phone
5. User enters verification code
6. System confirms account creation and grants access

**Subflows:**

- **Profile Setup:** User adds address, payment method, and preferences after registration
- **Driver Registration:** Drivers provide license, vehicle, and insurance details for approval
- **Restaurant Registration:** Restaurants provide business license and upload initial menu

**Alternate Flows:**

- **A1:** Verification code not received → system resends or offers alternative method
- **A2:** Duplicate email/phone detected → system prompts user to log in instead
- **A3:** Invalid input (weak password, missing fields) → system requests corrections

**Postconditions:**

- User account created and ready for use

---

# UC-12: Log In and Authenticate User

**Actors:** Customer, Driver, Restaurant Staff, Admin
**Description:** User accesses their account securely through authentication.

**Preconditions:**

- User already has a registered account

**Main Flow:**

1. User opens app/website
2. User enters email/phone and password
3. System verifies credentials
4. System grants access to dashboard or home screen

**Subflows:**

- **Two-Factor Authentication:** System requests one-time code for added security
- **Social Login:** User logs in with Google, Apple, or Facebook account

**Alternate Flows:**

- **A1:** Incorrect credentials → system shows error message and retry option
- **A2:** Multiple failed attempts → system locks account temporarily
- **A3:** Forgotten password → user requests reset link

**Postconditions:**

- User successfully logged in with secured access

---

# UC-13: Update User Profile

**Actors:** Customer, Driver, Restaurant Staff
**Description:** User updates personal details, payment methods, or preferences.

**Preconditions:**

- User logged into account

**Main Flow:**

1. User navigates to profile settings
2. User updates personal info (name, email, phone, address)

3. User updates payment methods or delivery preferences
4. System validates and saves updates

**Subflows:**

- **Customer Profile:** Customer adds new addresses, updates saved cards
- **Driver Profile:** Driver updates vehicle information, insurance documents
- **Restaurant Profile:** Restaurant updates contact details, operating hours

**Alternate Flows:**

- **A1:** Invalid data entry → system rejects and prompts correction
- **A2:** Payment method expired → system requires replacement
- **A3:** Profile update conflicts with platform policy (e.g., fake info) → admin review required

**Postconditions:**

- Updated profile information stored and visible in system

---

# UC-14: Customer Support Interaction

**Actors:** Customer, Driver, Restaurant, Support Agent, Admin
**Description:** Users contact support for order-related or technical issues.

**Preconditions:**

- User logged into account or provides order reference

**Main Flow:**

1. User navigates to “Help/Support” section
2. User selects type of issue (order, payment, technical)
3. System suggests automated FAQs or solutions
4. If unresolved, case escalates to support agent
5. Support agent communicates with user and resolves issue

**Subflows:**

- **Chat Support:** User engages with live chat or chatbot
- **Phone/Email Support:** User chooses alternate contact methods
- **Escalation:** If issue unresolved, escalated to admin

**Alternate Flows:**

- **A1:** User abandons support request → case closed automatically
- **A2:** Issue requires investigation (fraud, missing delivery) → longer resolution time
- **A3:** Duplicate complaint detected → system merges tickets

**Postconditions:**

- User receives assistance or issue resolved
- Support case recorded for tracking

---

# UC-15: Driver Availability Management

**Actors:** Driver, System
**Description:** Drivers set their availability to receive or stop receiving delivery requests.

**Preconditions:**

- Driver has active and approved account
- Driver logged into platform

**Main Flow:**

1. Driver opens app
2. Driver toggles availability status (online/offline)
3. System updates driver's status in dispatch algorithm
4. System only assigns orders to available drivers

**Subflows:**

- **Scheduled Availability:** Driver sets working hours in advance
- **Auto-Timeout:** If driver ignores requests repeatedly, system sets status to offline

**Alternate Flows:**

- **A1:** Driver tries to go online without required documents (insurance, license) → system blocks
- **A2:** Connectivity issues → system shows driver as offline until resolved
- **A3:** Driver attempts to go offline during an active order → system prevents until delivery complete

**Postconditions:**

- Driver availability status updated and reflected in system

---

# UC-16: Driver Navigation Assistance

**Actors:** Driver, System (Maps API)
**Description:** Driver uses in-app navigation to reach the restaurant and customer locations.

