Online Pizza Ordering System

[Minimum Viable Prototype](#_ziauw5idii1a) **2**

[User Interface Sketch](#_669oze5brcy) 2

[Architectural Diagram](#_x30atcrzq70p) 3

[Class & Implementation Summary](#_vltgt06mb8tx) **3**

[Objects](#_r8ccsrgqn0n) 4

[ORM Database for Customer, FillStore & MenuInventory](#_ewr8jcszktyt) 4

[Design Patterns](#_kix4kyxe31iz) 5

[Factory: Inventory](#_b0idrcnoail4) 5

[Iteration: Cart Items](#_wiuoqkpv69g9) 6

[Composition: Order Summary](#_3r9dcnjwj7io) 7

[Mediator: Order System](#_gosq6h46stqo) 8

[Requirements (Use Cases)](#_6a4iqjbqabyp) **9**

[Class Diagrams](#_wuhmy6lpozg1) **10**

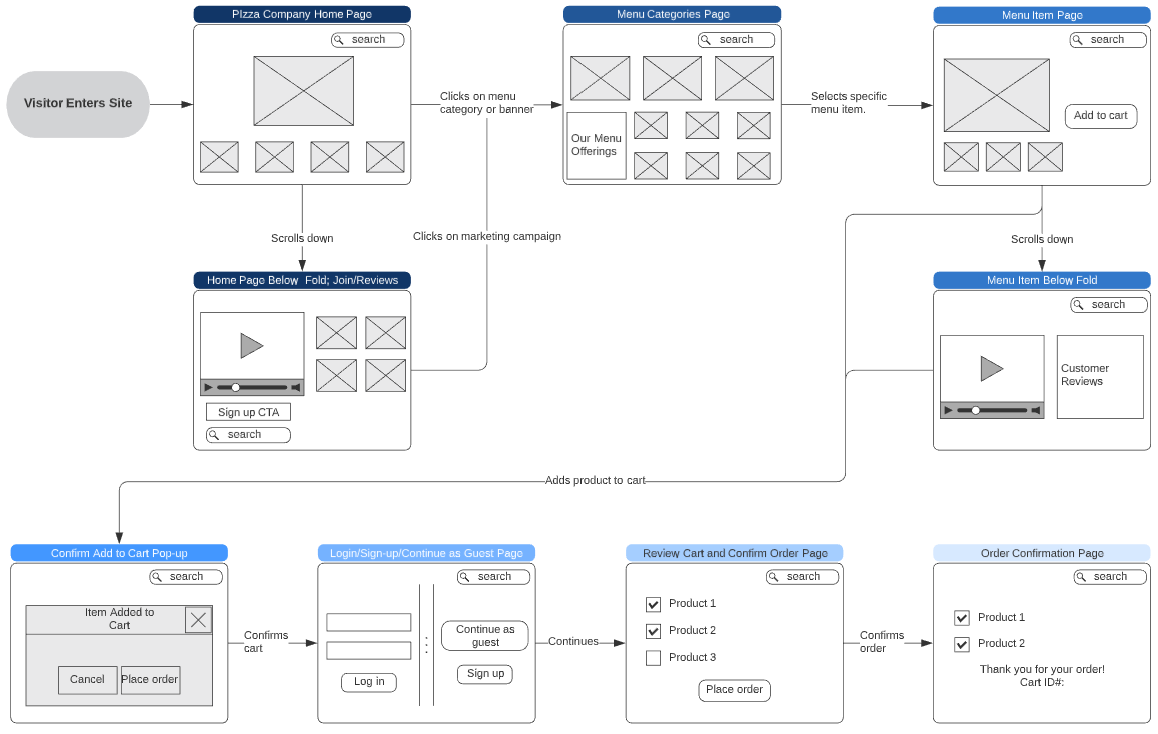
[Sequence Diagram](#_k97s3qauues8) **11**

[Appendix: Use Case Elaboration](#_oslgwbn7w7co) **12**

# 

# Minimum Viable Prototype

## User Interface Sketch



## Architectural Diagram

# 

# Class & Implementation Summary

The deliverable implements object classes and sample instances to demonstrate the end-to-end order creation, fulfillment and status update process.

