

KIOSK-BASED MOVIE RENTAL SYSTEM

Version 1.0

This document contains a general description of the requirements for the project. Please note the version number since we are likely to revise it after student feedback. The deliverables for each phase are mentioned at the end of document.

1. PROJECT DESCRIPTION

You will develop a simple kiosk-based movie rental system for a video store called Video-Mart. The movie rental system allows customers to choose a movie to rent by going to a kiosk in the store. Here is how the system works. A customer walks up to the kiosk in the Video-mart store to select one or more movies based on titles, actors, genres, awards, etc. We describe various screens below that help the customers make his/her choice. We will also provide the definition of the user interface screens for other categories of users. They include store managers who control the video inventory and staff personnel in the store who perform entry of new movies into the store inventory and check-in/check-out of movies. You will design the database that supports the functionality of the Video-Mart rental kiosks and the software itself (if you choose to do the heavyweight option). The specifications of the system tasks are described in the next section.

2. SYSTEM FUNCTIONALITY

TASK T1. Login

Users: All users

Description:

The users of the system are categorized into company employees and movie customers. There are two types of employee: staff and managers. The positions are mutually exclusive. There is more than one manager in the company. Every user in the system must have a unique username.

Reference: Illustration 1.



Username

Password

Illustration 1

TASK T2. Task List for Company Staff

Users: Company staff

Description:

This GUI shows a list of tasks for company staff. A company staff member can perform the following actions: add a new movie, update movie information, view a list of movies/detailed information of a movie, check-in return movies, and check-out rented movies.

Reference: Illustration 2

[Add Movie](#)
[Update Movie Information](#)
[List of Movies/Movie Detail](#)
[Check-in](#)
[Check-out](#)

Illustration 2

TASK T3. Add Movie

Users: Company staff

Description:

This GUI is used to enter a new movie into the system. A movie entry must include a title, a year, a director, at least one genre, and at least one actor. It should be noted that an actor in one movie can be a director in another movie.

After the user clicks on the “Add Movie” button, the movie entry will be added to the system and the detailed information of the movie will be shown in the “Recently Added Movie” section. When the user clicks on the update link, the system will open a new page to allow the user to update the information about the movie.

Reference: Illustration 3.

Title 2009 ▾
 Genre Actor Director
 Action Adventure Comedy Drama Fantasy Horror Sci-Fic
 Tom Hanks Gary Sinise Sally Field Ian McKellen Reese Witherspoon Lindsay Lohan
 Tom Hanks Gary Sinise Sally Field Ian McKellen Reese Witherspoon Lindsay Lohan
 Add Movie

Recently Added Movie
Title Forrest Gump
Year 1994
Director Robert Zemeckis
Actors
 Gary Sinise
 Tom Hanks
[Update](#)

Illustration 3

TASK T4. Update Movie Information

Users: Company staff

Description:

Additional information about a movie will be entered on this page. The staff users can add/remove genres, actors, and movie awards from this screen. Here are the business rules:

Copies:

- A movie can have many copies. When a new copy is added, a unique ID will be automatically generated for it. Receiving new copies of a movie will increase the number of copies and discarding a damaged copy will decrease the number of copies.

Movie and Actor Awards:

- The database is initially populated with five different types of awards: Golden Globe, MTV Movie Award, Oscar, NBR, and Golden Raspberry. The database should be designed in a way such that additional awards could be added at a later time (although you will not have to implement the GUI to do this—assume they would be added manually by a database administrator.)
- Every award can be divided into two groups: movie awards and actor awards.

- A movie award will be associated with a movie and a year. There are 10 categories of movie award: best art direction, best cinematography, best editing, best director, best picture, best sound effects, best visual effects, best screenplay, worst director, and worst screenplay.
- An actor award will be associated with a movie, an actor in the movie, and a year. There are 8 categories of actor awards: best actor, best supporting actor, best actress, best supporting actress, worst actor, worst supporting actor, worst actress, worst supporting actress.
- The drop-down list of actors for the actor award section should ONLY show the list of all actors of the current selected movie regardless of whether or not they received an award. An actor cannot receive more than one award for the same movie.
- For a given year and a given award category of an award type (e.g., Oscar for Best Picture) only one award is given (i.e., only one Oscar for Best Picture would be given in 2009 across all movies).
- The year of an award can be different from the year a movie is released.

