

南京农业大学

数据库课程设计报告



题 目: _____ 学生家教管理系统 _____

姓 名: _____ 彭婉婷(组长) 徐欣桐 _____

学 院: _____ 信息科学技术学院 _____

专 业: _____ 计算机科学与技术 _____

班 级: _____ 计科 162 计科 162 _____

学 号: _____ 19216225 31116424 _____

指导教师: _____ 黄 芬 职 称: _____ 副教授 _____

2018 年 6 月 5 日

目录

目录

摘要	1
关键词	1
引言	1
1 课题分析	2
1.1 课题背景	2
1.1.1 国内研究概况	2
1.1.2 国外研究概况	2
1.2 研究目的及意义	2
1.3 本文所做的工作	2
2 技术路线及开发工具	3
2.1 技术路线	3
2.1.1 技术思路	3
2.1.2 技术路线图	3
2.2 开发技术及工具	3
2.2.1 开发技术	3
2.2.2 开发工具	5
3 功能分析与设计	5
3.1 需求分析	5
3.1.1 用户调研	5
3.1.2 初步设计	5
3.2 系统设计	6
3.2.1 系统流程设计	6
3.2.2 系统功能分析	6
3.3 概念结构设计	7
3.3.1 数据库设计	7
3.3.2 E-R 图	7
3.4 系统核心功能设计	11
3.4.1 核心功能顺序图	11
3.5 逻辑结构设计及物理设计	11
3.5.1 数据库关系表	11
3.5.2 数据库触发器	17
3.5.3 数据库函数	19
3.6 模块划分	19
4 开发与实现	20
4.1 系统核心程序编写	20
4.1.1 访问 MySQL 数据库	20
4.1.2 用户登录	20
4.1.3 用户注册	21
4.1.4 客户界面	24
4.1.5 家教界面	26
4.1.6 管理员界面	26

4.2 系统实施	27
4.2.1 用户登录注销功能.....	27
4.2.2 用户注册功能	28
4.2.3 客户查询家教	28
4.2.4 客户预约家教	29
4.2.5 客户付费	29
4.2.6 用户填写教学评价.....	30
4.2.7 家教回应预约	30
4.2.8 查看教学安排表	30
4.2.9 家教申请管理	31
4.2.10 客户缴费管理	31
4.2.11 客户/家教修改信息.....	31
5 系统测试与应用	32
5.1 系统测试运行环境	32
5.1.1 服务器端运行环境.....	32
5.1.2 客户端（浏览器）运行环境.....	32
5.2 数据库测试与运行	32
5.2.1 登录界面	32
5.2.2 客户/家教注册界面	32
5.2.3 客户界面	33
5.2.4 家教界面	35
5.2.5 管理员界面	35
5.2.6 help 界面	36
6 总结与展望	37
6.1 总结	37
6.2 展望	37
参考文献	38

学生家教管理系统

摘要:如今,很多家长都希望孩子成才,随着家长对孩子教育的重视程度越来越高,家教也越来越受家长的青睐,因为一对一教学使得授课更有针对性,能提高学生的学习效率。而随着互联网的发展,家教平台的应用是很有必要的,它使得家长可以在网上找家教,既可以通过查找家教的学历等信息来选择自己心仪的家教,再根据这个家教教过的学生的历史评价来确定自己的选择,又可以找到离自己最近的家教,节省在上课路上花费的时间,同时,对于有能力的应聘者,在工作之余也可以以家教的身份来帮助学生提高成绩,并且,教的越用心,学生越满意,获得的工资也越高,而维系双方关系的平台通过客户填写的问卷计算家教的反馈评分,从而成为决定工资的一部分因素,平台可以通过家教完成的小时数和反馈评分约束家教高质量地完成自己的工作,这样就可以得到一个多赢的结果。为此,在访问几个家长以及认真分析广大学生的需求后,设计本系统。本系统面向中小學生及其家长。基于 MySQL、HTML、JavaScript 等技术,在分析了家长预约老师方便性的需求后,设计并实现了学生家教管理系统。系统主要实现了用户登录注册功能、课程预约功能、查看个人信息、课程安排及评价功能,管理员审核管理功能等。

关键词: 学生; 家教; 预约; MySQL; JavaScript

引言:

改革开放以来,中国的经济飞速发展。一个处于经济快速增长、科技水平与日俱增的过渡时期,企业衡量人才的标准亦逐渐趋向多元化、理性化的良性发展轨道。家长越来越深刻的认识到家教的重要性。为了让孩子将来有一个美好的人生发展平台,越来越多的家长更加注重孩子教育,作为课外补习一种最有力的辅导方式——家教悄然升温。随着家教行业竞争的加剧、国内教育制度的不断深化改革,各类家教机构更加注重学生的素质教育,不再单一为提高孩子学习成绩而努力。教学方法,经营模式呈现多元化趋势,以适应市场为先导,更加地注重孩子学习兴趣、学习习惯的培养,增强家长对家庭教育的认知,以孩子个性资质为前提,以令孩子勤学、乐学、善学为目标,帮助孩子树立远大目标并鼓励孩子为之奋斗,同时将融入国外先进教育理念,致力于提高孩子综合素质。如何利用信息化的管理手段,提高对客户的服务水平和建立对客户需求快速反应的管理机制,在激烈的市场竞争中显得尤为重要。很多家长想让孩子在课下有更多获得教育的机会,但是自己要上班,不一定有足够的时间接送,而且,离家远的教学机构在路上花费的时间太长,既想找离家近点的家教老师,又想把握教学质量,而现在的小区里与许多私人家教,因为宣传力度不够,不为人所知,而且,家教市场出现了许多不良中介,牟取暴利,所以我们开发了学生家教管理系统,让家长可以通过搜索家教的住址和历史评价为孩子找到一个离家近且能保证教学质量的家教,同时,这些家教可以被更多家长发现,而平台的推优管理制度能够促使家教更好地指导孩子学习,一些公认的优秀家教可以被更多人了解。

1 课题分析

1.1 课题背景

计算机技术的发展使得人们更倾向于通过网络查询自己需要的信息，如今，高考是孩子奋斗的目标，对于快中考或高考的学生家长，他们希望为自己的孩子找到优秀的家教，让他们在假期、周末甚至每天晚上来辅导孩子学习，对于低年级的孩子家长，他们想培养孩子各方面的兴趣，比如绘画音乐方面。而私人家教尤其是大学生有着被中介公司坑骗的可能，而家长也不一定满意中介介绍的家教。

随着网络与人们生活的越累越密切，不少家教网站也像雨后春笋一样出现在我们面前，但也摆脱不了中介的本质，仍然需要给予网站一定金额^[3]，所以，我们决定开发学生家教管理系统，能够方便用户进行线上的预约；同时系统分为三个模块，对于客户，能够查找家教信息进行预约，查看预约结果和家教历史评价并进行评分评价；对于家教，可以填写空闲时间让客户预约，查看自己的预约信息以及历史教学安排评价；对于管理员，能够审核家教信息，管理平台并选出最优家教进行展示。

1.1.1 国内研究概况

家长比较信任有名的教育机构，找中介又觉得不靠谱，国内的教学机构如学而思、学大教育等都有自己的家教网，南京也有南大师大家教中心【高校联盟】这种让高校大学生当家教的平台，可以让经历过高考的考生给临近高考的考生一些全面的复习，并以其亲身经历指导考生高考，而常青藤、家教 114 等家教网站可以选择城市并筛选家教信息，使得家长可以找到心仪的老师，但预约则需要打电话，还是离不开人工服务，而有些网站仍然需要缴纳会员费等，注册的人不是很多。

1.1.2 国外研究概况

家教在国外被称为“影子教育体系”，国外关于家教信息管理的网站挺多，如：<http://teachinghome.com> 等网站，可以让家长足不出户找家教。国内外家长的眼界不一样，国外的孩子从小就跟着父母在全世界到处跑，参加各种展览会接受各种资源，学习多种语言，而家教的教学也以语言艺术为主，而不是像中国家教以辅导语数外物化生为主。

1.2 研究目的及意义

学生家教管理系统的应用，一方面，使家长能更方便的预约家教，通过地址条件查询找到离自己最近的家教，使孩子花在路上的时间减少，通过评分排序与家教历史评价，找到自己最满意的家教，还可以避免诈骗问题，家长每次的评分可以激励家教争取最优家教资格，拿到提成，从而保证了家教的教学质量，家长也可以放心；另一方面，家教不用找中介推销自己，只要上网填写自己的信息，展示自己的优势，就可以等着家长找你预约时间，并选择是否答应，通过这个家教管理平台教导孩子学习并赚取一定的工资。

1.3 本文所做的工作

本文数据库系统的相关理论用于实践，在系统开发需求分析的过程中，具体所做的工作有：

- (1) 在该系统的可行性分析和规划阶段，对于学生家教管理系统的现状做了简单的介绍，和实现系统的技术路线。
- (2) 在该系统的需求分析和总体设计阶段，给出了系统实现的功能模块和基本架构。
- (3) 在详细设计和具体实施阶段完成了对于学生家教管理系统的功能模块设计、功能设计、数据库设计，并完成实现了具体的功能。

2 技术路线及开发工具

2.1 技术路线

2.1.1 技术思路

本系统在着重分析当前学生家教管理系统的 之后，详细设计了体育馆场地预约管理系统的功能页面，完成信息的查询和场地预定、个人中心、管理场地等功能，结合 MySQL、HTML、JavaScript 等技术，在 eclipse 集成环境下完成实现了整个系统。

2.1.2 技术路线图

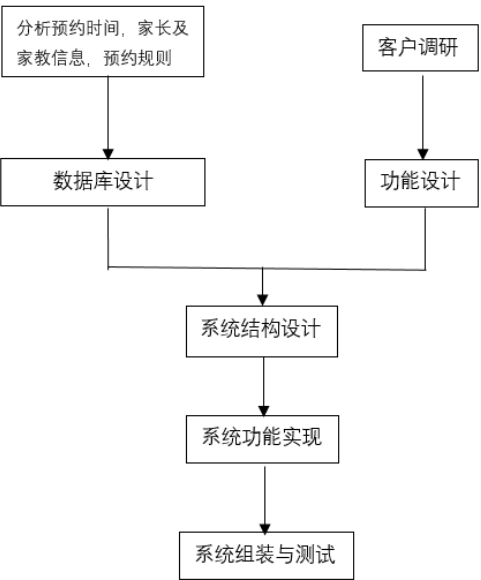


图 2-1 技术路线图

2.2 开发技术及工具

2.2.1 开发技术

本系统前端开发技术采用 HTML5、CSS、Jquery、javascript，后台开发技术采用 ssm 框架(Spring+SpringMvc+Mybatis)、MySQL。