**Preconditions:**

- Driver logged in and assigned an order
- GPS and internet available

**Main Flow:**

1. Driver opens assigned order in app
2. System provides step-by-step navigation to restaurant
3. After pickup, system updates route to customer address
4. Driver follows directions until delivery is complete

**Subflows:**

- **Traffic Updates:** System reroutes if traffic or roadblocks are detected
- **Multi-Stop Route:** If multiple orders assigned, system provides optimized delivery sequence
- **Voice Navigation:** Driver enables hands-free voice guidance

**Alternate Flows:**

- **A1:** GPS signal lost → system provides fallback text instructions
- **A2:** Customer updates delivery address mid-route → system recalculates path
- **A3:** Driver deviates from route → system recalculates ETA

**Postconditions:**

- Driver successfully navigates to restaurant and customer location

---

# UC-17: Push Notifications and Alerts

**Actors:** Customer, Driver, Restaurant, System
**Description:** System sends real-time notifications to keep stakeholders updated.

**Preconditions:**

- User has enabled notifications on device

**Main Flow:**

1. Customer receives notifications about order acceptance, preparation, and delivery status
2. Driver receives alerts for new delivery requests and updates
3. Restaurant receives alerts for new orders
4. System ensures notifications are timely and accurate

**Subflows:**

- **Delivery Updates:** Customer notified when driver is nearby
- **Payment Alerts:** Customer notified of successful/failed payments
- **Promotion Notifications:** Customers receive personalized offers

**Alternate Flows:**

- **A1:** Notifications disabled → user only sees updates inside app
- **A2:** Network issues delay notifications → system queues and retries
- **A3:** User unsubscribes from promotions → system suppresses marketing alerts

**Postconditions:**

- Relevant users receive real-time alerts and updates

---

# UC-18: Manage Customer Addresses

**Actors:** Customer, System
**Description:** Customers add, update, and manage delivery addresses.

**Preconditions:**

- Customer logged into account

**Main Flow:**

1. Customer navigates to profile settings
2. Customer adds a new address (home, work, etc.)
3. System validates address format and coverage area
4. Customer saves updated address list

**Subflows:**

- **Geolocation Autofill:** Customer allows GPS → system auto-fills current address
- **Nickname Addresses:** Customer labels addresses (e.g., “Home,” “Office”)
- **Delivery Zone Check:** System validates address against serviceable areas

**Alternate Flows:**

- **A1:** Invalid address entry → system requests corrections
- **A2:** Address outside coverage area → system rejects or suggests pickup option
- **A3:** Duplicate address detected → system prompts to confirm merge

**Postconditions:**

- Updated address list stored and available for future orders

---

# UC-19: Promotional Campaign Management

**Actors:** Admin, Restaurant Staff, Marketing Partners
**Description:** Restaurants and admins create, manage, and monitor promotions.

**Preconditions:**

- Restaurant/admin logged into account with promo permissions

**Main Flow:**

1. Restaurant or admin navigates to promotions dashboard
2. User creates a new promotion (discount, coupon, free delivery)
3. System validates rules (dates, conditions, duplication)
4. System publishes promotion and applies it to orders

**Subflows:**

- **Discount Setup:** User sets percentage/flat-rate discount
- **Coupon Codes:** System generates unique codes for campaign
- **Targeting:** Promotions may apply to specific customers or areas

**Alternate Flows:**

- **A1:** Invalid promotion dates or values → system rejects setup
- **A2:** Duplicate promo code entered → system requests alternative
- **A3:** Promotion flagged as abuse (e.g., infinite discounts) → admin intervenes

**Postconditions:**

- Active promotions available to customers
- Orders apply valid promotions automatically

---

# UC-20: Generate Sales and Performance Reports

**Actors:** Admin, Restaurant Staff
**Description:** Users generate performance reports to analyze sales and delivery efficiency.

**Preconditions:**

- User has reporting permissions
- Data for selected period available

**Main Flow:**

1. User navigates to reporting dashboard
2. User selects report type (sales, orders, delivery times, driver performance)
3. User sets time period and filters (restaurant, region, etc.)
4. System generates report with analytics and visualizations

**Subflows:**

- **Export Reports:** User downloads report as CSV, PDF, or Excel
- **Automated Reports:** System sends scheduled reports to emails
- **Comparative Analysis:** User compares current vs. past performance

**Alternate Flows:**

- **A1:** No data available for period → system displays empty report
- **A2:** Server error during generation → system retries or notifies user
- **A3:** Unauthorized user attempts report generation → system denies access

**Postconditions:**

- Reports generated and available for viewing/exporting

---

# UC-21: Handle Payment Refunds

**Actors:** Customer, Restaurant, Admin, Payment Provider
**Description:** Customers request refunds for failed orders or unsatisfactory experiences.

