

# 抢课脚本 – 可行性分析与需求分析文档

姓名：杨浩然

学号：161220156

## 1. 引言

### 1.1 报告撰写目的

抢课脚本是针对南京大学本科生希望选到自己所需课程的需求而设计开发的，该脚本可以完成模拟登录教务网、模拟刷新选课界面、模拟手动选择课程。

此可行性研究报告，是为了在尽可能短的时间内用最小的代价来确定抢课脚本是否可以实现，从而确定具体开发流程。

### 1.2 项目背景

“抢课”现象在当今大部分高校中是十分普遍的现象，其产生的原因一般是学校教学资源有限而不能满足每位同学对课程需求。很多同学为了选上自己想上的课，通宵达旦地守在电脑旁，不断刷新着教务网选课的网页，等待着上课名额的突然出现。这一现象也促成了同学们进行课程交易的行为，即拿自己不急需的课程去跟其同学换取自己急需的课程。现在教务处针对这种行为采取了缓放课的措施，即课程退选后不会马上出现在可选课程中，而是在退课后的一段时间内的任意一个时间点随机放出。抢课以及课程交易很多都是在半夜进行的，浪费了同学们太多的休息时间和精力，因此用脚本进行自动抢课对有需求的同学来讲是很有必要的。

### 1.3 开发目标

使用 Python 语言，设计并实现一款针对南京大学教务网选课系统的自动抢课脚本。可进行抢课的时段为开学时期的课程补选阶段（因为期末的选课阶段是靠随机抽签来决定是否选中课程的，所以不存在抢课的问题），抢课的种类应包含学校所开的除专业必修的其他所有课程：通识研讨课、公选课、通选课（思政军事、计算机、数学、英语）、跨专业选修课、体育。脚本可以同时多种课程进行补选（每种课程仅限一门）。每个在校的本科生都能使用该脚本。

### 1.4 开发内容简介

第一步：模拟登录。该脚本的功能实质上是模拟同学们的操作进行课程的补选，因此，第一步就是要登录到学校的教务系统；

第二步：选择需要进行课程补选的课程种类；

第三步：填写每个种类中需要进行补选的课程编号；

第四步：获取课程页面的源代码，获取课程 ID；

第五步：伪造成浏览器，向教务网服务器传输数据（选课的实质就是从本地传送选课信息到教务网服务器，教务网服务器接收信息后对学生信息进行相对应的修改）；

第六步：刷新页面，重复第四步直到选课成功或出现异常中断为止。

### 1.5 工作基础介绍

大二上学期对 python 基础进行过较为系统的学习，因此开发者具有较好的 python 编程基础，利用 Python 做过爬虫，爬取过网上的一些资料，对本项目中的所需的 requests 库的一些基本功能比较熟悉。抢课脚本的构思在上期期末就开始，因此想法比较成熟。现在补选课程即将结束，但是上学期期末完成了模拟登录和访问课程页面两个功能，且开学选课时对所需要的数据进行了提前收集，准备较为充分，即使选课阶段结束后仍可保证项目的正常开发。

## 2. 可行性分析

## 2.1 系统基本要求

(1) 系统功能概述：该系统主要目的就是帮助学生高效率地抢课，实现该功能有三个步骤——模拟登录、模拟刷新、模拟选课。

模拟登录：从教务网获得验证码和 Cookie，再输入学号密码，完成模拟登录后才可以进行后续其他的步骤。

模拟刷新：模拟手动选课时的不单刷新页面的操作，以降低被服务器拒绝访问的概率。

模拟选课：构造选课信息，向服务器不断发出选课请求。

(2) 运行环境：一台可以连接校园网的 PC，且该 PC 上应装有 Python 3 的运行环境，该脚本还依赖于三个第三方库——requests、PIL 和 lxml（即 python 环境中应该安装有这三个第三方库）。