HTML（Hypertext Markup Language,超文本标记语言），用于描述和绘制网页上的文字、图像、动画、声音、表格、表格和超链接等内容。HTML 不是一种计算机编程语言，而是一种网页描述标志，通过标记符号来标记要显示的网页中的各个部分。HTML 主要有简易性、可扩展性、平台无关性、通用性等特点。

CSS（Cascading Style Sheets，层叠样式表），是一种用来表现 HTML 或 XML 等文件样式的计算机语言。CSS 能够对网页中的对象的位置排版进行像素级的精确控制，支持几乎所有的字体字号样式，拥有对网页对象和模型样式编辑的能力，并能够进行初步交互设计。CSS 在布局、样式控制方面有着巨大的优势，实现了网页内容结构和格式控制相分离，从而使网页更加美观，维护更加容易^[4]。

JavaScript 是一种基于对象且事件驱动的客户端程序，它被嵌套在 HTML 网页中。JavaScript 程序可以检测网页中的各种事件并作出反应，也可以实现动态改变网页的 CSS 样式和结构并与页面各种元素交互等操作^[1]。

MySQL 是一种关联数据库的管理系统，关联数据库将数据保存在不同的表中，而不是将所有数据放在一个大仓库内，这样就增加了速度并提高了灵活性。MySQL 所使用的 SQL 语言是用于访问数据库的最常用标准化语言。由于其体积小、速度快、总体拥有成本低，尤其是开放源码这一特点，一般中小型网站的开发都选择 MySQL 作为网站数据库，并且它具有支持跨平台性、运行速度快、开源、功能强大等特点。

B/S 结构（Browser/Server，浏览器/服务器模式），是 Web 兴起后的一种网络结构

模式，Web 浏览器如 IE、MozillaFireFox、Chrome 等是客户端最主要的应用软件。这种模式统一了客户端，将系统功能实现的核心部分集中到服务器上，简化了系统的开发、维护和使用。在 B/S 模式中，浏览器以超文本形式向 Web 服务器提出访问数据库服务器，数据库服务器得到请求后，进行验证并处理，然后将处理后的结果返回给 Web 服务器，Web 端再一次将得到的所有结果进行转化，换成 HTML 形式，转发给浏览器以页面的形式显示。

SSM 框架是 spring MVC ， spring 和 mybatis 框架的整合，是标准的 MVC 模式，常作为数据源较简单的 web 项目的框架，将整个系统划分为表现层，controller 层，service 层，DAO 层四层；使用 spring MVC 负责请求的转发和视图管理；spring 实现业务对象管理，mybatis 作为数据对象的持久化引擎。

要完成一个功能：

- 1.先写实体类 entity，定义对象的属性，（可以参照数据库中表的字段来设置，数据库的设计应该在所有编码开始之前）。
- 2.写 Mapper.xml(Mybatis)，其中定义功能，对应要对数据库进行的那些操作，比如 insert、selectAll、selectByKey、delete、update 等。
- 3.写 Mapper.java，将 Mapper.xml 中的操作按照 id 映射成 Java 函数。
- 4.写 Service.java，为控制层提供服务，接受控制层的参数，完成相应的功能，并返回给控制层。
- 5.写 Controller.java，连接页面请求和服务层，获取页面请求的参数，通过自动装配，映射不同的 URL 到相应的处理函数，并获取参数，对参数进行处理，之后传给服务层。
- 6.写 JSP 页面调用，请求哪些参数，需要获取什么数据。

所谓数据库，即就是数据存放的地方，需要长期存放在计算机内，有组织、可共享的数据集合。数据库中的数据按照一定的数据模型组织、存储和描述，具有较小的冗余度、较高的数据独立性和易扩展性，可以为不同的用户所共享。

而数据库的内容通过数据库管理系统（Database Management System, DBMS）来管理。数据库管理系统是指数据库系统中对于数据进行管理的软件，它是数据库系统的核心组成部分，用户对数据库的一切操作，比如定义、查询、更新以及各种控制，都是通过数据库管理系统来操作的。

关系数据库是由数据表和表之间的关联组成。其中的数据表通常是一个由列和行组成的二维表，每一个数据表分别说明数据库中某一个特定方面或部分的对象及其属性。数据表中的行通常叫做元祖，它代表众多具有相同属性的对象中的一个；数据库表中的列通常叫做属性，它代表相应数据库表中存储对象的共有属性。

关系数据库由于其建立在严格的数学概念基础之上，概念单一，结构简单、清晰、用户易懂易用，存取路径等物理操作对用户透明等特点，成为当今的最主流数据库技术。

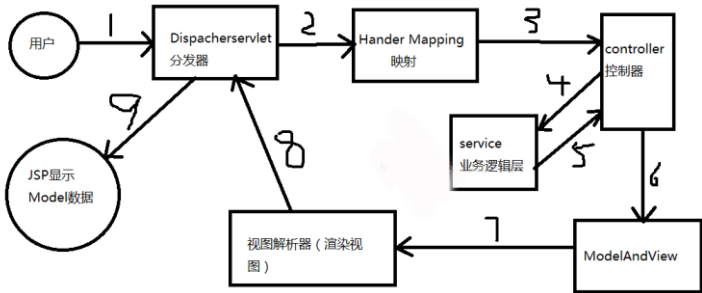


图 2-2 SpringMVC 原理图

2.2.2 开发工具

本系统在 Windows10 操作系统上开发, web 浏览器采用 tomcat,数据库服务器采用 MySQL。

Tomcat 是 Apache 软件基金会 (Apache Software Foundation) 的 Jakarta 项目中的一个核心项目,由 Apache、Sun 和其他一些公司及个人共同开发而成。由于有了 Sun 的参与和支持,最新的 Servlet 和 JSP 规范总是能在 Tomcat 中得到体现, Tomcat 5 支持最新的 Servlet 2.4 和 JSP 2.0 规范。因为 Tomcat 技术先进、性能稳定,而且免费,因而深受 Java 爱好者的喜爱并得到了部分软件开发商的认可,成为目前比较流行的 Web 应用服务器。

MySQL 是一个关系型数据库管理系统,由瑞典 MySQL AB 公司开发,目前属于 Oracle 旗下公司。MySQL 是最流行的关系型数据库管理系统,在 Web 应用方面 MySQL 是最好的 RDBMS(Relational Database Management System,关系数据库管理系统)应用软件之一。MySQL 是一种关联数据库管理系统,关联数据库将数据保存在不同的表中,而不是将所有的数据放在一个大仓库内,这样就增加了速度并提高了灵活性。MySQL 所使用的 SQL 语言是用于访问数据库的最常用标准化语言。MySQL 软件采用了双授权政策,它分为社区版和商业版,由于其体积小、速度快、总体拥有成本低,尤其是开放源码这一特点,一般中小型网站的开发都选择 MySQL 作为网站数据库^[8]。

3 功能分析与设计

3.1 需求分析

3.1.1 用户调研

本系统的用户角色主要是本校学生和他们的家长,为了能够更加清晰的体现本系统在家长预约家教方面的运用,并且使得本系统可以被用户最大限度地使用,在实际开发以前要针对用户做一定的相关资料收集,一定的考察,让系统的功能、环境等都可能的符合实际使用情况。

3.1.2 初步设计

根据系统的功能需求分析,可以将系统大体分为以下几大模块:

对于客户:

- (1) 注册登录/注销功能,修改个人信息功能;
- (2) 另外申请成为家教,并有相应的家教功能;
- (3) 通过输入条件查询到年龄,地址,学历,有无任教经历,优势科目,评分对应的应聘者的所有信息,历史评价,以及有空闲的时间段;
- (4) 填写客户预约表(家教 id,客户 id,科目,时间段 id,补课费用(yuan/h),补课科目,补课目标);
- (5) 通过自己的 id 查找所有教学安排表中未填写的完成的小时数进行填写,选填历史评价,通过评价生成反馈评分。

对于家教:

- (1) 注册登录/注销功能,修改个人信息功能;
- (2) 另外注册成为客户,并有相应的客户功能;
- (3) 通过自己的 id 查找预约信息(姓名,孩子名字,联系电话,地址,所处年级,时间段 id,补课费用(yuan/h),补课科目,补课目标),拒绝或答应;
- (4) 通过自己的 id 查找自己的教学安排评价记录(客户姓名,补课费用,时间段 id,完成的小时数,反馈评分,联系电话);
- (5) 通过自己的 id 查找自己的工资(客户姓名,工资),家教工资=原始工资*完成时间*(反馈评分/100)。

对于管理员：

- (1) 审批应聘者申请表；
- (2) 确认客户缴费。

3.2 系统设计

3.2.1 系统流程设计

根据对需求分析阶段所得到的信息进行分析，将系统流程细化如图 3-1：

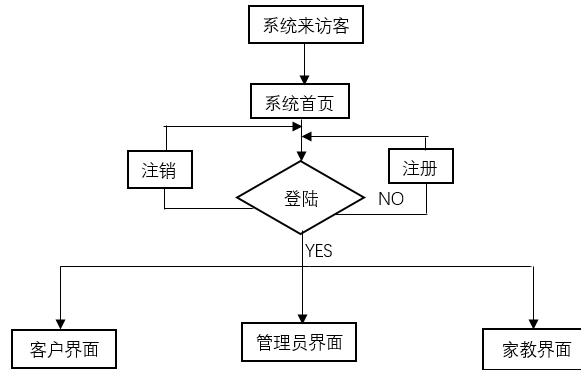


图 3-1 系统流程图

3.2.2 系统功能分析

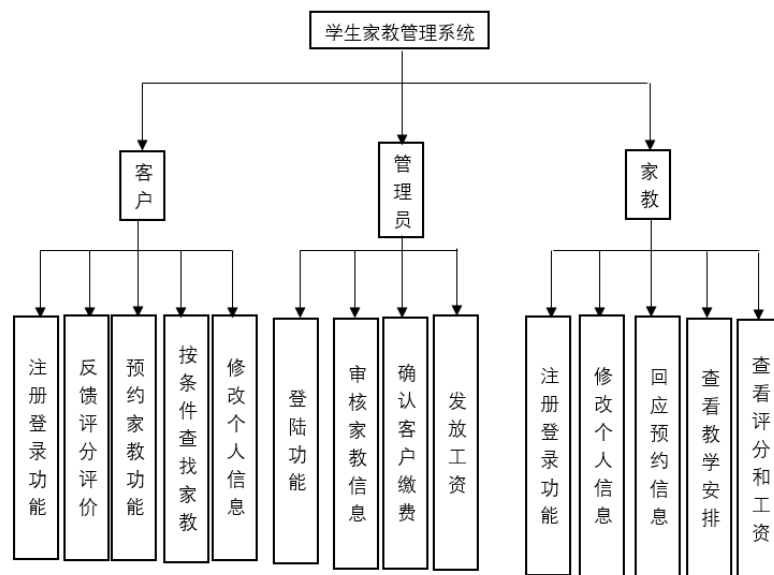


图 3-2 系统功能图

本系统主要功能根据使用者分为 3 个大模块：

对于客户：

- (1) 注册登录/注销功能，修改个人信息功能；
- (2) 另外申请成为家教，并有相应的家教功能；
- (3) 通过输入条件查询到年龄，地址，学历，有无任教经历，优势科目，评分对应的应聘者的所有信息，历史评价，以及有空闲的时间段；
- (4) 填写客户预约表（家教 id，客户 id，科目，时间段 id，补课费用（yuan/h），补课科目，补课目标）；
- (5) 通过自己的 id 查找所有教学安排表中未填写的完成的小时数进行填写，选填历史评价，通过评价生成反馈评分。

