

分类号_____

编 号_____

U D C _____

密 级_____

中国民航飞行学院

毕业设计（论文）

题 目_____顺风车系统_____

作者姓名_____张健_____

指导教师姓名及职称_____张黔川 副教授_____

二级学院及专业名称_____计算机学院 信息与计算科学_____

提交日__2017 年 5 月 10 日__ 答辩日期__2017 年 6 月 10 日__

答辩委员会主任__傅强__ 评 阅 人__李桂花__

2017 年 6 月 5 日

顺风车系统

学生：张健 指导老师：张黔川

摘 要

顺风车系统是一个服务于普通民众的 C2C 便民系统。

顺风车系统用于搭建一个便民的顺风车服务，人们在开车去一个地方的同时也能顺路搭载其他乘客，并凭借个人意愿选择收取少量费用或者不收取费用。同时这种行为模式也对缓解交通压力，加强沟通信任，促进节能减排等都有所帮助。

本文首先进行需求分析，分析得出需要建立的各个模块。用 power designer 设计并创建后台数据库。采用 SSM 框架结构，并用 eclipse 进行详细设计（MVC），包括网站界面，前台数据处理以及将数据处理结果反馈至前台页面。本系统主要分为三个模块，账户模块（用户可以登录和注册账号，修改个人信息等。同一个账号可以同时属于车主以及乘客两种角色）、发布行车信息模块（车主发布一条自己即将开始的行程信息，包括开始时间，起点，途径哪些地理位置，可搭载人数，车牌号，收费情况，联系电话等内容）、搭顺风车模块（乘客在车辆行程页面可以查看到当前地理位置附近（或者选择指定地理位置）的行程信息，并选择自己愿意搭载的车辆，然后即可联系车主）以及评价模块（用户和车主分别对彼此做出评价）。

关键词：顺风车，数据库，SSM

Hitching a ride of system

Student: Zhangjian Supervisor: Zhangqianchuan

Abstract:

Hitching a ride of system is a service for ordinary people C2C convenient system.

Hitching a ride of system is used to build a convenient lift service, people driving to a place can drop with other passengers at the same time , and the drivers can choose to charge a small fee or pay for free. It also helps to reduce the traffic problems, strengthen the communication and trust , and promote energy conservation and emission.

This article first analyzes the requirements and analyzes the modules that need to be built. We have designed and created a background database with power designer. Adopt the SSM framework and use eclipse for detailed design (MVC), including the site interface, the foreground data processing, and the feedback of the data processing results to the front page. This system is mainly divided into three modules, account module (the user can login and registration , modify personal information, etc. The same account can belong to drivers and passengers at the same time two roles), release the information of driving module (owner issued an upcoming travel information, including start time, the starting point, way which the geographical position, can accommodate the number of the license plate number, charge, contact phone number, etc), hitch a ride module (passengers in vehicles travel page can see the current location near (or designated location) of travel information, and select the vehicle they want to pick up, then you can contact the owner) and evaluation module (users and owners make evaluation to each other) respectively.

Keywords: Hitching a ride, database, SSM

目录

目录	- 2 -
第一章 引言	- 3 -
1.1 课题的研究意义	- 3 -
1.2 课题提出的背景	- 3 -
第二章 研究现状及设计目标	- 4 -
2.1 系统设计相关原理	- 4 -
2.1.1 Java 语言介绍	- 4 -
2.1.2 SSM 框架介绍	- 9 -
2.2 Servlet 技术	- 10 -
2.2.1 servlet 简介	- 10 -
2.2.2 servlet 特征	- 11 -
2.3 课题要达到的设计目标	- 12 -
第三章 系统的需求分析	- 13 -
3.1 可行性分析	- 13 -
3.2 开发环境	- 14 -
3.3 研究设计中要解决的问题	- 14 -
3.4 系统结构描述	- 15 -
第四章 系统总体设计	- 17 -
4.1 顺风车系统设计实现的策略	- 17 -
4.2 系统运行效果图	- 18 -
第五章 系统详细设计	- 21 -
5.1 数据库设计	- 21 -
5.1.1 基本的 ADO 编程模型	- 21 -
5.1.2 数据库选择	- 22 -
5.2 数据库结构图	- 22 -
5.2.1 数据库 CDM (概念数据模型) 图	- 22 -
5.2.2 数据库结构表	- 23 -
第六章 程序模块实现	- 25 -
6.1 分模块详述系统各部分的实现方法	- 25 -
6.1.1 寻找顺风车模块的实现方法	- 25 -
6.1.2 发布顺风车行程模块的实现方法	- 26 -
6.1.3 顺风行程模块的实现方法	- 27 -
6.2 关于课题的下一步工作	- 28 -
第七章 系统测试	- 30 -
7.1 测试目的	- 30 -
7.2 测试结果	- 30 -
结束语	- 32 -
致谢	- 33 -
参考文献	- 34 -

第一章 引言

1.1 课题的研究意义

顺风车系统是一个服务于普通民众的 C2C 便民系统。

现在人们的出行方式可谓多种多样，以公交和地铁的公共交通构成了经济出行的主要部分，但有些时候在公交和地铁未能方便到达的区域，主要还是依赖出租车等形式，但因为其较高的单价而受到不少消费者的抵触，所以顺风车平台应运而生，能够很好的解决这个问题，它能够在提供丰富出行路线选择的同时，将费用降至最低，成为一个新的经济实惠的出行方式。

顺风车系统用于搭建一个便民的顺风车服务，人们在开车去一个地方的同时也能顺路搭载其他乘客，并凭借个人意愿选择收取少量费用或者不收取费用。这对于车主来说这本身是一件助人为乐的好事情，对于搭车的人来说也是一个非常经济划算的选择。同时这种行为模式也对缓解交通压力，加强沟通信任，促进节能减排等都有所帮助。

1.2 课题提出的背景

1998 年，王永为数万路人提供顺风车，为顺风车的发展开创先河。2012 年 1 月 10 日，王永联合邓飞、赵普、郎永淳、陈伟鸿五位公益人士在新浪微博发起“春节回家顺风车”活动，倡议大家开顺风搭载同路的老乡一起回家，缓解春运困难。顺风车由此正式发展起来，并为以后的发展打下基础。2016 年 7 月 28 日国务院办公厅发布了《关于深化改革推进出租汽车行业健康发展的指导意见》和《网络预约出租车经营服务管理暂行办法》网络约车管理办法，明确了网络约车合法地位。

大气污染越发严重，其中一个重要的原因就是车辆的尾气排放。去年年底全国各地不同程度的连续雾霾天气更是将其推向高峰。而日益增加的私家车，大部分都是空载，造成交通堵塞的同时也造成了资源的极大浪费。此项目的着眼于环节日益恶化的环境问题，缓解交通压力，充分利用车辆资源。

第二章 研究现状及设计目标

2.1 系统设计相关原理

2.1.1 Java 语言介绍

Java 是一门面向对象编程语言，不仅吸收了 C++ 语言的各种优点，还摒弃了 C++ 里难以理解的多继承、指针等概念，因此 Java 语言具有功能强大和简单易用两个特征。Java 语言作为静态面向对象编程语言的代表，极好地实现了面向对象理论，允许程序员以优雅的思维方式进行复杂的编程^[1]。Java 具有简单性、面向对象、分布式、健壮性、安全性、平台独立与可移植性、多线程、动态性等特点^[2]。Java 可以编写桌面应用程序、Web 应用程序、分布式系统和嵌入式系统应用程序等^[3]。

1) 编程环境:

