**《西工大拼车系统》**

**概要设计说明书**

**学 院 软件学院**

**年 级 2020**

**组 员 李家源**

**吴安康**

**王健挺**

**吴紫妍**

**2023 年 4 月 20日**

**版本历史**

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| 2023.4.19 | v0.1.0 | 概要设计文档完成 | 吴紫妍 |
| 2023.4.20 | v0.2.0 | 功能部分修改 | 吴紫妍 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

[1.引言 1](#_Toc116049859)

[1.1编写目的（必填） 1](#_Toc116049860)

[1.2项目背景 1](#_Toc116049861)

[1.3定义 2](#_Toc116049862)

[1.4参考资料 2](#_Toc116049863)

[2.任务概述 2](#_Toc116049864)

[2.1目标 2](#_Toc116049865)

[2.2运行环境（必填） 3](#_Toc116049866)

[2.3开发环境（必填） 3](#_Toc116049867)

[2.4需求概述 3](#_Toc116049868)

[3.总体设计 4](#_Toc116049870)

[3.1基本设计概念和处理流程 4](#_Toc116049871)

[3.2系统总体功能结构（必填） 8](#_Toc116049872)

[3.3功能需求描述（必填） 8](#_Toc116049873)

[3.4界面设计（必填） 1](#_Toc116049874)2

[4.接口设计 1](#_Toc116049875)2

[4.1外部接口 1](#_Toc116049876)2

[4.2内部接口（必填） 1](#_Toc116049877)2

[5.数据结构设计 1](#_Toc116049878)3

[5.1逻辑结构设计 1](#_Toc116049879)3

[6.运行设计 1](#_Toc116049881)4

[6.1运行控制 1](#_Toc116049883)4

[6.2运行时间 1](#_Toc116049884)4

[7.出错处理设计 1](#_Toc116049885)5

[7.1出错输出信息 15](#_Toc116049886)

[7.2出错补救措施 15](#_Toc116049887)

[8.安全保密设计 15](#_Toc116049888)

[9.维护设计 15](#_Toc116049889)

**概要设计说明书**

# 1.引言

## 1.1编写目的（必填）

《西工大拼车系统》概要设计说明书用于在开发此系统的过程中为系统做一个总体的结构设计，在经过评审后进一步细化，分别对每一模块研究细化的解决方案、接口和数据库等方面的设计，明确描述系统的输入输出类型逻辑算法以及调用关系。

为开发人员、评审人员和测试人员确定方案和编写测试用例提供依据。

## 1.2项目背景

本产品能够在Windows、MacOS、Linux系统上实现部署，用于提供西北工业大学新校区至老校区以及地铁二号线、六号线地铁站的拼车服务。

本系统基于《西工大拼车系统软件需求规格说明书》实现用户所需功能需求及非功能需求（详见《西工大拼车系统软件需求规格说明书》*5.1.1功能需求总表*），在联网状态下能够提供上述需求对应服务，在规定时间内对用户输入做出反应成功率达95%，界面清晰，易用性高，前后端分离开发便于维护，基于Java+SpringBoot开发可移植性高。

本《概要设计》主要描述系统的功能及运行环境、系统性能等非功能需求；描述系统接口设计、数据结构设计、运行设计、出错设计、安全保密设计以及维护设计。

本系统任务提出者：**李家源**

开发者：**王健挺，李家源，吴安康，吴紫妍**

用户：**西工大师生、工作人员**

## 1.3定义

|  |  |  |
| --- | --- | --- |
| **序号** | **描述** | **描述解释** |
| 1 | 车主发车订单 | 填写该发车订单的车主昵称、车牌号、发车日期、终点、价格、剩余座位数以及备注信息 |
| 2 | 乘客拼车申请 | 选择乘车人数、备注 |
| 3 | 我的历史发车信息 | 历史发车订单的终点和时间 |
| 4 | 我的历史发车详细信息 | 每单的车主昵称、车牌号、发车日期、终点、每个拼车账号的人数及备注 |
| 5 | 我的历史拼车信息 | 历史拼车订单的终点和时间 |
| 6 | 我的历史拼车详细信息 | 每单的车主昵称、车牌号、发车日期、终点、每个拼车账号的人数及备注 |
| 7 | 司机信息（管理员） | 每个师傅的昵称、联系电话、订单数目、订单成功率、积分 |

