

Project 2: Persistent Form Proposal

Group 15: Alia Hanafy, Sheza Syed, Hamda Yousuf

(1010427012, 1010455589, 1010560055)

[Github Repository Link] <https://github.com/shezajsyed/project2persistentform.git>

Project/Idea Overview:

Our main idea is to create a portal for individuals to purchase movie tickets at a specific theater specializing in (as of now, romance or holiday) movies during the month of December. We plan to offer 5 different films to choose from, as well as a good amount of seats to choose from, approximately 20 to 30 with regards to their availability. We will include multiple windows to showcase the process, as well as use different classes to organize the commands.

How we plan to fulfill the requirements for a functional GUI:

We will use Tkinter to build the fully-functional GUI. Functions this will allow us to access include Label, Button, etc. Users will be able to click buttons, fill in entries, toggle/use menus, and more. We will implement “data persistence” by making sure the interface allows us to save and load data. This will be shown through the user’s actions building upon each other. E.g. the chosen seat or seats are “saved” into the database so that they show up when the user needs to pay for them.

To follow the criteria of the CRUD functionality model, the user should be able to complete the following four functions. The user will be able to **create** by adding in their data. The user will be able to **read** the displayed data. The user will be able to **update** or edit their data if they change their mind about the amount of or the location of the seats. The user will be able to **delete** data by being able to remove a seat if they change their mind about the amount. The interface will enable data entry by allowing the user to type in their information or select their desired options based on their own individual needs and wants. The user will be able to input the required info (e.g. payment details). We will have a Github repo to showcase our work.

How can we make our project successful?

Once our project is completed, it should first prompt the user to log-in or sign-up with a username and password. They should then be taken to a screen with the movie options to choose from. This interface is for one specific theater so there is no need to choose the location, all they need to do is specify the date. Once the movie and date are chosen, they will be taken to a window to select their seat. They will be able to specify how many seats they want. The user will then be taken to the payment window. The user inserts payment details and the interface should make sure it only accepts correct input types. For example, only numbers for the credit card number, as well as a certain number of characters. Then the user confirms their order and receives their ticket confirmation.

What tools, technologies, and libraries could we use to create our project?

1. Tkinter - to create the overall GUI and add features like buttons and labels [ttk]
2. Matplotlib and Pandas - for data analysis [could show most popular movies/days]
3. SQLite3 or JSON - for storing the movie/booking data
4. Github - to manage version control and allow for collaboration

Supplementary Information: [Project Breakdown]

Classes

- Movies
- Seats
- Dates/showtimes
- Payment

Windows

- Log-In
- Movie Selection
- Seat Selection
- Payment Details
- Ticket Confirmation

Data Entry

- User will enter login information (**username and passcode**)
- User will then be able to **select a movie and showtime**
- User selects **seat**
 - Unavailable seats will be marked off so that the customer is not able to choose them
- User inserts **payment details** and pays for ticket

Data Persistence

- Storing data so it can be retrieved and used for later bookings on the user's account

Business system

- The business we have selected is the cinema industry
- It will feature aspects required for a customer to book a ticket to watch a movie
- Will feature login functionality to access the ticket booking system
- Inventory management to manage how many tickets are available as well as show times