SE Proj 1b1

Team: Anmol Koul, Arnav Mejari, Om Kumar Singh, Shweta Patki

New use cases

1. Place Food Order

Story: A customer wants to order food through the app. The order is successful only if items are available, payment is successful, and the restaurant accepts the order.
Scenario: Customer places order successfully

Preconditions:

- The customer is logged into the app.
- The customer has selected at least one item.

Main Flow:

- The customer selects food items [Select Items].
- The customer reviews the cart and proceeds to checkout [Checkout].
- The customer enters their delivery address [Enter Address].
- The customer makes payment [Make Payment].
- The system validates payment [Validate Payment].
- The restaurant accepts the order [Accept Order].
- The order is confirmed [Order Confirmed].

Subflows:

- [Select Items] The customer browses the menu and adds items to the cart. Includes handling of quantity adjustments.
- [Checkout] The customer reviews the order summary, including price and delivery information.
- [Enter Address] Customer can use saved addresses or enter a new one. Includes address validation.
- [Make Payment] The customer selects a payment method (card, wallet, cash on delivery).
- [Validate Payment] If valid, proceed. If invalid, display error and allow retry or cancellation.
- [Accept Order] Restaurant confirms or rejects order within a time limit.
- [Order Confirmed] The system generates an order ID and sends notifications to customer and restaurant.
- [Cancel Order] Order is canceled, refunds are processed if applicable, and notifications are sent.

Alternative Flows:

- [Backend fails] System error during order placement. Retry mechanism is presented to user.
- [Out of stock] Items unavailable. User is notified and can modify order or cancel.
- [Invalid Address] Customer enters an invalid address. Address correction prompted.
- [Payment Failure] Payment fails due to insufficient funds or card issues. Retry or alternative payment offered.

2. Process Payment

Story: The system processes a customer's payment for an order. Payment succeeds if details are correct and funds are available.
Scenario: Payment is successful

Preconditions:

- The customer has items in the cart.
- The customer has proceeded to checkout.

Main Flow:

- The customer enters payment details [Enter Payment].

- The system submits details to the payment gateway [Send Request].
- The gateway verifies funds [Verify Funds].
- The payment succeeds [Confirm Payment].

Subflows:
- [Enter Payment] The customer selects a payment method and enters details. Includes input validation.
- [Send Request] Securely transmits payment information to the gateway. Includes handling of timeouts.
- [Verify Funds] The gateway verifies funds; success leads to [Confirm Payment]; failure leads to [Payment Failed].
- [Confirm Payment] Payment is confirmed; the order proceeds. Includes transaction ID generation.
- [Payment Failed] Payment rejected; user notified and can retry or choose another method.

Alternative Flows:
- [Gateway Down] The payment gateway is unavailable. Retry after a delay is suggested.
- [Invalid Payment Details] Incorrect payment information. Error message and opportunity to correct.
- [Fraud Detection] Suspicious activity triggers additional verification steps.

3. Assign Delivery Driver

Story: The system assigns a driver to an accepted order. A driver is assigned if one is available nearby.
Scenario: Driver assigned successfully

Preconditions:
- An order has been placed and accepted by the restaurant.

Main Flow:
- The system identifies nearby available drivers [Find Driver].
- The system sends a delivery request to the nearest driver [Send Request].
- The driver accepts the request [Driver Accepts].
- The driver is assigned to the order [Assign Success].

Subflows:
- [Find Driver] System queries driver location and availability using GPS data. Includes prioritization based on proximity and rating.
- [Send Request] Push notification sent to the driver with order details and estimated compensation.
- [Driver Accepts] Driver accepts or rejects the request. Rejection triggers [Find Driver] again.
- [Assign Success] Order is linked to the driver. Includes updating order status and notifying the customer.
- [Driver Declines] Driver declines the request. Another driver is assigned or order is delayed.

Alternative Flows:
- [No Drivers Available] No drivers are available nearby. Order is delayed or canceled with notification to the customer and restaurant.
- [Driver Unavailable] Selected driver becomes unavailable. System attempts to assign another driver.

4. Update Order Status

Story: The system updates the order's status as it progresses through the delivery process.
Scenario: Order status updated successfully

Preconditions:
- An order exists.

