

# 《风》

## 数据库设计说明书

所属学院：至诚学院

团队名称：风雨五组

指导教师：傅明建

项目时间：2021-2022 学年

# 目 录

版本信息 .....	- 2 -
第一章 引言 .....	- 3 -
1.1 编写目的 .....	- 3 -
1.2 背景 .....	- 3 -
1.3 参考资料 .....	- 3 -
第二章 外部设计 .....	- 4 -
2.1 标识符和状态 .....	- 4 -
2.2 命名约定 .....	- 4 -
2.2.1 表命名约定 .....	- 4 -
2.2.2 字段命名规范 .....	- 4 -
2.3 设计约定 .....	- 4 -
第三章 结构设计 .....	- 5 -
3.1 概念结构设计 .....	- 5 -
3.1.1 实体和属性的定义 .....	- 5 -
3.1.2 局部 E-R 图 .....	- 6 -
3.1.3 全局 E-R 图 .....	- 7 -
3.2 逻辑结构设计 .....	- 8 -
(1) 用户关系模式 .....	- 8 -
(2) 管理员关系模式 .....	- 8 -
(3) 订单关系模式 .....	- 8 -
(4) 反馈关系模式 .....	- 8 -
(5) 帮助关系模式 .....	- 8 -
3.3 物理结构设计 .....	- 8 -
第四章 运用设计 .....	- 11 -
4.1 安全保密设计 .....	- 11 -
4.2 数据库设计 .....	- 11 -
4.2.1 创建表 .....	- 11 -
第五章 数据库验收标准 .....	- 12 -
5.1 数据库验收步骤 .....	- 12 -

## 版本信息

版本/状态	修订人	修改日期	备注
第 1 版	风雨五组	2022/4/20	初版

# 第一章 引言

## 1.1 编写目的

本数据库设计说明书是关于跑腿代拿 APP 的数据库设计，主要包括系统需求分析、概念结构设计、逻辑结构设计、物理结构设计、数据库设计、安全保密设计等。

本数据库设计说明书的预期读者为：用户，开发人员，测试人员

本数据库设计说明书是根据系统需求分析设计所编写的。

## 1.2 背景

随着电商以及网络的快速发展，快递已经与我们的日常生活密不可分，网络信息时代，人们的日常生活跟网络息息相关，尤其是当代大学生，很早就开始用手机，习惯于网购的生活，取快递似乎已成为我们日常生活中难以割舍的一部分。

大学生活中，我们往往会因为各种原因或者碰到各种各样的突发状况而没空去领取我们的快递，比如天气不好，驿站太远等等。身为大学生的我们在日常生活中也经常有这种不想去拿快递的烦恼。不仅仅是快递，日常学习生活任务繁忙，但是总不可避免的可能需要去什么地方取一下东西，从用户需求上来看，跑腿代拿平台十分具有必要性和可行性。

