Use-case and class diagram

1.

|  |  |
| --- | --- |
| User requests to exit the application. |  |
|  | System attempts to save the data and if successful, application terminates. If failed the system tells user about the failure and asks if user would like to try again. |
| If system fails to save data user answers yes or no to trying again. |  |
|  | If yes, system goes back to step 2. If no, application terminates. |

2.

|  |  |
| --- | --- |
| User requests to add a client. |  |
|  | System asks for client’s name, address, and phone number. |
| User enters client’s name, address, and phone number. |  |
|  | System generates a unique ID for the client, sets the client’s balance to 0, and attempts to enter the client’s information. It then informs user of the result and asks if user would like to add another client. |
| User answers yes or no. |  |
|  | If yes, system goes back to step 2. If no, system returns to main menu. |

3.

|  |  |
| --- | --- |
| User requests to remove a client. |  |
|  | System asks for the unique ID of the client. |
| User enters the unique ID of the client they wish to remove. |  |
|  | System attempts to remove the client and informs user of the results. System then asks if the user would like to remove another client. |
| User answers yes or no. |  |
|  | If yes, system goes back to step 2. If no, system returns to main menu. |

4.

|  |  |
| --- | --- |
| User requests for the system to generate and output a list of all clients. |  |
|  | System attempts to retrieve all information for each client. If all clients’ information is retrieved, the information is displayed to the user. If system failed to generate the list it outputs an error message and asks if user would like to try again. |
| If system fails to generate list, user answers yes or no to trying again. |  |
|  | If yes, system goes back to step 2. If no, system returns to main menu. |

5.

|  |  |
| --- | --- |
| User requests to add a new customer. |  |
|  | System asks user for the customer’s name, address, phone number, and exactly one credit card number along with that card’s expiration date. |
| User enters the customer’s name, address, phone number, and exactly one credit card number along with that card’s expiration date. |  |
|  | System generates a unique ID for the customer and attempts to enter the customer’s information. It then informs the user of the result and asks if user would like to add another customer. |
| User answers yes or no. |  |
|  | If yes, system goes back to step 2. If no, system returns to the main menu. |

6.

|  |  |
| --- | --- |
| User requests to remove a customer. |  |
|  | System asks for the unique ID of the customer. |
| User enters the unique ID of the customer they wish to remove. |  |
|  | System attempts to delete all information pertaining to the specified customer ID and informs user of the result. System then asks if user would like to remove another customer. |
| User answers yes or no. |  |
|  | If yes, system goes back to step 2. If no, system returns to main menu. |

7.

|  |  |
| --- | --- |
| User requests to add a credit card. |  |
|  | System asks for the customer ID to be tied to the card, the card number, and the expiration date of the card. |
| User enters the customer ID, the card number, and the card’s expiration date. |  |
|  | System attempts to save the card’s information and informs user of the result. System then asks is the user would like to add another credit card. |
| User answers yes or no. |  |
|  | If yes, system goes back to step 2. Is no, system returns to the main menu. |

8.

|  |  |
| --- | --- |
| User requests to remove a credit card. |  |
|  | System asks for the credit card number. |
| User enters the credit card number. |  |
|  | System attempts to remove the card. If this is the only card attributed to the customer ID it holds, the system refuses to remove its information. The user is informed of results appropriately before system asks if they would like to try removing another card. |
| User answers yes or no. |  |
|  | If yes, system goes back to step 2. If no, system returns to the main menu. |

9.

|  |  |
| --- | --- |
| User requests for the system to generate and output a list of all customer information. |  |
|  | System attempts to retrieve all customer information, including all credit card information attributed to the customers ID’s, and informs user of the results. System then generates an organized list of all acquired information and outputs it. If system failed to generate the list it outputs an error message instead and asks if user would like to try again. |
| If system fails to generate list, user answers yes or no to trying again. |  |
|  | If yes, system goes back to step 2. If no, system returns to the main menu. |

10.

|  |  |
| --- | --- |
| User requests to add a play. |  |
|  | System asks for the play’s name, the client ID of the client wishing to present the play, and the dates that the client wishes to use the theater. |
| User enters the name of the play, the client ID of the client wishing to present the play, and the dates that the client wishes to use the theater. |  |
|  | System attempts to add the play. Is even one of the dates desired is already scheduled to be used by another client the system will be unable to add the play. System informs user of the result appropriately. If system is unable to add the play due to scheduling issue it asks user if they would like to change the desired dates. If play is successfully added the system asks user if they would like to add another play. |
| User answers yes or no to either question posed by system. |  |
|  | If yes to either question, system goes back to step 2. If no to either question, system returns to the main menu. |

11.

|  |  |
| --- | --- |
| User requests for the system to generate a list of all plays. |  |
|  | System attempts to retrieve all names of plays as well as the dates associated with them and informs user of results. If successful, system generates an organized list an outputs it. If failed, system asks user if they would like to try again. |
| If system fails to generate list user answers yes or no to trying again |  |
|  | If yes, system goes back to step 2. If no, system returns to the main menu. |

12.

|  |  |
| --- | --- |
| User requests to store all data. |  |
|  | System attempts to store all information and informs user of the results. If failed, system asks if the user would like to try again |
| If save failed user answers yes or no to trying again. |  |
|  | If yes, system goes back to step 2. If no, system returns to the main menu. |

13.

|  |  |
| --- | --- |
| User starts up/logs in to system. |  |
|  | System checks to see if it can load data from storage. If no data is found It continues normal operation. If data is found on storage, the user is asked if they would like to load that data |
| If data was found user answers yes or no |  |
|  | If yes skip to row 6. If no continue to the main menu. |
| User requests to retrieve data |  |
|  | System checks to see if any data accessing commands have been used. If so, the operation fails. If not, the program shows a success message. |
|  | If the operation fails, the user is told that it failed and is asked if they want to try again. |
| User answers yes or no |  |
|  | If yes, skip to row 6. |

14.

|  |  |
| --- | --- |
| User requests help from system. |  |
|  | System generates and outputs a list of all commands that the user can input as well as a short description of what each command accomplishes. |

Customers

CreditCards

(\*) -> (1)

Theater

Customers

(\*) -> (1)

Theater

Clients

(\*) -> (1)

Clients

Plays

(\*) -> (1)

**Clients**

Name

Address

Phone #

Client ID

Balance

**Customer**

Name

Address

Phone #

Customer ID

**CreditCard**

Card #

Expiration Date

Customer ID

**Theater**

Name

Address

Phone #

Capacity

**Plays**

Name

Dates

Client ID