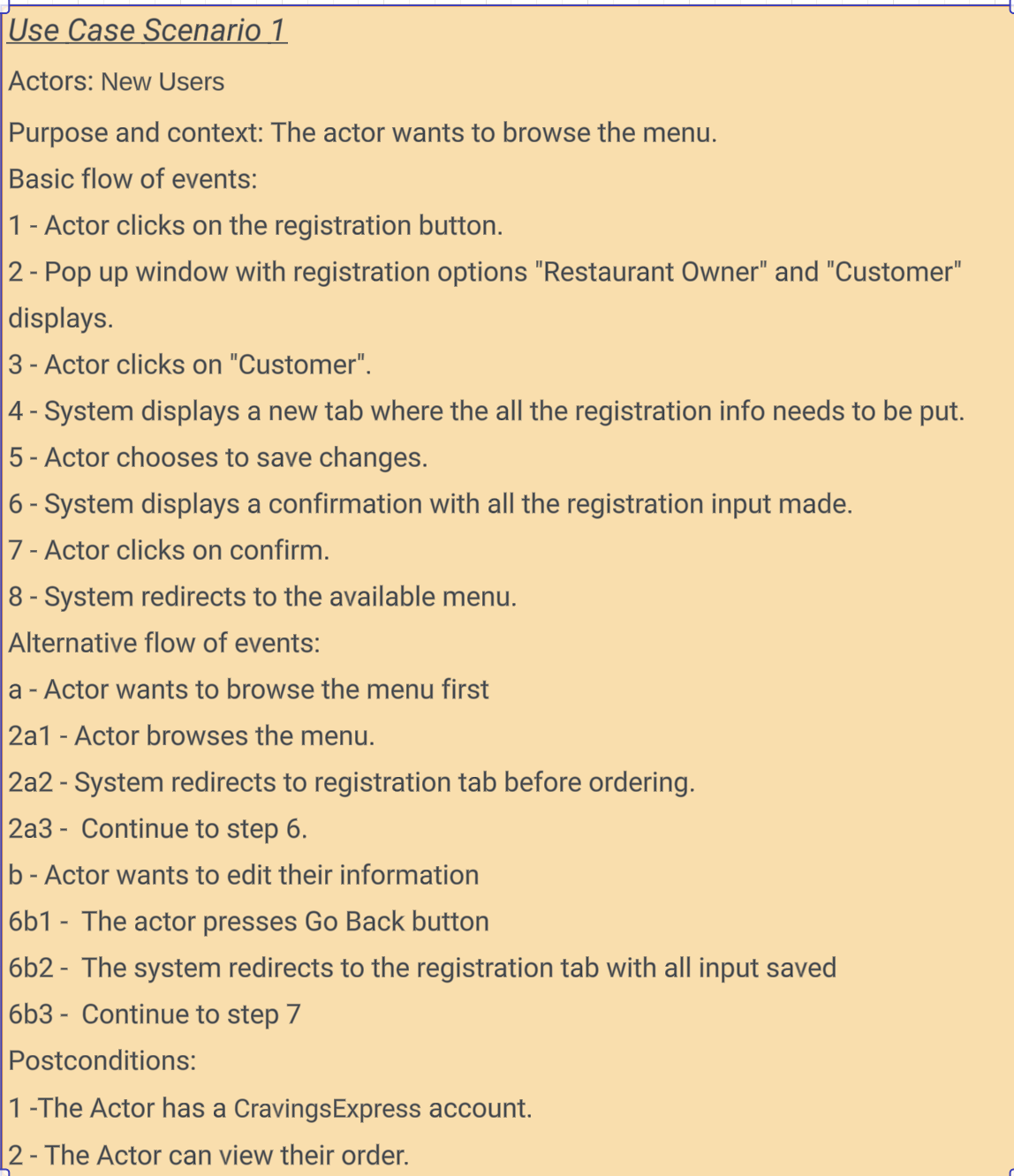
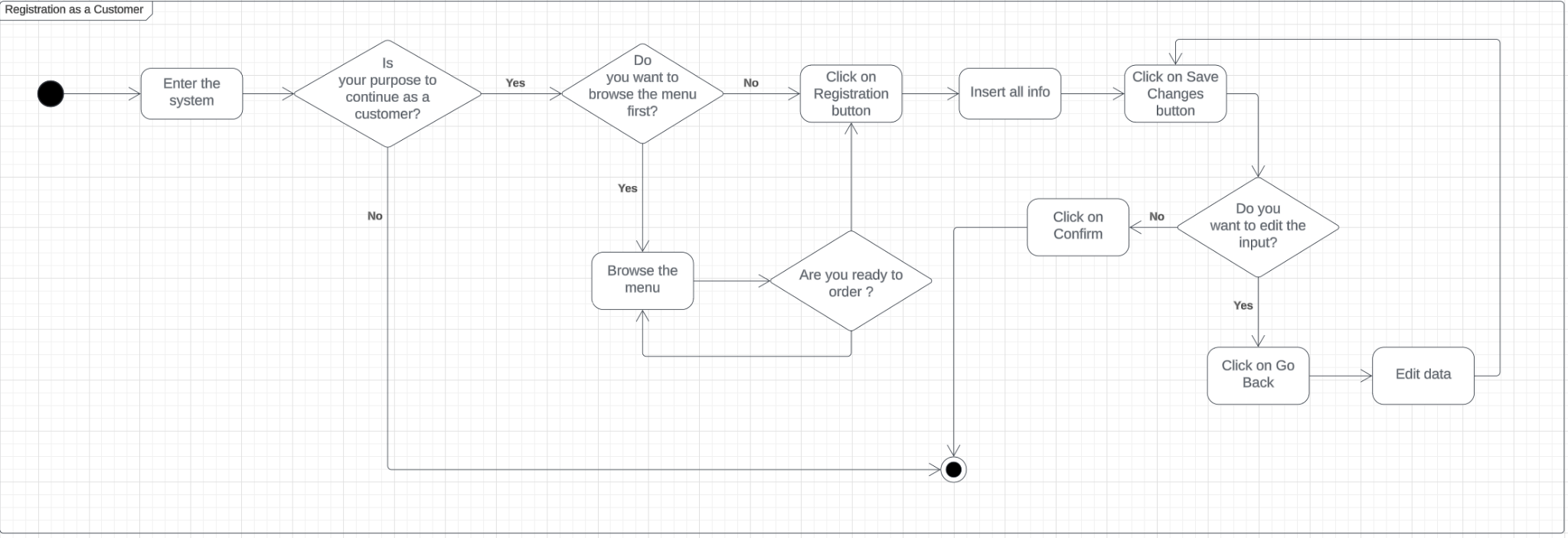
# **User Requirements** CarvingsExpress was created to provide the best experience for users while ordering any meal. And provide additional orders and reviews to restaurants. Everything starts with the registration process. There are two options for users: register as a consumer or as a restaurant. Also, to get rid of bots and dos, there is email confirmation.As a consumer, name, delivery address and phone number should be provided.As a restaurant, restaurant’s name, address is necessary. And after basic verification by side services, menu items, and prices uploading function appears.Or as a delivery driver through exciting account in our partners app. Consumers can filter restaurants based on dietary preferences like vegetarian, gluten-free, or vegan. And then browse through restaurants sorted in a way that was chosen: new, top rate to low rate, low rate to top rate. After choosing the exact one, the user redirects to restaurant info: menu, including food descriptions, prices, availability and reviews. All items that were chosen are collected in a cart, where consumers can apply their discount codes and choose the time slot available. From the previous tab user proceeds to notes for order such as: additional ingredients. At the end, consumers can choose suitable payment methods: credit cards, debit cards, integrated payment systems such as PayPal, Blik and “cash upon delivery” is also available. Confirmation of order is being sent after the payment went through. After receiving an order, a top-up window appears where customers can leave basic stars rate or more detailed for restaurant and driver. Each order is recorded to the database and can be easily reordered.

