# 软件构造基础 选题

## 成员与分工（初期，暂定）

软件后台代码编写：

邢雄 2018320110266

杨翕然 2018302110276

李星鹏 201830211024

张衡2018302110176

软件前端代码编写：

杨帆 2018302110278

邱壮 2018302110287

相关文档书写与UML图绘制：

预计平摊，做到每个成员都要书写部分

## 选题背景

在日常生活中游览网页时，我们经常会遇到自己喜欢的图片或视频等资源，想要把它们保存到本地反复欣赏。然而，在实际操作时普通用户容易遇到各种难以处理的问题：  
（1）一般的下载操作需要依次逐个手动进行游览、选中、点击、保存、确认等机械重复的操作，当下载量巨大时十分繁琐耗时。  
（2）大多数网站并没有提供直接的资源下载按钮，用户往往不知道如何下载。  
（3）许多网站会对下载做出限制，如需要用户注册登录甚至成为会员才能正常下载，导致资源在眼前却无法下载的状况。  
（4）少数网站甚至禁止用户的常规下载操作，如禁止右键点击、禁止常规下载菜单的弹出等等。  
 在这种选题背景下，我们小组计划基于C#语言、运用网络爬虫和html文本原理、结合文件管理和文件筛选过滤技术，制作一个带有基本图形界面的网络资源爬虫程序。

## 功能预估

核心功能：

通过网络爬虫技术，用户可以使用本产品输入对应网页的网址以及需要下载的图片的格式或者其他的约束限制（图片，视频的大小，主题等等），从而自动批量下载获取该网页上的对应资源文件。

其他功能：

软件提供友好易操作的用户交互页面

用户可以预览爬取的资源文件

用户可以查看文件爬取的情况

用户可以批量爬取某原网址下一定数量子网页的特定资源

用户批量爬取某一主题对应的相关资源

## 技术难点预估

后台：

1.不同的网站对图片资源的存储方式具有差异性，同时由于图片资源类型多样(例: png, jpg, gif等等)，同一网站对图片资源的处理也可能具有差异性。在爬取资源时可能会涉及到多种爬取资源的策略。  
2.如何保证爬取图片资源时，图片没有损失，保证资源的完整性。  
3.在爬取资源文件时，需要考虑网站的反爬虫机制，以及图片资源被加密等无法下载的情况  
4.在获取图片资源时，会涉及到大量网页的访问以及数据的下载，我们需要保证爬取图片资源的时间不能太长，保证爬取的性能。(可以考虑并行编程)  
5.根据用户的需求，我们需要对爬取下来的资源进行分类 (例如: 根据文件类型，图片大小，图片时间等等)，并且对图片资源进行筛选，过滤掉部分无效的图片(例如:网站图标等)  
6.获取资源的详细信息，为用户显示这些详细信息(例如: 根据文件类型，图片大小，图片生成时间等等)  
7.下载图片前，为用户提供图片预览功能，用户可以根据预览效果选择需要下载的图片进行下载  
8.根据资源的大小，下载时间等，为用户显示当前图片的下载进度以及预估剩余时间  
9.计算资源的实际下载时间，下载结束之后显示实际下载时间  
10.资源下载失败时，先再次尝试下载，若多次下载仍然未成功，需要提示用户并显示失败原因，用户点击可再次尝试进行下载。

前端：

1.在原有系统中，窗口之间会有数据上的交互，如：保存、数据传递。  
2.在原有系统中，窗口之间会有行为上的交互，如：当用户在预览图片这个窗口中对图片做出一些操作时，原窗口如何响应一些相关操作。  
3.如何使系统用所有窗口操作风格统一。  
4.当用户不按照窗口的打开顺序关闭窗口时，产生的数据混乱问题，例如：用户有窗口A打开窗口B，再在窗口B中打开窗口C，此时用户关闭窗口A。  
5.控件分布合理与美感问题。  
6.界面美观问题。