**软件工程1**

考勤系统软件设计报告



**小组成员：田家硕、白嘉欣**

**毛文瑞、李青颖**

**班 级：软件工程 2班**

2020年6月9日

目录

[考勤系统软件设计报告 1](#_Toc42626532)

[考勤系统软件设计报告 2](#_Toc42626533)

[1. 作用范围： 2](#_Toc42626534)

[1.1 系统目标： 2](#_Toc42626535)

[1.2 主要的系统功能： 3](#_Toc42626536)

[1.3 外部数据库定义： 3](#_Toc42626537)

[1.4 主要的设计约束和限制： 3](#_Toc42626538)

[2. 文档： 4](#_Toc42626539)

[2.1 现有的文档： 4](#_Toc42626540)

[2.2 技术参考书： 4](#_Toc42626541)

[3. 总体设计： 5](#_Toc42626542)

[3.1 系统功能： 5](#_Toc42626543)

[3.2 技术架构： 6](#_Toc42626544)

[3.3 部署图： 6](#_Toc42626545)

[4. 详细设计： 6](#_Toc42626546)

[4.1 登录模块： 6](#_Toc42626547)

[5. 数据库设计： 8](#_Toc42626548)

[5.1 ER图 8](#_Toc42626549)

[5.2数据库建表 8](#_Toc42626550)

# 考勤系统软件设计报告

## 1. 作用范围：

### 1.1 系统目标：

为规范员工的出勤行为，减少各部门之间的矛盾，提高员工的工作积极性，本考勤系统记录了员工的正常出勤、迟到早退、请假、外出工作等行为。员工本人及各部门均可查看。

内置请假系统，员工可以在系统中提交申请，管理者可以在线审批，简化了请假的流程而且方便突发情况时查看所需员工是否在岗。员工可以在请假系统中查看自己的休假情况，包括可休时长、已休时常和剩余时常，并且每一个带薪假期都会标注生效时间和失效时间。使员工和管理者清晰的掌握休假情况以准确计算薪金。

### 1.2 主要的系统功能：

#### 1. 打卡记录

这一过程不需要做出改变，保持原来上班打卡的规则。所有打卡记录都会保存在系统中，供行政部统计考勤信息和管理带薪假期。方便财务部根据考勤情况调整员工的薪金。

本系统会对打卡记录每晚00：30进行计算处理，生成考勤表，表中统计了该员工的无薪假期天数、迟到、早退、缺勤天数。

#### 2. 请假申请

假期主要分为带薪假期和无薪假期两大类。

均需要提交申请，等待审批。带薪假期包括年假、婚假、产检假、产假、陪产假、哺乳假、加班补休，无薪假期包括病假、事假。

员工三天以下假期（带薪或无薪）需提交部门经理处审批，副总经理有最终审批的权限。三天以上假期需提交总经理处审批。每个人的请假情况和剩余假期数均可在系统中查看。员工拥有对请假申请的撤销权利。