* Main Objects: Customer, Fulfillment Store, Menu Inventory [ORM Databases]
* Cart Actions and Review [Iteration]
* Order Summary [Composition]
* System Mediator [FillStore Status Reporting]

## 

## Objects

### ORM Database for Customer, FillStore & MenuInventory

* Simulate Customer as ORM database; also a proxy representation for FillStore and Inventory databases due to categorically similar object purposes (to hold data attributes about the object only per my design).
* FillStore and Inventory are assumed because these would be implemented via an API link to the Pizza Store system in the real world as demonstrated in the system architecture section of this document.



## Design Patterns

### Factory: Inventory

* Simulate customer requests to view specialty pizza ingredient details in the app.



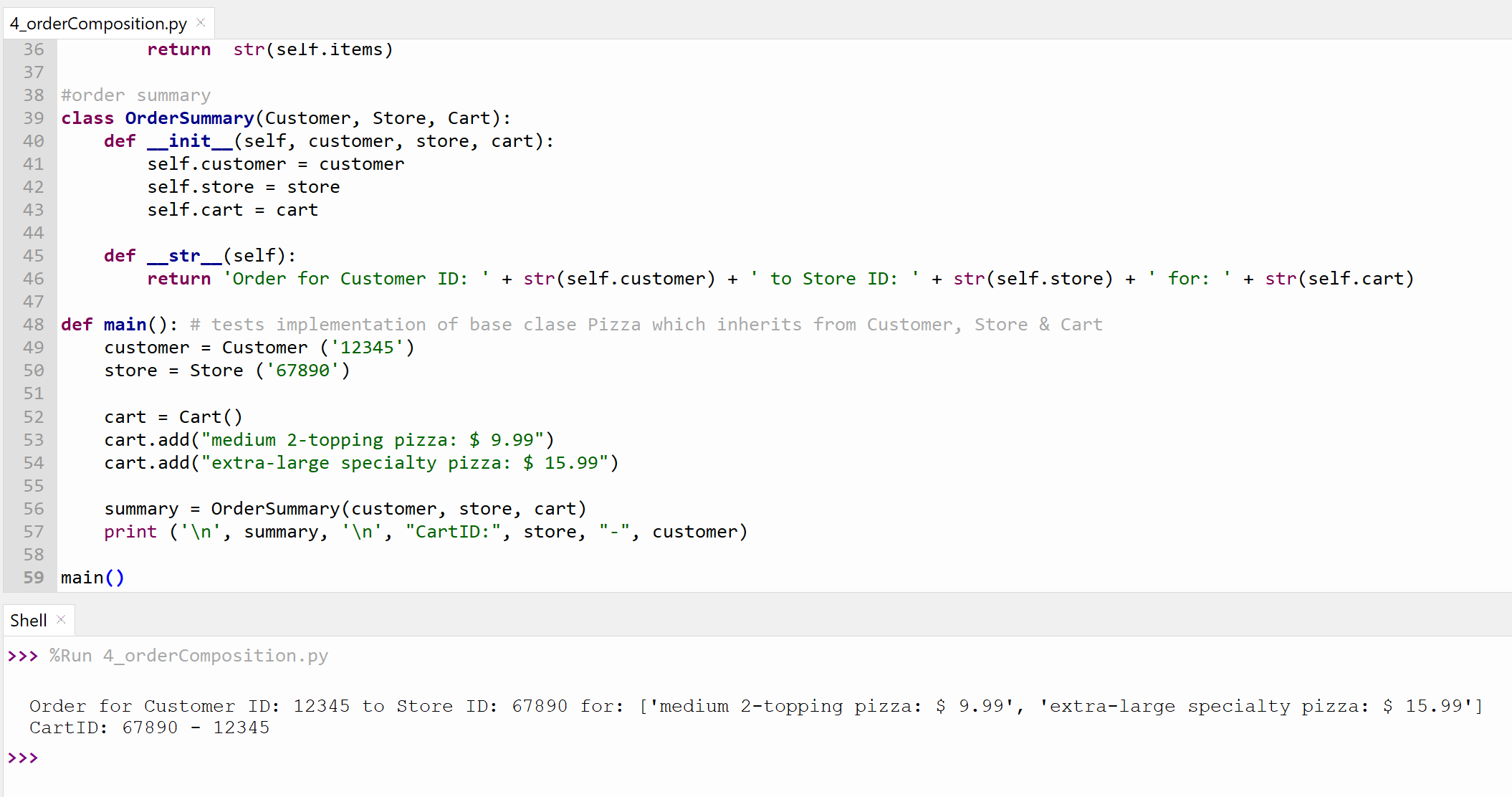
### Iteration: Cart Items

* Simulate mock items added to cart and subsequently retrieved from cart to add to Order Summary.

