Alfred Ababio

Table of Contents

Problem Statement [Page 2](file:///C:\Users\Alfred\Documents\Fall%202013%20Semester\CS3500%20-%20Object-Oriented%20Design\Workspace\Assignment%209\Movie%20Ticketing%20System.docx#Problem_Statement)

User Cases [Page 2](file:///C:\Users\Alfred\Documents\Fall%202013%20Semester\CS3500%20-%20Object-Oriented%20Design\Workspace\Assignment%209\Movie%20Ticketing%20System.docx#Use_Case)

Requirements [Page 2](file:///C:\Users\Alfred\Documents\Fall%202013%20Semester\CS3500%20-%20Object-Oriented%20Design\Workspace\Assignment%209\Movie%20Ticketing%20System.docx#Requirement)

Requirements (cont.) [Page 3](file:///C:\Users\Alfred\Documents\Fall%202013%20Semester\CS3500%20-%20Object-Oriented%20Design\Workspace\Assignment%209\Movie%20Ticketing%20System.docx#Requirement_Functional)

Data Model [Page 3](file:///C:\Users\Alfred\Documents\Fall%202013%20Semester\CS3500%20-%20Object-Oriented%20Design\Workspace\Assignment%209\Movie%20Ticketing%20System.docx#Data_model)

Model Dependency Diagram [Page 3](file:///C:\Users\Alfred\Documents\Fall%202013%20Semester\CS3500%20-%20Object-Oriented%20Design\Workspace\Assignment%209\Movie%20Ticketing%20System.docx#MDD)

UML diagram [Page 4](file:///C:\Users\Alfred\Documents\Fall%202013%20Semester\CS3500%20-%20Object-Oriented%20Design\Workspace\Assignment%209\Movie%20Ticketing%20System.docx#UML)

Movie Ticketing System

**Problem Statement:**

Your goal is to design a Java application that will manage the sales of movie tickets in a cinema complex. The complex consists of several theaters that show movies at different times. The same movie may be showing in more than one theater.

~~The ticket prices may be different for children, adults, and seniors. There may also be a different price for the matinees.~~

The ticket prices may be different for any type and price of ticket.

~~The cinema complex wants to be able to analyze its sales by various criteria – tickets sold to seniors, tickets for a specific movie, matinee ticket sales, etc.~~

The cinema complex must handle sales by type of ticket sold for each Show

~~Some cinema complexes may also have luxury theaters with higher fixed price for all patrons.~~

**Use Case 1: Buying a ticket**

1. ~~User selects movie, theater, time and ticket type (in that order)~~
2. ~~System looks for a movie that represents the appropriate selection of the user~~
3. ~~System checks if the movie has seats left~~
4. ~~System pushes the movie to GUI~~
5. ~~GUI displays the movie choice and prompts the user for confirmation for ticket sale~~
6. Console will display main menu
7. User will select 1 for “Buy A Ticket”
8. Console will list Showings
9. Console will prompt user to select a Showing
10. Console will prompt user for each Ticket
11. System will check seats left against the number of tickets requested
12. System will calculate the total price for the order
13. Console will print the order log
14. System will save the order log to its own log location and the Showing’s log.
15. Console will then display the main menu

**User Case 2: Buying a sold out ticket**

1. ~~User selects a movie, theater, time and ticket type (in that order)~~
2. ~~System looks for a movie that represents the appropriate selection of the user~~
3. ~~System checks if the movie has seats left, but doesn’t.~~
4. ~~GUI displays the “Sold Out” message\~~
5. Console will display the Main Menu
6. User will select 1 for “Buy A Ticket”
7. Console will list Showings
8. Console will prompt user to select a Showing
9. Console will prompt user for each Ticket
10. System will check seats left against the number of tickets requested
11. Console will output a message for the user to see that tickets are sold out
12. System will save the order log with a total price of 0 to it’s own log
13. Console will then display the main menu

**~~User Case 3: User can search by movie criteria~~**

1. ~~User selects any of the selection criteria while leaving others blank (for ex. Just Movie Title)~~
2. ~~System looks up the criteria it was given~~
3. ~~System checks seats left in all of the movies that match the criteria~~
4. ~~GUI then shows the user all the choices that it came up with~~

**User Case 3: User can initialize data**

1. Console displays Main Menu
2. User selects 4 for “Initialize Data”
3. Console lists the types of data that can be initialized
4. User selects the type of data they wish to initialize
5. User sets the fields in the constructors for their data
6. System adds new data to its database
7. Console displays the main menu

**~~User Case 4: System Manager requests ticket sales~~**

1. ~~User requests certain ticket sales from System~~
2. ~~System sends the logging data to the GUI~~
3. ~~GUI displays the logging data~~

**User Case 4: User can request Sales Report**

1. Console displays Main Menu
2. User selects 2 for “Request Sales Report”
3. System will send Console the sales report
4. Console will print the sales report
5. Console will then display the main menu

**User Case 5: User can request Manager Reports**

1. Console displays Main Menu
2. User selects 3 for “Request Manager Report”
3. System will generate a new Manager Report
4. System will add the new Report to the existing reports
5. Console will print all the reports
6. Console will then display the main menu

**Requirements**

**Non-functional:**

~~The System will keep track of movie listings.~~

The System will keep track of Movies

The System will keep track of Theaters

The System will keep track of Shows

The System will keep track of prices of Tickets

~~The System will also do the search according to the input and push its search results to the Console~~

~~The System will keep track of movie tickets sold.~~

The System will keep track of total sales

The System will keep track of # of tickets sold per movie

~~The GUI will handle displaying formatted sales logs.~~

The Console will handle displaying formatted sales logs.

**Functional:**

~~The GUI shall allow the user to input the movie, theater, time, ticket type, or ticket quantity, or any combination of that data.~~

The Console will allow the user to select a showing and buy tickets for that showing

~~The GUI will display “sold out” messages.~~

The Console will allow user to initialize Movie, Theater, Showing, and Ticket objects.

~~The System Manager can add and remove movies from the movie listing for the theater.~~

~~The System Manager can request logs of sales using the GUI by any combination of the input data.~~

The Console can print sales logs

The Console can print manager reports

**Data model:**

Movie name: The name of the movie

~~Movie hour: The hour of the movie~~

~~Movie minute: The minute of the movie~~

~~Movie theater: The theater that the movie is playing at~~

Movie length: The length of the movie

Theater name: The name of the theater

~~Theater movies: The list of movies playing at the theater~~

Theater seatsMax: The maximum capacity of the theater

Theater seatsLeft: The amount of seats unfilled for the theater

Showing movies: The Movie playing in the Showing

Showing theaters: The Theater playing in the Showing

Showing start: The start time of the Showing

System theaters: The list of theaters in the movie complex

System movies: The list of movies in the movie complex

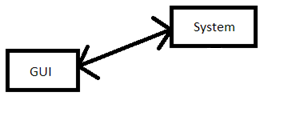
System shows: The list of showings for the movie complex

System tickets: The list of tickets for the movie complex

~~Every movie has a theater and every theater has a list of movies.~~

~~… and additional classes as needed; classes for user interactions, managers functions, reporting functions.~~

**Module dependency diagram:**



**UML diagram:**