## 1.3 参考资料

- [1] 《校易数据库设计说明书》
- [2] 《风-需求设计说明书》
- [3] 《数据库表结构设计的几条标准》

## 第二章 外部设计

### 2.1 标识符和状态

数据库软件的名称：SQL Server

数据库的名称：WIND\_SQL

### 2.2 命名约定

#### 2.2.1 表命名约定

(1) 字母全部大写原则

所有数据库对象命名字母全部大写。

(2) 字符范围原则

只能使用英文字母、下划线、数字进行命名，首位字符必须是英文字母。

(3) 分段命名原则

命名中多个单词间采用下划线分隔，以便阅读同时方便某些工具对数据库对象的映射。如 XXX\_XXX\_XXX，但不限于三段式。

(4) 简单命名原则

命名尽可能简单，避免太长的命名，尽量使用缩写形式，但是缩写也要能够表达命名的含义。

#### 2.2.2 字段命名规范

(1) 必须有一主键，主键不直接用 ID，而是表名+ID

(2) 常用的字段 name，不直接用 name，而是表名+Name

(3) 避免中文拼音

(4) 避免名称过长

(5) 避免保留字

### 2.3 设计约定

(1) 表必须要有主键。

(2) 一个字段只表示一个含义。

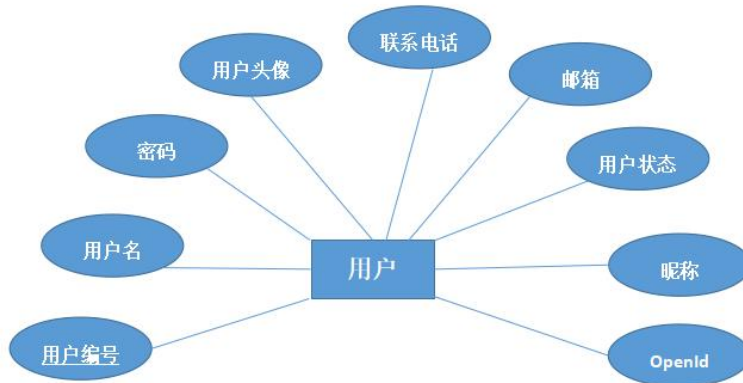
(3) 禁止使用复杂数据类型（数组，自定义类型等）。