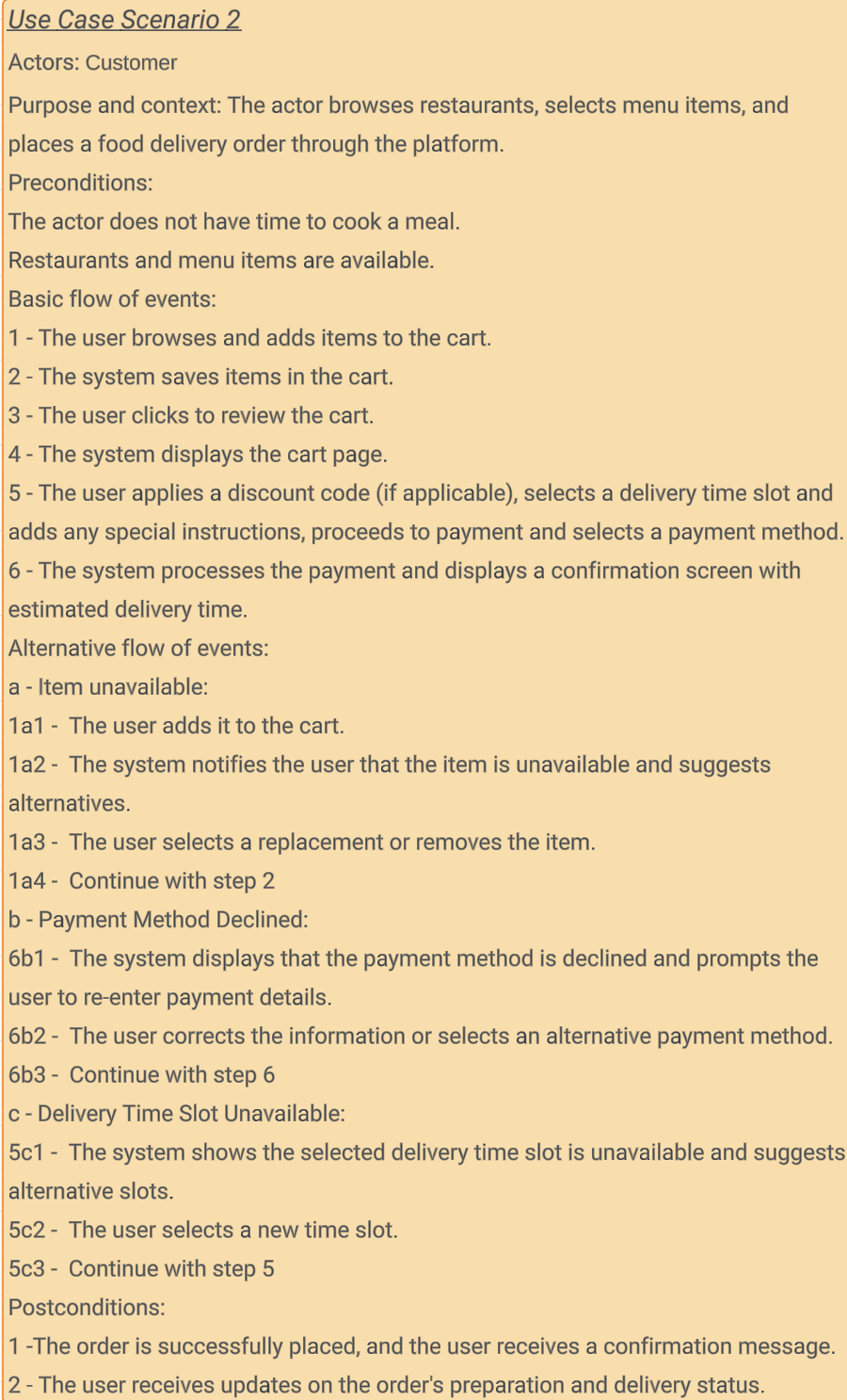
# Restaurant has the functionality to refresh products availability, actual menu and working schedule, also there is a simple interface for designing lending that will be seen to customers. Main functionality of the restaurant interface is an up to date fast order receiving process and an understandable menu for updating order status, which is visible for customers. All orders and detailed data are stored in a special database that can be used for business analysis. As for drivers we use our partner’s companies. And the login process executes through an exciting courier service’s account. Interface for delivery shows the status of order and the location of the restaurant, where it should be picked up. App connects the nearest available driver with the restaurant, based on the provided restaurant address and time left for completing the order.