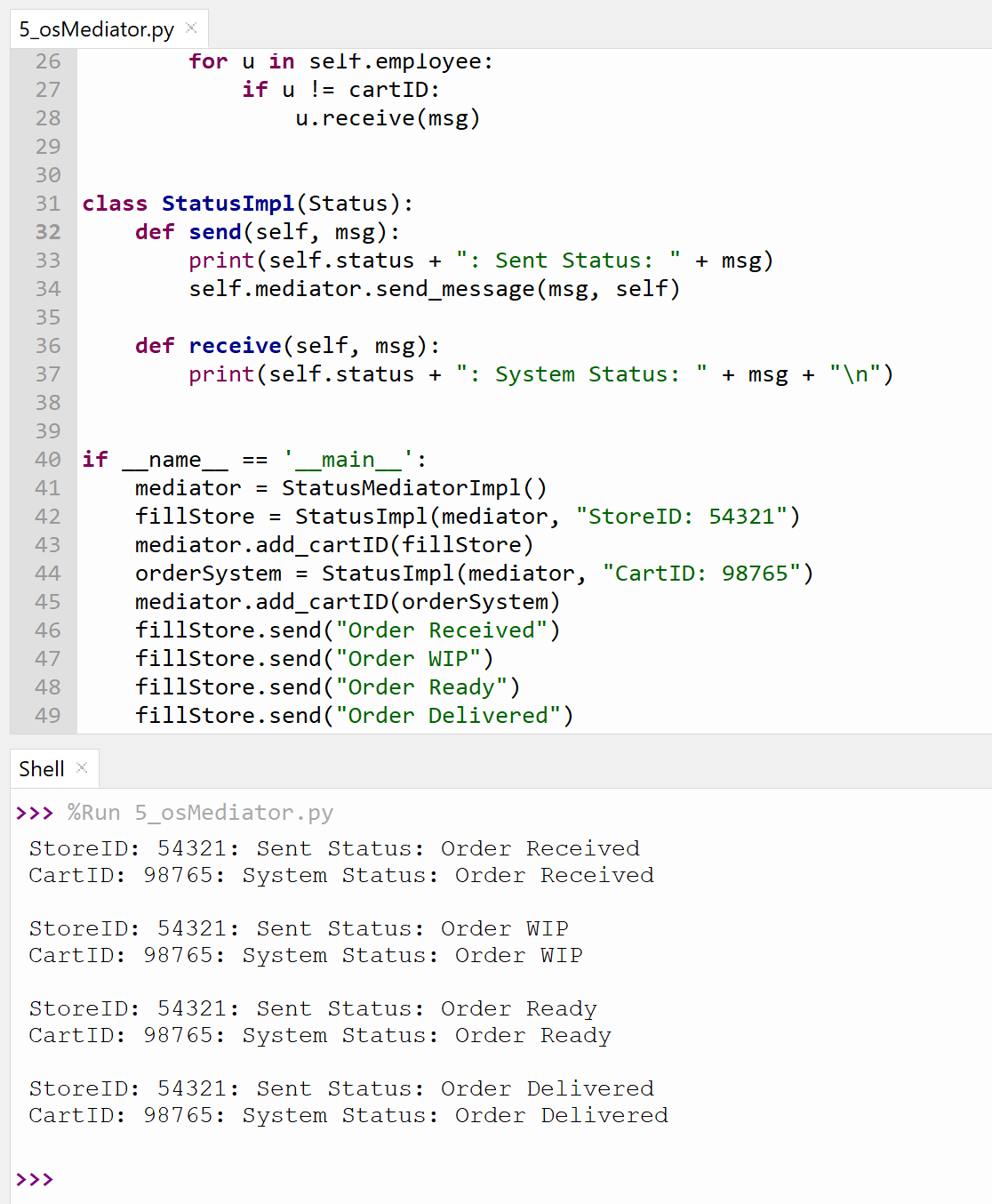


### Composition: Order Summary

* Simulate mock order summary of Customer, FillStore and Cart data.



### Mediator: Order System

* Simulate Bridge between Customer and Fillstore to provide Order Status notifications customer (i.e., received, wip, ready, delivered).

# Requirements (Use Cases)

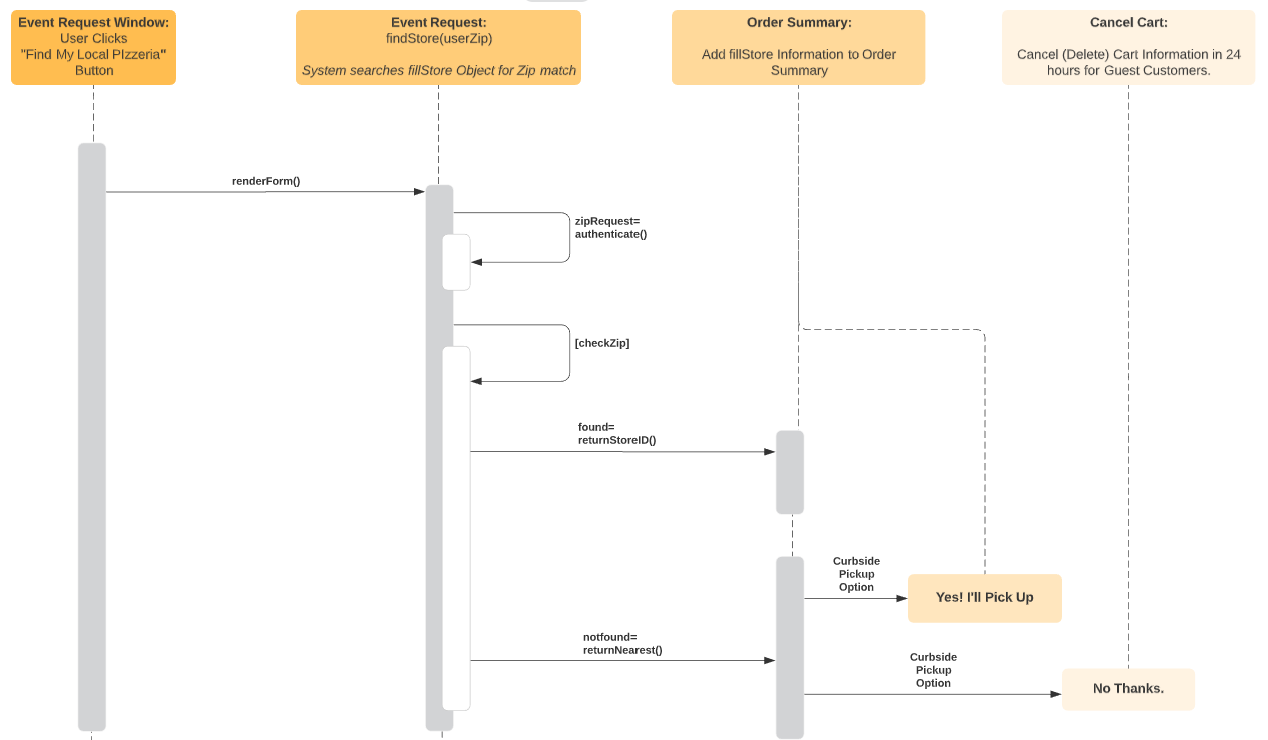
1. Customers can create orders as a guest (without email registry) or as a member.
2. Customers can create a registered member account and earn loyalty program rewards points for every dollar spent.
3. Customers can browse through menu items available to them for stores that offer delivery and/or pick-up to a specified address.
4. Customers can check out by making payments online with a credit/debit/gift card.
5. Customers can redeem coupons, gift cards, or rewards points by entering a unique code at checkout.
6. Customers can interact with the system to check on current order or past order information.
7. Customers can add items to cart from the past order information detail interface.
8. System will store a unique member ID to hold customer information such as name, address, email, payment information.
9. System will store a unique cart ID for hold order information for Guest Customer orders (name, address, fulfillment store ID, items ordered, etc.).
10. System will match customers with Fulfillment Store location using the Company Database that specifies each store location’s service range by zip code.
11. System will know the specific inventory items available at each Fulfillment Store using the Company Database linked to it’s inventory/product management systems via API.
12. System will forward orders to the appropriate Fulfillment Store.
13. Fulfillment Stores need to interact with the system to provide order status updates (e.g., order received, order in process, order ready for pick up, order on its way for delivery, order complete, etc.).
14. System will trigger alerts and/or warnings to the Customer and Fulfillment store based on elapsed service level response times (e.g. order not received at stored, delivery may be late, etc.)
15. System will trigger alerts for standard order progress updates to Customers.