## 2.2 系统开发要求

(1) 开发周期：第 1 周收集各种数据（Cookie、课程 ID、选课信息等），第 2-3 周代码及文档编写，第四周测试。

(2) 开发环境搭建：安装 Python 3 环境，安装集成开发环境 Pycharm，安装第三方库 requests 及 PIL。

(3) 费用开支与效益分析：由于不准备将该脚本放到云端不断运行，因此不会产生服务器租用费用，访问教务网不需要付费，由于该脚本为个人使用，所以开发和为人员均为同一人且人数为 1，且开发该脚本不存在任何盈利性的目的。综上，费用开支为 0，收益也为 0。

## 2.3 其他各因素的可行性分析

(1) 可用性方面：该脚本最早期的简单 demo 版本成功帮助开发者抢到了一门课程，且该脚本主要采用面向过程的编程方法，因此开发者可以成为很好的使用者，可用性较佳；

(2) 法律道德方面：我国目前没有法律来规范高校选课的具体行为，学校教务处也没有明确规定禁止使用第三方辅助工具进行抢课。虽然有些同学会认为不公平，但是该脚本人人都是可以开发，所以开发者认为开发脚本无可厚非；

(3) 开发工具及所需数据方面：脚本需要的一切参数和参数值都能利用谷歌浏览器开发者工具找到，且该脚本所使用的 requests 第三方库也有着完备的文档可供参考，以及该项目所需的多进程编程知识比较简单，所以技术细节实现上不存在大的困难；

(4) 网络方面：教务网对访问次数及数据提交次数（即抢课申请次数）没有限制，意思是说，你大量快速访问教务网时，教务网并不会封掉用户的 IP 地址，但是以防万一在程序中需要设定程序睡眠时间。这种情况下，程序应该可以在不出错的情况下持续运行；

(6) 其他方面：登录学校教务网时，密码账号学校没有做任何保护措施（即没有任何加密算法），直接以明文传送，所以模拟登录出了验证 Cookie 外没有任何技术难度，其中验证 Cookie 的方法也已在方案中给出，根据实验表明该方法切实有效，具有可行性。

## 3. 可行性分析总结

上述可行性分析，参考了现有数据、技术和方法，了解了同学们的看法，且该脚本构造简单，开发要求低，具备进一步进行需求分析和开发的条件。

## 4. 项目功能需求

### 4.1 基本需求分析

使用 Python 语言，设计并实现一款针对南京大学教务网选课系统的自动抢课脚本。可进行抢课的时段为开学时期的课程补选阶段（因为期末的选课阶段是靠随机抽签来决定是否选中课程

的，所以不存在抢课的问题），抢课的种类应包含学校所开的除专业必修的其他所有课程：通识研讨课、公选课、通修课（思政军事、计算机、数学、英语）、跨专业选修课、体育。脚本可以同时多种课程进行补选（每种课程仅限一门）。且每个在校的本科生都能使用该脚本。

