

**期末项目设计报告**

|  |  |  |  |
| --- | --- | --- | --- |
| 题 目 | 基于Oracle的某项目的数据库设计 | | |
| 课程 | Oracle数据库应用 | | |
| 学 院 | 信息科学与工程学院 | | |
| 专 业 | 软件工程 | 年级 | 2018级 |
| 学生姓名 | 覃茂辉 | 学号 | 201810414216 |
| 指导教师 | 赵卫东 | 职称 | 副教授 |

|  |  |  |  |
| --- | --- | --- | --- |
| **评分项** | **评分标准** | **满分** | **得分** |
| 文档整体 | 文档内容详实、规范，美观大方 | 10 |  |
| 表设计 | 表，表空间设计合理，数据合理 | 20 |  |
| 用户管理 | 权限及用户分配方案设计正确 | 20 |  |
| PL/SQL设计 | 存储过程和函数设计正确 | 30 |  |
| 备份方案 | 备份方案设计正确 | 20 |  |
| **得分合计** | | |  |

2021 年 6 月 1 日

# 基于Oracle的城市周边游平台数据库设计

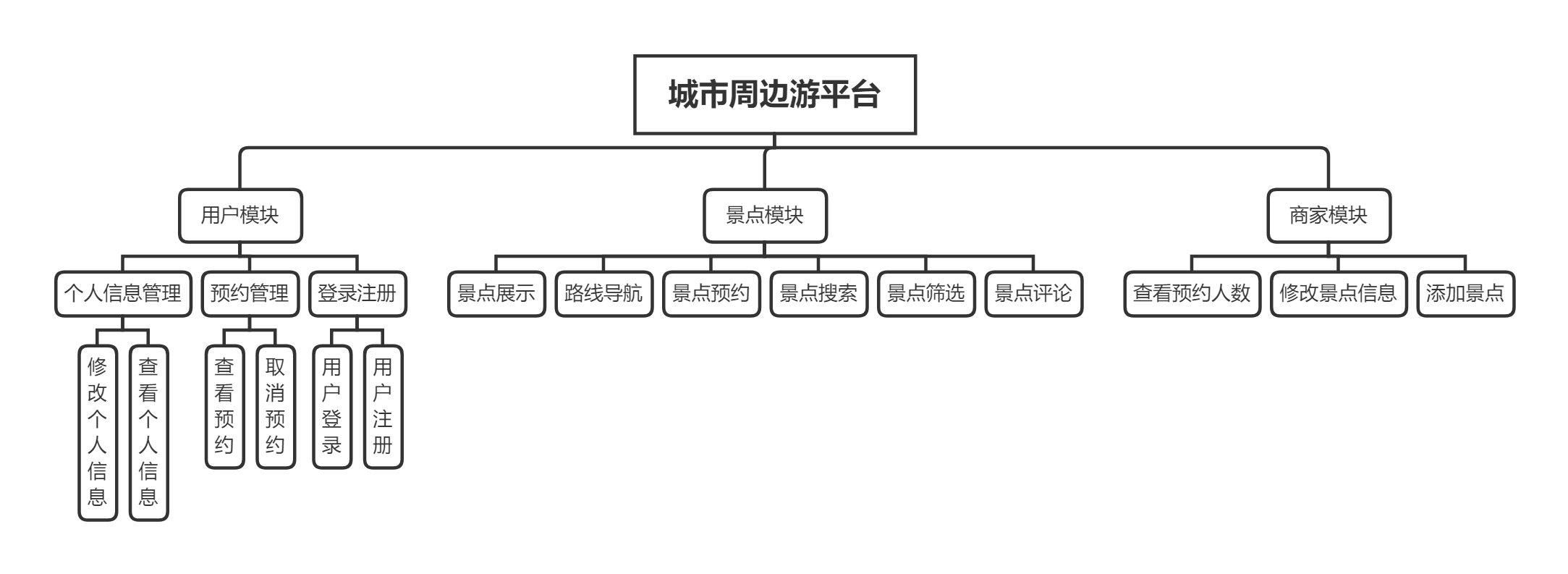
### 成都大学计算机学院

| 学号 | 班级 | 姓名 |
| --- | --- | --- |
| 201810414216 | 软件(本)18-2 | 覃茂辉 |

## 1.概述

* 基于oracle的城市周边游平台设计。前端采用vue+element-ui搭建，后台采用SSM框架+oracle数据库。
* 用户可通过切换城市查看不同城市周边的景点。
* 用户可对距离目的地距离、关键字、评分进行景点搜索。
* 用户可点击查看详情查看景点相关信息和导航路线（百度地图API）。
* 用户登陆后可进行景点预约并在我的预约中查看或取消预约。
* 用户登陆后可对景点进行评论。
* 商户可添加、修改景点信息，查看预约景点的人数。