## 第三章 结构设计

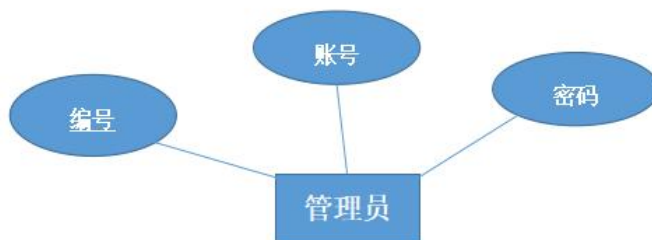
### 3.1 概念结构设计

#### 3.1.1 实体和属性的定义

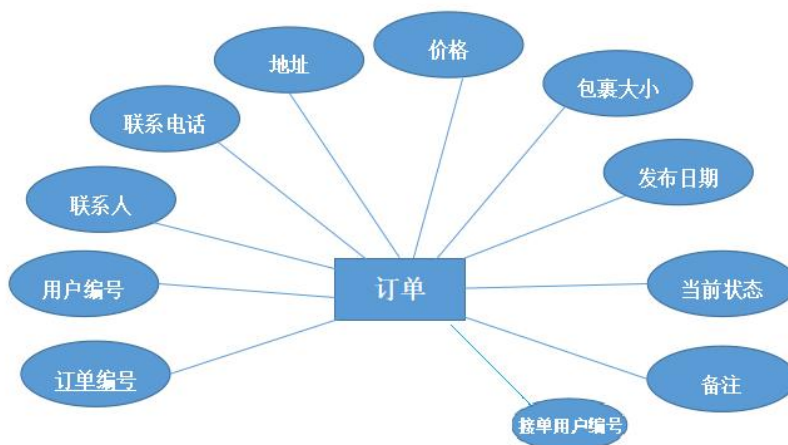
##### (1) 用户关系模块



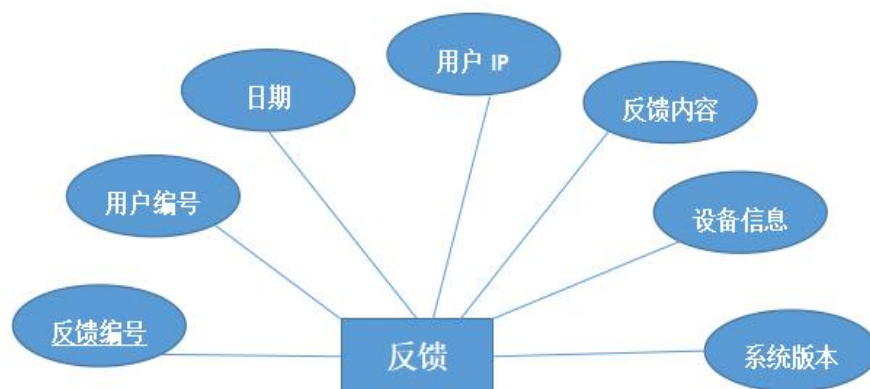
##### (2) 管理员关系模块



##### (3) 订单关系模块



#### (4) 反馈关系模块

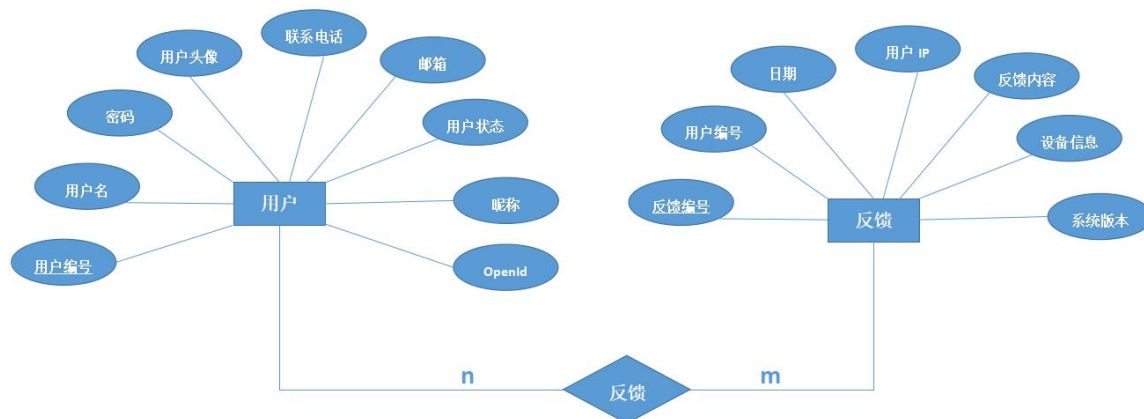


#### (5) 帮助关系模块

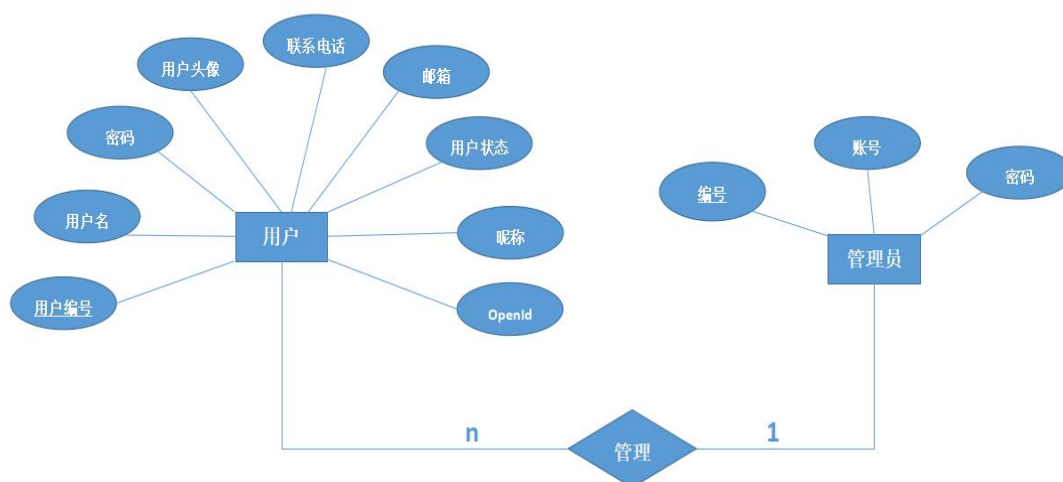


### 3.1.2 局部 E-R 图

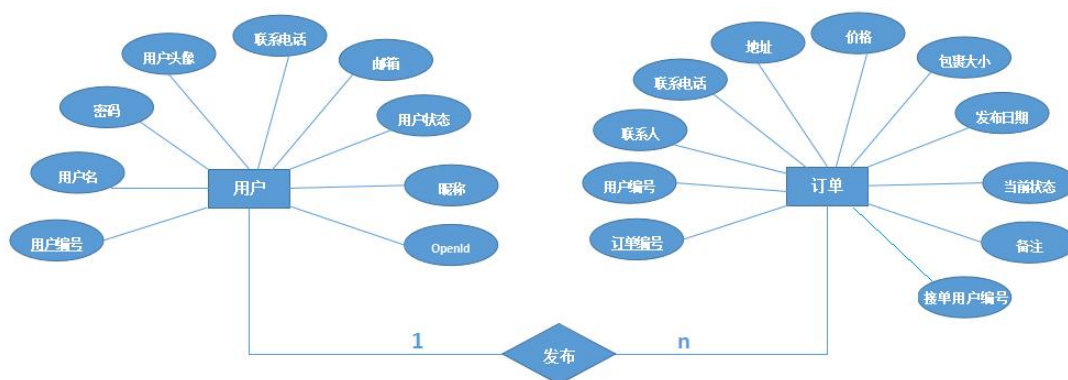
#### (1) 用户与反馈关系



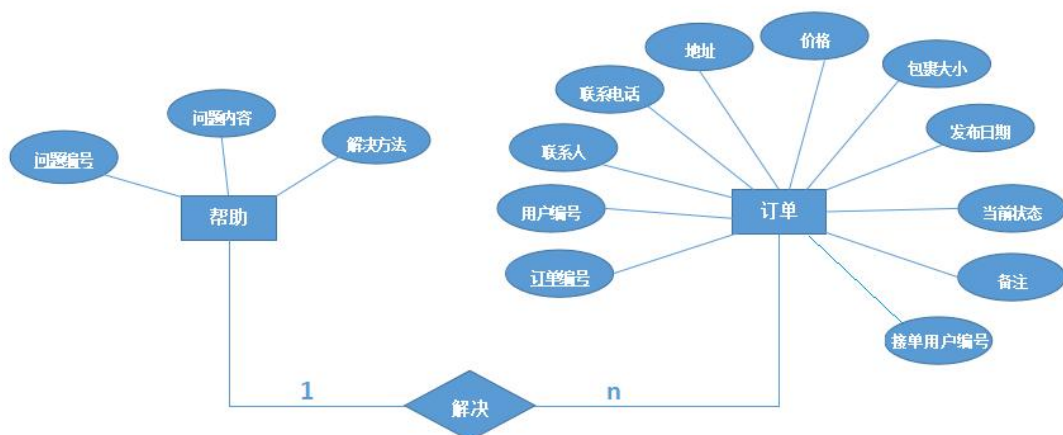
## (2) 用户与管理员关系



## (3) 用户与订单关系

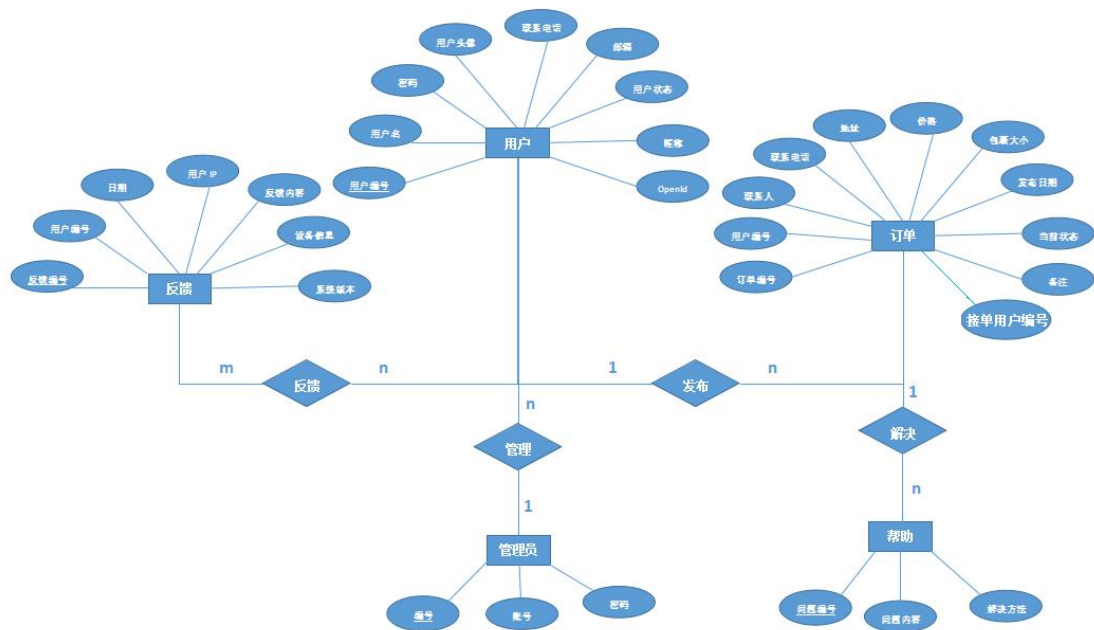