In the following illustration for the movie “Forrest Gump”, under Actors, there is a drop down list called “Actor” which will bring up all current actors from the database. Add and Remove buttons to the right are used to add and remove actors to this movie. Similar reasoning applies to the displays called “Genre”, “Movie awards” and “Actor/Actress Awards,” all of which must accept multiple entries.

Reference: Illustration 4.

Forrest Gump

Copies

Copy ▾	Add
111	Remove

Actors

Actor ▾	Add
Gary Sinise	Remove

Genres

Genre ▾	Add
Drama	Remove

Movie Awards

Award ▾	Award Category ▾	Year ▾	Add
Oscar	Best Visual Effects	1995	Remove
Oscar	Best Picture	1995	Remove

Actor/Actress Awards

Award ▾	Actor Award Category ▾	Year ▾	Recipient ▾	Add
NBR	Best Supporting Actor	1994	Gary Sinise	Remove

Illustration 4

TASK T5. List of Movies and Movie Detail

Users: Company staff

Description:

This GUI allows the staff users to browse the list of movies in the database.

List of Movies:

- The movie list contains the movie’s title, year, director, total number of awards (including movie and actor awards), total number of copies in the inventory, and the number of copies available for rent.
- Clicking on a movie title shows the detailed information of the movie. This might appear in a section of the screen above the movie list or in a popup display.
- Clicking on *update* at the end of a row navigates to the update page for the corresponding movie (see Task 4).
- Movies are ordered by title in ascending order.

Movie Detail:

- Actors should be ordered by name
- Awards should be ordered by alphabetically and the year should be displayed. For each award, show the movie awards, then the actor awards.

Implementation Notes:

- Only one SQL statement (on views and tables) should be used to render the list of movies.
- Only one SQL statement (on views and tables) should be used to render the list of awards for a given movie.

Reference: Illustration 5.

Movie Detail

Title Forrest Gump
Year 1994
Director Robert Zemeckis
Actors
 Gary Sinise
 Tom Hanks

Awards
Oscar 1995
Best Visual Effect
Best Picture
Best Actor Tom Hanks
NBR 1994
Best Supporting Actor Gary Sinise

List of Movies

Title	Year	Director	#Awards	#Copies	Availability	Action
Forrest Gump	1994	Robert Zemeckis	5	10	3	Update
The Lord of the Rings	2002	Peter Jackson	4	10	6	Update

Illustration 5

TASK T6. Check-Out

Users: Company staff

Description:

This task shows the list of movies currently requested by customers. When a customer rents a movie the status of the movie will be changed to “In Process”. The availability of the movie will be decreased by one. The status of the movie will change to “Delivered” after a staff member clicks on the “Check Out” button of the pending movie.

Reference: Illustration 6.

Movies Pending for Check-out

Item ID	Customer	Address	Title	Action
114	John Smith	115 Rockberry, Atlanta, GA	The Lord of the Rings	<input type="button" value="Check Out"/>
115	John Smith	115 Rockberry, Atlanta, GA	Forrest Gump	<input type="button" value="Check Out"/>
221	Michael Jordan	215 Ferst Drive, Atlanta, GA	Tommorrow Never Dies	<input type="button" value="Check Out"/>

Illustration 6

TASK T7. Check-In

Users: Company staff

Description: When a movie is checked in, the number of available copies of that movie will be increased by one.

Reference: Illustration 7.

Movies Pending for Check-in for

Title	Action
X-men: Wolverin	<input type="button" value="Checkin"/>
Terminator 3	<input type="button" value="Checkin"/>
Romeo and Juliet	<input type="button" value="Checkin"/>

Illustration 7

TASK T8. Task List for Managers

Users: Manager

Description:

This task is a navigation screen that allows a manager to choose among three movie summary reports grouped by title, genre, or award.

Reference: Illustration 8.

[Movie Summary By Title](#)
[Movie Summary By Genre](#)
[Movie Summary By Award](#)

Illustration 8

TASK T9. Movie Summary by Title

Users: Manager

Description:

This report shows the number of total rentals for each movie to date. The list should be sorted alphabetically by the movie title.

Implementation Notes:

- Only one SQL statement (on views and tables) should be used to render the list of movies.

Reference: Illustration 9.