# 

# Class Diagrams

# 

# Sequence Diagram



# 

# Appendix: Use Case Elaboration

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Create New Order | |
| **ID** | **UC001** | |
| **Scenario** | Guest Customer (GC) or Member Customer (MC) places online food orders. | |
| **Triggering Event** | GC/MC wants to order food online. | |
| **Brief Description** | GC wants to order menu items online without creating a Member Customer Account; MC's want to order menu items with faster checkout and earn rewards points. | |
| **Actors** | GC, MC, Order System (OS), Menu Inventory (MI), Fulfillment Store (FS) | |
| **Assumptions** | 1. GC is not in system and does not have a Member Account  2. MC is in system and does have a Member Account | |
| **Frequency of Use** | Daily, during operating hours only or schedule ahead if closed | |
| **Related Use Cases** | Retrieve Past Order(s); applies to MC's only | |
| **Stakeholders** | Fulfillment Stores, Pizza Company | |
| **Preconditions** | GC/MC has items in the shopping cart and is ready to checkout. | |
| **Postconditions** | GC/MC completes the checkout process and the system returns an order confirmation receipt to GC/MC. | |
| **Main Course** | 1. GC creates a new order online by browsing the menu and adding desired item(s) and quantity to the shopping cart. 2. When GC clicks "checkout" button, system will prompt three options: (a) Checkout as Guest, (b) Checkout as Member, or (c) Register for Member Rewards Account: 3. GC completes checkout use cases (review cart, retrieve saved payment info or add new, confirm delivery or pickup point and time). | |
| Alternate **Course** | 1. Not confirmed received in system by fulfillment store within SLA. 2. System will reprint/re-send order to fulfillment store with DUPLICATE in header.. 3. If the fulfillment store confirms received order in the system within SLA, then proceed to order status use cases. 4. If no confirmation receipt is returned, send email and/or text notification to GC/MC containing: (a) Order ID#, (b) Fulfillment Store Tel #, (c) Message: Dear Customer, there may be an issue with processing your order, please contact <Your Store> and provide <Your Order ID#> when speaking to one of our friendly on-site customer associates. 5. Algorithm **assumes** Customer Store directly; temp cart ID expires within 24 hours for GC (never for MC). 6. Fulfillment Store finds Order ID# in system, confirms receipt and proceeds with Order Status Use Cases. | |

