Will Alonzo, Brian Antalek, Robert Kilmer, Brandon Williams

CSC 380 Link: https://github.com/VerbatumAura/CSC380

5/06/2018

Movie Theater Software

**Project Description**

1. Project Description in form of User Narrative :

* Filling the need for a sleek, new interface for viewing and reserving seats/times of the latest hit movies.
* The layout will have to be clear and concise, showing only the most relevant information to aid in a user
* quickly finding what they want and choosing a variety of options, including seating, times, theaters, and the ability to
* create an account within the system for quick checkout.

2. List of requirements :

* Login UI, Main Screen UI, Movie Descriptions and Cover art panel, Seat Reserving UI, Checkout/Guest process and payment entering.
* Following a Kanban style of development, design and implementation will be relatively straight forward, focusing on each
* element of the software 1 by 1.
* Login UI : Simple Design of Login/Password entering, checking to a user saved within the system. Guest login will be available to choose. Create Account function will also be available.
* Main Screen UI : 4 Panels of the current top movie cover arts, with mouse-over functionality showcasing the movie cover art and a short description of the movie.
* News panel showing the current movie news, such as release dates, newly released info on a movie, and more.
* Quick-link panel of movie titles, times, and theater numbers for the next 48-72 hours, allowing a user to quickly click a showtime in which they want to see a specific movie. This will also display the current time, dynamically adding/removing movies and showtimes that are no longer able to sell tickets.
* Movie Descriptions and Cover Art Panel : 4 Equally sized, horizontal structured panels showcasing the cover art of current top movies. On mouseover,
  + a small panel will appear alongside the cursor showing the art and a short description of the movie. This will allow users to look through movies they may
  + not be initially interested in to get an idea if they would like the movie or not. A button on this panel will also allow a user to immediately show times and
  + locations for this movie.
* Seat Reserving UI : Upon selecting a movie and time, a mockup of the theater seating arrangement will allow users to select which seats they would like to reserve,
* albeit those seats are not already taken. A count of seats x ticket prices will show the user before submitting their seat requests how much they are paying in tickets.
* Checkout/Guest Process : Checkout will be a simple UI showing the movie, time, theater, how many seats are being reserved and their price. If the user has logged into a
  + account from the account screen, they will have the option of using a previously-entered credit card or entering a new one for future purchases. If the user selected guest
  + at login, they will be prompted to make an account to take use of various features, or to continue. Guest checkout will not allow pick up at theatre option,
  + nor will their credit card be saved after entering.
* Payment entering : Simple UI with fields for name, address, credit card number with expiration date and security code entrance. If user is logged into to an account,
  + they can choose for their credit card to be saved for future payments. Pickup at theater will also be an available option, where payment will be needed.

