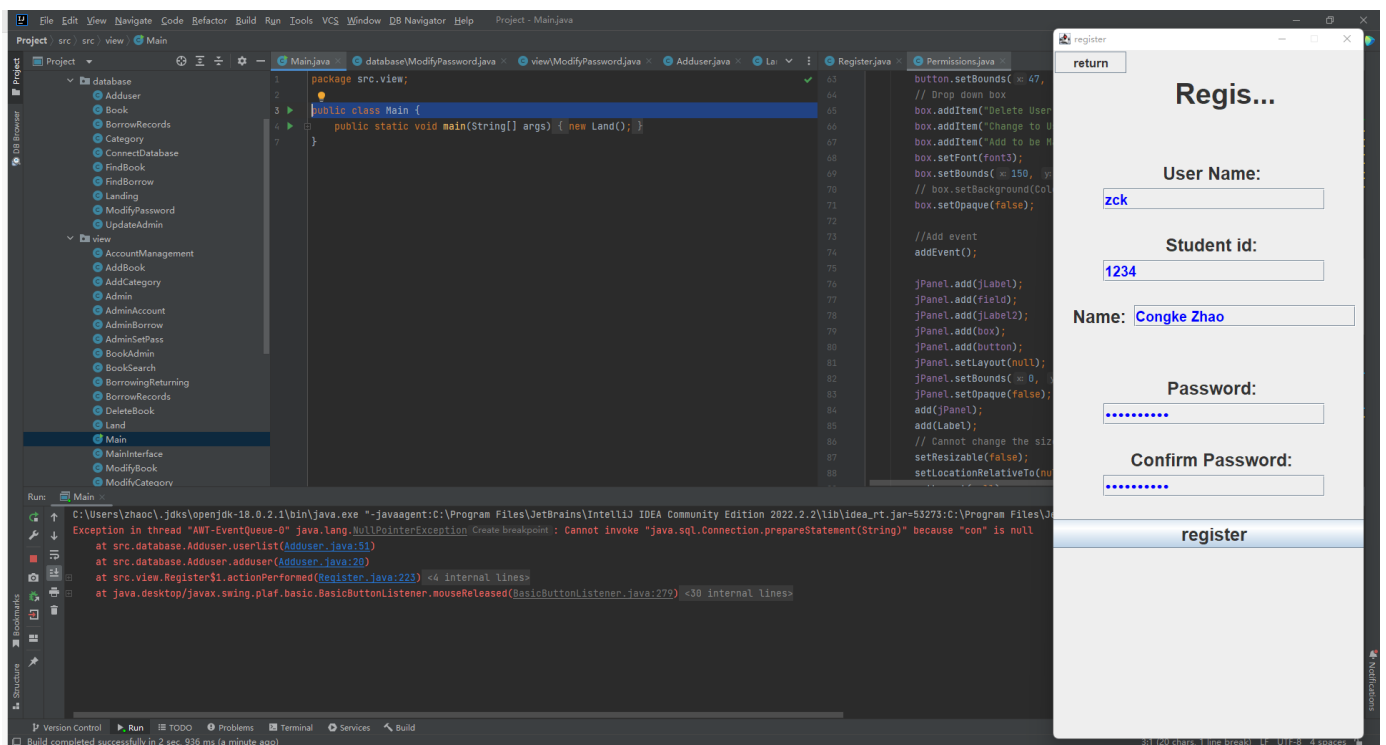


Project 7

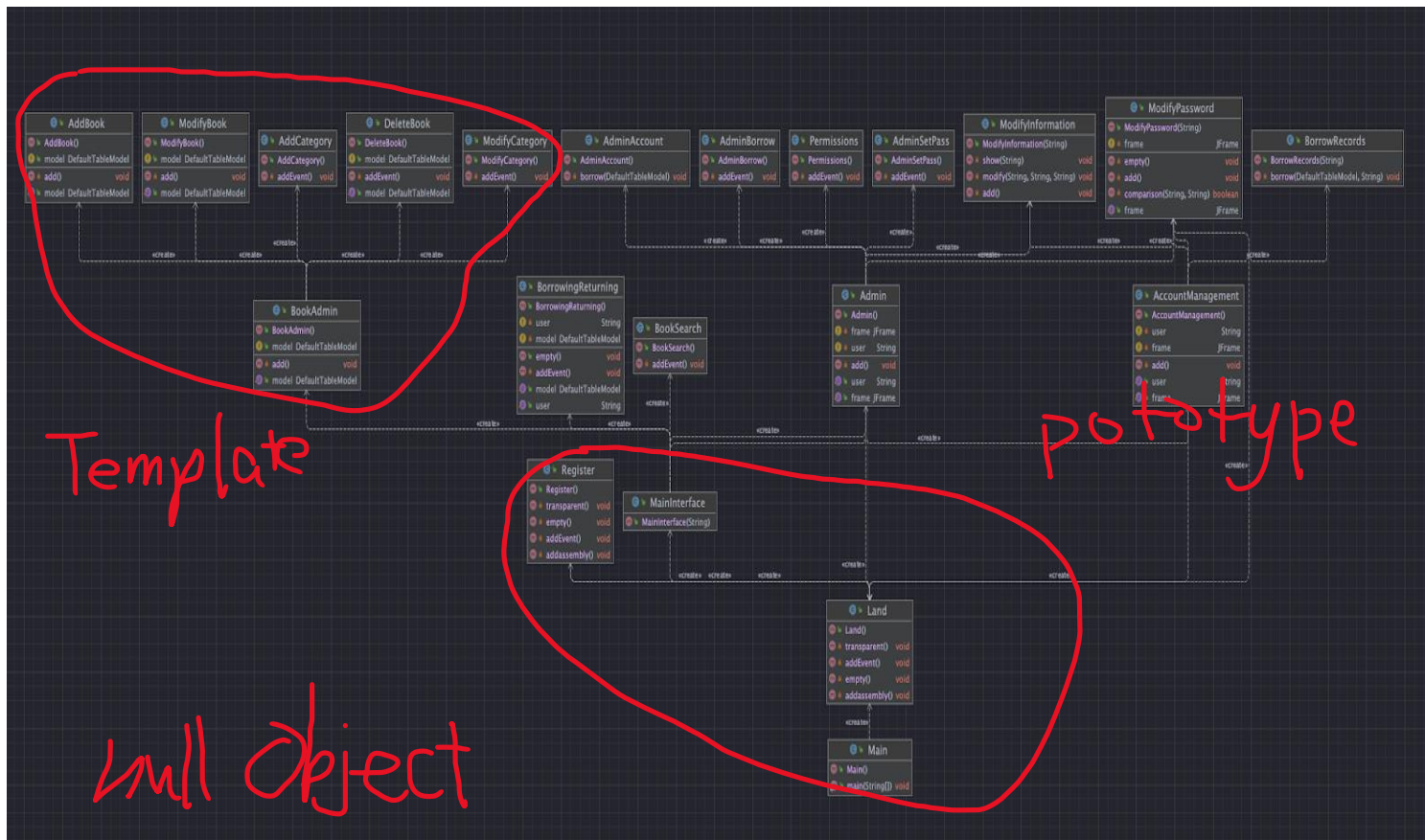
Book management system

Congke Zhao & Rui Zhang

Statement: Because Rui got Covid-19 we could not continue to make the project, our project was basically completed, but there were always bugs in the running phase. I cannot solve this problem by myself, so we actually do not finish the project. There is a big change between 5 and 6, but there do not have much change between 6 and 7. The big change is the design pattern, because we use java to do the interface directly, we thought this is more convenient. Therefore, we change a lot about the design pattern.