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Member Customer Account Management | |
| **ID** | **UC002** | |
| **Scenario** | A Guest Customer (GC) wants to register as a member. | |
| **Triggering Event** | GC wants to become a member customer (MC). This can occur either at check out or when visiting the website home or main page. It is not required to place an order in order to create a member customer account. | |
| **Brief Description** | A Guest Customer (GC) wants to register as a member, save personal & payment information for faster checkout, and earn program rewards points. After the initial registry, MC’s can manage their personal information with encrypted password account login. | |
| **Actors** | GC, MC, Order System (OS) | |
| **Assumptions** | 1. GC is not in system and does not have a Member Account  2. MC is in system and does have a Member Account | |
| **Frequency of Use** | Daily, during system available hours (when site is not under maintenance). | |
| **Related Use Cases** | Retrieve Past Order(s). | |
| **Stakeholders** | Fulfillment Stores, Pizza Company | |
| **Preconditions** | GC is not previously in system records. | |
| **Postconditions** | GC becomes MC by completing member profile setup: login ID, password, address, store location, payment information, etc. | |
| **Main Course** | 1. GC click “register” button, system will prompt sections for user to complete in sections:    1. Personal Information    2. Delivery Address    3. Billing Address    4. Credit Card Information 2. System will allow users to review provided information in summary format. There will be links to edit or remove items in the cart pre-checkout summary. 3. GC clicks “register me!” button if they are satisfied with the summary of their personal account information. 4. System returns message confirming registration and returns a Loyalty Rewards Program ID# as the confirmation receipt. | |
| Alternate **Course** | Customer chooses not to register a member account. | |

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Add Items to Cart | |
| **ID** | UC003 | |
| **Scenario** | Guest Customer (GC) or Member Customer (MC) wants to browse the application or website to order food for DELIVERY or PICK-UP. | |
| **Triggering Event** | GC/MC is interested in food usually offered by pizza restaurants. | |
| **Brief Description** | System to display items available to the customer as defined for each fulfillment store; the number of product SKUs available vary across stores. | |
| **Actors** | GC, MC, Order System (OS), Menu Inventory (MI), Fulfillment Store (FS) | |
| **Assumptions** | Pizza Company/Fulfillment Stores have provided a database of all locations and scope of items carried at each Fulfillment Store. | |
| **Frequency of Use** | Daily, during operating hours only or schedule ahead if during non-business hours | |
| **Related Use Cases** | Remove Items from Cart | |
| **Stakeholders** | Fulfillment Stores, Pizza Company | |
| **Preconditions** | The use case for identifying fulfillment stores is completed. | |
| **Postconditions** | System displays menu items available per selected fulfillment store and customers may add/remove items or modify quantities of items in the cart. | |
| **Main Course** | 1. Customer browses menu items by category or keyword search (e.g. appetizers, beverages, desserts, pizzas, sides, etc.) 2. Customer adds quantities to cart by clicking “+” button next to pictured menu item to add item to cart with quantity 1 3. If item is already in cart and Customer clicks “+” it will increase quantity in cart by 1 4. System will display the number of items in the cart as a badge on the cart icon. 5. System allows Customer to expand a “mini-cart” when hovering over the cart icon, showing the image of item , quantity and order subtotal. | |
| **Alternate Course** | 1. An item that is temporarily out of stock cannot be added to cart. 2. When the Customer clicks the “+” button, the system will return “Temporarily Backordered Item” and will suggest 2-3 different items in-stock belonging to the same menu category. 3. Customers may proceed to add other items, but must forego items not in stock. | |

