**1系统概况**

**本章需简单介绍开发项目基本情况，如系统的编写目的，系统的开发背景**

**系统的实现自标等**

系统开发背景:学院对每一门课程的成绩实施透明、严格管理，并要求在每次考试结束后提交详细的考试分析报告。现有的考试成绩管理完全采用纸质的方式手工实现，容易出错且造成一定的资源浪费。为了提高考试成绩管理的效率，减少手工实现出错的机率，需要实现一个综合的成绩管理系统。成绩管理系统的具体要求如下：

以教师为单位配备系统：要求为每一个代课教师均配备一个成绩管理系统，需要同时管理本科生与研究生的课程成绩。与成绩管理系统相关的教师信息包括工资号、教师姓名和职称

与成绩管理系统相关的学生信息包括学号、姓名、性别。

教师每年可能会带多门课程，也可能一门课程都没有带

学生可能会选修多门课程。对于必修课和限选课，必须进行选择：对于任选课程，按照学分要求自行选择。

每个学生均属于一个班级，具有一个专业方向，课程可能以专业方向为单位进行开设，也可能以班级为单位进行开设。

每门课程包括课程编号、课程名称、课程类型、学时、学分、课程性质

每门课程的考试试卷具有两级结构：大题和小题。针对每一道题目，需要记录题目考察的具体知识点。

每门课程的成绩由多个部分组成，包括：出勤成绩、平时作业成绩、大作业成绩和考试成绩，每类成绩占有一定的权重。根据权重计算最终的成绩。

系统的编写目的：

设计并实现一个成绩管理系统。

系统的实现目标:

基于要求设计并实现一个成绩管理系统。

系统采用CS结构。

系统用户只有教师，但应具有一定的对系统的管理能力。

系统可以将数据导出为excel表。

**2系统需求分析**

**2.1系统总体需求**

**系统总体需求主要根据系统功能分析系统设计之前的规划工作，包括分析系**

**统元素的种类、功能、各数据项以及系统整体性能模式。另外，也包括系统各实体之间的联系，系统各模块功能的分析，软硬件平台的介绍等。**

系统采用MySQL作为数据库，在测试过程中将MySQL作为CS模型中的Server端运行在一台Linux（Debian）虚拟机中，该Linux虚拟机的IP地址设置为192.168.158.134，MySQL的远程端口为3306。而CS模型中的Cilent端使用Python完成，使用pymysql库连接MySQL服务端，使用xlwings库实现成绩单导出为Excel，使用PySide6库实现UI界面，使用PySide6-Fluent-Widgets库美化界面。使用Nuitka库进行可执行文件打包。实体有教师、班级、学生、课程、题目、试卷。

**2.2系统整体功能介绍**

**针对某一特定的系统，介绍该整体系统的功能结构，最好给出整体的功能结**

**构图。**

成绩管理系统的Client端由三个py文件组成分别是main.py、UILoginWindow.py、UIMainWindow.py。UILoginWindow.py和UIMainWindow.py两个文件分别负责控制登录界面和主界面两个UI界面的样式，另一个main.py文件则负责控制程序的行为。分别由两个类LoginWindow类和MainWindow类控制两个界面：登录界面和主界面。LoginWindow类中定义了login函数并将槽与登录按键的点击信号绑定。login函数负责验证按照输入的账号密码是否能访问ip地址为192.168.158.134且端口号为3306的MySQL服务器的big\_job数据库，若能则关闭登录界面并唤醒主界面否则要求用户重新登录。MainWindow类含有startF、tabelShow、edit、submit、scoreChange、tableC和fileoutB七个函数。startF函数是在主界面唤醒时被调用，负责读取登录用户的班级数据并在UI界面上更新；tabelShow函数的槽与加载按键的点击信号绑定，当点击加载按钮时函数读取数据库中内容并将其按照表格形式在UI界面中显示；edit函数的槽与id多选框的改变信号绑定，tableC函数的槽与表格所选项的改变信号绑定，两个函数负责在用户点击表格中的学生或在id多选框中切换学生时，修改框能立刻更新；submit函数的槽与提交按键的点击信号绑定，当提交按键被点击时，数据库中的对应学生的数据即被修改；scoreChange函数的槽与各个输入框的改变信号绑定，当输入成绩变化时，显示的最终成绩也同样更新；fileoutB函数的槽与导出按键的点击信号绑定，当导出按键被点击时，程序将当前显示的数据导出为以对应班级名称为文件名的Excel文件。