## 4.2 系统设计方法

第一步：模拟登录。该脚本的功能实质上是模拟同学们的操作进行课程的补选，因此，第一步就是要登录到学校的教务系统；

第二步：选择需要进行课程补选的课程种类；

第三步：填写每个种类中需要进行补选的课程编号；

第四步：获取课程页面的源代码，获取课程 ID；

第五步：伪造成浏览器，向教务网服务器传输数据（选课的本质就是从本地传送选课信息到教务网服务器，教务网服务器接收信息后对学生信息进行相对应的修改）；

第六步：刷新页面，重复第四步直到选课成功或出现异常中断为止。

## 5. 开发环境、硬件及软件需求

5.1 开发工具：Pycharm、谷歌浏览器开发者工具、Sublime Text 3。

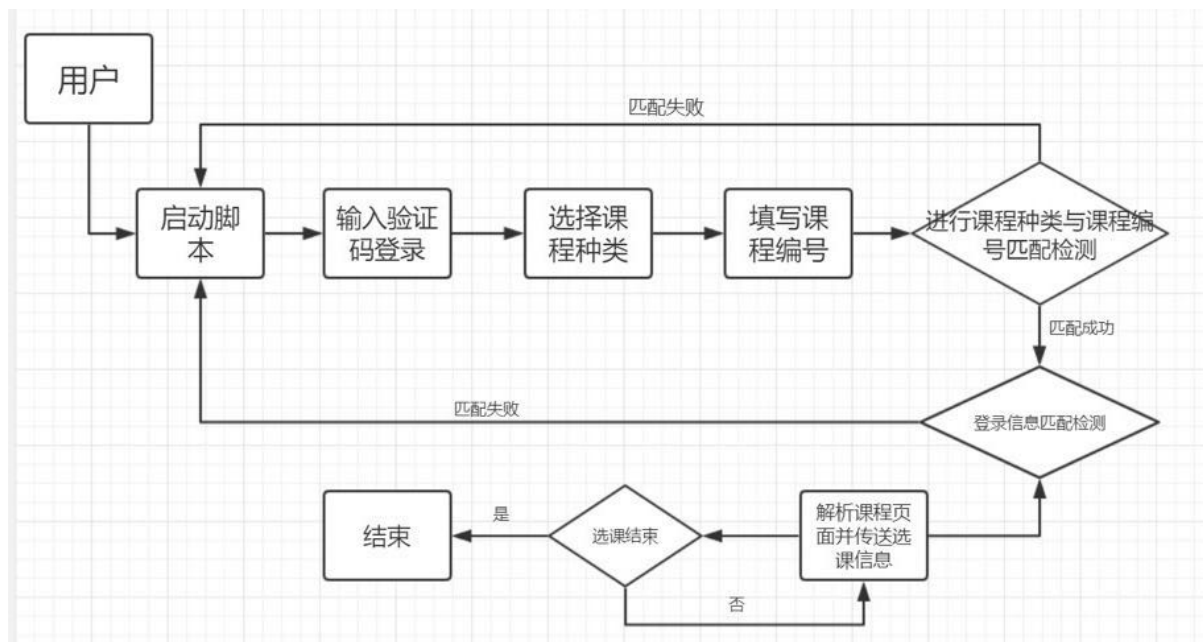
名称	参考网站	作用	图标
Pycharm	<a href="https://www.jetbrains.com/pycharm/">https://www.jetbrains.com/pycharm/</a>	方便进行 Python 程序的集成开发和测试	
谷歌浏览器开发者工具	<a href="https://devework.com/google-chrome-developer-tools-tutorial.html">https://devework.com/google-chrome-developer-tools-tutorial.html</a>	分析网络传输数据并方便获取必要的信息	
Sublime Text 3	<a href="https://www.sublimetext.com/">https://www.sublimetext.com/</a>	轻量级的文本编辑器，可随时方便地对代码进行小的调整和测试	