审批通过后，会将员工请假的通知发送给相应的项目经理。后续如果撤回申请也会通知项目经理。

#### 3. 外出申请

外出时提交申请与审批流程与请假流程大致相似。在系统中会记录外出时间和回公司的时间。

### 1.3 外部数据库定义：

采用JDBC技术连接使用MySQL来设计数据库。

### 1.4 主要的设计约束和限制：

#### 1.4.1 数据库设计规范：

1）尽可能在数据模型上控制业务对象的约束关系

2）数据模型总的唯一性约束[比如用户ID]，一定要在数据库层面得到控制；

3）sql语句尽量不要依据业务逻辑以及动态拼接的sql字符串，而是采用预编译的方式；否则有sql注入的风险

5）如果主表与子表是一对一的关系，主键尽量相同。

6）数据库是各个业务系统的私有资源；其他系统对于该数据结构应该是透明的；只能通过接口和事件去访问和修改数据。

#### 1.4.2 外部交互设计规范：

1）异步消息处理的时候，最好先落地到本地库再进行处理；这样避免消息的丢失，以及消息队列的堆积。

## 2. 文档：

### 2.1 现有的文档：

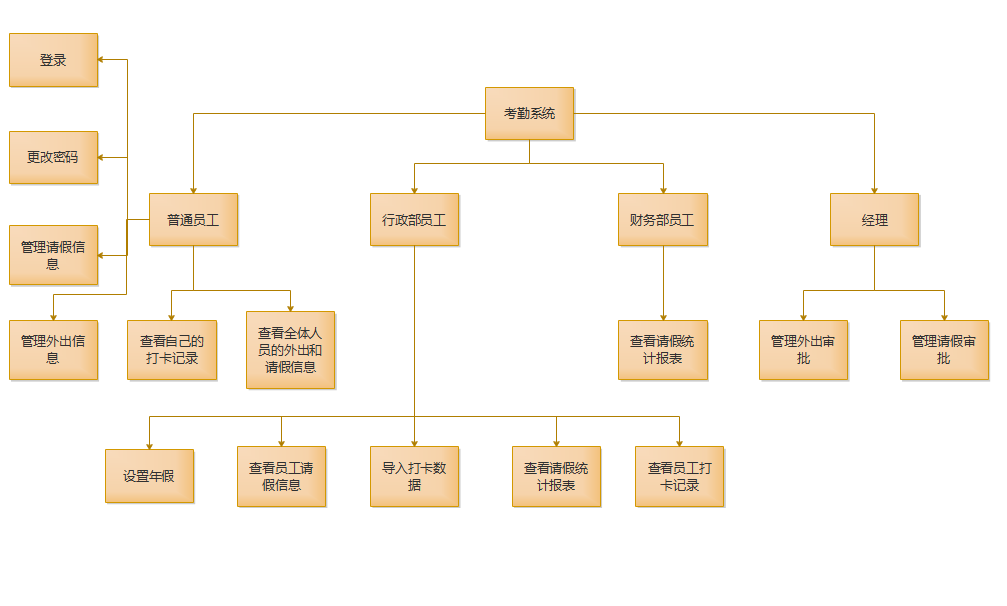
|  |  |
| --- | --- |
| 文档名称 | 日期/版本 |
| 《考勤系统需求分析》 | 2020.6.3 |

### 2.2 技术参考书：

|  |  |  |
| --- | --- | --- |
| 资料名称 | 版本/日期 | 说明 |
| Uml大战需求分析 | 2012年2月第1版 | 无 |

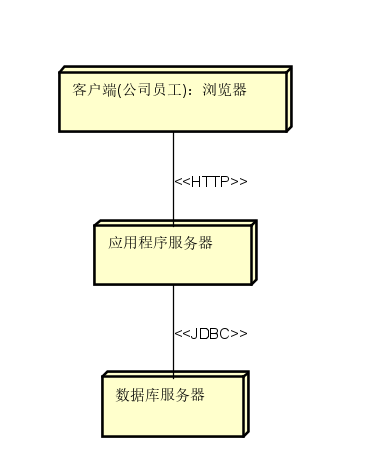
## 3. 总体设计：

### 3.1 系统功能：



### 3.2 技术架构：

### 3.3 部署图：



## 详细设计：

### 4.1 登录模块：

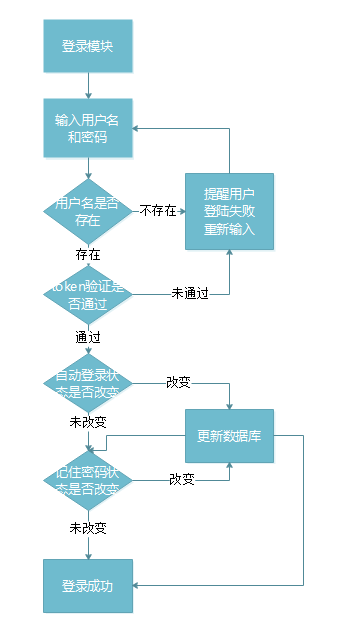
#### 4.1.1. 处理过程陈述：

用户输入用户名和密码，系统检测该用户是否存在，存在则进行token验证。若用户存在且验证成功，再继续检测自动登录按钮和记住密码的状态是否有所改变，若有则更改完后，提醒用户登录成功。若用户不存在或者token验证失败，则提醒用户登录失败。