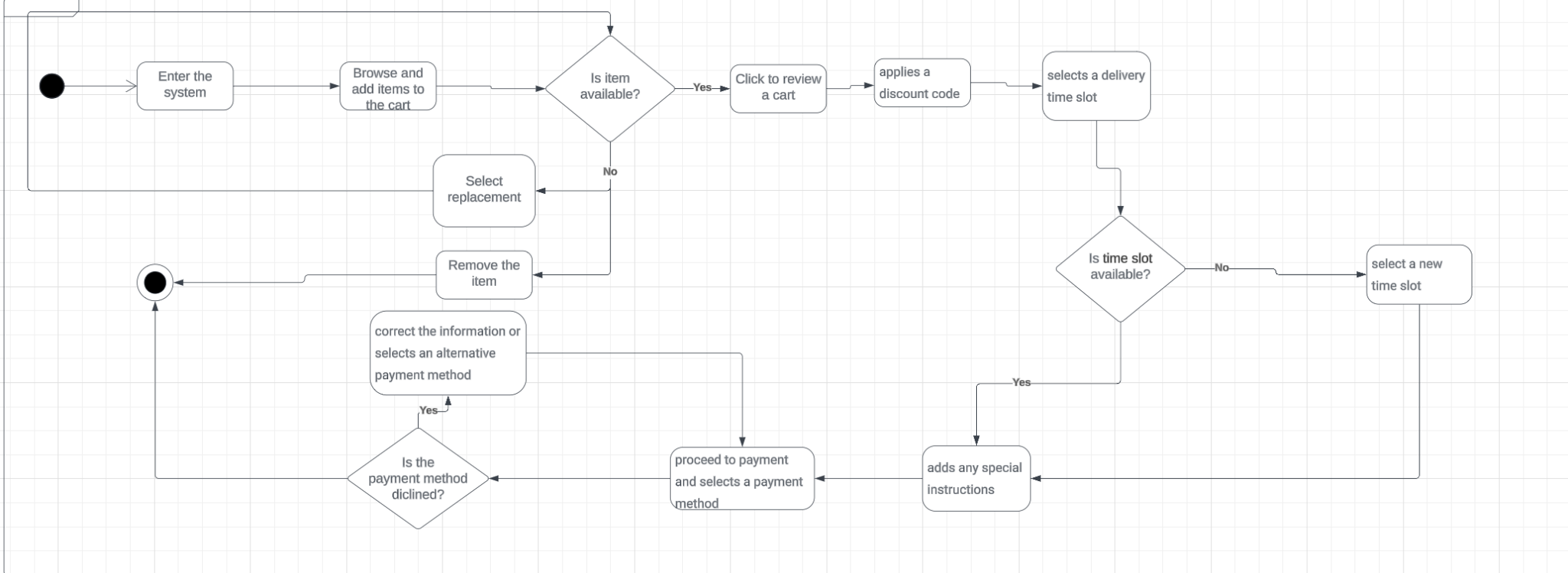
# **Use case diagram**

# **Use case scenarios and activity diagrams**

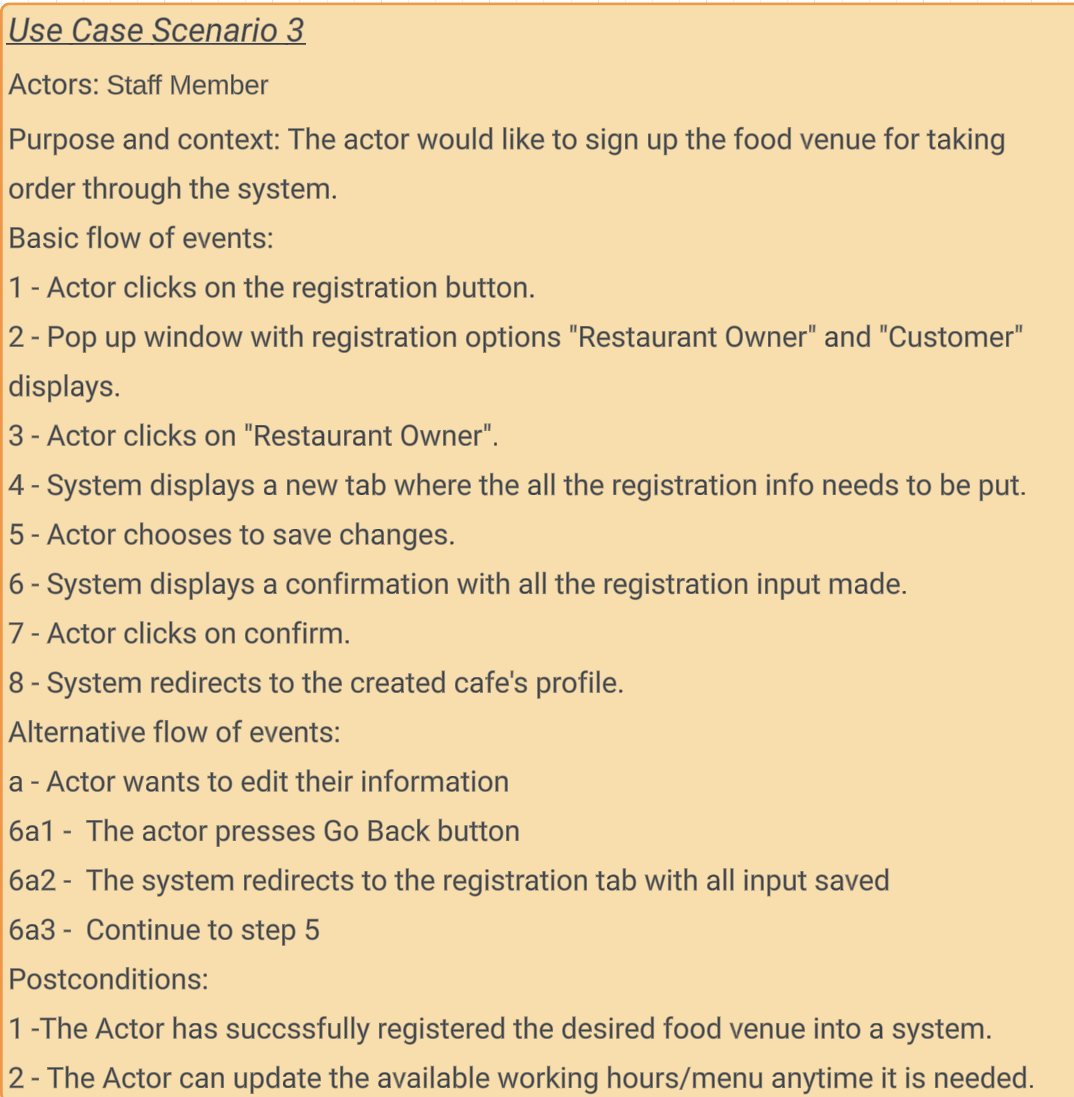
