Object-oriented Programming
Project proposal

Xie Youtian 2430026233

April 30, 2025

1 Introduction

I choose event ticket booking system as my project. Below are a kind of business logic of event ticket booking system in my recognition and may not be 100% accurate.

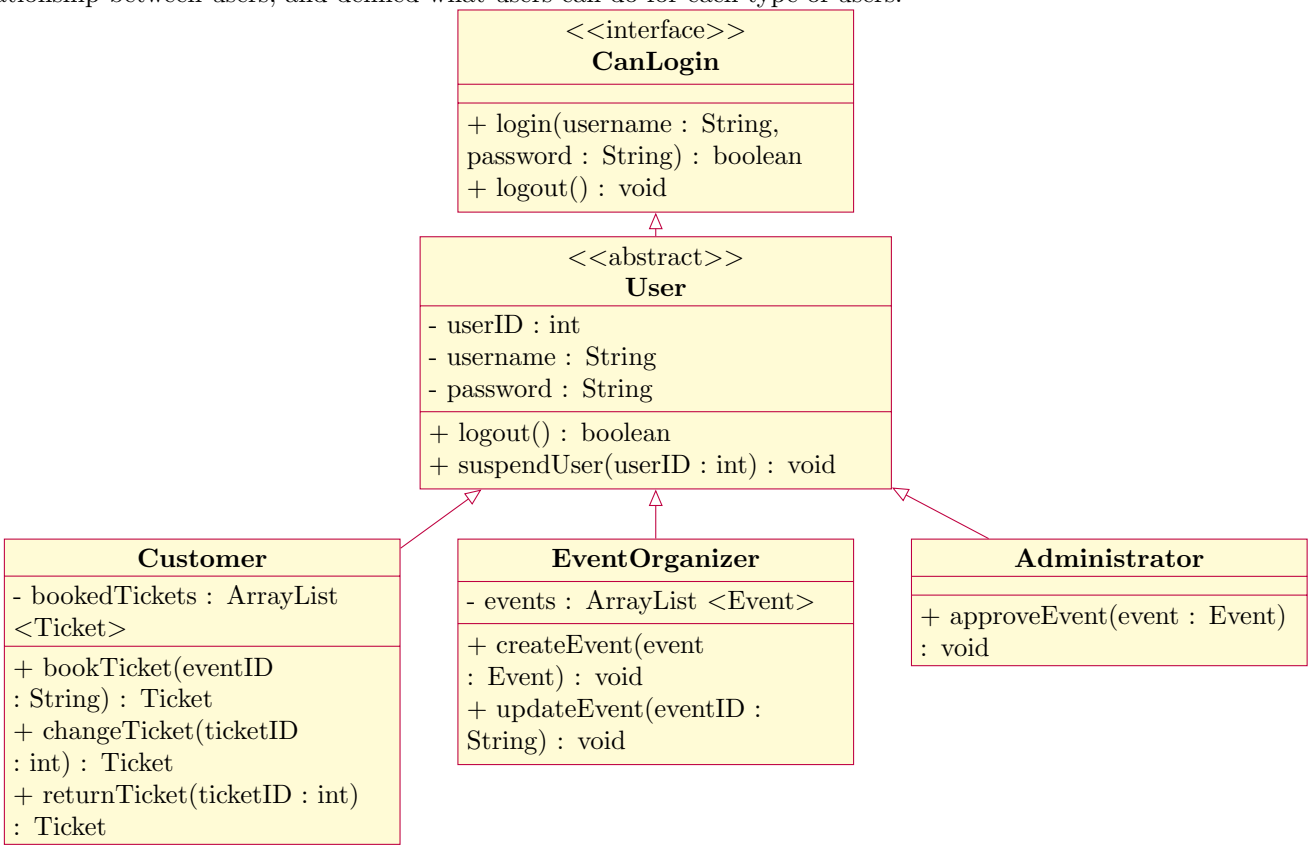
In a ticket booking system, users have 3 roles: customer, event organizer and administrators. We operate with events and tickets. To simplify the design, we assume that there is no seat arrangement, like ACGN conventions or business fairs.

So, the business logic is clear:

1. Customer book, return and change tickets.
2. Event organizer apply for events and maintain event informations.
3. System administrator manage users and approve event informations.

