Parallel Processing Project- Movie Ticket Booking

Member Name	Member ID	Level
آية محمد عبدالحليم محمود الشيخ	20220096	3
جودى محمد اسامة عبيدو	20220127	3
بسمله حسام ماهر	20220108	3
منة الله هيثم صلاح حسن	20220508	3
لجين عمار محمود خطاب	20220354	3
سارة عماد زكي عثمان	20220197	3

Project Codes and Documentation link:

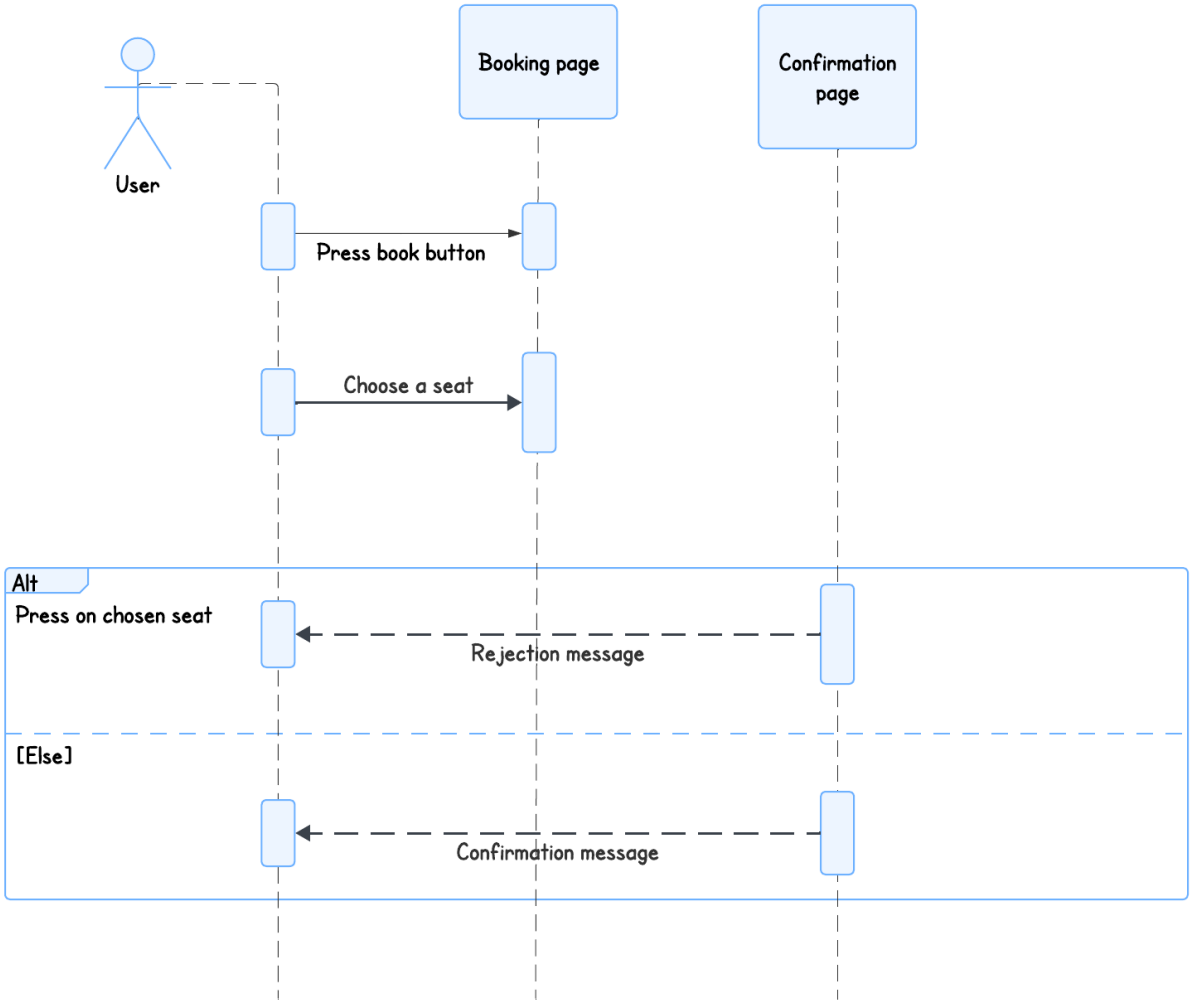
https://github.com/Judyzox/parallel-processing-movie-booking-system.git