JDK (Java Development Kit) 称为 Java 开发包或 Java 开发工具，是一个编写 Java 的 Applet 小程序和应用程序的程序开发环境。JDK 是整个 Java 的核心，包括了 Java 运行环境 (Java Runtime Environment)，一些 Java 工具和 Java 的核心类库 (Java API)。不论什么 Java 应用服务器实质都是内置了某个版本的 JDK。主流的 JDK 是 Sun 公司发布的 JDK，除了 Sun 之外，还有很多公司和组织都开发了自己的 JDK，例如，IBM 公司开发的 JDK，BEA 公司的 Jrocket，还有 GNU 组织开发的 JDK^[4]。另外，可以把 Java API 类库中的 Java SE API 子集和 Java 虚拟机这两部分统称为 JRE (Java Runtime Environment)，JRE 是支持 Java 程序运行的标准环境^[5]。JRE 是个运行环境，JDK 是个开发环境。因此写 Java 程序的时候需要 JDK，而运行 Java 程序的时候就需要 JRE。而 JDK 里面已经包含了 JRE，因此只要安装了 JDK，就可以编辑 Java 程序，也可以正常运行 Java 程序。但由于 JDK 包含了许多与运行无关的内容，占用的空间较大，因此运行普通的 Java 程序无须安装 JDK，而只需要安装 JRE 即可^[6]。

2) 语言特点:

a) 简单性

Java 看起来设计得很像 C++，但是为了使语言小和容易熟悉，设计者们把 C++ 语言中许多可用的特征去掉了，这些特征是一般程序员很少使用的。例如，Java 不支持 go to 语句，代之以提供 break 和 continue 语句以及异常处理。Java 还剔除了 C++ 的操作符重载 (overload) 和多继承特征，并且不使用主文件，免去了预处理程序。因为 Java 没有结构，数组和串都是对象，所以不需要指针。Java 能够自动处理对象的引用和间接引用，实现自动的无用单元收集，使用户不必为存储管理问题烦恼，能更多的时间和精力花在研发上。

b) 面向对象

Java 是一个面向对象的语言。对程序员来说，这意味着要注意应中的数据 and 操纵数据的方法 (method)，而不是严格地用过程来思考。在一个面向对象的系统中，类 (class) 是数据和操作数据的方法的集合。数据和方法一起描述对象 (object) 的状态和行为。每一对象是其状态和行为的封装。类是按一定体系和层次安排的，使得子类可以从超类继承行为。在这个类层次体系中有一个根类，它是具有一般行为的类。Java 程序是用类来组织的。Java 还包括一个类的扩展集合，分别组成各种程序包 (Package)，用户可以在自己的程序中使用。例如，Java 提供产生图形用户接口部件的类 (java.awt 包)，这里 awt 是抽象窗口工具集 (abstract windowing toolkit) 的缩写，处理输入输出的类 (java.io 包) 和支持网络功能的类 (java.net 包)。

c) 分布性

Java 设计成支持在网络上应用，它是分布式语言。Java 既支持各种层次的网络连接，又以 Socket 类支持可靠的流 (stream) 网络连接，所以用户可以产生分布式的客户机和服务器。网络变成软件应用的分布运载工具。Java 程序只要编写一次，就可到处运行。

d) 编译和解释性

Java 编译程序生成字节码 (byte-code)，而不是通常的机器码。Java 字节码提供对体系结构中性的目标文件格式，代码设计成可有效地传送程序到多个平台。Java 程序可以在任何实现了 Java 解释程序和运行系统 (run-time system) 的系统上运行。在一个

解释性的环境中，程序开发的标准“链接”阶段大大消失了。如果说 Java 还有一个链接阶段，它只是把新类装进环境的过程，它是增量式的、轻量级的过程。因此，Java 支持快速原型和容易试验，它将导致快速程序开发。这是一个与传统的、耗时的“编译、链接和测试”形成鲜明对比的精巧的开发过程。

e) 稳健性

Java 原来是用作编写消费类家用电子产品软件的语言，所以它是被设计成写高可靠和稳健软件的。Java 消除了某些编程错误，使得用它写可靠软件相当容易。Java 是一个强类型语言，它允许扩展编译时检查潜在类型不匹配问题的功能。Java 要求显式的方法声明，它不支持 C 风格的隐式声明。这些严格的要求保证编译程序能捕捉调用错误，这就导致更可靠的程序。可靠性方面最重要的增强之一是 Java 的存储模型。Java 不支持指针，它消除重写存储和讹误数据的可能性。类似地，Java 自动的“无用单元收集”预防存储漏泄和其它有关动态存储分配和解除分配的有害错误。Java 解释程序也执行许多运行时的检查，诸如验证所有数组和串访问是否在界限之内。

异常处理是 Java 中使得程序更稳健的另一个特征。异常是某种类似于错误的异常条件出现的信号。使用 try/catch/finally 语句，程序员可以找到出错的处理代码，这就简化了出错处理和恢复的任务。

f) 安全性

Java 的存储分配模型是它防御恶意代码的主要方法之一。Java 没有指针，所以程序员不能得到隐蔽起来的内幕和伪造指针去指向存储器。更重要的是，Java 编译程序不处理存储安排决策，所以程序员不能通过查看声明去猜测类的实际存储安排。编译的 Java 代码中的存储引用在运行时由 Java 解释程序决定实际存储地址。Java 运行系统使用字节码验证过程来保证装载到网络上的代码不违背任何 Java 语言限制。这个安全机制部分包括类如何从网上装载。例如，装载的类是放在分开的名字空间而不是局部类，预防恶意的小应用程序用它自己的版本来代替标准 Java 类。

g) 可移植性

Java 使得语言声明不依赖于实现的方面。例如，Java 显式说明每个基本数据类型的

大小和它的运算行为（这些数据类型由 Java 语法描述）。Java 环境本身对新的硬件平台和操作系统是可移植的。Java 编译程序也用 Java 编写，而 Java 运行系统用 ANSIC 语言编写。

h) 高性能

Java 是一种先编译后解释的语言，所以它不如全编译性语言快。但是有些情况下性能是很要紧的，为了支持这些情况，Java 设计者制作了“及时”编译程序，它能在运行时把 Java 字节码翻译成特定 CPU（中央处理器）的机器代码，也就是实现全编译了。Java 字节码格式设计时考虑到这些“及时”编译程序的需要，所以生成机器代码的过程相当简单，它能产生相当好的代码。

i) 多线索性

Java 是多线索语言，它提供支持多线索的执行（也称为轻便过程），能处理不同任务，使具有线索的程序设计很容易。Java 的 lang 包提供一个 Thread 类，它支持开始线索、运行线索、停止线索和检查线索状态的方法。Java 的线索支持也包括一组同步原语。这些原语是基于监督程序和条件变量风范，由 C. A. R. Hoare 开发的广泛使用的同步化方案。用关键词 synchronized，程序员可以说明某些方法在一个类中不能并发地运行。这些方法在监督程序控制之下，确保变量维持在一个一致的状态。

j) 动态性

Java 语言设计成适应于变化的环境，它是一个动态的语言。例如，Java 中的类是根据需要载入的，甚至有些是通过网络获取的^[7]。

3) 技术应用

a) Android 应用

许多的 Android 应用都是 Java 程序员开发者开发。虽然 Android 运用了不同的 JVM 以及不同的封装方式，但是代码还是用 Java 语言所编写。相当一部分的手机中都支持 JAVA 游戏，这就使很多非编程人员都认识了 JAVA。

b) 在金融业应用的服务器程序

Java 在金融服务业的应用非常广泛，很多第三方交易系统、银行、金融机构都选择用 Java 开发，因为相对而言，Java 较安全^[8]。大型跨国投资银行用 Java 来编写前台和后台的电子交易系统，结算和确认系统，数据处理项目以及其他项目。大多数情况下，Java 被用在服务器端开发，但多数没有任何前端，它们通常是从一个服务器（上一级）接收数据，处理后发向另一个处理系统（下一级处理）。

c) 网站

Java 在电子商务领域以及网站开发领域占据了一定的席位。开发人员可以运用许多不同的框架来创建 web 项目，SpringMVC，Struts2.0 以及 frameworks。即使是简单的 servlet，jsp 和以 struts 为基础的网站在政府项目中也经常被用到。例如医疗救护、保险、教育、国防以及其他的不同部门网站都是以 Java 为基础来开发的。

d) 嵌入式领域

Java 在嵌入式领域发展空间很大。在这个平台上，只需 130KB 就能够使用 Java 技术（在智能卡或者传感器上）。

e) 大数据技术

Hadoop 以及其他大数据处理技术很多都是用 Java，例如 Apache 的基于 Java 的 HBase 和 Accumulo 以及 Elasticsearch。

f) 高频交易的空间

Java 平台提高了这个平台的特性和即使编译，他同时也能够像 C++ 一样传递数据。正是由于这个原因，Java 成为的程序员编写交易平台的语言，因为虽然性能不比 C++，但开发人员可以避开安全性，可移植性和可维护性等问题。

g) 科学应用

Java 在科学应用中是很好选择，包括自然语言处理。最主要的原因是因为 Java 比

C++或者其他语言相对其安全性、便携性、可维护性以及其它高级语言的并发性更好^[9]。

2.1.2 SSM 框架介绍

SSM (Spring+SpringMVC+MyBatis) 框架集由 Spring、SpringMVC、MyBatis 三个开源框架整合而成，常作为数据源较简单的 web 项目的框架。