对于家教：

- (1) 注册登录/注销功能，修改个人信息功能；

- (2) 另外注册成为客户，并有相应的客户功能；
- (3) 通过自己的 id 查找预约信息（姓名，孩子名字，联系电话，地址，所处年级，时间段 id，补课费用（yuan/h），补课科目，补课目标），拒绝或答应；
- (4) 通过自己的 id 查找自己的教学安排评价记录（客户姓名，补课费用，时间段 id，完成的小时数，反馈评分，联系电话）；
- (5) 通过自己的 id 查找自己的工资（客户姓名，工资），家教工资=原始工资*完成时间*（反馈评分/100）。

对于管理员：

- (1) 审批应聘者申请表；
- (2) 确认客户已经缴费。

3.3 概念结构设计

3.3.1 数据库设计

每张表的主码使用下划线表明，外码使用斜体表明：

- (1) 用户登陆表 (用户 ID，用户密码)
- (2) 管理员登陆表 (管理员 ID，密码)
- (3) 上课时间段表 (时间段 ID，上课星期，开始时间，结束时间) (全码)
- (4) 家教空闲时间表 (用户 ID，上课星期，开始时间，结束时间，是否被预约)
- (5) 家教应聘申请表 (用户 ID，姓名，性别，年龄，联系电话，地址，学历，有无任教经历，最高学历就读学校，优势科目，获奖情况，证件图片地址)
- (6) 家教表 (用户 ID，姓名，性别，年龄，联系电话，地址，学历，有无任教经历，最高学历就读学校，优势科目，获奖情况，综合评分)
- (7) 客户表 (用户 ID，姓名，性别，年龄，联系电话，地址，孩子名字，所处年级)
- (8) 客户预约表 (编号，*家教 ID*，*客户 ID*，科目，*时间段 ID*，补课费用（yuan/h），补课目标，应缴费用，是否缴纳，是否同意)
- (9) 教学安排表(月)(编号，*家教 ID*，*客户 ID*，*时间段 ID*，上课科目，补课费用（yuan/h），补课目标，已完成的小时数，反馈评分，评价)

3.3.2 E-R 图

1) 上课时间实体 E-R 图：

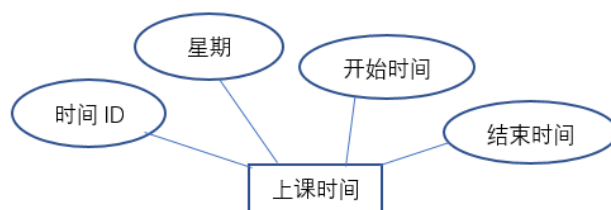


图 3-3 上课时间实体及属性

2) 管理者实体 E-R 图：

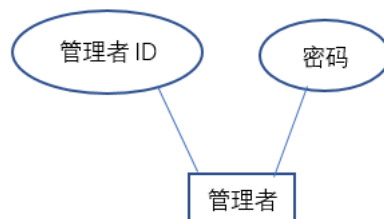


图 3-4 管理者实体及属性

3) 申请者实体 E-R 图:



图 3-5 申请者实体及属性

4) 客户实体 E-R 图:

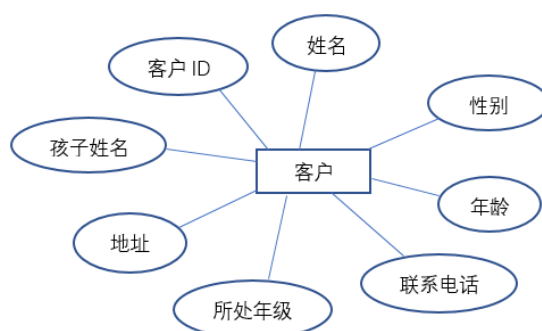


图 3-6 客户实体及属性

5) 家教实体 E-R 图:



图 3-7 家教实体及属性

6) 教学安排实体 E-R 图:

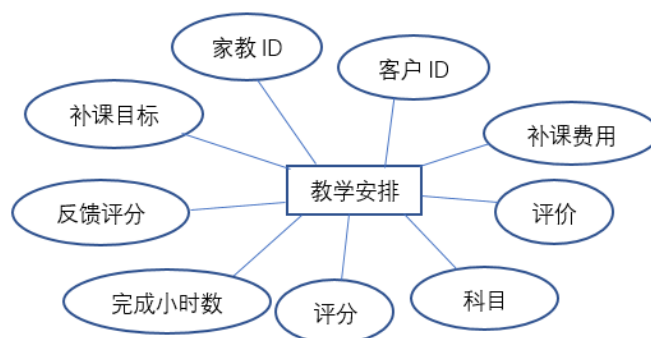


图 3-8 教学安排实体及属性

7) 系统总 E-R 图:

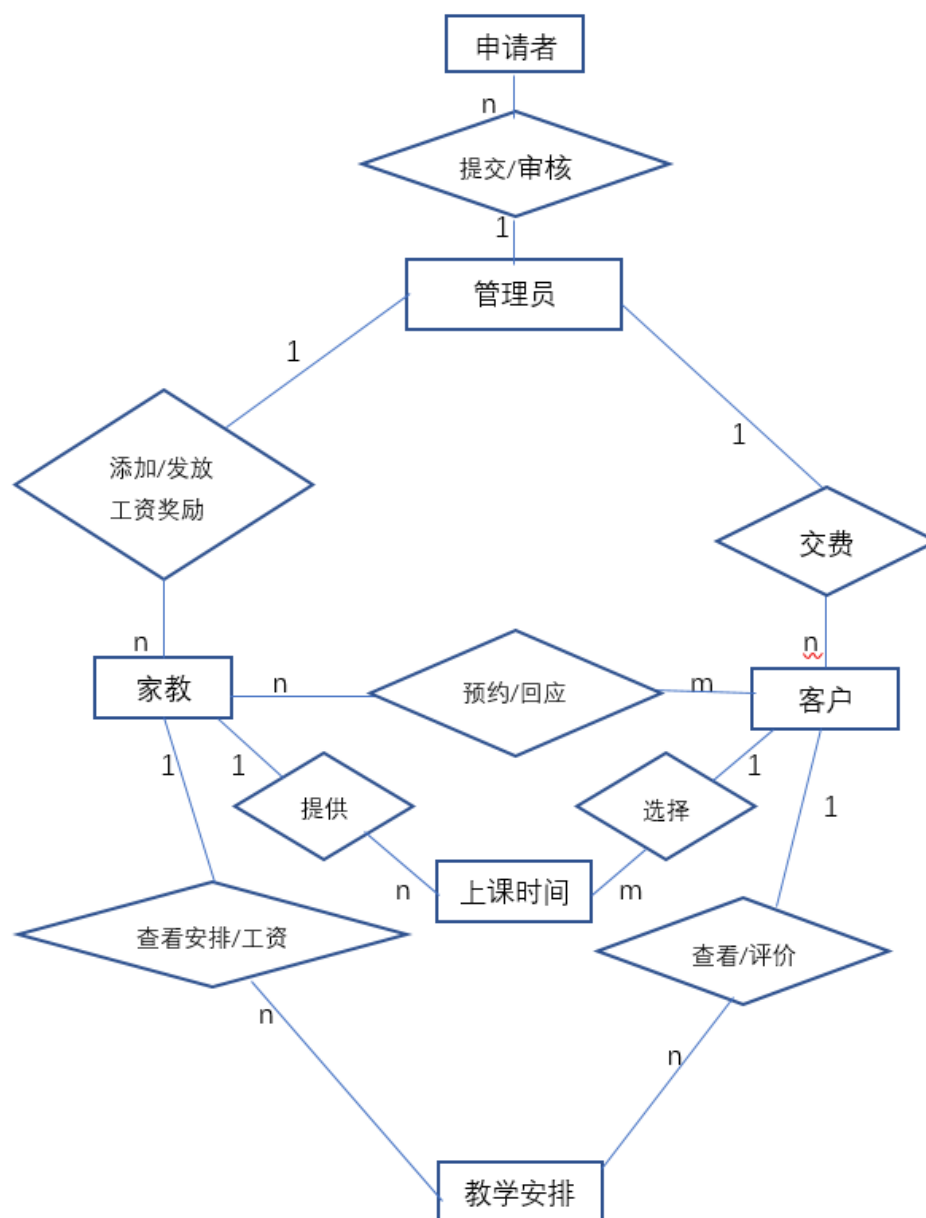


图 3-9 系统总 E-R 图 (无属性)

