自动下载

需求说明书

|  |  |  |
| --- | --- | --- |
| 文件状态：  [ ] 草稿  [ ] 正式发布  [√] 正在修改 | 文件标识： | 001 |
| 当前版本： | 1.0 |
| 作 者： | 毛宏 |
| 完成日期： |  |

目录

[1. 文档介绍 3](#_Toc507524348)

[2. 业务需求 3](#_Toc507524349)

[3. 用户需求 3](#_Toc507524350)

[3.1软件功能 3](#_Toc507524351)

[3.2操作方式 3](#_Toc507524352)

[3.4使用范围 3](#_Toc507524353)

[3.5工作流程 3](#_Toc507524354)

[3.6用户对软件应用的展望 4](#_Toc507524355)

[4. 系统需求 4](#_Toc507524356)

[4.1功能需求 4](#_Toc507524357)

[4.2性能需求 4](#_Toc507524358)

[4.3环境需求 4](#_Toc507524359)

[4.2技术需求+ 5](#_Toc507524360)

1. 文档介绍

本文主要对网页数据自动下载做详细需求分析，本文包括对自动下载的功能需求详述，非功能性需求，对系统的性能，环境，技术以及可行性进行分析。以及业务的用例图、流程图，系统用例图、流程图，包括技术的调研。

1. 业务需求

患者病历的数据迁移比较麻烦，随着患者数量的增加工作量也会增加，所以设计一个辅助工具，用于自动下载所有患者的所有病历数据。减少工作量，便于使用。

1. 用户需求

## 3.1软件功能

帮助工作人员快速把所有患者的病历数据下载下来。

## 3.2操作方式

没有特殊操作，运行就可直接自动下载数据

## 3.4使用范围

暂时只针对特指网站的特定数据进行下载。

## 3.5工作流程

用户通过傻瓜式的操作一键下载数据，然后可在工作目录文件夹下打开下载的数据。

## 3.6用户对软件应用的展望

可复用性，希望可以通过简单的修改就可以下载更多网站的不同种类的数据。

1. 系统需求

## 4.1功能需求

|  |  |
| --- | --- |
| **功能类别** | **功能明细** |
| 模拟登陆 | 模拟登陆网站，输入用户名和密码进行登录 |
| 模拟验证码 | 模拟验证码出现的情况进行解决 |
| 选择患者列表 | 登录成功后选择患者列表 |
| 患者选择 | 选择患者列表的每一个患者 |
| 患者病历存储 | 存储每个患者的所有病历 |
| 病历数据存储 | 将病历中的数据以可视化的形式存储 |
| 病历数据转存 | 将病历数据从数据库中取出存储到另一个数据库中 |
| 病历数据格式转换 | 将取出的数据处理成目标数据库需要的格式 |

## 4.2性能需求

能够在有验证码的情况下顺利完成模拟登录，登录后能够成功获取所有患者信息，并能够将所有患者的所有病历信息数据全部下载并保存下来，保存后的数据能简单的让人进行使用。

## 4.3环境需求

## 4.2技术需求

### Cookie

**Cookie出现的主要目的是为了解决HTTP无状态的问题，是HTTP拓展协议。就好像之前所说的问题，用来保存用户的状态信息，比如说用户名，密码信息，购物车信息等等。Cookie通过浏览器来在客户端管理，不同浏览器之间互相独立。也有说法cookie分为内存和硬盘两种储存方式，硬盘上的cookie是浏览器共享的。**

**Cookie主要有名字、值、缓存时间、路径和域：   
名字和域就是键值对没什么好说的。   
Cookie有缓存时间限制，如果超过时限会消失，如果cookie没有设置缓存时间，则时间为本次会话，浏览器窗口关闭后就消失了。   
路径和域构成了Cookie的作用范围，传送给服务器的cooki必须是作用范围大于请求资源的域才传送。   
Cookie有大小限制，最大4k。另外浏览器通常也也限制每个站点cookie的数量和cookie的总量。**

**cookie产生的方式：正常来说是通过浏览器或者说http交互的，但是实际上也可以通过程序语言自己构建cookie。**

### Token

**有了Cookie和Session的知识，那token就比较好解释了，Token我理解的话就是Session ID的一种。**

**Cookie和Sesssion其实没有什么可以比较的东西，只能说通过Cookie，再更上一层次的设计上实现了为无状态的HTTP协议的Session机制。**

**token并不是cookie的一种，可以是cookie验证后的产物，但是token可以以cookie的形式储存，传递的时候也可以用cookie的形式来传送，也可以写入url里传送，也就时所谓的URL重写技术。另外Cookie通常是浏览器维护的，但是token不一定，可以以其他终端的形式维护。**

