# Use Case Scenarios

Please note the following formatting styles:

* Candidate Object: Object
* Candidate Operation: operation

### Login:

John McClane selects loginfrom the Main Menu. The GUI Controller displays the Login View. John enters his Username and Password in the Login View. The Login View Controller registers John’s input and passes the data to the Database Controller. The Database Controller queries the Database for a Registered User matching the provided Username and Password. If the queryis unsuccessful, the Login View will prompt John to re-enter their Username and Password until a successful queryis returned or John returnsto the Main Menu. Since John has correctly entered his Username and Password, he is returnedto the Main Menu as a Registered User.

|  |  |  |
| --- | --- | --- |
| Noun | Filter | Class Type |
| John McClane | Filtered (User Actor) | N/A |
| Login | Filtered (Attribute of Main Menu) | N/A |
| Main Menu | Candidate Object | Boundary |
| GUI Controller | Candidate Object | Control |
| Login View | Candidate Object | Boundary |
| Login View Controller | Candidate Object | Control |
| Username | Filtered (Attribute of Registered User) | N/A |
| Password | Filtered – (Attribute of Registered User) | N/A |
| Database Controller | Candidate Object | Control |
| Database | Filtered (Database Actor) | N/A |
| Registered User | Candidate Object | Entity |

### Select Ticket:

John selects select ticket from the Main Menu. The GUI Controller displays the Movie View The Movie View displaysall Movie options, including both Publicly-Announced Movies and Early-Access Movies. John browses Movie options before selecting the Die Hard Movie.

The Theatre View displays the available Theatres that are showing Die Hard and waits for John to make a selection. John selects the Country Hills Theatre. Upon selection of the Theatre, the Showtime View displays Die Hard’s Showtimes and prompts John to make a selection. John selects Dec. 27th at 5 PM. Since this Showtime is for an Early-Access Movie, the Showtime View Controller checks whether or not 10% of the Seats have already been reserved by Registered Users. Since only 3% of the Seats have been reserved thus far, the Showtime View Controller accepts John’s Showtime selection. Once a Showtime is selected, the Seats View displays the Seats with a colour hue to indicate each Seat’s Availability Status. John selects Seat D7 from the remaining available Seats. With all the necessary Ticket information selected, a Ticket is added to John’s Cart. John is then prompted to the either return to the Main Menu or proceeds to the Payment View with his selected Ticket information (Movie, showtime, etc.) to complete his transaction. John elects to proceed to the Payment View, so the GUI Controller displays that view.

|  |  |  |
| --- | --- | --- |
| Noun | Filter | Class Type |
| John McClane | Filtered (User Actor) | N/A |
| Main Menu | Candidate Object | Boundary |
| GUI Controller | Candidate Object | Control |
| Movie View | Candidate Object | Boundary |
| Movie | Candidate Object | Entity |
| Publically-Announced Movie | Filtered (Attribute of Movie) | N/A |
| Early-Access Movie | Filtered (Attribute of Movie) | N/A |
| Theatre View | Candidate Object | Boundary |
| Theatre | Candidate Object | Entity |
| Showtime View | Candidate Object | Boundary |
| Showtime | Candidate Object | Entity |
| Showtime View Controller | Candidate Object | Control |
| Registered User | Candidate Object | Entity |
| Seat View | Candidate Object | Boundary |
| Seat | Candidate Object | Entity |
| Availability Status | Filtered (Attribute of Seat) | N/A |
| Payment View | Candidate Object | Boundary |
| Ticket | Candidate Object | Entity |
| Cart | Candidate Object | Entity |

### Checkout:

Since John is logged in as a Registered User, the Payment View Controller queries the Database Controller and retrieves John’s Payment Information. The Payment View Controller pre-populates the Payment View form with John’s Payment Information, including Card Type and Card Number. John enters his Pin Number and reviews his Payment Information before pressing submit. The Payment View Controller passes the Payment Information to the Financial Controller so the Financial Institution may verify the Payment Information.

With the Payment Information verified, the Database Controller records the Ticket and Receipt data to the Database before emailing John a copy of the Ticket and Ticket Receipt. John is flashed a success message before being returned to the Main Menu by the GUI Controller.

|  |  |  |
| --- | --- | --- |
| Noun | Filter | Class Type |
| John McClane | Filtered (User Actor) | N/A |
| Registered User | Candidate Object | Entity |
| Payment View Controller | Candidate Object | Control |
| Database Controller | Candidate Object | Boundary |
| Payment Information | Candidate Object | Entity |
| Payment View | Candidate Object | Boundary |
| Card Type | Filtered (Attribute of Payment Information) | N/A |
| Card Number | Filtered (Attribute of Payment Information) | N/A |
| Pin Number | Filtered (Attribute of Payment Information) | N/A |
| Financial Controller | Candidate Object | Boundary |
| Financial Institution | Filtered (Financial Institution Actor) | N/A |
| Ticket | Candidate Object | Entity |
| Ticket Receipt | Candidate Object | Entity |
| Database | Filtered (Database Actor) | N/A |
| Main Menu | Candidate Object | Boundary |
| GUI Controller | Candidate Object | Control |

### Refund Ticket:

John selects refund ticket from the Main Menu. The GUI Controller displays the Refund View. John enters his Receipt Number into the form and submits the data. The Refund View Controller passes the data to the Database Controller which queriesthe Database for a Ticket Receipt matching the Receipt Number entered by John. Since John has correctly entered an existing Receipt Number and the Ticket’s Showtime is more than 72 hours away, the previously reserved Seat is changed to an Available Status and John’s Credit Card is refunded the full price of the Ticket since he is logged in as a Registered User. An email of the Refund Receipt is sent to John’s email account. The Refund View Controller flashes a success message before returning John to the Main Menu.

