课程编号：C0801207020

面向对象程序
课程设计报告

姓　　名	XXX	学　　号	XXX
班　　级	XXX	指　导　教　师	XXX
开　设　学　期	2016-2017 第二学期		
开　设　时　间	第 25 周——第 26 周		
报　告　日　期	2017.8.27		
评　定　成　绩		评　　定　　人	
		评　定　日　期	

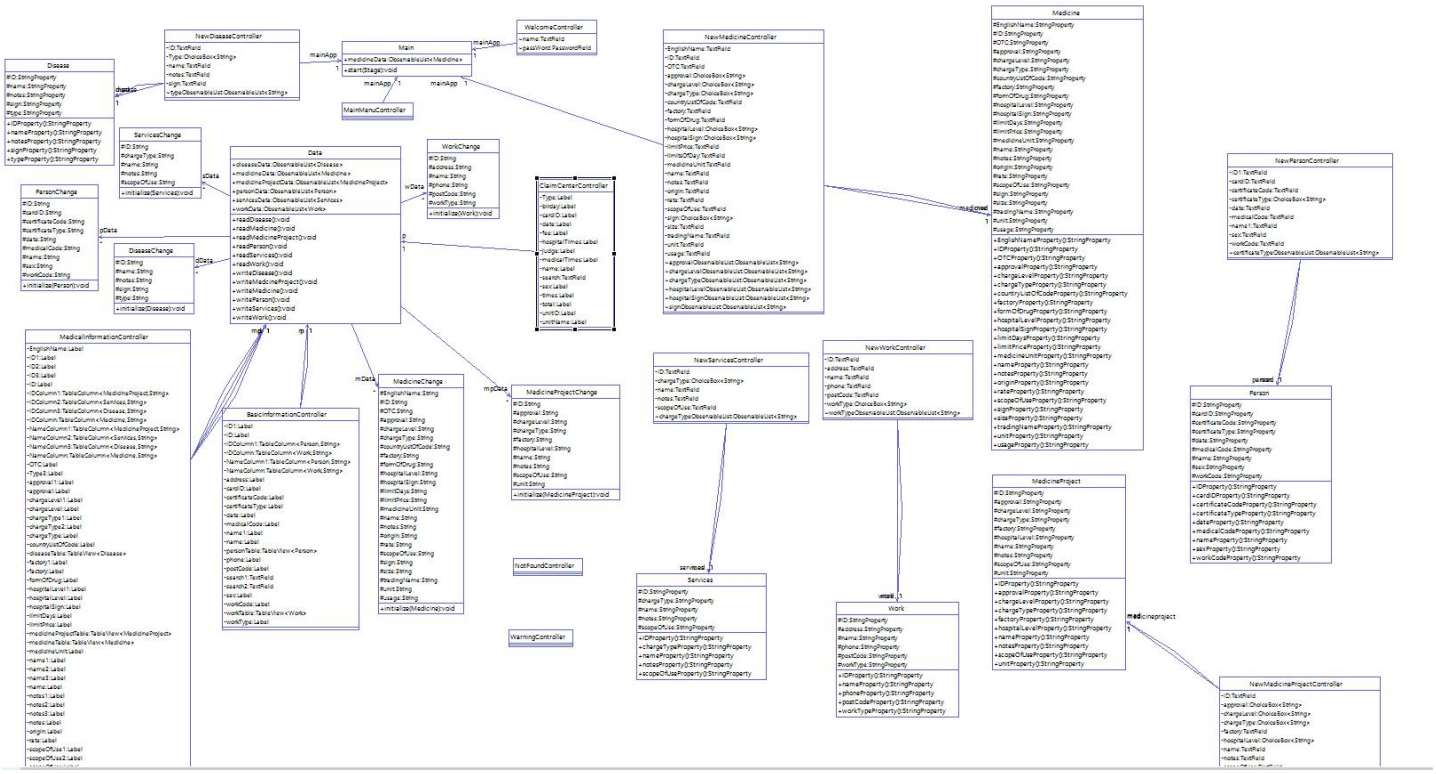
东北大学软件学院

1. 问题定义

我的医疗保险中心报销系统实现了需求规约中的公共业务需求以及医疗基本信息维护的需求，实现了其中的增加，删除，修改，查找的功能。还完成了中心报销的显示功能。

2. 系统设计

(1) 类图



(2) 存储设计

存储情况：

药品信息存储在“medicine.txt”中

医疗项目信息存储在“medicineProject.txt”中

服务设施项目信息存储在“services.txt”中

参保单位信息存储在“work.txt”中

个人信息存储在“person.txt”中

病种信息存储在“disease.txt” 中

使用文件存储。因为 JavaFX 中 TableView 的原因，需要使用 ObservableList 来显示表格信息, 而表格信息无法使用 ObservableList 来存储, 需要将信息转存到 ArrayList 中来存储。除此，由于要使表格能够随时根据操作显示变化，需要使用到 StringProperty 的数据类型，因此需要创建一个专门的类进行数据类型的转换。于是如此存储：

(a) 建立一个专门的类来进行数据类型的转换 e.g. medicine->medicineChange

(b) 建立一个 ObservableList 来操作信息，一个 ArrayList 来存储信息；

```
public ObservableList<Medicine> medicineData= FXCollections.observableArrayList();
public ArrayList<MedicineChange> mData=new ArrayList<> ();
```

（c）写入文件时，遍历操作时使用的ObservableList，将文件信息一个一个对象地写入ArrayList中，再写入对应存储此信息的文件中。

```
//writing data to file
    public void writeMedicine (){
        try {
            FileOutputStream outputStream=new FileOutputStream("medicine.txt");
            ObjectOutputStream objectOutputStream = new
ObjectOutputStream(outputStream);
        for (Medicine medicine: medicineData) {
        MedicineChange i =new MedicineChange ();
