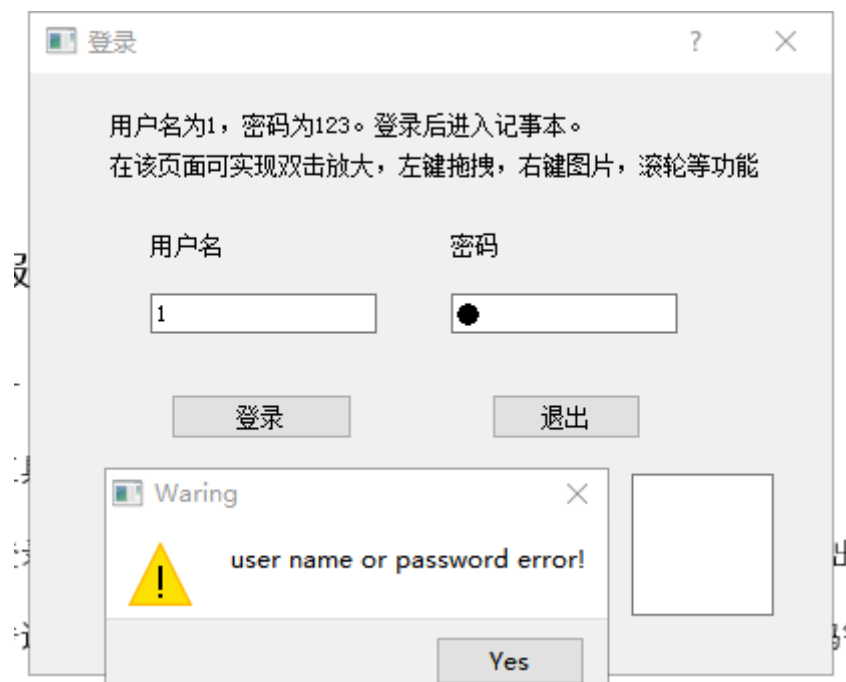


# 大作业报告 1613415 潘巧巧

## 1、设计：

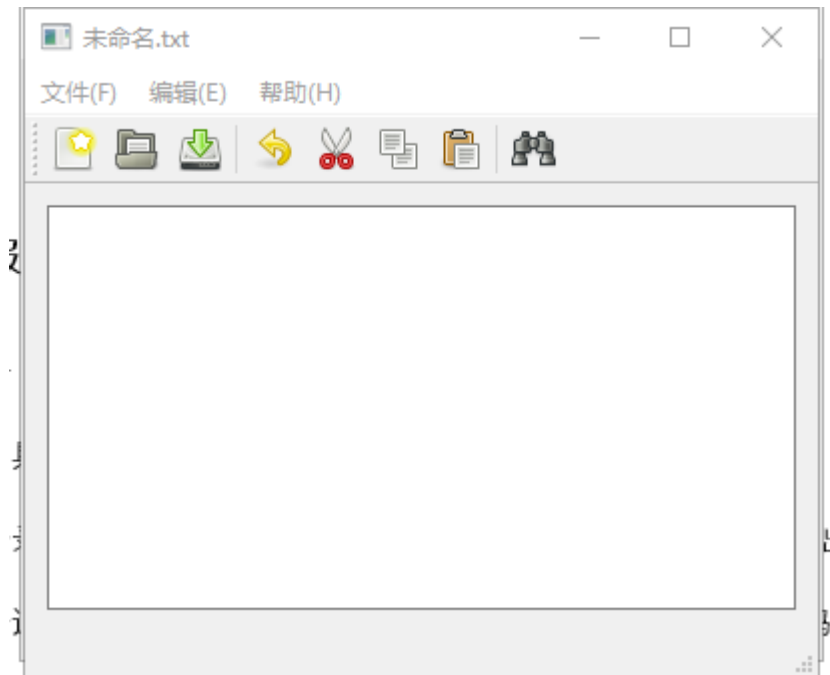
该工具包括两个界面。即【登录】和【记事本】。

①【**登录界面**】 通过输入用户名和密码进行登录，含有“登录”和“退出”按钮，点击退出或右上角的×直接退出无法进入记事本，用户名或密码错误会弹窗提示，点击弹窗的 yes 按钮后用户名和密码会被清空，用户可再次输入。本次设计默认用户名为 1，密码为 123（密码输入时显示圆点进行密码保护），也可通过 mysql、sqlsevers 等数据库系统保存账号密码。在登录界面运用了课上学习的关于信号的知识，可以在该界面上实现双击放大、鼠标左键拖拽对话框、鼠标右键显示图片、TextEdit 中通过滚轮放大/缩小字号等功能。