The actor wants to browse the menu  


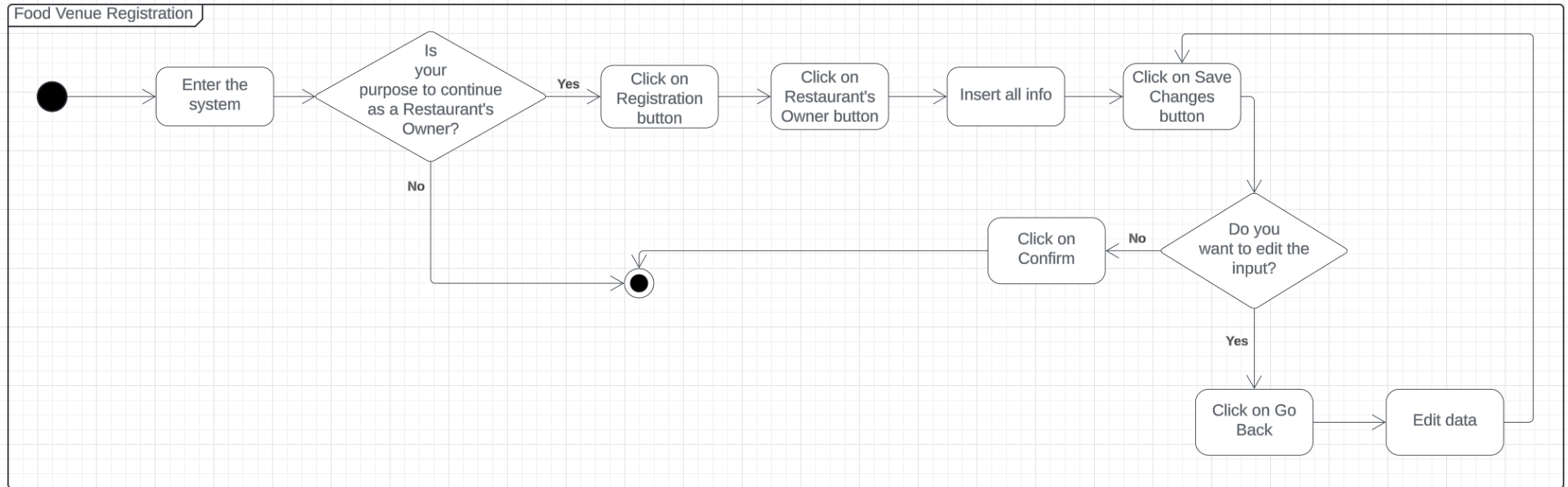


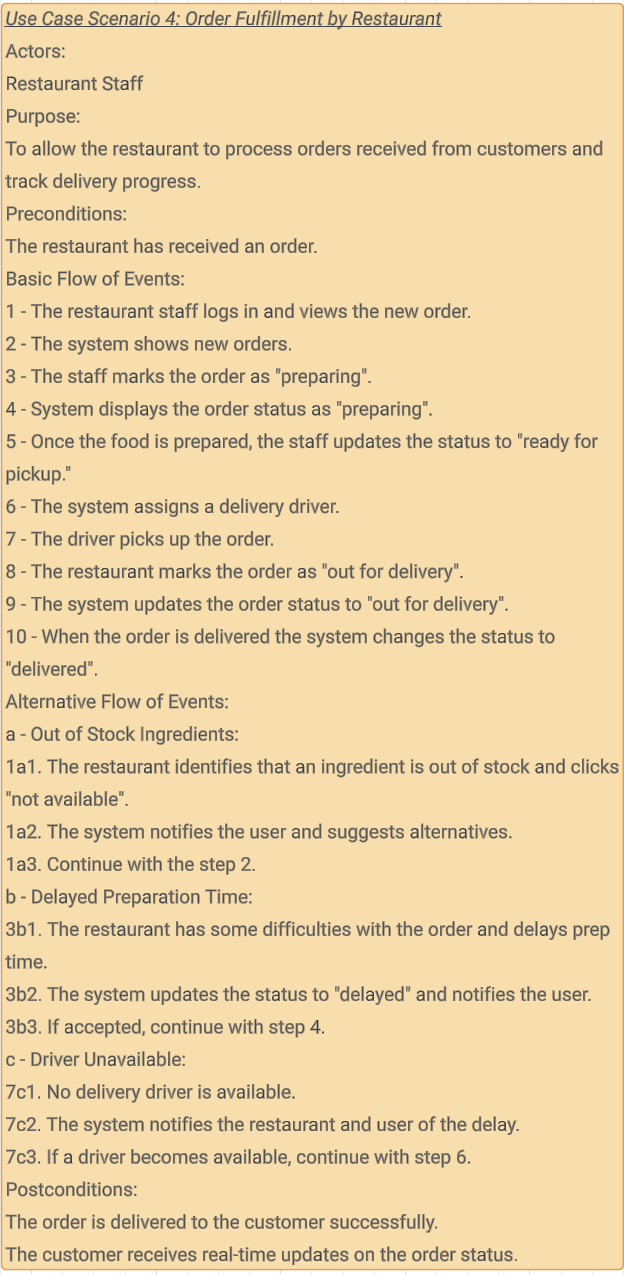
