CS26/L – Software Fundamentals and Development

4381

ScreenPass

Cinema Ticketing System

Submitted to:

Mary Iana Bennel Buisan

Submitted by:

Jonathan M. Maningo

Sam Christian M. Ugmad

December 16, 2025

Table of Contents

List of Figures

List of Tables

Figure 1: Logo v2

The Project

ScreenPass is a small-scale cinema ticketing application currently built as a local desktop system. It manages movie data, schedules, seat layouts, and booking records. It uses an integrated interface for viewing available screenings, selecting seats, processing reservations, and generating a digital receipt. The system stores all operational data in a local database to maintain consistency across sessions.

Problem 1: Inconsistent Data and Booking Management

Basic ticketing solutions often rely on separate text files or disconnected spreadsheets to track schedules and seat status. Research on reservation systems indicates that these unstructured approaches lead to synchronization errors, where the user interface displays availability that does not match the backend records [1]. This disconnect increases the risk of double-booking and data corruption.

Solution 1: Relational Database with 3NF Schema

ScreenPass implements a normalized SQL database structure (3rd Normal Form) to centrally manage seat availability, showtime schedules, and booking history. By enforcing foreign key constraints and atomic transactions, the system ensures that every booking update is immediately reflected across the entire application. This architecture aligns with findings in real-time occupancy research, where structured data models are shown to significantly improve system accuracy and reliability [2].

Problem 2: Poor Usability in Ticketing Interfaces

Legacy ticketing systems display seat availability via textual lists or simple tables, which slows user decisions and reduces clarity. Studies on online reservation workflows suggest that abstract data presentation reduces user efficiency and increases interaction errors, as users struggle to visualize their location within the venue [3].

Research on online movie ticket reservation systems shows that unclear interface layout, confusing workflows, and poorly presented screening information reduce user efficiency and increase interaction errors, as users struggle to visualize their location within the venue [4].

Solution 2: Clear Movie, Showtime, and Seat Selection Interface

ScreenPass replaces textual selection with a graphical seat map representing the physical cinema layout. Users interact directly with a dynamic grid of buttons. Users select seats visually, and the interface updates immediately to prevent overlapping bookings. Booking a seat marks it as unavailable and grayed out, and the system prevents selection of seats that have already been booked.

The system provides immediate visual feedback preventing the selection of taken seats and calculating totals in real-time. The layout avoids unnecessary navigation steps and displays information in a format that improves ease of use, matching usability considerations identified in reservation-system testing studies [4].

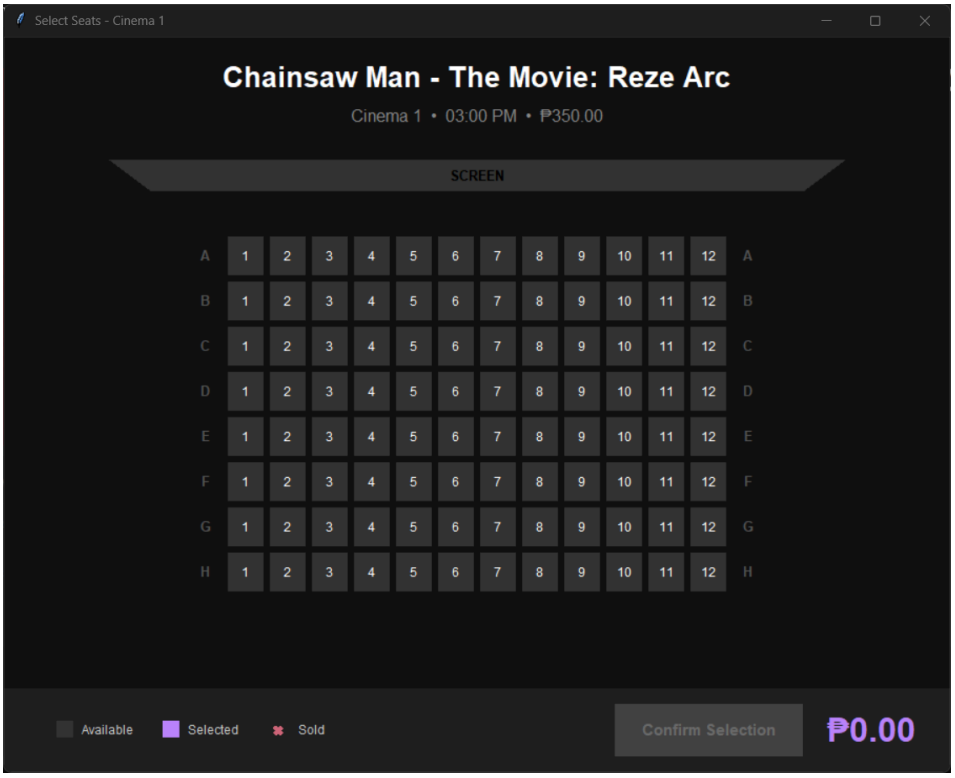

Figure 2: Seat Map

Problem 3: Inflexible Venue and Schedule Management

Many ticketing systems rely on hard-coded seat layouts or manual scheduling processes that fail to adapt to physical venue changes or complex rotation cycles [5]. This rigidity forces administrators to manually verify time slots for conflicts and limits the ability to reconfigure theater spaces (e.g., upgrading a Standard hall to VIP) without rewriting code or breaking existing booking records.

Solution 3: Dynamic Hall Configuration & Conflict-Free Scheduling

ScreenPass implements a dynamic hall management system that allows administrators to define custom grid dimensions (rows × columns) and seat classes, which the application renders programmatically. The scheduling engine integrates a conflict-detection algorithm that prevents overlapping showtimes within the same

hall. This ensures that the system scales effortlessly to different theater sizes while maintaining operational integrity.