Main Flow:

- An event occurs (order received, preparing, ready, picked up, en route, delivered).
- The system updates the order status accordingly [Update Status].
- The system notifies relevant parties (customer, restaurant, driver) [Notify Parties].

Subflows:

- [Update Status] Database update of order status. Includes timestamps.
- [Notify Parties] Notifications sent via push notifications, SMS, or email. Includes customizable notification settings.
- [Handle Exceptions] Unusual situations (delayed, canceled) are handled with appropriate notifications.

Alternative Flows:

- [Notification Failure] Notification fails to be sent. System logs the error and attempts retry.
- [Status Update Failure] System fails to update status. Error is logged, and retry is attempted.

5. Track Delivery in Real Time

Story: A customer can track the delivery driver's location in real time.
Scenario: Customer successfully tracks delivery.

Preconditions:

- The order is assigned to a driver.
- The driver has accepted the order.

Main Flow:

- The system receives location updates from the driver [Receive Location].
- The system updates the customer's view of the delivery location [Update Map].

Subflows:

- [Receive Location] GPS data is received from the driver's app. Includes error handling for loss of signal.
- [Update Map] Real-time map update for the customer showing driver's location. Includes visual cues (e.g., ETA).

Alternative Flows:

- [GPS Signal Lost] Driver's GPS signal is temporarily lost. Map displays last known location.
- [Driver App Issue] Driver's app malfunctions, preventing location updates. Customer is notified.

6. Manage Restaurant Menu

Story: Restaurant owners can manage their restaurant's menu on the platform.
Scenario: Restaurant owner successfully adds a menu item.

Preconditions:

- The restaurant owner is logged in.

Main Flow:

- The owner accesses the menu management section [Access Menu].
- The owner adds a new item [AddItem].
- The system validates the input [Validate Input].
- The menu is updated [Update Menu].

Subflows:

- [Access Menu] Secure access to menu management tools.

- [AddItem] Input fields for item name, description, price, image, and other relevant information.
- [Validate Input] Checks for correct data types and range of values (e.g., positive price).
- [Update Menu] Database update of menu information.

Alternative Flows:

- [Invalid Input] Incorrect input detected. Error messages and instructions for correction are provided.
- [Duplicate Item] Attempting to add an item that already exists. Appropriate notification given.

7. Handle Complaints/Refunds

Story: Customers can submit complaints, and the system handles resolutions and refunds.
Scenario: Refund issued successfully.

Preconditions:

- An order has been completed.

Main Flow:

- The customer submits a complaint [Submit Complaint].
- The support team reviews the complaint [Review Complaint].
- The support team investigates and contacts relevant parties [Investigate].
- The support team decides on a resolution (refund or other action) [Decide Resolution].
- The resolution is implemented [Implement Resolution].

Subflows:

- [Submit Complaint] Customer provides details of complaint via form or other channels.
- [Review Complaint] Support team assesses the validity of the complaint.
- [Investigate] Support contacts restaurant and/or driver to gather information.
- [Decide Resolution] Based on investigation, a resolution is determined.
- [Implement Resolution] Refund is processed, or other action is taken (e.g., replacement order).

Alternative Flows:

- [Complaint Rejected] Complaint is deemed invalid; customer is notified.
- [Partial Refund] Partial refund is issued instead of a full refund.

8. Rate and Review Order

Story: Customers can rate and review their order experience.
Scenario: Customer submits a review successfully.

Preconditions:

- The order is delivered.

Main Flow:

- The customer is prompted to rate and review the order [Prompt Review].
- The customer submits the rating and review [Submit Review].
- The review is saved and becomes visible (after moderation, if needed) [Save Review].

Subflows:

- [Prompt Review] Post-delivery prompt to rate the order (star rating and optional text review).
- [Submit Review] Input validation of review text (e.g., length limitations, profanity filter).
- [Save Review] Stores the review in the database and associates it with the order. May include a moderation process.

Alternative Flows:

- [Review Rejected] Review is deemed inappropriate and rejected by moderation.
- [Review Editing] Allows user to edit their review before submission.

9. Monitor Business Performance (Restaurant)

Story: Restaurant owners can monitor their business performance through various metrics.
Scenario: Restaurant owner views sales data.

