

**《软件工程》项目报告**

**题目： 驾校预约系统**

**课程名称： 软件工程**

**专业班级： 计算机科学与技术1707**

**学 号： U201714807**

**姓 名： 李俊欣**

**同组成员： 胡志强**

**苏俊杰**

**指导教师： 刘 宏**

**报告日期： 2019.11.05**

**计算机科学与技术学院**

# 任 务 书

**一 总体要求**

1. 综合运用软件工程的思想，协同完成一个软件项目的开发，掌软件工程相关的技术和方法；

2. 组成小组进行选题，通过调研完成项目的需求分析，并详细说明小组成员的分工、项目的时间管理等方面。

3. 根据需求分析进行总体设计、详细设计、编码与测试等。

**二 基本内容**

根据给出的题目任选一题，自行组队，设计与开发中软件过程必须包括：

**1. 问题概述、需求分析：**正确使用相关工具和方法说明所开发软件的问题定义和需求分析，比如NABCD模型，Microsoft Visio，StarUML等工具 (20%)；

**2. 原型系统设计、概要设计、详细设计**：主要说明所开发软件的架构、数据结构及主要算法设计，比如墨刀等工具（35%）；

**3. 编码与测试**：编码规范，运用码云等平台进行版本管理，设计测试计划和测试用例（30%）；

**4．功能创新**：与众不同、特别吸引用户的创新（10%）；

**5. 用户反馈**：包括用户的使用记录，照片，视频等（5%）。

**目录**

[任 务 书 I](#_Toc24667720)

[1 问题定义 1](#_Toc24667721)

[1.1项目背景与意义 1](#_Toc24667722)

[1.1.1 项目背景 1](#_Toc24667723)

[1.1.2 项目意义 1](#_Toc24667724)

[1.2 项目基本目标 2](#_Toc24667725)

[1.3 可行性分析 2](#_Toc24667726)

[1.3.1 Technically 2](#_Toc24667727)

[1.3.2 Operationally 3](#_Toc24667728)

[1.3.3 Economically 5](#_Toc24667729)

[1.4人员管理和项目进度管理 5](#_Toc24667730)

[1.4.1软件规模 5](#_Toc24667731)

[1.4.2人员管理 6](#_Toc24667732)

[1.4.3项目进度管理 6](#_Toc24667733)

[2 需求分析 8](#_Toc24667734)

[2.1 E-R图、数据流图 8](#_Toc24667735)

[2.2 用例图 9](#_Toc24667736)

[2.3 原型系统设计 10](#_Toc24667737)

[3 概要设计和详细设计 13](#_Toc24667738)

[3.1 系统结构 13](#_Toc24667739)

[3.1.1 功能说明 13](#_Toc24667740)

[3.1.2 接口设计 2](#_Toc24667740)0

[3.2 类图 20](#_Toc24667741)

[3.3关键数据结构定义 21](#_Toc24667742)

[3.4 关键算法设计 22](#_Toc24667743)

[3.5 数据管理说明 25](#_Toc24667744)

[4 实现与测试 26](#_Toc24667745)

[4.1实现环境与代码管理 26](#_Toc24667746)

[4.1.1软件实现 26](#_Toc24667747)

[4.1.2代码管理 26](#_Toc24667748)

[4.2 关键函数说明 27](#_Toc24667749)

[4.3 测试计划和测试用例 33](#_Toc24667750)

[4.3.1 测试方法 33](#_Toc24667751)

[4.3.2 模块一 用户注册模块-黑盒测试 33](#_Toc24667752)

[4.3.3 模块二 用户信息录入模块-白盒测试 36](#_Toc24667753)

[4.3.4 模块三 学员的预约课程模块-白盒测试 39](#_Toc24667754)

[4.4 结果分析 42](#_Toc24667755)

[5 总结 43](#_Toc24667756)

[5.1 用户反馈 43](#_Toc24667757)

[5.1 全文总结 44](#_Toc24667758)

[6 体会 46](#_Toc24667759)

[附录 48](#_Toc24667760)

# 1 问题定义

## 1.1项目背景与意义

### 1.1.1 项目背景

(Need 需求)随着生活质量的提高，为方便出行，车已经似乎成了家家必备的东西，因此越来越多的人开始在驾校报名学车考驾照。然而，当前驾校市场在学员学习各项科目的过程中，通常因为学员人数多而进度不一、教练和车辆不足等问题往往不好安排教学，使得学员浪费大量时间且效率不高、教练苦于教学安排。同时，传统的学员信息记录十分老旧、费时，并且没有实时性，没有一个这样的工具能方便学员随时更改个人信息和驾校集中管理学生信息。所以，基于以上需求，无论是学员还是驾校方都希望有这样一个可靠且方便的可以共享学车信息的平台。

### 1.1.2 项目意义

(Benefit 好处)在科技飞度发展的今天，几乎没有一个人或者一个企业能离开互联网得到发展，因此当下这样的产品正是应运而生的时候。根据项目背景分析，相比传统模型，我们对产品的初步设计将包含信息管理、教学安排、提升效率、节省时间和数据分析这样五方面分析亮点和优势：

一、方便驾校的人员信息管理：由于每个人都要使用手机注册信息，所以无论是驾校教练还是学员的基本信息都在平台的数据库中，对于驾校来说，非常利于其信息管理，同时这种实行实名制的方式也提高了安全性。

二、方便教练教学安排：通常驾校教学模式是多人同时教学，因为学习进度和接受能力不一样往往很麻烦，因此现在改变了预约方式之后，每个人每次进行一小时一对一的练习，针对性强了，教练方便教学。

三、提高学员学习效率：通常驾校教学模式是多人同时教学，而实际上，因为轮流实践的等待时间往往白白浪费。因此

现在改变了预约方式之后，每个人每次进行一小时一对一的练习，不仅针对性强了，而且节省了大量的时间。

四、节省时间：学生不再使用传统口头预约，教练直接在平台开放教学时间，学生直接在平台预约指定时间指定教练，双方均可登录即可看到课程信息、教练信息和学生信息，节省大量时间。

五、后台数据分析：对于以前驾校的操作模式，往往没有进行大规模的数据统计分析，因此在招收学生和制定学习计划的时候往往没有针对性，我们将我们的模型设计成以后台数据库为支撑的驾校管理平台，通过整理分析学员学习进度得出经验，方便驾校调整学院根据自身条件招收心仪的学生，以达到经济效益最大化。

## 1.2 项目基本目标

根据上述的项目意义和初步设计，根据时间进度分析，设定的项目完成时间大概是在一个月以内（10月底之前），做成在经济支持的情况下，撰写好项目报告，以及开发出能在驾校实际使用的中小型驾校预约平台。

## 1.3 可行性分析

(Approach 做法)在项目进行前，我们在技术、操作、经济三方面上进行了可行性分析认为我们的项目是可行的，并且得出了关于我们项目开发的独特思路，具体三方面分析如下：

### 1.3.1 Technically

根据市面上已有的类似预约平台，不论是医院的挂号平台、银行服务预约还是已有的简易驾校预约平台来看，搭建一个这样的预约平台并没有太多技术上的障碍。基于已有的市场实现，目前有的使用网页开发、App开发、以及现在越来越流行的微信小程序开发，兼顾各方面考虑，我们选择了使用QT集成工具和SQL Server进行开发。相比于其他的平台，我们选择QT从技术上来讲有如下优势：

1.QT有良好的跨平台性质

QT支持以下操作系统：Microsoft Windows85/98、Microsoft Windows NT、Linux、Solaris等等，因此在开发人员选择开发环境时可以倾向于自己更熟悉的。

2.QT的面向对象

QT的良好封装机制是的QT的模块化程度非常高，可重用性较好，对于用户开发来说是非常方便的。使用signals/slots的安全类型代替callback，使得各个元件之间的协同工作变得十分简单。

3.QT中丰富的API

QT包括多达250个以上的C＋＋类，还提供基于模板的collections、serialization、file、I/O device、directory management和date/time类。甚至还包括正则表达式的处理功能。

4.QT支持2D/3D图形渲染，支持OpenGL  
　　5.QT中大量的开发文档

帮助我们在项目开发过程中仅靠开发文档就能自己解决大量的难题，对于程序员来说不得不说是一个很好的福利。

而后台数据库我们使用SQL Server建立平台中的多张数据表以及布置它们之间的关系，它的优点是数据库表格可视化，方便管理，只需使用数据连接，通过简单的SQL命令语言就可实现和QT合作开发，同时它的索引创建是加快数据检索速度的一个最大支撑，同时使用优化隐藏器，提高系统的性能。

在SQL Server和QT合作开发下，我们使用打包工具进行打包成可执行文件，方便进行用户试用。考虑到经济问题，如在实际中使用，则只需在服务器上配置后台数据库，用户在本地操作后的信息上实时传到服务器上进行表的修改即可，这样就能实现多用户共同操作。

### 1.3.2 Operationally

我们选择上述的技术开发在Windows环境下使用的应用软件，只需要用户有一台PC机即可操作，另外在驾校中一般也会有机房，考虑资源的话没有问题。

对于用户来说，我们的应用设计是尽最大可能适应用户操作习惯的，在进行市场调研已存在的产品之后，我们在功能和界面的设计上更能得到用户的青睐。而对于驾校方面来说，我们提出了教练操作，将教练看作是另一类用户，扩大了使用范围使双方操作更灵活。同时特别区别于现在已有的产品，在数据分析实现驾校效益最大化方面做了另外的预留设计，设计在服务器端可实现管理人员的数据获取、分析、调整和版本更新处理，传统和我们设计的逻辑模型变化如图1-1和图1-2所示。

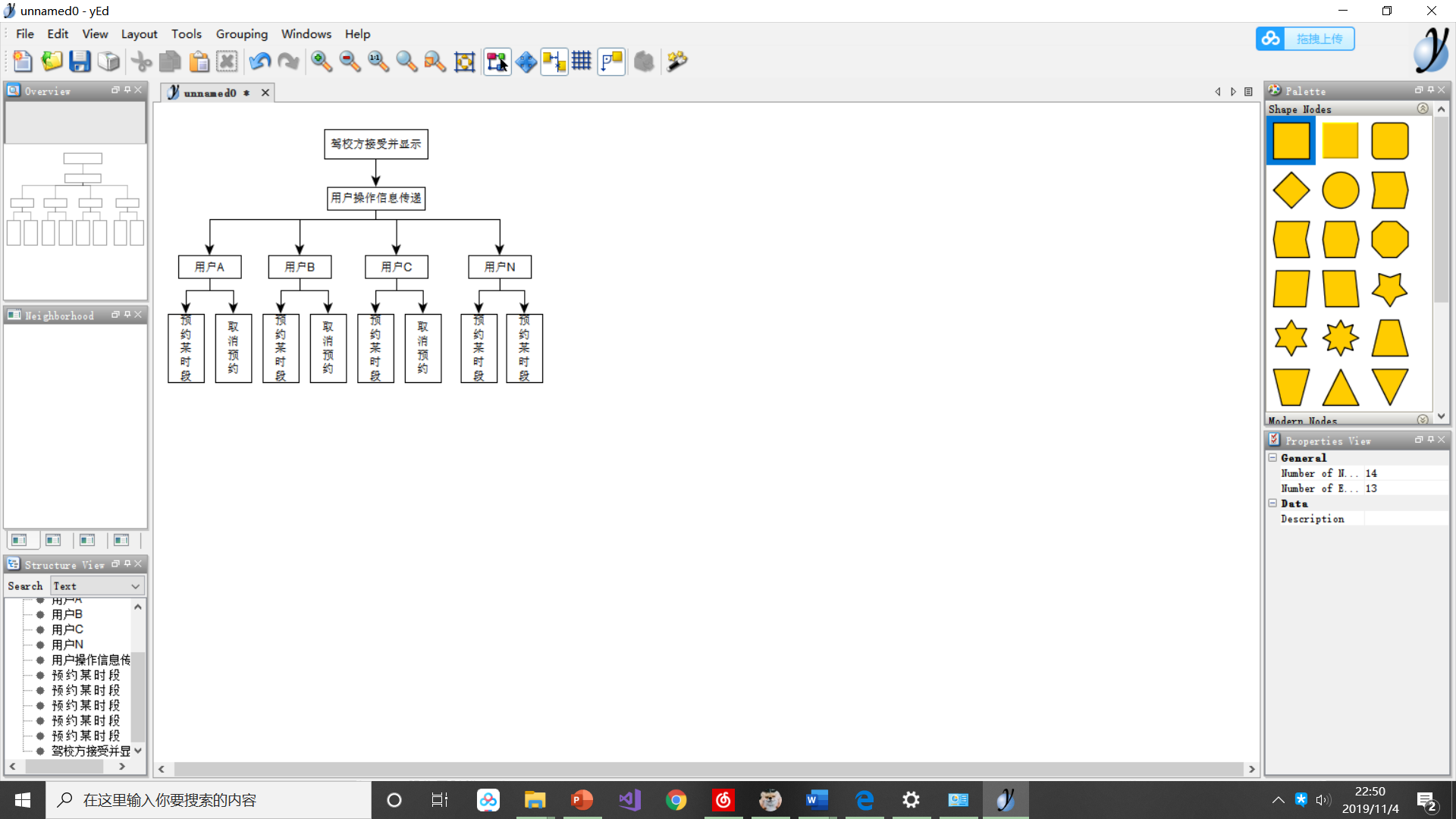
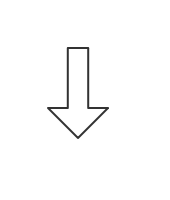