5.2 编程语言：Python，参考网站：<https://www.python.org/>

5.3 硬件需求：个人 PC，操作系统为 WIN10/MAC OS/UBUNTU 16，配有网卡，并有 Python 3 运行环境。

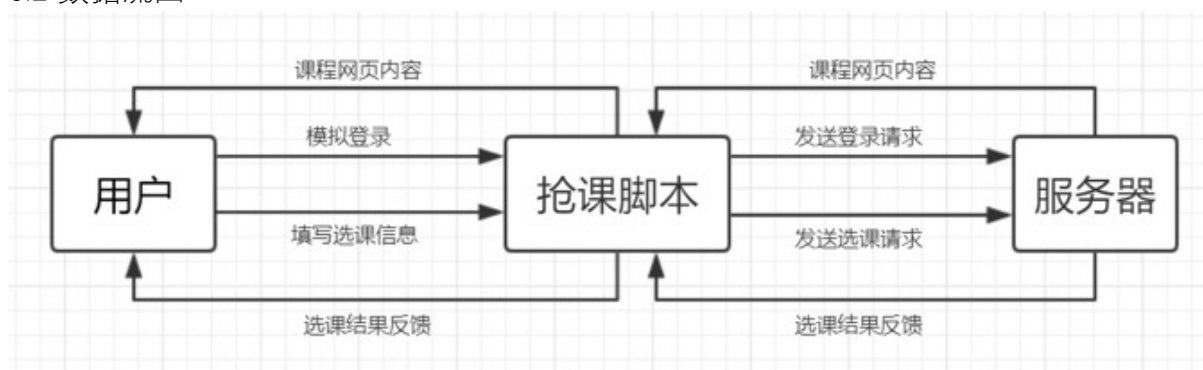
## 6. 系统结构化功能建模、数据流图

6.1 系统结构功能流程图：



(通过该流程图可看出，一旦课程匹配失败、账号密码或者验证码错误均需重新启动脚本，这样设计的原因是，从中断点返回程序开头用程序语言描述较为麻烦，且该脚本只是面向开发者个人。另外该脚本不能在输入账号密码及验证码的同时就进行检测，因为这些数据脚本是没有权力访问的，为了简便代码编写，将其与课程检测代码放在一起。)

## 6.2 数据流图：



数据的具体说明：

实体名称	数据流	数据简介	数据类型
用户	模拟登录信息	一串由用户填写的字符串（即验证码，学号密码已内嵌在脚本代码之中）	Char
用户	选课信息	分别为课程种类编号、课程编号两个字符串，将会被脚本进行处理。	Char
抢课脚本	登录请求数据	一个字典结构，其内容为学号、密码、验证码。	Dict

抢课脚本	选课请求	一个字典结构，其主要内容较为复杂，详情请参见项目计划书。	Dict
服务器、抢课脚本	课程网页内容	一个 HTML 文档，包含了抢课页面的所有信息，脚本会从其中提取所需的信息。	Char
服务器、抢课脚本	课程网页内容	一个 HTML 文档，包含了抢课结果信息（如成功、失败及原因）	Char

## 7. 具体实施方案

第一步：模拟登录。模拟登录最重要的一部分就是 Cookie 验证，Cookie 指某些网站为了辨别用户身份、进行 session 跟踪而储存在用户本地终端上的数据（通常会被加密）。在提交账号密码及验证码的信息时需要进行 Cookie 验证才能成功登录，其具体原理尚不十分清楚，但经过实验观察可知，登录时要验证的 Cookie 是验证码 Cookie 的一部分，所以具体实施方案是获取验证码 Cookie 并对其进行修改。关于验证码部分，将直接采取下载验证码图片到本地然后手动输入（用机器学习的方法来自动识别难度较大，不予考虑）；

第二步：选择需要进行课程补选的课程种类。以字符串的形式输入，用空格来进行分割不同种类，然后将其存储到列表中；

第三步：填写每个种类中需要进行补选的课程编号。仍采用字符串形式输入，用空格来分割，转化成列表。再将其与第二步中课程种类的列表组合成字典，方便后续操作；

第四步：获取课程页面的源代码，获取课程 ID。课程 ID 是抢课系统中最关键的一部分，但是课程 ID 并不是在网页上面直接能看到的课程编号，ID 隐藏在网页源代码中并没有被显示给出。观察各个种类课程的网页源代码发现，ID 与课程编号是紧密连在一起，可以使用 python 的正则表达式库 re 轻松地提取出来；

第五步：首先是伪造浏览器头，根据试验，伪造的浏览器头中需要包含 Cookie、User-Agent 及 Referer 三个元素，这三个元素可以利用谷歌浏览器等带有开发者工具的浏览器获得。接下来是组织需要进行传送的数据，手工进行各类课程的选择操作，然后利用谷歌浏览器开发者工具进行抓包，观察到各类的课程的补选需提交的数据由三种元素构成，其中的 method 和 classId 两个元素无论选什么类型的课都需要用到，第三个元素分别列出如下：

①、通识、研讨课第三个元素为 campus，在本抢课脚本中 campus 的参数总是为“仙林校区”；

②、通修课（思政军事、数学、英语、计算机）第三个元素为 courseKind，其中数学对应的 courseKind 为“13”，英语对应的 courseKind 为“14”，思政军事类对应的 courseKind 为“15”，计算机对应的 courseKind 为“16”；