## 2.系统总体结构



## 3.表设计

## ADMIN表

| 字段 | 类型 | 主键，外键 | 可以为空 | 默认值 | 约束 | 说明 |
| --- | --- | --- | --- | --- | --- | --- |
| ADMIN\_ID | NUMBER(8,0) | 主键 | 否 |  |  | 管理员ID |
| ADMIN\_NAME | VARCHAR2(50 BYTE) |  | 否 |  |  | 管理员用户名 |
| ADMIN\_PASSWORD | VARCHAR2(512 BYTE) |  | 否 | 123456 |  | 管理员密码 |

## COMMENT表

| 字段 | 类型 | 主键，外键 | 可以为空 | 默认值 | 约束 | 说明 |
| --- | --- | --- | --- | --- | --- | --- |
| COMMENT\_ID | NUMBER(8,0) | 主键 | 否 |  |  | 评论ID |
| PLACE\_ID | NUMBER(8,0) |  | 否 |  |  | 评论所对应的景点ID |
| USER\_ID | NUMBER(8,0) |  | 否 |  |  | 发布评论的用户ID |
| USERNAME | VARCHAR2(50 BYTE) |  | 否 |  |  | 发布评论的用户名称 |
| DATETIME | DATE |  | 否 |  |  | 发布评论的时间 |
| COMMENTTEXT | VARCHAR2(512 BYTE) |  | 否 |  |  | 评论的内容 |

## PLACE表

| 字段 | 类型 | 主键，外键 | 可以为空 | 默认值 | 约束 | 说明 |
| --- | --- | --- | --- | --- | --- | --- |
| PLACE\_ID | NUMBER(18,0) | 主键 | 否 |  |  | 景点ID |
| PLACE\_TITLE | NUMBER(8,0) |  | 否 |  |  | 景点标题 |
| PLACE\_DISTANCE | NUMBER(8,0) |  | 否 |  |  | 用户到景点距离 |
| PLACE\_RATE | VARCHAR2(50 BYTE) |  | 否 |  |  | 景点的评分 |
| PLACE\_CITY | DATE |  | 否 |  |  | 景点所处的城市 |
| PLACE\_PIC | VARCHAR2(512 BYTE) |  | 否 |  |  | 景点的封面图片地址 |
| PLACE\_INTRODUCE | VARCHAR2(512 BYTE) |  | 否 |  |  | 景点的描述 |
| PLACE\_ADDRESS | VARCHAR2(108 BYTE) |  | 否 |  |  | 景点的地址 |
| PLACE\_PHONE | VARCHAR2(20 BYTE) |  | 否 |  |  | 景点的联系电话 |

## RESERVATION表

| 字段 | 类型 | 主键，外键 | 可以为空 | 默认值 | 约束 | 说明 |
| --- | --- | --- | --- | --- | --- | --- |
| RESERVATION\_ID | NUMBER(18,0) | 主键 | 否 |  |  | 预约ID |
| RESERVATION\_PLACE\_ID | NUMBER(8,0) |  | 否 |  |  | 预约景点ID |
| RESERVATION\_USER\_ID | NUMBER(8,0) |  | 否 |  |  | 预约用户ID |
| RESERVATION\_NOW\_TIME | DATE |  | 否 |  |  | 发起预约的时间 |
| RESERVATION\_AIM\_TIME | DATE |  | 否 |  |  | 预约的时间点 |
| RESERVATION\_USER\_NAME | VARCHAR2(50 BYTE) |  | 否 |  |  | 预约用户的用户名 |
| RESERVATION\_USER\_NUMS | NUMBER(5,0) |  | 否 |  |  | 预约的用户人数 |
| RESERVATION\_PLACE\_NAME | NUMBER(5,0) |  | 否 |  |  | 预约的景点名 |