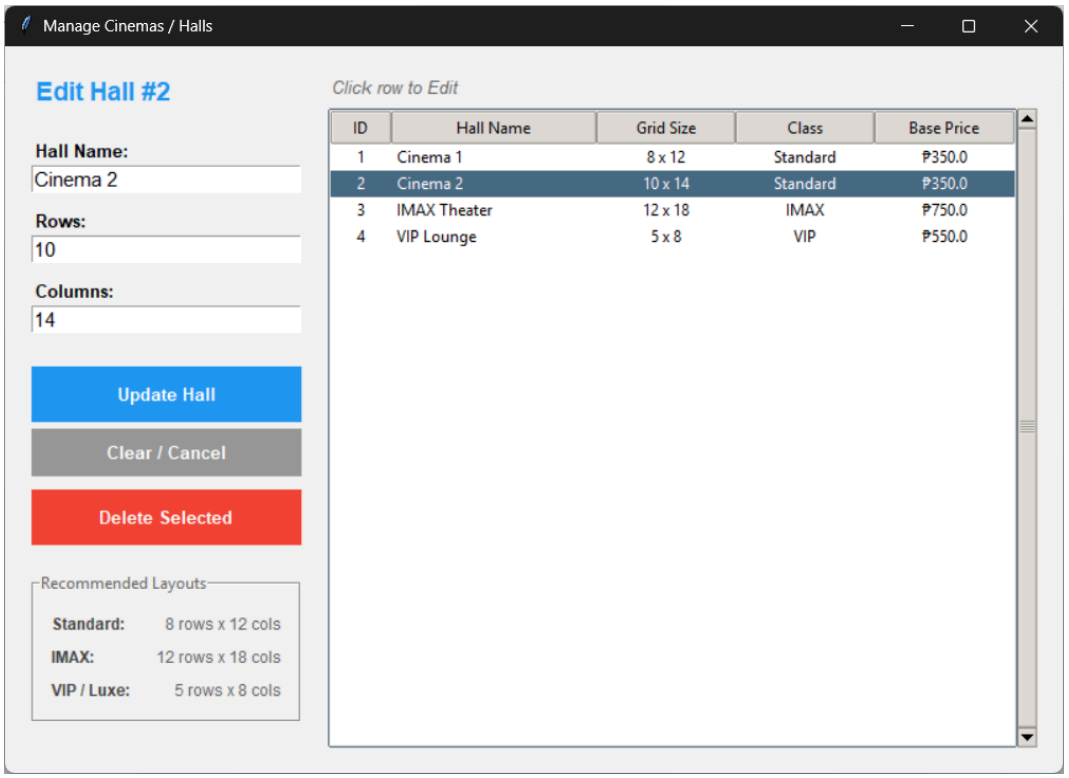

Figure 3: Dynamic Hall Configuration

Conclusion

ScreenPass integrates a structured data model, a streamlined interface, and dynamic cinema configuration and control. These design choices reflect findings from reservation-system research and user-experience studies, resulting in a functional desktop cinema ticketing application.

Tools

Language: Python 3.14

GUI Framework: Tkinter (Standard Library)

Database: MySQL / MariaDB

API: The Movie Database (TMDB)

Libraries: requests (API), mysql-connector-python (DB), Pillow (Image Processing).

Entity Relationship Diagram

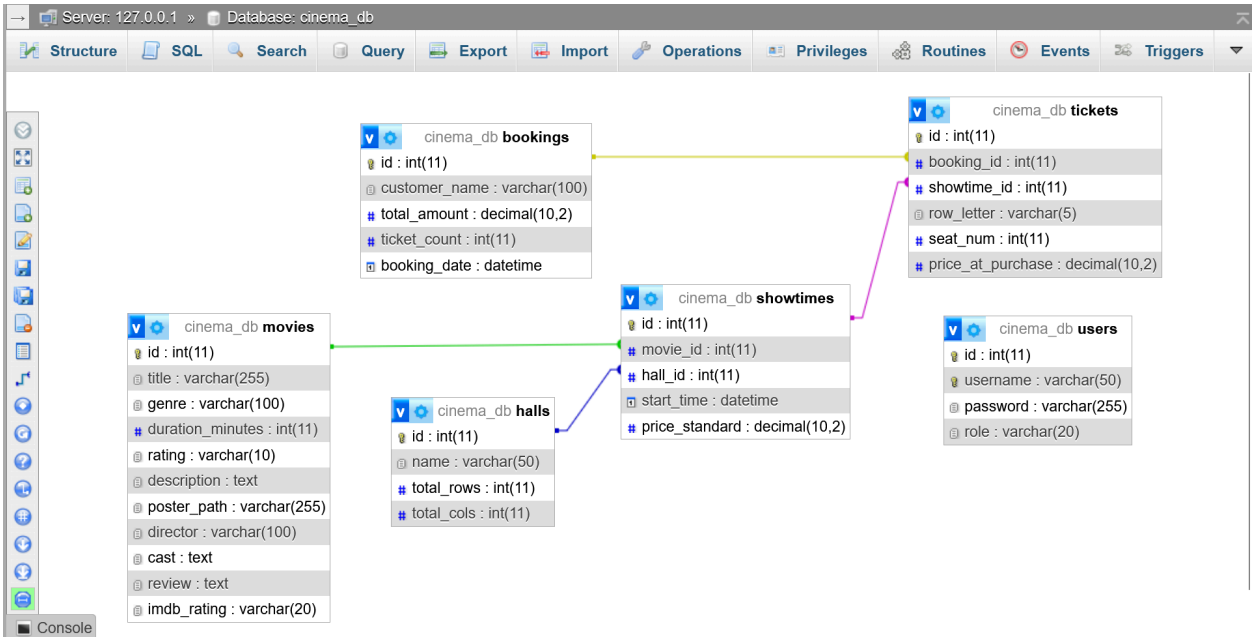

Figure 4: Cinema Ticketing System ERD

Data Dictionary

Table 1: Bookings

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	**id**	int(11)			No	*None*		AUTO_INCREMENT
2	**customer_name**	varchar(100)	utf8mb4_general_ci		Yes	*NULL*		
3	**total_amount**	decimal(10,2)			Yes	*NULL*		
4	**ticket_count**	int(11)			Yes	*NULL*		
5	**booking_date**	datetime			Yes	current_timestamp()		