2 Design of OOP Concepts with UML

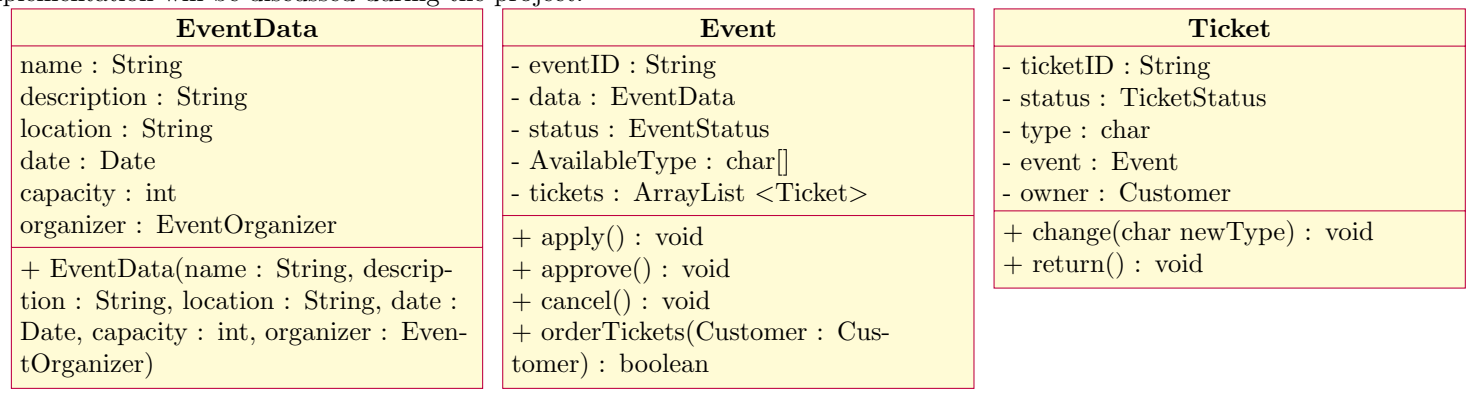
Below is a inheritance system of interfaces and classes of users. We use a simple inheritance system to represent the relationship between users, and defined what users can do for each type of users.



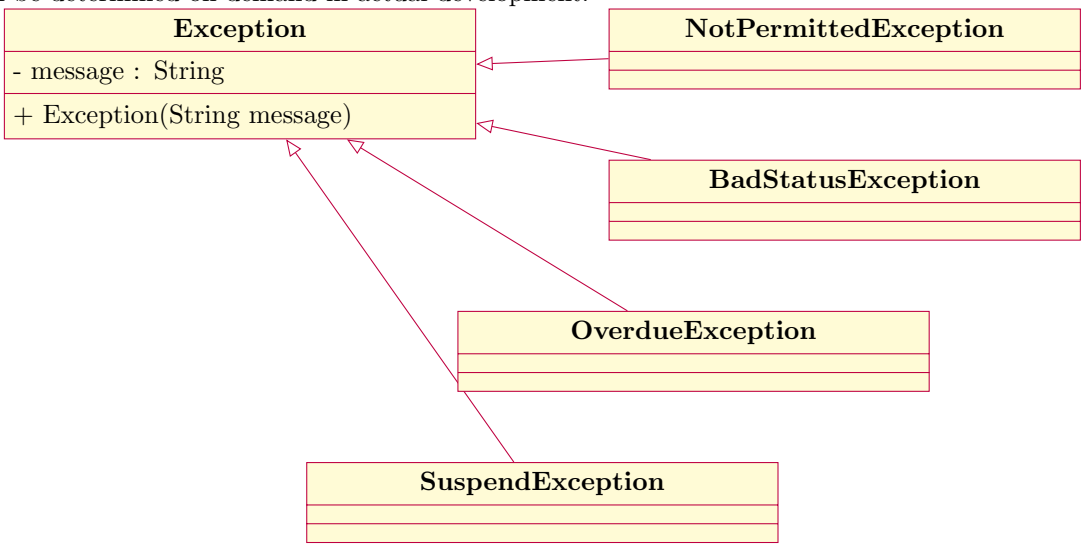
Then let's move to the next step, we need to define the relationship in events. First, there is a enumeration type named **EventStatus** that has 5 values:

- **PLANNING** - Event is planning, but not ready for approval.
  The organizer of this event and administrators can see this event,
  but administrators can't approve it as it is not shown in the waiting list.
- **APPLYING** - Event is applying for approval. When the event organizer updates its information
  it will go back to **PLANNING** status.
  Administrators can see this event in the waiting list and decide whether to approve it or not.
- **APPROVED** - Event is approved. All users can see this event, but customers can't book tickets.
  When the event organizer update its information it will go back to **PLANNING** status.
- **DISAPPROVED** - Event is disapproved, and behaviours are same as **PLANNING** except on display.
- **AVAILABLE** - Event is available for booking.
- **CANCELLED** - Event is cancelled. All users can see this event, but customers can't book tickets.
  Other behaviours are same as **PLANNING**.