Spring 是一个开源框架，Spring 是于 2003 年兴起的一个轻量级的 Java 开发框架，由 Rod Johnson 在其著作 Expert One-On-One J2EE Development and Design 中阐述的部分理念和原型衍生而来。它是为了解决企业应用开发的复杂性而创建的。Spring 使用基本的 JavaBean 来完成以前只可能由 EJB 完成的事情。然而，Spring 的用途不仅限于服务器端的开发。从简单性、可测试性和松耦合的角度而言，任何 Java 应用都可以从 Spring 中受益。简单来说，Spring 是一个轻量级的控制反转 (IoC) 和面向切面 (AOP) 的容器框架。

Spring MVC 属于 SpringFrameWork 的后续产品，已经融合在 Spring Web Flow 里面。Spring MVC 分离了控制器、模型对象、分派器以及处理程序对象的角色，这种分离让它们更容易进行定制。SpringMVC 优点角色之间的分工比较明确，相比 Struts 不用过多的配置过多的属性。分工是清楚的，并且膨胀点是相当灵活，并且可以容易地扩大，虽然几乎不需要它。由于 pojo 作为命令对象，框架上的一些 API 就不需要去继承了，直接作为业务对象。可以跟其他的 spring 框架很好的相结合。可适配性能比较优秀，通过 HandlerAdapter 能够很好的支撑其他的一些类来当处理器。可定制性比较强，通过 HandlerMapping、ViewResolver 等可以进行一些比较简单的定制。

MyBatis 本是 apache 的一个开源项目 iBatis,2010 年这个项目由 apache software foundation 迁移到 google code,并且改名为 MyBatis。MyBatis 是一个基于 Java 的持久层框架。iBATIS 提供的持久层框架包括 SQL Maps Data Access Objects(DAO) MyBatis 消除了几乎所有的 JDBC 代码和参数的手工设置以及结果集的检索。MyBatis 使用简单的 XML 或注解用于配置和原始映射，将接口和 Java 的 POJOs (Plain Old Java Objects, 普通的 Java 对象) 映射成数据库中的记录^[10]。

2.2 Servlet 技术

2.2.1 servlet 简介

Servlet (Server Applet), 全称 Java Servlet。是用 Java 编写的服务器端程序。其主要功能在于交互式地浏览和修改数据,生成动态 Web 内容。狭义的 Servlet 是指 Java 语言实现的一个接口,广义的 Servlet 是指任何实现了这个 Servlet 接口的类,一般情况下,人们将 Servlet 理解为后者。

Servlet 运行于支持 Java 的应用服务器中。从原理上讲,Servlet 可以响应任何类型的请求,但绝大多数情况下 Servlet 只用来扩展基于 HTTP 协议的 Web 服务器。

服务器上需要一些程序,常常是根据用户输入访问数据库的程序。这些通常是使用公共网关接口(Common Gateway Interface, CGI)应用程序完成的。然而,在服务器上运行 Java,这种程序可使用 Java 编程语言实现。在通信量大的服务器上,JavaServlet 的优点在于它们的执行速度更快于 CGI 程序。各个用户请求被激活成单个程序中的一个线程,而无需创建单独的进程,这意味着服务器端处理请求的系统开销将明显降低。

最早支持 Servlet 技术的是 JavaSoft 的 Java Web Server。此后,一些其它的基于 Java 的 Web Server 开始支持标准的 Servlet API。Servlet 的主要功能在于交互式地浏览和修改数据,生成动态 Web 内容。这个过程为:

客户端发送请求至服务器端;

服务器将请求信息发送至 Servlet;

Servlet 生成响应内容并将其传给服务器。响应内容动态生成,通常取决于客户端的请求;

服务器将响应返回给客户端。

Servlet 看起来像是通常的 Java 程序。Servlet 导入特定的属于 Java Servlet API 的包。因为是对象字节码,可动态地从网络加载,可以说 Servlet 对 Server 就

如同 Applet 对 Client 一样,但是,由于 Servlet 运行于 Server 中,它们并不需要一个图形用户界面。从这个角度讲,Servlet 也被称为 FacelessObject。

一个 Servlet 就是 Java 编程语言中的一个类,它被用来扩展服务器的性能,服务器上驻留着可以通过“请求-响应”编程模型来访问的应用程序。虽然 Servlet 可以对任何类型的请求产生响应,但通常只用来扩展 Web 服务器的应用程序。

工作模式为,客户端发送请求至服务器,服务器启动并调用 Servlet,Servlet 根据客户端请求生成响应内容并将其传给服务器,服务器将响应返回客户端^[11]。

2.2.2 servlet 特征

Servlet 最大的优势就在于一方面它是用 Java 语言写的,是一个 Java 类,因而,Java 语言赋予了它强大的功能,另一方面,它又可以用来处理客户端的请求,并且可以返回响应.这两方面的结合使 Servlet 成为功能非常强大的服务端语言,总地来说,它主要有以下方法的优势:

- 1) 可移植性好,由于 Servlet 用 Java 编写,Servlet API 具有完善的标准,因此,为 WEB 服务器写的 Servlet 无需任何实质上的改动即可移植到 Apache,Microsoft IIS 或者 WebStar,几乎所有的主流服务器都直接或通过插件支持 Servlet。
- 2) 功能强大.由于 Servlet 本质上是 Java 类,它可以使用 JavaAPI 核心的所有功能,这些功能包括 WEB 和 URL 访问,图像处理,数据压缩,多线程,JSBC,序列化对象等。
- 3) 安全性好.ServLet 的安全问题有多个层次的保障,首先,Servlet 是 Java 类,它可以使用 Java 语言的安全框架,其次,Servlet API 是类型安全的,再次,窗口也可以给 Servlet 的安全进行管理。
- 4) 简洁.Servlet 代码面向对象,在封闭方面有天生的优势。
- 5) 高效耐久.一旦 Servlet 载入到 WEB 服务器中,它就会驻留在内存中,这样就加快了响应的速度^[12]。

2.3 课题要达到的设计目标

顺风车系统设计要达到的目标是：

- (1) 能够实现车主发布行程。
- (2) 能够实现用户对符合要求的顺风车进行查询。
- (3) 能够实现车主的信誉评价，并存入数据库作为车主显示顺序的标准。
- (4) 能够实现车主用户同账号功能，即同一个账号可以作为车主或用户。
- (5) 能够实现用户自主选择符合条件的车辆。
- (6) 能够实现车主和用户自主沟通。

第三章 系统的需求分析

3.1 可行性分析

可行性研究就是预开发软件系统的总体目标，给出它的功能、性能、可靠性以及其他方面的要求。其目的是用最小的代价在尽可能短的时间内确定问题是否能够解决、是否值得去解决。

下面从四个方面分析本系统的可行性：

(1) 技术可行性

本系统采用 navicat for mysql 进行后台数据库的管理、操作和维护，用 eclipse 进行前台界面设计、与后台数据库的接口、数据的录入、查询、浏览等功能。根据 navicat for mysql 的特点与 eclipse 强大方便快捷的开发工具，以及他们当前的广泛实际应用，充分说明本网站在技术方面可行。