If John had not been logged in as a Registered User, a Coupon worth 85% of the original Ticket Price would have been included in the Refund Receipt emailed to John.

|  |  |  |
| --- | --- | --- |
| Noun | Filter | Class Type |
| John McClane | Filtered (User Actor) | N/A |
| Main Menu | Candidate Object | Boundary |
| GUI Controller | Candidate Object | Control |
| Refund View | Candidate Object | Boundary |
| Receipt Number | Candidate Object | Entity |
| Database | Filtered (Database Actor) | N/A |
| Ticket Receipt | Candidate Object | Entity |
| Showtime | Candidate Object | Entity |
| Seat | Candidate Object | Entity |
| Available Status | Filtered (Attribute of Seat) | N/A |
| Credit Card | Filtered (Attribute of Payment Information) | N/A |
| Ticket | Candidate Object | Entity |
| Registered User | Candidate Object | Entity |
| Refund Receipt | Candidate Object | Entity |
| Financial Controller | Candidate Object | Boundary |
| Refund View Controller | Candidate Object | Control |
| Coupon | Candidate Object | Entity |

### Register Account

Sally wants to register as a Registered User. She selects register user from the Main Menu. The GUI Controller displays the Register View. Sally enters her information and presses the submit button. The Register View Controller verifies that all required information is provided before passing the information to the Database. The Database Controller confirms that the requested username is available before committing the new Register User information to the Database.

Sally is flashed a success message that reminds her to pay her Annual Dues before reserving any Early-Access Movies. Sally accepts the message and is returned to the Main Menu where pay annual dues is now available as an option to Sally.

|  |  |  |
| --- | --- | --- |
| Noun | Filter | Class Type |
| Sally | Filtered (User Actor) | N/A |
| Registered User | Candidate Object | Entity |
| Main Menu | Candidate Object | Boundary |
| GUI Controller | Candidate Object | Control |
| Register View | Candidate Object | Boundary |
| Register View Controller | Candidate Object | Control |
| Database Controller | Candidate Object | Boundary |
| Database | Filtered (Database Actor) | N/A |
| Annual Dues | Filtered (Attribute of Registered User) | N/A |

### Pay Annual

Sally selects pay annual from the Main Menu. Since Sally is logged in as a Registered User, the Payment View Controller queries the Database Controller and retrieves Sally’s Payment Information The Payment View Controller pre-populates the Payment View form with Sally’s Payment Information, including Card Type and Card Number. Sally enters his Pin Number and reviews his Payment Information before pressing submit. The Payment View Controller passes the Payment Information to the Financial Controller so the Financial Institution may verify the Payment Information.

With the Payment Information verified, the Database Controller records the Annual Due payment and Receipt data to the Database before emailing Sally a copy of the Annual Dues Receipt. Sally is flashed a success message before being returned to the Main Menu by the GUI Controller.

|  |  |  |
| --- | --- | --- |
| Noun | Filter | Class Type |
| Sally | Filtered (User Actor) | N/A |
| Registered User | Candidate Object | Entity |
| Payment View Controller | Candidate Object | Control |
| Database Controller | Candidate Object | Boundary |
| Payment Information | Candidate Object | Entity |
| Payment View | Candidate Object | Boundary |
| Card Type | Filtered (Attribute of Payment Information) | N/A |
| Card Number | Filtered (Attribute of Payment Information) | N/A |
| Pin Number | Filtered (Attribute of Payment Information) | N/A |
| Financial Controller | Candidate Object | Boundary |
| Financial Institution | Filtered (Financial Institution Actor) | N/A |
| Annual Dues Receipt | Candidate Object | Entity |

### Issue Movie News

Bill, a manager at Country Hills Theatre, is logged in as a Manager and selects issue movie news from the Main Menu. The GUI Controller displays the Issue News View. Bill selects the upload button and adds the new Movie News file he intends to publish. The Issue News View Controller paints the Issue News View with the uploaded Movie News text so Bill may review before issuing. Bill selects the submit button and the Issue News View Controller passes the new Movie News to the Boss Controller. The Boss Controller queries the Database Controller for the Emails of all Registered Users before emailing the new Movie News to all Registered Users.

Bill is flashed a success message before being returned to the Main Menu.

|  |  |  |
| --- | --- | --- |
| Noun | Filter | Class Type |
| Bill | Filtered (User Actor) | N/A |
| Manager | Candidate Object | Entity |
| Theatre | Candidate Object | Entity |
| Main Menu | Candidate Object | Boundary |
| GUI Controller | Candidate Object | Controller |
| Issue News View | Candidate Object | Boundary |
| Movie News | Candidate Object | Entity |
| Issue News View Controller | Candidate Object | Controller |
| Boss Controller | Candidate Object | Controller |
| Database Controller | Candidate Object | Boundary |
| Emails | Filtered (Attribute of Registered User) | N/A |
| Registered User | Candidate Object | Entity |