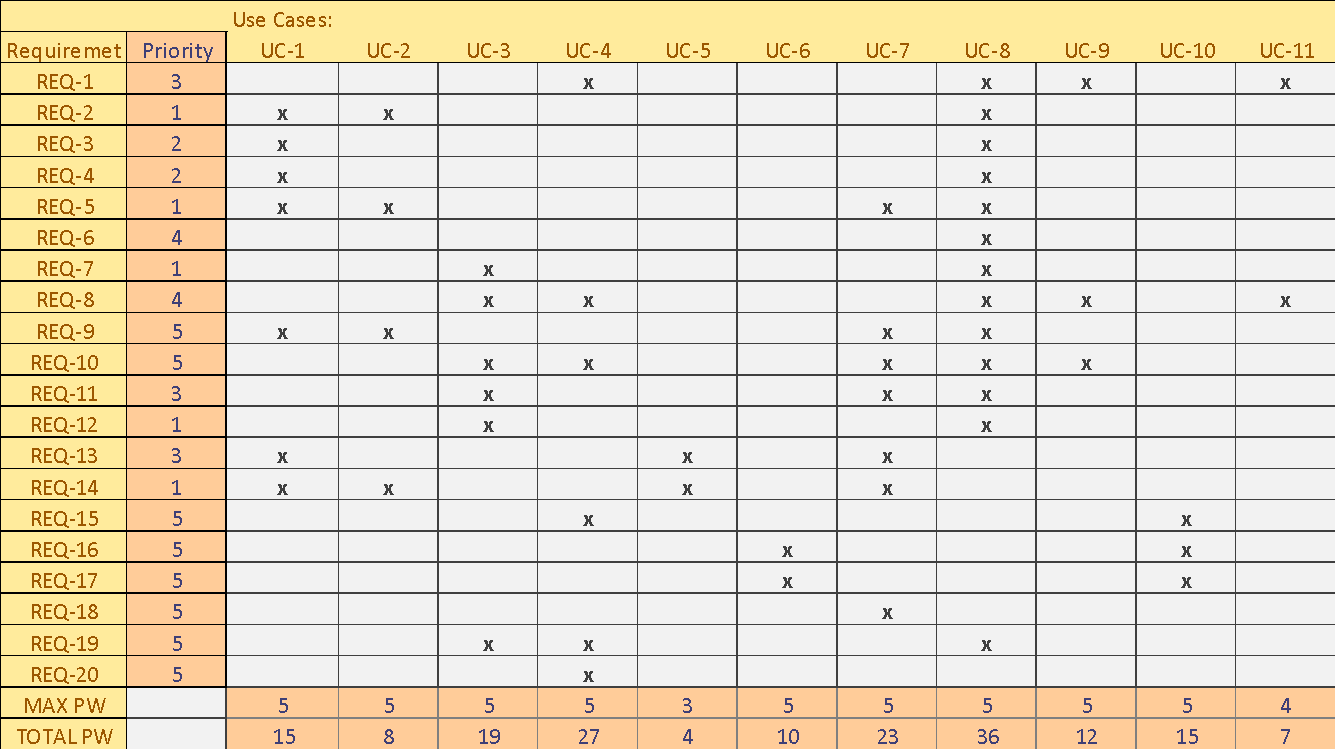
**System Requirements** *Figure 1*.

|  |  |  |
| --- | --- | --- |
| **Requirements** | **Priority** | **Description** |
| req-1 | 3 | Store and log user information. |
| req-2 | 1 | Allow users to select movies according to date and time. |
| req-3 | 2 | Allow users to select from available seats. |
| req-4 | 2 | Interface shows available/unavailable seats |
| req-5 | 1 | System blocks out sold out theaters ,seats, movies before current time, and sold out seats. |
| req-6 | 4 | Quick list of movies and times allowing for faster purchases. |
| req-7 | 1 | Checkout and refund functions. |
| req-8 | 4 | Option to create a user profile to store information and make check-out easier. |
| req-9 | 5 | Clock with current time and dates |
| req-10 | 5 | Newsletter with discount options (accounts with a verified email will receive coupon codes for discounted prices) |
| req-11 | 3 | Discount tickets, e.g. senior, student, matinee, etc |
| req-12 | 1 | Randomly generated confirmation number stored which allow a user to refund a purchase |
| req-13 | 3 | Admin access with ability to change showings, pricing, and assign movies to theaters |
| req-14 | 15 | Make sure there is no overlap between showings of movies in a theater |
| req-15 | 5 | Account preferences for user profile, including username, profile picture, and basic information |
| req-16 | 5 | Moderated forums for movie discussion with our users |
| req-17 | 5 | Movie review system that stores movie reviews and produces overall user rating |
| req-18 | 5 | Allow space for advertisements |
| req-19 | 5 | Encrypt USER CC number and account information |
| req-20 | 5 | Email confirmations |