Below are 3 classes handling event information and ticketing logic, and each **EventData** is corresponding to a **Event**. Implementation will be discussed during the project.

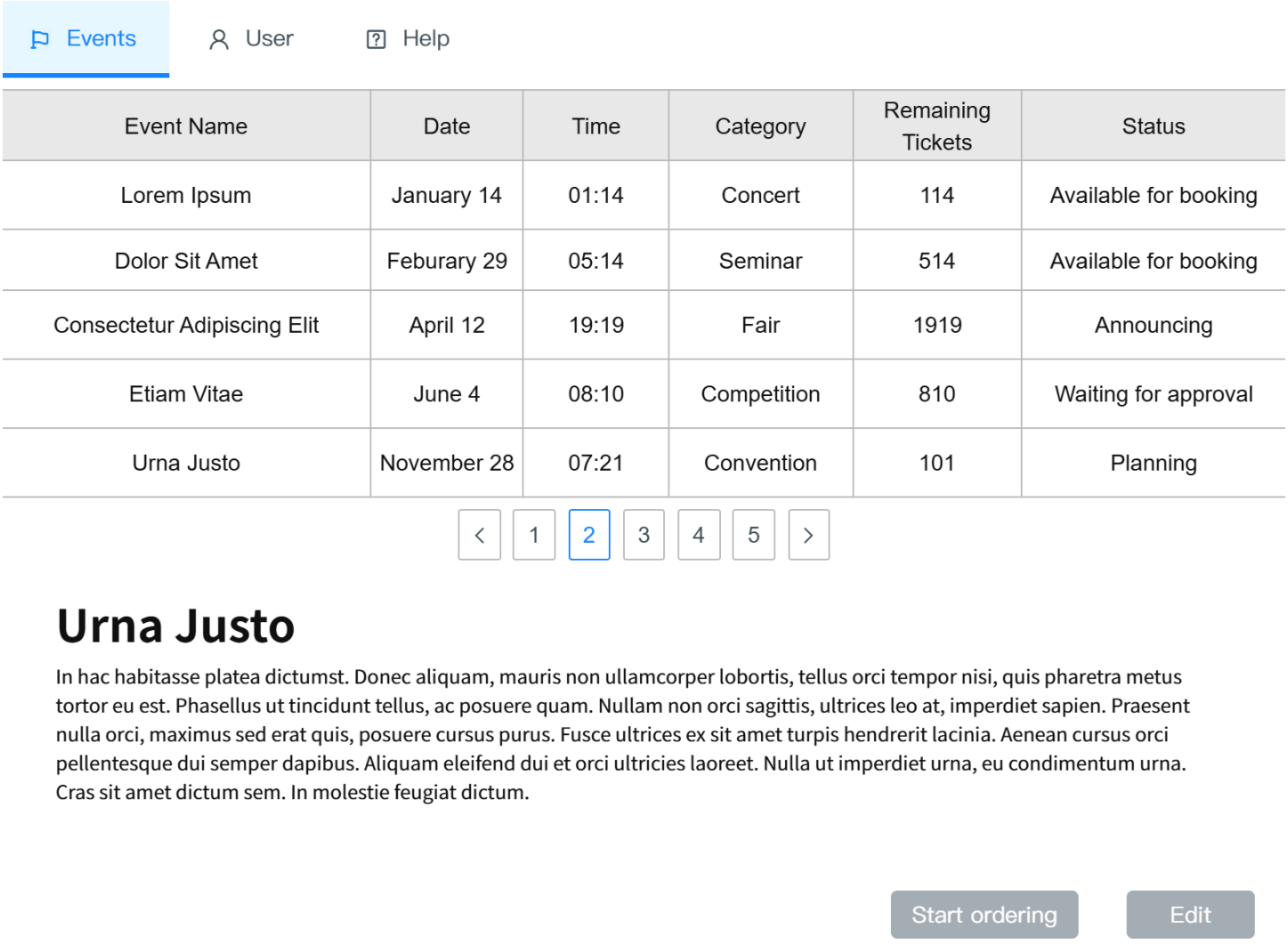


Finally, exception handling is also important. Below are expections used in the program and usage of these wxceptions will be determined on demand in actual development.



3 Design of user interface

The prototype of main window is shown in the following picture, which does not represent the final design as Swing library is old and looks ugly while can't be configured easily to look so modern and hygiene.



4 Storage and I/O

Data will be stored in a SQLite database, with 3 tables: users, tickets and event data. Table name for each table will be named as `tb_user`, `tb_ticket` and `tb_event`.
