（一）确定子系统功能：

1. 登录模块需要设计

包括修改密码、找回账号、~~注册（教务端统一开通？）~~

（2）家长端子系统：

参与者：家长parent

用例：

1.查看学生基本画像checkStudentInfo()

2.查看学业画像checkStudentGradeInfo()

3.查看消费记录checkStudentConsumptionInfo()

4.查看考勤记录checkAttendanceRecord()

5.查看成长档案（需要调用查看学业画像）- checkTrackInfo()

6.是否需要增加一个（评价老师的功能？包括评价班主任和学科老师）

7.学生应该也可以补充或者写入自己的个人信息，比如父母、联系方式、家庭住址这些信息？

（3）教师端子系统

参与者：

班主任teacher（考虑——老师的权限、比如班主任、学科教师）

用例：

1.查看班级学生信息

是否包含了家长端的功能？

查看学生基本画像、查看学业画像、查看考勤记录、查看成长档案

录入学生综合评价

2.查看班级学业信息（比较模糊）

总成绩排名表（很多张）

各个科目成绩排名表（很多张）

总成绩分段占比表（很多张）

各个科目的成绩分段占比表（很多张）

分数特征升降图（总成绩、各个科目的最高分、最低分、平均分）一张

3.查看老师基本画像

4.是否需要增加一个修改老师个人信息的功能？老师应该是可以具备这个功能的，它应该是查看老师基本画像的扩展

5.查看学生评价

权限不同，如果是班主任，他应该可以查询班级各个学业的全部信息、也可以录入学生的综合评价、查看学生的各类信息

如果是普通教师，他应该只拥有查看学生的科目的学业信息、学生的总体成绩信息、应该无权查看其他学科的信息？也无权查看学生的基本画像？

（4）教务端子系统（增删查改？？）教务端不是什么功能、什么权限都有

过细，应该是从功能的角度出发而不是从数据库的角度出发

1.信息查询（查找方式应该有两种、一种是搜索、一种是直接看表）

（应该通过搜索查找吧？？，怎么会是通过输入表名查找呢？）查询学生的基本画像、学业画像、考勤记录、成长档案、~~查看消费记录、查看学生学期消费总额、平均每月消费总而~~