成绩管理系统的Server端为运行在Linux（Debian）虚拟机上MySQL服务器，IP地址为192.168.158.134，MySQL的远程端口为3306。数据库名称为big\_job，数据库的设计满足3NF范式。为保证数据的安全性，每个教师都有对应的数据库账户名和密码，并去除其所有全局性权限，仅将big\_job数据库的SELECT权限和big\_job数据库中Studen\_Course表上Achievement\_exam、Achievement\_bighomework、Achievement\_peacetime三个列的UPDATE权限。

**2.3 系统元素、实体介绍**

**分析系统中包含的元素项，实体以及他们之间的联系。**

教师实体含有的元素项有工资号、姓名、职称；课程实体含有的元素项有课程编号、课程名称、课程类型、学时、学分、课程性质；学生实体含有的元素项有学号、姓名、性别；班级实体含有的元素项有班级编号、专业方向、年级信息；成绩实体含有的元素项有考试成绩、大作业成绩、平时成绩；题目实体含有的元素项有题目编号、考查知识点。

教师实体与课程实体之间存在一对多关系；课程实体与学生实体之间存在多对多关系；学生实体与班级实体之间存在多对一关系；学生实体和课程实体共同与成绩实体之间存在一对多关系；课程实体与试卷实体之间存在一对一关系；试卷与题目之间存在一对多关系。

**2.4 软硬件平台介绍**

**对系统中应用到的软硬件平台给予详细的介绍。**

系统采用MySQL作为数据库，在测试过程中将MySQL作为CS模型中的Server端运行在一台Linux（Debian）虚拟机中，该Linux虚拟机的IP地址设置为192.168.158.134，MySQL的远程端口为3306。而CS模型中的Cilent端使用Python完成。