**Preconditions:**

- Order already placed and paid for
- Refund policy applicable

**Main Flow:**

1. Customer initiates refund request through app
2. System validates order eligibility for refund
3. Refund request forwarded to restaurant/admin for review
4. Upon approval, system initiates refund with payment provider
5. Customer notified of refund status

**Subflows:**

- **Partial Refund:** Restaurant/admin approves partial amount (e.g., missing item)
- **Automatic Refund:** If order not prepared/delivered within time limit, system triggers automatic refund
- **Escalation:** If restaurant rejects request, customer escalates to admin

**Alternate Flows:**

- **A1:** Ineligible order → system informs customer of refund denial
- **A2:** Payment provider rejects refund → system retries or flags issue
- **A3:** Refund delayed → system provides estimated processing time

**Postconditions:**

- Refund processed or denial communicated to customer

---

# UC-22: Customer Reviews and Ratings

**Actors:** Customer, Restaurant, Driver, System
**Description:** Customers provide ratings and reviews after order completion.

**Preconditions:**

- Customer has completed order
- Review window still open

**Main Flow:**

1. Customer receives prompt to rate experience
2. Customer rates restaurant and/or driver
3. Customer optionally writes review or adds photos
4. System saves rating and updates overall averages

**Subflows:**

- **Anonymous Reviews:** Customer submits review without public name
- **Photo Reviews:** Customer uploads food images

- **Driver Feedback:** Customer rates delivery speed and professionalism

**Alternate Flows:**

- **A1:** Customer skips rating → system closes request
- **A2:** Inappropriate content detected → system flags for moderation
- **A3:** Review submitted late → system rejects input

**Postconditions:**

- Ratings/reviews stored and visible to other users

---

# UC-23: Driver Earnings and Payouts

**Actors:** Driver, Admin, Payment Provider
**Description:** Drivers view earnings and request payouts for completed deliveries.

**Preconditions:**

- Driver has completed deliveries
- Payment account connected

**Main Flow:**

1. Driver opens “Earnings” dashboard
2. System shows breakdown of completed deliveries and total earnings
3. Driver requests payout
4. System processes payout through linked bank or wallet
5. Driver receives confirmation of transaction

**Subflows:**

- **Daily/Weekly Reports:** Driver views earnings history by time period
- **Automatic Payouts:** System deposits earnings on fixed schedule
- **Incentive Tracking:** System shows bonuses and surge pay

**Alternate Flows:**

- **A1:** Payout request fails → system retries or informs driver
- **A2:** Invalid bank account details → system requests correction
- **A3:** Payout delayed due to verification → system provides ETA

**Postconditions:**

- Driver earnings successfully disbursed or pending resolution

---

# UC-24: Order Cancellation Management

**Actors:** Customer, Driver, Restaurant, Admin
**Description:** Stakeholders cancel orders under certain conditions.

**Preconditions:**

- Active order exists in system

**Main Flow:**

1. Customer, driver, or restaurant initiates cancellation
2. System checks cancellation rules (time, order status, stakeholder)
3. System applies refund rules where applicable
4. Relevant parties notified of cancellation

**Subflows:**

- **Customer Cancellation:** Allowed before restaurant confirms order
- **Driver Cancellation:** Allowed if within set time frame after assignment
- **Restaurant Cancellation:** Allowed if items unavailable or restaurant closed

**Alternate Flows:**

- **A1:** Cancellation requested after order prepared → partial refund only
- **A2:** Driver cancels mid-delivery → system reassigns to next available driver
- **A3:** Multiple cancellations from one party → system flags account for review

**Postconditions:**

- Order status updated to “Cancelled”
- Refunds or penalties applied as per policy

---

# UC-25: Multi-Order/Group Ordering

**Actors:** Customer, Restaurant, System
**Description:** Customers place group orders from one restaurant for multiple participants.

**Preconditions:**

- Restaurant supports group orders

**Main Flow:**

1. Customer initiates group order and invites participants
2. Participants add items to shared cart
3. System merges items into single order for restaurant
4. Customer confirms and pays for total order

**Subflows:**

- **Split Payments:** Participants pay for their own items individually
- **Host-Only Payment:** Host covers full order cost
- **Live Updates:** Participants see real-time cart changes

**Alternate Flows:**

- **A1:** Participant fails to confirm items in time → system excludes items
- **A2:** Payment split fails → host covers balance
- **A3:** Restaurant rejects large group order → system notifies participants