            i.initialize(medicine);
                mData.add(i);
            }
              objectOutputStream.writeObject(mData);
              outputStream.close();
              mData.clear();
        } catch (FileNotFoundException e) {
            e. printStackTrace ();
        } catch (IOException e) {
            e. printStackTrace();
        }
          }
```

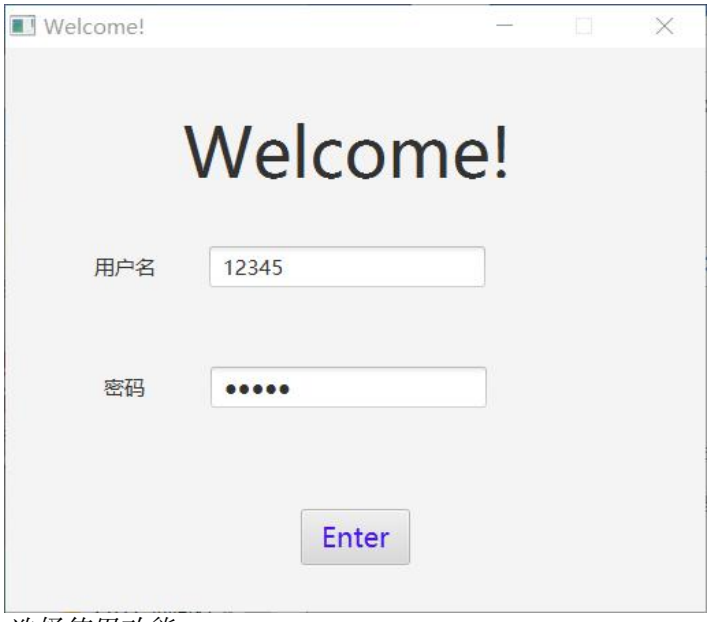
（d）读取文件时：创建一个 ArrayList，将文件中的信息读取到 ArrayList 中，然后同样通过遍历，将 ArrayList 中的信息读取出来存入 ObservableList 中，用于操作。

```
//get data from file
      public   void readMedicine(){
        FileInputStream reader;
        try{
            reader=new FileInputStream("medicine.txt");
          System.out.println("fileopenok");
          ObjectInputStream objectInputStream=new ObjectInputStream(reader);
        ArrayList<MedicineChange> array= (ArrayList<MedicineChange>)
objectInputStream.readObject();
         for (MedicineChange personProxy : array) {
            System.out.println("f");
```

```
            medicineData.add(personProxy.getMedicine());
        }
        reader.close();
        } catch (FileNotFoundException e) {
            e.printStackTrace();
        } catch (IOException e) {
            e.printStackTrace();
        } catch (ClassNotFoundException e) {
            e.printStackTrace();
    }
    }
```