Title	Year	#Copies	Availability	#Rents
Forrest Gump	1994	10	3	15
The Lord of the Rings	2002	10	6	10

Illustration 9

TASK T10. Movie Summary by Genre

Users: Manager

Description:

This report shows the list of genres and the corresponding number of awards, number of distinct movie titles, copies (total number of copies of all movies for each genre), and total number of rentals for each genre.

Implementation Notes:

- Only one SQL statement (on views and tables) should be used to render the GUI.

Reference: Illustration 10.

Genre	#Awards	#Movies	#Copies	#Rents
Action	10	30	60	200
Adventure	5	10	20	70
Comedy	5	20	30	80
Drama	7	15	40	80
Fantasy	2	10	30	30
Horror	0	20	20	50
Sci-Fic	3	30	90	100

Illustration 10

TASK T11. Movie Summary by Award

Users: Manager

Description:

This report shows the list of genres and their corresponding number of awards.

Implementation Notes:

- Only one SQL statement (on views and tables) should be used to render the GUI.

Reference: Illustration 11.

Genre	Oscar	NBR	Golden Globe	Golden Raspberry
Action	5	2	2	1
Adventure	5	3	2	0
Comedy	5	5	0	0
Drama	7	0	2	5
Fantasy	2	0	2	0
Horror	0	0	0	0
Sci-Fic	3	1	2	0

Illustration 11

TASK T12. Register

Users: Customer

Description:

This screen allows a new customer to register with the system. The system should track the customer's username (which must be unique), password, full name, and mailing address. If the user chooses a username that is already taken, an error should be displayed.

Reference: Illustration 12.

Register

A registration form titled "Register". It contains four input fields: "Username", "Password", "Full Name", and "Address". The "Address" field is a larger text area. Below the fields is a "Register" button.

Illustration 12

TASK T13. Task List for Customers

Users: Customer

Description:

After logging into the system, the customer is presented with a menu with two options: rent and view status/history.

Reference: Illustration 13.

[Rent](#)
[View Status/History](#)

Illustration 13

TASK T14. Rent Movie

Users: Customer

Description:

This screen allows the user to search the list of movies and select a movie to rent. The search fields are title, year, genre, actor, director. The user should be able to select multiple values for the genre, actor, and director fields. Selections within a field should be ORed together. Selections across fields should be ANDed. For example, if the user selects genres "Action" and "Comedy" along with actors "Tom Hanks" and "Sally Field" then the search results would contain movies within the genre "Action" OR "Comedy" (or both) that "Tom Hanks" OR "Sally Field" (or both) acted in.

If no option is chosen for a particular field, then that field will not participate in the search query. For example, if the user does not select any genres, then movies with any genre might be returned depending on the other search options. If the user clicks search without selecting any field values, then all movies should be returned.

When a user rents a movie, the number of available copies for that movie should decrease. A movie can be rented only if the availability is greater than zero. Using this screen, the customer can rent one title at a time.

The customer cannot have more than 5 currently rented movies altogether. An error should be displayed to the user if he or she attempts to rent a movie when he or she has 5 movies already checked out.

Implementation Notes:

- Only one SQL statement (on views and tables) should be used to render the GUI.

Reference: Illustration 14.

Title 2009 ▾

Genre Actor Director

Action	Tom Hanks	Tom Hanks
Adventure	Gary Sinise	Gary Sinise
Comedy	Sally Field	Sally Field
Drama	Ian McKellen	Ian McKellen
Fantasy	Reese Witherspoon	Reese Witherspoon
Horror	Lindsay Lohan	Lindsay Lohan
Sci-Fic		

Search Results

Title	Year	Director	#Awards	#Copies	Availability	Action
Forrest Gump	1994	Robert Zemeckis	5	10	3	Rent
The Lord of the Rings	2002	Peter Jackson	4	10	6	Rent

Movie Detail

Title Forrest Gump
Year 1994
Director Robert Zemeckis
Actors
 Gary Sinise
 Tom Hanks

Awards
Oscar 1995
 Best Visual Effect
 Best Picture
 Best Actor Tom Hanks
NBR 1994
 Best Supporting Actor Gary Sinise

Illustration 14

TASK T15. View Status/History

Users: Customer

Description: This GUI consists of two reports:

- View Status shows the list of currently rented movies to that customer, including the order date, title, and status.
- History shows the list of rented movies in the past including the order date, the title, and the date the movie was returned.