UML:



```
class FindBook {
    FindBook()
    allbook(DefaultTableModel) void
    findbookid(DefaultTableModel, int) void
    findauthor(DefaultTableModel, String) void
    findcategory(DefaultTableModel, String) void
    findbookname(DefaultTableModel, String) void
}
```

```
class BorrowRecords {
    BorrowRecords()
    Borrow(String, int, String) void
    ChangeBorrowState(int) void
    Return(String, int) void
    ChangeReturnState(int) void
    comparison(String, int) boolean
}
```

```
class UpdateAdmin {
    UpdateAdmin()
    sureuser(String) boolean
    updateuser(String) void
    updateadmin(String) void
    updatepass(String, String) void
    deleteuser(String) void
}
```

```
class Book {
    Book()
    modifybook(int, String, String, String, String, String) void
    addbook(String, String, String, String) void
    findcategory(JComboBox<String>) void
    deletebook(int) void
}
```

```
class FindBorrow {
    FindBorrow()
    allborrow(DefaultTableModel) void
    bookidborrow(DefaultTableModel, int) void
    userborrow(DefaultTableModel, String) void
}
```

```
class Adduser {
    Adduser()
    adduser(String, String, String, String) Boolean
    userlist() boolean
}
```

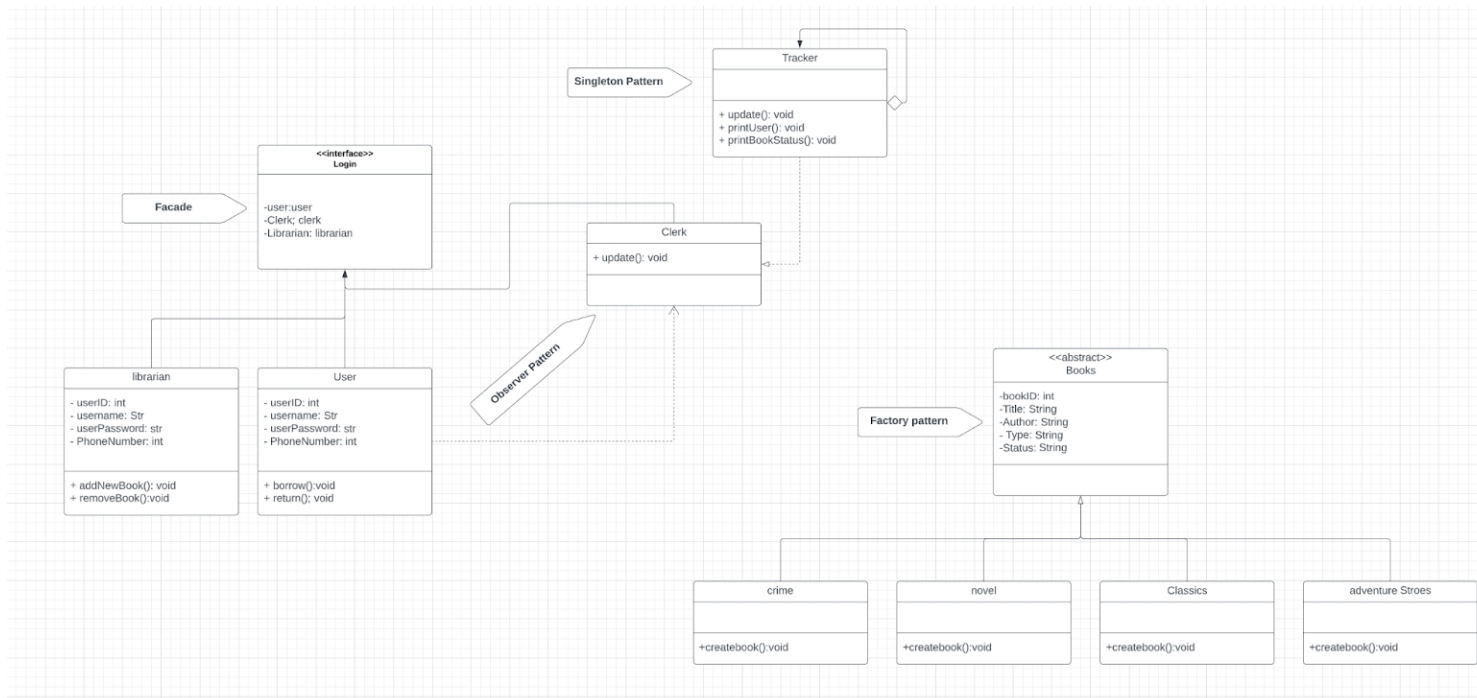
```
class Category {
    Category()
    addcategory(String) boolean
    setcategory(String, String) boolean
}
```

```
class Landing {
    Landing()
    test(String, String) Boolean
    sureadmin(String) boolean
}
```

```
class ModifyPassword {
    ModifyPassword()
    modifypass(String, String) void
}
```

```
class ConnectDatabase {
    ConnectDatabase()
    connectDB() Connection
}
```

