

**SWE 445**

**Phase #1**

**Software Requirements Specifications**

**Group #4**

**Software Requirements Specifications**

* 1. Identify all actors, use cases and develop a use case model of the system.
     + **Actors:**

- Customer

- System

- Bank

* + - **Use Cases:**

- Sign-up

- Log-in

- View Item

- Search item

- Browse item

- View Recommended items

- Add item to shopping cart

- Add item to wish list

- Add payment method

- Checkout

- Track order

- Verify payment

- Check credentials

- Save card details

- Save purchase history

- Verify Transaction

- Transfer funds

* 1. Using the results of section (a), identify mis-use case actors, mis-use cases and develop a mis-use case model of the system
     + **Mis-use case actors:**

**-** Attacker

* + - **Mis-use cases:**

- Steal credentials

- Steal card details

- Make a purchase

- SQL injection

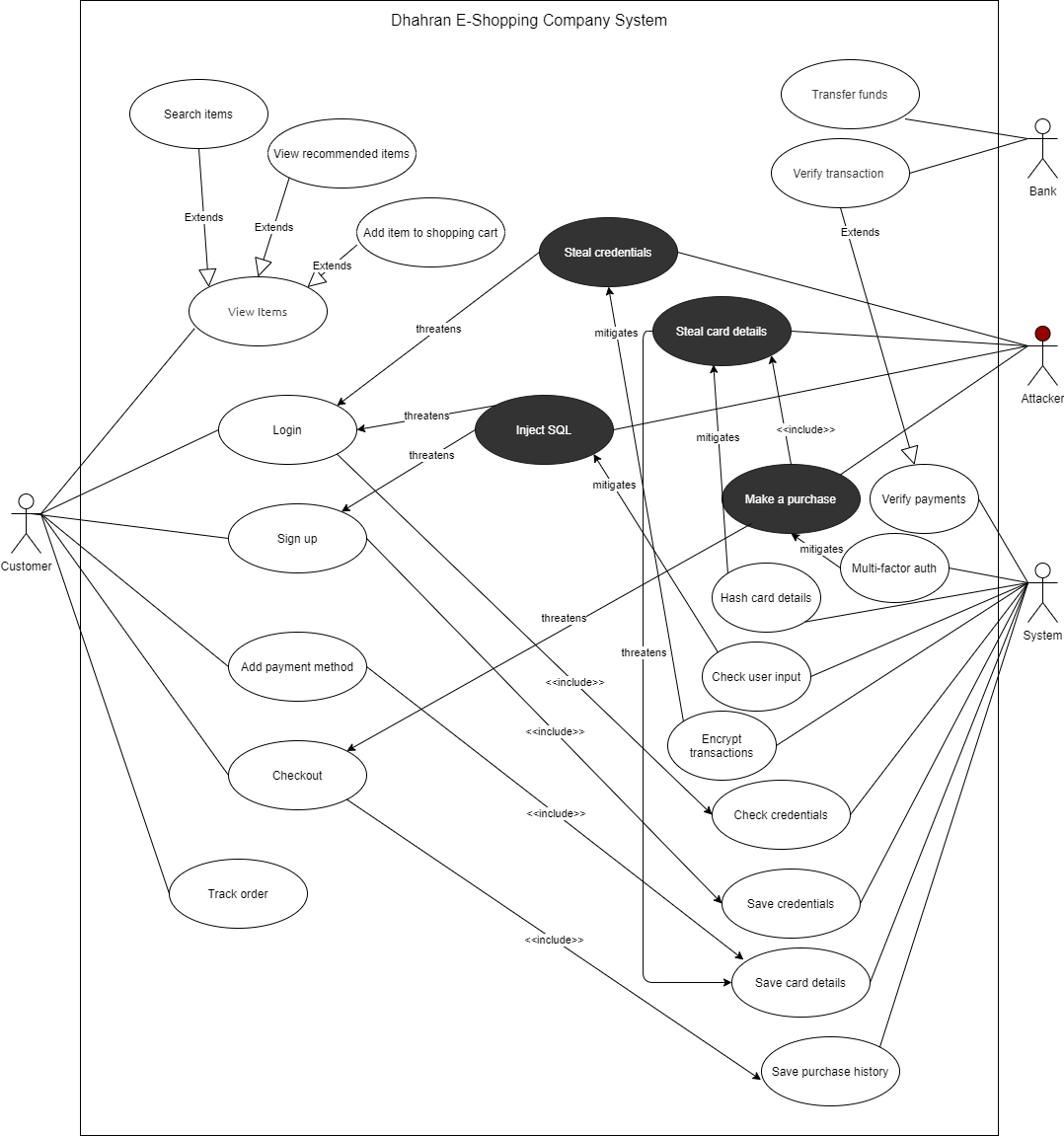
* 1. Add use cases (in the final model) to mitigate mis-use cases identified in section (b).