**User Stories** *Figure 2*.

|  |  |
| --- | --- |
| **Identifier** | **User Story** |
| st-1 | As Theater management, I can assign movies to specific theaters and times as well as list available seats for sale |
| st-2 | As a movie goer, I can pick a movie and a time/date |
| st-3 | As a user, I can decide whether to create a members account or not |
| st-4 | As Theater management, I can edit showtimes or cancel showings |
| st-5 | As a user, I can read news/reviews on movies that are either trending or that I am interested in seeing |
| st-6 | As management, we can adjust movie listings and prices depending on popularity and matinee showtimes |
| st-7 | As a consumer, if I can’t make it to the movie, I can opt in for a refund or exchange for a later showing |
| st-8 | As management, I can decide when to offer special rewards and discounts for members by sending out newsletters |
| st-9 | As a user, I can sign up for newsletters offering exclusive showtimes and discounts |
| st-10 | As a movie goer, I can leave movie and/or theater reviews in regards to my thoughts on what I thought about the movie and if the theater needs attention in regards to repairs or cleanliness. |
| st-11 | As a consumer, I can purchase gift cards without having to browse through movie selections |

**Traceability Matrix Table** *Figure 3*.



***Detailed Use Cases: Figure 4 – Figure 14***

**Use Case 1** *Figure 4*

|  |  |
| --- | --- |
| USE CASE UC - #1 | As Theater management, I can assign movies to specific theaters and times as well as list available seats for sale |
| Related REQ’s | REQ-2 REQ-3 REQ-4 REQ-5 REQ-9 REQ-13 REQ-14 |
| Initiating Actor | User with Admin privileges, sets master schedule for the week. |
| Actor’s Goal | Set showtimes for the consumers, and allow for transactions to start being made. |
| Participating Actors | Movie Theater Management |
| Preconditions | System is able to freely change variables and store info in memory. |
| Postconditions | Movie page is dynamic and able to change after every scheduling. |
| Flow of Events for Main Scenario |  |
| → 1. | Actor changes frames, and panels to differentiate movies and showtimes. |
| ← 2. | System implements changes, allowing consumers the ability to start booking showtimes. |
| Flow of Events for Extensions (Alternate Scenarios) |  |
| → 1. | Invalid Data could crash the system or leave an entire theater unable  to compile. |
| ← 2. | User could input bad data the system didn’t defensively program against and cause errors in data handling. |

**Use Case 2** *Figure - 5*

|  |  |
| --- | --- |
| USE CASE UC- #2 | As a movie goer, I can pick a movie and a time/date |
| Related REQ’s | REQ-2 REQ-5 REQ-9 REQ-14 |
| Initiating Actor | Customer uses system to pick a movie. |
| Actor’s Goal | Complete a transaction and purchase a ticket to the desired movie |
| Participating Actors | Customer and the system |
| Preconditions | Data storage and retrieval, working memory of what is available. |
| Postconditions | Data storage and retrieval updated memory of what is available. |
| Flow of Events for Main Scenario |  |
| → 1. | Customer selects a movie. |
| ← 2. | System shows what is available. |
| → 3. | User selects from available seats. |
| Flow of Events for Extensions (Alternate Scenarios) |  |
| → 1a | Invalid Data could crash the system or leave an entire theater unable  to compile. |
| ← 2a. | User could input bad data the system didn’t defensively program against and cause errors in data handling. |

**Use Case 3** *Figure - 6*

|  |  |
| --- | --- |
| USE CASE UC- #3 | As a consumer, I can select what kind of payment option I wish to use |
| Related REQ’s | REQ-7 REQ-8 REQ-10 REQ-11 REQ-12 REQ-19 |
| Initiating Actor | Customer |
| Actor’s Goal | Use credit card to confirm purchase/reservation of seats |
| Participating Actors | Customer and system |
| Preconditions | System can verify the user inputs a valid credit card number, thus enabling the seat(s) to be available for purchase. |
| Postconditions | Seats selected for purchase become unavailable to other customers to purchase. |
| Flow of Events for Main Scenario |  |
| → 1. | Customer inputs credit card number |
| ← 2. | System verifies credit card number, and allows seats to be selected. |
| → 3. | User selects seats to buy. |
| ← 4. | System locks those seats to future consumers. |
| Flow of Events for Extensions (Alternate Scenarios) |  |
| → 1a. | User puts in an invalid card number, |
| ← 2a. | System blocks purchase. |
| → 1b. | User selects a seat already purchased. |
| ← 2a. | System ignores request until a valid seat is chosen, or canceled. |

