**设计文档**

**院（系）：** **信息工程学院**

**专业班级：** **17计算机软件与理论**

**学生姓名：** **张鑫涛**

**学生学号：**　 **1120170705**

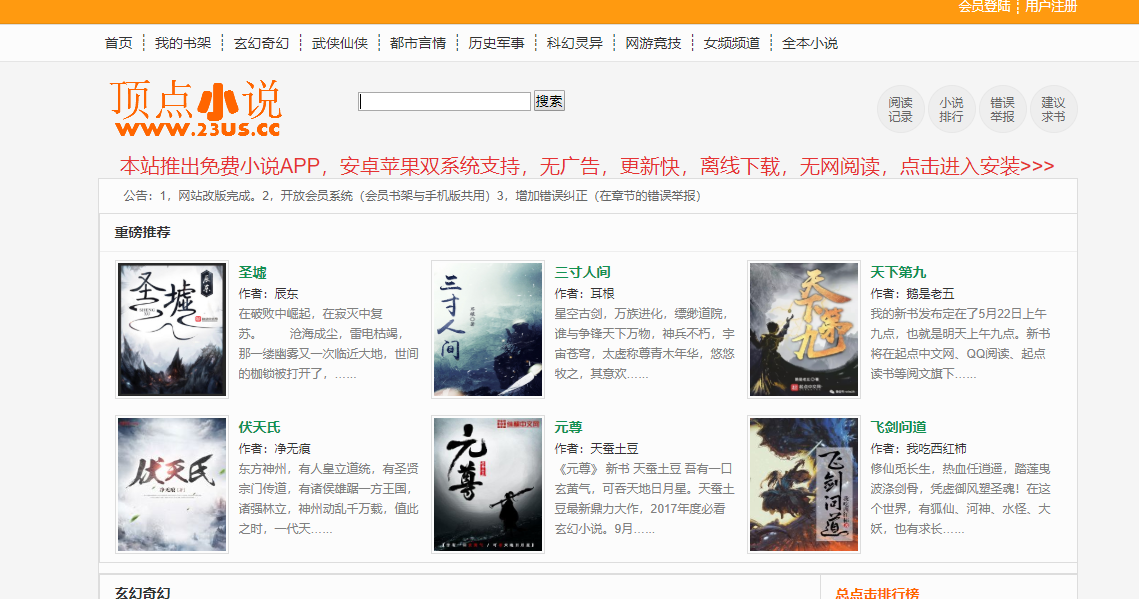
**任课老师：** **毛波**

**完成时间**：  **2018年6月23日**

对于本项目的小说下载软件，设计思路是这样的：

1. 使用 selenium + PhantomJS 模拟浏览器访问顶点小说网首页<https://www.23us.cc/>；
2. 用户输入下载的书名，然后找到首页搜索框的，输入书名，再模拟点击回车进行搜索；
3. 浏览器切换到搜索页界面，并且返回搜索结果页第一页 JS 渲染后的源码；
4. 将搜索得到的书籍信息以列表形式返回给用户，内容包括书名，作者，风格（该网站暂时只提供这些信息）
5. 用户根据软件返回的信息再次输入想要下载的小说，这次是精确匹配；
6. 根据用户输入的书名匹配到小说对应的URL链接；
7. 根据上一步得到的链接，进入小说章节页面，然后获取小说全部章节的URL，然后访问每个章节，使用xpath得到每一章节的正文（一开始用的是re正则匹配，因为速度太慢，后改用xpath）；
8. 将获取到的正文保存下来，存放到txt文件中，完成下载。
9. 将以上功能用一个简单的GUI包装一下，打包成exe文件。

下面是具体实现思路：

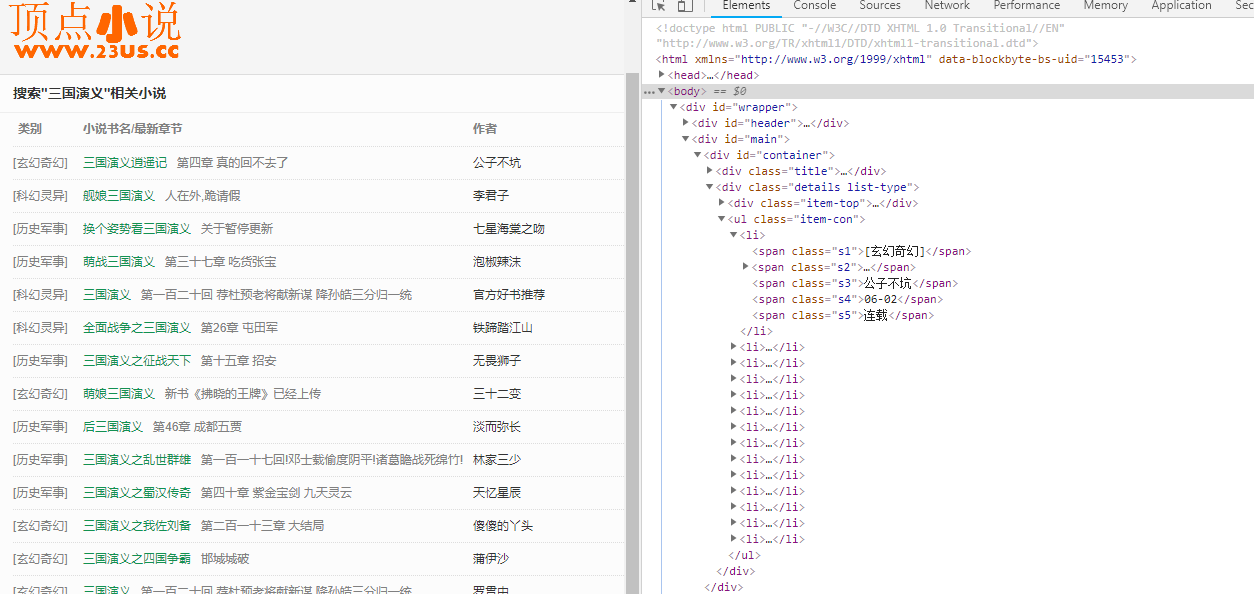








在顶点网搜索框中输入想要搜索的小说后，会跳转到一个结果页面，观察得到每一个跳转网页的网址前面一部分都是一样的，只有最后不同，因此可以轻松实现1、2两点。



按F12进入开发者模式，观察网页源码会发现，所有小说的类别、书名、最新章节以及作者等内容都放在class=‘item-con’节点的子节点li中，有多少搜索结果就有多少li节点，因此在用requests.get函数获得网页源码后，再通过xpath定位，可以获得这些内容，第3、4点可以实现。



小说的URL链接地址藏在每个li节点的span[2]中，通过用户第二次输入的小说名与得到的书名进行精确匹配，取出其中的链接，进入小说的章节页面，第5、6两点实现。



同样的，观察网页源码发现，每一章节的链接，章节名都放在id=main，class=chapterlist下的dd节点中，因此通过findall（dd）方法可以获得这些信息。最后，再进入每一章节，通过xpath获得每一章节的小说内容，保存下来。



最后，使用eric6和Qt designer软件实现界面开发，最后用pyinstaller方法打包成exe文件，这里不进行详述。

代码分为两部分，一个是定义了一个大类，里面定义了一些函数分别实现搜索结果页数据、小说章节页数据、章节内容页数据等功能，另外一个则是实例化该类，并具体实现下载的功能。