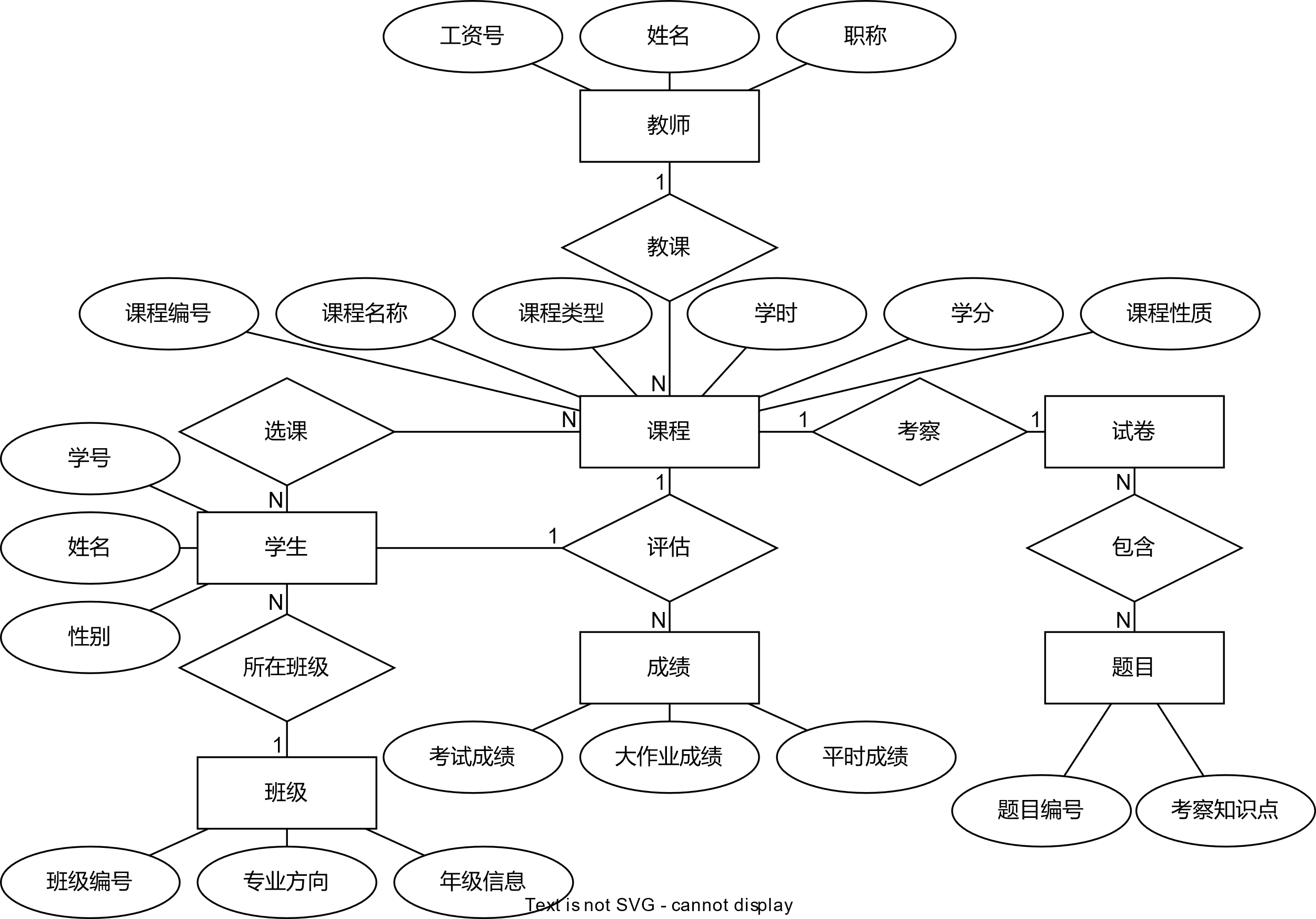
**3系统设计与实现**

**系统设计与实现包括系统的概念设计、各功能模块的设计、逻辑设计、界面**

**的设计以及数据库的设计等。**

**3.1 概念设计**

**给出系统各实体之间关系的E-R图，并作相应的介绍，并给出系统整体结构**

**功能。**

**3.2 模块功能设计**

**Teacher(教师)**

**表名 类型 说明 键**

**Teacher\_id integer 工资号 主键**

**Teacher\_name varchar 姓名**

**Teacher\_job varchar 职称**

**Class(班级)**

**表名 类型 说明 键**

**Class\_id integer 班级编号 主键**

**Class\_direction varchar 专业方向**

**Class\_grade integer 年级信息**

**Course(课程)**

**表名 类型 说明 键**

**Course\_id integer 课程编号 主键**

**Course\_name varchar 课程名称**

**Course\_type varchar 课程类型**

**Course\_time integer 学时**

**Course\_score integer 学分**

**Course\_properties varchar 课程性质**

**Teacher\_id integer 工资号 外键(Teacher)**

**Student(学生)**

**表名 类型 说明 键**

**Student\_id integer 学号 主键**

**Student\_name varchar 姓名**

**Student\_sex boolean 性别**

**Class\_id integer 班级编号 外键(Class)**

**Studen\_Course(学生课程中间表)**

**表名 类型 说明 键**

**Student\_id integer 学号 外键(Student)复合主键**

**Course\_id integer 课程编号 外键(Course)复合主键**

**Achievement\_exam real 考试成绩**

**Achievement\_bighomework real 大作业成绩**

**Achievement\_peacetime real 平时成绩**

**TestPaper(试卷)**

**表名 类型 说明 键**

**TestPaper\_id integer 试卷编号 主键**

**TestPaper\_num integer 大题数**

**Course\_id integer 课程编号 外键(Course)**

**Subject(题目)**

**表名 类型 说明 键**

**Subject\_id integer 题目编号 主键**

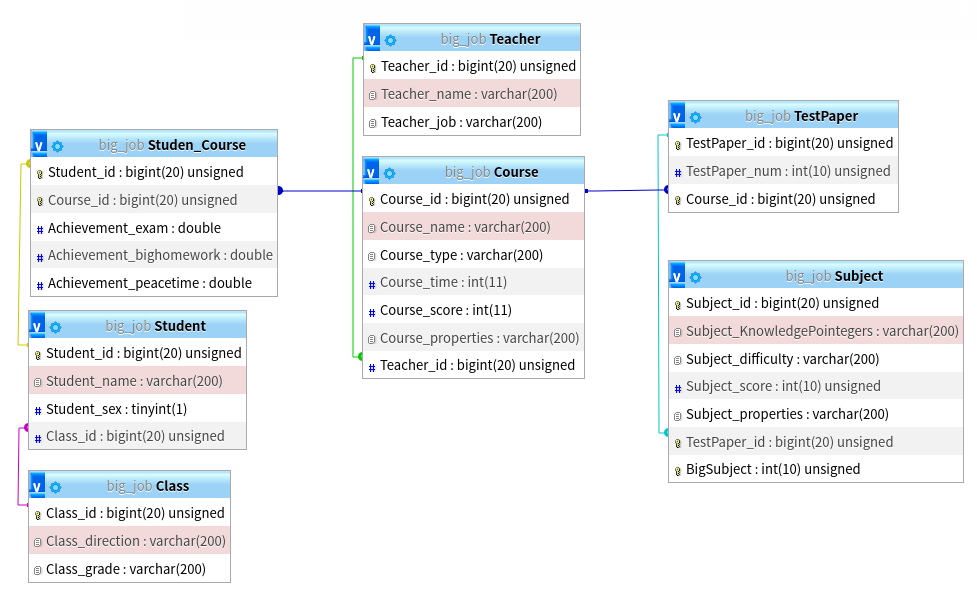
**Subject\_KnowledgePointegers varchar 考察知识点**