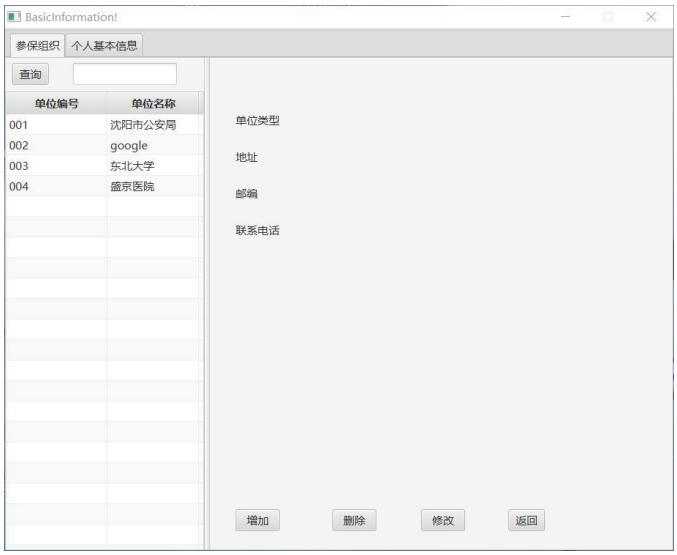
（3）界面设计

进入欢迎界面

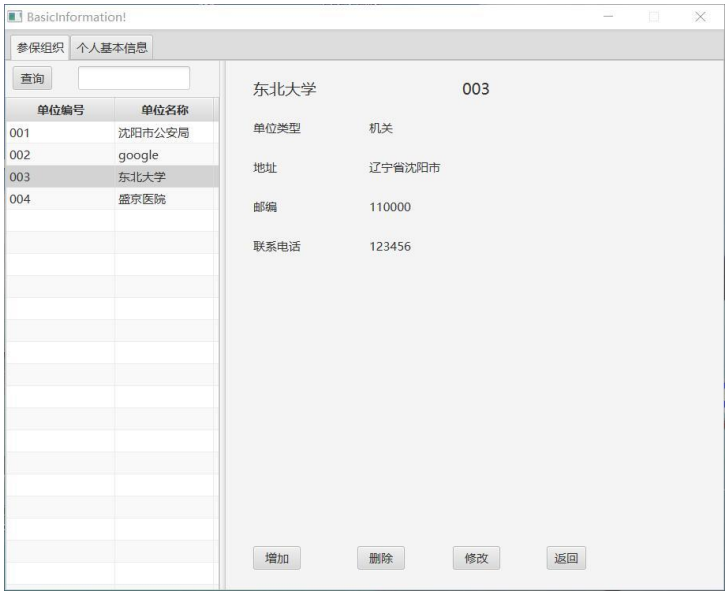


选择使用功能

公共业务界面



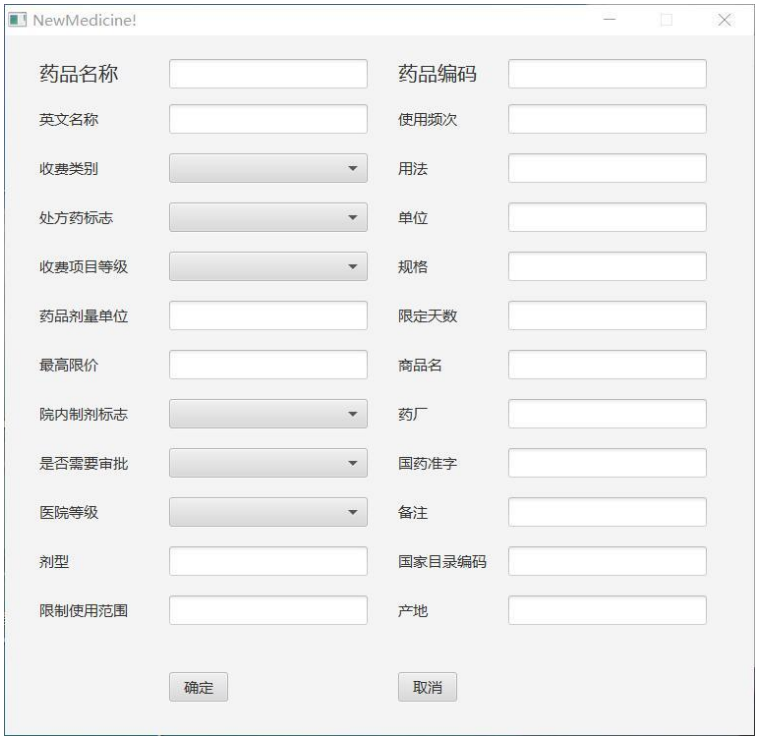
点击表格查看详细信息



医疗信息维护界面

输入 ID 或者姓名查找对应的信息

新增药品信息