## USER表

| 字段 | 类型 | 主键，外键 | 可以为空 | 默认值 | 约束 | 说明 |
| --- | --- | --- | --- | --- | --- | --- |
| USER\_ID | NUMBER(18,0) | 主键 | 否 |  |  | 用户ID |
| USER\_PASSWORD | VARCHAR2(48 BYTE) |  | 否 |  |  | 用户密码 |
| USER\_EMAIL | VARCHAR2(56 BYTE) |  | 否 |  |  | 用户邮箱地址 |
| USER\_NAME | VARCHAR2(50 BYTE) |  | 否 |  |  | 用户昵称 |

## 4.表空间设计

#### 数据存储表空间创建

由于需要存储的表数量较多，数据量较大（超过5万条），同时为方便权限管理以及性能考虑，将数据量最多、隐秘性要求最高的的**USER**表、**ADMIN**表放入**MAOHUI01**表空间，将数据量较小的**RESERVATION**表、**PLACE**表、**COMMENT**表放入表空间**MAOHUI02**中，为**MAOHUI01**表空间分配650M空间，为**MAOHUI02**表空间分配350M空间。

SQL语句：

创建表空间MAOHUI01:

create tablespace MAOHUI01 datafile '/home/oracle/app/oracle/oradata/orcl/pdborcl/MAOHUI01.dbf' size 650m AUTOEXTEND ON NEXT 1024M MAXSIZE UNLIMITED;

创建表空间MAOHUI02:

create tablespace MAOHUI02 datafile '/home/oracle/app/oracle/oradata/orcl/pdborcl/MAOHUI01.dbf' size 350m AUTOEXTEND ON NEXT 512M MAXSIZE UNLIMITED;

#### Undo表空间

Undo 类型的表空间，当你对一张表或一条记录进行修改的时候，它会对修改之前的信息进行保存，这样可以保证数据的回滚。Undo 只包含undo类型的对象，不能包含任何其他对象，只适合于数据文件和区间管理。

为保证用户数据安全，决定创建Undo表空间用于回滚数据。

创建Undo表空间：

create undo tablespace undo1 datafile '/home/oracle/app/oracle/oradata/orcl/pdborcl/MAOHUIUndo.dbf' size 600m;

#### Temporary表空间

临时表空间，相当于一个临时的垃圾场。用于排序操作，比如你要做一次大数据量的查询，但在内存无法存储这么大量的数据，然后会在磁盘上建立一个临时的表空间用记存放这些数据。Oracle就会用这个临时表空间做排序，存储中间结果。

由于数据量较大，为方便快速查询，避免内存不足，决定创建Temprary表空间用户存放临时数据。

创建Temporary表空间：

create temporary tablespace TEMP datafile '/home/oracle/app/oracle/oradata/orcl/pdborcl/MAOHUITemporary.dbf' size 50m extent management local uniform size 15m;

## 5.用户管理

#### 创建角色

根据本系统的设计，本系统主要有两种角色，即商家和普通用户。

商家角色可以查询ADMIN表、PLACE表、RESERVATION表、COMMENT表，修改ADMIN表、PLACE表。

普通用户角色可以查询并修改USER表、RESERVATION表、COMMENT表,查询PLACE表。

* 创建商家角色：
* create role admin;  
    
  grant create session to admin;  
    
  grant select,update on ADMIN to admin;  
    
  grant select,update,delete on PLACE to admin;  
    
  grant select on RESERVATION to admin;  
    
  grant select on COMMENT to admin;
* 创建普通用户角色：
* create role user;  
    
  grant create session to user;  
    
  grant select,update on USER to user;  
    
  grant select,update,delete on RESERVATION to user;  
    
  grant select on PLACE to user;  
    
  grant select,insert on COMMENT to admin;

#### 创建用户

创建user1和admin1分别隶属于user角色和admin角色，并分别存放于表空间MAOHUI01和MAOHUI02，共用同一个Temperary表空间temp存放临时数据。

* 创建user1：
* create user user1 identified by 123 default tablespace MAOHUI01 temporary tablespace TEMP quota 50M on MAOHUI01；
* grant user to user1;
* 创建admin1：
* create user admin1 identified by 123 default tablespace MAOHUI02 temporary tablespace TEMP quota 50M on MAOHUI02；
* grant admin to admin1;

