

水滴打卡数据库设计说明书

学院：福州大学至诚学院

团队名称：一只水队

指导老师：张栋

目 录

第一章	引言
1.1	编写目的
1.2	背景
1.3	定义
1.4	参考资料
第二章	外部设计
2.1	标识符和状态
2.2	使用它的程序
2.3	支持软件
第三章	结构设计
3.1	概念结构设计
3.2	逻辑结构设计
3.3	物理结构设计
第四章	运用设计
4.1	数据字典设计
4.2	安全保密设计
第五章	数据库验证验收标准
5.1	数据库数据体的验收
5.2	数据库安全性的验收

1. 引言

1.1 编写目的

在现在的日常生活中，许多人都喜欢用“打卡”的方式来记录自己的日常生活，或是“打卡”一处景点，一家家餐厅，亦或是每日健身、学习等长时间坚持的“打卡”。前者更多的是记录生活，便于确定下一个目标。而后者则主要是让自己更加坚定信念，养成一个良好的习惯，直观的了解到自己的时间利用，随着打卡天数和时长的不断增加可以不断的激励自我，将所做的事能够长期坚持下去；同时能够与志同道合者相互竞争，增强自信，激发潜力。此外在之前的问答环节中，图书馆的老师也提到了对打卡小程序的期待，并希望能和图书馆的读书打卡活动相结合。由此我们认为，我们学院的图书馆有一个自己的“打卡”小程序是有必要也有益处的。

1.2 背景

- a. 待开发软件系统的名称：水滴打卡
- b. 本项目的任务提出者：福州大学至诚学院
- c. 本项目开发者：一只水队
- d. 本项目用户：本校内学生

1.3 定义

PK：主键约束

FK：外键约束

Not Null:非空约束

1.4 参考资料

- 1. 数据库表结构设计的几条准则

<https://www.cnblogs.com/wyq178/p/8549715.html>

- 2. 数据库表设计（一对多、多对多）

<https://blog.csdn.net/fighteryang/article/details/82848505>

- 3. 数据库验收规定

<https://wenku.baidu.com/view/1759be18c281e53a5802ff0b.html>

2. 外部设计

2.1 标识符的状态

数据表名称	标识符或名称	描述	状态
用户表	用户的唯一 ID	保存用户信息	使用
管理员表	管理员的唯一 ID	保存管理员信息	使用
打卡时长表	NULL	保存学生打卡时长	使用
打卡次数表	NULL	保存学生打卡的次数	使用
学生排行表	NULL	保存学生的排行榜信息	使用

2.2 使用它的程序

水滴打卡

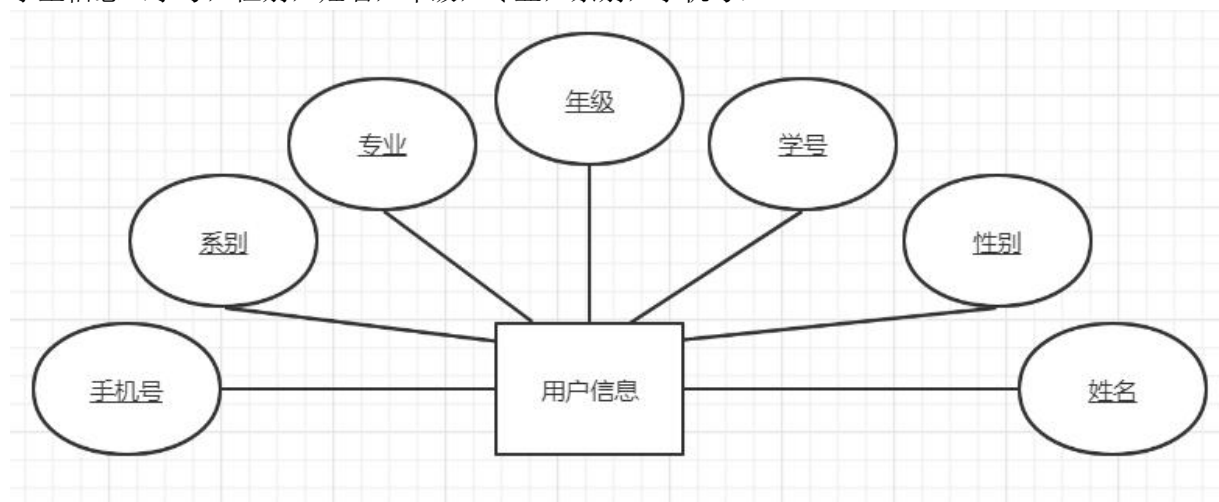
2.3 支持软件

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 旗下产品。MySQL 是一种关系数据库管理系统，关系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。