Table 2: Halls

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	**id**	int(11)			No	*None*		AUTO_INCREMENT
2	**name**	varchar(50)	utf8mb4_general_ci		No	*None*		
3	**total_rows**	int(11)			No	*None*		
4	**total_cols**	int(11)			No	*None*		

Table 3: Movies

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	**id**	int(11)			No	*None*		AUTO_INCREMENT
2	**title**	varchar(255)	utf8mb4_general_ci		No	*None*		
3	**genre**	varchar(100)	utf8mb4_general_ci		Yes	*NULL*		
4	**duration_minutes**	int(11)			Yes	*NULL*		
5	**rating**	varchar(10)	utf8mb4_general_ci		Yes	*NULL*		
6	**description**	text	utf8mb4_general_ci		Yes	*NULL*		
7	**poster_path**	varchar(255)	utf8mb4_general_ci		Yes	*NULL*		
8	**director**	varchar(100)	utf8mb4_general_ci		Yes	*NULL*		
9	**cast**	text	utf8mb4_general_ci		Yes	*NULL*		
10	**review**	text	utf8mb4_general_ci		Yes	*NULL*		
11	**imdb_rating**	varchar(20)	utf8mb4_general_ci		Yes	*NULL*		

Table 4: Showtimes

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	**id**	int(11)			No	*None*		AUTO_INCREMENT
2	**movie_id**	int(11)			No	*None*		
3	**hall_id**	int(11)			No	*None*		
4	**start_time**	datetime			No	*None*		
5	**price_standard**	decimal(10,2)			No	*None*		

Table 5: Tickets

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	**id**	int(11)			No	*None*		AUTO_INCREMENT
2	**booking_id**	int(11)			No	*None*		
3	**showtime_id**	int(11)			No	*None*		
4	**row_letter**	varchar(5)	utf8mb4_general_ci		No	*None*		
5	**seat_num**	int(11)			No	*None*		
6	**price_at_purchase**	decimal(10,2)			Yes	*NULL*		

Table 6: Users

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	**id**	int(11)			No	*None*		AUTO_INCREMENT
2	**username**	varchar(50)	utf8mb4_general_ci		No	*None*		
3	**password**	varchar(255)	utf8mb4_general_ci		No	*None*		
4	**role**	varchar(20)	utf8mb4_general_ci		Yes	admin		

User Interface

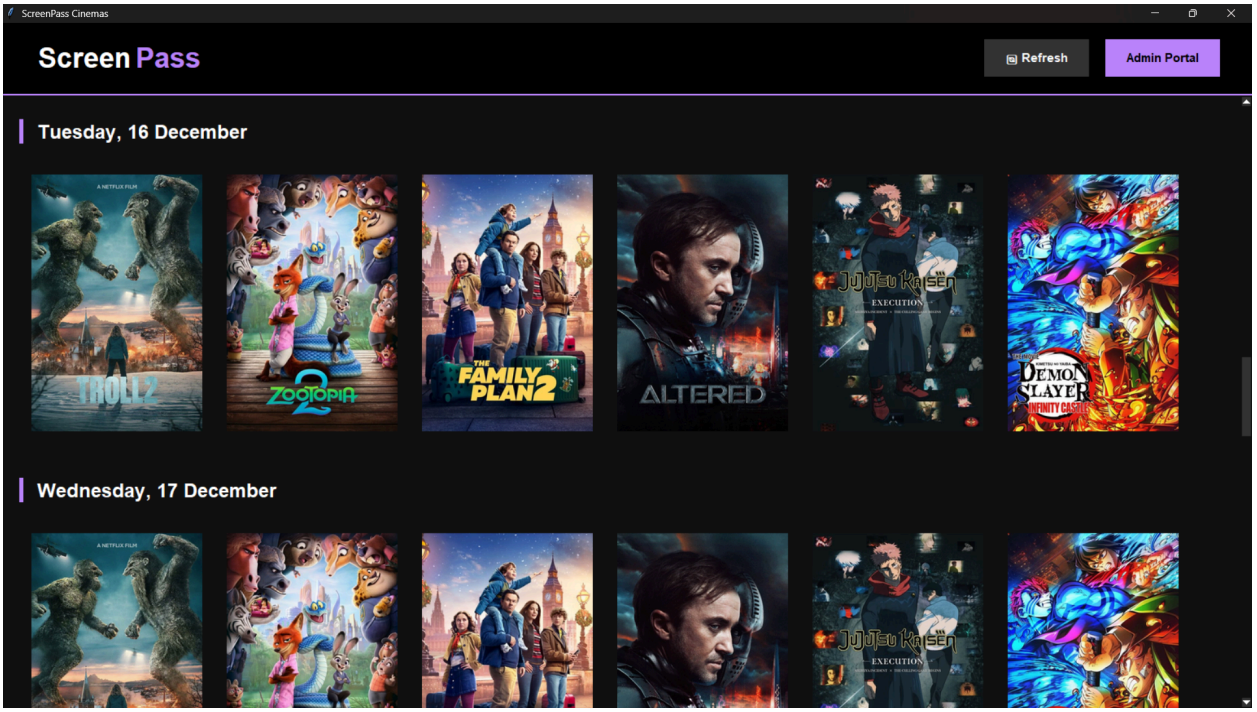

Figure 5: Customer Home Dashboard (customer_home)

Description:

The Customer Home Dashboard serves as the primary landing page for the application, designed with dark mode to reduce eye strain and emulate popular streaming platforms.

The main content area utilizes a scrollable vertical feed that groups movie screenings by date. Movies are represented by high-resolution poster art arranged in a grid. Users browse this timeline to see what is playing on specific days. Hovering on the poster reveals additional movie details while clicking on the movie poster transitions the user to the Showtime Selection screen for that specific film.