-Encrypt data (for stealing credentials)

-Hash card details ( for stealing card details)

-Check user input (for SQL injection)

-Multi-factor authentication (for Make a purchase)



* 1. Write description for all identified use cases/mis-use cases.

|  |  |
| --- | --- |
| UC-01: Sign-up | |
| Description: | This use case allows customer to register in our system |
| Actors: | Customer |
| Main Flow: | 1. The customer opens our Home page 2. The customer clicks on “Sign-up” 3. The customer fills the registration form 4. The customer submits the form |
| Alternative(s): | In step 3 of the Main flow, if the customers details are already in our system, the system informs the customers and prompts them to enter different details. |

|  |  |
| --- | --- |
| UC-02: Log in | |
| Description: | This use case allows customer to log into our system |
| Actors: | Customer |
| Main Flow: | 1. The customer opens our Home page 2. The customer clicks on “Log in” 3. The customer fills the log-in form 4. The customer submits the form |
| Alternative(s): | In step 3 of the Main flow, if the customers details are not in our system, the system informs the customers and prompts them to register first. |

|  |  |
| --- | --- |
| UC-03: View Item | |
| Description: | The customer can view the item’s price, image (if available), description, and the seller. |
| Actors: | Customer |
| Main Flow: | 1. The customer opens our website 2. The customer navigates to an item 3. The system displays the item’s information |
| Alternative(s): | In step 2 of the Main flow, the customer can navigates to an item through:   * Searching for an item * Browsing our collection * Viewing the recommended items * The customers’ wish list * The customers’ shopping cart |

|  |  |
| --- | --- |
| UC-04: Checkout | |
| Description: | The customers can checkout and complete purchases after adding items to the shopping cart |
| Actors: | Customer |
| Main Flow: | 1. The customer opens the shopping cart 2. The customer clicks on “Checkout” 3. The customer chooses a payment method 4. The system processes the purchase |
| Alternative(s): | In step 2 of the Main flow, if the customer is not logged in, the system prompts the customer to log in first. |

|  |  |
| --- | --- |
| UC-05: Search Item | |
| Description: | Customers can search items by name |
| Actors: | Customer |
| Main Flow: | 1. The customer navigates to the search bar 2. The customer types the item’s name 3. The system displays a list of items matching the search keywords |
| Alternative(s): | In step 3 of the Main flow, if the system did not find any matching items, no items will be displayed |

|  |  |
| --- | --- |
| UC-06: Browse Items | |
| Description: | The customer could browse our collections of items |
| Actors: | Customer |
| Main Flow: | 1. The system display items on the Home page 2. The customer filers items based on category or price |
| Alternative(s): | No alternative flow |

|  |  |
| --- | --- |
| UC-07: View Recommended Items | |
| Description: | The system recommends items based on the users’ activities on the website |
| Actors: | Customer |
| Main Flow: | 1. The systems display recommended items on the Home page 2. The customer chooses items based on interest |
| Alternative(s): | No alternative flow |

|  |  |
| --- | --- |
| UC-08: Add Item to Shopping Cart | |
| Description: | Customers could add items to their shopping cart |
| Actors: | Customer |
| Main Flow: | 1. The customer navigates to the desired items 2. The customer enters the wanted quantity 3. The customer adds the item to the shopping cart |
| Alternative(s): | In step 2 of the Main flow, if the item is out of stock, the system notifies the customer and shows similar items |

|  |  |
| --- | --- |
| UC-09: Add Item to Wish List | |
| Description: | Customers could add items to their Wish list |
| Actors: | Customer |
| Main Flow: | 1. The customer navigates to the desired items 2. The customer clicks on “add to wish list” |
| Alternative(s): | In step 2 of the Main flow, if the customer is not logged in, the system prompt the customer to log in before adding items to with list |

|  |  |
| --- | --- |
| UC-10: Add Payment Method | |
| Description: | The customer can add a payment method. |
| Actors: | Customer |
| Main Flow: | 1- The customer navigates to the payment method  2- The customer chooses a payment method  3- The customer enters the card information |
| Alternative(s): |  |

|  |  |
| --- | --- |
| UC-11: Track order | |
| Description: | The customer can track an order, the system should display the order tracking details |
| Actors: | Customer |
| Main Flow: | 1- The customer goes to the tracking page  2- The customer enters the tracking number  3- The backend checks for the tracking number  4- The order details shows up |
| Alternative(s): | In step 3 of the main flow , if the user provides an invalid tracking number the system alerts the customer to enter a correct tracking number |

|  |  |
| --- | --- |
| UC-12: Verify payment | |
| Description: | After the customer makes a purchase, our system should verify the payment with the bank |
| Actors: | Server |
| Main Flow: | 1. The customer makes a purchase 2. The server sends payment to the bank 3. The bank checks the payment 4. The bank transfer funds to the server |
| Alternative(s): | In step 3 of the Main flow, if the user’s balance in not enough for the payment, the bank rejects the transaction |

|  |  |
| --- | --- |
| UC-13: Check credentials | |
| Description: | When the customer logs in, the server checks the credentials to either approve or reject the log in attempt |
| Actors: | Server |
| Main Flow: | 1. The customer logs in 2. The server checks the credentials entered by the user 3. The server allows the customer to log into the account |
| Alternative(s): | In step 2 of the Main flow, if the credentials entered by the users are incorrect, the server rejects the log in attempt |