(2) 经济可行性

经济可行性分析（能否赢利）是，从经济的角度分析网站系统的规划方案有无实现的可能和开发的价值；分析网站系统所带来的经济效益是否超过开发和维护网站所需要的费用。经济上的可行性是企业行为的基础。

由于顺风车系统的开发设计是为完成毕业设计任务而进行的，所以不存在赢利问题。所在此也就不讨论经济可行性问题。

(3) 开发环境可行性

网站开发需要的开发环境与平台无关，只需要一台安装了 Java 运行环境 Jdk 的计算机即可。所以开发环境可行性是没有任何问题的。

3.2 开发环境

顺风车系统开发环境如下：

- (1) 硬件配置环境：无特殊要求
- (2) 操作系统：无特殊要求、
- (3) 数据库：开发过程中数据库工具采用了 mysql^[13]。Mysql 有如下优点：它使用核心线程是完全多线程，支持多处理器。有多种列类型：1、2、3、4、和 8 字节长度自有符号 / 无符号整数、FLOAT、DOUBLE、CHAR、VARCHAR、TEXT、BLOB、DATE、TIME、DATETIME、TIMESTAMP、YEAR、和 ENUM 类型^[14]。它通过一个高度优化的类库实现 SQL 函数库并像他们能达到的一样快速，通常在查询初始化后不该有任何内存分配。没有内存漏洞。全面支持 SQL 的 GROUP BY 和 ORDER BY 子句，支持聚合函数 (COUNT ()、COUNT (DISTINCT)、AVG ()、STD ()、SUM ()、MAX () 和 MIN ())^[15]。你可以在同一查询中混来自不同数据库的表。支持 ANSI SQL 的 LEFT OUTER JOIN 和 ODBC^[16]。所有列都有缺省值。你可以用 INSERT 插入一个表列的子集，那些没用明确给定值的列设置为他们的缺省值。MySQL 可以工作在不同的平台上^[17]。支持 C、C++、Java、Perl、PHP、Python 和 TCL API^[18]。
- (4) 编码工具：采用 eclipse。Eclipse 是一个开放源代码的、基于 Java 的可扩展开发平台。就其本身而言，它只是一个框架和一组服务，用于通过插件组件构建开发环境。幸运的是，Eclipse 附带了一个标准的插件集，包括 Java 开发工具 (Java Development Kit, JDK)。^[19]

3.3 研究设计中要解决的问题

任何一套系统的编辑和设计中都会遇到这样或者那样的问题，经常遇到这样或者那样的困难也是免不了的。遇到了问题和困难，我们就要想办法去解决这些问题和困难。我们可以翻阅和参考资料和书籍，请教教师和同学，还可以在网上查找一些相应的有关材料。我们必须先要分析出现问题的原因，是什么地方出了问题，然后再寻找解决问题和困难的方法。要从多个角度思考问题。为了解决困难我们要多做一些实验性的工作，

利用多种方法，多种函数，去攻克困难。我们千万不要钻牛角筋，走死胡同，这样做是没有用的，是徒劳的。

在研究设计顺风车系统中要解决的问题有以下几个方面：

1) 角色管理

系统设置两种角色，即车主和乘客。登录界面选择车主或乘客登录。

2) 车主显示先后问题

为车主增加评分功能，评分高的优先显示。

3) 行程管理

根据添加时间显示同一车主的行程信息

4) 线路查询

可以根据日期或关键字检索查询相关出行信息

5) 行程发布

车主提前在系统中发布自己即将开始的行程，并填写所有的相关信息。包括开始时间，起点，途径哪些地理位置，可搭载人数，车牌号，收费情况，联系电话等内容。

3.4 系统结构描述

MVC 是一种设计模式，是比较简单的 Web 界面的开发的设计模式，我们只需要直接写一个 JSP^[20] 页面即可完成，但现在将 Web 客户端三层，完整的名字叫模型视图控制器，可以减低系统的耦合，从而达到使系统架构和分层变得清晰的目的。系统结构如图所示。