Preconditions:

- The restaurant owner is logged into the dashboard.

Main Flow:

- The owner accesses the analytics dashboard [Access Dashboard].
- The owner selects a reporting period and metrics [Select Metrics].
- The system fetches and displays relevant data [Fetch Data].

Subflows:

- [Access Dashboard] Secure access to the restaurant's performance analytics.
- [Select Metrics] Options for various metrics such as revenue, order volume, average order value, etc., and selection of timeframes (daily, weekly, monthly).
- [Fetch Data] Retrieves data from database and generates report (e.g., charts, tables).

Alternative Flows:

- [Data Fetch Error] Data retrieval fails. Error message displayed, with retry option.
- [Insufficient Data] Data unavailable for the selected period. Appropriate message displayed.

10. Manage Driver Payments

Story: The system manages driver payments.
Scenario: Driver receives payment successfully.

Preconditions:

- The driver has completed deliveries within a payment cycle.

Main Flow:

- The system calculates the driver's earnings [Calculate Earnings].
- The system deducts platform fees [Deduct Fees].
- The system initiates the payout [Initiate Payout].
- The driver receives the payment [Receive Payment].

Subflows:

- [Calculate Earnings] Based on completed deliveries and associated fees.
- [Deduct Fees] Platform commission and other fees are deducted.
- [Initiate Payout] Payment sent via chosen method (e.g., direct deposit, mobile payment).
- [Receive Payment] Driver is notified that payment is processed.

Alternative Flows:

- [Payment Failure] Payment fails due to banking issues. Driver is notified and further steps are outlined.

- [Insufficient Earnings] Driver earnings are below the minimum payout threshold. Payment is held until sufficient earnings are accumulated.

11. Manage Customer Accounts

Story: Customers can manage their account information.
Scenario: Customer successfully updates their address.

Preconditions:

- The customer is logged in.

Main Flow:

- The customer accesses their account settings [Access Settings].
- The customer updates their address [Update Address].
- The system validates and saves the address [Save Address].

Subflows:

- [Access Settings] Secure access to account management.
- [Update Address] Input fields for address information.
- [Save Address] Validation and update of customer address in the database.

Alternative Flows:

- [Invalid Address] Incorrect address format. Error message and correction prompted.
- [Address Already Exists] Attempting to add an address that is already saved.

12. Manage Restaurant Profiles

Story: Restaurants can manage their profile information.
Scenario: Restaurant successfully updates its operating hours.

Preconditions:

- The restaurant manager is logged in.

Main Flow:

- The manager accesses the restaurant profile [Access Profile].
- The manager updates operating hours [Update Hours].
- The system saves the updated hours [Save Hours].

Subflows:

- [Access Profile] Secure access to restaurant profile management.
- [Update Hours] Input fields for specifying daily operating hours.
- [Save Hours] Update of operating hours in the database.

Alternative Flows:

- [Invalid Hours Format] Error message if the format of the operating hours is incorrect.
- [Overlap with Existing Orders] System alerts manager if changes conflict with existing orders.

13. Customer Support Chat

Story: Customers can engage in live chat with customer support.
Scenario: Customer successfully gets their question answered.

Preconditions:

- Customer is logged in or provides necessary information.

Main Flow:

- Customer initiates a chat session [Start Chat].
- Customer support agent accepts the session [Accept Chat].
- Agent and customer exchange messages [Exchange Messages].
- Session concludes [End Chat].

Subflows:

- [Start Chat] Customer selects chat option and provides brief description of issue.
- [Accept Chat] Agent accepts the chat and greets the customer.
- [Exchange Messages] Real-time message exchange between customer and agent.
- [End Chat] Session concludes, possibly with a resolution summary.

Alternative Flows:

- [No Agent Available] Customer placed in a queue.
- [Chat Timeout] Chat session times out due to inactivity.

14. Restaurant Search and Filtering

Story: Customers can find restaurants based on various criteria.
Scenario: Customer successfully finds a restaurant.

Preconditions:

- Customer is on the restaurant search screen.

Main Flow:

- Customer enters search criteria (cuisine, location, etc.) [Enter Criteria].
- System retrieves matching restaurants [Search Restaurants].
- System displays results [Display Results].