**唯一需要说明的是，为何有了Cookie还需要实现Session而不是只用Cookie：**

**一方面Cookie本身有一定的限制：**

**首先，cookie并不是万能的。cookie有很多限制，比如说不能跨域访问。跨域访问还有很多其他的问题需要考虑，这里不赘述了。此外token还是一种常用的防止CSRF的手段。**

**其次，cookie是明文传送的。除非你使用https的协议，http协议是明文传递的。明文传递cookie很容易被盗取，但是token就不会有这么明显的意义。**

**此外，即使不考虑安全的问题。每次都通过cookie来验证用户状态也是一个麻烦的问题。耗费了大量的资源且不必要。服务器验证之后给用户一个有一定时限的token，这样用户可以通过token免去验证的烦恼。**

**最后，Cookie本身时可以删除的。**

**另一方面Session并不只是实现了记录的内容。更多的时候用于标示和跟踪用户的作用。**

### 表单提交

**所有系统的登录都是表单的post提交，所以只需要在html中模拟写出该系统的表单需要提交的参数并赋值，提交表单即可跳转。**

### 模拟登陆

**发送Http请求实现模拟登录**

**模拟登陆的原理很简单，就是发送一个Http请求服务器获得响应，然后客户端获取到cookie即可实现模拟登陆**

### 模拟Http发送Post或Get请求

在post的时候有时也用的到cookie，像登录163发邮件时候就需要发送cookie，所以在外部一个cookie属性随时保存CookieContainer cookie = new CookieContainer();

**注意：**有时候请求会重定向，但我们就需要从重定向url获取东西，像QQ登录成功后获取sid，但上面的会自动根据重定向地址跳转。我们可以用：

request.AllowAutoRedirect = false;设置重定向禁用，你就可以从headers的Location属性中获取重定向地址

### Json解析

https://www.crifan.com/summary\_what\_is\_json\_and\_how\_to\_process\_json\_string/

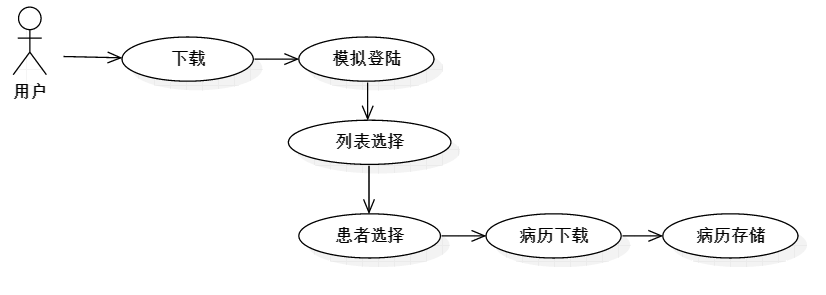
抓包

1. 绘制用例图，活动图，类图，时序图（完成）

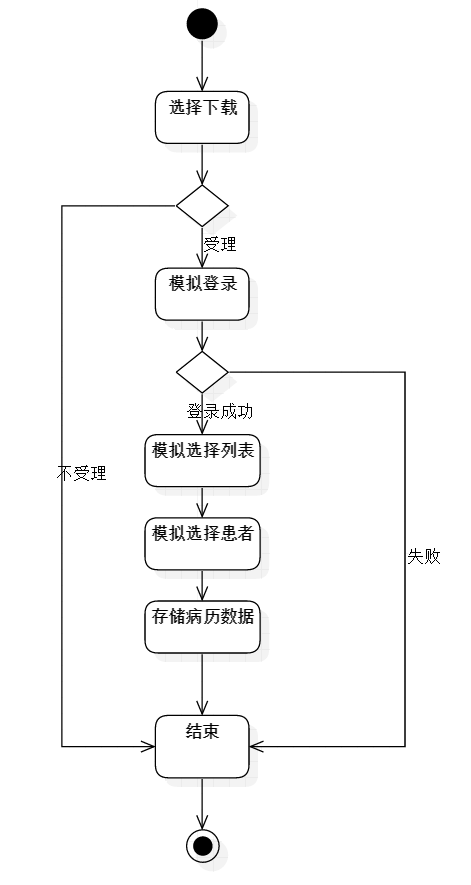
2、技术调研：Http请求，URL，cokkie

1. 系统设计

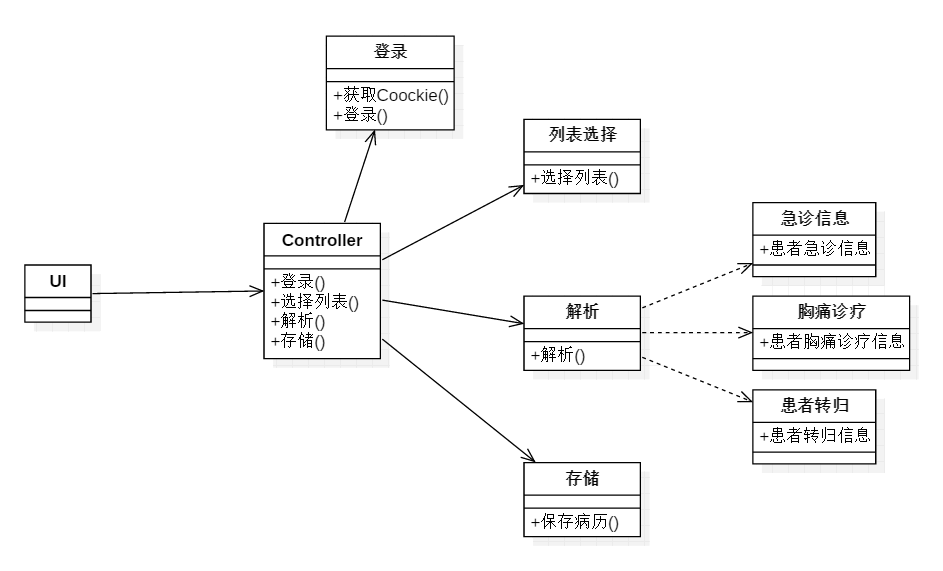
### 用例图设计



### 活动图设计



### 类图设计



### 时序图设计