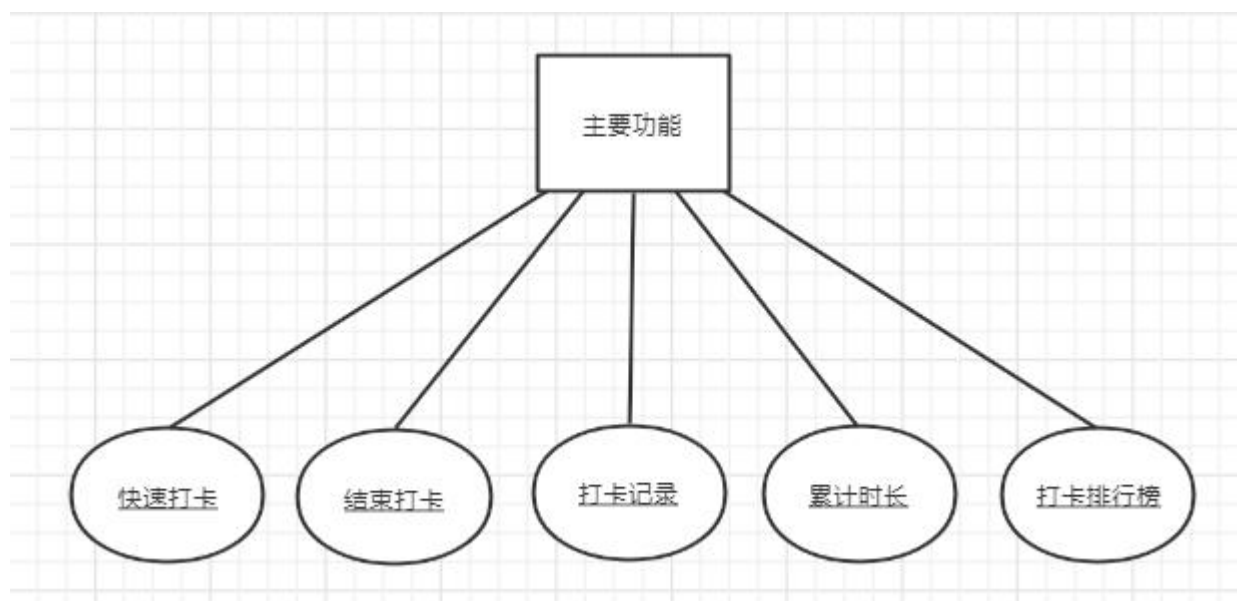
3. 结构设计

3.1 概念结构设计

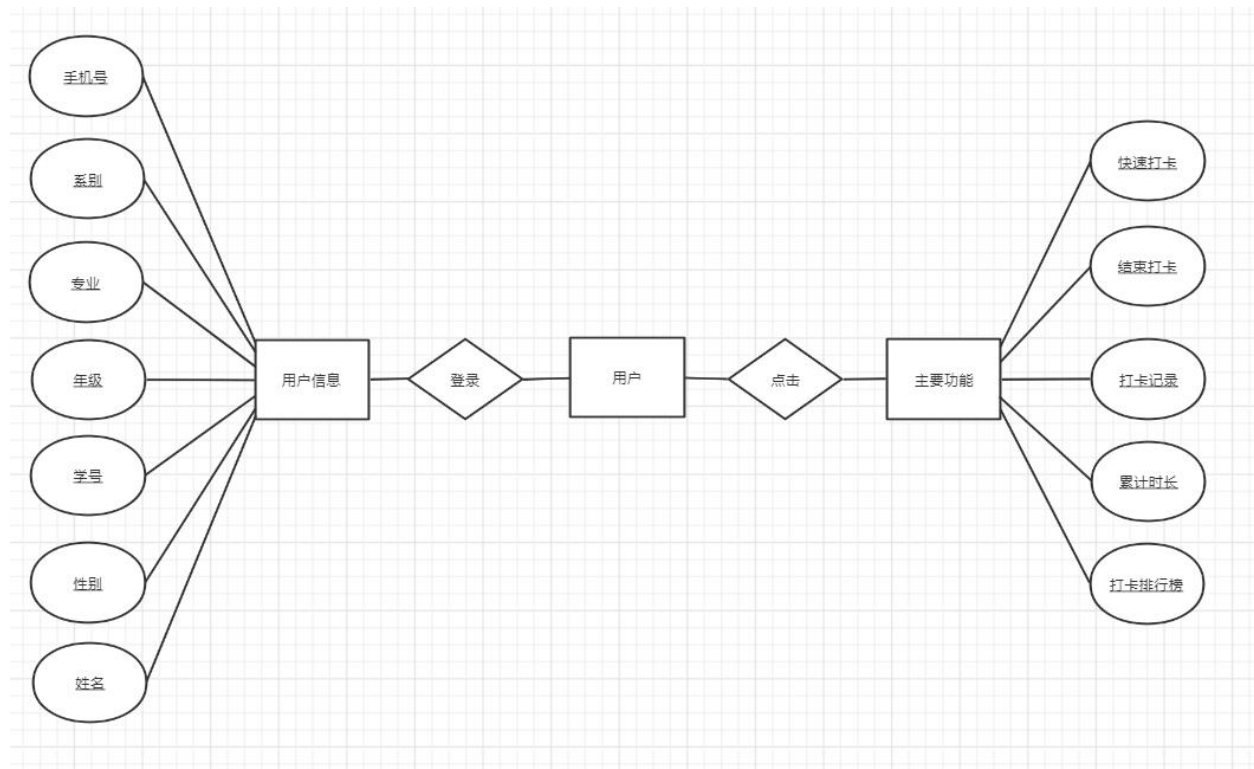
学生信息（学号，性别，姓名，年级，专业，系别，手机号）



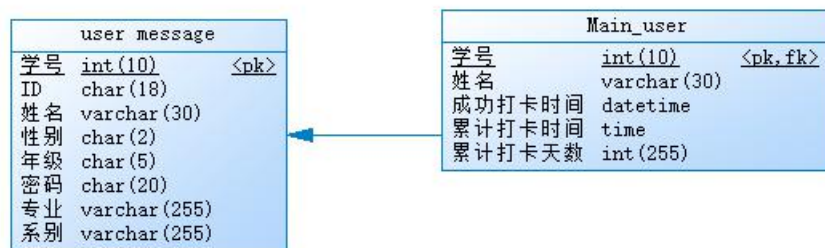
功能（打卡成功时间，累计时长）



完整 ER 图



3.2 逻辑结构设计



3.3 物理结构设计

[illegible][illegible]

4. 运用设计

4.1 运用字典设计

学生用户表结构

字段名	数据类型	长度	主键	非空	描述
学号	INT	10	PK	NOT NULL	学生学号
名字	VARCHAR	30		NOT NULL	学生姓名
性别	CHAR	2		NOT NULL	学生性别
年级	CHAR	5		NOT NULL	学生年级
系别	VARCHAR	255		NOT NULL	学生系别
专业	VARCHAR	255		NOT NULL	学生专业
手机号	INT	20		NOT NULL	学生手机号
密码	CHAR	20		NOT NULL	登录密码

功能表

字段名	数据类型	长度	主键	非空	描述
学号	INT	10	PK	NOT NULL	学生学号
名字	VARCHAR	30		NOT NULL	学生名字
打卡时间	DATETIME	255		NOT NULL	学生打卡成功时间
打卡天数	INT	255			累计打卡天数
打卡累计时长	TIME	255			累计打卡时长

4.2 安全保密设计

通过区分不同的访问者、不同的访问类型和不同的数据对象，进行分别对待而获得的数据库安全保密设计考虑。数据库由专门数据库管理员对数据库操作。管理员权限最大，可以控制所有数据。

数据库名称为: water daka

```
drop table if exists "user message";
```

```
/*=====*/
/* Table: "user message" */
/*=====*/
create table "user message"
(
    Sno                int(10) not null,
    ID                 char(18) not null,
    name               varchar(30) not null,
    sex                char(2) not null,
    Grade              char(5) not null,
    password            char(20) not null,
    major              varchar(255) not null,
    Department          varchar(255) not null,
    primary key (Sno)
);
```

```
drop table if exists Main_user;
```

```
/*=====*/
/* Table: Main_user */
/*=====*/
create table Main_user
(
    Sno                int(10) not null,
    name               varchar(30) not null,
    "Suc time"         datetime not null,
    "Acu time"         time,
    Count              int(255),
    primary key (Sno)
);
```

```
alter table Main_user add constraint FK_打卡记录 foreign key (Sno)
references "user message" (Sno) on delete restrict on update restrict;
```

5.数据库验证验收标准

5.1 数据库数据体的验收

数据的正确性	每列的原子性符合第一范式	√
数据的正确性	有唯一的标识符	√
数据的完整性	都存储单一实体类型的数据	√

5.2 数据库的安全性验收

数据库的安全建立在操作系统的安全至上，数据库必须具有如下的安全措施：

数据库用户权限设置	进入系统是认证用户权限	√
数据库的备份与恢复	数据有备份日志	√
数据库的权限设置	通过视图机制把要保密的数据对无权存取的用户隐藏起来	√