**Use Case 4** *Figure – 7*

|  |  |
| --- | --- |
| USE CASE UC- #4 | As a user, I can decide whether to create a members account or not |
| Related REQ’s | REQ-1 REQ-8 REQ-10 REQ-15 REQ-19 REQ-20 |
| Initiating Actor | Customer |
| Actor’s Goal | Create a unique profile that can come with perks in the future. |
| Participating Actors | Customer, System |
| Preconditions | System has the ability to create and store customer information. |
| Postconditions | Information is recallable for future logins. |
| Flow of Events for Main Scenario |  |
| → 1. | Customer inputs valid information to create a profile. that the system writes to a file for future referencing. |
| ← 2. | The system writes to a file for future referencing. |
| Flow of Events for Extensions (Alternate Scenarios) |  |
| → 1a. | User enters invalid information. |
| ← 2a. | System blocks entry, and leads consumer to location the problem resides for error handling. |

**Use Case 5** *Figure - 8*

|  |  |
| --- | --- |
| USE CASE UC- #5 | As Theater management, I can edit showtimes or cancel showings |
| Related REQ’s | REQ-13 REQ-14 |
| Initiating Actor | Management |
| Actor’s Goal | Maintain catalogue of movies and see available showtimes. |
| Participating Actors | Management/Admins and System |
| Preconditions | System can handle various changes to catalogue. |
| Postconditions | Changes are implemented. |
| Flow of Events for Main Scenario |  |
| → 1. | Management cancels or changes movie. |
| ← 2. | System recognizes change, and refunds purchases, prevents future purchase, etc. |
| Flow of Events for Extensions (Alternate Scenarios) |  |
| → 1a. | Management causes entire theater room to be invisible to consumer. |
| ← 2a. | Users can accidently purchase tickets to corrupt areas, causing poor customer satisfaction. |

**Use Case 6** *Figure - 9*

|  |  |
| --- | --- |
| USE CASE UC- #6 | As a user, I can read news/reviews on movies that are either trending or that I am interested in seeing |
| Related REQ’s | REQ-16 REQ-17 |
| Initiating Actor | Customer |
| Actor’s Goal | Gather information based on feedback of other users. |
| Participating Actors | Customer, System and Management. |
| Preconditions | System can support a forum with countless users adding information |
| Postconditions | System can accurately sort through information. |
| Flow of Events for Main Scenario |  |
| → 1. | Customer writes a review/reads a review. |
| ← 2. | System recognizes, and posts to the thread. |
| Flow of Events for Extensions (Alternate Scenarios) |  |
| → 1a. | User writes a review or posts something the application cannot support. |
| ← 2a. | System becomes unusable. |

**Use Case 7** *Figure - 10*

|  |  |
| --- | --- |
| USE CASE UC- #7 | As management, we can adjust movie listings and prices depending on popularity and matinee showtimes |
| Related REQ’s | REQ-5 REQ-9 REQ-10 REQ-11 REQ-13 REQ-14 REQ-18 |
| Initiating Actor | Management. |
| Actor’s Goal | Make changes to support the firms marketing efforts and push sales. |
| Participating Actors | Management, System and Consumer. |
| Preconditions | System can react to a change in pricing even after purchases have been made. |
| Postconditions | System lists tickets for new updates price. |
| Flow of Events for Main Scenario |  |
| → 1. | Management adjusts ticket price from $10 to $7. |
| ← 2. | System lists and sells respective ticket for $7. |
| Flow of Events for Extensions (Alternate Scenarios) |  |
| ← 1a. | System doesn’t acknowledge the change. |

**Use Case 8** *Figure - 11*

|  |  |
| --- | --- |
| USE CASE UC- #8 | As a consumer, if I can’t make it to the movie, I can opt in for a refund or exchange for a later showing |
| Related REQ’s | REQ-1 REQ-2 REQ-3 REQ-4 REQ-5 REQ-6 REQ-7 REQ-8 REQ-9 REQ-10 REQ-11 REQ-12 REQ-19 |
| Initiating Actor | Consumer |
| Actor’s Goal | Refund ticket |
| Participating Actors | Consumer and System. |
| Preconditions | System recognizes specific consumer made a purchase. |
| Postconditions | System locates purchase and refunds Customer the amount they purchased. |
| Flow of Events for Main Scenario |  |
| → 1. | Customer requests refund from System. |
| ← 2. | System locates purchase and refunds customer if criteria is met. |
| Flow of Events for Extensions (Alternate Scenarios) |  |
| ← 1. | System cannot execute. |