#### 4.1.2. 接口描述：

调用接口前后端进行数据交换的时候使用HTTP协议，传输数据为json格式。

#### 4.1.3. 流程图描述：

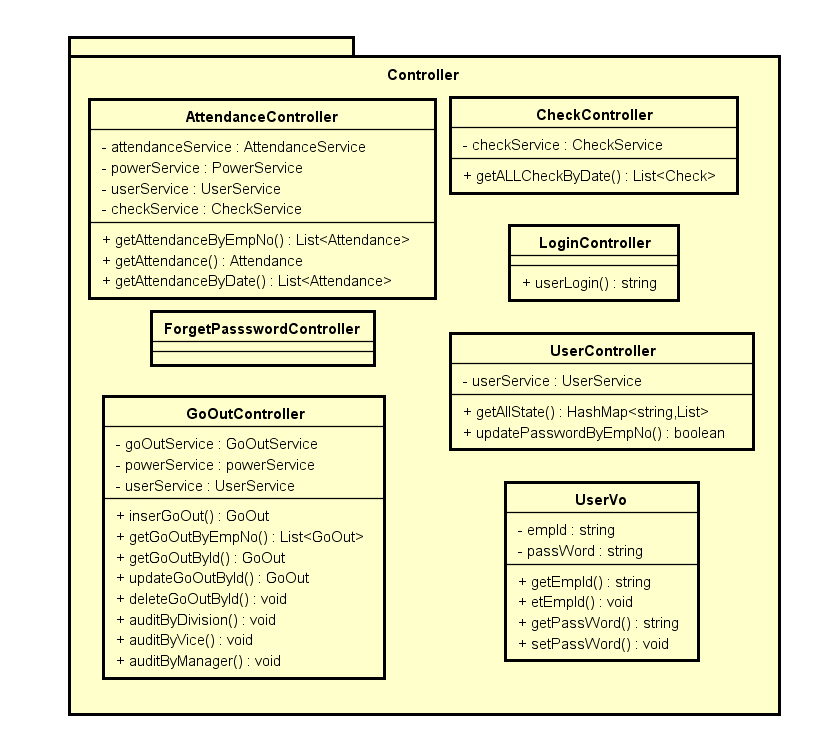


* + 1. 数据组织：

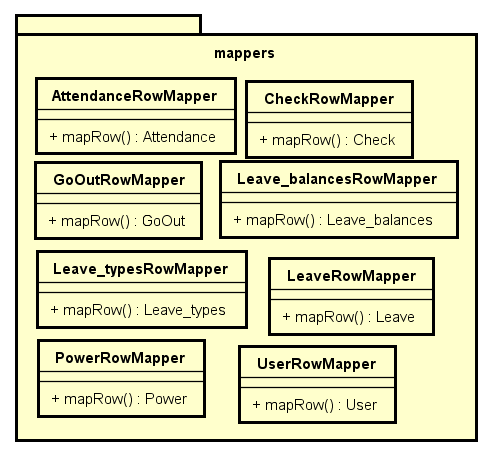
用户名和密码的信息

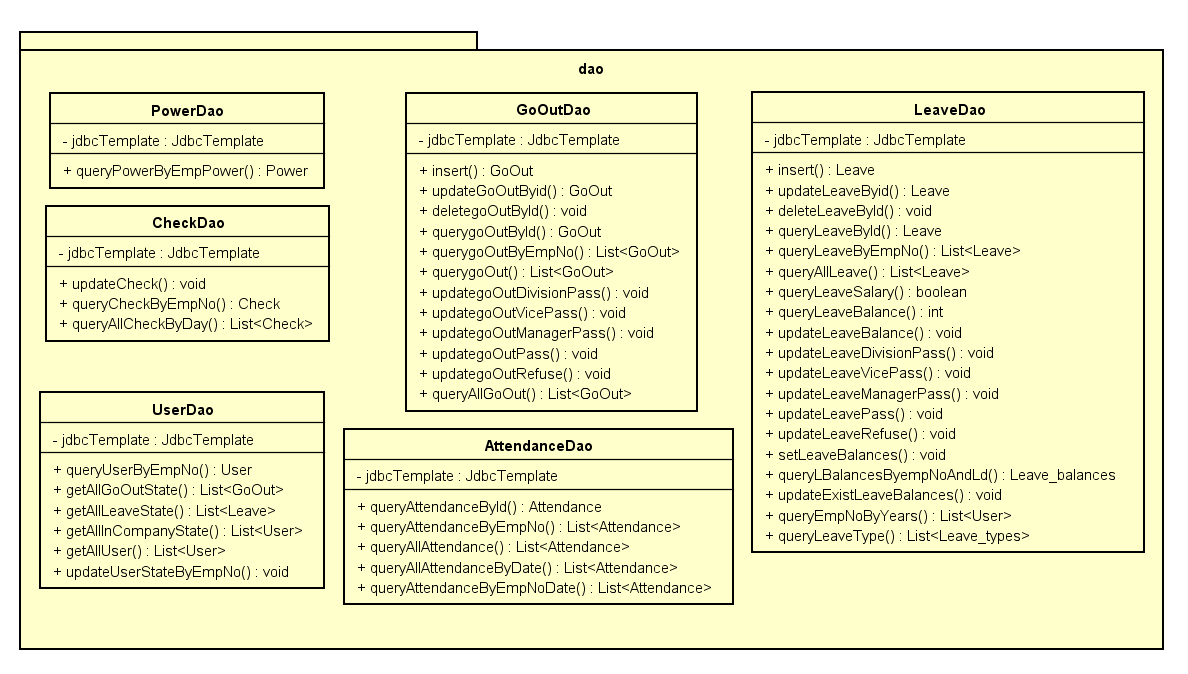
## 系统关键类抽象

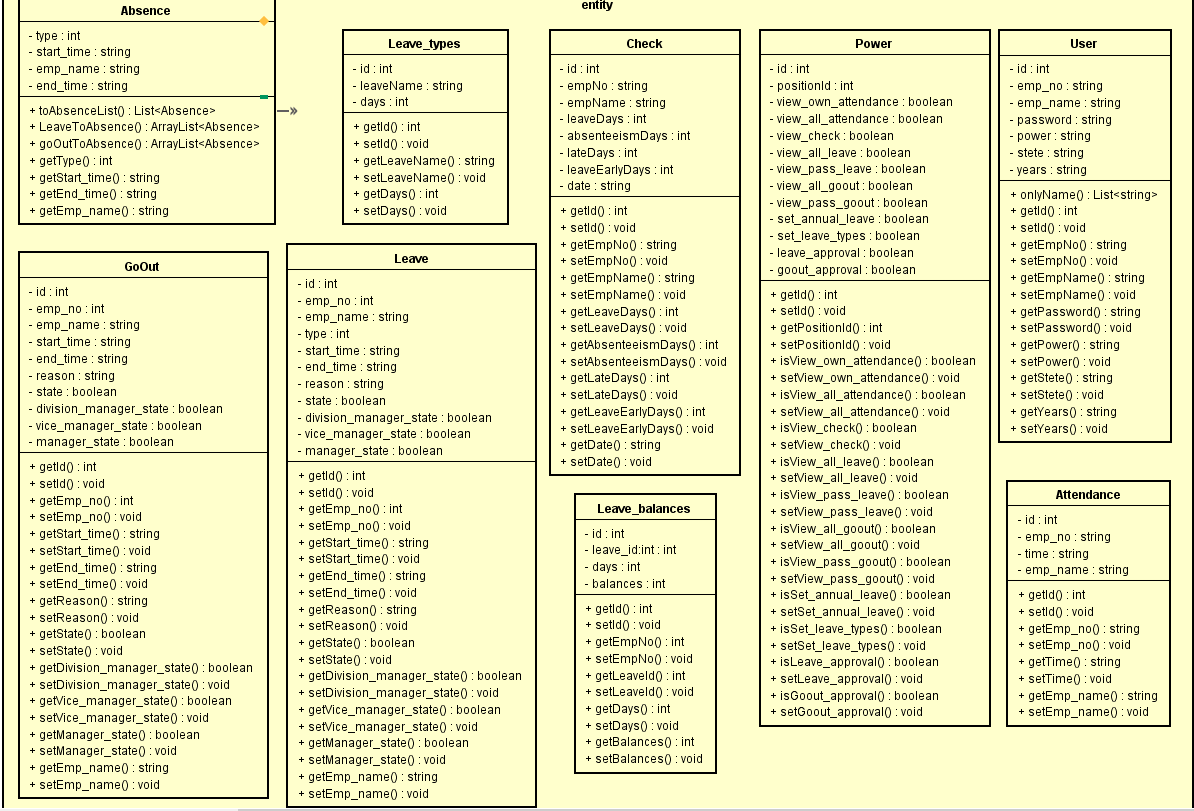
### 5.1Controller



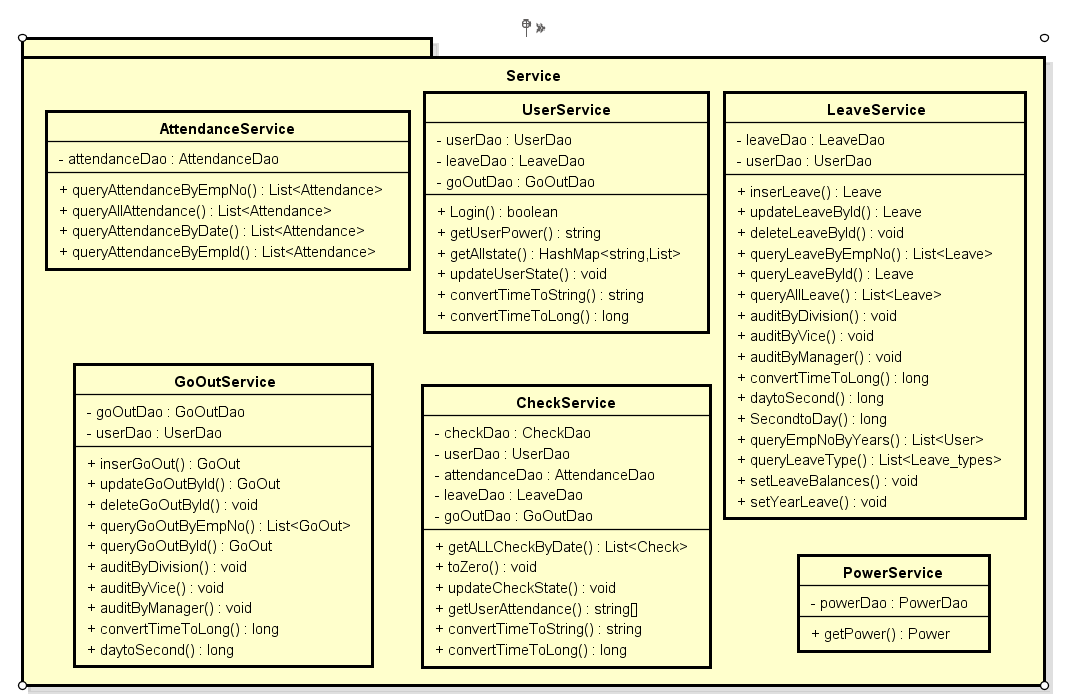
### 5.2Repository





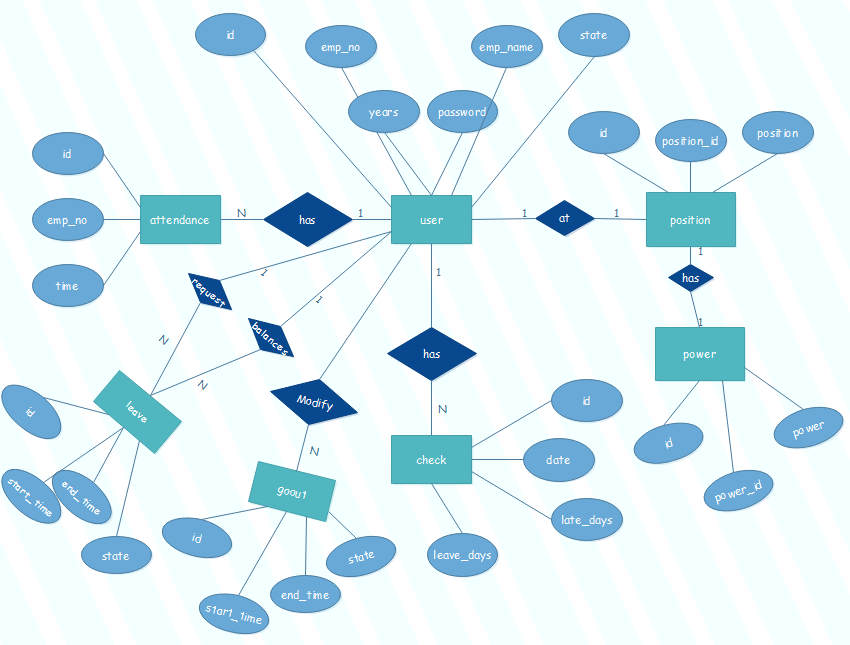