②【**记事本界面**】 用户名和密码正确输入后进入记事本界面。默认标题为“未命名”，包含新建、打开、保存、另存为、关闭、退出、查找、

复制、粘贴、剪切、撤销、版本信息共 12 项功能。其中前 6 项属于“文件”功能，最后一项属于“帮助”功能，其余项属于“编辑”功能。除“版本信息”外所有功能均有快捷键，其中“新建、打开、保存、撤销、剪切、复制、粘贴、查找”有快捷图标。



## 2、实现的思路和方法

① 登录界面。在 `logindlg.ui` 中实现基本元素的插入，在 `logindlg.cpp` 中的构造函数中 `ui->setupUi(this);` 以及设置密码为不可见。添加各项功能元素（如确认登录按钮）的私有槽 `private slots`，在槽函数 `on_loginbtn_clicked()` 中设置用户名和密码，并使得用户名或密码错误时弹窗，弹窗结束后情况用户名和密码以使用户重新输入。鼠标左右键点击、双击、滚轮功能按课件 PPT 实现。

② 记事本界面。

A. 在 `mainwindow.ui` 中实现基本元素插入，包括功能项（文件、编辑、帮助）及其各自细节项如快捷键（新建、打开、保存、另存为、关闭、退出、查找、复制、粘贴、剪切、撤销、版本信息共 12 项功能）、快捷按钮及图标（新建、打开、保存、撤销、剪切、复制、粘贴、查找）、文本编辑框 `TextEdit` 等，并设置他们的相关信息（如文本编辑框默认全屏、建立一个 `resources` 保存各项功能的图标）等。

B. 将各项功能“转到槽”，在 `mainwindow.cpp` 中各自的槽函数中实现功能。

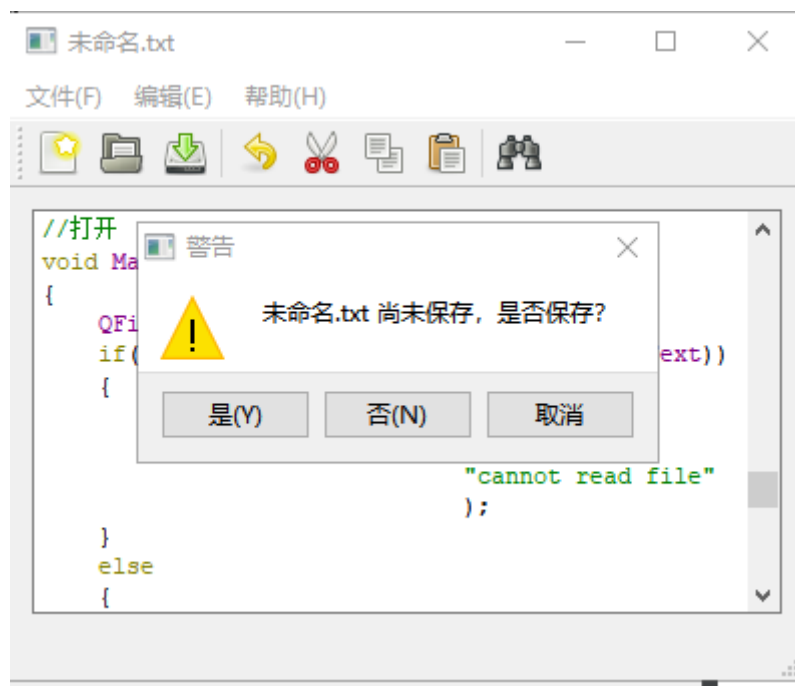
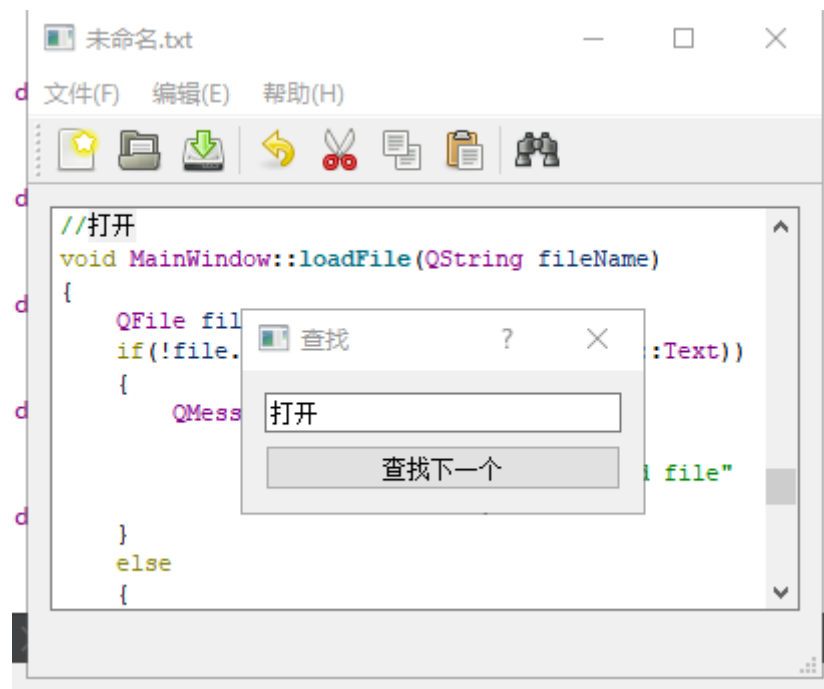
a) **【新建文档】** 槽函数 `on_action_N_triggered()` 执行函数 `newFile()`。  
`newFile()` 中首先通过函数 `maybeSave()` 判断文档是否需要保存，如果已经保存完了，则新建文档，并进行初始化。  
`bool maybeSave()` 函数用于判断文档是否需要保存。如果文档没有被修改过则不需要保存，可直接返回 1。如果文档被修改过了，则需要调用函数 `save()` 进行保存。