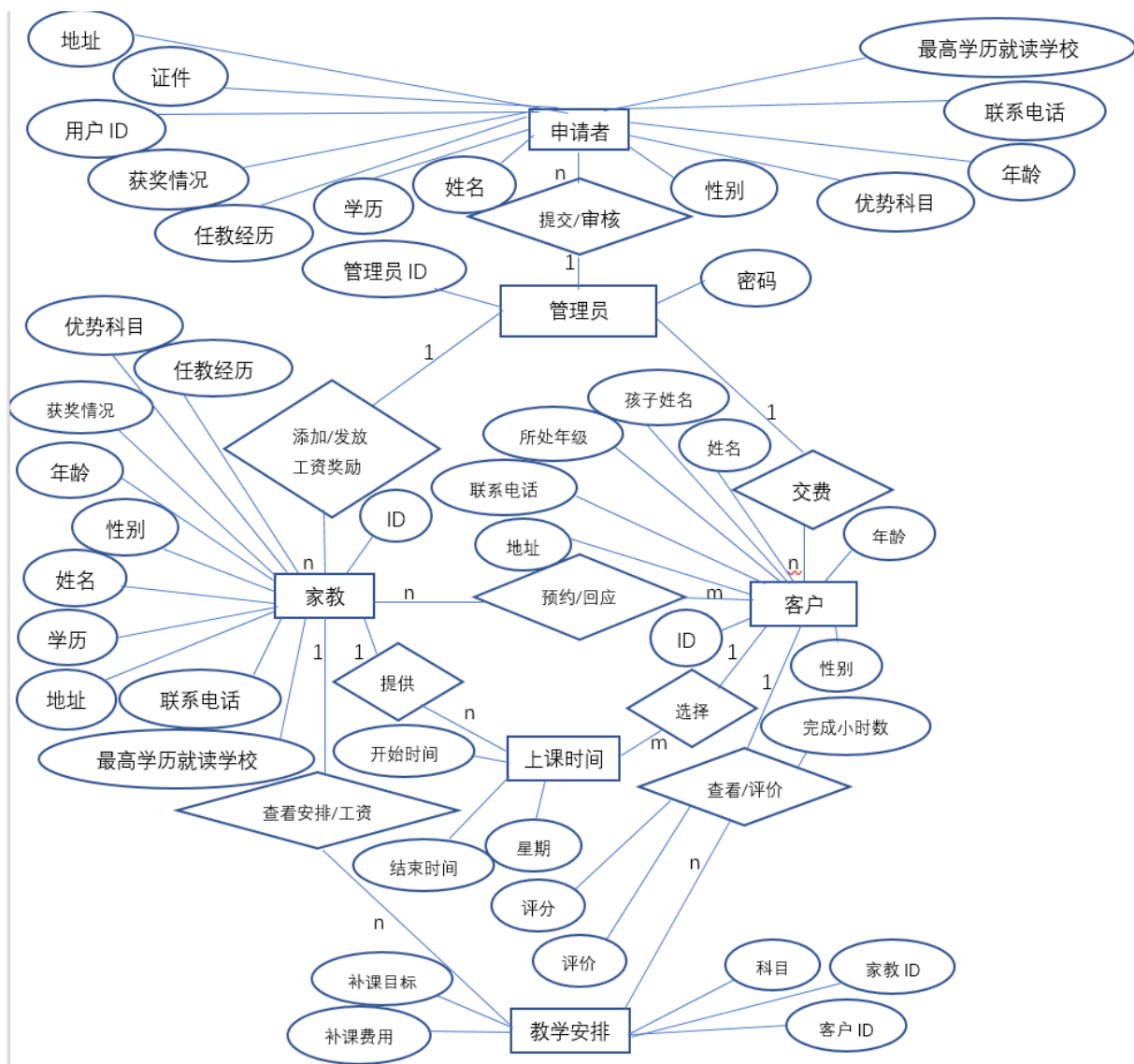


图 3-10 系统总 E-R 图（有属性）

3.4 系统核心功能设计

3.4.1 核心功能顺序图

UML 通过视图化的表示机制从多个侧面对系统的分析和设计模型进行刻画，其中广泛运用的顺序图是属于行为图的一种，强调了对对象之间发生消息传递的时间顺序。

系统整体数据流动的顺序图如图 3-10:

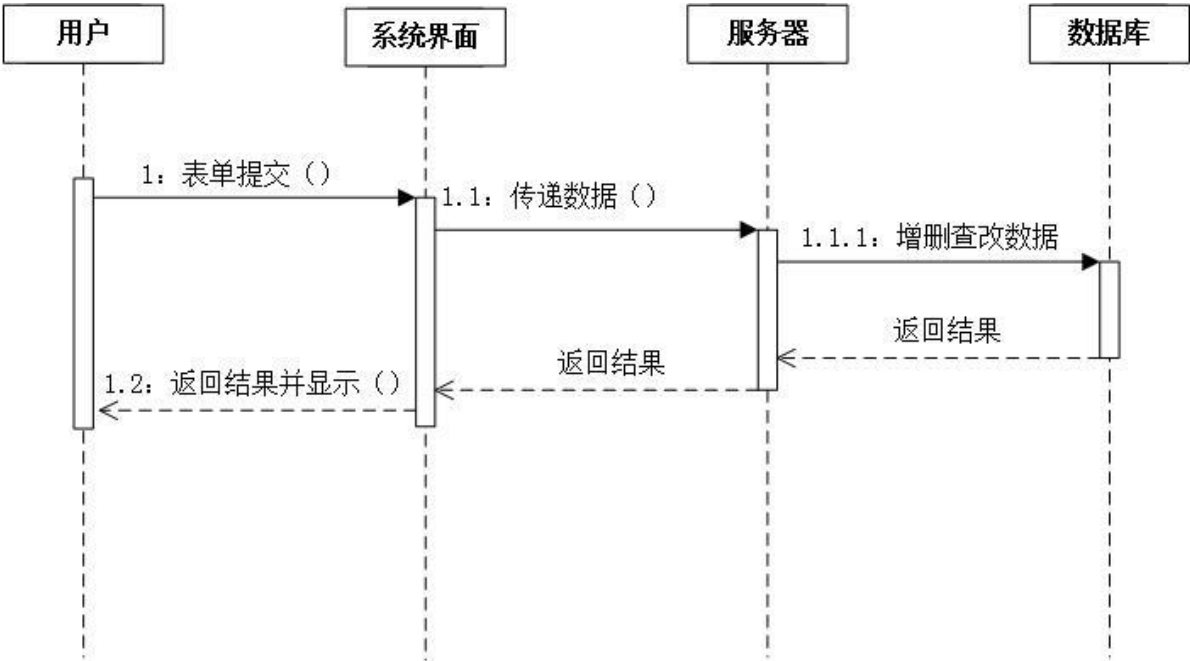


图 3-11 系统顺序图

3.5 逻辑结构设计及物理设计

3.5.1 数据库关系表

表 3-1 USERLOGIN (用户登录表)

序号	字段	别名	数据类型	数据长度	是否主码	是否可空
1	USERID	用户 ID	varchar	22	是	否
2	PASSWORD	密码	varchar	22	否	否

其创建代码如下:

```
CREATE TABLE IF NOT EXISTS `USERLOGIN`(  
    `USERID` VARCHAR(22) NOT NULL, #用户 id  
    `PASSWORD` VARCHAR(22) NOT NULL, #登录密码  
    PRIMARY KEY(`USERID`)  
);
```

表 3-2 ADMINISTRATORS (管理员登录表)

序号	字段	别名	数据类型	数据长度	是否主码	是否可空
1	USERID	管理员 ID	varchar	22	是	否
2	PASSWORD	密码	varchar	22	否	否

其创建代码如下：

```
CREATE TABLE IF NOT EXISTS `ADMINISTRATORS`(  
    `USERID` VARCHAR(22) NOT NULL, #管理员 id  
    `PASSWORD` VARCHAR(22) NOT NULL, #登录密码  
    PRIMARY KEY(`USERID`)  
);
```

表 3-3 SCHOOLTIME（上课时间段表）

序号	字段	别名	数据类型	数据长度	是否主码	是否可空
1	TIMEID	上课时间段 ID	int unsigned	默认	是	否
2	WEEK	星期	varchar	8	是	否
3	BEGINTIME	开始时间	time	默认	是	否
4	ENDTIME	结束时间	time	默认	是	否

其创建代码如下：

```
CREATE TABLE IF NOT EXISTS `SCHOOLTIME`(  
    `TIMEID` INT UNSIGNED NOT NULL, #上课的时间段 ID  
    `WEEK` VARCHAR(8) NOT NULL, #上课星期  
    `BEGINTIME` TIME NOT NULL, #开始时间  
    `ENDTIME` TIME NOT NULL, #结束时间  
    PRIMARY KEY(`TIMEID`, `WEEK`, `BEGINTIME`, `ENDTIME`)  
);
```

表 3-4 TIMESLOT（家教空闲时间段表）

序号	字段	别名	数据类型	数据长度	是否主码	是否可空
1	USERID	家教 ID	varchar	22	是, 外码	否
2	WEEK	星期	varchar	8	是	否
3	BEGINTIME	开始时间	time	默认	是	否
4	ENDTIME	结束时间	time	默认	是	否
5	OPPOINTMENT	是否被预约	int	默认	否	否

其创建代码如下：

```
CREATE TABLE IF NOT EXISTS `TIMESLOT`(  
    `USERID` VARCHAR(22) NOT NULL, #用户 id  
    `WEEK` VARCHAR(8) NOT NULL, #上课星期  
    `BEGINTIME` TIME NOT NULL, #开始时间  
    `ENDTIME` TIME NOT NULL, #结束时间  
    `OPPOINTMENT` INT NOT NULL, #是否被预约  
    PRIMARY KEY(`USERID`, `WEEK`, `BEGINTIME`, `ENDTIME`)  
);
```

表 3-5 APPLY (应聘申请表)

序号	字段	别名	数据类型	数据长度	是否主码	是否可空
1	USERID	用户 ID	varchar	22	是, 外码	否
2	ANAME	应聘者姓名	varchar	10	否	否
3	ASEX	性别	char	2	否	否
4	AAGE	年龄	int	默认	否	否
5	ATELENUMBER	联系电话	int	默认	否	否
6	AADDRESS	地址	varchar	12	否	否
7	EDUCATION	学历	varchar	50	否	否
8	TEACHINGEXPERIENCE	有无任教经历	varchar	10	否	否
9	LASTSCHOOL	最高学历就读学校	varchar	26	否	是
10	GOODSUBJECTS	教的科目	varchar	50	否	否
11	AWARDS	获奖情况	varchar	100	否	是
12	DOCUMENTADDRESS	证件图片地址	varchar	50	是	否

其创建代码如下:

```
CREATE TABLE IF NOT EXISTS `APPLY` (
  `USERID` VARCHAR(22) NOT NULL, #用户 id
  `ANAME` VARCHAR(10) NOT NULL, #应聘者姓名
  `ASEX` CHAR(2) CHECK(`ASEX` IN ('男','女')), #性别, 只能输入男或女
  `AAGE` INT UNSIGNED NOT NULL, #应聘者年龄
  `ATELENUMBER` VARCHAR(12) NOT NULL, #应聘者的电话号码
  `AADDRESS` VARCHAR(50) NOT NULL, #应聘者地址
  `EDUCATION` VARCHAR(10) CHECK(`EDUCATION` IN ('高中','大专','本科','研究生','博士')), #应聘者的学历
  `TEACHINGEXPERIENCE` VARCHAR(2) CHECK(`TEACHINGEXPERIENCE` IN ('有','无')), #应聘者有无任教经历
  `LASTSCHOOL` VARCHAR(26), #应聘者最高学历就读学校, 可以不填写
  `GOODSUBJECTS` VARCHAR(50) NOT NULL, #应聘者可以教的科目
  `AWARDS` VARCHAR(100), #应聘者的获奖情况, 可以不填写
  `DOCUMENTADDRESS` VARCHAR(50) NOT NULL, #证件图片地址
  PRIMARY KEY (`USERID`, `DOCUMENTADDRESS`),
  #定义参照完整性, 参照用户登录表的 userid, 级联删除和更新
  FOREIGN KEY (`USERID`) REFERENCES `USERLOGIN` (`USERID`)
  ON DELETE CASCADE
  ON UPDATE CASCADE
);
```


表 3-6 TUTOR (家教表)