**Use Case 9** *Figure - 12*

|  |  |
| --- | --- |
| USE CASE UC- #9 | As a user, I can sign up for newsletters offering exclusive showtimes and discounts |
| Related REQ’s | REQ-1 REQ-8 REQ-10 |
| Initiating Actor | Management |
| Actor’s Goal | Send marketing material to customers with user accounts. |
| Participating Actors | Management, System. |
| Preconditions | Log of Customer email information |
| Postconditions | Way for Management to access user Emails to send out promotional material. |
| Flow of Events for Main Scenario |  |
| → 1. | Management runs System and asks for User Info. |
| ← 2. | System returns Info. |
| Flow of Events for Extensions (Alternate Scenarios) |  |
| ← 1. | System cannot retrieve data. |

**Use Case 10** *Figure - 13*

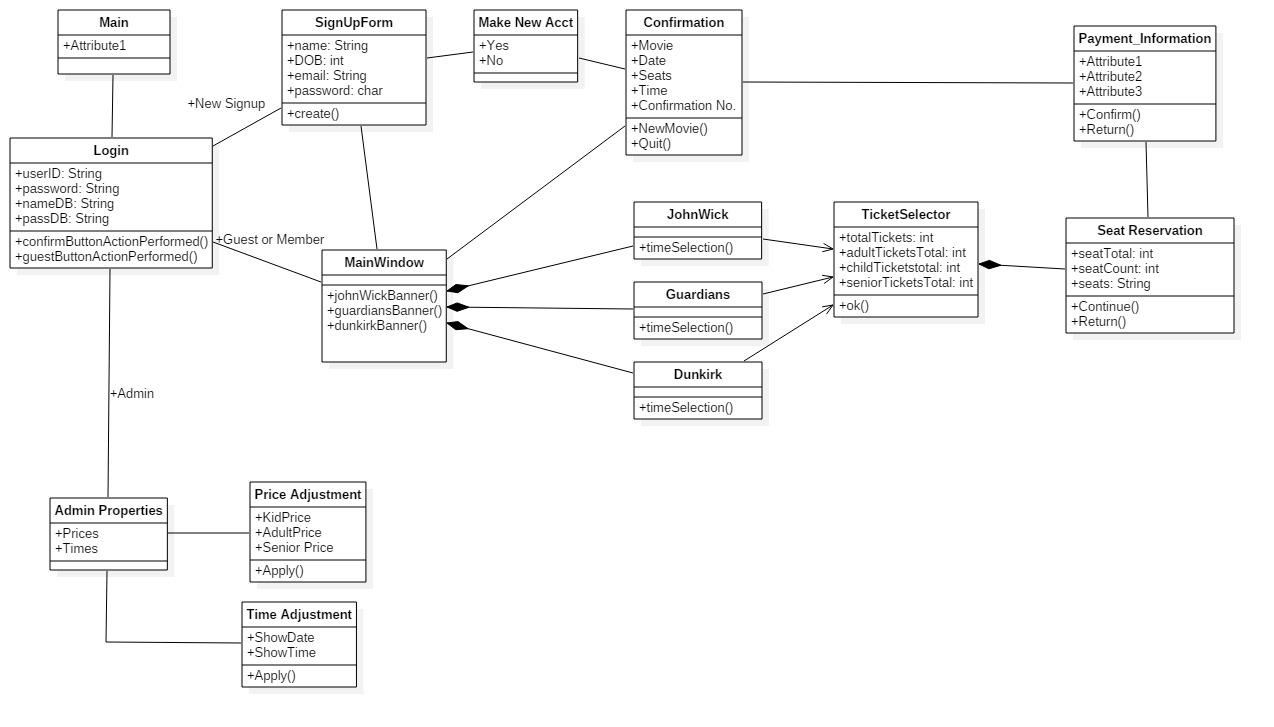
|  |  |
| --- | --- |
| USE CASE UC- #10 | As a movie goer, I can leave movie and/or theater reviews in regards to my thoughts on what I thought about the movie and if the theater needs attention in regards to repairs or cleanliness. |
| Related REQ’s | REQ-15 REQ-16 REQ-17 |
| Initiating Actor | Customer |
| Actor’s Goal | Leave Reviews |
| Participating Actors | Customer and System |
| Preconditions | System can handle forum processes. |
| Postconditions | Users can update information |
| Flow of Events for Main Scenario |  |
| → 1. | User creates post. |
| ← 2. | System reads post, and adds it to the thread. |
| Flow of Events for Extensions (Alternate Scenarios) |  |
| ← 1. | System cannot retrieve data. |