### 5.3Service



## 6. 数据库设计：

### 6.1 ER图



### 6.2数据库建表

#### User



CREATE TABLE `user` (

`id` int(11) NOT NULL AUTO\_INCREMENT,

`emp\_no` int(11) NOT NULL,

`emp\_name` varchar(255) COLLATE utf8\_bin NOT NULL,

`password` varchar(28) COLLATE utf8\_bin NOT NULL,

`power` int(11) NOT NULL,

`state` varchar(255) COLLATE utf8\_bin NOT NULL,

`years` int(11) NOT NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=5 DEFAULT CHARSET=utf8 COLLATE=utf8\_bin;

#### Power

#### 

CREATE TABLE `power` (

`id` int(11) NOT NULL AUTO\_INCREMENT,

`positionId` int(11) NOT NULL,

`view\_own\_attendance` tinyint(1) NOT NULL,

`view\_all\_attendence` tinyint(1) NOT NULL,

`view\_check` tinyint(1) NOT NULL,

`view\_all\_leave` tinyint(1) NOT NULL,

`view\_pass\_leave` tinyint(1) NOT NULL,

`view\_all\_goout` tinyint(1) NOT NULL,

`view\_pass\_goout` tinyint(1) NOT NULL,

`set\_annual\_leave` tinyint(1) NOT NULL,

`set\_leave\_types` tinyint(1) NOT NULL,

`leave\_approval` tinyint(1) NOT NULL,

`goout\_approval` tinyint(1) NOT NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=5 DEFAULT CHARSET=utf8 COLLATE=utf8\_bin;

#### Position

#### 

CREATE TABLE `position` (

`id` int(11) NOT NULL AUTO\_INCREMENT,

`position` varchar(255) COLLATE utf8\_bin NOT NULL,

`powerId` int(11) NOT NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=5 DEFAULT CHARSET=utf8 COLLATE=utf8\_bin;

