**PROJECT TITLE:**

Imtiaz Go

**GROUP MEMBERS:**

Huzaifa M. Arshad (21F-9323)

Usama Akram (21F-9217)

**USE CASE SPECIFICATIONS**

* **Enter Store:**

|  |  |
| --- | --- |
| **SECTION** | **CONTENT** |
| **DESIGNATION** | UC101 |
| **NAME** | Enter Store |
| **AUTHORS** | Huzaifa M. Arshad |
| **PRIORITY** | High |
| **CRITICALITY** | High |
| **SOURCE** | System Requirement Document (SRD) |
| **RESPONSIBLE** | * Customer * Store System |
| **DESCRIPTION** | This use case allows customers to enter the "Imtiaz Go" store by identifying themselves through a unique QR code or store app. This is the first step in creating a seamless shopping experience. |
| **TRIGGER EVENT** | Customer approaches store entrance. |
| **ACTORS** | * Customer * Store System |
| **PRE-CONDITION** | The customer has installed the "Imtiaz Go" app and has an account |
| **POST-CONDITION** | The customer is identified and allowed to enter the store for shopping. |
| **RESULT** | The customer enters the store and begins their shopping experience. |
| **MAIN SCENARIO** | 1. The customer opens the "Imtiaz Go" app on their smartphone. 2. The customer selects the 'Enter Store' option, generating a unique QR code. 3. The customer scans the QR code at the store entrance. 4. The system verifies the customer’s account. 5. Upon successful verification, the entrance gate opens. 6. The customer enters the store. |
| **ALTERNATIVE SCENARIO** | No alternate scenario. |
| **EXCEPTION SCENARIOS** | * The QR code does not scan properly. * The system cannot verify the customer’s account. |
| **QUALITIES** | 1. **Security:** Ensure secure verification of customer identity. 2. **Usability:** Entry process should be quick and straightforward. 3. **Efficiency:** System should verify customers rapidly to prevent entry delays. 4. **Reliability:** Entry mechanism should work consistently for all customers. |

* **SCAN PRODUCT:**

|  |  |
| --- | --- |
| **SECTION** | **CONTENT** |
| **DESIGNATION** | UC102 |
| **NAME** | SCAN PRODUCT |
| **AUTHORS** | HUZAIFA M. ARSHAD |
| **PRIORITY** | High |
| **CRITICALITY** | Critical |
| **SOURCE** | System Requirement Document |
| **RESPONSIBLE** | Store System |
| **DESCRIPTION** | Customers select items they wish to purchase by simply adding them to their physical cart. The store’s system automatically detects these items and adds them to the customer’s virtual cart. |
| **TRIGGER EVENT** | Customer picks up an item |
| **ACTORS** | * Customer * Store System |
| **PRE-CONDITION** | Customer is inside the store and has been successfully identified. |
| **POST-CONDITION** | Items are added to the customer’s virtual shopping cart. |
| **RESULT** | The customer's virtual cart reflects all selected items. |
| **MAIN SCENARIO** | 1. The customer picks up an item from the shelf. 2. The store’s system detects the item through RFID/NFC or similar technology. 3. The item is automatically added to the customer’s virtual cart. 4. The customer continues shopping or selects more items. |
| **ALTERNATIVE SCENARIO** | Customer decides to return an item to the shelf. |
| **EXCEPTION SCENARIOS** | An item is picked up but not detected by the system.  The customer picks up one item but the system adds a different item.  The system adds an item multiple times by mistake. |
| **QUALITIES** | 1. **Accuracy:** The system must accurately track items picked up or returned by the customer. 2. **Usability:** Customers should not need to perform any additional actions to add items to their cart. 3. **Efficiency:** Item detection and addition to the virtual cart should happen in real-time. 4. **Reliability:** The system should consistently detect all customer interactions with items. 5. **Error Handling:** The system should have mechanisms to handle common issues like product not scanning, incorrect barcodes, or items with missing barcodes. 6. **Security:** Ensure that scanned data is secure and that the system cannot be easily manipulated to record incorrect products. |