**Use Case 11** *Figure - 14*

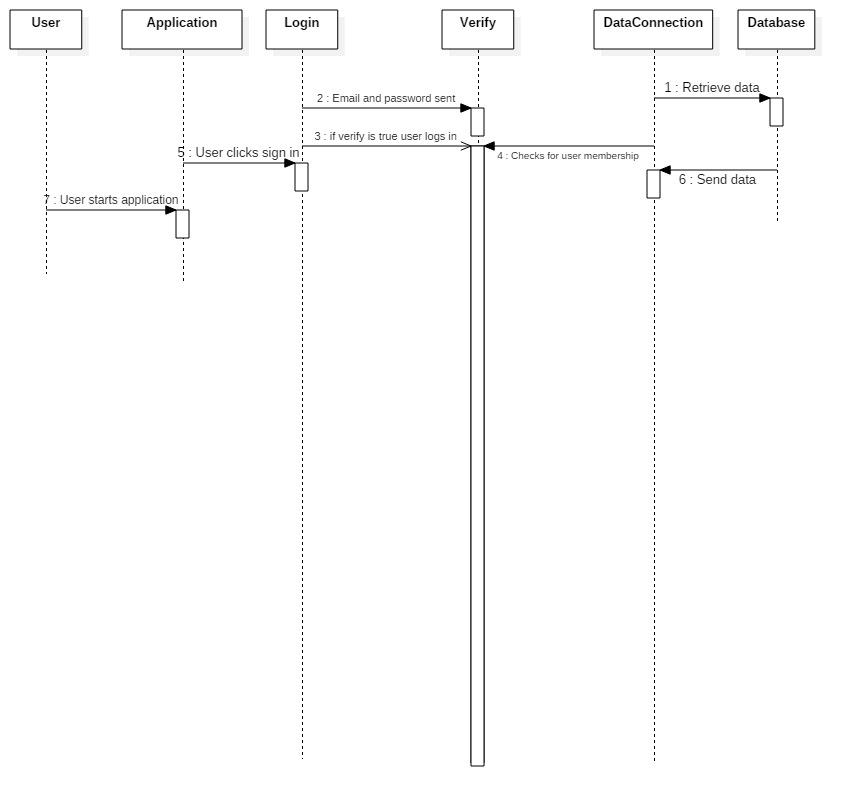
|  |  |
| --- | --- |
| USE CASE UC- #11 | As a consumer, I can purchase gift cards without having to browse through movie selections |
| Related REQ’s | REQ-1 REQ-8 |
| Initiating Actor | Consumer |
| Actor’s Goal | Purchase credit |
| Participating Actors | System, Customer |
| Preconditions | System can log credit purchases. |
| Postconditions | Credit purchases can be matched to an individual |
| Flow of Events for Main Scenario |  |
| → 1. | User requests to purchase a gift card. |
| ← 2. | System authenticates credit card data and uses and logs gift card credit to the users profile for future reference. |
| Flow of Events for Extensions (Alternate Scenarios) |  |
| ← 1. | System cannot retrieve data. |

**UML Diagrams**

**Class Diagram** *Figure-15*



**Sequence Diagram** *Figure-16*



**Sequence Diagram** *Figure-17*