## 1.4参考资料

|  |  |  |  |
| --- | --- | --- | --- |
| 文档编号 | 标题 | 修订版本 | 日期 |
| 01 | 《西工大拼车系统软件需求规格说明书》 | v0.2 | 2023.4.18 |
| 02 | 《西工大拼车系统项目开发计划》 | v0.2 | 2023.4.20 |
| 03 | 西工大拼车系统软件详细设计说明书》 | v0.2 | 2023.4.25 |

# 2.任务概述

## 2.1目标

该项目开发主要用于便利西工大周围交通，目标让广大本校师生及工作人员应用到此款软件。本软件主要使用JavaScript+Java语言，使用Bootstrap + Spring Boot框架以及MySQL数据库实现开发。

## 2.2运行环境（必填）

Java6+环境下PC端联网使用

## 2.3开发环境（必填）

Windows 10

|  |  |  |
| --- | --- | --- |
| 资源类型 | 软件名称 | 用途 |
| 编程开发工具 | WebStorm  Idea | 负责代码开发 |
| 文档编写工具 | Office | 负责编制文档 |
| 图片处理工具 | Xmind、PowerDesigner | 负责图片处理、功能图绘制 |
| 即时通讯工具 | QQ | 开会交流 |
| 配置库 | Github | 负责文档、代码存储共享 |
| 外部接口 | 高德地图 | 提供地图接口 |

## 2.4需求概述

功能需求：

1. 车主能够发布拼车信息
2. 乘客能够选择合适的车辆进行拼单申请
3. 车主能够决定是否同意此拼单申请
4. 管理员能够导出一段时间内的拼车记录或某位车主的拼单记录，并 按照车主拼单成功的数量对车主进行排名，表彰。

界面要求：

1. 界面统一风格
2. 管理员和用户界面分离

性能要求：

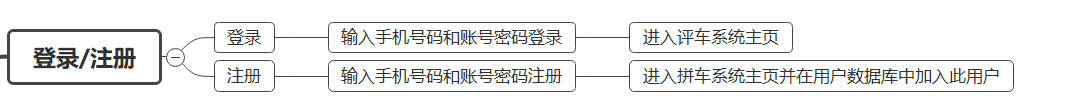
1. 响应时间在1-2s内
2. 响应失败有提示信息
3. 软件可用、易用性高

# 3.总体设计

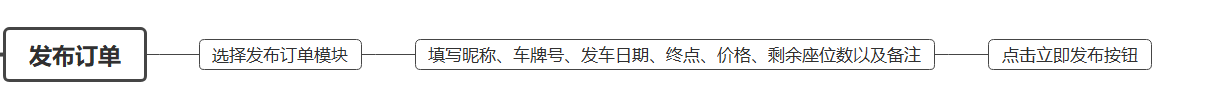
## 3.1基本设计概念和处理流程

描述每个功能模块的定义及其处理流程。

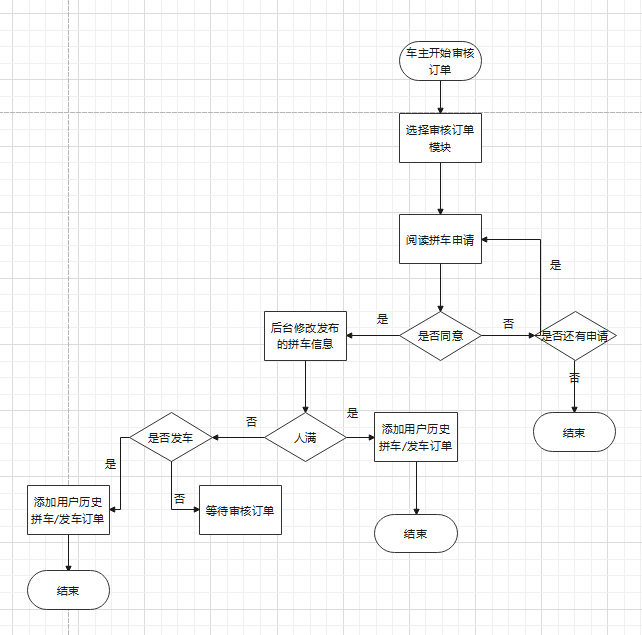
1. 登录注册模块
2. 定义：用户执行登录，初次使用执行注册
3. 处理流程：