Subflows:

- [Enter Criteria] User inputs search terms through text input, selection from menus, or map interaction.
- [Search Restaurants] System queries the database based on the provided criteria.
- [Display Results] Displays results on the screen, possibly with pagination and sorting options.

Alternative Flows:

- [No Results] No restaurants match the search criteria. Message to the user.
- [Search Error] System error during search.

15. Push Notifications

Story: The system sends push notifications to keep users updated.
Scenario: Customer receives an order update notification.

Preconditions:

- Customer has enabled push notifications.
- An event relevant to the customer occurs (e.g., order status change).

Main Flow:

- An event triggers a notification [Trigger Notification].

- System prepares notification message [Prepare Message].
- System sends notification [Send Notification].

Subflows:
- [Trigger Notification] System detects an event needing notification.
- [Prepare Message] Constructs the notification message with relevant details.
- [Send Notification] Sends notification via the push notification service.

Alternative Flows:
- [Notification Failure] Notification fails to be sent.
- [Push Notification Disabled] User has push notifications disabled.

16. Driver App Login

Story: Delivery drivers can log in to their app.
Scenario: Driver logs in successfully.

Preconditions:
- Driver has a registered account.

Main Flow:
- Driver enters credentials [Enter Credentials].
- System authenticates credentials [Authenticate].
- Driver is logged in [Login Success].

Subflows:
- [Enter Credentials] Driver inputs username and password.
- [Authenticate] System verifies credentials against the database.
- [Login Success] Driver is granted access to the driver app.

Alternative Flows:
- [Invalid Credentials] Incorrect username or password. Retry is allowed.
- [Account Locked] Account is temporarily locked due to multiple failed login attempts.

17. Restaurant Onboarding

Story: Restaurants can register and onboard their business.
Scenario: Restaurant completes onboarding successfully.

Preconditions:
- Restaurant has all necessary information.

Main Flow:
- Restaurant fills out the registration form [Fill Form].
- Restaurant uploads necessary documents [Upload Docs].
- System validates information and documents [Validate].
- Restaurant is approved and activated [Activate].

Subflows:
- [Fill Form] Provides business information, contact details, menu details, etc.
- [Upload Docs] Uploads necessary permits, licenses, and other documentation.
- [Validate] System checks for completeness and validity of provided information and documents.

- [Activate] Restaurant account is created and activated.

Alternative Flows:

- [Incomplete Information] Missing information. Prompts the user to provide missing data.
- [Document Rejection] Uploaded documents are deemed insufficient.

18. Driver Onboarding

Story: Drivers can register and onboard their account.
Scenario: Driver completes onboarding successfully.

Preconditions:

- Driver has all necessary information and documentation.

Main Flow:

- Driver fills out the registration form [Fill Form].
- Driver uploads necessary documents (license, insurance, etc.) [Upload Docs].
- System validates information and documents [Validate].
- Driver is approved and activated [Activate].

Subflows:

- [Fill Form] Driver provides personal details, vehicle information, banking information, etc.
- [Upload Docs] Uploads necessary documentation (driver's license, vehicle registration, insurance).
- [Validate] System verifies documents and information.
- [Activate] Driver account is created and activated.

Alternative Flows:

- [Incomplete Information] Missing required information. Prompts driver to provide missing details.
- [Document Rejection] Uploaded documents are deemed invalid or insufficient.

19. Customer Profile Management (Photo Upload)

Story: Customers can upload a profile picture.
Scenario: Customer successfully uploads a profile picture.

Preconditions:

- Customer is logged in.

Main Flow:

- Customer accesses profile settings [Access Profile].
- Customer selects a photo from their device [Choose Photo].
- System uploads and processes photo [Upload Photo].
- Photo is displayed in the profile [Display Photo].

Subflows:

- [Access Profile] Secure access to user profile settings.
- [Choose Photo] User selects a photo from their device's gallery or camera.
- [Upload Photo] System uploads the photo and may perform image processing (resizing, compression).
- [Display Photo] Updated profile picture is displayed.

Alternative Flows:

- [Invalid File Type] Incorrect file type selected. Error message and guidance.

- [Upload Failure] Failure to upload photo due to network or system error.