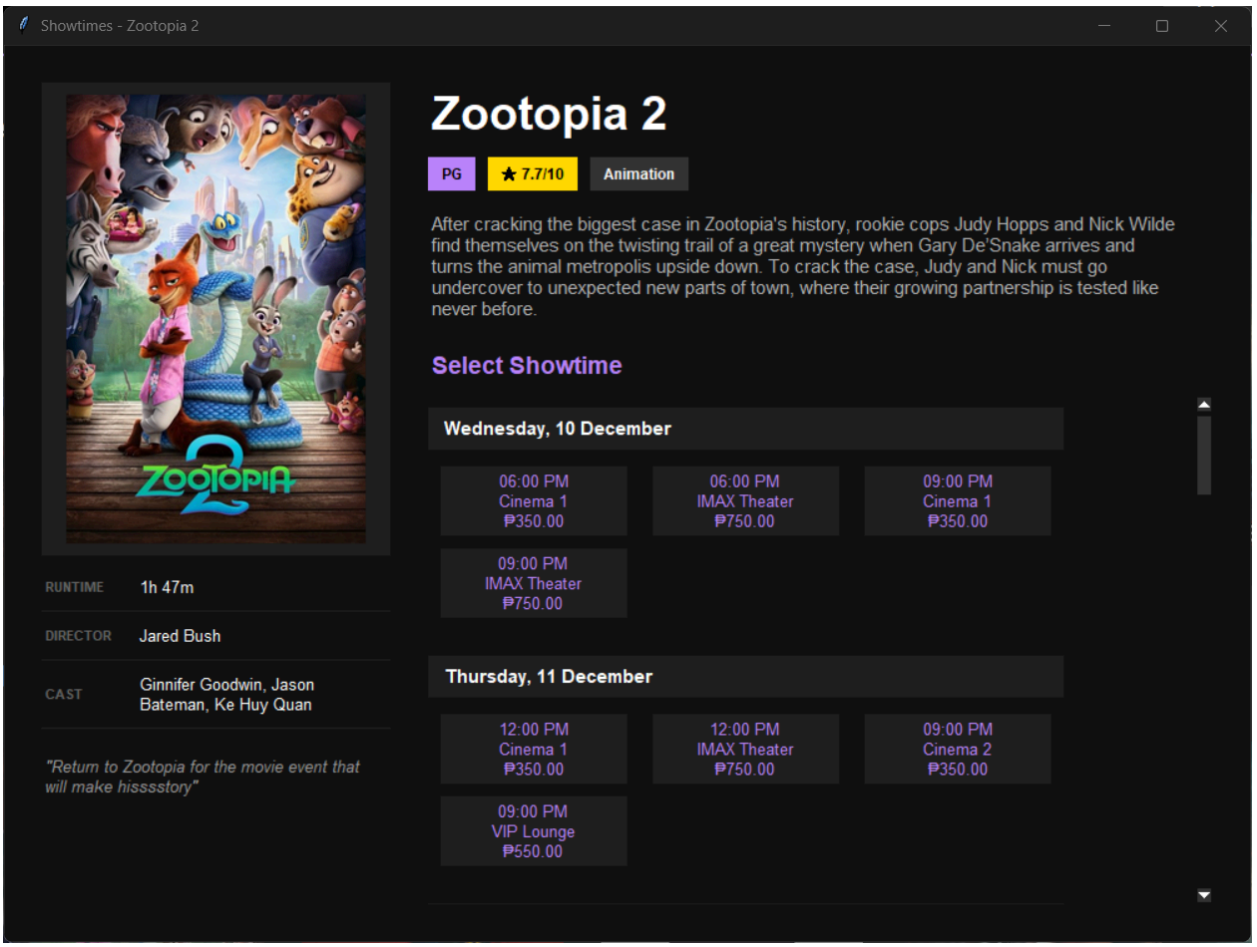

Figure 6: Showtime and Movie Details (customer_showtime_select)

Description:

Upon selecting a movie from the dashboard, the user is presented with the detailed Showtime Selection interface. This window is structured to provide all necessary decision-making information before booking:

The left sidebar displays the movie's official poster and technical credits. This section includes the runtime, the director, the primary cast list, and the movie's tagline displayed as a quote at the bottom.

The top section features the movie title followed by color-coded metadata badges indicating the MPAA rating (e.g., PG), the average review score (e.g., 7.7/10), and the genre. Below this is a full plot synopsis.

The lower portion of the main area displays available showtimes grouped by date. Each showtime is represented as a distinct button containing the start time, the specific hall (e.g., "IMAX Theater" or "Cinema 1"), and the ticket price. Clicking a time slot initiates the seat reservation process.