#### Leave\_types

#### 

CREATE TABLE `leave\_types` (

`id` int(11) NOT NULL AUTO\_INCREMENT,

`leave\_name` varchar(255) COLLATE utf8\_bin NOT NULL,

`days` int(11) NOT NULL,

`is\_salary` tinyint(1) NOT NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=7 DEFAULT CHARSET=utf8 COLLATE=utf8\_bin ROW\_FORMAT=DYNAMIC;

#### Leave\_request

#### 

CREATE TABLE `leave\_request` (

`id` int(11) NOT NULL AUTO\_INCREMENT,

`emp\_no` int(11) NOT NULL,

`emp\_name` varchar(255) COLLATE utf8\_bin NOT NULL,

`type` int(11) NOT NULL,

`start\_time` datetime NOT NULL,

`end\_time` datetime NOT NULL,

`reason` varchar(255) COLLATE utf8\_bin NOT NULL,

`state` tinyint(1) NOT NULL,

`division\_manager\_state` tinyint(1) NOT NULL,

`vice\_manager\_state` tinyint(1) NOT NULL,

`manager\_state` tinyint(1) NOT NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=4 DEFAULT CHARSET=utf8 COLLATE=utf8\_bin;

#### Leave\_balance

CREATE TABLE `leave\_balances` (

`id` int(11) NOT NULL AUTO\_INCREMENT,

`emp\_no` int(11) NOT NULL,

`leave\_id` int(11) NOT NULL,

`days` int(11) NOT NULL,

`balances` int(11) NOT NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=3 DEFAULT CHARSET=utf8 COLLATE=utf8\_bin;

#### Goout

CREATE TABLE `goout` (

`id` int(11) NOT NULL AUTO\_INCREMENT,

`emp\_no` int(11) NOT NULL,

`start\_time` datetime NOT NULL,

`end\_time` datetime NOT NULL,

`reason` varchar(255) COLLATE utf8\_bin NOT NULL,

`state` tinyint(1) NOT NULL,

`division\_manager\_state` tinyint(1) NOT NULL,

`vice\_manager\_state` tinyint(1) NOT NULL,

`manager\_state` tinyint(1) NOT NULL,

`emp\_name` varchar(255) COLLATE utf8\_bin NOT NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=3 DEFAULT CHARSET=utf8 COLLATE=utf8\_bin;

#### Checktable

CREATE TABLE `checktable` (

`id` int(11) NOT NULL AUTO\_INCREMENT,

`emp\_no` int(11) NOT NULL,

`leave\_days` int(11) NOT NULL,

`absenteeism\_days` int(11) NOT NULL,

`late\_days` int(11) NOT NULL,

`leave\_early\_days` int(11) NOT NULL,

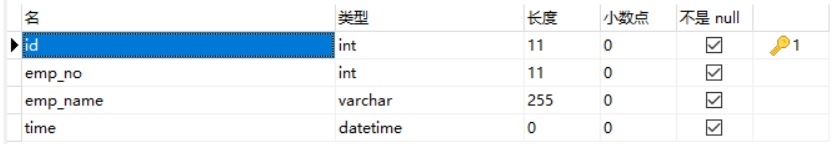
`emp\_name` varchar(255) COLLATE utf8\_bin NOT NULL,

`date` datetime NOT NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=13 DEFAULT CHARSET=utf8 COLLATE=utf8\_bin ROW\_FORMAT=DYNAMIC;

#### Attendance



CREATE TABLE `attendance` (

`id` int(11) NOT NULL AUTO\_INCREMENT,

`emp\_no` int(11) NOT NULL,

`emp\_name` varchar(255) COLLATE utf8\_bin NOT NULL,

`time` datetime NOT NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=10 DEFAULT CHARSET=utf8 COLLATE=utf8\_bin ROW\_FORMAT=DYNAMIC;