查看老师的基本画像

应该也是通过搜索查找吧？？查看老师基本画像（重叠）

（录入之前是否要查询学生/老师信息，如果查的到则修改新的信息、查不到则增加新的记录？）

2.信息录入（增加新的信息）

录入新学生：基本画像（姓名、班级、性别这些初始信息）（开通账号）

（消费记录应该是学校的机器能够自动识别写入，学业情况应该是根据学业录入自动生成和匹配？）

录入新老师：（开通账号）

录入新来老师的信息？设置老师管理的班级-授予老师权限（设置是否是班主任）

录入考试成绩：直接录入全级的成绩-自动归类写入到各个班级的表中

录入考勤记录：直接录入全级的考勤记录-自动归类写入到各个学生的表中

3.信息修改（修改之前应该要先查询学生信息）

要修改什么？

修改学生的基本信息？

修改老师的基本信息可以吗？

修改成绩可以吗？

注销学生账号？（已毕业）/注销老师账号？（已离职）

修改应该是基于查询：应该是信息查询的扩展功能？也应该是学业信息统计的扩展功能？

4.学业信息统计

查看学生人数、教师人数（柱形图）

查看男女比例（全校、各班）（扇形图）

统计考勤情况：

统计表、趋势图（全级每一周的考勤人数、每个班每一周的考勤人数、）

缺勤人数特别多的学生名单

统计年级学业：每一次考试全级学生（总成绩排名表、各个科目排行表）

学生分数在各个分数段的人数分布情况（扇形图、条形图）

全级学生在各个科目的平均成绩波动图、在总分的平均成绩波动图

统计班级学业：类似于老师端的班级学业信息，不同的是，教务可以查每一个班级的班级学业信息

班级学业比较：

各个班级总分（平均分）排名、平均分排名、最高分、最低分排名比较

（二）制作用例图-搞清楚用户需求

（三）类图制作

先抽象-不管具体的细节，从类的功能出发，建立基本的类和基本的关系

对象：人、家长 、老师、班主任、教务员

确定对象属性：

人：姓名、账号、密码、身份（表明权限）（抽象类）

家长：姓名、账号、密码、身份、

老师：姓名、账号、密码、身份、管理的班级

班主任：姓名、账号、密码、身份、管理的班级

教务员：姓名、账号、密码、身份

确定方法：

人：登录、修改密码

家长：登录、修改密码、

查看学生画像、修改学生信息、查看学业画像、查看成长档案、评价老师、

查看消费记录

老师：登录、修改密码、查看学科的学业情况、查看学生评价、查看基本信息、修改基本信息

班主任：登录、修改密码、查看班级学科情况、查看班级学生信息、查看班级学业信息、评价学生、查看学生评价、查看基本信息、修改基本信息

教务员：登录、修改密码、信息录入、信息修改、信息查询、学业信息统计（这些方法可能不会是单个的方法，可能是一个类）

1. 设计模式

采用MVC框架模式：

模型（Model），视图（View），控制器（Controller）

Model对象封装数据和基本行为

模型对象表示特殊的知识和专长。它们保存应用程序的数据，并定义处理该数据的逻辑。

---------------------------------------------------------------

View对象向用户提供信息

视图对象知道如何显示，允许用户编辑应用程序模型中的数据。

[(36条消息) 三层架构与MVC架构\_努力的阿迅的博客-CSDN博客](https://blog.csdn.net/qq_58665528/article/details/121008499)

[MVC 模式 | 菜鸟教程 (runoob.com)](https://www.runoob.com/design-pattern/mvc-pattern.html)

-----------------------------------------------------------

家长：登录、修改密码、

查看学生画像、修改学生信息、查看学业画像、查看成长档案、评价老师、

查看消费记录

登录功能:输入密码，然后验证

修改密码：录入密码，更新密码

查看学生画像：学生id作为标识符到数据库中查找

查看学业画像、查看成长档案、查看消费记录（过程相似）(多表查询)

评价老师：

---------------------------------------------------------------

老师：登录、修改密码、查看学科的学业情况、查看学生评价、查看基本信息、修改基本信息

查看学科的学业情况、查看学业评价、查看基本信息

修改基本信息

---------------------------------------------------------------

班主任：登录、修改密码、查看班级学科情况、查看班级学生信息、查看班级学业信息、评价学生、查看学生评价、查看基本信息、修改基本信息

查看班级学科情况、查看班级学生信息、查看班级学业信息、查看学生评价、查看基本信息

修改基本信息、评价学生

------------------------------------------------------------------

教务员：登录、修改密码、信息录入、信息修改、信息查询、学业信息统计（这些方法可能不会是单个的方法，可能是一个类）

信息查询、学业信息统计

信息录入、信息修改

Model直接与数据库对接，封装对数据库各种表的增删查改

Controller与model对接，当view需要显示数据时，调用controller,再去调model的数据，返回给view；需要修改数据时，也是通过controller，去调用model，model验证通过以后再去修改数据库，修改更新以后的数据再通过controller去更新view

--------------------------------------------------------------

Model:

Parents类、teacher类、headteacher类、manager类、person类

Person类基本属性：

Name:名字

Id:账号

Password:密码

Identity:身份

Status:状态

操作：

登录register()

修改密码changePassword()

查找信息search\_info()

删除信息delete\_info()

增加信息add\_info()

改变信息alter\_info()

Parents()类

修改学生信息

查看学业画像

查看成长档案

评价老师

查看消费记录

----------------------------------------------------------------

Controller:

ParentsController

teacherController

headTeacherController

managerController

更新数据的方法、调用view方法

-----------------------------------------------------------------

View:

ParentsView

teacherView

headTeacherView

managerView

定义显示的方法

Main：

登录->person

登录成功创建子类的model、controller、view

查找：

界面设计-》用户点击模块-》调用controller

Controlle

r自身的查询方法，自身的方法调用model的查询方法，得到数据调用自身的显示方法，该方法调用view的显示方法进行显示

改：

界面设计-》用户点击模块-》controller自身的修改方法，自身的方法调用model的修改方法，model修改database，修改以后如果需要显示则调用view的显示方法进行显示