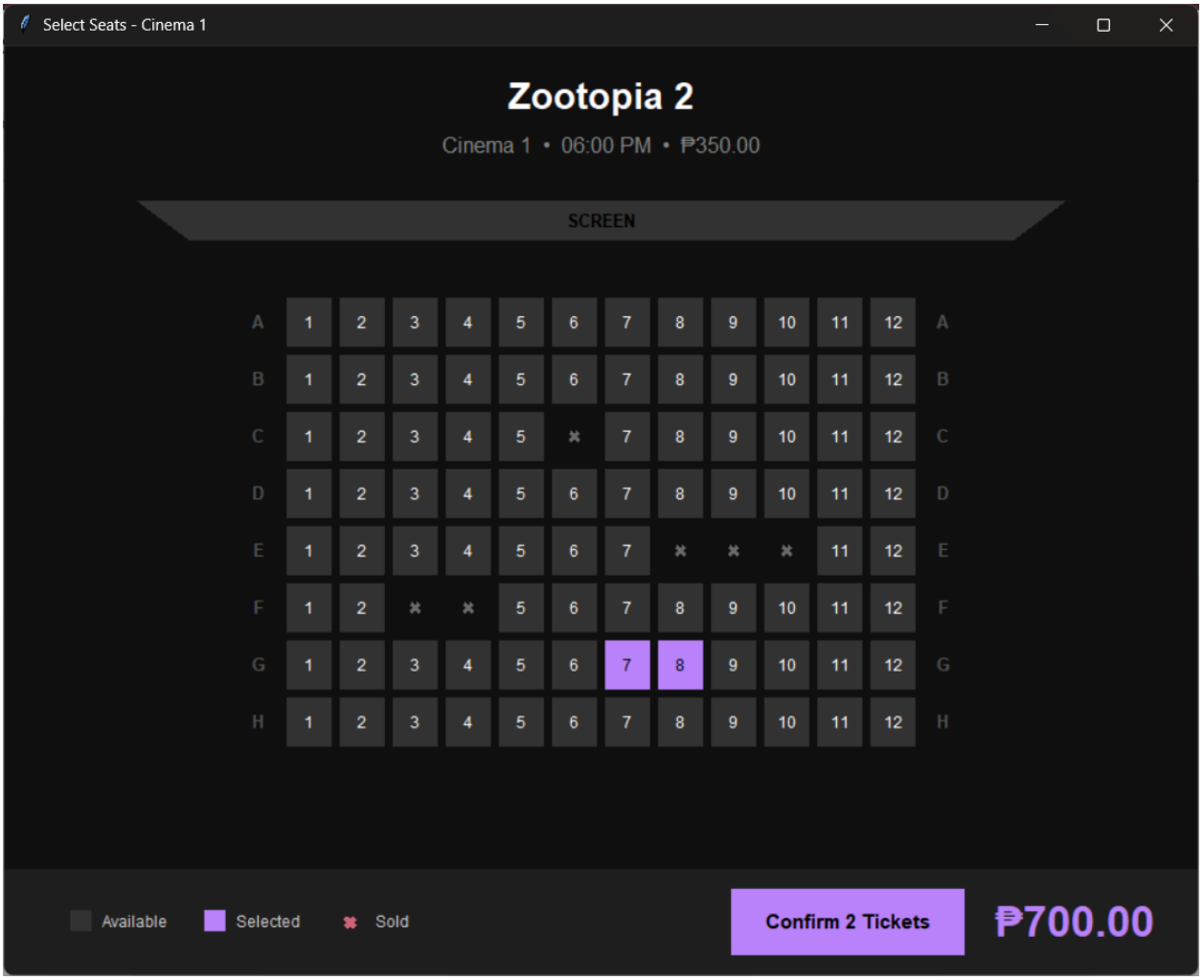

Figure 7: Seat Selection Interface (seat_map)

Description:

Once a showtime is selected, the application opens the Seat Selection window, visualized as a modal overlay. This interface is designed to simulate the physical layout of the cinema hall:

The center of the window features a dynamic button matrix representing the seating arrangement (Rows A–H). The system uses visual cues to indicate seat status: dark grey for Available, a purple accent color for Selected, and a marked 'X' for Sold/Unavailable seats.

The footer contains a legend for the color codes and a dynamic total price indicator. As users toggle seats, the total price and the "Confirm Tickets" button text update instantly to reflect the current selection.

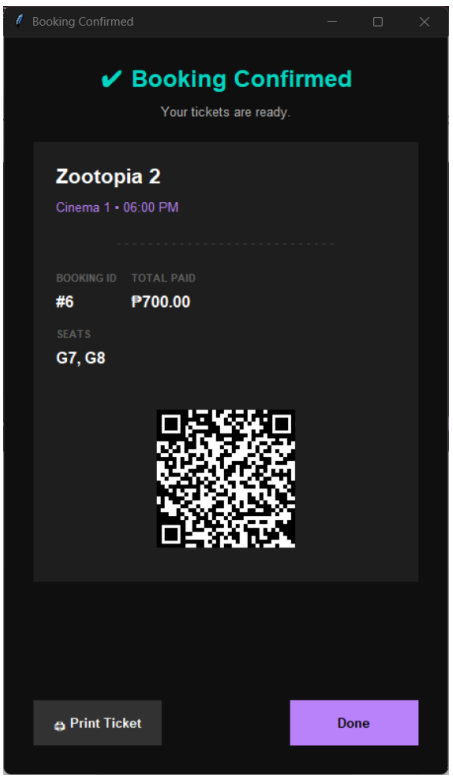

Figure 8: Booking Confirmation Receipt (receipt_window)

Description:

Upon successfully confirming a purchase, the system generates a Booking Confirmation window. This screen acts as a digital receipt and entry pass:

The upper section explicitly confirms the transaction status and lists critical details, including the specific movie, cinema hall, showtime, unique Booking ID, and the specific seat numbers assigned (e.g., "G7, G8").

A generated QR code dominates the center of the card, intended for scanning by theater staff at the point of entry to validate the booking.

The footer provides a "Print Ticket" option for users who prefer a physical copy, and a "Done" button to close the transaction loop and return the user to the dashboard.

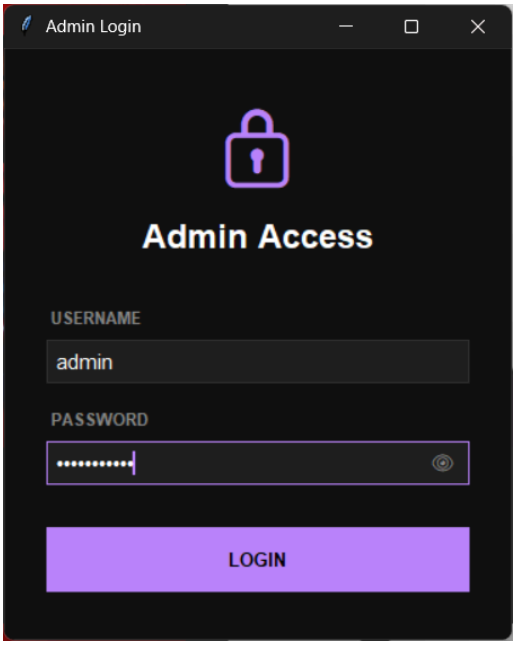

Figure 9: Admin Login Panel (admin_login)

Description:

To ensure system security, administrative functions are gated behind a dedicated login modal. The panel requires a valid username and password. The password field includes a "show/hide" toggle (eye icon). Upon successful authentication, the modal closes and launches the main Admin Dashboard.