序号	字段	别名	数据类型	数据长度	是否主码	是否可空
1	USERID	用户 ID	varchar	22	是, 外码	否
2	ANAME	应聘者姓名	varchar	10	否	否
3	ASEX	性别	char	2	否	否
4	AAGE	年龄	int	默认	否	否
5	ATELENUMBER	联系电话	int	默认	否	否
6	AADDRESS	地址	varchar	12	否	否
7	EDUCATION	学历	varchar	50	否	否
8	TEACHINGEXPERIENCE	有无任教经历	varchar	10	否	否
9	LASTSCHOOL	最高学历就读学校	varchar	26	否	是
10	GOODSUBJECTS	教的科目	varchar	50	否	否
11	AWARDS	获奖情况	varchar	100	否	是
12	SCORE	综合评分	int	默认	否	否

其创建代码如下:

```

CREATE TABLE IF NOT EXISTS 'TUTOR'(
  'USERID' VARCHAR(22) NOT NULL, #用户 id
  'ANAME' VARCHAR(10) NOT NULL, #应聘者姓名
  'ASEX' CHAR(2) CHECK('ASEX' IN ('男','女')), #性别, 只能输入男或女
  'AAGE' INT UNSIGNED NOT NULL, #应聘者年龄
  'ATELENUMBER' VARCHAR(12) NOT NULL, #应聘者的电话号码
  'AADDRESS' VARCHAR(50) NOT NULL, #应聘者地址
  'EDUCATION' VARCHAR(10) CHECK('EDUCATION' IN ('高中','大专','本科','研究生','博士')), #应聘者的学历
  'TEACHINGEXPERIENCE' VARCHAR(2) CHECK('TEACHINGEXPERIENCE' IN ('有','无')), #应聘者有无任教经历
  'LASTSCHOOL' VARCHAR(26), #应聘者最高学历就读学校, 可以不填写
  'GOODSUBJECTS' VARCHAR(50) NOT NULL, #应聘者可以教的科目
  'AWARDS' VARCHAR(100), #应聘者的获奖情况, 可以不填写
  'SCORE' INT NOT NULL, #应聘者的评分, 没有开始上课的应聘者这项为 NULL
  PRIMARY KEY ('USERID'),
  #定义参照完整性, 参照用户登录表的 userid, 级联删除和更新
  FOREIGN KEY ('USERID') REFERENCES 'USERLOGIN'('USERID')
  ON DELETE CASCADE
  ON UPDATE CASCADE
);

```

表 3-7 CUSTOMER（客户表）

序号	字段	别名	数据类型	数据长度	是否主码	是否可空
1	USERID	用户 ID	varchar	22	是， 外码	否
2	CNAME	应聘者姓名	varchar	10	否	否
3	CSEX	性别	char	2	否	否
4	CAGE	年龄	int	默认	否	否
5	CTELENUMBER	联系电话	int	默认	否	否
6	CADDRESS	地址	varchar	12	否	否
7	CHILDNAME	孩子姓名	varchar	10	否	否
8	GRADE	孩子就读年级	varchar	15	否	否

其创建代码如下：

```
CREATE TABLE IF NOT EXISTS `CUSTOMER` (
  `USERID` VARCHAR(22) NOT NULL, #用户 id
  `CNAME` VARCHAR(10) NOT NULL, #客户姓名
  `CSEX` CHAR(2) CHECK(`ASEX` IN ('男','女')), #性别，只能输入男或女
  `CAGE` INT UNSIGNED NOT NULL, #客户年龄
  `CTELENUMBER` VARCHAR(12) NOT NULL, #客户的电话号码
  `CADDRESS` VARCHAR(50) NOT NULL, #客户地址
  `CHILDNAME` VARCHAR(10) NOT NULL, #客户孩子姓名
  `GRADE` VARCHAR(15) NOT NULL, #客户孩子的就读年级
  PRIMARY KEY (`USERID`),
  #定义参照完整性,参照用户登录表的 userid，级联删除和更新
  FOREIGN KEY (`USERID`) REFERENCES `USERLOGIN`(`USERID`)
  ON DELETE CASCADE
  ON UPDATE CASCADE
);
```

表 3-8 CUSTOMERRESERVATION（客户预约表）

序号	字段	别名	数据类型	数据长度	是否主码	是否可空
1	NUMBER	编号	int	默认	是	否
2	AUSERID	家教 ID	varchar	22	外码	否
3	CUSERID	客户 ID	varchar	22	外码	否
4	SUBJECTS	上课科目	varchar	10	否	否
5	TIMEID	上课时间段	int	默认	外码	否
6	FEE	费用（元/ 小时）	int	默认	否	否
7	REMEDIALGOAL	补课目标	varchar	80	否	是

8	ASSESSMENT	应缴费用	int	默认	否	否
9	PAYMENT	是否缴费	varchar	2	否	否
10	AGREE	是否答应	varchar	2	否	否

其创建代码如下：

```
CREATE TABLE IF NOT EXISTS `CUSTOMERRESERVATION` (
  `NUMBER` INT AUTO_INCREMENT, #编号
  `AUSERID` VARCHAR(22) NOT NULL, #家教 id
  `CUSERID` VARCHAR(22) NOT NULL, #客户 id
  `SUBJECTS` VARCHAR(50) NOT NULL, #上课科目
  `TIMEID` INT UNSIGNED NOT NULL, #上课的时间段
  `FEE` INT NOT NULL, #上课的费用 （元/小时）
  `REMEDIALGOAL` VARCHAR(80), #补课目标描述
  `ASSESSMENT` INT NOT NULL, #应缴费用
  `PAYMENT` VARCHAR(2) NOT NULL CHECK(`OPPOINTMENT` IN ('是','否')), #客
  户是否缴费
  `AGREE` VARCHAR(2) NOT NULL CHECK(`OPPOINTMENT` IN ('是','否')), #家教
  是否此项预约单
  PRIMARY KEY (`NUMBER`), #主键为家教 id, 和客户 id
  #定义参照完整性, 参照用户登录表的 userid, 级联删除或更新
  FOREIGN KEY (`AUSERID`) REFERENCES `USERLOGIN` (`USERID`)
  ON DELETE CASCADE
  ON UPDATE CASCADE,
  #定义参照完整性, 参照用户登录表的 userid, 级联删除或更新
  FOREIGN KEY (`CUSERID`) REFERENCES `USERLOGIN` (`USERID`)
  ON DELETE CASCADE
  ON UPDATE CASCADE,
  #定义参照完整性, 参照时间段, 级联删除或更新
  FOREIGN KEY (`TIMEID`) REFERENCES `SCHOOLTIME` (`TIMEID`)
  ON DELETE CASCADE
  ON UPDATE CASCADE
);
```

表 3-9 TEACHINGSCHEDULE（教学安排表）

序号	字段	别名	数据类型	数据长度	是否主码	是否可空
1	NUMBER	编号	int	默认	是	否
2	AUSERID	家教 ID	varchar	22	外码	否
3	CUSERID	客户 ID	varchar	22	外码	否
4	SUBJECTS	上课科目	varchar	10	否	否
5	TIMEID	上课时间段	int	默认	外码	否
6	FEE	费用 （元/小时）	int	默认	否	否

7	REMEDIALGOAL	补课目标	varchar	80	否	是
8	COMPLETEDHOURS	已完成补课小时数	int	默认	否	否
9	FEEDBACKSCORE	反馈评分	int	默认	否	否
10	EVALUATE	上课评价	varchar	200	否	是

其创建代码如下：

```
CREATE TABLE IF NOT EXISTS `TEACHINGSCHEDULE` (
  `NUMBER` INT NOT NULL, #编号
  `AUSERID` VARCHAR(22) NOT NULL, #家教 id
  `CUSERID` VARCHAR(22) NOT NULL, #客户 id
  `SUBJECTS` VARCHAR(50) NOT NULL, #上课科目
  `TIMEID` INT UNSIGNED NOT NULL, #上课的时间段
  `FEE` VARCHAR(10) NOT NULL, #上课的费用 （元/小时）
  `REMEDIALGOAL` VARCHAR(80), #补课目标描述
  `COMPLETEDHOURS` INT NOT NULL, #已完成的补课小时数
  `FEEDBACKSCORE` INT NOT NULL, #每月客户填写一次反馈评分
  `EVALUATE` VARCHAR(200), #每月客户填写一次家教的上课评价
  PRIMARY KEY (`NUMBER`), #主键为家教 id, 和客户 id
  #定义参照完整性, 参照用户登录表的 userid, 级联删除或更新
  FOREIGN KEY (`AUSERID`) REFERENCES `USERLOGIN` (`USERID`)
  ON DELETE CASCADE
  ON UPDATE CASCADE,
  #定义参照完整性, 参照用户登录表的 userid, 级联删除或更新
  FOREIGN KEY (`CUSERID`) REFERENCES `USERLOGIN` (`USERID`)
  ON DELETE CASCADE
  ON UPDATE CASCADE,
  #定义参照完整性, 参照时间段, 级联删除或更新
  FOREIGN KEY (`TIMEID`) REFERENCES `SCHOOLTIME` (`TIMEID`)
  ON DELETE CASCADE
  ON UPDATE CASCADE
);
```

3.5.2 数据库触发器

触发器作为一中特殊的存储过程，与表紧密相连，可以看作表定义的一部分，当用户修改指定表或者视图中的数据时，触发器将会自动执行。本系统由于其复杂性，必须要使用触发器机制来处理相关的数据库表的操作，本系统一共创建了 2 个触发器，其核心代码如下：

（1）该触发器用于填写已完成时间和评分评价时生成家教综合评分，更新教师空闲时间表：

```
DROP TRIGGER IF EXISTS `UPDATE_TIMESLOT`;
CREATE TRIGGER `UPDATE_TIMESLOT`
AFTER UPDATE ON `TEACHINGSCHEDULE`
FOR EACH ROW
BEGIN
```

```

IF(NEW.COMPLETEDHOURS NOT LIKE '-1')
THEN
UPDATE `TUTOR`
SET `TUTOR`.SCORE = (`TUTOR`.SCORE*10+NEW.FEEDBACKSCORE)/11
WHERE `TUTOR`.USERID LIKE NEW.AUSERID;
UPDATE `TIMESLOT`
SET `TIMESLOT`.OPPOINTMENT = -1
WHERE `TIMESLOT`.OPPOINTMENT = NEW.TIMEID;
END IF;
END