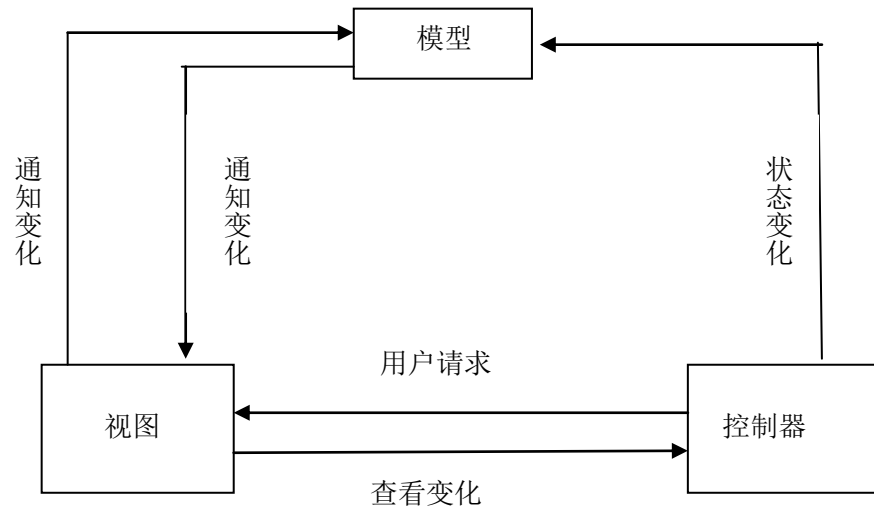


图 3-1 系统层结构处理流程

第四章 系统总体设计

4.1 顺风车系统设计实现的策略

顺风车系统的设计思想把整个系统分成登录，注册两个模块。具体分别如下所示：

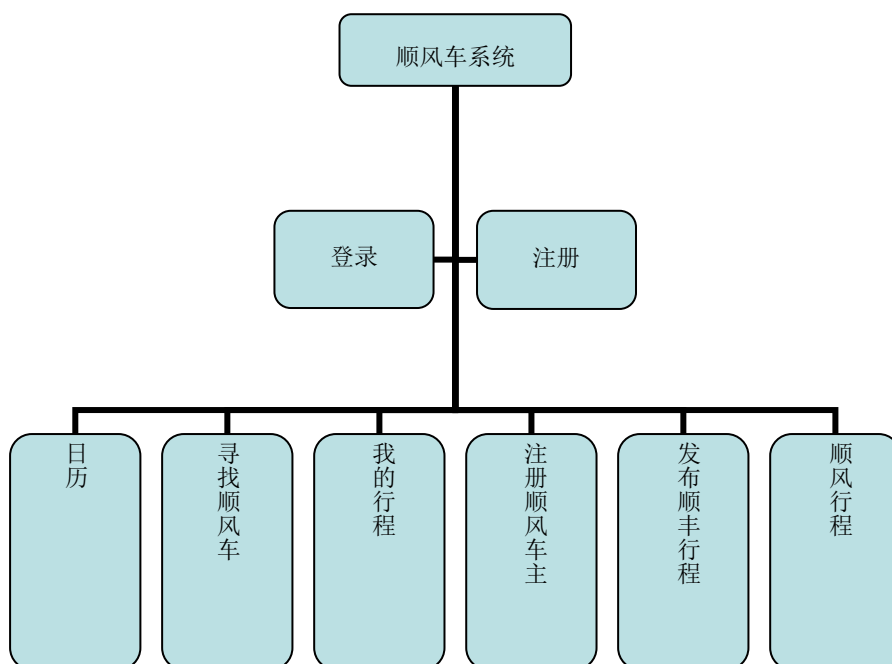


图 4-1 系统的结构图

登录为已经注册顺风车的用户或车主登录。

注册为未注册的用户注册为用户。

日历界面为成功登录用户会自动跳转界面。

寻找顺风车界面为普通用户查询合适的顺风车界面。

我的行程界面，若普通用户则显示过往的出行记录和车主对你的评价；若为车主则显示车主过往的所有出行记录和用户对车主的评价。

注册顺风车主界面为已用户注册为普通但尚未成功注册车主的用户提供。通过提供车牌号和车型成为车主。

发布顺风行程为车主用户将即将出行的信息以及可以搭乘人数和价格发布。

顺风行程为所有已发布的顺风车信息。

4.2 系统运行效果图



顺风车系统
用户登陆

1234

密码

登录

注册

@2017 毕业设计

图 4-2 登录界面



顺风车系统用户注册

手机号

姓名

密码

年龄

提交 返回

图 4-3 注册界面

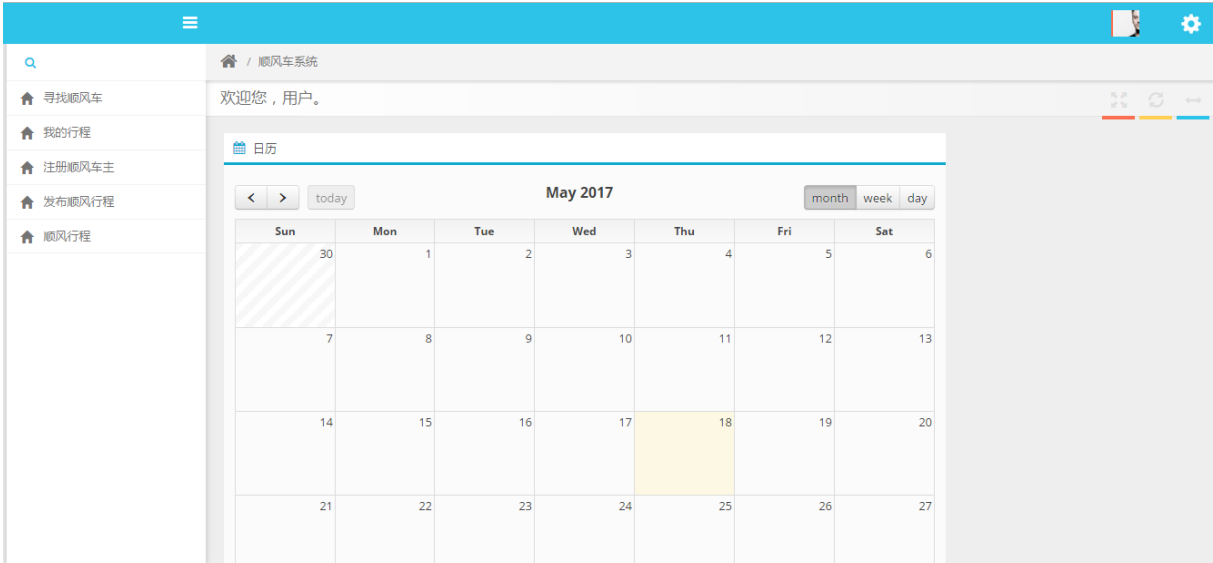


图 4-4 日历界面

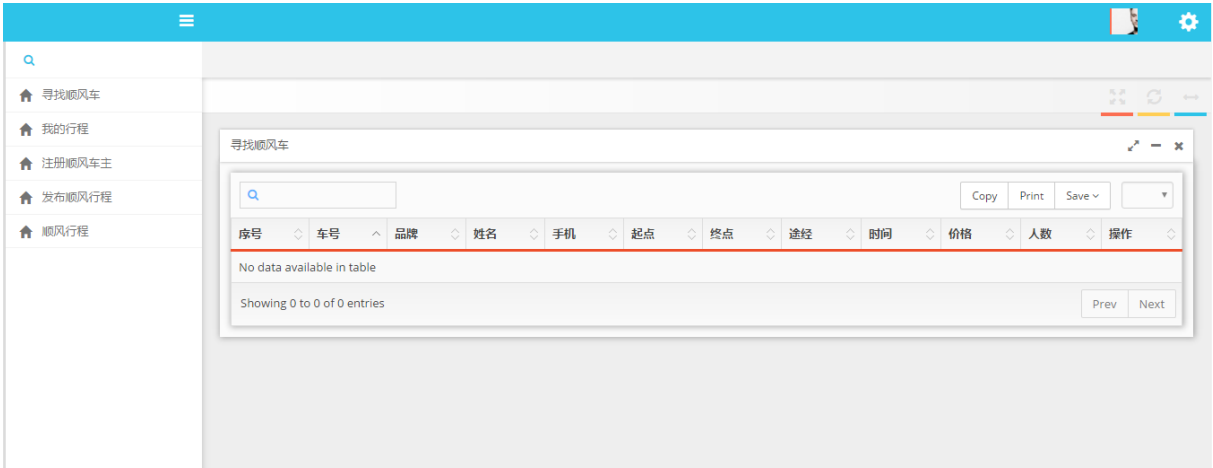


图 4-5 寻找顺风车界面



图 4-6 我的行程界面

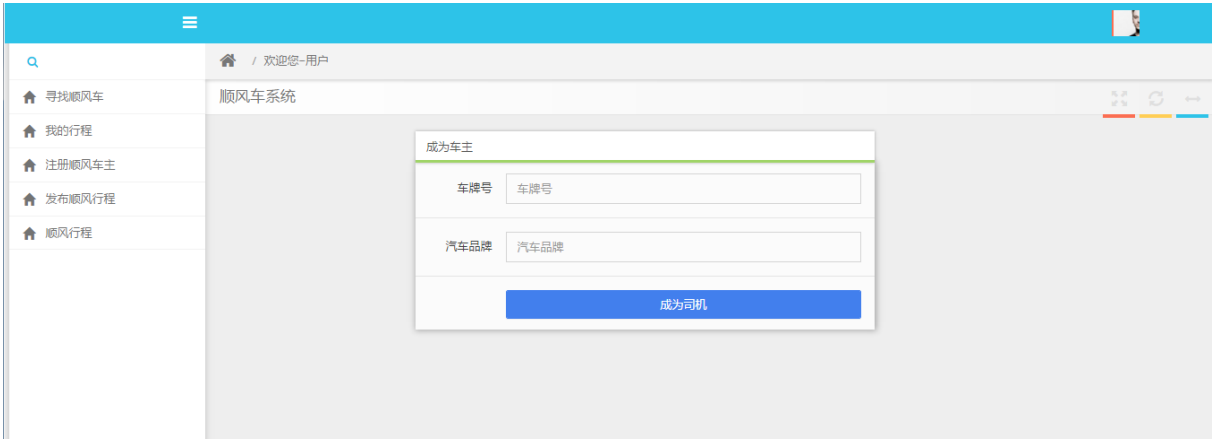


图 4-7 注册顺风车主界面



图 4-8 发布顺风行程界面

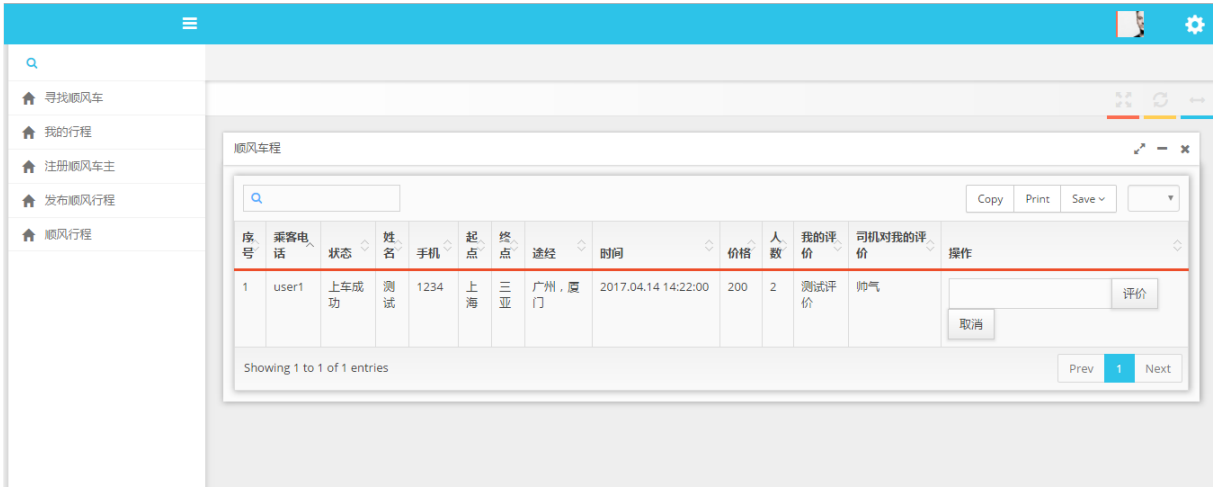


图 4-9 顺风行程界面

第五章系统详细设计

5.1 数据库设计

数据库被定义为是在计算机存储设备上长期存放的、有组织的、可共享的、合理存放的、互相关联的数据集合。

由于本章讲述的对文件传输系统的实现中要多次连接数据库，并进行查询、增加、修改等操作，因此，下面有必要先简述一下，ActiveX 数据对象（ADO），以及 ADO 编程^[21]。