图 1-1现有的传统的逻辑模型



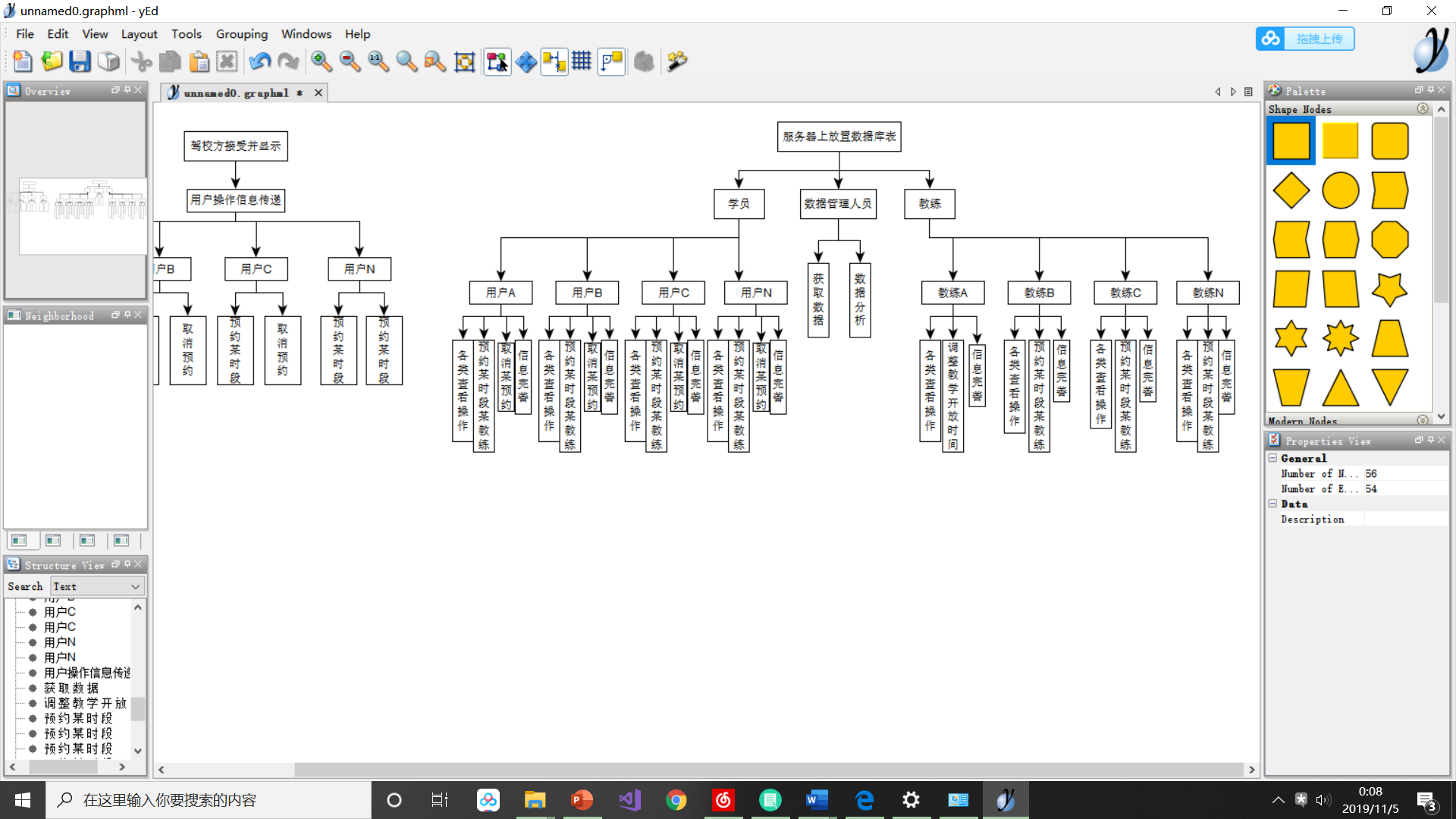


图1-2设计新的逻辑模型

(Competitors 竞争)目前存在不少类似的app和微信小程序，但这些平台仅仅只是将学员的预约信息告知教练，并没有为学员提供教练的信息、为教练提供学生信息。同时据调研反馈，还有很多用户体验不好的地方，比如预约取消不方便、对用户操作不友好、甚至有些平台还有bug，反而因为这些缺点常常让使用者感到头疼大过方便。另外，也有发展棘手的地方，比如很多驾校已经在使用自己的app或者嫌麻烦根本没有使用过类似的程序，所以平台前期发展较难。

(Delivery 推广)另外在商业上，当我们的初步模型建立成功后，如果可以，我们想要在学校的驾校点先进行推广，看看同学们的反应，并且收集意见反馈，不断进行改进。在小地方实施了之后，我们可以联系一些比较大的公司作为代理销售给其他地方的驾校，在有一定用户的基础之上，我们的推广就会比较容易了。当然了，我们的产品初始是不以盈利为目的的，这也将是推广我们的产品的优势之一。另外如果成功，我们再去向未使用过的驾校进行推广，讲出其先例的好处。最后，我们的推广一定要有反馈促使我们不断进步，所以将在软件内为用户设置反馈的链接，用户的意见将是我们宝贵的财富！

### 1.3.3 Economically

尽管我们的模式在现有模型上新增了很多，但是在现有的设备基础上还是可以得到实现的。

对于开发，现在网上有大量的学习资料和面向学生实践的软件可供使用，所以开发的经济成本比较小。即使是发布使用，需要用到服务器，现在各种云服务也给我们提供了这样的服务器租用，所以经济上来说是可行的。

## 1.4人员管理和项目进度管理

### 1.4.1软件规模

根据我们的设计和估计，这样一个驾校预约平台是一个中小型系统，对程序的规模大小进行估计，由每个人对程序的最小规模（a）、最大规模（b）和最可能的规模（m）分别算出平均值后，再利用下式计算程序规模的估计值，预估为5000LOC。



### 1.4.2人员管理

我们小组共三个人员，共同规划设计项目，在初期提出自己的想法，进行汇总讨论，在项目进行的过程中遇到问题及时交流、改进，这是项目的总线。

我们管理的方法依赖每周两次的讨论会和码云的代码管理。每周的讨论重在汇报自己的工作进行得如何，遇到的问题，大家能够一起当面帮助解决。而码云上我们更重在多人的代码管理，我们每个成员在平台上注册了自己的账号，我们团队建立了1个自己的仓库，三个人都加入到这个项目中。通过码云，我们将自己写好的代码上传到云上去，以供小组内其他成员下载修改或方便模块的连接。 而具体到每个人，根据每个人的兴趣、擅长的工作进行了如下划分，使得每个人能够尽自己的最大努力为项目做出贡献：

李俊欣：学员板块的开发和平台界面美化素材收集；

苏俊杰：教练板块的开发和远程数据库的连接；

胡志强：数据库的建立和界面优化设计。

报告部分，具体分工如下：

李俊欣：1 问题定义、2 需求分析、用户反馈、总结、体会

苏俊杰：3 概要设计和详细设计、总结、体会

胡志强：4 实现与测试、用户反馈、总结、体会

### 1.4.3项目进度管理

首先使用网络工程图将整个项目划分成子任务，并标明它们各个作业之间的依赖关系，网络工程图对进行系统分析和指定进度计划来说是一个强有力的工具。如图1-3所示。

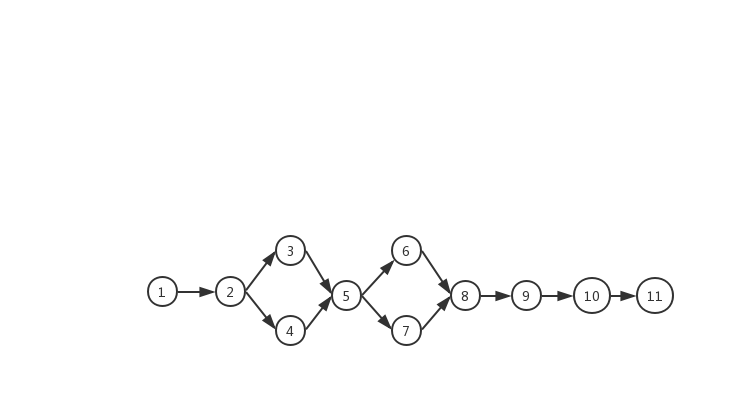


图1-3 网络工程图

图中：1-2 选题和分工 5-6 ui界面开发

2-3 市场调研 5-7 数据库建立

2-4 初步模型设计， 6-8 数据库连接  
写产品说明书

3-5 调研后模型改进 7-8 数据库连接

4-5 模型二次修改 8-9 测试

9-10 代码改进 10-11 试用推广，  
得到反馈，再次改进

根据时间预估，使用关键路径的方法，列出每一项子任务的LET（结束），EET（开始），持续时间和机动时间，如表1-1所示。

表1-1 项目工程网络中的时间进度（单位：天）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 作业 | LET（结束） | EET（开始） | 持续时间 | 机动时间 |
| 1-2 | 3 | 0 | 3 | 0 |
| 2-3 | 7 | 3 | 4 | 0 |
| 2-4 | 8 | 3 | 5 | 0 |
| 3-5 | 12 | 8 | 3 | 1 |
| 4-5 | 12 | 8 | 4 | 0 |
| 5-6 | 30 | 12 | 14 | 4 |
| 5-7 | 17 | 12 | 4 | 1 |
| 6-8 | 33 | 30 | 3 | 0 |
| 7-8 | 33 | 17 | 2 | 14 |
| 8-9 | 37 | 33 | 3 | 1 |
| 9-10 | 40 | 37 | 3 | 0 |
| 10-11 | 55 | 40 | 15 | 0 |

# 2 需求分析

## 2.1 E-R图、数据流图

根据问题定义和产品初步设计，我们分析出了驾校预约平台的ER图，用来说明项目规划中的实体，即一些具体或者抽象的概念，然后以一对一，一对多，多对多来说明它们之间的关系，如图2-1所示。