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Payment Processing | |
| **ID** | UC004 | |
| **Scenario** | Customer submits order for payment authorization. | |
| **Triggering Event** | Customers complete the checkout process by clicking the “Order Now” button. | |
| **Brief Description** | System will connect to API (e.g. paypal, square, etc.) for secure payment authorization. | |
| **Actors** | Order System (OS), Payment API-Interfaces | |
| **Assumptions** | The customer has provided payment information in the appropriate format. | |
| **Frequency of Use** | Daily, during operating hours or for schedule-ahead orders if order is placed during closed hours. | |
| **Related Use Cases** | Discount Code Redemption | |
| **Stakeholders** | Customers, Fulfillment Stores, Pizza Company | |
| **Preconditions** | Customer completed the checkout process and provided payment information. | |
| **Postconditions** | Payment processing services API authorizes payment securely. | |
| **Main Course** | 1. Customer has completed the check-out process and clicked the “Order Now” button. 2. System calls on payment processing API and stands by for response. 3. If successfully authorized, the system will generate an order confirmation (unique cart ID) and summarize contents of the order placed on screen. 4. System will give the customer receipt options to: download/save, view, print, and send as email. | |
| **Alternate Course** | 1. Issue with authorizing the provided payment information. 2. System will print message “unable to authorize payment method provided”. 3. System will prompt “do you wish to try an alternate payment method?” (Click “Yes” or “No”).    1. If Yes, the system will navigate to the payment information section of checkout and allow the user to provide new/different card information, or to select a card from their stored cards profile (if any).    2. If No, the system will prompt the customer on the options to contact fulfillment store or cancel order. (Click, “Contact Store” or “Cancel Order”). 4. If “Contact Store” is selected, System will print a proforma (or “mock”) order summary using Cart ID# and print message “There may be issues processing your payment online, Contact your store at <telephone number> and reference your <cart ID #> within <store hours of operation> for further assistance. Your cart items will be reserved for 48 hours.    1. If customer contacts store, Fulfillment Staff may try entering payment information on their local point-of-sale systems.    2. If payment is authorized, Fulfillment Staff need to convert the Proforma Order to Final Order by changing the order status appropriately in the system by looking up Card ID#. 5. If “Cancel Order” is selected, the system will prompt a review, “Are you sure you want to cancel?”.    1. System will cancel order if the customer clicks “Yes, I’m sure!”; or    2. System will go back to the alternate payment sequence (in steps 3-4) if the customer clicks “No, Let Me Try Again!” | |

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Discount Code Validation & Acceptance | |
| **ID** | UC005 | |
| **Scenario** | Customer Enters a code for coupon, gift card or rewards redemption at check out. | |
| **Triggering Event** | Customer enters a code in the “Promotional Code” field and clicks the “Apply to Order” button. | |
| **Brief Description** | System will connect the Pizza Company database to verify against the list of authorized codes and apply the discount to the pre-tax subtotal accordingly. | |
| **Actors** | Order System (OS), Pizza Company | |
| **Assumptions** | Pizza Company actively manages a database of valid codes and amount of discount that reduce cart subtotal price; system dependent on this source of truth for verification. | |
| **Frequency of Use** | Daily, during operating hours or for schedule-ahead orders if order is placed during closed hours. | |
| **Related Use Cases** | Payment Processing | |
| **Stakeholders** | Customers, Fulfillment Stores, Pizza Company | |
| **Preconditions** | Customer enters a code in the “Promotional Code” field and clicks the “Apply to Order” button. | |
| **Postconditions** | System accepts or rejects code entered; if valid the discount amount is shown as a price reduction line item below the pre-tax subtotal. | |
| **Main Course** | 1. Customer is in the check-out process of reviewing cart items. 2. After reviewing the cart items and before providing payment information, System will prompt for optional promotional code entry. 3. Customer enters code. 4. System calls on Pizza Company database and check code entered for a valid match. 5. If successfully authorized, the system will apply the discount amount to the pre-tax subtotal. | |
| **Alternate Course** | 1. Issue with authorizing the provided promotional code information. 2. System will print a message below the code entry text box “unable to authorize code provided” and will simultaneously clear the previously entered code. 3. The user will have the option to try entering the code until it is accepted or to leave the field blank in order to proceed to final checkout. | |