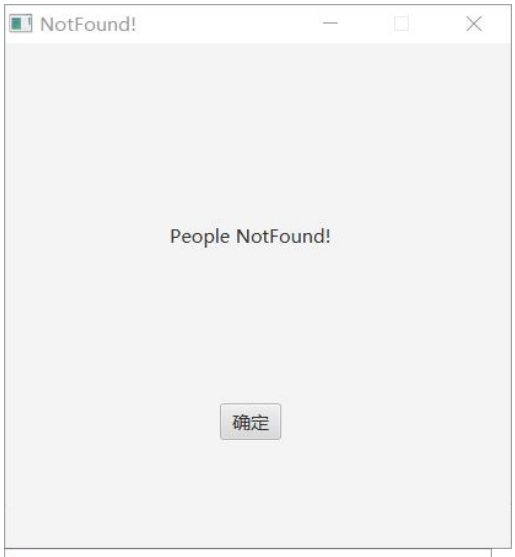


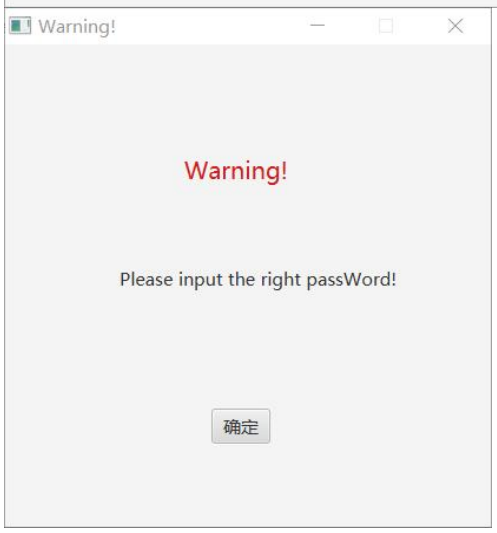
修改药品信息界面

报销界面


ClaimCenter!
查询
姓名
身份证号
性别
出生日期
医疗人员类别
单位编号
单位名称
本次住院次数
上次出院时间
上次出院诊断
本年中心报销累计
本年个人自费累计
本年医疗费用累计
费用
结算
录入处方明细
录入就诊信息
打印
返回