5.1.1 基本的 ADO 编程模型

ActiveX 数据对象作为微软公司数据库接口的组成部分，其定义了一组 COM（通用对象模型）对象，用以操作不同的数据源的数据。ADO 为不同的数据类型提供了一个非常通用的接口，而你则不用也没必要去管这些接口究竟是在单机还是分布在网络中。因为，在 ADO 中使用的是 COM 对象，是它们很容易在众多的编程环境中使用，包括：VB，Visual C，甚至 Java Script^[22]。

其中，使用 ADO 的执行操作按以下过程进行。

- (1) 连接到数据源。同时，可确定对数据源的所有更改是否已成功或没有发生。
- (2) 指定访问数据源的命令，同时可带变量参数，或优化执行。
- (3) 执行命令。
- (4) 如果这个命令是数据按表中的行的形式返回，则将这些行存储在易于检查、操作或更改的缓存中。
- (5) 适当情况下，可使用缓存行的更改内容来更新数据源。

(6) 提供常规方法检测错误（通常有建立连接或执行命令造成）。

这些只是在典型情况下，编程模型中使用的步骤，由于 ADO 又很强的灵活性，因此，一般情况下，模型只需执行部分步骤。本文的各个功能模型中使用数据库时，也只是执行上面的部分操作。

5.1.2 数据库选择

现在，世界上绝大数的数据库都采用关系模型，至于关系模型的优点，以及概念、完整性约束等内容，由于本文不是专门研究数据库的，这里就不进行详细的讨论了，只需知道这里的数据库也是关系数据库[23]。目前，主流的数据库系统主要有 Oracle，Microsoft SQL Server 2000，Sybase，Microsoft Access，Informix，IBM 的 DB2 等。由于 ADO 是微软公司的产品，因此它主要支持 Microsoft SQL Server 2000 和 Microsoft Access 两种产品。Microsoft SQL Server 2000 是一种专门的大型数据库系统，提供完全的数据库操作；Microsoft Access 只是 Microsoft Office 附属的，提供一般的数据库操作。而本系统所涉及的数据库内只含有一张表，所执行的工作也只是一般的增加、查询、删除和修改等操作，为了简单易行，所以选择了 MySQL 来设计数据库^[24]。

5.2 数据库结构图

5.2.1 数据库 CDM（概念数据模型）图

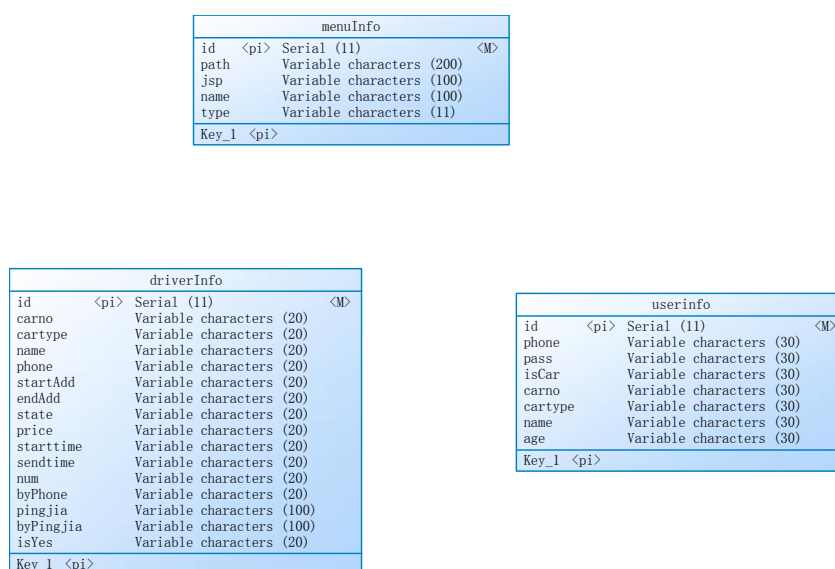


图 5-1 CDM 图

5.2.2 数据库结构表

利用 navicat for mysql 创建数据库，库文件名为 db_sfc.sql.

根据以上数据项和数据结构，及其 ER 图，数据库包含如下几个表文件：

(1) 用户信息如表 5-1 所示：

表 5-1 用户信息表

字段名称	数据类型	字段大小
id	int	11
phone	varchar	30
pass	varchar	30
isCar	varchar	30
carno	varchar	30
cartype	varchar	30
name	varchar	30
age	varchar	30

(2) 出行信息如表 5-2 所示：

表 5-2 出行信息表

字段名称	数据类型	字段大小
id	int	11
carno	varchar	20
cartype	varchar	20
name	varchar	20
phone	varchar	20
startAdd	varchar	20
endAdd	varchar	20
state	varchar	20

price	varchar	20
starttime	varchar	20
sendtime	varchar	20
num	varchar	20
byPhone	varchar	20
pingjia	varchar	100
byPingjia	varchar	100
isYes	varchar	20

(3) 菜单信息如表 5-3 所示：

表 5-3 菜单信息表

字段名称	数据类型	字段大小
id	int	11
path	varchar	200
jsp	varchar	100
name	varchar	100
type	varchar	11

到此，系统的全部数据库的表结构设计已经全部列出，在下面的章节里，将全面介绍顺风车系统的实现^[25]。

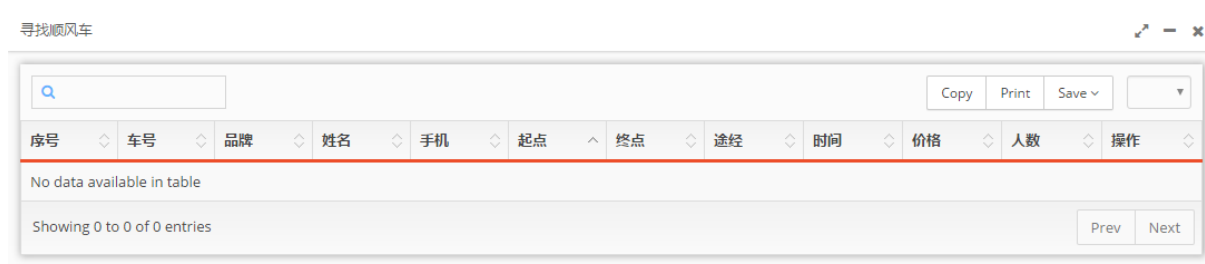
第六章 程序模块实现

6.1 分模块详述系统各部分的实现方法

6.1.1 寻找顺风车模块的实现方法

用户登录后，选择进入寻找顺风车界面（orderCarList.jsp），在这里用户可以看到出行车主的所有信息，包括车号，品牌，车主姓名，车主手机以及关于出行的一些信息，包括起点，终点，途径地点，所需时间，支付费用以及可以搭乘人数^[26]。