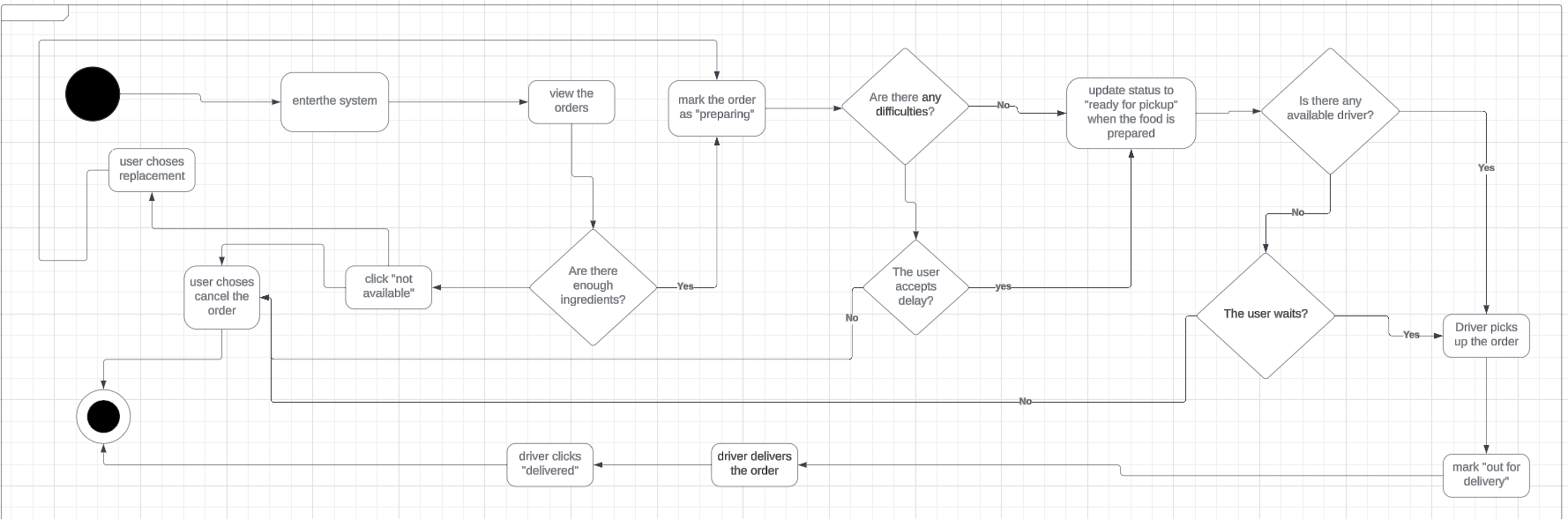
The actor browses restaurants, selects menu items, and places a food delivery order through the platform