Reference: Illustration 15.

Currently Rented Movies

Order Date	Title	Status
5/12	X-men: Wolfverin	Processing
5/11	Basic Instinct 2	Delivered

History

Order Date	Title	Return Date
4/12	American Pie 2	4/15
4/11	Some Like It Hot	4/20

Illustration 15

3. GENERAL INFORMATION

Each group must have 3 or 4 members. As a group, you will decide whether to complete the lightweight or heavyweight project options. The two options are identical for phases I and II, but differ in the deliverable for phase III. Note that the option of whether you wish to do heavy or lightweight can wait until you get into phase III and as late as the final submission of phase III.

Heavyweight Option

Groups choosing this option will demo a working implementation of their project to the TA. The implementation must include a Java or web-based GUI (Graphical User Interface) that uses JDBC (Java Database Connectivity) or ODBC (Open Database Connectivity) for database access. The SQL statements you create in phase II will be embedded inside your system's source code.

Lightweight Option

Groups choosing the lightweight option will submit working SQL statements for each of the project tasks and demo the SQL statements to the TA. This option may be appealing to groups with little or no experience programming GUIs.

Database Access

Students in sections A and B will be provided with access to the Oracle Database Management System on ACME. See the course web page for further information on how to access Oracle from the ACME command line or from a Java program.

Students in section R (Georgia Tech Lorraine) will be required to install an open-source DBMS package locally. Two recommended DBMSs are MySQL and PostgreSQL. Students in sections A and B can also take this option.

- <http://www.mysql.com/>
- <http://www.postgresql.org/>

4. DELIVERABLES

Due dates for each of the phases are located on the respective class websites.

PHASE I (hard copy)

The deliverables include:

1. ER-Diagram (ER).
2. Information Flow Diagram.
3. A list of logical constraints that will be enforced. (Do not include data-type constraints, but rather semantic business logic related constraints).
4. Any assumptions made. Include explanations.

Notes:

1. The ER must capture the constraints of the system as much as possible whenever applicable, i.e. total participation, super/sub class, weak entities. If there is any drop down list in the system, it should be modeled by an entity.
2. The design must satisfy all the constraints. You are allowed to make up additional assumptions and constraints as long as they do not conflict with the specified constraints and requirements. If possible, those additional assumptions and constraints should be included in the E-R diagram.
3. You must turn in your report in hardcopy in class attached with cover a page that includes the names of all group members, their class section and their email address.

Slides on database design methodology useful for phase I:

http://www.cc.gatech.edu/classes/AY2008/cs4400_spring/methodologyFall2002.ppt

PHASE II (hard copy)

1. Cover Page.
2. Copy of the E-R Diagram (either from phase I (with any revisions) or from the solution).
3. Copy of the Information Flow Diagram from phase I (either from phase I (with any revisions) or from the solution).
4. Relational Schema Diagram (with primary and foreign keys identified, referential integrity is shown by arrows).
5. Create Table statements, including domain constraints, integrity constraints, primary keys, and foreign keys.
6. SQL statements for each task (*follows the template in the phase II guideline*).

Notes: A set of SQL statements may be required in order to complete one task. However, in such cases, the last SQL statement should show the output according to the specification. If mentioned, the returned tuples must be ordered according to the specification. The last SQL should resemble the output as much as possible. Views and nested queries may be used to support the tasks. A nested query can be broken down into views to make the query more readable.

PHASE III

Prior to the demo, the TA will give a sample data set. The database has to be populated with this data set prior to the demo. **5% will be deducted from the grade otherwise.**

LIGHTWEIGHT: Write a set of working SQL statements to perform all of the tasks specified in the project description.

HEAVYWEIGHT: Implement a working application with all functionality described in this document. Your source code should be mailed to the respective TA who grades your project by the deadline.

Deliverables for Phase 3 are:

- Copy of the Create Table statements from phase II (with any revisions),

- Source code (documented) for your system (soft copy),
- A set of working SQL statements for all project tasks (*Lightweight Option*)
- A functional GUI with embedded SQL statements that accesses your database (*Heavyweight Option*)
- A system demo to one of the TAs. You will use SQLPLUS or MySQL if you choose the lightweight option.