③、跨院系选修课的第三个元素为 academy，具体参数值参见附录；

④、经典悦读课的第三个元素为 type，文学艺术、历史文明、哲学宗教、经济社会、自然生命、全球化与领导力六个种类对应的参数值分别为 1、2、3、4、5、6；

⑤、体育课对应的第三个元素为“\_”，即一个空元素，没有任何参数值，但是传输数据时也要将其添加到数据字典中，以免出现传输错误的问题。

在该步骤中，伪造的浏览器头 header 和传输的数据 data，两者的类型均为字典。

第六步：刷新页面，重复第四步直到选课成功或出现异常中断为止。脚本的不断刷新页面可以理解为不断对某个 URL 不断进行访问（使用 python 的 requests 库中的 get 方法，但刷新页面提交数据有什么影响，刷新页面在脚本里是不是必要的，这两个问题尚不清楚，但不会影响程序正常开发及使用）。至于重复第四五步，将第四五六步放在一个 while 循环中即可实现。

（PS：因为该脚本可以同时多种课程进行补选操作，所以会采用多进程编程）

## 8. 详细设计说明

### 8.1 外部模块引用

(1) 该脚本主要涉及了网络编程，python 开源市场中主要有 urllib、urllib2、requests 等第三方模块，其中 urllib 及 urllib2 主要是用 python2 进行开发的，与本项目所使用的 python3 不一定兼容，requests 库是用 python3 对 urllib 进行了优化和集成，使用的方便性大大提高，因此本项目将引用 requests 库；

(2) 该项目有从网页源代码中提取信息的功能模块，且本脚本采用的提取信息方式是：获得网页源代码的文本，根据目标信息构建正则表达式然后与源代码文本进行比对，找出所需信息，python3 支持的正则表达式库是 re，因此本项目将引用 re 库；

(3) 在利用该脚本进行模拟登录时，会获取验证码并自动弹出以方便用户输入，因此需要第三方库 PIL 中的 PIL.Image.show()方法，所以本项目将引用 PIL 库；

(4) 该脚本为了提高抢课效率采用了多线程编程，因此需要以下几个方法：往线程池中添加线程、启动线程、关闭线程等方法，上述方法包括在了 python 自带的 threading 模块中，因此该项目将引用 threading 库；

(5) 为了防止过于频繁的访问导致本机 IP 地址被封锁，该脚本在抢课时设定了休眠时间，因此需要 time.sleep()方法，因此该项目将引用 time 库；

(6) 为了优化阅读课类选课方法的编写，放弃使用 re 正则匹配，而采用了 lxml 库的方法对 html 文件进行解析，提高了匹配效率和精度，因此该项目将引用 lxml 库；

(7) 在设置课程种类或者填写课程代码时的错误会导致程序的终止，为了使程序终止，将采用 sys.exit()方法，因此该项目还将引用 sys 库。

### 8.2 公共类设计

类名：SelectCourse

类功能：定义抢课类，定义了抢课中将被使用的几个基本方法

调用方法：

```
SelectCourse.setCourseKind(self, course_kind);
SelectCourse.getSelectedCourse(self, academy);
SelectCourse.getClassId(self, class_number, elect_course_text);
SelectCourse.getSelectedCourseUrl(self, class_number, class_id, academy);
SelectCourse.getMethod(self, class_id, academy);
SelectCourse.startGrabing(self, select_course_url, method, i);
```

方法功能说明：

SelectCourse.setCourseKind(self, course\_kind)，将某个抢课过程实例化后，使用该方法对抢课种类进行设置，可供设置的种类如下：

编号	可供设置的种类名
1	public
2	discuss
3	common_math_course
4	common_English_course
5	common_tm_course
6	common_computer_course
7	kuayuanxi
8	Lecture_and_Art

9	History_and_Culture
10	Philosophy_and_Religion
11	Economic_and_Society
12	Nature_and_Life
13	Globalization_and_Leadership
14	gymClass

(在使用过程中，用户只需输入对应的编号即可，脚本会自行根据编号进行抢课种类设置)