* **Checkout:**

|  |  |
| --- | --- |
| **SECTION** | **CONTENT** |
| **DESIGNATION** | UC103 |
| **NAME** | Checkout |
| **AUTHORS** | HUZAIFA M. ARSHAD |
| **PRIORITY** | High |
| **CRITICALITY** | Critical |
| **SOURCE** | System Requirement Document |
| **RESPONSIBLE** | Store System |
| **DESCRIPTION** | This use case involves automatically calculating the total cost of items selected by the customer, applying any applicable discounts or promotions, and processing payment using the customer's linked payment method. |
| **TRIGGER EVENT** | Customer is exiting the store. |
| **ACTORS** | * Store System |
| **PRE-CONDITION** | Customer has completed item selection. |
| **POST-CONDITION** | Payment is successfully processed, and the customer's account is charged. |
| **RESULT** | Seamless completion of the shopping process without the need for manual intervention. |
| **MAIN SCENARIO** | 1. The customer indicates the completion of shopping by exiting the store or confirming the end of their shopping session in the app. 2. The system retrieves the list of items selected by the customer and their respective prices. 3. Any applicable discounts or promotions are applied to the total cost. 4. The system securely processes the payment using the customer's linked payment method (e.g., credit card, digital wallet). 5. Upon successful payment processing, the system generates a digital receipt. |
| **ALTERNATIVE SCENARIO** | Customer's payment method fails to process, and the system prompts for an alternative payment method. |
| **EXCEPTION SCENARIOS** | 1. Technical issues prevent the system from retrieving item prices or applying discounts. 2. Payment processing fails due to network connectivity issues. |
| **QUALITIES** | 1. **Accuracy:** The system must accurately calculate the total cost of items, apply discounts, and process payments to ensure that customers are charged the correct amount. 2. **Reliability:** Automatic billing should be reliable, ensuring that payments are processed securely and consistently without errors or interruptions. 3. **Efficiency:** The billing process should be efficient, minimizing the time taken to calculate the total cost and process payments to provide a seamless shopping experience. 4. **Security:** Payment processing must be secure to protect customer payment information and prevent unauthorized access or fraudulent activities**.** |

* **Send Receipt:**

|  |  |
| --- | --- |
| **SECTION** | **CONTENT** |
| **DESIGNATION** | UC104 |
| **NAME** | Send Receipt |
| **AUTHORS** | HUZAIFA M. ARSHAD |
| **PRIORITY** | Medium |
| **CRITICALITY** | High |
| **SOURCE** | System Requirement Document |
| **RESPONSIBLE** | Store System |
| **DESCRIPTION** | This use case involves automatically generating and sending a digital receipt to the customer's registered email address upon successful completion of the automatic billing process. |
| **TRIGGER EVENT** | Successful completion of the Automatic Billing process. |
| **ACTORS** | * Store System |
| **PRE-CONDITION** | Automatic Billing process completes successfully. |
| **POST-CONDITION** | Digital receipt is sent to the customer's email address. |
| **RESULT** | Customer receives a record of their purchase. |
| **MAIN SCENARIO** | 1. Upon successful completion of the Automatic Billing process, the system generates a digital receipt containing details of the transaction (e.g., items purchased, total amount, date and time). 2. The system retrieves the customer's registered email address. 3. The system securely sends the digital receipt to the customer's email address using the Email System. |
| **ALTERNATIVE SCENARIO** | Customer's email address is not available, and the system prompts for an alternative method of receipt delivery (e.g., SMS, in-app notification). |
| **EXCEPTION SCENARIOS** | Technical issues prevent the generation or sending of the digital receipt.  Customer's email server rejects receipt delivery due to spam filters or other restrictions. |
| **QUALITIES** | 1. **Accuracy:** The digital receipt should accurately reflect the details of the transaction, including items purchased, total amount, date, and time, to provide customers with an accurate record of their purchase. 2. **Reliability:** Sending receipts should be reliable, ensuring that customers receive their receipts promptly after completing the transaction without delays or failures. 3. **Accessibility:** Receipts should be accessible to customers through their preferred communication channels (e.g., email, SMS, in-app notification), ensuring convenience and ease of access. |