20. Promotional Offers Management (Restaurant)

Story: Restaurants can manage their promotional offers.
Scenario: Restaurant adds a new promotional offer.

Preconditions:
- Restaurant is logged in.

Main Flow:
- Restaurant accesses promotion settings [Access Promotions].
- Restaurant creates a new offer [Create Offer].
- System validates offer details [Validate Offer].
- Offer is activated and becomes visible [Activate Offer].

Subflows:
- [Access Promotions] Secure access to promotion management tools.
- [Create Offer] Input fields to define the offer (discount type, percentage, duration, etc.).
- [Validate Offer] Checks for valid offer parameters and conflicts with existing offers.
- [Activate Offer] Offer is added to the database and becomes visible to customers.

Alternative Flows:
- [Invalid Offer Details] Incorrect parameters. Error message and correction prompted.
- [Offer Overlap] Offer conflicts with existing offers. Alert to restaurant.

21. Browse Restaurant Menu

Story: A customer wants to browse available restaurants and their menus. The customer should be able to filter and sort restaurants based on various criteria.
Scenario: Customer successfully browses the menu of a restaurant.

Preconditions:
- The customer is logged into the app.
- The app has an internet connection.

Main Flow:
- The customer views the list of restaurants [View Restaurants].
- The customer selects a restaurant [Select Restaurant].
- The customer views the restaurant's menu [View Menu].
- The customer browses menu categories [Browse Categories].
- The customer views menu items details [View Item Details].
- The customer adds items to the cart [Add to Cart].

Subflows:
- [View Restaurants]: The system displays restaurants based on location and other filters.
- [Select Restaurant]: The customer taps on a restaurant from the list.
- [View Menu]: The system displays the menu items, categorized and with images.
- [Browse Categories]: The customer navigates through different food categories (e.g., appetizers, main courses, desserts).
- [View Item Details]: The system shows detailed information for selected items (description, price, ingredients, dietary information).

- [Add to Cart]: The system adds the selected item to the customer's cart. If the item is out of stock, it notifies the user [Out of Stock].

Alternative Flows:

- [No Internet Connection]: The app displays a message indicating a lack of internet connectivity.
- [Restaurant Unavailable]: The selected restaurant is temporarily unavailable or closed. The app displays a message indicating this.
- [App Crash]: The app unexpectedly crashes.
- [Out of Stock]: Selected item is out of stock; the system prompts to select another item or removes it from the cart.

22. Manage Customer Addresses

Story: A customer wants to manage their delivery addresses. The customer can add, edit, delete, and select default addresses.
Scenario: Customer successfully adds a new delivery address.

Preconditions:

- The customer is logged into the app.

Main Flow:

- The customer navigates to the address management screen [Access Addresses].
- The customer selects "Add Address" [Add Address].
- The customer enters address details [Enter Details].
- The system validates the address [Validate Address].
- The system saves the address [Save Address].
- The customer selects the new address as default (optional) [Set Default].

Subflows:

- [Access Addresses]: The customer accesses the address book section of the app.
- [Add Address]: The customer initiates the process to add a new address.
- [Enter Details]: The customer provides street address, city, state, zip code, etc.
- [Validate Address]: The system checks for correct formatting and completeness of address information.
- [Save Address]: The system stores the address details in the customer's profile.
- [Set Default]: The customer marks the address as their preferred delivery location.

Alternative Flows:

- [Invalid Address]: The system detects invalid address format; displays an error and prompts for correction.
- [Address Already Exists]: The system detects an existing address with the same details; notifies the user.

23. Apply Coupon Code

Story: A customer wants to apply a coupon code to their order to receive a discount.
Scenario: Customer successfully applies a valid coupon code.

Preconditions:

- The customer has items in their cart.

Main Flow:

- The customer proceeds to checkout [Proceed to Checkout].
- The customer enters the coupon code [Enter Coupon].
- The system validates the coupon code [Validate Coupon].
- The system applies the discount [Apply Discount].
- The updated order total is displayed [Display Total].