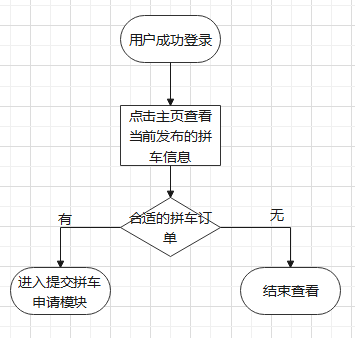
1. 发布拼车订单模块
2. 定义：车主填写信息发布拼车订单
3. 处理流程



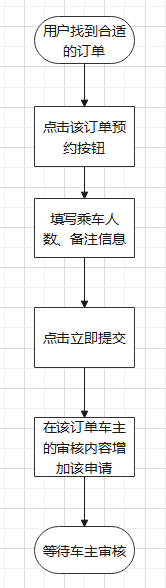
1. 审核拼车申请模块
2. 定义：车主审核用户发出的拼车申请
3. 处理流程



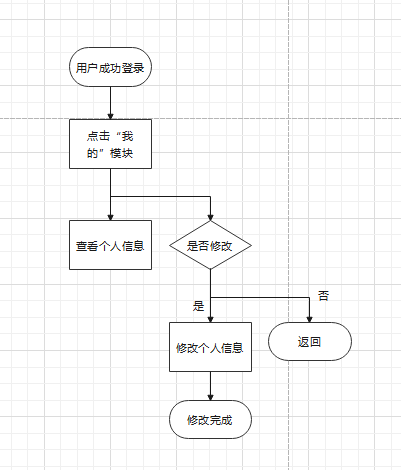
1. 查看现有拼车订单
2. 定义：用户查看当前发布的可拼车信息
3. 处理流程



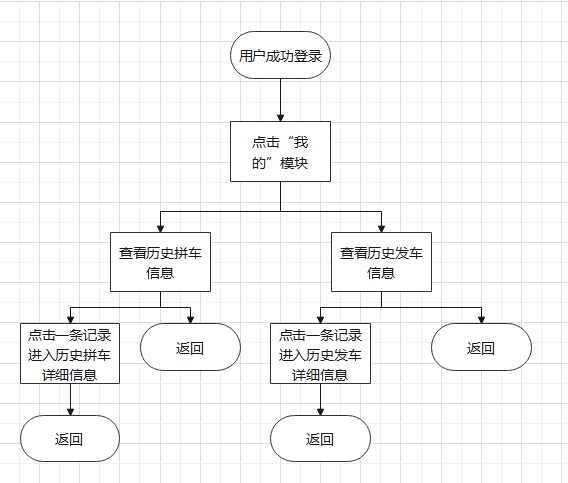
1. 提交拼车申请
2. 定义：用户对于选定的拼车订单提交拼车申请
3. 处理流程



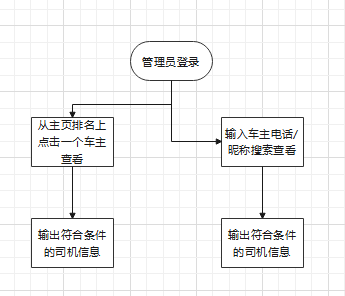
1. 查看/修改我的个人信息
2. 定义：查看/修改用户个人信息
3. 处理流程



1. 查看我的历史订单（发车/乘车）
2. 定义：查看用户历史发车/乘车信息
3. 处理流程



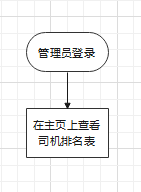
1. 查看拼车记录（管理员）
2. 定义：管理员查看所有车主的拼车记录，可以选择搜索某位车主单独查看
3. 处理流程