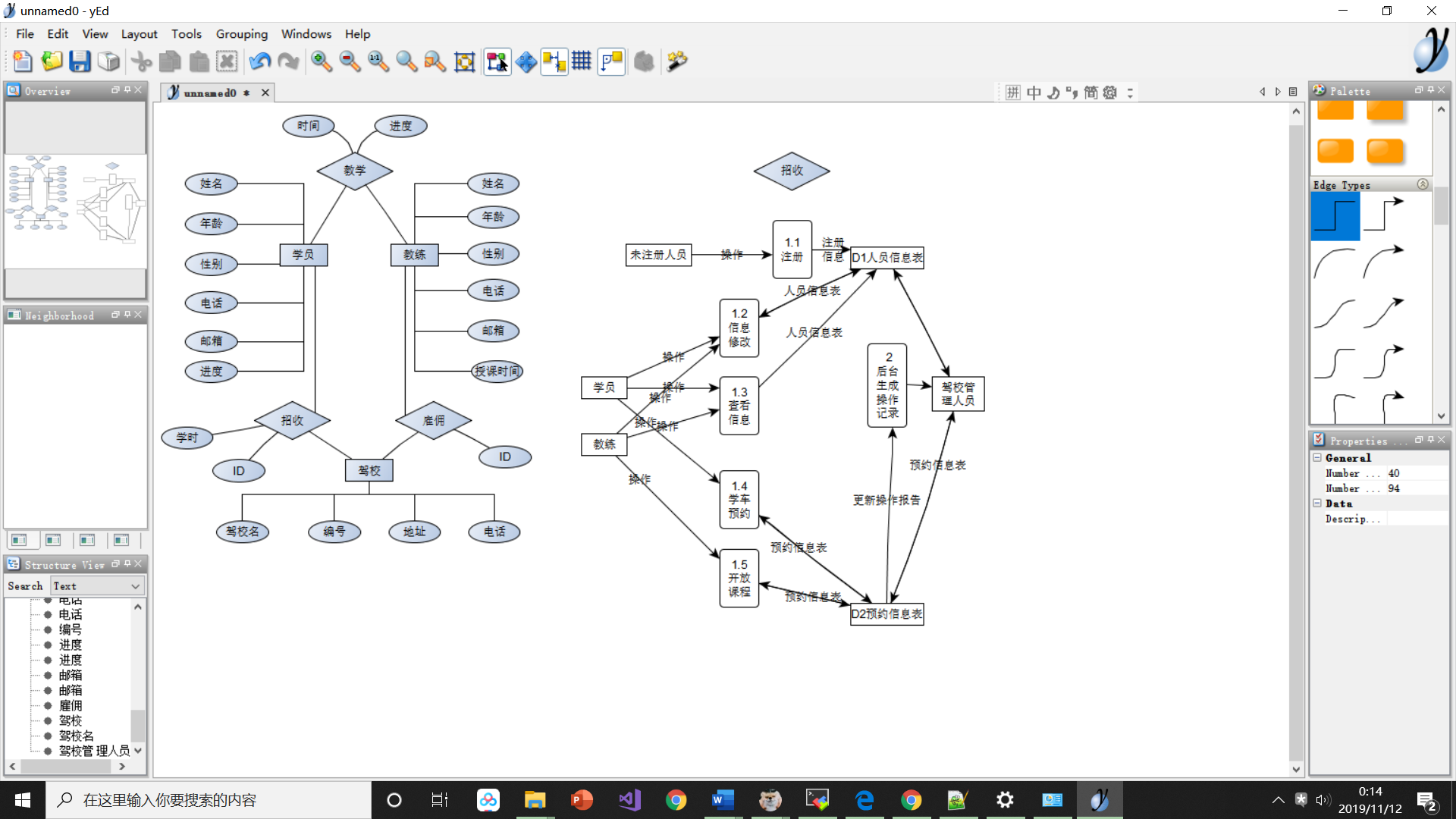


图2-1 项目ER实体图

数据流图是软件系统逻辑模型的一种图形表示，对我们的驾校预约平台来说，对数据的加工、变换，我们通过图来跟踪数据流，显示信息的输入和输出，如图2-2所示。

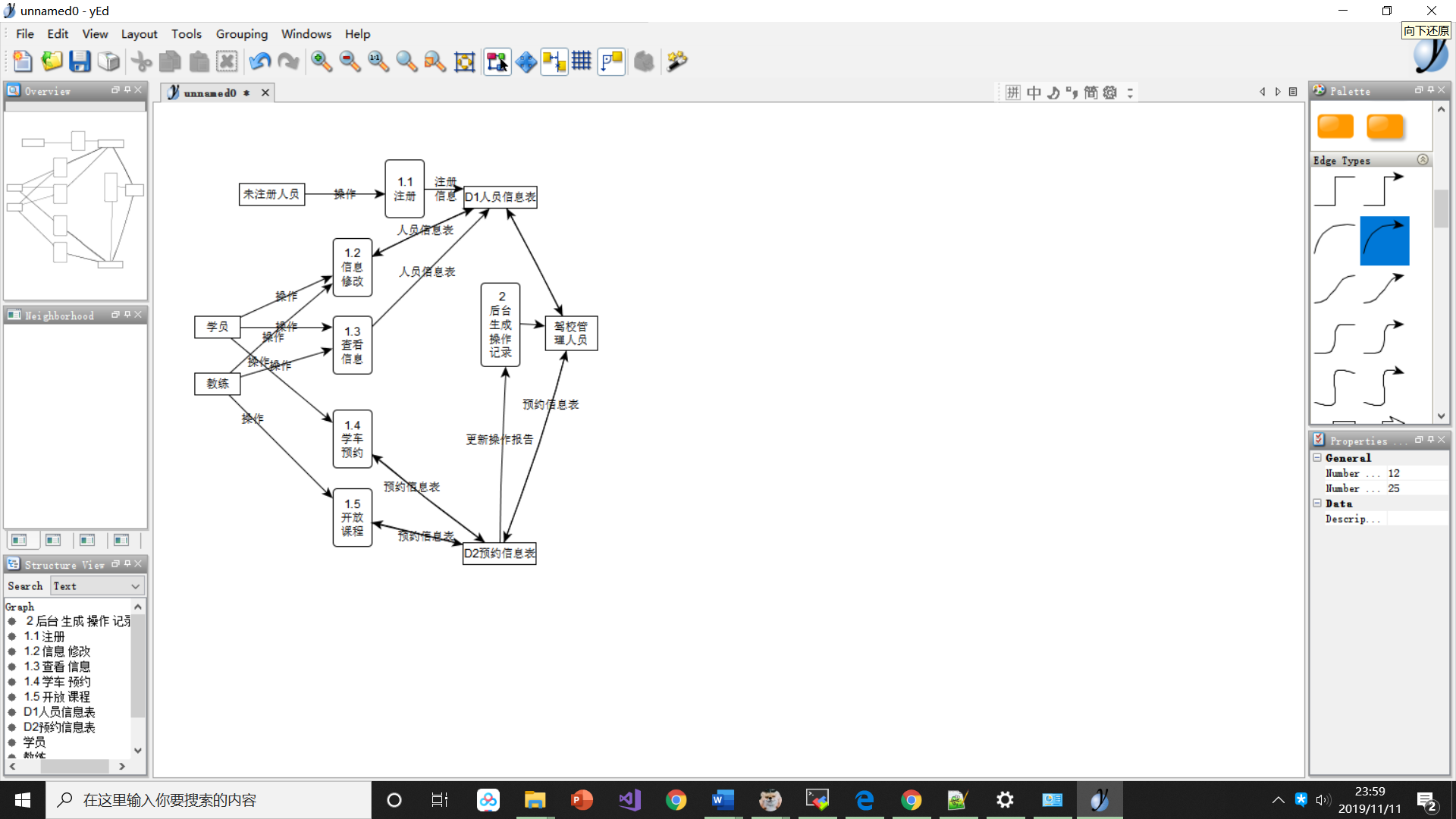


图2-2 预约平台数据流图

## 2.2 用例图

我们使用用例图2-3来更形象生动地指出参与者（Actor）、用例（Use Case）、边界以及它们之间的关系。我们设计的驾校预约平台中参与者主要为学员、教练、驾校管理人员、以及未注册人员。

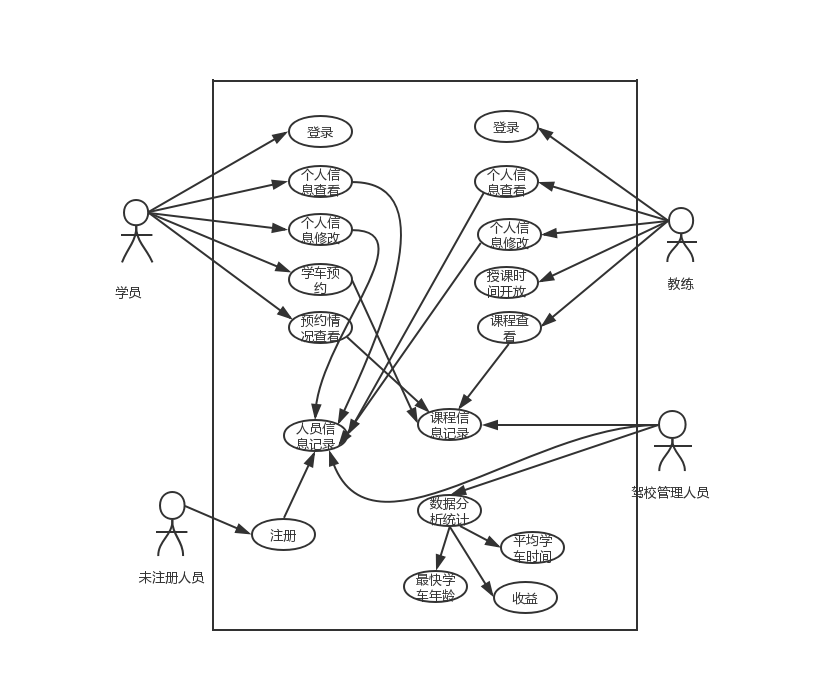


图2-3 项目用例图

## 2.3 原型系统设计

我们使用快速原型模型来建立我们的初步模型。快速原型的设计是不带反馈环的，所以我们初步模型的开发基本上是线性的，而且快速的。但是不可避免的是用户使用的过程中产生的反馈可能需要返回收集到需求阶段。我们在调研和开发完成后交予用户使用就是在进行这样的过程，如图2-4所示。

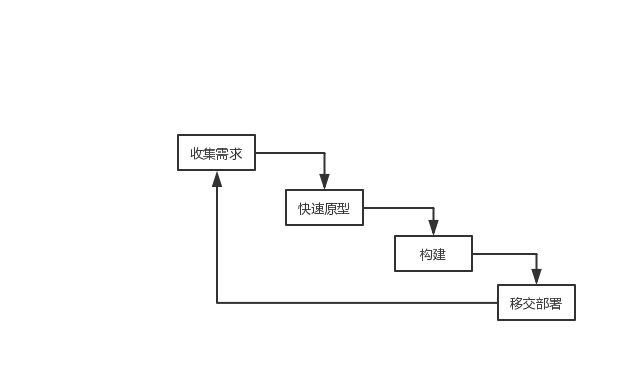


图2-4 快速原型模型

在主界面的地方设置三个按钮：“注册”、“登录”、“退出”；注册界面以框图的形式填写手机号码、密码、密码再次确认,用户点击“注册”按钮可实现注册；学员的界面主要分为三个板块：“我的信息”、“预约课程”、“预约情况”，其中“我的信息”中以“姓名”、“性别”、“邮箱”、“年龄”、“修改密码”、“确认密码”为主要内容，其中“预约课程”可从下拉框中选择教练，信息自动显示在信息框中，点击“点击预约”按钮即可实现预约，用户查看已经预约课程中的“预约课程”便可知道今明两天的课程安排；教练的界面主要分为三个板块：“我的信息”、“时间安排”、“预约情况”，其中“我的信息”中以“姓名”、“性别”、“邮箱”、“年龄”、“修改密码”、“确认密码”为主要内容，其中“时间安排”中设置16个按钮可以对今明两天中各8个时段进行开放或关闭时间操作，用户查看已经预约课程中的“预约课程”便可知道今明两天的课程安排。

我们利用windows界面ui对我们的平台进行了简单的设计（初步模型），下面是几个主界面，如图2-5至图2-13所示。

图2-5 主界面 图2-6 注册界面

图2-7 学员预约界面 图2-8 学员信息界面

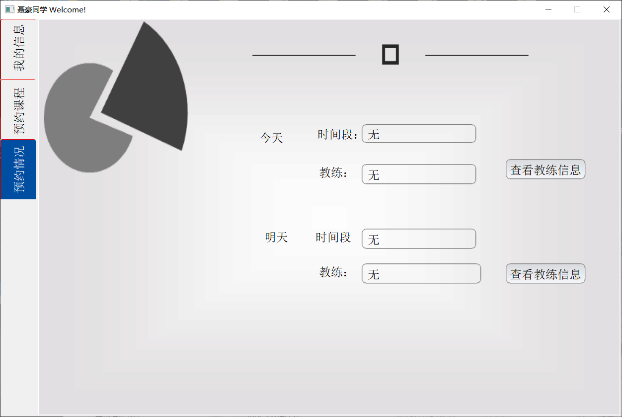
 

图2-9 学员预约情况界面 图2-10 学生预约时间界面

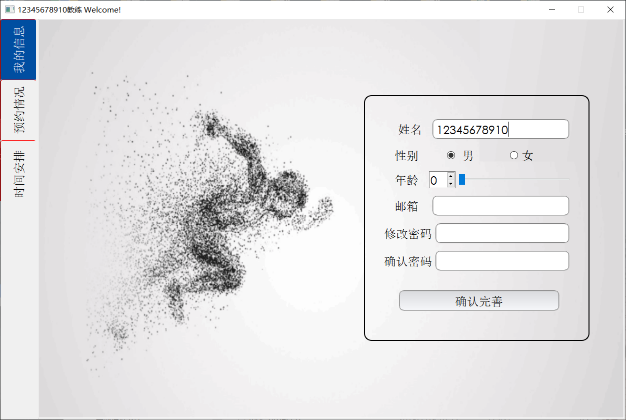
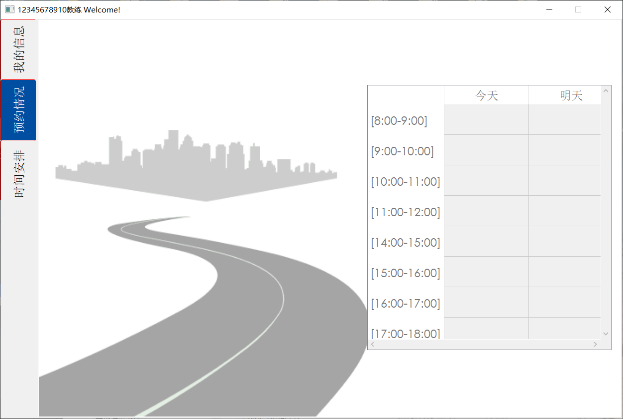
 

图2-11 教练信息界面 图2-12 教练课程安排界面



图2-13 教练时间开放界面

# 3 概要设计和详细设计

## 3.1 系统结构

### 3.1.1 功能说明

根据用户需求分析可得，本驾校预约系统应该包括五个系统，其中包括用户注册系统，用户登录系统，学员系统，教练系统，数据存储和检索系统。

下面用H图进行具体显示。（系统的H图，如图1-1所示）

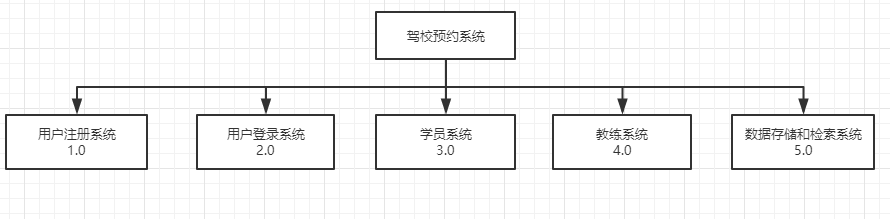


图3-1 H图-驾校预约系统

其中用户注册系统主要是获取用户输入的用户名和密码，将该用户的信息保存在数据存储系统中；用户登录系统主要是从数据存储系统检索出已注册的用户信息，并与查看与输入的用户信息是否匹配；学员系统主要是从数据存储系统中检索出学员信息和所能预约的教练信息，并将它们显示给用户，然后根据所能预约的教练信息进行学车预约；教练系统主要是从数据存储系统中检索出教练信息和本教练的已被预约的各个时间段信息，并叫它们显示给用户，本系统还能根据用户的需求，更改教练的空闲时间，并将它们存入数据存储系统。数据存储和检索系统主要是能对输入的信息进行检索、删除、更新。