跳出异常界面


NotFound!
People NotFound!
确定


Warning!
Warning!
Please input the right passWord!
确定

3. 系统实现

(1) 增加、修改功能：为了程序的简洁清晰，增加以及修改功能使用的是同一个界面。每一次增加一条信息，对于增加功能，即传入一个空的对象供其修改，然后存入文件。而每次修改一条信息，对于修改功能，即传入目标修改的对象，并进行判断，如果信息未曾修改则不会删除原信息，如果信息被修改了，则会将原信息删除并保存新修改的信息。

```
private void handleOK(){
        Data m=new Data();
        m.readWork();
        try{
            if (med.getName()!=null){
                int i=0;
                for (Work mp:m.workData){
                    if (med.getName().equals(mp.getName())){
                        m.workData.remove(i);
                        m.writeWork();
                        break;
                    }
                    i++;
                }
            }
        }
```

此处进行了一次判断，如果信息未曾更改，则会保持原信息，否则，更改信息。

```
        catch(Exception e){

        }

        work.setName(new SimpleStringProperty(name.getText()));
        m.getWorkData().add(work);
        m.writeWork();
        NewWork.getStage().close();
        BasicInformation.showStage();
        }
```

设置 set 方法，使进行修改的时候，信息栏目能够显示原先的内容供其修改。

```
private Work med;
    public void setWork(Work med) {
        this.med=med;
        try{
```

```
            if (med.getName()!=null){
              name.setText(med.getName());

            }
        }catch(Exception e){

        }
    }
```

(2) 查找功能：查找时，进行一次遍历，如果输入的姓名或者 ID 和存储的信息有相同的，则显示出查找的对象的具体信息，否则，会跳出异常界面“查无此人”。

```
private void searchMedicineProject() {
        Data p = new Data();
        p.readMedicineProject();
        for (MedicineProject med : p.getMedicineProjectData()) {
            if (med.getName().equals(search2.getText()) ||
med.getID().equals(search2.getText())) {
                medicineProjectTable.getSelectionModel().selectedItemProperty()
                        .addListener((observable, oldValue, newValue) ->
showMedicineProjectDetails(newValue));
                showMedicineProjectDetails(med);
            }
        }
                }
```

(3) 删除功能：获得选择的目标，使用 remove 删除目标信息，并写入文件进行保存。

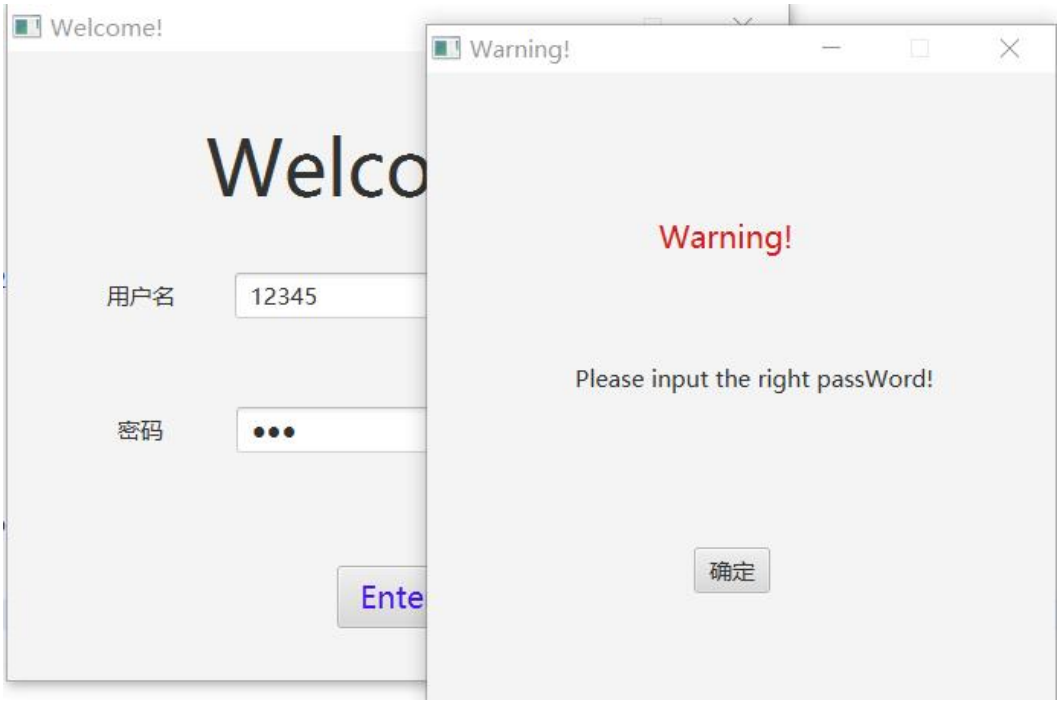
```
private void handleDeleteMedicineProject() {
        int selectedIndex =
medicineProjectTable.getSelectionModel().getSelectedIndex();
        medicineProjectTable.getItems().remove(selectedIndex);
        mp.writeMedcineProject();
}
```