```

（2）该触发器用于客户已缴费，并且家教同意预约后生成教学安排表，并更新教师空闲时间表

```

DROP TRIGGER IF EXISTS `INSERT_TEACHINGSCHEDULE`;
CREATE TRIGGER `INSERT_TEACHINGSCHEDULE`
AFTER UPDATE ON `CUSTOMERRESERVATION`
FOR EACH ROW
BEGIN
DECLARE C INT;
SET C = (SELECT MAX(`NUMBER`)+1 FROM `TEACHINGSCHEDULE`);
IF(NEW.`PAYMENT`='是')AND(NEW.`AGREE`='是')
THEN
INSERT INTO
`TEACHINGSCHEDULE`(`NUMBER`,`AUSERID`,`CUSERID`,`SUBJECTS`,`TIMEID`,`
FEE`,`REMEDIALGOAL`,`COMPLETEDHOURS`)
SELECT C,`AUSERID`,`CUSERID`,`SUBJECTS`,`TIMEID`,`FEE`,`REMEDIALGOAL`, -1
FROM `CUSTOMERRESERVATION`
WHERE `AUSERID` = NEW.`AUSERID` AND `CUSERID` = NEW.`CUSERID`;
UPDATE `TIMESLOT` aaa
SET aaa.OPPOINTMENT = new.timeid
WHERE EXISTS(
SELECT 1
FROM `CUSTOMERRESERVATION`,`SCHOOLTIME`
WHERE   aaa.USERID   LIKE   `CUSTOMERRESERVATION`.AUSERID   AND
`CUSTOMERRESERVATION`.TIMEID = `SCHOOLTIME`.TIMEID AND aaa.`WEEK`
LIKE `SCHOOLTIME`.`WEEK`
AND aaa.BEGINTIME <= `SCHOOLTIME`.BEGINTIME AND aaa.ENDTIME >=
`SCHOOLTIME`.ENDTIME
);
END IF;
END

```

3.5.3 数据库函数

数据库函数是指当需要分析数据清单中的数值是否符合特定条件时，使用数据库工作表函数：

删除函数，如果预约表是否缴费和是否同意都是是，则删除预约信息。

```
CREATE PROCEDURE timer()  
BEGIN  
DELETE FROM customerreservation  
WHERE customerreservation.AGREE LIKE '是'  
AND customerreservation.PAYMENT LIKE '是';  
END
```

3.6 模块划分

按照功能分析图将系统划分为如下几个模块：

对于客户：

- (1) 注册登录/注销功能；
- (2) 通过输入条件查询到年龄，地址，学历，有无任教经历，优势科目，评分对应的应聘者的所有信息，历史评价，以及有空闲的时间段；
- (3) 填写客户预约表（家教 id，客户 id，科目，时间段 id，补课费用（yuan/h），补课科目，补课目标），并进行缴费；
- (4) 通过自己的 id 查找所有教学安排表中未填写的完成的小时数进行填写，选填历史评价，通过评价生成反馈评分。

对于家教：

- (1) 注册登录/注销功能；
- (2) 通过自己的 id 查找预约信息（姓名，孩子名字，联系电话，地址，所处年级，时间段 id，补课费用（yuan/h），补课科目，补课目标），拒绝或答应；
- (3) 通过自己的 id 查找自己的教学安排评价记录（客户姓名，补课费用，时间段 id，完成的小时数，反馈评分，联系电话）；
- (4) 通过自己的 id 查找自己的工资（客户姓名，工资），家教工资=原始工资*完成时间*（反馈评分/100）。

对于管理员：

- (1) 审批应聘者申请表；
- (2) 确认客户已经缴费。

4 开发与实现

4.1 系统核心程序编写

4.1.1 访问 MySQL 数据库

连接数据库服务器，就是客户端向已运行的数据库服务器发出连接请求，成功后就可以对数据库进行访问，执行相应的操作。

Mapper.xml(Mybatis)，其中定义功能，对应要对数据库进行的那些操作，比如 insert、selectAll、selectByKey、delete、update。

4.1.2 用户登录

管理员，客户和家教都需要登录访问网站，客户登录才可以预约家教，并查看自己的信息和未评价的教学安排表，家教登陆才可以看到自己的教学历史记录和工资用户。登录的实现代码如下：

```
<script>
    function denglu(event){
        var userid = $("#form-username").val();
        var password = $("#form-password").val();
        alert("用户 id:"+userid+" 密码:" + password);
        var data;
        data= {
            'userid':userid,
            'password':password,
        };
        $.ajax({
            type : 'post',//使用 post 方法访问后台
            dataType : 'json',//返回 json 格式的数据
            async:false,
            url : '/zsmartcity-fortress-web/denglu.json',//要访问的后台地址
            data : data,//要发送的数据
            success : function(data) { //data 为返回的数据，在这里做数据绑定
                if(data.success){
                    alert(data.chenggong);
                    if(data.chenggong == 1){
                        alert("登录成功！");
                        window.open("https://www.baidu.com");
                    }else{
                        alert("用户名或密码不正确！");
                    }
                }else{
                    alert("返回失败");
                }
            },
            error : function() {
                alert("错误");
                alert(data.chenggong);
            }
        });
    }
};
```

```

    }
</script>
<select id="denglu" parameterType="com.ztesoft.zsmartcity.fortress.entity.Administrators"
resultType="java.lang.Integer">
SELECT COUNT(*)
FROM `userlogin`
WHERE `PASSWORD` =      #{password,jdbcType=VARCHAR}      AND      USERID      =
#{userid,jdbcType=VARCHAR}
</select>

```

4.1.3 用户注册

注册时对于家教和客户的填写要求不一样，具体要求如下：

对于客户：

```

<script>
function kehuzhuce(){
    var userid = $("#userid").val();
    var password = $("#password").val();
    var cname = $("#cname").val();
    var cage = $("#cage").val();
    var ctelenumner = $("#ctelenumner").val();
    var caddress = $("#caddress").val();
    var childname = $("#childname").val();
    var grade = $("#grade").val();
    if(userid == "" || password == "" || cname == "" || cage == "" || ctelenumner == "" ||
caddress == "" || caddress == "" || childname == "" || grade == ""){
        alert("输入有误，请重新输入（注意*为必填项）");
    }else{
        var data;
        data= {
            'userid':userid,
            'password':password,
            'cname':cname,
            'cage':cage,
            'ctelenumner':ctelenumner,
            'childname':childname,
            'caddress':caddress,
            'grade':grade,
        };
        $.ajax({
            type : 'post',//使用 post 方法访问后台
            dataType : 'json',//返回 json 格式的数据
            async:false,
            url : '/zsmartcity-fortress-web/yonghuzhuce.json',//要访问的后台地址
            data : data,//要发送的数据
            success : function(data) { //data 为返回的数据，在这里做数据绑定
                if(data.success){
                    alert(data.chenggong);
                }
            }
        });
    }
}

```



```

        if(data.chenggong == 1){
            alert("您的用户名已被注册，请重新输入其他用户名");
        }else{
            alert("插入数据成功！");
        }
    }else{
        alert("返回失败");
    }
},
error : function() {
    alert("错误");
    alert(data.chenggong);
}
});
}
}
</script>
<insert id="kehuzengjia" parameterType="com.ztesoft.zsmartcity.fortress.entity.Customer">
insert into customer (USERID, CNAME, CSEX, CAGE, CTELENUMBER, CADDRESS, CHILDNAME,
GRADE) values
({userid,jdbcType=VARCHAR},{cname,jdbcType=VARCHAR},{csex,jdbcType=CHAR},{cage,
jdbcType=INTEGER},{ctelenumner,jdbcType=VARCHAR},{caddress,jdbcType=VARCHAR},{c
hildname,jdbcType=VARCHAR},{grade,jdbcType=VARCHAR})
</insert>
对于家教:
<script>
function jiajiaozhuze(){
    var userid = $("#userid").val();
    var password = $("#password").val();
    var aname = $("#aname").val();
    var asex = $("#asex").val();
    var aage = $("#aage").val();
    var atelenumner = $("#atelenumner").val();
    var aaddress = $("#aaddress").val();
    var education = $("#education").val();
    var teachingexperience = $("#teachingexperience").val();
    var lastschool = $("#lastschool").val();
    var goodsubjects = $("#goodsubjects").val();
    var awards = $("#awards").val();
    var documentaddress = $("#documentaddress").val();
    if(userid == "" || password == "" || aname == "" || asex == "" || aage == "" ||
atelenumner == "" || aaddress == "" || education == ""
|| teachingexperience == "" || goodsubjects == "" || documentaddress == ""){
        alert("输入有误，请重新输入（注意*为必填项）");
    }else{
        var data;

```

```

data= {
    'userid':userid,
    'password':password,
    'aname':aname,
    'asex':asex,
    'aage':aage,
    'atelenumber':atelenumber,
    'aaddress':aaddress,
    'education':education,
    'teachingexperience':teachingexperience,
    'lastschool':lastschool,
    'goodssubjects':goodssubjects,
    'awards':awards,
    'documentaddress':documentaddress,
};
$.ajax({
    type : 'post',//使用 post 方法访问后台
    dataType : 'json',//返回 json 格式的数据
    async:false,
    url : '/zsmartcity-fortress-web/yonghu1zhuce.json',//要访问的后台地址
    data : data,//要发送的数据
    success : function(data) {//data 为返回的数据，在这里做数据绑定
        if(data.success){
            alert(data.chenggong);
            if(data.chenggong == 1){
                alert("您的用户名已被注册，请重新输入其他用户名");
            }else{
                alert("插入数据成功！");
            }
        }else{
            alert("返回失败");
        }
    },
    error : function() {
        alert("错误");
        alert(data.chenggong);
    }
});
}
}

<insert id="shenqingzhezengjia"
parameterType="com.ztesoft.zsmartcity.fortress.entity.Apply">
insert into apply (USERID, DOCUMENTADDRESS, ANAME, ASEX, AAGE, ATELENUMBER,
AADDRESS, EDUCATION, TEACHINGEXPERIENCE, LASTSCHOOL, GOODSUBJECTS, AWARDS )
values(#{userid,jdbcType=VARCHAR},#{documentaddress,jdbcType=VARCHAR},#{aname,jdbc
Type=VARCHAR},#{asex,jdbcType=CHAR},#{aage,jdbcType=INTEGER},#{atelenumber,jdbcType

```

```
=VARCHAR},{aaddress,jdbcType=VARCHAR},{education,jdbcType=VARCHAR},{teachingexperience,jdbcType=VARCHAR},{lastschool,jdbcType=VARCHAR},{goodssubjects,jdbcType=VARCHAR},{awards,jdbcType=VARCHAR}) </insert>
<delete id="shanchuyuyue" parameterType="com.ztesoft.zsmartcity.fortress.entity.Apply">
delete from apply where USERID like #{userid}
</delete>
```

如果管理员同意了家教的申请，那么删除申请表，将家教信息插入家教表。

```
<select
id="zhuceceshi"parameterType="com.ztesoft.zsmartcity.fortress.entity.Administrators"
resultType="java.lang.Integer"> SELECT COUNT(*) FROM `userlogin` WHERE USERID =
#{userid,jdbcType=VARCHAR}
</select>
```