Subflows:

- [Proceed to Checkout]: The customer moves to the payment and order confirmation screen.
- [Enter Coupon]: The customer inputs the alphanumeric code.
- [Validate Coupon]: The system verifies the code against its database (validity, expiry date, usage limits).
- [Apply Discount]: The discount is calculated and applied to the order total.
- [Display Total]: The app updates the final price reflecting the discount.

Alternative Flows:

- [Invalid Coupon Code]: The system detects an invalid code; displays an error and prompts for a correct one.
- [Coupon Expired]: The system detects an expired code; displays an error and advises the customer.
- [Coupon Already Used]: The system detects that the coupon has already been used; displays an error.

24. Contact Customer Support

Story: A customer needs to contact customer support for assistance with an order or other issue.
Scenario: Customer successfully initiates a chat with support.

Preconditions:

- The customer is logged into the app.

Main Flow:

- The customer navigates to the customer support section [Access Support].
- The customer selects "Start Chat" [Start Chat].
- The system connects the customer to a support representative [Connect to Agent].
- The customer communicates with the support representative [Chat with Agent].
- The chat session is concluded [End Chat].

Subflows:

- [Access Support]: The customer locates the support contact option within the app.
- [Start Chat]: The customer initiates a live chat session.
- [Connect to Agent]: The system attempts to connect the customer to an available agent.
- [Chat with Agent]: The customer and agent exchange messages to resolve the issue.
- [End Chat]: The chat session is closed by either party.

Alternative Flows:

- [No Agents Available]: The system informs the customer that no agents are currently available and provides alternative contact options (email, phone).
- [Chat Disconnection]: The chat session is unexpectedly disconnected. The system prompts the customer to reconnect or contact support through a different method.

25. View Order History

Story: A customer wants to view their past orders.
Scenario: Customer successfully views their order history.

Preconditions:

- The customer is logged into the app.

Main Flow:

- The customer navigates to the order history section [Access History].
- The system retrieves the customer's order history [Retrieve Orders].

- The system displays a list of past orders [Display Orders].
- The customer selects an order to view details [View Order Details].

Subflows:

- [Access History]: The customer finds the order history section in their account.
- [Retrieve Orders]: The system fetches order data from the database, filtering by customer ID.
- [Display Orders]: The system shows a list of orders with key information (restaurant, date, total).
- [View Order Details]: Selecting an order displays detailed information (items, address, payment, status).

Alternative Flows:

- [No Orders Found]: The system indicates that no past orders are found for the customer.
- [Error Retrieving Orders]: The system encounters an error during data retrieval; displays an error message.

26. Rate Restaurant

Story: A customer wants to rate and review a restaurant after an order is completed.
Scenario: Customer successfully submits a restaurant review.

Preconditions:

- The customer has completed an order from the restaurant.

Main Flow:

- The customer views the restaurant's details page [Access Restaurant Page].
- The customer selects the rating option [Access Rating].
- The customer provides a star rating [Provide Rating].
- The customer optionally adds a text review [Add Review].
- The system saves the rating and review [Save Review].

Subflows:

- [Access Restaurant Page]: The customer accesses the restaurant's profile page.
- [Access Rating]: The customer finds the rating section on the restaurant page.
- [Provide Rating]: The customer selects a star rating (e.g., 1-5 stars).
- [Add Review]: The customer optionally inputs a text-based review.
- [Save Review]: The system stores the rating and review associated with the customer and the restaurant.

Alternative Flows:

- [Review Already Submitted]: The system detects that a review has already been submitted for this order and restaurant.
- [Character Limit Exceeded]: The review text exceeds a defined character limit.

27. Restaurant Accepts Order

Story: A restaurant accepts an incoming order.
Scenario: Restaurant successfully accepts an order.

Preconditions:

- An order has been placed and is pending.

Main Flow:

- The restaurant receives an order notification [Receive Order].
- The restaurant views order details [View Order].
- The restaurant accepts the order [Accept Order].

- The system updates the order status [Update Status].
- The customer receives an order confirmation [Notify Customer].

Subflows:

- [Receive Order]: The restaurant receives a notification (push notification, app alert).
- [View Order]: The restaurant accesses the order details (items, address, customer information).
- [Accept Order]: The restaurant confirms the order acceptance.
- [Update Status]: The system changes the order status to "Accepted".
- [Notify Customer]: The customer receives an app notification that the order is accepted.