System Description:

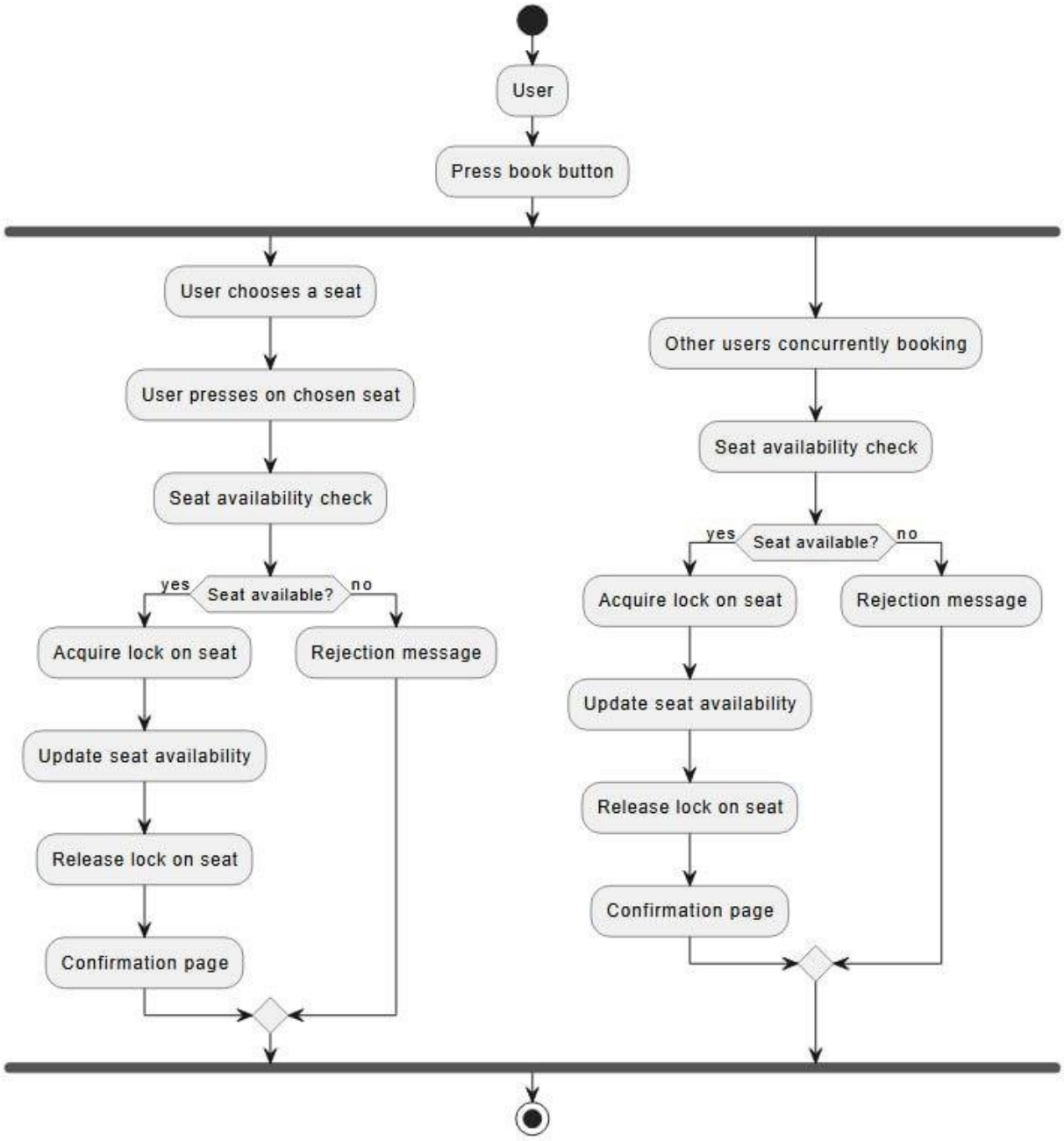
- Simulate a system where multiple users can book movie tickets at the same time.
- Uses multithreading, where threads are made for user booking actions, prevents double-booking with synchronization, and dynamically update seat availability.