这段代码用来检验用户名是否已经注册，如果用户名已经注册，会提示更换用户名。

```
<insert
id="zengjiakongxian"
parameterType="com.ztesoft.zsmartcity.fortress.entity.Timeslot">
insert into timeslot
<trim suffixOverrides="," suffix=")" prefix="(">
<if test="userid != null"> USERID, </if>
<if test="week != null"> WEEK, </if>
<if test="begintime != null"> BEGINTIME, </if>
<if test="endtime != null"> ENDTIME, </if>
<if test="oppointment != null"> OPPOINTMENT, </if>
</trim>
<trim suffixOverrides="," suffix=")" prefix="values (">
<if test="userid != null"> #{userid}, </if>
<if test="week != null"> #{week}, </if>
<if test="begintime != null"> #{begintime}, </if>
<if test="endtime != null"> #{endtime}, </if>
<if test="oppointment != null"> #{oppointment}, </if>
</trim>
</insert>
```

家教注册时填写有空的上课时间，最多五个时间段。

4.1.4 客户界面

1 查询并预约家教信息：

```
<select
id="jiaoshichaxun"parameterType="com.ztesoft.zsmartcity.fortress.entity.TutorSearch"result
Map="BaseResultMap">
select TUTOR.* from TUTOR,`TIMESLOT` WHERE `TIMESLOT`.USERID LIKE TUTOR.USERID
AND `TIMESLOT`.OPPOINTMENT = -1
<if test="lowestage != null"> and `TUTOR`.AAGE >=#{lowestage,jdbcType=INTEGER} </if>
<if test="highestage != null"> and `TUTOR`.AAGE <=#{highestage,jdbcType=INTEGER} </if>
<if test="education != null"> and `TUTOR`.EDUCATION like #{education,jdbcType=VARCHAR}
</if>
<if test="teachingexperience != null"> and `TUTOR`.TEACHINGEXPERIENCE like
#{teachingexperience,jdbcType=VARCHAR} </if>
<if test="aaddress != null"> and `TUTOR`.AADDRESS like concat(concat('%',#{aaddress}),'%')
```

```

</if>
<if test="goodsubjects != null"> and `TUTOR`.GOODSUBJECTS like
concat(concat('%',{goodsubjects}),'%') </if>
<if test="score != null"> and `TUTOR`.SCORE >=#{score,jdbcType=INTEGER} </if>
<if test="aname != null"> and `TUTOR`.ANAME like concat(concat('%',{aname}),'%') </if>
<if test="week != null"> and `TIMESLOT`.WEEK like #{week,jdbcType=VARCHAR} </if>
<if test="begintime != null"> and `TIMESLOT`.BEGINTIME <= #{begintime} </if>
<if test="endtime != null"> and `TIMESLOT`.ENDTIME >= #{endtime} </if>
</select>

```

客户可以通过条件查找到满足条件的家教进行预约，而游客只可以通过条件查找家教，不可以进行预约操作。

```

<insert
id="yuyuebiaozengjia" parameterType="com.ztesoft.zsmartcity.fortress.entity.Customerreservation"> insert into customerreservation (AUSERID, CUSERID, SUBJECTS, TIMEID, FEE,
REMEDIALGOAL, ASSESSMENT, PAYMENT,AGREE)
values(#{auserid,jdbcType=VARCHAR},#{cuserid,jdbcType=VARCHAR},
#{subjects,jdbcType=VARCHAR},
#{timeid,jdbcType=INTEGER},#{fee,jdbcType=INTEGER},#{remedialgoal,jdbcType=VARCHAR},
#{assessment,jdbcType=INTEGER},#{payment,jdbcType=VARCHAR},#{agree,jdbcType=VARCH
AR})
</insert>

```

客户填写完预约表后，数据库增加一条预约信息。

```

<select id="huodetimeid" resultType="java.lang.Integer"> select max(timeid)+1 from
SCHOOLTIME </select>

```

客户预约时间后，获得 timeid，每个 timeid 与预约的家教的空闲时间段中 oppointment 相联系。

```

<insert
id="yuyueshijian" parameterType="com.ztesoft.zsmartcity.fortress.entity.SchooltimeKey">
insert into schooltime (TIMEID, WEEK, BEGINTIME, ENDTIME) values
(#{timeid,jdbcType=INTEGER}, #{week,jdbcType=VARCHAR}, #{begintime,jdbcType=TIME},
#{endtime,jdbcType=TIME}) </insert>

```

获得客户预约的家教空闲时间，增加一条上课时间信息。

2 填写历史评价

```

<update
id="pingjiajiajiao" parameterType="com.ztesoft.zsmartcity.fortress.entity.Teachingschedule">
update teachingschedule
<set>COMPLETEDHOURS=#{completedhours,jdbcType=INTEGER},FEEDBACKSCORE=#{feedba
ckscore,jdbcType=INTEGER},
<if test="evaluate != null"> EVALUATE = #{evaluate,jdbcType=VARCHAR}, </if>
</set>
where NUMBER = #{number,jdbcType=INTEGER}
</update>

```

客户填写反馈未评分的教学安排，填写评分，选填评价。

3 客户缴费

```

<select
id="shouqianle" resultMap="BaseResultMap"> SELECT CNAME,customerreservation.*

```

```
FROM customerreservation,customer
WHERE PAYMENT = '否' AND CUSTOMER.USERID LIKE customerreservation.CUSERID
</select>
```

客户查看自己没有缴费的预约订单。

```
<update
id="yijingjiaole"
parameterType="com.ztesoft.zsmartcity.fortress.entity.Customerreservation">
update customerreservation
<set> PAYMENT = #{payment,jdbcType=VARCHAR} </set>
where NUMBER = #{number,jdbcType=INTEGER}
```

4.1.5 家教界面

1 查看家教个人信息

```
<select
id="gerenxingxi"parameterType="com.ztesoft.zsmartcity.fortress.entity.TutorSearch"resultM
ap="BaseResultMap"> select * from TUTOR WHERE USERID LIKE #{userid}
</select>
```

2 回应预约信息

```
<update
id="tongyiyuyue"parameterType="com.ztesoft.zsmartcity.fortress.entity.Customerreservatio
n">
update customerreservation
<set> AGREE = #{agree,jdbcType=VARCHAR} </set>
where NUMBER = #{number,jdbcType=INTEGER}
</update>
```

家教同意或者拒绝客户的预约信息。

```
<select id="jiaoshichakehu" resultMap="BaseResultMap">
Select CNAME,WEEK,BEGINTIME,ENDTIME,CUSTOMERRESERVATION.*
from CUSTOMERRESERVATION,CUSTOMER,SCHOOLTIME
WHERE CUSTOMERRESERVATION.CUSERID LIKE CUSTOMER.USERID AND AUSERID LIKE
#{auserid} AND CUSTOMERRESERVATION.TIMEID = SCHOOLTIME.TIMEID </select>
```

家教查看自己的预约信息。

3 查看教学安排

```
<select id="shangkeshijian" resultMap="BaseResultMap">
Select CNAME,WEEK,BEGINTIME,ENDTIME,TEACHINGSCHEDULE.*
from TEACHINGSCHEDULE,CUSTOMER,TIMESLOT
WHERE TEACHINGSCHEDULE.CUSERID LIKE CUSTOMER.USERID AND AUSERID LIKE #{auserid}
AND TEACHINGSCHEDULE.AUSERID = TIMESLOT.USERID </select>
```

在教学安排表中查看上课时间。

4.1.6 管理员界面

1 申请管理

```
<select
id="guanlichaxunjiajiao"parameterType="com.ztesoft.zsmartcity.fortress.entity.Apply"result
Map="BaseResultMap"> select * from apply </select>
管理员查看家教申请。
<delete id="shanchuyuyue" parameterType="com.ztesoft.zsmartcity.fortress.entity.Apply">
delete from apply where USERID like #{userid} </delete>
```

```
<insert id="tongyijiajiao" parameterType="com.ztesoft.zsmartcity.fortress.entity.Apply">
INSERT INTO `TUTOR`(`USERID`,`ANAME`,`ASEX`,`AGE`,`ATELENUMBER`,`ADDRESS`,
`EDUCATION`,`TEACHINGEXPERIENCE`,`LASTSCHOOL`,`GOODSUBJECTS`,`AWARDS`,`SCORE`)
SELECT`USERID`,`ANAME`,`ASEX`,`AGE`,`ATELENUMBER`,`ADDRESS`,`EDUCATION`,`TEACHI
NGEXPERIENCE`,`LASTSCHOOL`,`GOODSUBJECTS`,`AWARDS`,80 FROM `APPLY` WHERE
`USERID` like #{userid}; </insert>
```

管理员同意申请，删除申请，家教信息插入家教表。

2 缴费管理

```
<select id="shangkeshijian" resultMap="BaseResultMap">
```

```
<update
```

```
id="yijingjiaole"
```

```
parameterType="com.ztesoft.zsmartcity.fortress.entity.Customerreservation">
```

```
update customerreservation
```

```
<set> PAYMENT = #{payment,jdbcType=VARCHAR} </set>
```

```
where NUMBER = #{number,jdbcType=INTEGER}
```

3 客户信息管理

```
<select
```

```
id="kehuxingxi"parameterType="com.ztesoft.zsmartcity.fortress.entity.Customer"resultMap=
"BaseResultMap"> select * from CUSTOMER WHERE USERID LIKE #{userid}
```

4 家教信息管理

```
<select
```

```
id="gerenxingxi"parameterType="com.ztesoft.zsmartcity.fortress.entity.TutorSearch"resultM
ap="BaseResultMap"> select * from TUTOR WHERE USERID LIKE #{userid}
</select>
```

4.2 系统实施

4.2.1 用户登录注销功能

用户登录只需要输入账号和密码即可，登录后，按右上角的按钮点击登出可以注销。



图 4-1 登录界面

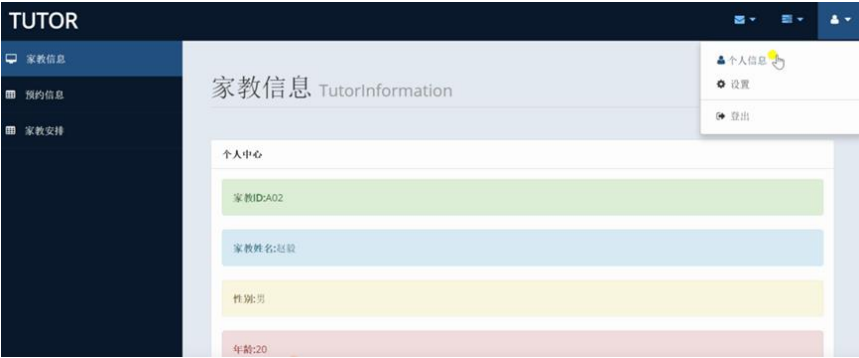


图 4-2 注销界面

4.2.2 用户注册功能

客户点击 APPLY-TUTOR 可以申请成为家教；家教点击首页 APPLY-CUSTOMER 可以申请成为家客户。



图 4-3 客户注册界面



图 4-4 家教注册界面

4.2.3 客户查询家教

客户登录成功之后，可以根据家教最高年龄，最低年龄，学历，有无家教经验，住址，教学学科，姓名，最低评分，上课开始或结束时间，上课星期这些条件来查找想要预约的家教并点击预约。如果是游客，只能按条件查找家教信息，不能进行预约。