## (4) 订单与帮助关系



### 3.1.3 全局 E-R 图





### 3.2 逻辑结构设计

#### (1) 用户关系模式

用户关系模式（用户编号，用户名，昵称，密码，用户头像，联系电话，邮箱，用户状态，OpenId）；

#### (2) 管理员关系模式

管理员关系模式（管理员编号，管理员账号，密码）；

#### (3) 订单关系模式

订单关系模式（订单编号，用户编号，接单用户编号，联系人，联系电话，地址，价格，包裹大小，发布日期，备注，当前状态）；

#### (4) 反馈关系模式

反馈关系模式（反馈编号，用户编号，日期，用户 IP，反馈内容，设备信息，系统版本）；

#### (5) 帮助关系模式

帮助关系模式（问题编号，问题内容，解决方法）；

### 3.3 物理结构设计

表 1 用户表

字段标识	字段名	数据类型	说明
------	-----	------	----

用户编号	u_id	int	PK, NOT NULL
用户名	u_name	varchar(64)	NOT NULL
密码	u_pwd	varchar(64)	NOT NULL
昵称	u_nickname	varchar(64)	
用户头像	u_img	varchar(50)	
联系电话	u_phone	varchar(20)	
邮箱	u_email	varchar(64)	
用户状态	u_state	varchar(64)	
OpenId	u_openid	varchar(64)	