具体的，五个系统的子模块如下图3-2所示。

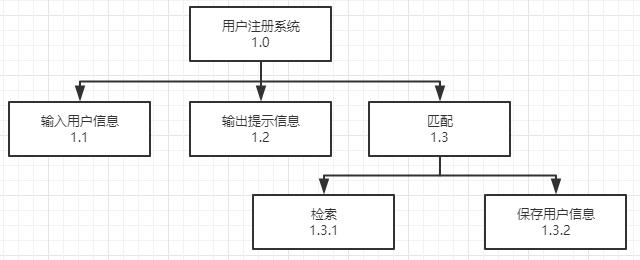


图3-2 H图-用户注册系统

用户注册系统主要是获取用户输入的用户名和密码，将该用户的信息保存在数据存储系统中。具体而言，本模块调用三个子模块，分别为输入用户信息、输出提示信息、匹配。输入用户信息子模块，主要是获取用户输入的用户名和密码；

输出提示信息子模块主要是在用户注册失败和注册成功的情况下，输出不同的提示信息来告知用户当前状态；匹配子模块主要是从数据存储系统中对输入的用户信息进行检索匹配，若当前输入的用户信息以前未注册，则将其保存在数据存储系统中。

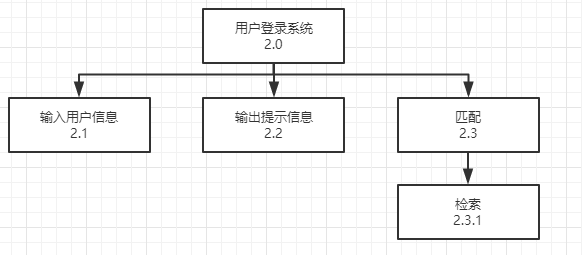


图3-3 H图-用户登录系统

用户登录系统主要是从数据存储系统检索出已注册的用户信息，并与查看与输入的用户信息是否匹配。该系统与用户注册系统的结构相似，只是在匹配子模块中从数据存储系统对输入的用户信息进行检索匹配，若以前未注册，直接输出提示信息，而不加以保存。

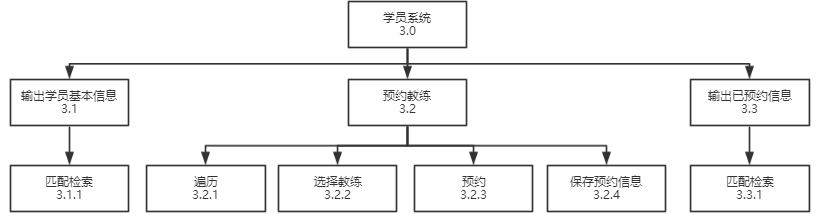


图3-4 H图-学员系统

学员系统主要是从数据存储系统中检索出学员信息和所能预约的教练信息，并将它们显示给用户，然后根据所能预约的教练信息进行学车预约。具体而言，本模块调用三个子模块，分别为输出学员基本信息、预约教练、输出已预约的信息。

输出学员基本信息子模块和输出已预约的信息子模块都是从数据存储系统中对该学员进行匹配检索，找到该学员的相关信息之后，将它们输出；预约教练子模块主要是在数据存储系统中对于所有教练进行遍历输出，学员能够选择一个教练输出他的空闲时间，根据他的空闲时间，学员可以每天预约一个教练的一个时间段，最后将这个预约信息保存到数据存储系统中。

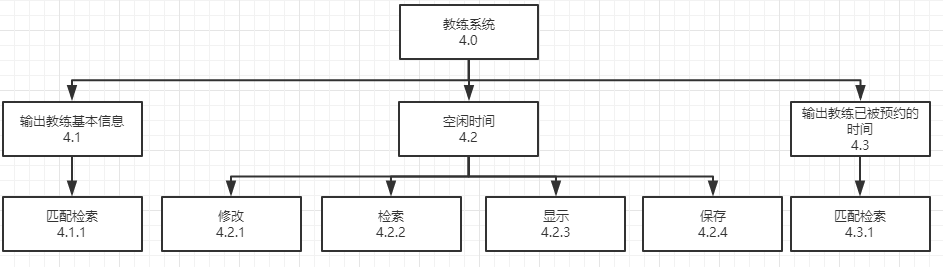


图3-5 H图-教练系统

教练系统主要是从数据存储系统中检索出教练信息和本教练的已被预约的各个时间段信息，并叫它们显示给用户，本系统还能根据用户的需求，更改教练的空闲时间，并将它们存入数据存储系统。具体而言，本模块调用三个子模块，分别为输出教练基本信息、空闲时间、输出教练已被预约的时间。

输出教练基本信息子模块和输出教练已被预约的时间子模块都是从数据存储系统中对该教练进行匹配检索，找到该教练的相关信息之后，将它们输出；空闲时间子模块主要是从数据存储系统中检索出该教练的空闲时间，并将其显示出来，用户可以根据自身的实际情况对这个空闲时间进行修改、保存。

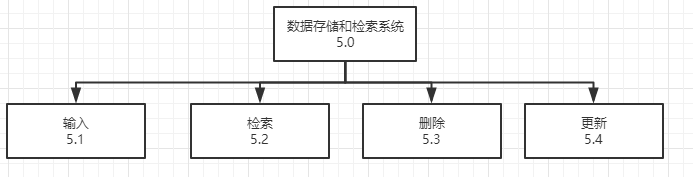


图3-6 H图-数据存储和检索系统

数据存储和检索系统主要是能对输入的信息进行检索、删除、更新。具体而言，本模块调用四个子模块，分别为输入、检索、删除、更新。

输入子模块主要是获取输入的数据；检索子模块主要是对于输入的数据在系统中进行检索；删除子模块主要是对于检索到的系统中的数据进行选择删除；更新子模块主要是更新系统中的数据。

### 3.1.2 接口设计

下面用IOP表来表示五个模块的处理过程。

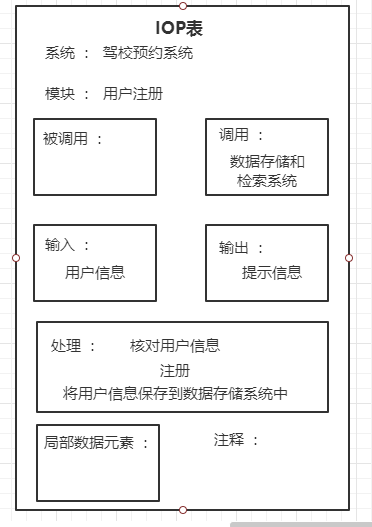


图3-7 IOP表-用户注册系统

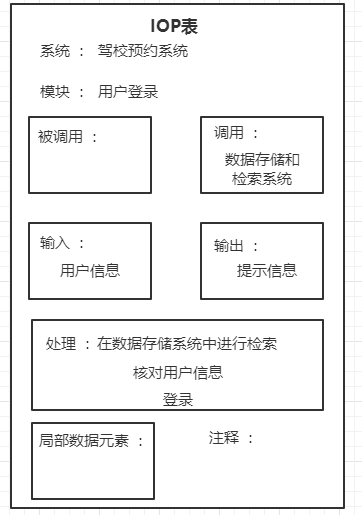


图3-8 IOP表-用户登录系统

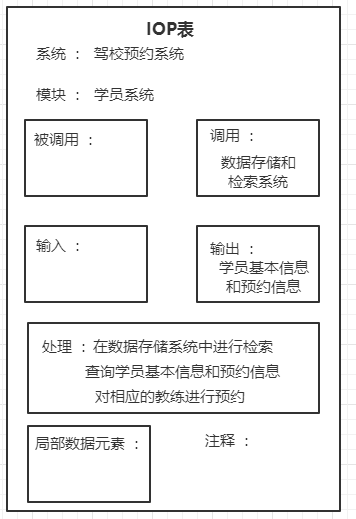


图3-9 IOP表-学员系统

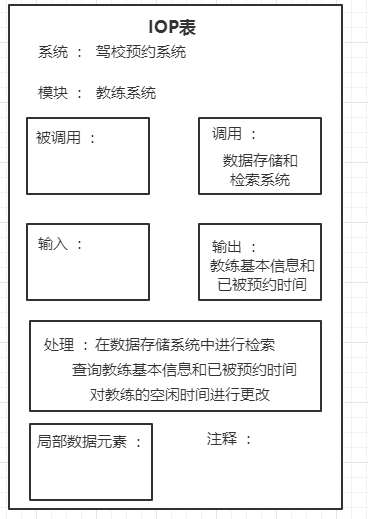


图3-10 IOP表-教练系统

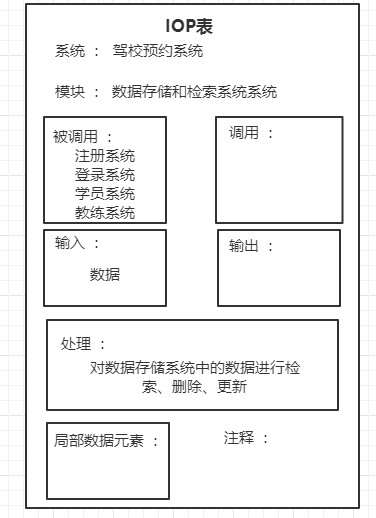


图3-11 IOP表-数据存储和检索系统

## 3.2 类图

运用类图等方法说明系统的设计。

本系统这主要提供窗口图形化显示，相应的功能的实现根据用户对窗口的相关操作，如点击、编辑文本等，来做相应的操作。（下面为本系统的类图描述）

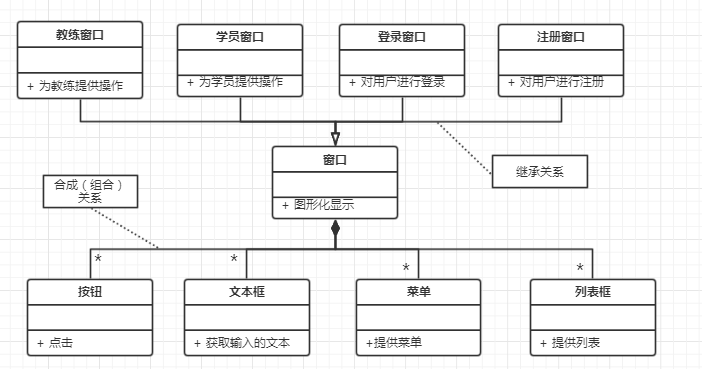


图3-12 类图-驾校预约系统

## 3.3关键数据结构定义

本系统主要包括四类数据，分别为学员基本信息、学员的预约信息、教练的基本信息和教练的空闲时间。为了更好地直观地描述数据，下面用Jackson图来显示各类数据。

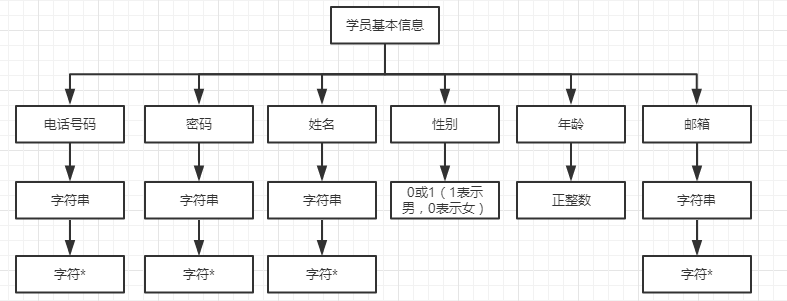


图3-13 Jackson图-学员基本信息

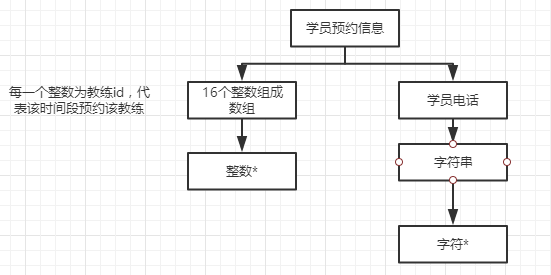


图3-14 Jackson图-学员预约信息

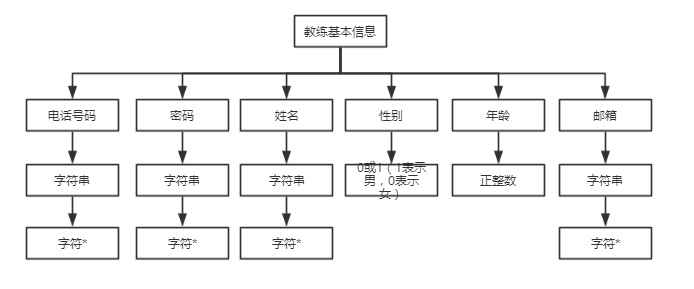


图3-15 Jackson图-教练基本信息

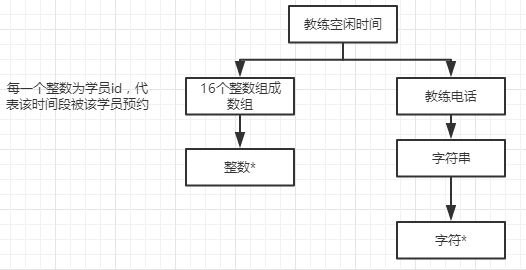


图3-16 Jackson图-教练空闲时间

## 3.4 关键算法设计

用户注册系统首先是判断输入的用户名是否是11位（此系统以电话号码作为用户名）和密码是否大于3位，若不满足，输出相关提示；若满足，再判断输入的用户名是否为空，若为空，输出相关提示；若不为空，再判断确认密码和输入的密码是否相等，若相等，则将用户信息保存到数据存系统中；若不相等，输出相关提示。（具体流程图如图3-17所示）