种类名分别对应的课程种类请参见附录；

SelectCourse.getSelectedCourse(self, academy), 获取某种课程种类的课程列表的网页源代码，并将其作为返回值；

SelectCourse.getClassId(self, class\_number, elect\_course\_text), 获取发送抢课请求时的课程信息，该信息隐藏在课程列表的网页源代码中，以列表形式返回；

SelectCourse.getSelectedCourseUrl(self, class\_number, class\_id, academy), 构建发送抢课请求的接收地址；

SelectCourse.getMethod(self, class\_id, academy), 根据不同的课程种类分别构建不同的请求信息中的 method，以字典形式返回；

SelectCourse.startGrabing(self, select\_course\_url, method, i), 将上述方法所得到的信息综合起来发送抢课请求并获取抢课结果已决定是否继续发送请求，抢课结果对应的返回值如下：

抢课结果	对应返回值
该课程补选阶段已结束	-1
操作成功	1
已经选修过该课程	-2
添加失败，你已经选过这个班级了	-3
每学期限选两门	-4
其他	-5
班级已满	0
班级人数已满	0

当返回值为 1、-1、-2、-3、-4 或 -5 时将结束抢课，当返回值为 0 时将继续进行抢课。

参数说明：

参数名	类型	说明
course_kind	char	用户设定的需要进行抢课操作的课程种类名
academy	char	在进行跨院系抢课时，抢课信息中需包含学院编号信息（详情参见附录）
class_number	char	用户设定的需要进行抢课操作的课程编号（不同课的编号可能相同）
elect_course_text	Html 文本	是某种类课程列表网站的网页源代码
class_id	char	每门课程都有一个唯一的 ID，在发送抢课请求时必需的信息
select_course_url	char	进行某门课程的抢课操作时，信息的传送目的地址
method	dict	是抢课信息的一部分，不同种类的课程的选课 method 不同

i	Int	用于记录某门课程对应的抢课申请次数
---	-----	-------------------

### 8.3 其他方法设计

其余方法均为不同课程种类的具体抢课过程。

方法名：

```

public_course();
discuss_course();
common_math_course();
common_English_course();
common_tm_course();
common_computer_course();
KuaYuanXi_course();
Lecture_and_Art();
History_and_Culture();
Philosophy_and_Religion();
Economic_and_Society();
Nature_and_Life();
Globalization_and_Leadership();
gymClass();
keep_grab();

```

以 public\_course()为例来介绍不同课程名方法的具体功能：

```

def public_course():
    PublicCourse = SelectCourses()
    PublicCourse.setCourseKind('public')
    elect_course_text = PublicCourse.getSelectedCourse(academy)
    class_number = dict_courses['1']
    class_id = PublicCourse.getClassId(class_number, elect_course_text)
    #print(elect_course_text)
    select_course_url = PublicCourse.getSelectedCourseUrl(class_number, class_id, academy)
    method = PublicCourse.getMethod(class_id, academy)
    i = 0
    while True:
        try:
            result = PublicCourse.startGrabing(select_course_url, method, i)
            if result == -1:
                print('选课结束')
                break
            i += 1
        except Exception as err:
            print(err)
            time.sleep(10)
            continue

```

其余课程抢课方法与上述代码基本一致。

另外 keep\_grab()方法只是一个 while 循环，其目的是为了保证父线程一直运行即保持各子线程出于运行状态。

## 9. 测试分析报告



由于学校安排的选课时间较短，而该项目开发周期较长，所以采取了边添加功能边测试的方法，导致了后续功能如：跨院系选课、悦读课选课、体育课选课还可能存在问题。

对前期的开发功能进行了压力测试，随机选择了通识课、研讨课、通修课等不同课程进行不断选课试验，不间断的访问次数为 35578 次-66732 次，显然，我们不能准确预测脚本在访问多少次服务器后会停止访问。