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Order Status | |
| **ID** | UC006 | |
| **Scenario** | Guest Customer (GC) or Member Customer (MC) wants to know the status of order placed. | |
| **Triggering Event** | Customer has placed an order and wants to track order status. | |
| **Brief Description** | Customer wants to track order status. | |
| **Actors** | Order System (OS), Fulfillment Store (FS) | |
| **Assumptions** | Customer has already placed an order and has an order confirmation ID and/or member ID. | |
| **Frequency of Use** | Daily, during operating hours only. | |
| **Related Use Cases** | Order Status: Make, Order Status: Ready for Pickup, Order Status: Delivery En Route, Order Status: Delivered | |
| **Stakeholders** | Customers, Fulfillment Stores, Pizza Company | |
| **Preconditions** | Customer has completed checkout and order successfully forwarded to print or display in Fulfillment Store systems. | |
| **Postconditions** | Staff at Fulfillment Store interacts with system to provide status update at the Order ID# level. | |
| **Main Course** | 1. Customer returns to the app/website and selects “Track My Order” from the navigation menu. 2. Customer enters Order ID# and email address in the text input boxes. This is the required path for Guest Customers (GC) to retrieve order status. Member Customers (MC) can use this method, or;    1. Alternatively, MC may login to their account and navigate to “My Orders” which will display all order history and latest order status for the customer in order from most recent to oldest. 3. System searches for Order ID# in the database and returns the most recent order status provided by the fulfillment store.    1. MC may also see all of their order history, noting that past orders will have “Order Complete/Closed” Status. | |
| **Alternate Course** | 1. System or Fulfillment Store issue (either order not sent at all, or fulfillment store did not confirm receipt): an order was created successfully but not confirmed received in system by fulfillment store within SLA. 2. System will reprint/resend order to fulfillment store with DUPLICATE annotated. 3. If the fulfillment store confirms order receipt after the resubmission, then proceed to order status use cases. 4. If no confirmation receipt is returned, send email and/or text notification to GC/MC containing: Order ID#, Fulfillment Store Tel #, and display, “Dear Customer, there may be an issue with processing your order, please contact <Your Store> and provide <Your Order ID#> when speaking to one of our friendly on-site customer associates.” 5. Assumes Customer calls Fulfillment Store Directly 6. Fulfillment Store staff look up Order ID# in system, confirms receipt in production system touch screen application and proceeds with Order Status Use Cases. | |

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Add Items to Cart from Past Order History | |
| **ID** | UC007 | |
| **Scenario** | Member Customer (MC) wants to browse the application or website to order food for DELIVERY or PICK-UP. | |
| **Triggering Event** | MC is reviewing past order details and wants to add one or all the items from a past order to the shopping cart. | |
| **Brief Description** | Valid items in a Member Customers order history can be added to cart. | |
| **Actors** | MC, Order System (OS), Menu Inventory (MI), Fulfillment Store (FS) | |
| **Assumptions** | Pizza Company/Fulfillment Stores have provided a database of all locations and scope of items carried at each Fulfillment Store.  System will display an “Add to Cart” button next to each item in the order detail review page if it is a valid and active product SKU. | |
| **Frequency of Use** | Daily, during operating hours only or schedule ahead if during non-business hours | |
| **Related Use Cases** | Add Items to Cart | |
| **Stakeholders** | Fulfillment Stores, Pizza Company | |
| **Preconditions** | Must be a Member Customer with at least one past order in transactional history. | |
| **Postconditions** | MC is able to add items from a past order detail view. | |
| **Main Course** | 1. Customer is reviewing order details from a past order; items are displayed similar to menu browse view inset on the Customer Order page. 2. Customer adds quantities to cart by clicking “+” button next to pictured menu item to add item to cart with quantity 1 3. If item is already in cart and Customer clicks “+” it will increase quantity in cart by 1 4. System will display the number of items in the cart as a badge on the cart icon. 5. System allows Customer to expand a “mini-cart” when hovering over the cart icon, showing the image of item , quantity and order subtotal. | |
| **Alternate Course** | An item that is temporarily out of stock cannot be added to cart.  When the Customer clicks the “+” button, the system will return “Temporarily Backordered Item” and will suggest 2-3 different items in-stock belonging to the same menu category.  Customers may proceed to add other items, but must forego items not in stock. | |