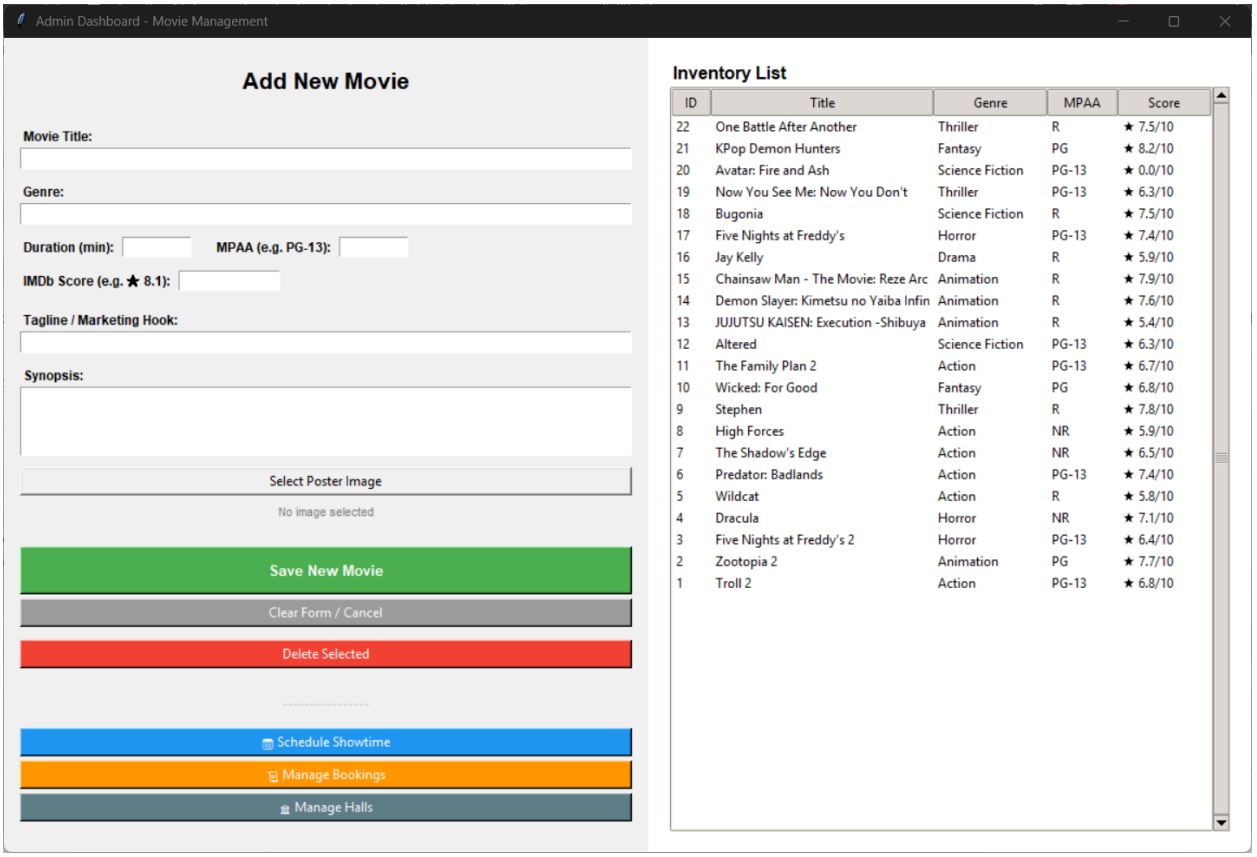

Figure 10: Admin Dashboard - Movie Management (admin_dashboard)

Description:

The Admin Dashboard is the central control hub for theater operations, providing CRUD (Create, Read, Update, Delete) capabilities for the movie catalog. The layout is split into two functional zones:

On the left, text fields let administrators add or update movie details. This includes fields for the title, genre, runtime, MPAA rating, synopsis and poster image. A bottom toolbar allows seamless switching between different administrative modules: "Schedule Showtime," "Manage Bookings," and "Manage Halls."

A scrollable table on the right provides a real-time view of the current list of available movies. Columns display key metrics (ID, Title, Genre, Rating, Score). Selecting any row in this table automatically populates the input form on the left.