示例图如下所示：



6-1 寻找顺风车模块图

主要实现代码如下：

orderCarList.jsp 部分代码

```
<div class="main-container container-fluid">
    <!-- Page Container -->
    <div class="page-container">
        <!-- Page Sidebar -->
        <div class="page-sidebar" id="sidebar">
            <!-- Page Sidebar Header-->
            <div class="sidebar-header-wrapper">
                <input type="text" class="searchinput" /> <i
                    class="searchicon fa fa-search"> </i>
                <div class="searchhelper">搜索需要的菜单</div>
            </div>
            <!-- /Page Sidebar Header -->
            <!-- Sidebar Menu -->

            <ul class="nav sidebar-menu">
                <!--Dashboard-->
```

```
<c:forEach
```

```
items="<%=request.getSession().getAttribute(\"menuList\")%>"
```

```
var="vi" varStatus="status">
```

```
<li>
```

```
<a href="<%=request.getContextPath()%>/menu/${vi.path}">
```

```
<i class="menu-icon glyphicon glyphicon-home"></i>
```

```
<span class="menu-text"> ${vi.name} </span>
```

```
</a>
```

```
</li>
```

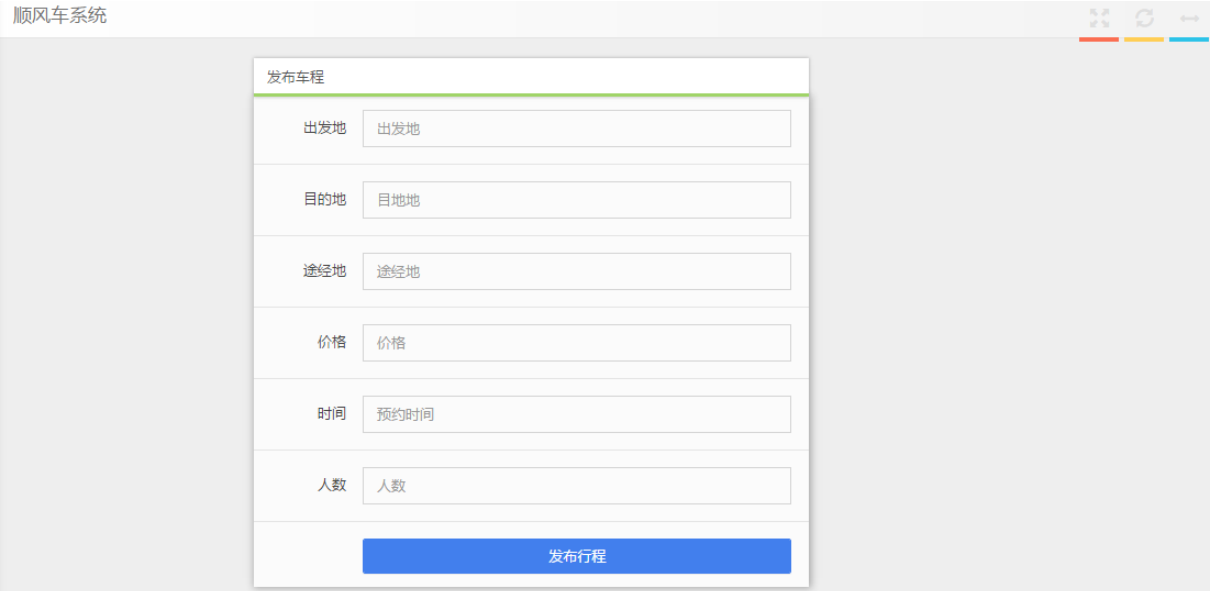
```
</c:forEach>
```

```
<!--UI Elements-->
```

```
</ul>
```

6.1.2 发布顺风车行程模块的实现方法

发布顺风车行程（adddriver.jsp）的实现方法是，注册成为顺风车主的用户填写出行相关信息发布顺风车行程^[27]。示例图如下所示：



顺风车系统

发布行程

出发地 出发地

目的地 目的地

途经地 途经地

价格 价格

时间 预约时间

人数 人数

发布行程

图 6-2 发布顺风车行程模块图

主要实现代码如下：

adddriver.jsp 部分代码

```
<div class="page-content">  
    <!-- Page Breadcrumb -->  
    <div class="page-breadcrumbs">
```

```

        <ul class="breadcrumb">
            <li>
                <i class="fa fa-home"></i>
                <a href="#"></a>
            </li>
            <li class="active">欢迎您~用户</li>
        </ul>
    </div>
    <!-- /Page Breadcrumb -->
    <!-- Page Header -->
    <div class="page-header position-relative">
        <div class="header-title">
            <h1>
                顺风车系统
            </h1>
        </div>
        <!--Header Buttons-->
        <div class="header-buttons">
            <a class="sidebar-toggler" href="#">
                <i class="fa fa-arrows-h"></i>
            </a>
            <a class="refresh" id="refresh-toggler" href="">
                <i class="glyphicon glyphicon-refresh"></i>
            </a>
            <a class="fullscreen" id="fullscreen-toggler" href="#">
                <i class="glyphicon glyphicon-fullscreen"></i>
            </a>
        </div>
        <!--Header Buttons End-->
    </div>

```

6.1.3 顺风行程模块的实现方法

顺风行程（driverList.jsp）的实现方法是，调用数据库中的出行信息表，将所有的出行记录显示^[28]。示例图如下所示：

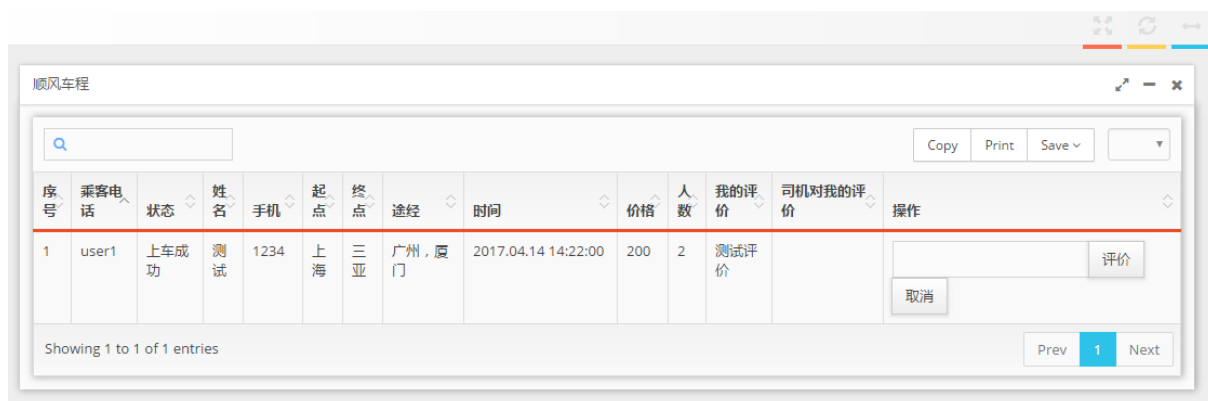


图 6-3 顺风行程模块图

主要代码如下：

driverList.jsp

```

<div class="main-container container-fluid">
    <!-- Page Container -->
    <div class="page-container">
        <!-- Page Sidebar -->
        <div class="page-sidebar" id="sidebar">
            <!-- Page Sidebar Header-->
            <div class="sidebar-header-wrapper">
                <input type="text" class="searchinput" /> <i
                    class="searchicon fa fa-search"> </i>
            </div>
            <!-- /Page Sidebar Header -->
            <!-- Sidebar Menu -->

            <ul class="nav sidebar-menu">
                <!--Dashboard-->
                <c:forEach

items="<%=request.getSession().getAttribute(\"menuList\")%>"
                    var="vi" varStatus="status">

                    <li>
                        <a href="<%=request.getContextPath()%>/menu/${vi.path}">
                            <i class="menu-icon glyphicon glyphicon-home"> </i>
                            <span class="menu-text"> ${vi.name} </span>
                        </a>
                    </li>
                </c:forEach>

            <!--UI Elements-->
        </ul>
    <!-- /Sidebar Menu -->

```

6.2 关于课题的下一步工作

(1) 在表格的设计方面还有些不足，在数据多的时候，数据在表格中就会挤到一起，这是需要改进的。

(2) 在选择数据库后台方面还有些不足的地方，由于计算机配置所限，只能采用mysql 数据库，mysql 数据库只能应用于小型的数据量，所以这是需要改进的^[29]。