**Postconditions:**

- Group order placed and processed
- Restaurant receives consolidated request

---

# UC-26: Loyalty and Rewards Program

**Actors:** Customer, System, Admin
**Description:** Customers earn and redeem rewards points through a loyalty program.

**Preconditions:**

- Customer has active account
- Loyalty program is enabled

**Main Flow:**

1. Customer places an order
2. System calculates earned points based on order value
3. Points added to customer's account balance
4. Customer views available rewards in loyalty dashboard
5. Customer redeems points during checkout for discounts

**Subflows:**

- **Tiered Rewards:** Customers unlock higher tiers with extra perks
- **Expiration Rules:** System tracks and expires unused points after set time
- **Promo Boosts:** Special events grant bonus points

**Alternate Flows:**

- **A1:** Customer tries to redeem more points than available → system denies
- **A2:** Expired points → system removes from balance
- **A3:** Fraud detected in rewards usage → admin review required

**Postconditions:**

- Points updated and applied to customer account
- Rewards redeemed if chosen

---

# UC-27: Real-Time Order Tracking

**Actors:** Customer, Driver, System
**Description:** Customers track their orders in real time from preparation to delivery.

**Preconditions:**

- Order placed and accepted by restaurant
- Driver assigned

**Main Flow:**

1. Customer opens order tracking screen
2. System displays restaurant status (“Preparing,” “Ready for Pickup”)
3. Driver location shown on live map after pickup
4. System provides estimated delivery time with live updates

**Subflows:**

- **Push Updates:** System sends notifications at each status change
- **ETA Adjustments:** System recalculates ETA if route changes
- **Driver Contact:** Customer uses chat/call option to reach driver

**Alternate Flows:**

- **A1:** Driver location unavailable → system shows last known update
- **A2:** Restaurant delays preparation → system updates ETA

- **A3:** Customer cancels order mid-tracking → tracking disabled

**Postconditions:**

- Customer kept informed until order delivered or cancelled

---

# UC-28: Restaurant Performance Dashboard

**Actors:** Restaurant Staff, Admin
**Description:** Restaurants track business performance with real-time dashboards.

**Preconditions:**

- Restaurant account active
- Data available for reporting period

**Main Flow:**

1. Restaurant logs into dashboard
2. System displays key metrics: sales, orders, ratings, cancellations
3. Restaurant filters by time period or category
4. System updates dashboard dynamically

**Subflows:**

- **Trend Analysis:** Restaurant compares weekly/monthly performance
- **Customer Feedback:** Restaurant reviews aggregated ratings and comments
- **Peak Hours Tracking:** Dashboard highlights busiest times

**Alternate Flows:**

- **A1:** No data for selected range → system displays empty dashboard
- **A2:** Restaurant requests export → system generates CSV/PDF
- **A3:** Unauthorized user → system denies access

**Postconditions:**

- Restaurant gains insights into performance for business decisions

---

# UC-29: Admin User and Role Management

**Actors:** Admin
**Description:** Admin manages internal user accounts and assigns system roles.

**Preconditions:**

- Admin logged in with sufficient permissions

**Main Flow:**

1. Admin navigates to user management dashboard
2. Admin views list of employees with current roles
3. Admin creates, updates, or removes user accounts
4. Admin assigns roles and permissions (support, finance, marketing, etc.)

**Subflows:**

- **Role Templates:** System provides default role configurations
- **Custom Permissions:** Admin fine-tunes access rights
- **Audit Logging:** System records all changes for compliance

**Alternate Flows:**

- **A1:** Admin tries to delete account with active tasks → system blocks
- **A2:** Role conflict detected → system prompts for resolution
- **A3:** Unauthorized attempt to modify roles → system denies and alerts

**Postconditions:**

- Updated internal user list with correct roles and permissions

---

# UC-30: Fraud Detection and Security Monitoring

**Actors:** System, Admin
**Description:** System detects fraudulent activity and ensures security of platform.

**Preconditions:**

- Monitoring tools active
- User activities logged

**Main Flow:**

1. System continuously scans transactions and activities
2. Suspicious patterns flagged (e.g., multiple refunds, stolen cards, fake accounts)

3. Alerts sent to admin for review
4. Admin investigates and takes action (suspend, warn, escalate)

**Subflows:**

- **Machine Learning Detection:** System applies algorithms to identify anomalies
- **Two-Factor Enforcement:** High-risk actions require re-authentication
- **Blacklist Management:** System blocks flagged users or cards

**Alternate Flows:**

- **A1:** False positive flagged → admin clears user account
- **A2:** Fraud detected mid-order → system cancels transaction
- **A3:** Admin ignores alert → system auto-escalates after threshold

**Postconditions:**

- Fraudulent activity prevented or mitigated
- Platform security maintained

---