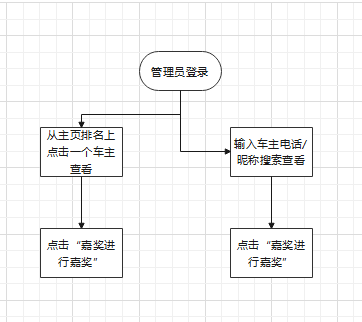
1. 查看司机排名表

(1)定义:查看司机订单成功率排名

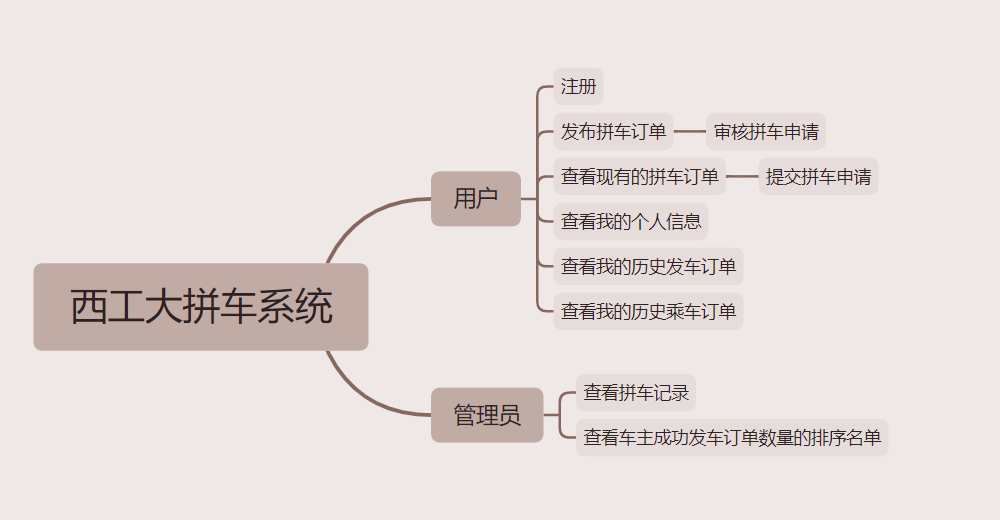
(2)处理流程:



1. 嘉奖司机
2. 定义：通过管理员人工操作，嘉奖司机一定积分
3. 处理流程:



## 3.2系统总体功能结构（必填）



先后关系和并列关系

## 3.3功能需求描述（必填）

### 3.3.1.**推荐行程信息**

【功能概述】

根据司机积分排序，在用户主页推荐当日可发车的拼车订单

【操作者】

用户

【输入】

无

【输出】

符合时间限制的车主发车订单信息

### 3.3.2.**乘客预约拼车订单**

【功能概述】

乘客对符合需求的拼车订单发起预约

【操作者】

用户—乘客

【输入】

点击主页上发车订单旁的预约按钮

【输出】

1.跳转功能3.3.3

2.输出地图信息

### 3.3.3.**乘客填写拼车申请**

【功能概述】

乘客填写拼车的申请

【操作者】

用户—乘客

【输入】

乘客拼车申请

限制：

选择乘车人数（可选人数限制小于该订单剩余座位数）

【输出】

无

### 3.3.4.**司机发布拼车订单**

【功能概述】

【操作者】

用户—车主

【输入】

1.点击“发布”

2.车主发车订单

限制：

（1）发车日期限制选择当日以后；

（2）终点限制选择其中之一：西北工业大学友谊校区、韦曲南二号线、西 安国际 医学中心六号线；

（3）剩余座位数限制选择：[1,5]；

【输出】

无

### 3.3.5.**司机审核拼车申请**

【功能概述】

司机审核顾客的订单申请，根据备注决定是否同意

【操作者】

用户—车主

【输入】

1.点击“审核”

2.点击订单申请下方的同意/拒绝按钮

【输出】

同意：无

拒绝：修改车主发车订单中的剩余座位数，若为0，订单结束

### 3.3.6.**查看我的历史发车信息**

【功能概述】

查看我的历史发车信息

【操作者】

用户—车主

【输入】

点击“我的”

【输出】

我的历史发车信息

### 3.3.7.**查看我的历史发车详细信息**

【功能概述】

查看我的历史发车详细信息

【操作者】

用户—车主

【输入】

点击“我的”

【输出】

我的历史发车详细信息

### 3.3.8.**查看我的历史拼车信息**

【功能概述】

查看我的历史拼车信息

【操作者】

用户—乘客

【输入】

无

【输出】

我的历史拼车信息

### 3.3.9.**查看我的历史拼车详细信息**

【功能概述】

查看我的历史拼车信息

【操作者】

用户—乘客

【输入】

无

【输出】

我的历史拼车信息

### 3.3.10.**查看我的个人信息**

【功能概述】

查看我的个人信息

【操作者】

用户

【输入】

点击“我的”