## 6.PL/SQL设计

#### 创建包定义

CREATE OR REPLACE PACKAGE MAOHUI\_PKG  
IS  
 --商家登录  
 FUNCTION admin\_login(  
 username VARCHAR2,   
 password VARCHAR2)  
 RETURN NUMBER;  
 --通过id查找商家账号  
 PROCEDURE admin\_findById(admin\_id IN NUMBER,admin OUT ADMIN%rowtype);  
 --修改商家账号信息  
 FUNCTION admin\_update(  
 username VARCHAR2,   
 password VARCHAR2,  
 admin\_id NUMBER)  
 RETURN NUMBER;  
 --添加景点  
 FUNCTION place\_insert(  
 place\_title VARCHAR2,   
 place\_distance NUMBER,  
 place\_rate NUMBER,  
 place\_city VARCHAR2,   
 place\_pic VARCHAR2)  
 RETURN NUMBER;  
 --按城市查找景点  
 PROCEDURE place\_findByCity(  
 city IN VARCHAR2,  
 places OUT PLACE%rowtype);  
 --按关键字、评分、距离、城市查找景点  
 PROCEDURE place\_search(  
 city IN VARCHAR2,  
 rate IN NUMBER,  
 distance IN VARCHAR2,  
 keywords IN VARCHAR2,  
 places OUT PLACE%rowtype);  
 --按id查找评论  
 PROCEDURE place\_search(  
 comment\_id IN NUMBER,  
 comment OUT COMMENT%rowtype);  
 --用户登录  
 FUNCTION user\_login(  
 username VARCHAR2,   
 password VARCHAR2)  
 RETURN NUMBER;  
 --修改用户信息  
 FUNCTION user\_update(  
 username VARCHAR2,   
 password VARCHAR2,  
 user\_id NUMBER)  
 RETURN NUMBER;  
 --通过用户id查找用户  
 PROCEDURE user\_findById(user\_id IN NUMBER,user OUT USER%rowtype);  
 --添加用户预约  
 FUNCTION reservation\_insert(  
 reservation\_place\_id NUMBER,  
 reservation\_user\_id NUMBER,  
 reservation\_nums NUMBER,  
 reservation\_now\_time DATE,   
 reservation\_aim\_time DATE,   
 reservation\_user\_name VARCHAR2,   
 reservation\_place\_title VARCHAR2)  
 RETURN NUMBER;  
 --取消预约  
 FUNCTION delete\_reservation(reservation\_id NUMBER)  
 RETURN NUMBER;  
END MAOHUI\_PKG;

#### 创建包主体

create or replace package body MAOHUI\_PKG  
is  
 --商家登录  
 function admin\_login(  
 username VARCHAR2,   
 password VARCHAR2)  
 return number is  
 num number(7,2);  
 begin  
 select count(\*) into num from ADMIN where ADMIN\_NAME=username and ADMIN\_PASSWORD=password;  
 return num;  
 end;  
   
 --通过id查找商家账号  
 PROCEDURE admin\_findById(  
 admin\_id IN NUMBER,  
 admin OUT ADMIN%rowtype)  
 is  
 BEGIN  
 select \* into admin from ADMIN where ADMIN\_ID=admin\_id;  
 end;  
   
 --修改商家账号信息  
 FUNCTION admin\_update(  
 username VARCHAR2,   
 password VARCHAR2,  
 admin\_id NUMBER)  
 RETURN NUMBER is  
 num number(7,2);  
 begin  
 num:= update ADMIN set ADMIN\_NAME=username,ADMIN\_PASSWORD=password where ADMIN\_ID=admin\_id;  
 end;  
   
 --添加景点  
 FUNCTION place\_insert(  
 place\_title VARCHAR2,   
 place\_distance NUMBER,  
 place\_rate NUMBER,  
 place\_city VARCHAR2,   
 place\_pic VARCHAR2)  
 RETURN NUMBER is  
 num number(7,2);  
 begin  
 num:= insert into PLACE(PLACE\_TITLE,PLACE\_DISTANCE,PLACE\_RATE,PLACE\_CITY,PLACE\_PIC) values (place\_title,place\_distance,place\_rate,place\_city,place\_pic);  
 end;  
   
 --按城市查找景点  
 PROCEDURE place\_findByCity(  
 city IN VARCHAR2,  
 places OUT PLACE%rowtype)  
 is  
 I BINARY\_INTEGER := 0;  
 BEGIN  
 FOR place IN ( select \* from PLACE where PLACE\_CITY=city; ) LOOP  
 places(i) := place;  
 I := I + 1;  
 END LOOP;  
 end;  
   
