**Testing**

|  |  |  |
| --- | --- | --- |
| **Requirement** | **Expected Outcome** | **Actual Outcome** |
| 1. GetLogin() | User can enter an email and password, then the system accepts and recognises whether they’re a first time customer or not. | The system asks user for email and password and accepts them, there isn’t much restriction to what they can enter as an email and password can be a combination of numbers and letters of any size. With storing return customers login in data I have managed to get it working as long as the application stays live, but of course when the console window is closed all stored data is lost. |
| 2. EnterProfile() | The expected outcome for the profile is for the system to be able to access the users purchase history and general details e.g. billing address… | The system is capable of using the login stored data to realise whether a customer needs to make a new profile or not. When a user does make a profile the system prompts for and accepts users input, but there isn’t much error handling with the amount of characters for each input. |
| 3. SelectShow() | The user selecting the show is needed because there could be multiple shows on the system at one time. | The outcome in this situation for my system is that a list of shows are displayed from the display show function. Then the user can input either 1, 2, 3, or 4 to select the show and if any other input is entered an error message will appear and then they’ll have to enter their choice again. |
| 4. DisplayShow() | Shows the customer what events are on in the venue from within a date range | The system outputs the shows to the user and prompts for an input. There is no input though for the user entering a date range, instead the shows dates run from current date to a couple months in the future. |
| 5. Initialise() | An initialise will be needed within this system for the seating floor plan fields to be displayed | Within the ShowSeat header file the system does initialise the floor plan by mapping out the data field with A’s for available. |
| 6. DisplaySeats | The system will need to be able to display the seating arrangement for the user to see. | In this function my system already has the field set up in initialise, so in this function I used two nested for loops to display headers on the field and spacing between the data, this is outputted from the main to the user. |
| 7. GetSeat() | The system needs to locate the customers selected seat so then it can reserve the seat | The outcome for this function is that the system asks for and accepts the users desired seat row and column number. This function then stores the users input and then uses it in order to replace an A for an H where the customer’s desired seat is. |
| 8. Deselect() | In the checkout process the system will need the user to make a final decision on paying for the ticket(s) or deselecting them | After the customer gets a total price for their tickets they will then get asked whether they would like to buy the tickets or not. If they reply yes, the customer will be forwarded to the pay function. But if they say no then the console will close. |
| 9. Pay() | The user will need to be able to enter their card details in order for the system to accept it and print the tickets | The system does prompt for and accepts the users card details. After it will send the user to a ticket screen. |
| 10. SetPrice() | The system will need to recognise how many tickets and what show the customer has selected in order to give them a set final price | The system doesn’t find out the quantity of tickets in this function, what it does instead is before the system asks for seat numbers (in the get seats function) it asks for the amount of tickets so they can select the seats for them as well. In this function the user gets asked what type of tickets they’re buying instead and performs a calculation to give the total price. |
| 11. PrintTicket() | At the end of the users transaction they will of course need the ticket, so the system needs to give them a ticket that can be printed off | My system allows the user to get to this function but unfortunately I didn’t have enough time personally to make an elaborate ticket. The method I would’ve used to obtain data for the ticket is the same as I used for the get seats function. |

**Screenshots**