图 4-5 家教查询界面

4.2.4 客户预约家教

预约信息里有家教的空闲时间，客户填写预约信息，并根据家教的空闲时间填写上课时间。

预约信息

重置按钮 返回按钮 提交按钮

补习科目 补习单价 补习总价

上课目标

补课星期 开始时间 结束时间

补课星期 开始时间 结束时间

补课星期 开始时间 结束时间

星期	开始时间	结束时间
星期三	08:00:00	09:00:00
星期六	08:00:00	09:00:00

图 4-6 预约界面

4.2.5 客户付费

客户缴纳费用给管理员代收，课程结束后管理员把工资给家教。若家教同意并且客户已经缴纳费用，则生成教学安排表。

CUSTOMER

家教查询 Tutor inquiry

家教评价 Tutor evaluation

客户缴费 Customer payment

客户缴费 Customer payment

家教姓名	是否同意	上课科目	上课每小时费用	总共应缴费用	缴费
李雅婷	是	化学	400	1600	缴费

图 4-7 客户付费页面 1

缴纳费用

家教姓名	是否同意	上课科目	上课每小时费用	总共应缴费用	缴费
李雅婷	是	化学	400	1600	缴费

您需要缴纳1600元补课费用

图 4-8 客户付费页面 2

4.2.6 用户填写教学评价

每完成一个月的课程，就填写一次对应的完成时间和家教评分，选填评价。



图 4-9 填写评价界面

4.2.7 家教回应预约

家教点击同意，这里有个定时任务，即过 10 秒删除一次家教同意并且客户已经缴纳费用的预约订单；若家教同意并且客户已经缴纳费用，则生成教学安排表；点击客户姓名可以看到客户个人信息。



图 4-10 家教回应预约界面

4.2.8 查看教学安排表

已完成时间为-1 时，说明一个月的课程还没有上完，客户还没有填写评价评分。



图 4-10 查看教学安排界面

4.2.9 家教申请管理

管理员查看家教的申请信息，如果审核通过，则点击通过，否则，点击拒绝。审核通过后，申请删除，家教信息插入家教表。



图 4-11 家教申请管理界面

4.2.10 客户缴费管理

管理员确认缴费，家教上完课程后可以得到相应的工资。



图 4-12 客户缴费管理界面

4.2.11 客户/家教修改信息



图 4-13 客户信息修改界面



图 4-14 家教信息修改界面

5 系统测试与应用

5.1 系统测试运行环境

系统运行环境有服务器端、因特网、客户端即浏览器构成。

5.1.1 服务器端运行环境

服务器端应由 tomcat 服务器、应用程序服务器、数据库服务器等组成。

服务器运行环境基本要求：操作系统为 Windows10，数据库服务器为 MySQL，

5.1.2 客户端（浏览器）运行环境

系统能够用于访问各类 Internet 的计算机，只要计算机能够支持 TCP/IP 协议，操作系统为 Windows 10，浏览器为 IE 11、Chrome 等都可以。

5.2 数据库测试与运行

5.2.1 登录界面

有唯一的管理人员账号和家教、客户的账号



图 5-1 主页界面



图 5-2 登录界面

5.2.2 客户/家教注册界面



图 5-3 客户注册界面

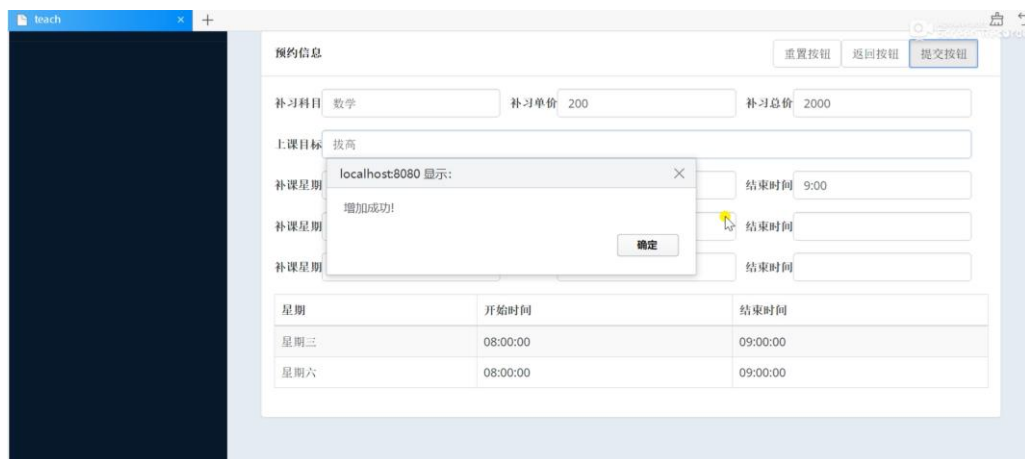


图 5-8 预约信息增加成功示意



图 5-9 选择缴费界面



图 5-10 评价成功界面

5.2.4 家教界面

查看家教个人信息和工资，回应预约信息，查看教学安排表。



图 5-11 家教信息页面



图 5-12 回应预约页面



图 5-13 教学安排页面

5.2.5 管理员界面



图 5-14 家教申请页面

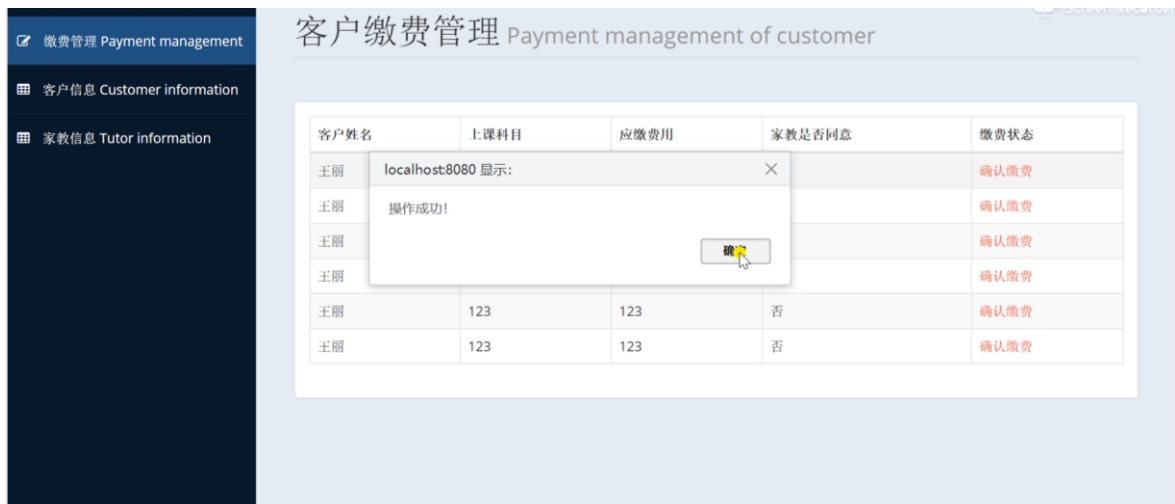


图 5-15 客户缴费确认提示

5.2.6 help 界面



图 5-16 使用说明界面

6 总结与展望

6.1 总结

本系统实现了如下核心功能：

对于客户：

- (1) 注册登录/注销功能，修改个人信息功能；
- (2) 另外申请成为家教，并有相应的家教功能；
- (3) 通过输入条件查询到年龄，地址，学历，有无任教经历，优势科目，评分对应的应聘者的所有信息，历史评价，以及有空闲的时间段；
- (4) 填写客户预约表（家教 id，客户 id，科目，时间段 id，补课费用（yuan/h），补课科目，补课目标）；
- (5) 通过自己的 id 查找所有教学安排表中未填写的完成的小时数进行填写，选填历史评价，通过评价生成反馈评分。

对于家教：

- (1) 注册登录/注销功能，修改个人信息功能；
- (2) 另外注册成为客户，并有相应的客户功能；
- (3) 通过自己的 id 查找预约信息（姓名，孩子名字，联系电话，地址，所处年级，时间段 id，补课费用（yuan/h），补课科目，补课目标），拒绝或答应；
- (4) 通过自己的 id 查找自己的教学安排评价记录（客户姓名，补课费用，时间段 id，完成的小时数，反馈评分，联系电话）；
- (5) 通过自己的 id 查找自己的工资（客户姓名，工资），家教工资=原始工资*完成时间*（反馈评分/100）。

对于管理员：

- (1) 审批应聘者申请表；
- (2) 确认客户已经缴费。

目前，本系统的大体核心功能都基本实现，但是整个系统的应用上来说还是有一定的不足，包括：

- (1) 管理员的功能不够完善，没有实现更加丰富的管理功能。
- (2) 界面设计有待改进，需要设计得更美观。

通过对本课题的研究以及原型系统的设计与开发，在进一步巩固了数据库知识的同时也大大提高了软件的开发与实践的能力。在开发实践过程中，本小组成员之间分工明确、相互合作，加强了成员之间的交流能力，提高了编程能力，同时也意识到团队的重要性以及软件开发标准的重要性。

6.2 展望

本系统面向广大学生家长和家教，如果能有一个足够的服务器，可以试着实施在本系统云端操作数据库和网页设计；希望本系统完善并取得实施，真正地造福广大学子。希望既能方便中小学生的课外学习生活，又可以给一些家教兼职的机会，得到多赢的结果。现今，高考制度让家长格外注重孩子的学科教育，尤其是语数外物化生等学科，但是，我们希望家教不仅能在这些考试学科上辅导学生，还可以在艺术方面帮助孩子陶冶情操，可以培养孩子在钢琴古筝素描国画等方面的兴趣；希望家教提供的课程也能够多种多样，让孩子接受多方面的指导。

参考文献

- [1] JavaScript 教程|菜鸟教程. www.runoob.com/js/js-tutorial.html
- [2] HTML/CSS/JavaScript 教程|W3school 在线教程. <http://www.w3school.com.cn/>
- [3] 项永明.大学生家教系统的设计与实现[J]. 吉林大学,2015
- [4] 喻浩.CSS+DIV 网页样式与布局从入门到精通[M].北京:清华大学出版社,2016.
- [5] 陈静.家教管理系统的设计与实现[J]. 山东大学,2011
- [6] 李海成.家教信息管理系统的设计与实现[D].电子科技大学,2011
- [7] 殷杰.基于 HTML 5 的跨平台移动家教信息服务系统的设计与实现[D].华南理工大学,2015
- [8] 李洁蕾.滇西地区家教信息服务平台的设计与实现[D].山东大学,2017
- [9] Database Management systems. Raghu Ramakrishnan,Johannes Gehrke. .2000
- [10] Query processing for SQL updates. Galindo-Legaria CA,,Stefani S,Waas F. SIGMOD Record . 2004