System Diagrams:

1)Sequence diagram:

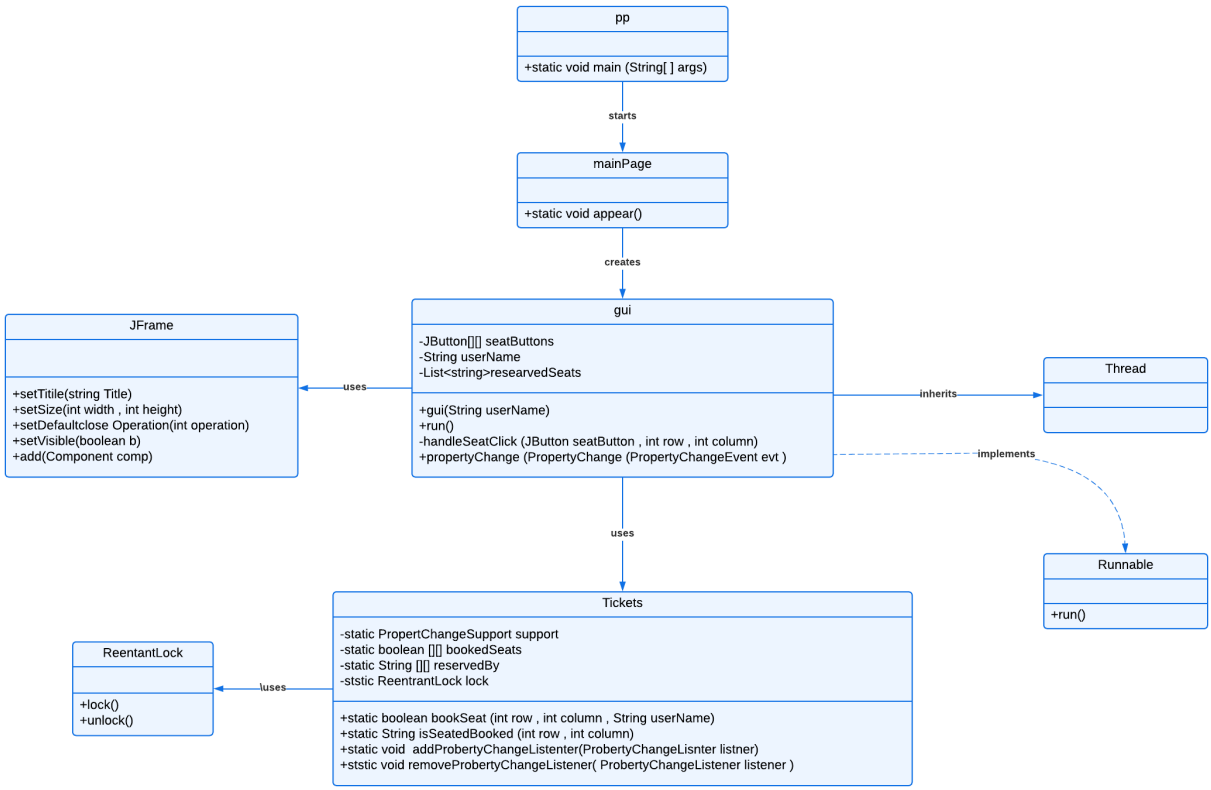


2)Activity diagram:

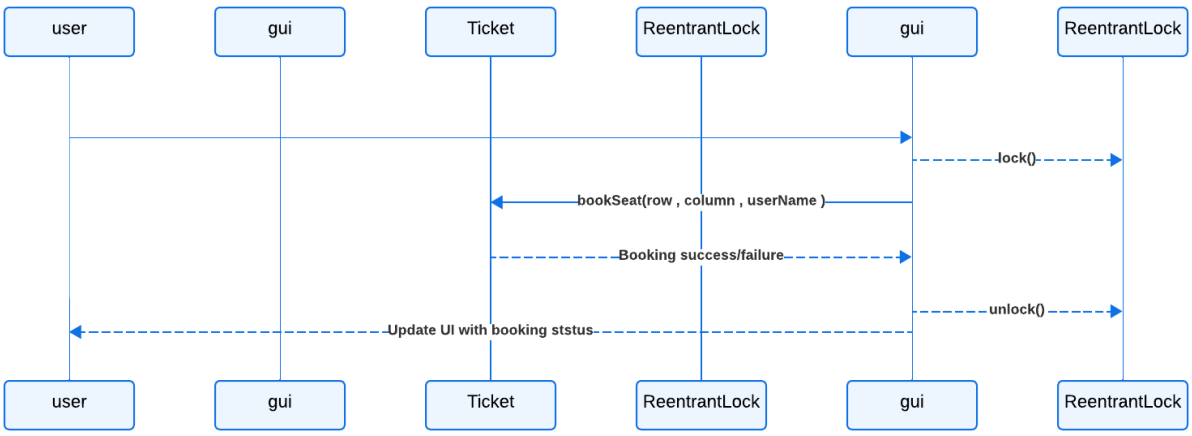


3) multithreading Clarification:

A) Class Diagram:



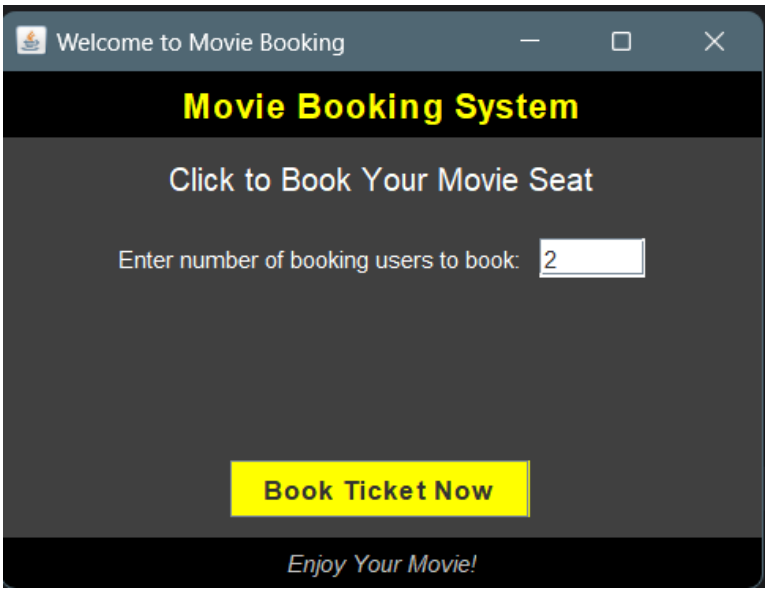
B)Thread Interaction Diagram:



Code Overview:

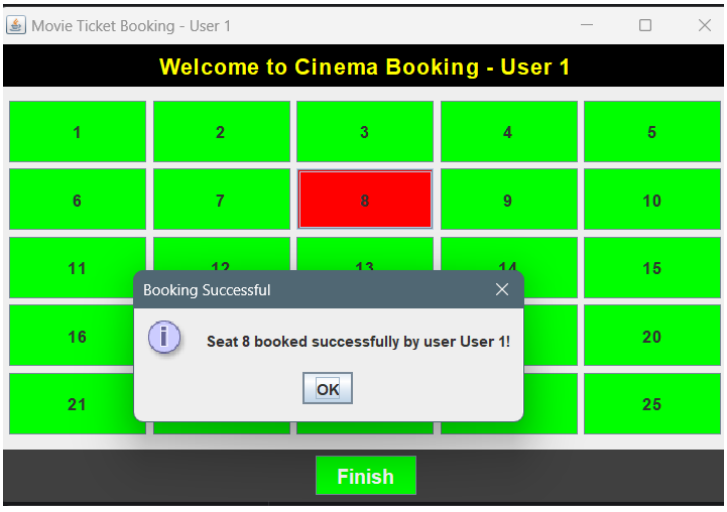
1) main page:

This program defines a main Page class with a display method to create a Swing-based GUI for a movie booking system. It uses a main frame with header, content, and footer panels, allowing users to enter the number of booking users and start corresponding threads (gui instances) for seat booking. The "Book Ticket Now" button validates user input and spawns the specified number of threads to handle bookings concurrently.



2) Thread Creation and user windows opening:

After entering the number of users, this creates 2 threads and 2 windows are opened allowing the users to book their seats.
When a user books a seat it directly changes to red indicating that its booked and reflects at the other user's window so that the other users know that this seat is already booked.
Every time a user books a seat a message appears to user approving his Booking.



3) Dealing with race conditions:

When any of the users try to access a seat which is already booked by another user a message is popped telling the user that this seat is already reserved by another user and it cannot be booked. This is because each seat is locked by a semaphore lock so that no 2 or more users can book the same seat.

4) Termination

After both users have made their bookings when they press finish the lock is released and a message appears informing the users with their booked seats and waiting for their confirmation. And once the users confirm their bookings the code terminates.