- b) **【保存文档】** 槽函数 `on_action_S_triggered()` 执行函数 `save()`。在 `mainwindow.h` 中定义了 `bool` 型变量 `isUntitled`, 返回 1 表示文件没有保存过, 返回 0 表示文件已经被保存过了, 以及 `QString` 类型变量 `curFile` 用于保存当前文件的路径。`save()` 中通过 `if` 语句进行分支, 如果文件没有保存过 (即 `isUntitled` 为 1), 则执行另存为函数 `saveAs()`, 如果文件被保存过了 (即 `isUntitled` 为 0), 则执行存储行为, 即执行函数 `saveFile(const QString &fileName)`。具体过程见代码。
- c) **【另存为】** 槽函数 `on_action_triggered()` 执行函数 `saveAs()`。函数中判断如果文档为空则不执行, 否则执行存储行为, 即执行函数 `saveFile(const QString &fileName)`。
- d) **【打开】** 槽函数 `on_action_O_triggered()` 首先通过函数 `maybeSave()` 判断当前文档是否需要保存, 其次获取目标文档的名字并判断要打开的目标文档是否合法, 均成功通过后执行函数 `loadFile(fileName)` 进行打开操作, 该函数中通过调用 `QTextStream`、`QFile` 等类的库函数执行打开操作。具体过程见代码。
- e) **【关闭】** 槽函数 `on_action_C_2_triggered()` 首先通过函数 `maybeSave()` 判断当前文档是否需要保存, 通过后执行语句 `ui->textEdit->setVisible(false)` 执行关闭当前文档的操作。

- f) **【退出】** 槽函数 `on_action_2_triggered()` 首先通过函数 `maybeSave()` 判断当前文档是否需要保存，通过后执行语句 `qApp->closeAllWindows()` 执行关闭当前文档的操作。
- g) **【查找】** 槽函数 `on_action_F_triggered()` 执行语句 `findDlg->show();`  
在查找的相关函数 `showFindText()` 中，通过函数 `find(str, QTextDocument::FindBackward)` 进行查找，查找顺序为从鼠标光标定位处向文件头查找。
- h) **【复制、粘贴、剪切、撤销】** Qt 提供了这些功能的函数实现和快捷键操作，所以只需连接信号与槽即可。  
例如复制操作只需：  

```
void MainWindow::on_action_C_triggered()
{
    ui->textEdit->copy();
}
```
- i) **【版本说明】** 在槽函数 `on_action_6_triggered()` 中使用类 `QMessageBox` 的 `message` 函数弹窗说明版本信息。

### 3、相关截图



#### 4、遇到的问题及解决方法

① 对槽函数和信号的概念混淆

② 不会处理复制、粘贴、剪切等功能。

解决方案：发现 QT 已经封装好相关函数和类，通过网上学习学会使用 `copy()` 等函数。

③ 登录界面中实现“按下鼠标右键使用自定义的图片作为鼠标指针”失败，无法显示图片。

解决方案：通过查阅资料发现图片位置错误，应该将图片放在工程文件