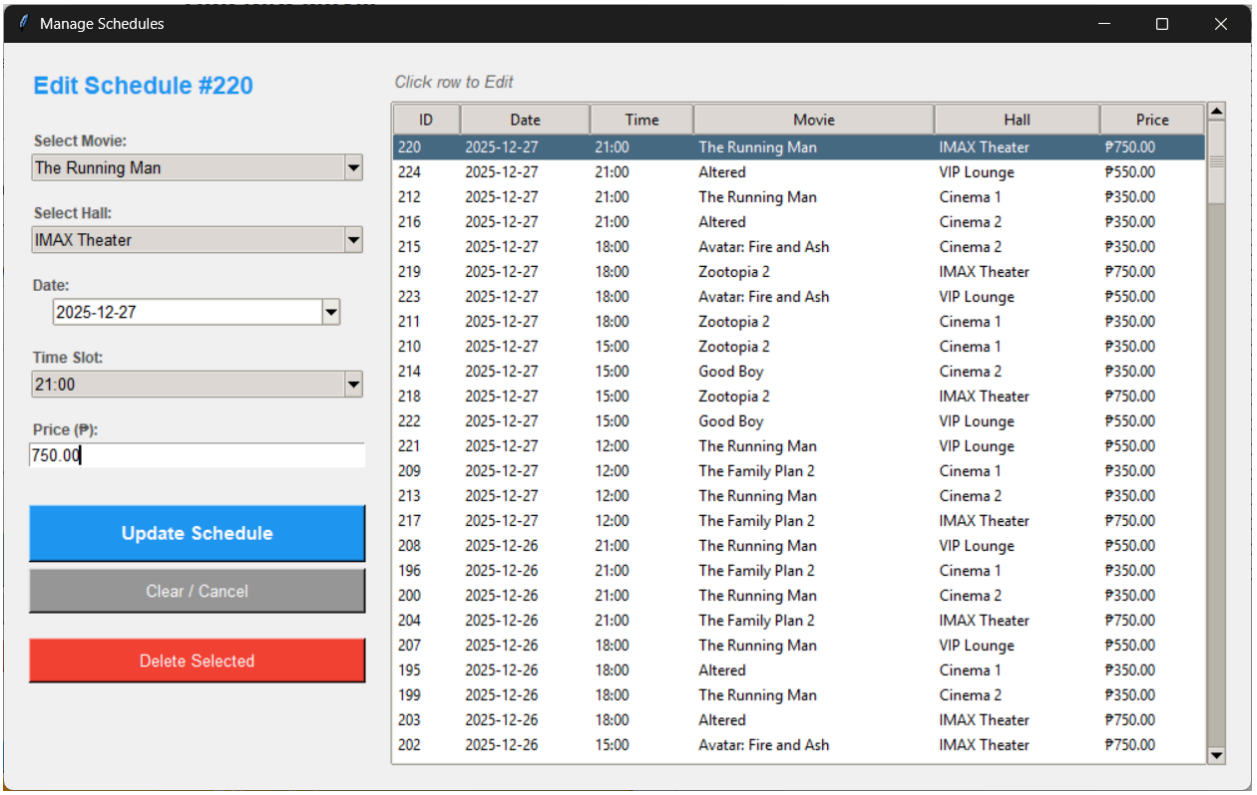

Figure 11: Admin Dashboard - Schedule Showtimes (admin_scheduler)

Description:

The Scheduling window is the module of the cinema's daily operations, allowing administrators to assign movies to specific halls and time slots.

The section on the left features dropdown menus for selecting movies and halls. A date picker and a strictly defined "Time Slot" dropdown (12:00, 15:00, 18:00, 21:00) ensure schedules align with operational hours. The price field auto-fills based on the selected hall type (e.g., changing from Standard to IMAX automatically

updates the price from ₱350 to ₱750), though it remains editable for specific cases and fluctuations.

The schedule table on the right shows a comprehensive grid displaying all active showtimes. The table provides an at-a-glance view of the ID, Date, Time, Movie Title, Hall Name, and Ticket Price.

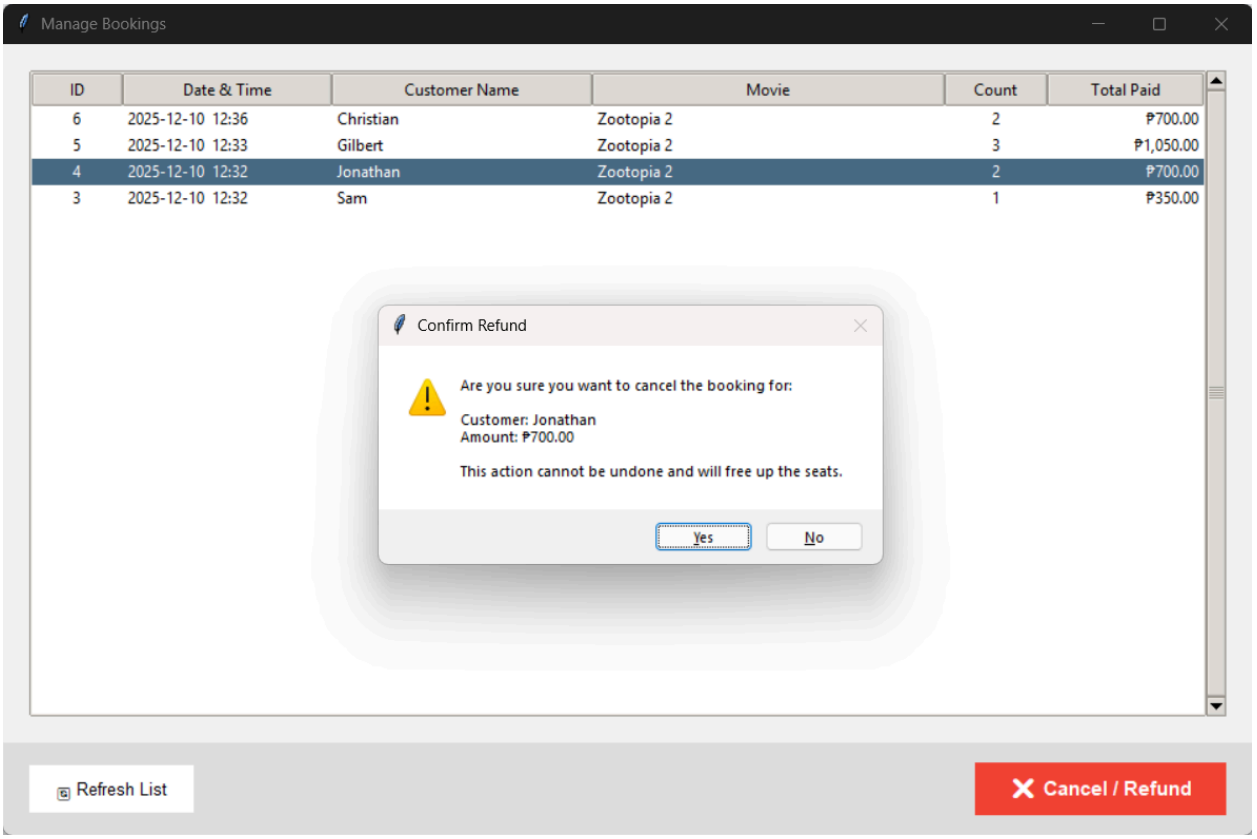

Figure 12: Admin Dashboard - Manage Bookings (admin_bookings)

Description:

This screen provides a transaction log and customer service interface for handling existing reservations. The main table lists every booking made in the system, showing the Booking ID, Date/Time of purchase, Customer Name, Movie, Ticket Count, and Total Amount Paid. This serves as a ledger for sales tracking.

Administrators can select a specific booking to initiate a refund. As shown in the figure overlay, a confirmation modal appears. Confirming this action deletes the booking record and immediately frees up the previously occupied seats in the database, making them available for purchase again.