* **Submit Complaints:**

|  |  |
| --- | --- |
| **SECTION** | **CONTENT** |
| **DESIGNATION** | UC105 |
| **NAME** | Submit Complaints |
| **AUTHORS** | HUZAIFA M. ARSHAD |
| **PRIORITY** | Medium |
| **CRITICALITY** | Medium |
| **SOURCE** | System Requirement Document |
| **RESPONSIBLE** | Store Admin |
| **DESCRIPTION** | This use case allows customers to submit complaints about their shopping experience, including issues with billing, product quality, or any other concerns directly through the app or an in-store kiosk. |
| **TRIGGER EVENT** | Customer Initiates Complain request |
| **ACTORS** | * Customer * Store Admin |
| **PRE-CONDITION** | The customer has completed a shopping session or has an issue related to the store services. |
| **POST-CONDITION** | The complaint is logged, and the customer receives a confirmation with a tracking number. |
| **RESULT** | Customer issues are addressed, improving customer satisfaction. |
| **MAIN SCENARIO** | 1. The customer accesses the complaint submission feature via the "Imtiaz Go" app or an in-store kiosk. 2. The customer fills out a complaint form, providing relevant details and any evidence. 3. The system logs the complaint and provides the customer with a confirmation and tracking number. 4. A store admin or customer service representative reviews the complaint. 5. Appropriate actions are taken to resolve the complaint. 6. The customer is informed about the complaint resolution. |
| **ALTERNATIVE SCENARIO** | Customer contacts customer service directly for urgent issues. |
| **EXCEPTION SCENARIOS** | 1. The system fails to log the complaint due to technical issues. 2. Customer submits a complaint but provides insufficient details to take action. |
| **QUALITIES** | 1. **Accessibility:** Customers can easily find and use the complaint submission feature. 2. **Responsiveness:** Complaints are addressed in a timely manner. 3. **Transparency:** Customers are kept informed about the status of their complaint. |

* **Manage Inventory:**

|  |  |
| --- | --- |
| **SECTION** | **CONTENT** |
| **DESIGNATION** | UC106 |
| **NAME** | Manage Inventory |
| **AUTHORS** | HUZAIFA M. ARSHAD |
| **PRIORITY** | High |
| **CRITICALITY** | Critical |
| **SOURCE** | System Requirement Document |
| **RESPONSIBLE** | Store Admin |
| **DESCRIPTION** | This use case involves the store admin or inventory manager overseeing the store’s inventory levels, restocking items, and removing outdated or expired products. |
| **TRIGGER EVENT** | Store Admin wants to check or change store’s inventory. |
| **ACTORS** | Store Admin |
| **PRE-CONDITION** | The store operates with an inventory that needs regular updating and management. |
| **POST-CONDITION** | Inventory is accurately tracked and maintained, ensuring products are available for customers. |
| **RESULT** | Efficient inventory management leads to better product availability and reduced losses from expired goods. |
| **MAIN SCENARIO** | 1. The Inventory Management System alerts the store admin about low stock levels or expired products. 2. The store admin reviews inventory reports and decides on restocking or removal of products. 3. Orders are placed for new stock, and expired products are removed from the shelves. 4. The Inventory Management System is updated to reflect changes. |
| **ALTERNATIVE SCENARIO** | No alternate scenario. |
| **EXCEPTION SCENARIOS** | 1. A discrepancy between the physical stock and the Inventory Management System records. 2. Failed delivery or restocking leads to stock shortages. |
| **QUALITIES** | 1. **Accuracy:** The Inventory Management System accurately reflects stock levels. 2. **Efficiency:** Inventory tasks are streamlined and automated where possible. 3. **Reliability:** The system reliably tracks and manages inventory, avoiding out-of-stock situations. |

* **Check Insights:**

|  |  |
| --- | --- |
| **SECTION** | **CONTENT** |
| **DESIGNATION** | UC107 |
| **NAME** | Check Insights |
| **AUTHORS** | HUZAIFA M. ARSHAD |
| **PRIORITY** | Medium |
| **CRITICALITY** | Medium |
| **SOURCE** | System Requirement Document |
| **RESPONSIBLE** | Store Admin |
| **DESCRIPTION** | This use case allows store admins or analysts to access and analyze sales data, customer behavior, and inventory levels to gain insights into store performance and customer preferences. |
| **TRIGGER EVENT** | Request for data analysis or report generation. |
| **ACTORS** | Store Admin |
| **PRE-CONDITION** | The store collects data from sales, customer interactions, and inventory management. |
| **POST-CONDITION** | Store admins have access to insights that help in making informed decisions. |
| **RESULT** | Improved store operations, marketing strategies, and customer satisfaction. |
| **MAIN SCENARIO** | 1. The store admin accesses the Data Analysis System. 2. The admin selects the type of insights or reports needed. 3. The system processes the request and generates insights or reports. 4. The admin reviews the insights to make informed decisions on store management, inventory, and marketing strategies. |
| **ALTERNATIVE SCENARIO** | Automated insights are generated weekly and sent to the admin’s dashboard. |
| **EXCEPTION SCENARIOS** | The Data Analysis System fails to generate reports due to technical issues. |
| **QUALITIES** | 1. **Accuracy:** The insights provided should accurately reflect store performance, customer behavior, and inventory trends, ensuring that decision-making is based on reliable data. 2. **Relevance:** Insights should be relevant to the needs of store administrators and analysts, providing actionable information that helps improve store operations, marketing strategies, and customer satisfaction**.** |