测试完逻辑功能后，由开发者本人及另外一名的同学进行实际功能测试，经过试运行后，开发者及另外一名同学均抢到了目标课程，显然实际使用效果较好（注意：若没有同学进行课程的退选，当某课程班级已满时是不可能选上的，因此选课的成功还需要一定的运气）。

## 10. 项目开发总结报告

本项目实际上是我开发的第一个具有实际使用意义的项目，与以前实验课上开发的课程项目有着天壤之别。曾经的课程项目只需要在老师给定的假设条件下完成给定的几个功能即可，逻辑上基本上没有难度，就是纯粹的写代码，但这个项目让我意识到了真正应用级项目的开发难度，本项目应属于那种最低端、最简单的应用程序，没有 UI、没有数据库等等，只有一个纯粹的逻辑体，但是开发这个逻辑体就已经极为不易。实际开发中遇到了太多的问题，外部环境不再有老师给的那么简单了，出现的问题都得靠自己去学习解决。

该项目的开发难度高于了我的预期，但是既然自己对这个项目有兴趣而且也下定了决心，就应该去努力去做，当我把第一个 demo 程序写好后信心倍增，为我添加其他功能做好了基础。因此我认识到了兴趣和信心对于开发项目来说极为重要。

该项目还有许多设计不合理的地方，由于时间有限、能力有限没有对其进行修改。我将主要的精力都投放在了主体逻辑实现上，开发这个项目对我的 python 编程水平有较大的提高。而且，对于这个项目我还有进一步的打算，等我学习完 html、JavaScript 等前端知识后，打算自己写一个网站，将这个简单的逻辑体放到线上网站中，供大家一起使用。

## 附录

1. 各学院所对应的代号如下：

```

<option value="01">文学院</option>
<option value="02">历史学院</option>
<option value="03">法学院</option>
<option value="04">哲学系</option>
<option value="05">新闻传播学院</option>
<option value="06">政府管理学院</option>
<option value="07">信息管理学院</option>
<option value="08">社会学院</option>
<option value="09">商学院</option>
<option value="10">外国语学院</option>
<option value="11">数学系</option>
<option value="12">物理学院</option>
<option value="13">化学化工学院</option>
<option value="14">生命科学学院</option>
<option value="15">地球科学与工程学院</option>
<option value="16">地理与海洋科学学院</option>
<option value="17">大气科学学院</option>
<option value="18">电子科学与工程学院</option>
<option value="19">现代工程与应用科学学院</option>
<option value="20">环境学院</option>
<option value="21">天文与空间科学学院</option>
<option value="22">计算机科学与技术系</option>
<option value="23">医学院</option>
<option value="24">匡亚明学院</option>
<option value="25">软件学院</option>
<option value="25">软件学院</option>
<option value="27">工程管理学院</option>
<option value="28">海外教育学院</option>
<option value="29">建筑与城市规划学院</option>
<option value="45">计算中心</option>

```

## 2. 项目开发周期安排：

本项目开发周期分为三周。

项目开始日期：2018 年 3 月 25 日

项目截止日期：2018 年 4 月 15 日

其中，3 月 25 日至 3 月 29 日进行可行性分析，并撰写可行性分析报告，3 月 30 日至 4 月 6 日进行需求分析，撰写详细需求分析报告，并给出程序解决方案，4 月 7 日至 4 月 13 日根据制订的方案和需求说明书进行编码，4 月 14 日至 4 月 15 日进行测试，检测完成产品的性能并撰写检测报告。

## 3. 种类名对应的课程种类如下：

种类名	实际课程种类
public	通识课
discuss	研讨课
common_math_course	通修数学课
common_English_course	通修英语课
common_tm_course	通修思政军事类课程
common_computer_course	同修计算机课程
kuayuanxi	跨院系选修课
Lecture_and_Art	悦读课：文学与艺术
History_and_Culture	悦读课：历史与文化
Philosophy_and_Religion	悦读课：哲学与宗教
Economic_and_Society	悦读课：经济与社会
Nature_and_Life	悦读课：自然与生命
Globalization_and_Leadership	悦读课：全球化与领导力
gymClass	体育课