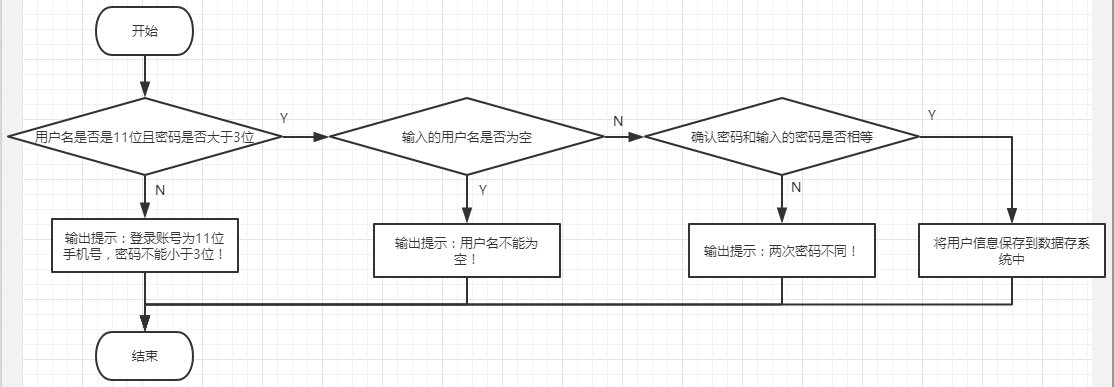


图3-17 流程图-用户注册系统

用户登录系统首先判断输入的用户名和密码是否为空，若为空，则输出相关提示信息；若不为空，则再在数据存储系统中查询该用户名和密码是否已经注册或者密码是否正确，若登录成功，则跳转相关界面，登录成功；若登录失败，则输出相关提示信息。（具体流程图如图3-18所示）

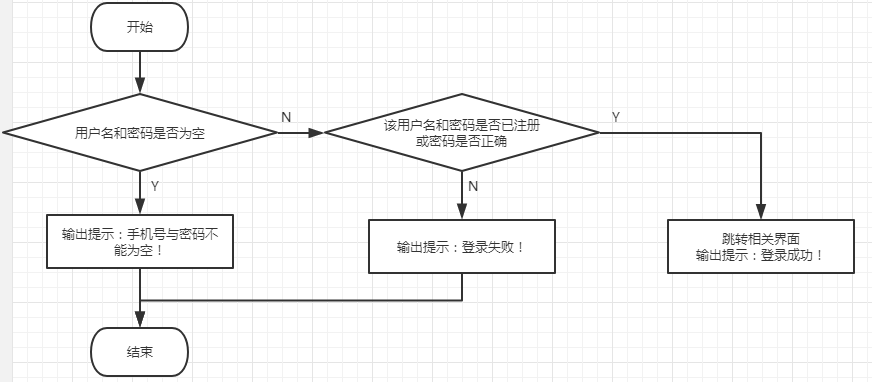


图3-18 流程图-用户登录系统

学员系统首先在数据存储系统中检索出该学员的所有信息，并将其输出，然后循环判断输入的操作，若为退出，则退出程序；若为修改基本信息，则进行相关信息的修改；若为修改预约信息，则进行相关教练的学车预约。（具体流程图如图3-19所示）

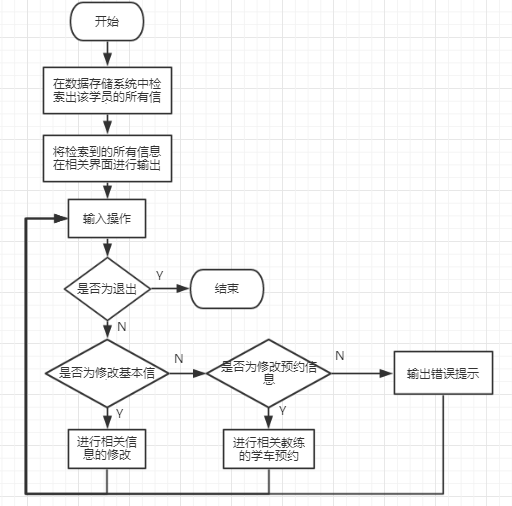
****

图3-19 流程图-学员系统

教练系统首先在数据存储系统中检索出该教练的所有信息，并将其输出，然后循环判断输入的操作，若为退出，则退出程序；若为修改基本信息，则进行相关信息的修改；若为修改空闲时间信息，则进行相关的空闲时间的修改。（具体流程图如图3-20所示）

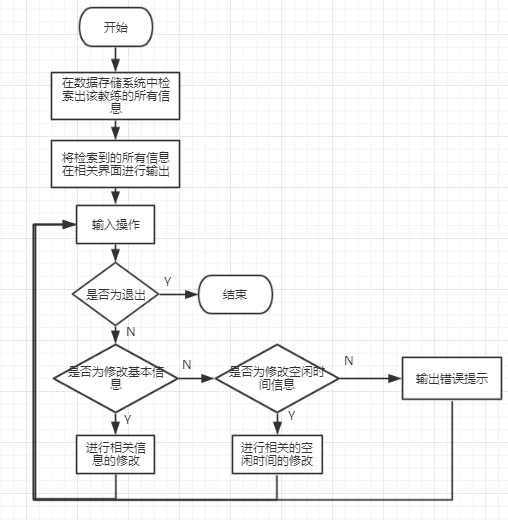


图1-20 流程图-教练系统

数据存储和检索系统主要是用数据库实现，所以基本是用sql语句实现的，例如insert语句、update语句、select语句等，流程图就在这做过多演示了。

## 3.5 数据管理说明

本系统为了更好地存储、检索和访问数据，首先是采用了本地关系型数据库Mysql或者Sql-Server，进行初期的软件设计，并用sql语句中的update进行数据更新，insert语句进行数据的插入，select语句进行数据的检索访问等。后期将租用云端数据库，方便多用户同时操作，而不必在本地安装数据库应用程序。

# 4 实现与测试

## 4.1实现环境与代码管理

### 4.1.1软件实现

本次开发软件为驾校预约系统，即实现学员与教练之间的课程的管理与时间的合理利用。本次软件开发的前端使用的是基于C++语言的Qt Creator 4.10.0 (Community)进行编写以及交互界面的UI设计，后端则是使用SQL Server2017进行数据的保存、管理以及更新。同时，使用ODBC数据源进行两者的连接，使用QT中自带的SQL模块进行两者之间的数据的传送。

### 4.1.2代码管理

本次开发目前共实现九个模块分别是注册模块、登录模块，教练的完善信息、查看预约情况、修改可预约时间模块以及学员的完善信息、查看教练信息、预约课程以及查看预约信息模块，加上后续的界面优,即共签入了十次代码，码云平台签入记录如图4-1，4-2所示。



图4-1 仓库代码



图4-2 码云平台代码签入记录

注：此处签入次数与代码文件总数不同，是因为小组成员刚开始对git上传不太不熟悉，出现的操作失误导致的。

## 4.2 关键函数说明

(1)插入数据函数

函数名称：insert(table\_name, item\_name，item\_values)

函数输入：表格名称，表格中插入列的名称，以及对应的插入的值。

函数输出：插入成功返回插入的行数。

函数功能：向数据库对应的表格和相应的列中插入数据，插入成功返回插入的行数，失败则返回0。

(2)更新数据函数

函数名称：update(table\_name, item\_name，item\_values,line\_id)

函数输入：表格名称，表格中要更新列的名称，要更新的值，以及要更新的行号。

函数输出：更新成功则返回被操作的记录条数

函数功能：向数据库对应的表格中和指定的行和列处更新替换数据，更新成功则返回操作记录数，失败则返回0。

(3)查询数据函数

函数名称：select(table\_name, item\_name，line\_id)

函数输入：表格名称，所要查询的数据列名（\* 表示查询改表格的所有列的数据）以及要查询的行号。

函数输出：查询各项的数据

函数功能：查询数据库中对应表格的某一行的数据，查询成功则可返回对应的数据，失败返回null。

(4)数据库语句的执行函数

函数名称：exec(QString)

函数输入：待执行的数据库语句

函数输出：执行成功返回true，执行失败返回false

函数功能：在前端中使用，执行需要操作的数据库函数，完成对应的插入，更新和查找数据的功能，供前端使用。执行成功返回true，失败返回false。

(5)注册函数

函数名称：void apply::**on\_pushButton\_clicked**()

函数输入：注册按钮的点击信号

函数输出：无

函数功能：读取ui界面对应的文本信息，根据checkbutton按钮确实是学员还是教练，从而来判断对要对数据库中哪张表格进行操作。其次根据用户名及密码的文本判断输入是否合乎规范，两次密码是否相同，若都满足要求则只需数据库相关语句，并清空文本信息，并发送返回主页面信号。该函数调用了上述的数据插入函数以及数据库语句操作函数。

函数流程如图4-3所示。

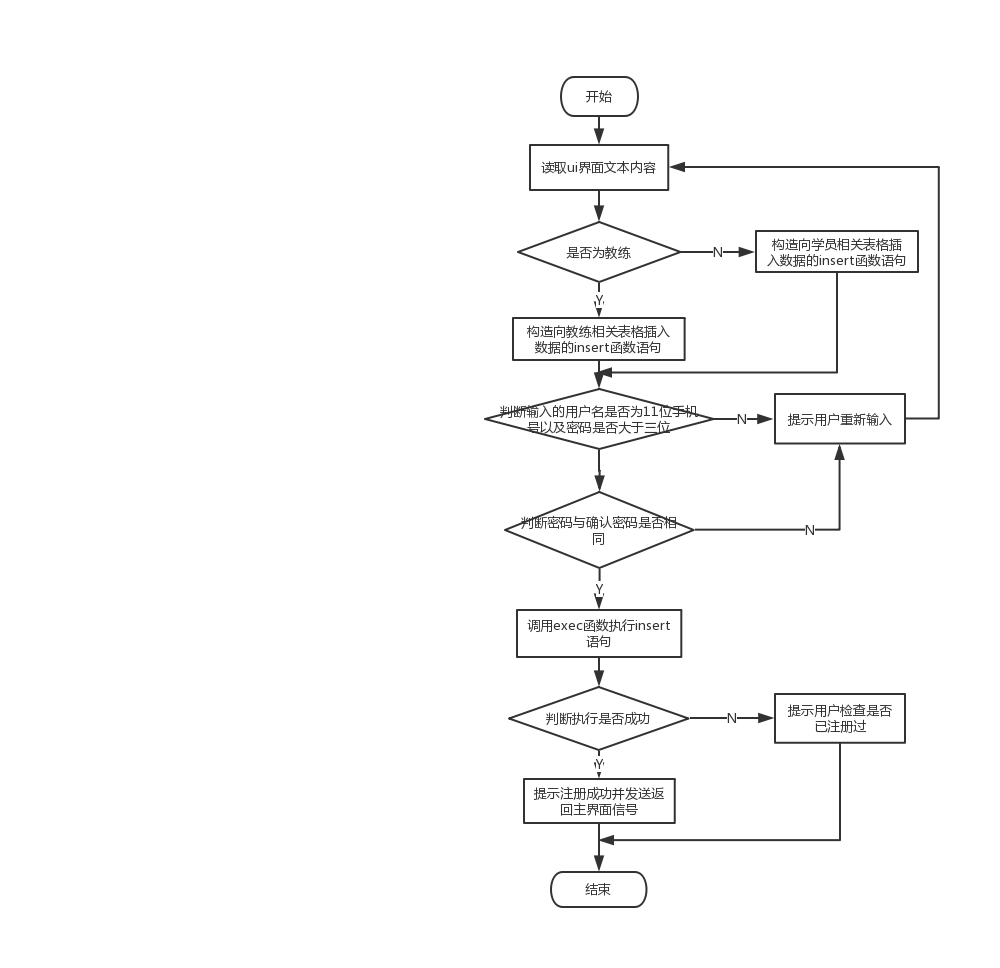


图4-3 注册函数

(6)登录函数

函数名称：void login::**on\_pushButton\_clicked**()

函数输入：登录按钮的点击信号

函数输出：无

函数功能：读取ui界面的文本信息，首先判断是否为空，为空则提示用户输入；再确定用户身份，从而构造参数不同的数据库select函数，从数据库中查询此用户。若查询成功，即密码与用户名相匹配，登录成功，在根据不同的身份进入不同的操作界面。同时，使用全局变量记录该用户在数据表格中的行号，以供后续界面使用。查询失败则提示用户登录失败。此函数调用了数据库函数select函数，以及exec()函数来执行数据库函数语句。

(7)教练的修改空闲时间函数 (有多个相同的点击槽函数，故只说明一处)

函数名称：void coach\_deal::**on\_pushButton\_1\_1\_clicked**()