|  |  |
| --- | --- |
| UC-14: Save credentials | |
| Description: | Once the customer signs up to the system , in the backend store the provided data in the database  After a customer adds |
| Actors: | Server |
| Main Flow: | 1- The customer provides credentials  2-The Server takes the data and stores it in the database |
| Alternative(s): | in step 1 of the main flow, if the credentials provided are already available in the database , alert the customer |

|  |  |
| --- | --- |
| UC-15: Save card details | |
| Description: | Save the card details, when a payment method is added , in the database  After a customer makes a purchase, the systems saves it |
| Actors: | Server |
| Main Flow: | 1- The customer enters  the card information  2- The server checks the information  3- The server takes the information of the card provided and stores it in the database |
| Alternative(s): | No alternative flow |

|  |  |
| --- | --- |
| UC-16: Save purchase history | |
| Description: | After a customer makes a purchase, the server saves it |
| Actors: | Server |
| Main Flow: | 1- The customer makes a purchase  2- The server saves that purchase in the database |
| Alternative(s): | No alternative flows |

|  |  |
| --- | --- |
| UC-17: Verify Transaction | |
| Description: | The bank receives the transaction from the server to verify it |
| Actors: | Bank |
| Main Flow: | 1- The server sends the transaction information to the bank  2- the bank receives the transaction  3- The bank sends the verification bank to the server |
| Alternative(s): | In step 3 in the main flow, if the transaction is not verified by the bank , send an alert |

|  |  |
| --- | --- |
| UC-18: Encrypt data | |
| Description: | The data must be encrypted before sending it on the network |
| Actors: | System |
| Main Flow: | 1. The customer enters sensitive data 2. The system encrypts that data |
| Alternative(s): | No alternative flow |

|  |  |
| --- | --- |
| UC-19: Hash card details | |
| Description: | The system must hash all users’ card details |
| Actors: | System |
| Main Flow: | 1. The customer enters the cards details 2. The system hashes the details 3. The hashed details are stored in the database |
| Alternative(s): | No alternative flow |

|  |  |
| --- | --- |
| UC-20: Check user’s input | |
| Description: | Users’ input must be checked before processing |
| Actors: | System |
| Main Flow: | 1. The customer enters any data 2. The system checks the input 3. The system process that input |
| Alternative(s): | In step 2 of the Main flow, if the input of the user is invalid, the system rejects the input |

|  |  |
| --- | --- |
| UC-21: Multi-factor authentication | |
| Description: | The system must use multi-factor authentication before processing a purchase |
| Actors: | System |
| Main Flow: | 1. The customer checkout a purchase 2. The system prompts the user for the card’s security code 3. The system sends a nonce to the phone number associated with that card 4. The system verifies the checkout process |
| Alternative(s): | In step 3 of the Main flow, if the customer fails to enter the right nonce, the system will cancel the purchase |

Misuse case

|  |  |
| --- | --- |
| UC-01: Steal Credentials | |
| Description: | The process used to steal credentials. |
| Actors: | Attacker |
| Main Flow: | 1- The customer logs into the system  2- The attacker tries to gather the credentials of the customer  3- The attacker steals the credentials and can use it in the system or can use it |
| Alternative(s): |  |
| Mitigation Points | Provide an encryption mechanism |
| Triggers | Always true, it can happen at any time |
| Assumptions | The system does not have an encryption mechanism |

|  |  |
| --- | --- |
| UC-02: Steal Card Details | |
| Description: | The process used by the attacker to steal card details |
| Actors: | Attacker |
| Main Flow: | 1- the customer enters the information of a card  2- the attacker tries to take the information of the card  3- the attacker steals the card and can use it |
| Alternative(s): |  |
| Mitigation Points | Use a hashing function to hash the details of the cards |
| Triggers | Always true, it can happen at any time |
| Assumptions | The system does not provide a hashing mechanism to hash details |

|  |  |
| --- | --- |
| UC-03: Make a purchase | |
| Description: | The attacker makes a purchase in behalf of the customer without his knowledge |
| Actors: | Attacker |
| Main Flow: | 1- the attacker gets the card details of a customer  2- the attacker adds some items  3- the attacker proceeds to checkout  4- the attacker buys those items |
| Alternative(s): |  |
| Mitigation Points | Use Multi-factor authentication to ensure that the customer is making the purchase |
| Triggers | Always true, this can happen at any time |
| Assumptions | The system does not provide any authentication mechanism |

|  |  |
| --- | --- |
| UC-04: SQL Injection | |
| Description: | The attacker performs SQL injection to change and destroy information in the system |
| Actors: | Attacker |
| Main Flow: | 1- the attacker successfully enter the database  2- the attacker proceeds to make changes in the database  3- the database is modified by the attacker |
| Alternative(s): |  |
| Mitigation Points | Check the inputs provided and perform integrity checking |
| Triggers | Always true, this can happen at any time |
| Assumptions | The system does not check inputs |