表 2 管理员表

字段标识	字段名	数据类型	说明
管理员编号	m_id	int	PK, NOT NULL
管理员账号	m_account	varchar(64)	NOT NULL
密码	m_pwd	varchar(64)	NOT NULL

表 3 订单表

字段标识	字段名	数据类型	说明
订单编号	o_id	int	PK, NOT NULL
用户编号	u_id	int	FK, NOT NULL
接受用户编号	a_id	int	NULL
联系人	o_contacts	varchar(64)	NOT NULL
联系电话	o_phone	varchar(20)	NOT NULL
地址	o_address	varchar(64)	NOT NULL
价格	o_price	numeric(4, 2)	NOT NULL
包裹大小	o_package	varchar(64)	NOT NULL
发布日期	o_date	date	NOT NULL
当前状态	o_state	int	NOT NULL
备注	o_remark	varchar(225)	

表 4 反馈表

字段标识	字段名	数据类型	说明
------	-----	------	----

反馈编号	f_id	int	PK, NOT NULL
用户编号	u_id	int	FK, NOT NULL
日期	f_date	date	NOT NULL
用户 IP	f_ip	varchar(50)	
反馈内容	f_content	varchar(64)	NOT NULL
设备信息	f_info	varchar(64)	
系统版本	f_version	varchar(64)	

表 5 帮助表

字段标识	字段名	数据类型	说明
问题编号	h_id	int	PK, NOT NULL
问题内容	h_content	varchar(64)	NOT NULL
解决方法	h_method	varchar(225)	NOT NULL

## 第四章 运用设计

### 4.1 安全保密设计

数据库基本安全架构, 用户分类不同类型的用户授予不同的数据管理权限, 分为两类, 一类是数据库管理员权限类, 还有时普通用户群体类。数据分类, 同一类权限的用户对数据库中数据管理和使用的范围又可能是不同的。

### 4.2 数据库设计

#### 4.2.1 创建表

```
/*创建用户表*/
create table WUSER(
    u_id      int PRIMARY KEY NOT NULL,
    u_name    varchar(64) NOT NULL,
    u_pwd     varchar(64) NOT NULL,
    u_nickname varchar(64),
    u_img     varchar(50),
    u_phone   varchar(20),
    u_email   varchar(64),
    u_state   varchar(64),
    u_openid  varchar(64)
)

/*创建管理员表*/
create table WMANAGER(
    m_id      int PRIMARY KEY NOT NULL,
    m_account  varchar(64) NOT NULL,
    m_name     varchar(64) NOT NULL
)

/*创建订单表*/
create table WORDER(
    o_id      int PRIMARY KEY NOT NULL,
    u_id      int NOT NULL,
    o_contact varchar(64) NOT NULL,
    o_phone   varchar(20) NOT NULL,
    o_address  varchar(64) NOT NULL,
    o_price   numeric(4,2) NOT NULL,
    o_package varchar(64) NOT NULL,
    o_date    date NOT NULL,
    o_state   int NOT NULL,
    o_remark  varchar(225)
)

/*建立订单表与用户表的外键*/
ALTER TABLE WORDER ADD CONSTRAINT fk_WUSER_u_id FOREIGN KEY(u_id) REFERENCES WUSER(u_id);

/*创建反馈表*/
create table WFEEDBACK(
    f_id      int PRIMARY KEY NOT NULL,
    u_id      int NOT NULL,
    f_date    date NOT NULL,
    f_ip      varchar(50),
    f_content  varchar(64) NOT NULL,
    f_info     varchar(64),
    f_version  varchar(64)
)

/*建立反馈表与用户表的外键*/
ALTER TABLE WFEEDBACK ADD CONSTRAINT fk_WFEEDBACK_u_id FOREIGN KEY(u_id) REFERENCES WUSER(u_id);

/*创建帮助表*/
create table WHELP(
    h_id      int PRIMARY KEY NOT NULL,
    h_content  varchar(64) NOT NULL,
    h_method   varchar(225) NOT NULL
)
```

## 第五章 数据库验收标准

### 5.1 数据库验收步骤