 --按关键字、评分、距离、城市查找景点  
 PROCEDURE place\_search(  
 city IN VARCHAR2,  
 rate IN NUMBER,  
 distance IN VARCHAR2,  
 keywords IN VARCHAR2,  
 places OUT PLACE%rowtype)  
 is  
 I BINARY\_INTEGER := 0;  
 BEGIN  
 FOR place IN ( select \* from PLACE where PLACE\_CITY=city and PLACE\_RATE&gt;=rate and PLACE\_DISTANCE&lt;=distance and locate(keyword,PLACE\_TITLE); ) LOOP  
 places(i) := place;  
 I := I + 1;  
 END LOOP;  
 end;  
   
 --按id查找评论  
 PROCEDURE place\_search(  
 comment\_id IN NUMBER,  
 comment OUT COMMENT%rowtype) IS  
 is  
 BEGIN  
 select \* into comment from COMMENT where COMMENT\_ID=comment\_id;  
 end;  
   
 --用户登录  
 FUNCTION user\_login(  
 username VARCHAR2,   
 password VARCHAR2)  
 RETURN NUMBER is  
 num number(7,2);  
 begin  
 select count(\*) into num from USER where USER\_NAME=username and USER\_PASSWORD=password;  
 return num;  
 end;  
   
 --修改用户信息  
 FUNCTION user\_update(  
 username VARCHAR2,   
 password VARCHAR2,  
 user\_id NUMBER)  
 RETURN NUMBER is  
 num number(7,2);  
 begin  
 num:= update USER set USER\_NAME=username,USER\_PASSWORD=password where USER\_ID=user\_id;  
 end;  
   
 --通过用户id查找用户  
 PROCEDURE user\_findById(  
 user\_id IN NUMBER,  
 user OUT USER%rowtype)  
 is  
 BEGIN  
 select \* into user from USER where USER\_ID=user\_id;  
 end;  
   
 --添加用户预约  
 FUNCTION reservation\_insert(  
 reservation\_place\_id NUMBER,  
 reservation\_user\_id NUMBER,  
 reservation\_nums NUMBER,  
 reservation\_now\_time DATE,   
 reservation\_aim\_time DATE,   
 reservation\_user\_name VARCHAR2,   
 reservation\_place\_title VARCHAR2)  
 RETURN NUMBER is  
 num NUMBER(7,2);  
 BEGIN  
 num := insert into RESERVATION(RESERVATION\_PLACE\_ID,RESERVATION\_USER\_ID,RESERVATION\_NOW\_TIME,RESERVATION\_AIM\_TIME,RESERVATION\_USER\_NAME,RESERVATION\_USER\_NUMS,RESERVATION\_PLACE\_NAME) values (reservation\_place\_id,reservation\_user\_id,reservation\_now\_time,reservation\_aim\_time,reservation\_user\_name,reservation\_nums,reservation\_place\_title);  
 END;  
   
 --取消预约  
 FUNCTION delete\_reservation(reservation\_id NUMBER)  
 RETURN NUMBER is  
 num NUMBER(7,2);  
 BEGIN  
 num := delete from RESERVATION where RESERVATION\_ID=reservation\_id;  
 END;  
   
end;

## 7.备份方案

#### 备份数据库的三种方法

* 冷备份
* 热备份
* 物理备份
* 由于本系统非关键应用系统，无需24小时不间断运行，考虑到冷备份简单易操作的特性，本系统拟采用冷备份的方式进行备份。

#### 冷备份

(1) 关闭数据库

shutdown immediate；

(2) 复制文件

1）复制三个文件夹

admin；

oradata(datafile， controlfile，redo)；

flash\_recovery\_area三个文件夹。

2）dbhome\_1下的内容

database(PWDfile、pfile)；

dbs(spfile)；

NETWORK/ADMIN(listener.ora、tnsnames.ora)。

(3) 恢复数据库

1）创建一个和原来一样的数据库。(安装路径和数据库名必须和原来一致)

2）停止数据库shutdown immediate；  
 3）复制安装目录下的admin、oradata、flash\_recovery\_area覆盖，复制database(PWDfile、pfile) 覆盖  
 4）启动数据库 startup;

备注：

无归档模式：将备份文件拷贝回原来的目录即可，然后启动数据库。  
归档模式：  
 ①将数据文件、控制文件拷贝回原来目录  
 ②SQL>startup mount;  
 SQL>recover database using backup controlfile;  
 ③将相应的归档日志和联机日志拖到CMD命令窗口进行跑日志。  
 ④SQL>alter database open resetlogs;