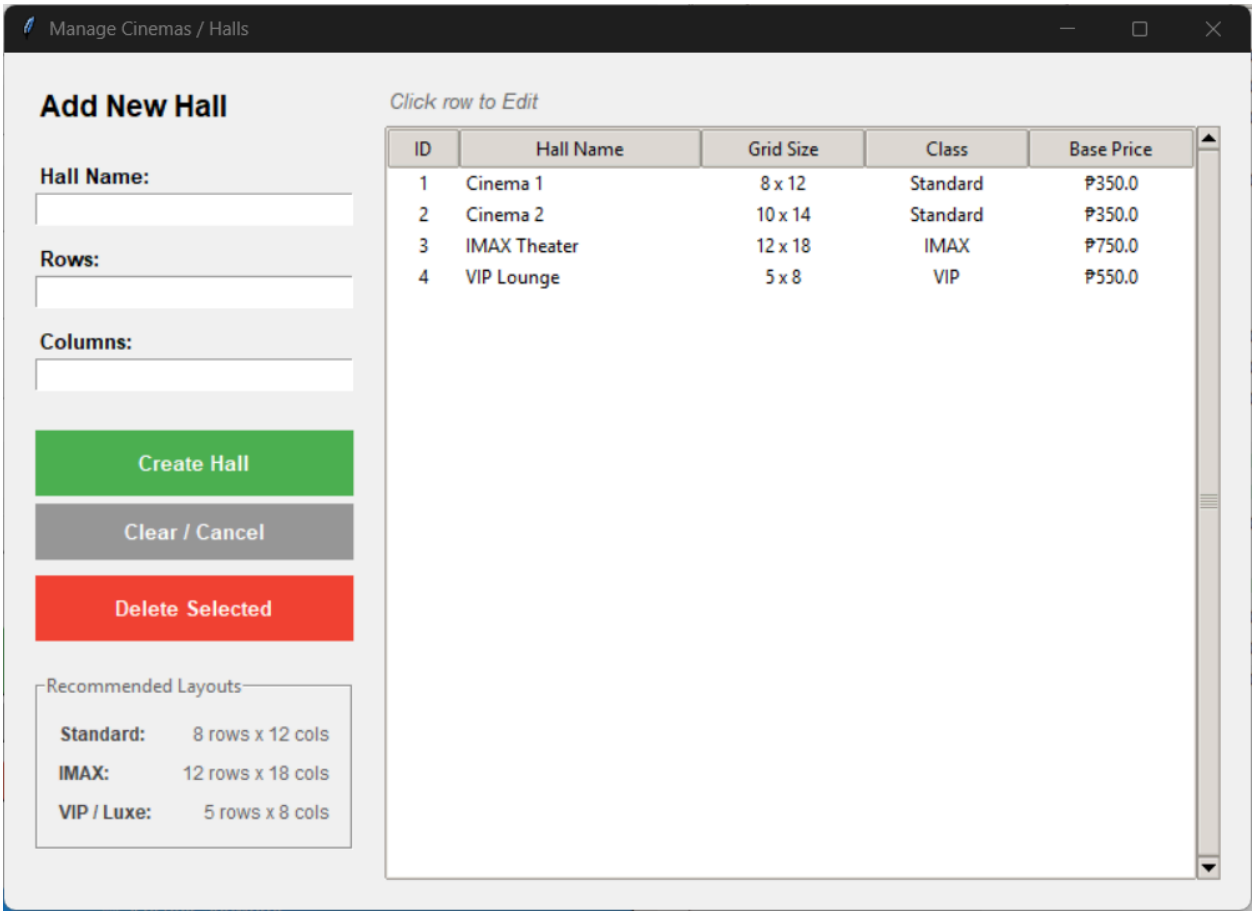

Figure 13: Admin Dashboard - Manage Halls (admin_halls)

Description:

The Hall Management interface allows the cinema to scale its physical infrastructure within the software.

Administrators can define new cinema spaces by specifying a name (e.g., "VIP Lounge") and the grid dimensions (Rows × Columns). This dynamic sizing directly dictates the layout of the Seat Selection screen seen in Figure X.

A "Recommended Layouts" guide is provided at the bottom left to standardize hall sizes (e.g., Standard is 8x12, IMAX is 12x18), to aid in creating seating maps.

The list view displays all configured halls along with their dimensions, classification (Standard/IMAX/VIP), and base pricing. This classification is critical as it determines the default ticket prices used throughout the scheduling system.

References

[1] J. Condell, J. McDevitt, D. McGilloway, J. McGlinchey, and G. Galway, “Seat Booking System for a Cineplex,” Computer and Information Science, vol. 1, no. 2, pp. 67–79, 2008. DOI: 10.5539/cis.v1n2p67. Available: https://ccsenet.org/journal/index.php/cis/article/view/1912

[2] M. J. García-Granja, E. B. Blázquez-Parra, G. Cimadomo, and F. Guzmán-Navarro, “Development of an innovative seat reservation system for university buildings based on BIM technology,” Buildings, vol. 12, no. 11, p. 1786, 2022, doi: 10.3390/buildings12111786.

[3] D. Jumanto, “User Experience Study on Self-Ticketing Machine in Cinema,” *US Journal*, 2024. Available: https://opsearch.us/index.php/us/article/view/115

[4] M. M. F. Guzman, J. A. B. Manaloto, A. D. D. R. Senina, and A. J. A. Portus, “Usability Testing of Online Movie Ticket Reservation in the Philippines,” Japanese Journal of Ergonomics, vol. 53, suppl. 2, pp. S592–S595, 2017. Available: https://www.researchgate.net/publication/319582892_F6-2_Usability_Testing_of_Online_Movie_Ticket_Reservation_in_the_Philippines

[5] J. Eliashberg, Q. Hegie, J. Ho, D. Huisman, S. Miller, S. Swami, C. Weinberg, and B. Wierenga, “Demand-driven scheduling of movies in a multiplex”, International Journal of Research in Marketing, vol. 26, no. 2, pp. 75–88, Jun. 2009.

Appendix

The source code for the ScreenPass Cinema Ticketing System is hosted on GitHub. This repository contains all production files, version control history, and related documentation assets.

- **Repository URL**: https://github.com/sgmad/cinema-ticketing-system
- **Project Title**: ScreenPass Cinema Ticketing System

The following essential non-code files are located in the repository root or subdirectories:

- **Database Schema**: /db/database_setup.py (Initialization script for the MySQL database)
- **Poster Assets**: /assets/sample_posters/ (Directory containing movie image files)