Alternative Flows:

- [Order Rejected]: The restaurant rejects the order due to unavailability of items or other reasons; notifies the customer and system.
- [Order Timeout]: The order is automatically canceled after a specified time if the restaurant does not respond.

28. Driver Picks Up Order

Story: A delivery driver picks up an order from a restaurant.
Scenario: Driver successfully picks up an order.

Preconditions:

- An order is assigned to a driver and the order status is "Accepted".

Main Flow:

- The driver arrives at the restaurant [Arrive at Restaurant].
- The driver confirms order pickup [Confirm Pickup].
- The system updates the order status [Update Status].

Subflows:

- [Arrive at Restaurant]: The driver uses the app's navigation to reach the restaurant.
- [Confirm Pickup]: The driver confirms in the app that they have picked up the order.
- [Update Status]: The system changes the order status to "Picked Up".

Alternative Flows:

- [Order Not Ready]: The restaurant informs the driver that the order is not yet ready. The system updates the status accordingly and notifies the customer if necessary.
- [Driver Cannot Locate Restaurant]: The driver is unable to locate the restaurant; contacts support or cancels the order.

29. Driver Completes Delivery

Story: A delivery driver completes a food delivery to the customer.
Scenario: Driver successfully completes delivery.

Preconditions:

- The order status is "Picked Up".

Main Flow:

- The driver arrives at the delivery address [Arrive at Address].
- The driver confirms delivery [Confirm Delivery].
- The system updates the order status to "Delivered" [Update Status].

- The customer is notified of the delivery [Notify Customer].

Subflows:
- [Arrive at Address]: The driver uses the app's navigation to reach the delivery location.
- [Confirm Delivery]: The driver confirms in the app that the order has been delivered.
- [Update Status]: The system changes the order status to "Delivered".
- [Notify Customer]: The customer receives a notification that the order has arrived.

Alternative Flows:
- [Customer Not Available]: The customer is not available at the delivery address. The driver contacts support or attempts redelivery (if applicable).
- [Incorrect Address]: The delivery address is incorrect. The driver contacts support for clarification.

30. View Driver Earnings

Story: A driver wants to view their earnings.
Scenario: Driver successfully views their earnings.

Preconditions:
- The driver is logged into the driver app.

Main Flow:
- The driver navigates to the earnings section [Access Earnings].
- The system retrieves the driver's earnings data [Retrieve Earnings].
- The system displays the driver's earnings [Display Earnings].

Subflows:
- [Access Earnings]: The driver accesses the earnings section of the driver app.
- [Retrieve Earnings]: The system retrieves the earnings information from the database, filtering by driver ID.
- [Display Earnings]: The system displays the total earnings, including details for each delivery completed.

Alternative Flows:
- [No Earnings Found]: The system indicates that no earnings are currently available for the driver.
- [Error Retrieving Earnings]: The system encounters an error during data retrieval; displays an error message.

Differences in LLM’s critique of our use cases

We used ChatGPT and Perplexity to critique our first 10 use cases, using zero-shot and careful prompting.
ChatGPT came up with 7 issues or ‘missing’ parts of our use cases, which we thought were valid critiques, but not quite complete. When questioned further, it acknowledged our suggestions on missing parts and suggested ways to incorporate them into the use cases. Some of these additions were about regulations and some were about reordering steps to avoid dark patterns, and some included slight wording changes or additions to make it clear that the delivery platform is working for the overall benefit of the users. It did also make some suggestions that were illogical, such as allowing the users to filter out sponsored posts, which would not work as a business model.

Perplexity mentions that our current use-case set is missing several critical regulatory and user-protection flows. After carefully investigating the use cases and being cross questioned, it validated the concerns we had regarding the regulations, especially when talking about the dark pattern violations. However, it did not add on much to the “critique” compared to the ChatGPT and did not expand much in the use-case scenarios. It also provided few use cases which were not directly or indirectly affecting the requirements, such as keeping control on child-targeted marketing, restaurant profiles, or real-time temperature logs.

Cost to run

Gemini Flash- Student account free access. (Test charge- INR2)
Perplexity pro- Student account free access.
ChatGPT 5- Free to use.
Total cost = $0