【输出】

我的个人信息

### 3.3.11.**修改我的个人信息**

【功能概述】

修改我的个人信息

【操作者】

用户

【输入】

1. 输入个人信息
2. 点击“更新”按钮

【输出】

个人信息更新

### 5.3.12.查看所有司机信息

【功能概述】

查看所有的司机信息

【操作者】管理员

【输入】

无

【输出】

根据司机信息中的订单成功率由高到低排名后显示所有司机信息

5.3.13.**查询指定司机信息**

【功能概述】

查询指定的司机信息

【操作者】

管理员

【输入】

司机电话或昵称

【输出】

符合输入条件的司机信息

5.3.14.**查看司机排名表**

【功能概述】

查看司机订单成功率排名

【操作者】

管理员

【输入】

无

【输出】

根据司机信息中的订单成功率由高到低排名后显示所有司机信息

5.3.15.**嘉奖司机**

【功能概述】

通过管理员人工操作，嘉奖司机一定积分

【操作者】

管理员

【输入】

单击司机信息下方的 Prize The Driver 按钮嘉奖积分

【输出】

无

## 3.4界面设计（必填）

本系统界面要求如下：

1. 界面中的文字标识，含义准确、清楚、用通用词汇；
2. 界面设计简洁、清晰。

（3）对于数值类型数据输入部分，采用可编辑的输入框，并对输入的内容进行校验，避免输入非数值数据。

（4）数据量大时候，采用分页浏览的形式。

（5）风格统一

# 4.接口设计

## 4.1外部接口

在提交拼车申请模块中运用到了高德地图接口，用于提供路线及地图

1. 接口描述

使用者：用户

业务目的：查询该拼车订单的行驶路线及停靠点

使用时机：查看拼车订单时

使用频率：不定

接口来源：高德地图

1. 接口内容

百度地图提供的始发地——目的地的驾车路线

百度地图导航路线信息

1. 设计约束

使用者可以通过Internet访问接口。

## 4.2内部接口（必填）

详见3.3

# 5.数据结构设计

## 5.1逻辑结构设计

1.用户

|  |  |  |
| --- | --- | --- |
| 名称 | 标识符 | 数据类型 |
| 用户名 | username | varchar(32) |
| 密码 | password | varchar(60) |
| 电话号码 | mobile | varchar(11) |
| 性别 | sex | varchar(1) |
| 角色判定 | role | tinyint  default 0 |

2.拼车信息

|  |  |  |
| --- | --- | --- |
| 名称 | 标识符 | 数据类型 |
| Id | owner\_id | bigint |
| 名称 | owner\_name | varchar(32) |
| 日期 | days | datetime |
| 终点 | ending | varchar(32) |
| 车牌号 | car\_num | varchar(10) |
| 价格 | price | float |
| 剩余人数 | remain | tinyint |
| 备注 | remark | varchar(64） |

3.订单表

|  |  |  |
| --- | --- | --- |
| 名称 | 标识符 | 数据类型 |
| 订单Id | order\_id | bigint |
| 乘车人数 | passenger\_num | tinyint |
| 终点 | ending | varchar(32) |
| Id | info\_id | bigint |
| 开始时间 | start\_time | datetime |
| 创建时间 | create\_time | datetime |
| 更新时间 | update\_time | datetime |
| 价格 | price | float |
| 备注 | remark | varchar(32) |
|  | order\_state | tinyint |

# 6.运行设计

## 6.1运行控制

1. 用户查看当前的订单需要访问到未结束的车主已发布订单列表。
2. 用户提交申请被通过后需要修改改订单信息/结束该订单。
3. 结束订单后用户无法看到该订单订单。
4. 订单结束后更新个人信息中的历史订单（发车和拼车）。
5. 订单结束后管理员界面更新该车主订单数。
6. 管理员嘉奖车主后排名更新。
7. 管理员嘉奖车主后用户主页已发布订单列表更新。

## 6.2运行时间

1.一般用户模块会经常运行，占用数据库使用时间的3/4  
2.管理员模块使用的最少，占用数据库使用时间的1/4。

**7.出错处理设计**

## 7.1出错输出信息

|  |  |
| --- | --- |
| 错误类型 | 错误提示 |
| 输入信息错误 | 输入信息错误，请重新登陆 |
| 没有选择条件，就进行查询或者修改 | 请选择所要查询或修改的信息 |
| 程序运行错误 | 程序运行错误，将自动关闭 |

## 7.2出错补救措施

说明故障出现后可能采取的变通措施，包括：

a.后备技术说明准备采用的后备技术，当原始系统数据万一丢失时启用的副本的建立和启动的技术，例如周期性地把磁盘信息记录到磁带上去就是对于磁盘媒体的一种后备技术；

b.降效技术说明准备采用的后备技术，使用另一个效率稍低的系统或方法来求得所需结果的某些部分，例如一个自动系统的降效技术可以是手工操作和数据的人工记录；

c.恢复及再启动技术说明将使用的恢复再启动技术，使软件从故障点恢复执行或使软件从头开始重新运行的方法。

**7.3系统恢复设计**

系统较小，没有涉及过多的模块，所以维护过程中只需要对数据库进行维护就行。

**8.安全保密设计**

防止非授权用户登录；

防止非法数据侵入；

密码存储及乱码传输；

数据私密。

**9.维护设计**

系统较小，没有涉及过多的模块，所以维护过程中只需要对数据库进行维护就行。