函数输入：按钮的点击信号

函数输出：无

函数功能：读取对应的按钮的文本信息，判断是否已被预约，若是则无法操作；否则再判断当前系统时间是否超过该课程时间，若超过，提示用户；若没有超过则再进行是否为有空的判断，若有空，点击则调用数据库update函数修改对应的时间段信息，同时更新按钮的文本信息；若为无空，同理可进行更新为有空。函数流程如图4-4所示。

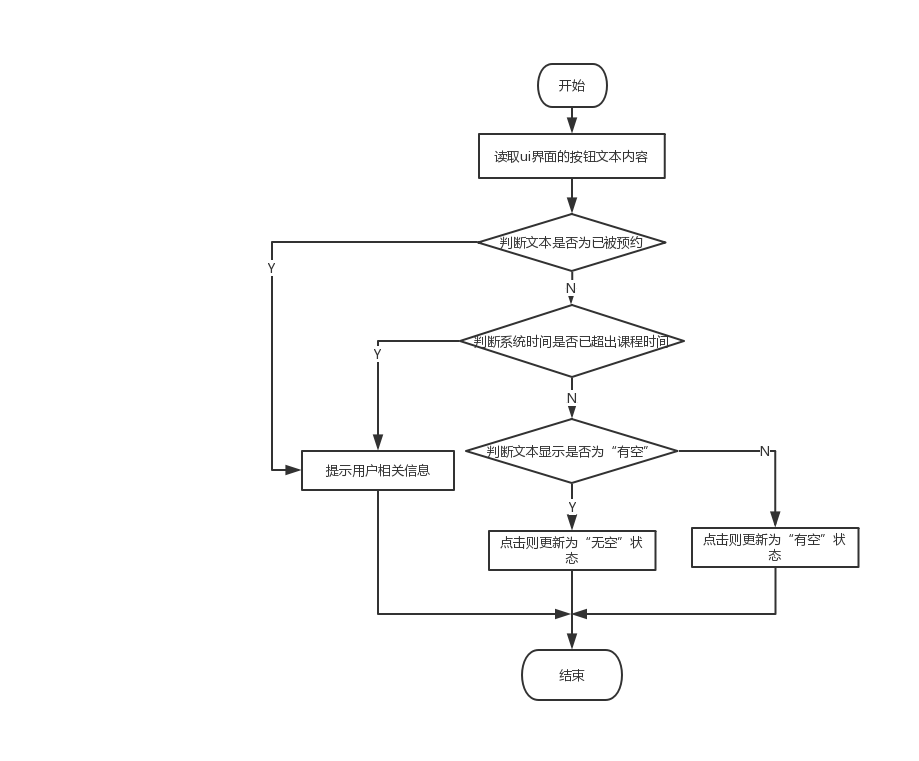


图4-4修改空闲时间函数

(8)学员的点击预约函数 (有多个相同的点击槽函数，故只说明一处)

函数名称：void order1::**on\_pushButton\_1\_1\_clicked**()

函数输入：按钮点击信号

函数输出：无

函数功能：读取对应的按钮的文本信息，判断显示是否为“预约”字样，若是则调用数据库select函数搜索该用户当天预约信息，查看该用户是否已预约其他教练或其他时间段，已经预约则提示用户已预约；没有预约则判断系统时间是否已经超过课程时间，如已超过，提示用户；若没有，则使用数据库update函数更新数据库内容，并且更新按钮文本信息。若显示为“预约成功”字样，点击则提示用户是否确定取消预约，根据操作更新内容。若为其他字样如“已被预约”以及“没空”，则点击无任何变化。函数流程如图4-5所示。

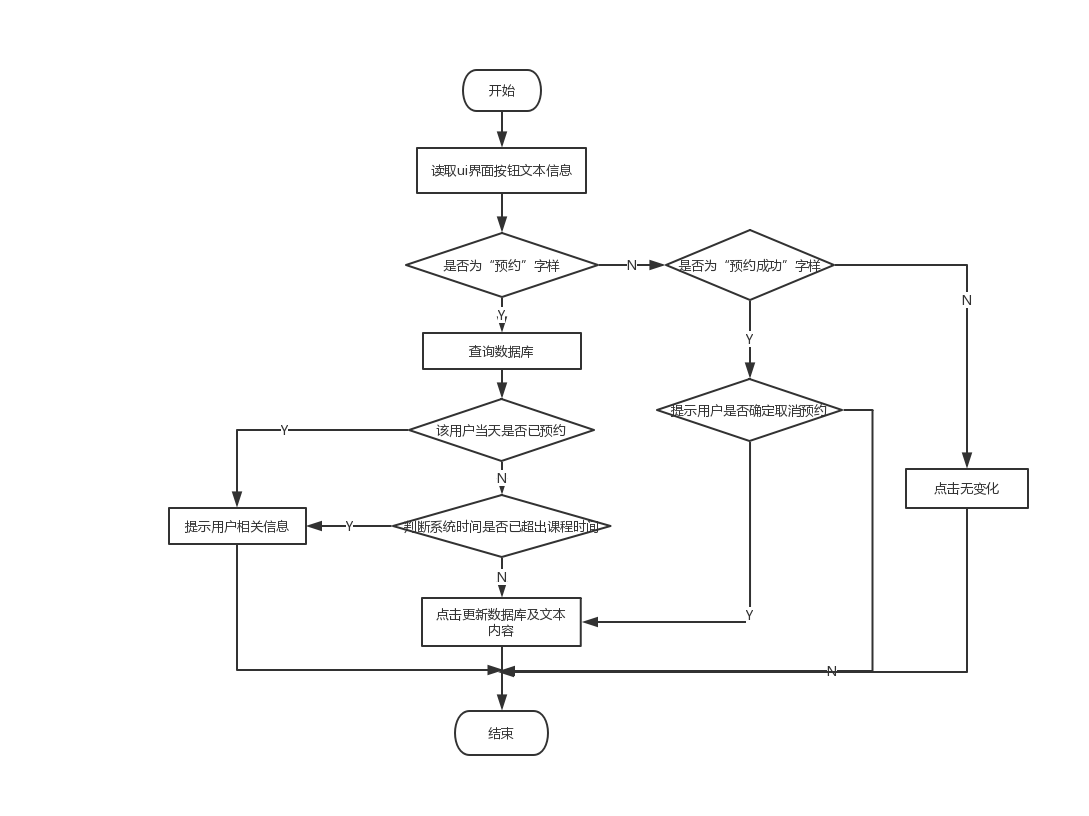


图4-5 学员点击预约函数

各函数的调用关系如图4-6，图4-7，图4-8，图4-9所示。

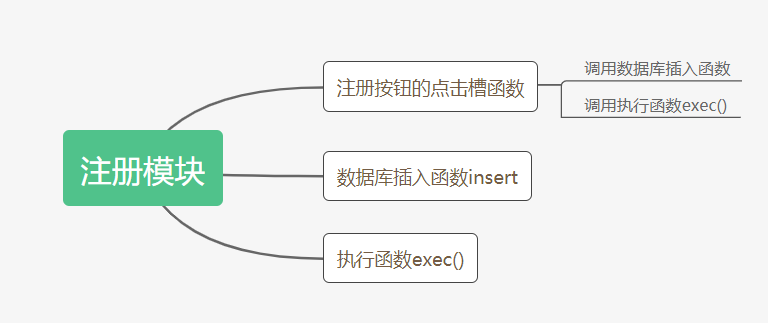


图4-6 注册模块函数关系图

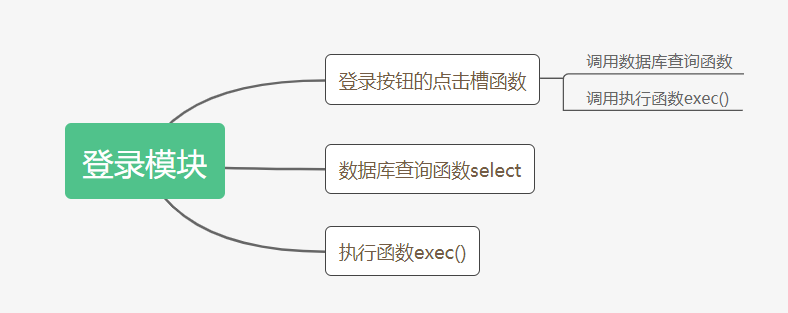


图4-7 登录模块函数关系图

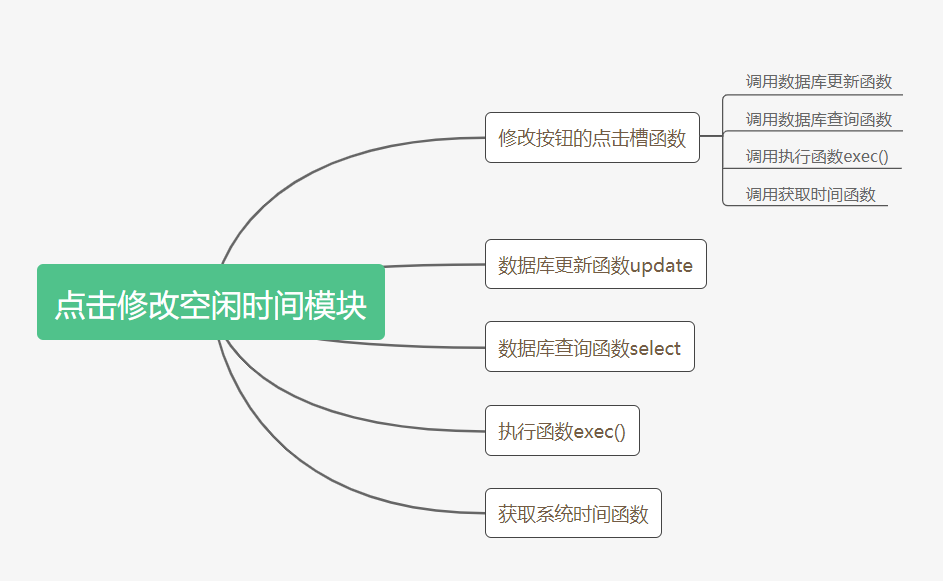


图4-8 修改空闲时间模块函数关系图

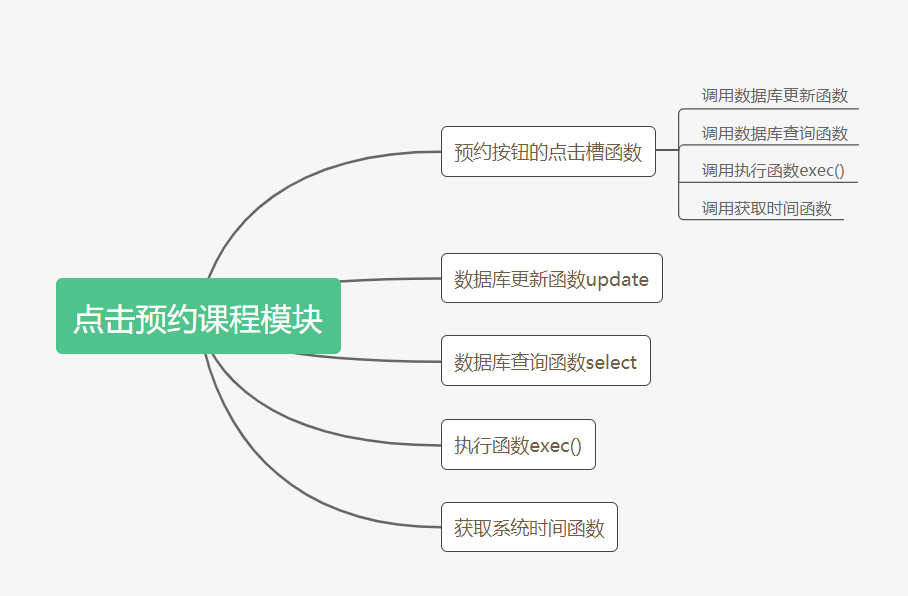


图4-9 预约课程模块函数关系图

## 4.3 测试计划和测试用例

### 4.3.1 测试方法

软件测试是为了发现程序中的错误而执行的过程，而目前测试系统中常用的方法有黑盒测试以及白盒测试两大模块。而在黑盒测试中又有等价类划分、边界值分析、以及错误推断等方法；在表盒测试中有逻辑覆盖测试、基本路径测试、条件测试和循环测试等方法。本次软件的开发中，考虑到我们的开发小组人数较少，各个成员交叉对其他人员的模块进行测试，即对测试者而言，被测试模块是未知的，此处我们选取注册模块进行黑盒测试。而将各个模块合并集成软件后，各个模块应经历了测试，故测试应是集成测试，而且软件对开发者而言应是已知的，则此处我们将对已知的学生预约功能以及用户的信息录入模块功能进行白盒测试。