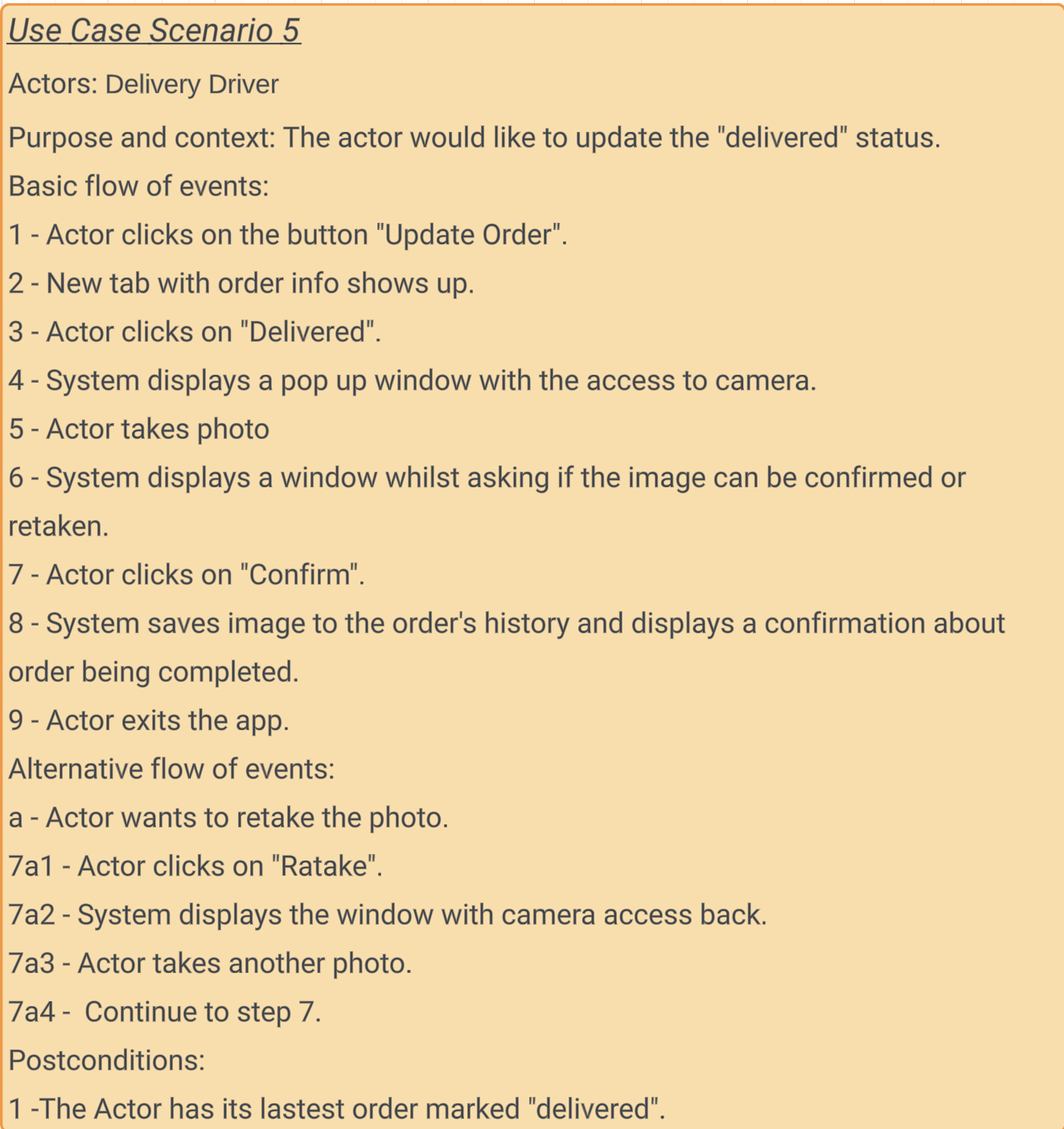


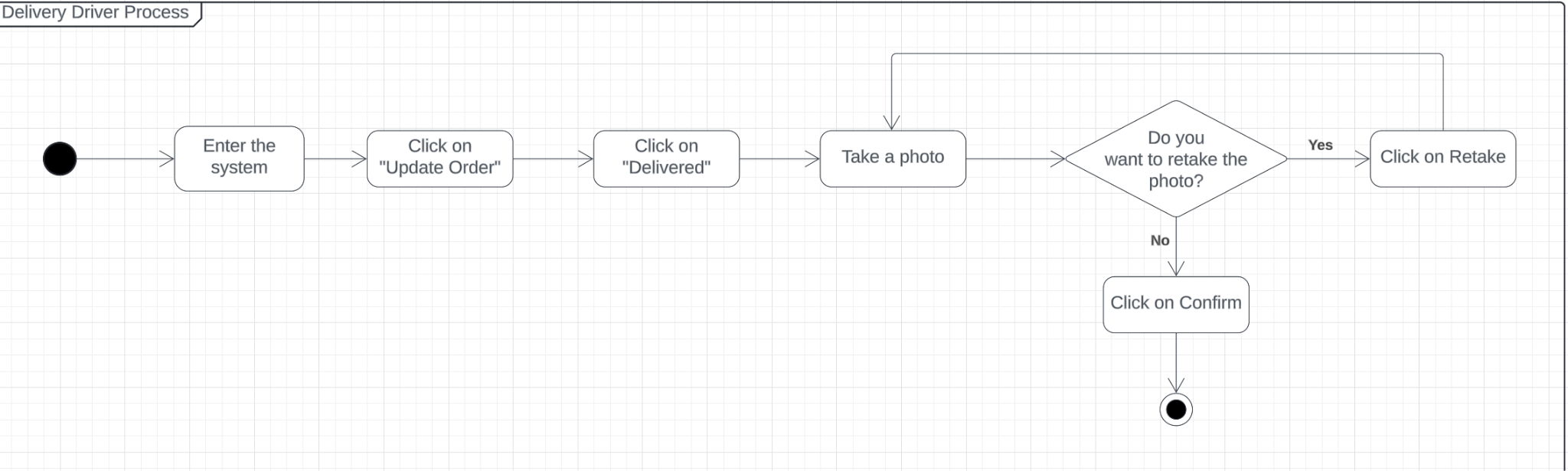
Food Venue Registration



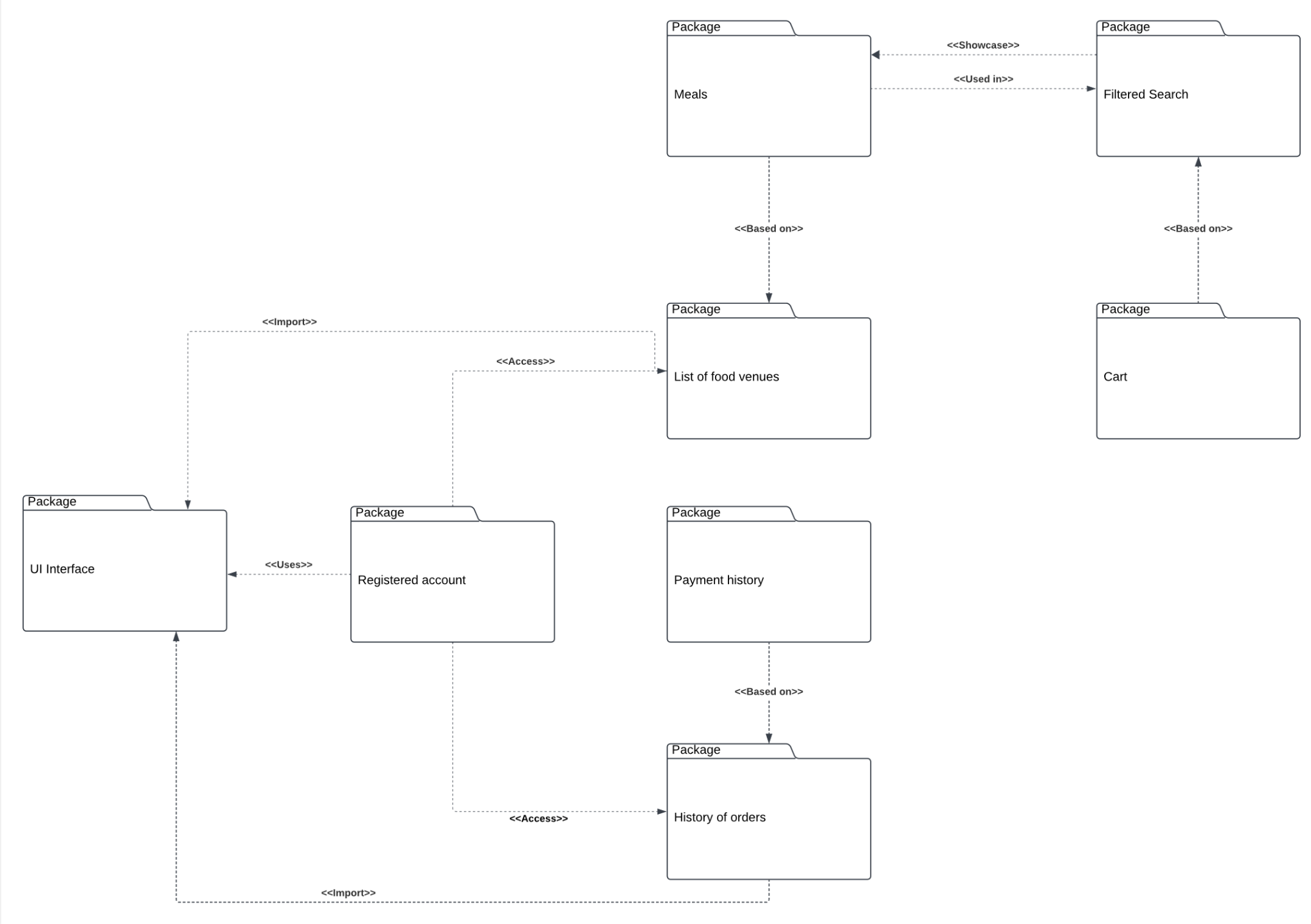


Restaurant to process orders received from customers and track delivery progress  


Delivery Driver Process  


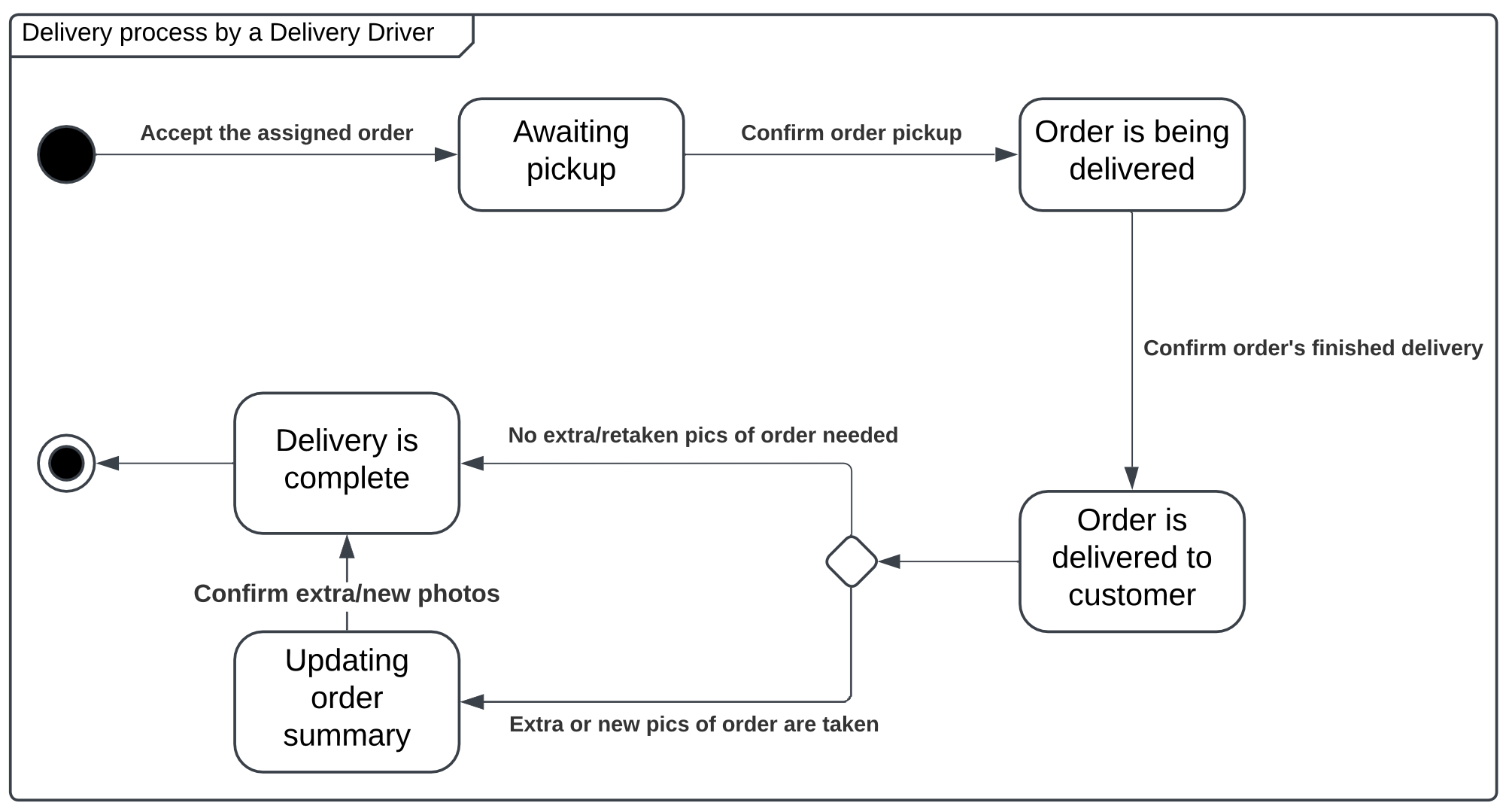


# **Package diagram**



Our package diagram consists of modules of which our system consists.   
  
UI interface collects all import data that were provided by users, and encrypts that to our database. And allows users to use in-app modules  
  
List of food represents all successful registrations for restaurants and up-to-date data about them.   
  
Cart package represents the end point of consumer’s choice that is based on packages as meals and filtering, which apply changes to the user's food venues list and cut meal list.   
  
History of orders collects all orders that were executed and based on payment status. Orders stores as “successful” and “unsuccessful” . That gives an ability to consumers to repeat order and for restaurants to make business analysis.

# **State Diagram**



Delivery process consists of such statuses:  
  
1.Awaiting to pick up - driver accepted order from restaurant and on the way to it  
2.Order is being delivered - courier picked up the order and confirmed that in app  
3.Order is delivered to customer - courier came to delivery address

# **Class diagram**