(3) 该系统只限于用户相对熟悉的地方顺风出行，和其他那些大型网约车软件有很大一部分差距。对顺风车，车主和用户的相互信任极其重要，而本系统没有对用户信息的真伪进行检测，可能存在极大的风险。

(4) 没有实现地图显示功能，即用户不能实时查看目的地与自己所需目的地之间的距离，以及是否合适。

第七章 系统测试

程序编写出来并不是软件开发的结束，因为，一个可靠的程序必须经得住考验，这就是调试与测试的工作了。不可否认的是，这一步是软件开发的最后一步，也是重要的一步。测试不仅包括，系统调试与测试，同时还包含后期的系统维护，当然，考虑到这是毕业设计的内容，因此，系统维护可以暂时不计算在内，但这不能影响测试在本系统开发中的重要地位。

7.1 测试目的

测试是一个为了寻找错误而运行程序的过程，它的目的就是找出尽可能多的程序漏洞，对系统进行改善，以便客户运行时得以顺利进行，而不发生故障。一个好的测试的标准就是指找到迄今为止尚未发现的错误，并且，一个成功的测试的标准则是指揭示了迄今为止尚未发现的错误，可见，发现错误后漏洞是一个测试的重要目标。

7.2 测试结果

经过对程序的调试与测试我发现了许多编程时容易出现的问题，有些问题是大家普遍存在的，而有些则是我个人容易出现的。

首先，代码运行前必须关闭外部 tomcat，否则 eclipse 中启动服务器会因为端口占用而不能正常启动 tomcat。

其次，代码运行时必须连接互联网，因为系统需要下载运行时所需要的 jar 包。否则会产生一系列的错误^[30]。

虽然系统可以在 eclipse 中运行，但是作者建议在外部浏览器访问，因为 eclipse 内置浏览器不能很好的展示系统的界面效果。

ADO 编程时，无连接状态下数据操作应注意的几个问题。由于保持连接状态下的数

据操作，有许多的不便之处，这时一定要注意，在操作之前要先对数据库和内存缓冲区进行绑定；而数据操作的对象都是内存中的数据；若进行了增加，修改等操作，在 ADO 编程^[31] 结尾处一定要更新数据库，将内存中的数据写入数据库；否则，所有修改数据库的数据操作都无效。

结束语

网络的飞速发展，必将给人类的生活带来方便，它表现在方方面面，顺风车为人们学习提供了很好的出行环境。现今，各种顺风车都在 WWW 上提供了线路查询和出行。在毕业设计中，我研究了如今 WWW 上的一些技术，总结了顺风车系统应具备的主要功能，查阅了许多关于顺风车系统中的技术问题的资料，并与同学交流来掌握 ASP 技术。同时在动手实践过程中，通过一些算法的实现，使我的编程能力有了一定的进步，数据结构的知识也有了一定的巩固，面向对象的思想 and 编程思想在头脑中也初步形成了。

我们采用目前最流行的基于 WWW 的三层体系结构来开发网上远程考试系统，中间层使用的是流行的 Servlet 技术。系统中我们实现了网上考试及远程交流等功能，功能比较完备，界面友好。

我认为我们的这个课题很好，能帮助您理解和消化很多以前在书本中未理解和涉及到的知识，但很遗憾我们并未将其完全实现，这是我们的遗憾，系统在交互性、针对性方面还有待加强，完善。

网络存在的实际价值就在于它给了人们一个交流的机会，交流最有价值的东西就是知识。网上考试系统必将随着网络的发展，而更加完善。会有越来越多的人投入这方面的研究。

这次毕业设计是我从理论走向实践迈出的坚实的一步。从分析到设计直到具体实现，原来学的理论知识对我有很大的指导，我也发现了其中有很多不足的地方。通过这次毕业设计，我深深体会到，一个好的程序设计人员必须具有认真、谨慎的作风，与人合作的精神，和坚忍不拔的毅力。我会向着这个方向去努力。

致谢

在本论文完成之际，首先要向我的辅导老师张黔川致以诚挚的谢意。在论文的写作过程中，张黔川老师给了我许许多多的帮助和关怀。在此我谨向张黔川老师表示衷心的感谢和深深的敬意。

同时，我要感谢我们学院给我们授课的各位老师，正是由于他们的传道、授业、解惑，让我学到了专业知识，并从他们身上学到了如何求知治学、如何为人处事。我也要感谢我的母校中国民航飞行学院，是她提供了良好的学习环境和生活环境，让我的大学生活丰富多姿，为我的人生留下精彩的一笔。

另外，衷心感谢我的同窗同学们，在我毕业论文写作中，与他们的探讨交流使我受益颇多；同时，他们也给了我很多无私的帮助和支持，我在次深表谢意。

最后，向我的亲爱的家人和亲爱的朋友表示深深的谢意，他们给予我的爱、理解、关心和支持是我不断前进的动力。

学无止境。明天，将是我终身学习另一天的开始。

参考文献

1. 李刚. 疯狂 Java 讲义 (第 2 版): 电子工业出版社, 2014: 2
2. 赵景晖. Java 程序设计: 北京机械工业出版社, 2005: 1-2
3. 明日科技. Java 从入门到精通 (第 3 版): 清华大学出版社, 2014
4. 王薇, 杜威. Java 程序设计与实践: 华中科技大学出版社, 2010: 9
5. 周志明. 深入理解 Java 虚拟机: 机械工业出版社, 2011: 4
6. 王薇, 杜威. Java 程序设计与实践: 华中科技大学出版社, 2010: 10
7. 程虎. Java 语言的特点[J]. 现代科学仪器, 1997(2).
8. 李钟尉, 陈丹丹, 张振坤. Java 编程宝典: 人民邮电出版社, 2010: 5
9. 现实世界中哪些地方用到了 Java(<http://www.csdn.net/article/1970-01-01/2823051>)
10. SSM 框架整合教程 (<http://blog.csdn.net/gebitan505/article/details/44455235>)
11. 沈泽刚, 秦玉平, Java Web 编程技术, 清华大学出版社, 2010;3
12. Servlet 的特点(<http://www.djjwz.com/news/news4000.html>)
13. 明日科技 编著. Java Web 从入门到精通. 北京: 清华大学出版社, 2012-09-01.
14. BruceEckel. 《Java 编程思想》. 机械工业出版社, 2003.
15. 覃庆炎 等编著. Java 开源项目: Spring+Hibernate+Struts 项目开发详解. 北京: 清华大学出版社, 2008 .
16. ADO 教程 (<http://www.runoob.com/ado/ado-tutorial.html>)
17. 王小东, Baron. 高性能 MySQL[M]. 北京: 电子工业出版社, 2010.
18. 邹竹彪. JSP 网络编程从入门到精通[M]. 北京: 清华大学出版社, 2007.
19. 科夫勒. MySQL 权威指南[M]. 北京: 人民邮电出版社, 2006.
20. Charles A. Bell. 深入理解 MySQL[M]. 北京: 人民邮电出版社, 2010.
21. 萨师煊, 王珊. 《数据库系统概论》. 高等教育出版社, 2002.
22. Brown 等. 《JSP 编程指南 (第二版)》. 电子工业出版社, 2003.
23. 清宏计算机工作室. 《JSP 编程技巧》. 机械工业出版社, 2004.
24. 朱红, 司光亚. 《JSP Web 编程指南》. 电子工业出版社, 2001.
25. Gavin King. Hibernate in Action. Manning Publications[M]. 2004
26. Cay S. Horstmann, Gary Cornell. Core Java2, Volume I-Fundamentals[M]. 北京: 机械工业出版社, 2004
27. Elliotle R. Java network programming O' Reilly[M]. 北京: 机械工业出版社, 2011.
28. Bruce E. Thinking in Java[M]. 北京: 人民邮电出版社, 2012.
29. Ted Husted. Struts In Action[M]. 北京: 电子工业出版社, 2013.
30. 张海藩. 《软件工程导论》. 清华大学出版社. 2003.
31. 姜承尧. MySQL 技术内幕[M]. 北京: 机械工业出版社, 2011.