### 4.3.2 模块一 用户注册模块-黑盒测试

在此模块中将要完成对教练和学员的注册功能测试。我们选取了黑盒测试中的等价类划分对此模块进行测试。

（1）输入规则

用户名为手机号，以11位的数字字符的形式输入

密码以数字、符号、英文或三者混合字符串，输入字符数大于3位

（2）等价类划分

对于用户名划分

有效等价类：

I. 11位的纯数字字符串符合手机号码格式

II． 11位的数字字符串但为无效的手机号码格式（此处通过后期与移动公司合作方可测试，此处不在测试说明）

无效等价类：

III．小于11位的数字字符串

IV． 大于11位的数字字符串

V． 出现非数字字符串

对于密码的划分

有效等价类：

I. 大于3位的纯数字字符串

II． 大于3位的纯英文字符串

III．大于3位的纯符号字符串

IV． 大于3位的混合型字符串

无效等价类：

V． 小于三位的字符串

VI．出现汉字字符串

（3）测试方案及结果

对于用户名的测试输入，结果如表4.3.2.1所示。

表4-1 用户名测试表

|  |  |  |  |
| --- | --- | --- | --- |
| 编号 | 等价类 | 输入 | 输出结果 |
| 1 | 11位的纯数字字符串符合手机号码格式 | 15102778649 |  |
| 2 | 11位的数字字符串但为无效的手机号码格式 | 12345678910 | 此处目前无法准确测试，后期预计输出  手机号不正确 |
| 3 | 小于11位的数字字符串 | 1510277864 |  |
| 4 | 大于11位的数字字符串 | 151027786499 |  |
| 5 | 出现非数字字符 | U5102778649 |  |

对于密码的测试输入如表4-2所示。

表4-2 密码测试

|  |  |  |  |
| --- | --- | --- | --- |
| 编号 | 等价类 | 输入 | 输出结果 |
| 1 | 大于3位的纯数字字符串 | 123456 |  |
| 2 | 大于3位的纯英文字符串 | abcdef |  |
| 3 | 大于3位的纯符号字符串 | \*-\*-\*- |  |
| 4 | 大于3位的混合型字符串 | ab12\*- |  |
| 5 | 小于三位的字符串 | 12 |  |
| 6 | 出现汉字字符串 | 一二三四 |  |

至此模块一测试完成，且测试功能无误。

### 4.3.3 模块二 用户信息录入模块-白盒测试

在此模块中将要完成对用户的信息录取登录及显示着一系列的功能进行测试。由于此时开发人员已完整的实现了模块的连接，知道程序如何运行，故我们选取了白盒测试中的基本路径覆盖测试对此模块进行测试。

（1）测试程序流程图

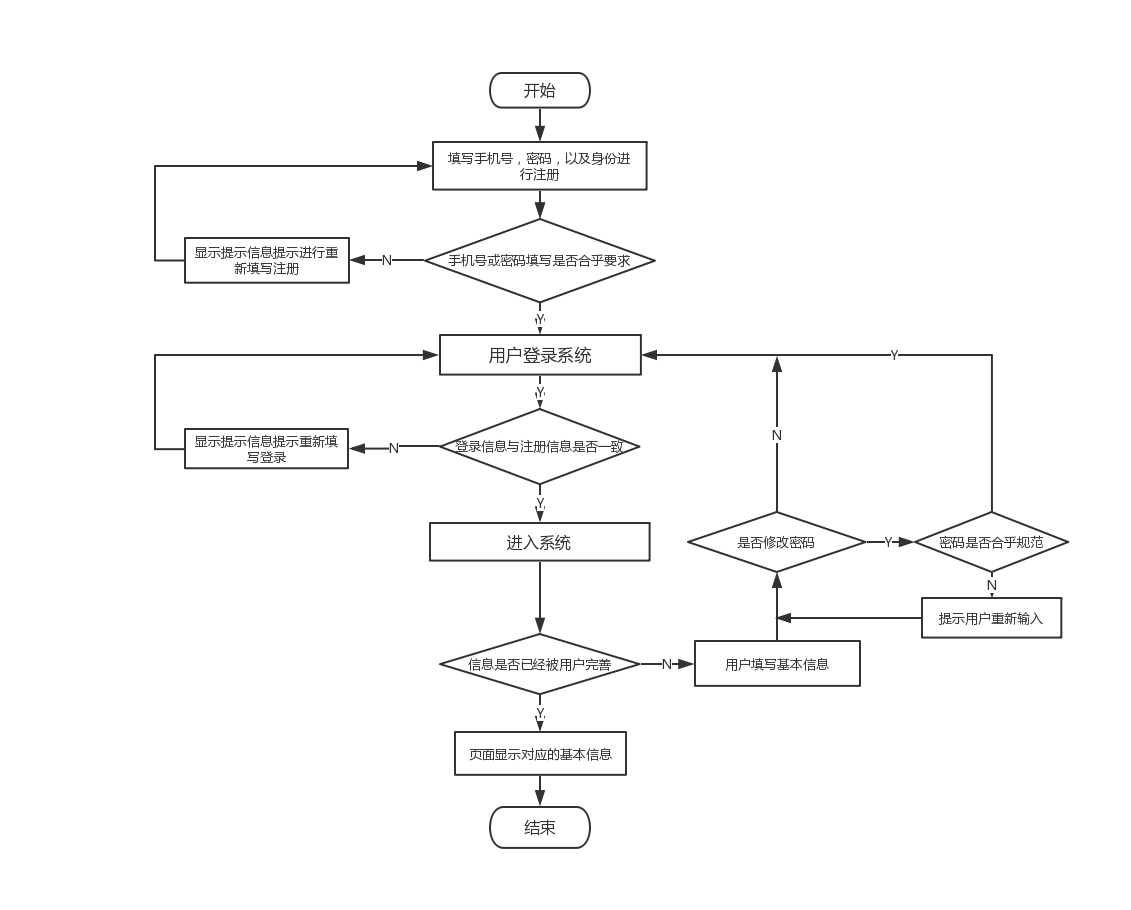


图4-10 模块二测试程序

（2）测试程序控制流图

程序控制流图如图4-11所示。

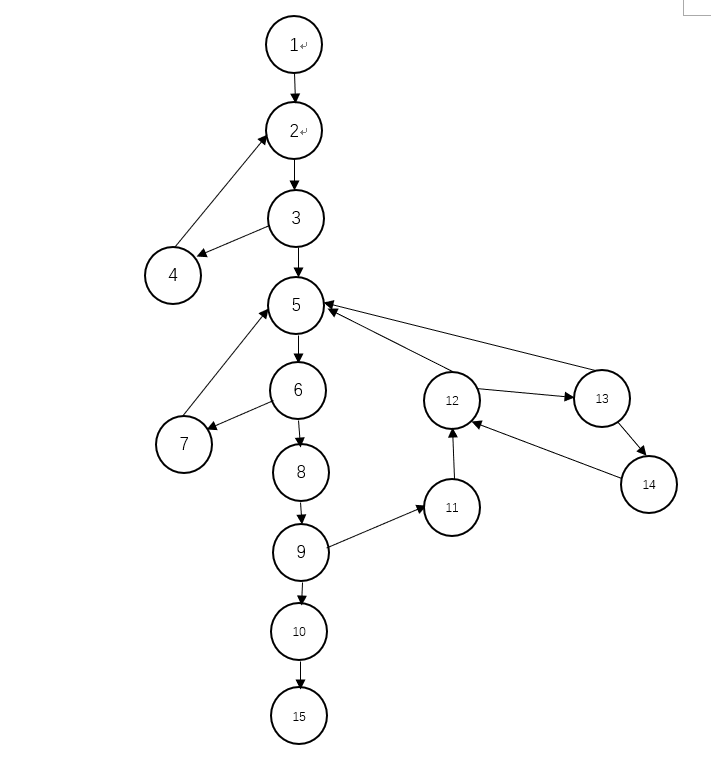


图4-11 程序控制流图

（3）测试路径

由图4.3.2-2知共有5个判定节点即有6条关键路径。

测试路径如下所示：

路径一：1-2-3-5-6-8-9-10-15（此路径需签在路径2,3,4,5,6,7中测试）

路径二：1-2-3-4-2-3-5…………

路径三：1-2-3-5-6-7-5-6-8…………

路径四：1-2-3-5-6-8-9-11-12-5…………

路径五：1-2-3-5-6-8-9-11-12-13-5…………

路径六：1-2-3-5-6-8-9-11-12-13-14-12………

（4）测试计划及输入

具体测试过程及测试结果如表4-3所示。

表4-3 测试用例及预期结果

|  |  |  |  |
| --- | --- | --- | --- |
| 路径 | 测试用例输入 | 期望结果 | 实际输出结果 |
| 2 | 注册时输入不和规范的手机号和登录密码 | 提示用户重新输入 | 弹出对话框提示用户格式  正确输入后提示注册成功 |
| 3 | 登录时输入不匹配信息 | 提示用户信息有误，重新输入 | 弹出对话框提示信息不匹配  正确输入后提示登录成功 |
| 4 | 完善或修改个人信息  但未修改登录登录密码 | 修改信息完成，下次登录后，显示信息正确 | 修改后，第二次登录显示的信息与第一次修改的一致 |
| 5 | 完善信息并修改登录密码，且密码与确认密码两栏输入一致且合乎规范 | 基本信息完善成功，且密码修改成功，重新登录时输入旧密码不能登录，输入新密码可进入系统 | 与预计结果一致 |
| 6 | 修改登录密码，但输入不密码不合乎规范 | 提示用户重新输入修改密码 | 弹出对话框并输出相应的提示信息 |

至此模块二测试完成，且测试功能无误。

### 4.3.4 模块三 学员的预约课程模块-白盒测试

在此模块中将要完成学员的预约两天课程的功能，以及取消预约的功能。我们将采用白盒测试，按照预约课程的逻辑流程进行测试。检测整个的预约模块是否像预测一样进行。

（1）测试程序流程图

预约功能有当天预约和第二天预约之分，两处模块相似，差别在于第二天预约的时候不用判断是否超过课程时间。故此处只给出了当天的预约模块的测试计划，第二天预约计划与此相似，此处不在说明。

预约功能测试计划如图4-12所示。

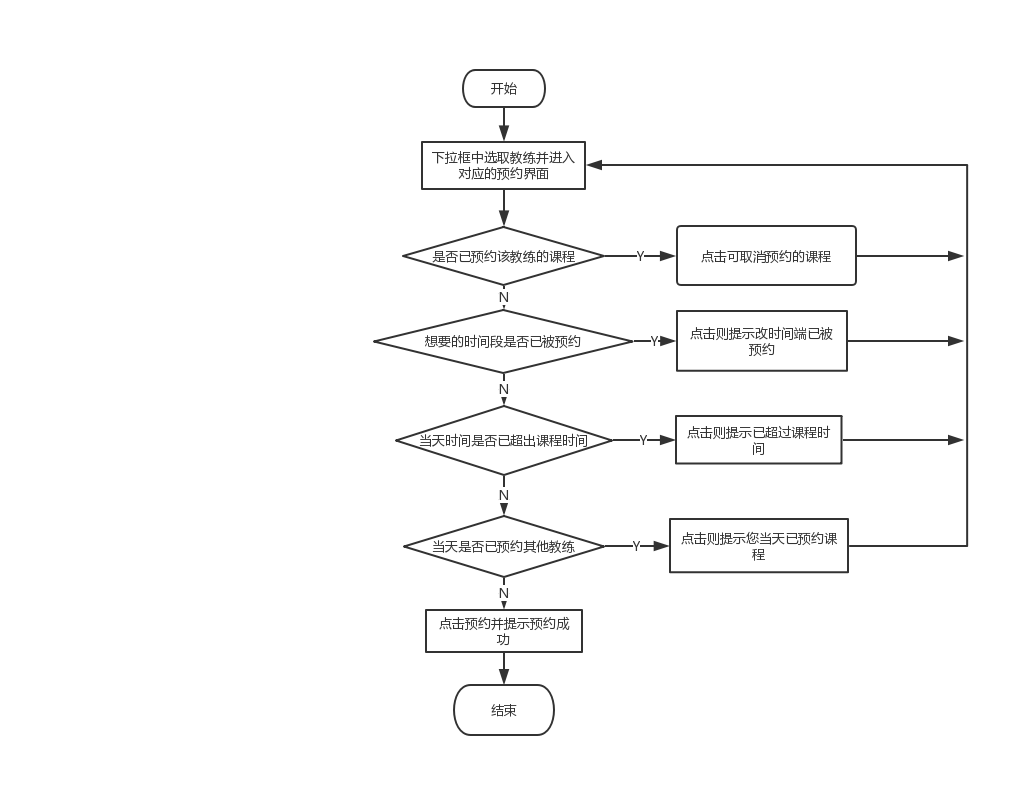


图4-12 预约模块测程序流程图

（2）测试程序控制流图

程序控制流图如图4-13所示。

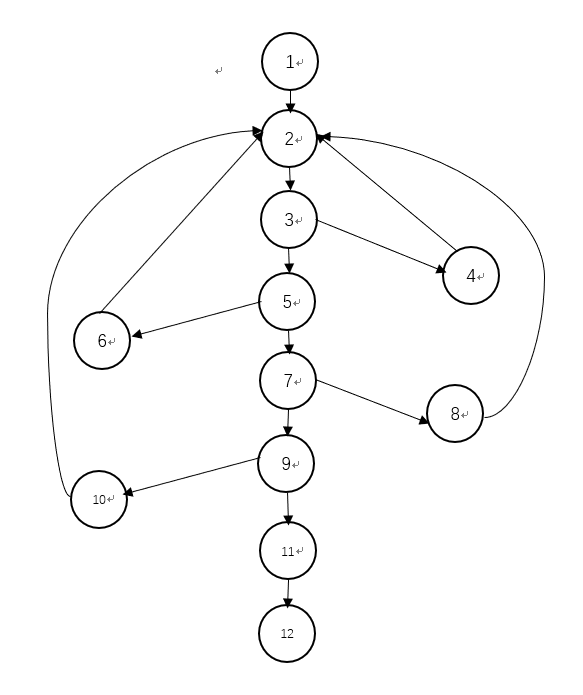


图4-13 预约模块程序控制流图

（3）测试路径

由图4.3.4.2所知共有4个判断节点，故共有5条独立路径需要测试

测试路径如下：

路径一：1-2-3-5-7-9-11-12（此路径需签在路径2,3,4,5,中测试）

路径二：1-2-3-4-2-3-5…………

路径三：1-2-3-5-6-2-3-5…………

路径四：1-2-3-5-7-8-2-3-5…………

路径五：1-2-3-5-7-9-10-2-3-5…………

（4）测试计划及输入

具体测试过程及测试结果如表4-4所示。

表4-4测试用例及预期结果

|  |  |  |  |
| --- | --- | --- | --- |
| 路径 | 测试用例输入 | 期望结果 | 实际输出结果 |
| 2 | 用户已预约该教练的该时间段课程，点击按钮 | 提示用户是否确认取消并根据用户选择作出对应的处理 | 弹出对话框，确认取消则原按钮的预约成功变为预约字样，不取消则不作处理 |
| 3 | 用户想要预约的时间段显示已被预约，点击按钮 | 提示用户该时间段已被预约 | 正确弹出消息框提示用户 |
| 4 | 当前时间已超过课程时间，点击按钮 | 提示用户已超过时间无法预约 | 正确弹出消息框提示用户 |
| 5 | 当天已预约其他教练，点击按钮 | 提示用户当天已预约课程 | 正确弹出消息框提示用户 |