(4) 登录功能：登录时进行一次判断，只有当密码和登录名相符时才可跳转到下一个界面，否则会跳出异常界面。

```
    private void welcome(){
        if(passWord.getText().equals("12345")&&name.getText().equals("12345")){
        Welcome.getStage().close();
        MainMenu.showStage();
        }else{
```

```
                Warning.showStage();
            }
                    }
    }
```

登录信息输入错误时会跳出异常界面



4. 遇到的问题及其解决方案

(1) 空指针问题：在打开，读取，写入文件时出现空指针问题。
解决方法：多次使用 System.out.println 方法在多处输出信息。如，在打开文件后，输出 System.out.println （“fileopen”）;如果输出成功仍有问题，则是读取的问题；同理，在一次又一次的输出中找到文件出现的问题。
效果：如果出错，则未输出，可以及其方便地找出出错的位置以供更正。

(2) Index 出错：在删除信息的时候，出现过删除不对应的信息、删除两条信息的问题。
解决方法：通过 print 方法输出对应的 index 来检测删除的具体信息栏目。然后多次进行检测，测试，找到其对应的 index 再进行删除。
效果：可以准确找到对应的 index，从而做出准确的判断。

5. 创新点

(1) 增加了登录功能，保证是工作人员才可登录对信息进行处理。
(2) 每一次进行修改的时候会显示上一次的信息，以供修改。
(3) 增加和修改使用了同一个界面，减少了程序的冗杂。

6. 总结

昨晚这次实验，我觉得对我们还是一次不小的挑战。对待一个程序，最先开始的不应该是埋头苦熬代码，而应该静下心来，先想好怎样去设计这个程序 ， 应该如何设计类与类

之间的关系，如何设计界面，如何才能使界面对用户友好。因此，最开始的设计部分，我认为，工程量不大，但确是最为重要的部分，在构思阶段，就想好如何设计这个程序，会为后续的程序实现阶段省下不少时间。而在代码编写阶段，我认为最为重要的不是一直埋头写程序，应该有所构思再开始进行代码的编写。完成一段先进行代码的运行可以为后续查找问题省下不少工程。同样，可以通过不停地进行 print 输出来检测代码的问题。第一次自己完成一个工程，需要在进行每一步的时候都要有所安排，后续才可以有条不紊地进行。

参考文献

例：

[1] 萨师煊，王珊.数据库系统概论（第三版）[M].北京: 高等教育出版社,2002,122-150.

[2] 侯捷.Java 编程思想[M].北京:机械工业出版社,2002,22-35.

教师评语或评价表格：

评价内容	具 体 要 求	分值	得分
平时表现	实践课过程中，无缺勤、迟到、早退现象，学习态度积极。	10	
实践验收	按实践要求合理设计并定义出系统定义、系统设计和系统实现，认真记录实验数据，原理及实验结果分析准确，归纳总结充分。程序结构清晰正确，程序代码书写规范，简洁。验收过程中能够正确回答问题。在解决问题的过程中有自己独到的见解，并能够有所创新。	60	
报告质量	实践报告格式规范，报告内容充实、正确，报告叙述逻辑严密，可准确反映出实践过程中出现的问题和解决方案，以及独立分析问题和解决问题的能力。	30	
总 分			