Project 5 UML:



The most important change is the use of patterns. In Project 5, we did not intend to use Java directly as the interface, but in the later production process, we found that Java seems to be able to directly serve as the front-end language, so we changed the pattern idea and made a major change to the UML diagram.

Database connection is learned from:

<https://stackoverflow.com/questions/38660104/how-to-connect-intellij-with-local-mysql>

<https://www.javatpoint.com/example-to-connect-to-the-mysql-database>

<https://blog.csdn.net/kongsanjin/article/details/96425826>

Java swing (interface):

<https://www.geeksforgeeks.org/java-swing-jpanel-with-examples/>

<https://docs.oracle.com/javase/7/docs/api/javax/swing/JFrame.html>

database:





<https://www.w3schools.com/MySQL/default.asp>





<https://www.youtube.com/watch?v=JR1c8p2apJM>

In the process of making the project, we give priority to writing code. After completing part of the code, we analyze and reorganize the code framework and redesign it. Obviously, this makes the code more cluttered and increases the workload. And we didn't list the relationship of each class or interface in advance, which made the production process very complicated. Of course, the biggest problem is that they are not proficient in code learning and cannot use code flexibly.











Our code has been completed for a long time. The main problem is the database connection. Although our code is not very different from project6, we mainly complete the database connection problem, although we still haven't solved the problem in the end.

I tried to edit the database with Navicat, and tried various versions of IntelliJ and JDK files. Similarly, I used different versions of MySQL. And Tomcat was used as the local server to try to connect to the database, but failed.

 apache-tomcat-8.5.75-windows-x64.zip
 apache-tomcat-9.0.58-windows-x64.zip
 apache-tomcat-10.0.16-windows-x64....
 jdk1.8_windows.exe

 javaz02.zip	2022/12/8 21:12	WinRAR ZIP arch...	102,400 KB
 mysql-5.5.41-winx64.zip	2022/12/8 22:33	WinRAR ZIP arch...	218,393 KB
 mysql-5.5.62-winx64.msi	2022/12/8 23:07	Windows Install...	38,532 KB
 mysql-connector-j-8.0.31.zip	2022/12/8 22:44	WinRAR ZIP arch...	4,973 KB

名称	修改日期	类型	:
----	------	----	---

 .idea	2022/12/9 0:36	文件夹	
 Navicat for MySQL	2022/3/26 21:36	文件夹	
 out	2022/11/15 23:00	文件夹	
 src	2022/12/8 21:33	文件夹	
 key.txt	2012/12/22 21:12	Text 源文件	
 mysql_navicat.rar	2022/12/4 12:21	RAR 文件	
 MySQL_python-1.2.5-cp27-none-win_...	2017/6/20 3:11	WHL 文件	
 mysql-installer-community-8.0.31.0.msi	2022/12/4 12:06	Windows Install...	
 navicat161_mysql_en_x64.exe	2022/12/4 11:48	应用程序	
 Project.iml	2022/11/15 22:39	IML 文件	