# 系统需求分析

# 排课子系统（刘权达）

①新课的审核，课程状态的修改（停开，开设），新课的申请。

②排课程：（只能操作当前开设状态为“开设”的课程）给课程分配教室，上课时间（教学周x-y，周几，第几节课-第几节课，学年，通过后台接口获取），任课教师，上课专业。

排课程需要注意：（1）教师的时间不可冲突；

（2）教室使用不可冲突（insert之前先select一下）。

③修改已排课程（换老师，时间，教室等，同样在update前检查有无冲突）。

④删除已排课程（注意级联删除，如果删除，则选修了这门课的全部学生的课表也要改变）。

# 选课子系统（高谦）

①控制选课的开始和结束（按专业控制）。

②设置某些课程为必修课（直接加入到该专业所有学生课表中）。

③学生在选课界面上选课，只能选择标有本专业标识的课。

注意：（1）防止课程时间冲突；

（2）防止教室容量不够（有数量限制）。

④生成学生课表，学生可在系统查看课表，下载课表。

⑤学生可在选课期间退选课程。

# 毕业设计子系统(余中瑞)

①教师发布研究课题相关信息。

②学生查看已发布课题，进行选择。

③学生查看毕业设计成绩信息（不仅仅是一个数字，还要有其他信息）。

成绩管理子系统（王艺琳）

①教师登陆成绩录入系统，录入学生成绩。

②在成绩录入期限内自由修改成绩。

③为方便教师操作，提供查询学生功能，一键定位。

④确认成绩无误后可提交成绩，任课教师再无修改权限，学生可查看到成绩信息（详细信息：平时分+期末考试成绩）。

⑤查看自己所教授课程学生的信息。

# 后台管理（孟庆强）

①开放/关闭成绩录入功能。

②新建学期，并将学期启用（启用后无法删除）。

③提供获取当前学期的方法函数。

④导入新生信息：学号，姓名，……，录入教师信息。

⑤转专业修改学生信息（学籍异动），维护教师和学生信息。

# 代码规范

①使用驼峰命名。

②每一个类，每一个方法写清注释（参数什么意思，返回的是什么）。

③dao和service层不处理异常，上抛或者统一异常处理（待定）。

④前端css样式加小组编号前缀防止css污染（ one\_name ， two\_name ， three\_name …… ）。

⑤每一个类之前写明编写这个类的作者。该类的方法中，如果不是该类创建者编写，写明方法的作者，方法上不写作者默认为类的作者。