**Subject\_difficulty varchar 难易程度**

**TestPaper\_id integer 试卷编号 外键(TestPaper)**

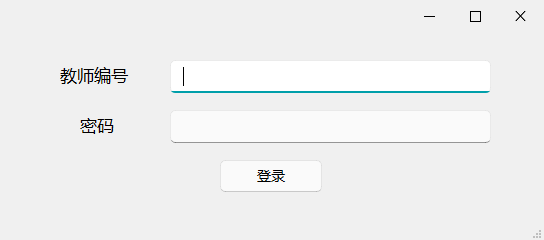
**BigSubject integer 所属大题号 外键(TestPaper)**

**3.3 逻辑设计**

****

**3.5 界面设计**

登录界面设计



教师输入教师编号和密码，检测输入的账号密码是否能访问ip地址为192.168.158.134且端口号为3306的MySQL服务器的big\_job数据库，若能则关闭登录界面并唤醒主界面否则要求用户重新登录。

主界面设计



当主界面刚被唤醒时如图所示，教师在右上角选择自己本次要管理的班级后点击加载即可获得数据库中数据并以表格的形式展示。



教师可在表格栏中点击或点击左下角下拉栏修改想要修改成绩的学生，点击提交即可将修改数据提交至数据库中。



点击文件到处即可将成绩单导出为Excel文件内容如下

9001班成绩单.xlsx

姓名 学号 班级 平时成绩 大作业成绩 考试成绩 总成绩

陈美欣 2350 202001 27 0 0 10.8

苏冰洁 2351 202002 12 20 100 56.8

沈冰枫 2352 202001 99 88 77 86.9

苏雄霖 2353 202002 13 0 0 5.2

段添池 2354 202001 0 0 0 0

薛磊 2355 202001 20 0 2 9

万建政 2356 202002 11 0 0 4.4

易天昊 2357 202001 0 0 0 0

唐国栋 2358 202002 0 0 0 0

**4系统总结**

**总结系统的设计与实现的心得、体会，以及系统的优缺点等。**

在设计与实现成绩管理系统的过程中，首先需要全面了解学院的需求和工作流程，包括教师管理课程、学生选修课程、成绩计算等。同时，还需要考虑数据的结构化和关联性，以便有效地存储和检索成绩数据。在系统设计过程中，需要注意系统的易用性和用户体验，确保教师能够方便地录入、查询和分析成绩数据。同时，系统应具备安全性，确保成绩数据不被未授权的人员访问和修改。

在实现过程中，采用合适的编程语言和技术来开发系统，例如使用python技术实现用户界面，使用数据库管理系统存储数据等。并且，适时进行测试和优化，以确保系统的性能和稳定性。

系统的优点：提高了成绩管理的效率：通过自动化管理成绩数据，减少了手工操作的时间和错误率，提高了成绩管理的效率。成绩管理系统提供便捷的查询功能，教师可以快速查看个人成绩、考试分析和班级整体表现。这种实时访问成绩信息的能力有助于及时了解学生的学习进展。系统可以方便地生成各类成绩报告，并支持数据导出为 Excel 表格等格式。这使得学院能够方便地将成绩数据用于进一步的分析和汇总。

系统的缺点：成绩管理系统需要技术支持和维护，如果出现技术问题，可能会影响系统的正常使用。当出现教师入职离职情况时，数据库管理人员需要手动进行用户的添加，删除，权限调整，若出现疏忽会极大的影响数据安全，很多教师习惯于使用Excel进行程序统计，系统缺少导入Excel的能力。