至此模块三测试完成，且测试功能无误。

## 4.4 结果分析

除了上文所阐述的三个基本模块的测试，我们还做了大量的测试，对我们的软件做了全面的测试与完善。本次软件开发目的是为了提高学员和驾校双方的时间利用率、方便驾校对学员的课程安排通过以及数据分析、以及学员对教练对的了解与选择。通过测试，我们可以得知，首先是系统的注册功能较为全面，可以用户注册的手机号的格式以及密码进行准确的判断，并且作出相应的提示，以确保系统对各个用户信息的准确录入。而通过对学员的预约模块的测试，我们也确保了学员每天可以预约多天但每天只能预约一次的功能，以及对各个教练的课程预约情况显示的正确性，和对取消预约模块中提示信息的完善。同时，通过后期大量的测试，我们也确保了教练模块的功能的准确性，满足了用户的需求和本次软件开发的目的。

# 5 总结

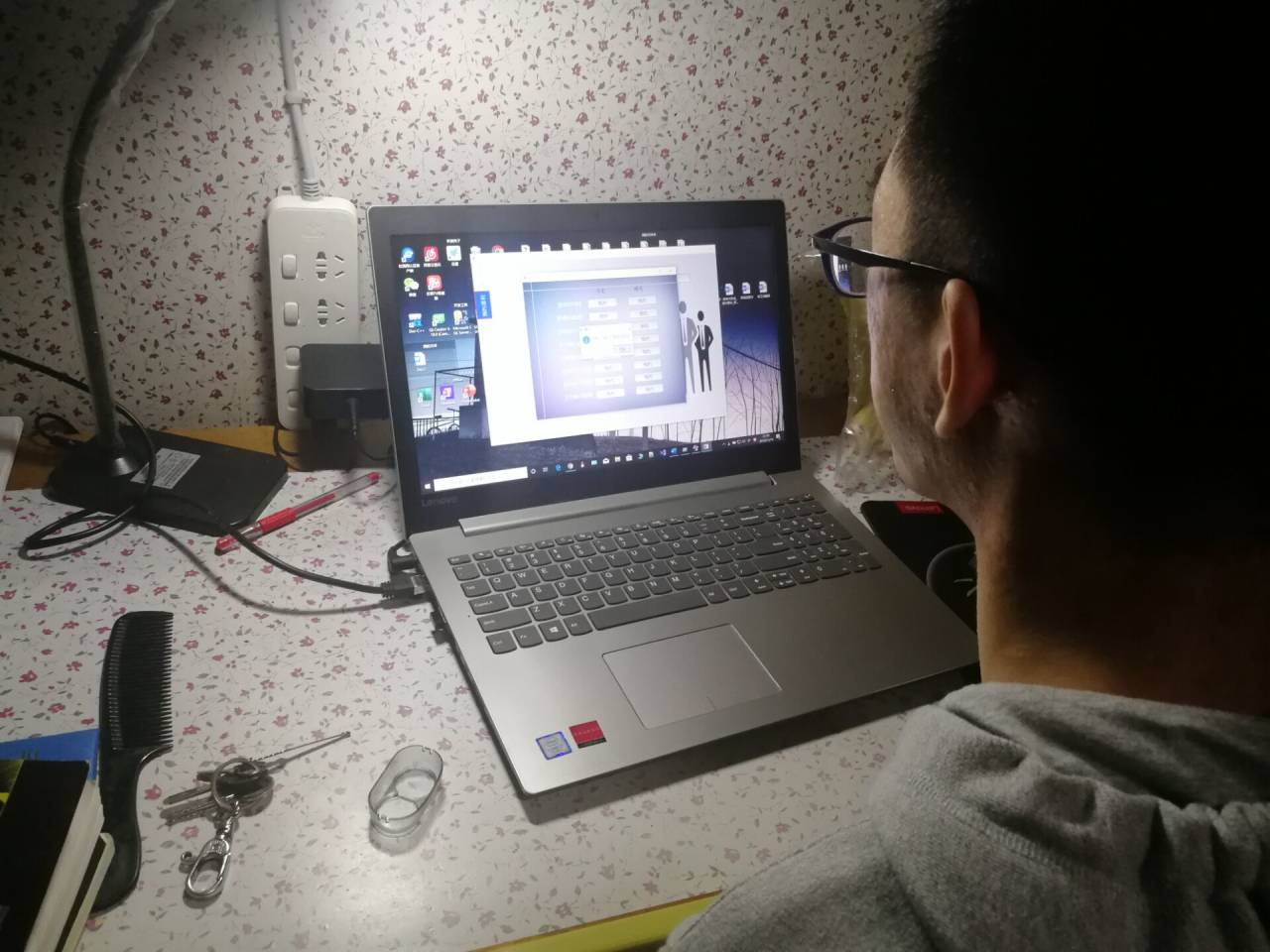
## 5.1 用户反馈

在我们的项目开发完成之后，我们将产品打包发到网上，让人们试用体验。下面是其中几个用户的反馈：

表5-1 部分用户试用反馈统计

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 用户编号 | 用户年龄 | 用户类型 | 用户评价分（十分制） | 优点 | 缺点 |
| A | 18 | 学生 | 8 | 方便预约 | 操作天数太少，只有两天 |
| B | 21 | 学生 | 7 | 界面简洁舒适 | 操作天数太少，只有两天 |
| C | 19 | 学生 | 8 | 方便节约时间 | 个人电脑操作不熟，试用不便 |
| D | 34 | 在职 | 9 | 第一次使用，惊叹 | 不怎么会操作电脑 |
| E | 45 | 在职 | 9 | 上班族很节约时间 | 功能太多，有的不需要 |
| F | 38 | 在职 | 9 | 产品很新颖，很实用 | 无 |

下面是我们的用户在使用我们的平台时的照片：



## 5.1 全文总结

以上就是我们项目的全部设计、实现和开发过程，根据分工，现在我对自己在项目中所做的工作进行总结，主要为下面六点：

（1）参与选题讨论： 我们一开始想法比较局限，想的项目都比较大，导致觉得并不好入手，在经过身边需求考察之后才确定驾校预约系统的题目。这个虽然比较朴素，没有很多花枝招展的东西，但是却是生活中比较实用的平台。在做的过程中，我们采集了各方的意见，发现，大家对我们的选题都比较感兴趣，这是我们能一路走下去的很重要的一个动力。最后经过多次讨论和多方调研，确定为驾校预约系统；

（2）参加项目规划： 除了讨论项目的模型之外，我负责项目规划中的开发软件的问题定义和需求分析，例NABCD模型。同时使用网络工程图，网络工程时间表、逻辑模型进行新旧对比、ER实体图、数据流图、用例图来形象详细说明各个部分。

（3）市场的调研：调研是在工程中必不可少的一步，虽然在很多的开发中没有做书面强调，但是却往往最重要。如果不知道当前市场用户对于你开发平台的需求，没有抓住用户的痛点，很难开发出一款好的产品；同样的，如果不了解当前市场上已经有的成品，也不能开发出胜于之上的产品，便无法与别人进行竞争，所以最好的做法就是取其精华，去其糟粕，找到共性，提出亮点，才能进行更高层次的设计开发。为此，我们在同学的小圈子里、学校的大圈子以及网络平台进行了一些问卷调查，并进行了数据总结，找到需求重点和目前缺失，然后回头再对我们的模型进行改进设计。

（4）项目的开发：我负责的部分是学员的功能设计和实现。我使用的开发工具是QT，开发环境是win10，语言选择C++，项目管理使用码云。我的工作具体说来就是在QT环境下，学员部分的界面功能的设计和开发。因为事先的接口已经和小组成员商量好，所以开发的过程中逻辑并不是问题。功能模块我主要是设计成了三个板块：个人信息板块、预约情况查询、进行预约操作。当然这是其中的大的部分，之间的小细节设计还很多，包括消息提醒、时钟记录等。

（5）界面的美化设计：在基本框架开发结束后，我们只是得到了一个简单的demo，在这个基础之上我们做界面美化，我负责收集素材。我设计的界面风格主要是以暗色调，简约型。

（6）推广和收集用户反馈：在我们的平台开发后，我们对产品进行了封装，先在同学中进行试用，让他们体验demo，从用户方面收集一些反馈信息。得到的反馈信息对于我们产品的优化改进是有很大意义的，我们能够从用户的意见反馈中得到一些珍贵的新颖的点子。

# 6 体会

进入大学以来，除了游戏开发导论，这是我们第一次开关于软件的专业课，第一次能够系统地学习软件设计和开发的理论知识并且运用到实践中，最后能在项目计划之内完成所有的工作，总的来说，是一次很棒的体验！

因为之前并没有多少开发经验，所以在整个项目的开发过程中避免不了遇到各种各样的问题，下面我对我遇到的一些挫折和解决过程进行总结，希望以后对这些方面能有所提高：

1.项目规划：项目的规划远比项目的实施要难。在我看来，项目的规划就像是出门前选路，你需要根据需求、时间、金钱选取最优的方案求解，而项目的实施只不过是按部就班地做罢了，因而在项目规划上也是我们做得最多得一件事，我也在上面花费了不少的心思。从理论课程的学习到具体项目的实践设计，需要经历一个很长的过渡阶段。虽然经过系统的软件开发设计学习，在实践中有了理论上的基础，但是还是得一步步分析从新旧模型对比、网络工程图、ER图、数据流图、用例图等一步步在我们的项目中分析画出来。最后写出项目企划和产品说明书；

2.开发语言的系统学习：我的开发环境的是win10的QT，使用C++开发。虽然有之前的C语言基础，对C++的应用并不困难，但是对于C++中一些重要的概念和应用并不熟，例如类、封装。说到这里，当然最重要的就是内存管理，总所周知，在C++中，内存管理是最让程序员头痛和饱受折磨的一件事。所以我认为，在项目开发之前，对C++进行一个系统的学习是很有必要的。于是，我在项目开始的前一段时间，跟进网上的C++课程学习，进行了C++的速成。同时，我很开心，在系统学习了之后就行有这样一个实践机会进行巩固，所以实际上在我看来，这次的项目开发，还促进了编程能力的提高；

3.开发环境的学习：虽然在之前的汇编课程中接触过C的windows编程，但是其涉足不深，而且与QT中的特定的开发环境还是有很大的区别的。所以我又在网上学习了QT开发速成，学习了一些基本操作。

4.开发环境的配置：我相信，不管是多么有经验的程序员，他曾经一定被开发环境痛苦折磨过。的确如此，不管是在配置QT还是在试图连接数据库的过程中我都遇到了开发环境的配置问题。而主要的问题总结起来还是这样几点：路径添加错误、版本和机器不匹配、SQL和QT的版本不适配、下载包缺失需要手动配置环境等。尤其是SQL和QT的连接，花了很久功夫在网上找解决办法才得到解决。所以在这个过程中，帮助最大的当然就是网上的各种教程和经验贴，当然缺少不了组员的相互帮助；

5.开发时bug调试：实际上，这是每个项目开放过程中都会遇到的问题，但是，我想要特别说明的是，QT是一个非常强大的开发平台，对于其中的每一个函数、类、槽、信号都有说明文档并且支持搜索查找到对应原型设计，因此在遇到报错的时候，只需要找到报错的问题，查手册就可以看到它的函数原型，这为程序调试节省了很多时间；

6.调研问题：在调研过程中会遇到一些不耐烦填问卷或者乱填问卷的人，所以我们使用了一定的小的奖励机制调动大家的积极性。

以上就是我在开发过程中遇到的问题，而正是有了这些问题，才让我在这次项目开发过程中同时也学到不少的东西，下面我对自己学到的东西做一个总结：

1.熟悉了软件设计的过程；

2.学习用各种图来形象表达工程中的动态；

3.学会了小组合作、人员调度和工作分配；

4.C++和QT的学习；

5.编程能力的提高；

6. 提高了编程中解决问题的能力；

7. 学会了写产品说明书。

最后感谢小组成员的相互帮助，感谢老师的耐心指导！

# 附录