Hospital ManageMent System

SUBJECT: DATABASE MANAGEMENT SYSTEM

DHARAMSINH DESAI UNIVERSITY

COMPUTER DEPARTMENT

FACULTY OF TECHNOLOGY, NADIAD

PREPARED BY:

Rajyaguru Keval(CE-102)

Rathod Bhargav(CE-107)

GUIDED BY:

PROF. J. M. Pandya

INDEX

PROJECT OVERVIEW.

Goals:-

The ultimate goal of 'MEDONYNE' is to fully computerize the administration of medical organizations such as hospitals,clinics,medical stores etc. The objectives is to check manual process which requires laborious effort make easier such as invoice generation,proper storing of medicines,incoming and outgoing of staff members,patient profile management.

Scopes:-

The MEDODYNE is equipped with some unique facilities like online appointment generation and current patient number check in system which will be quite useful for patient as the waiting time for one's appointment will be considerably low. Plus on the administration side, facilities like staff management will include recording in and out timing of staff,there profile and their pay scale. The MEDODYNE is show availability of the doctors,the consulting timing too.

.

Features:-

-Create your own league & compete with your friends.

-GUI would be easier to manage your team.

-Individual points of each player will be shown in the result.

1.The system provide online appointment generation.

2.The system tracks current patients.

3.System will provide tracking and timing of staff.

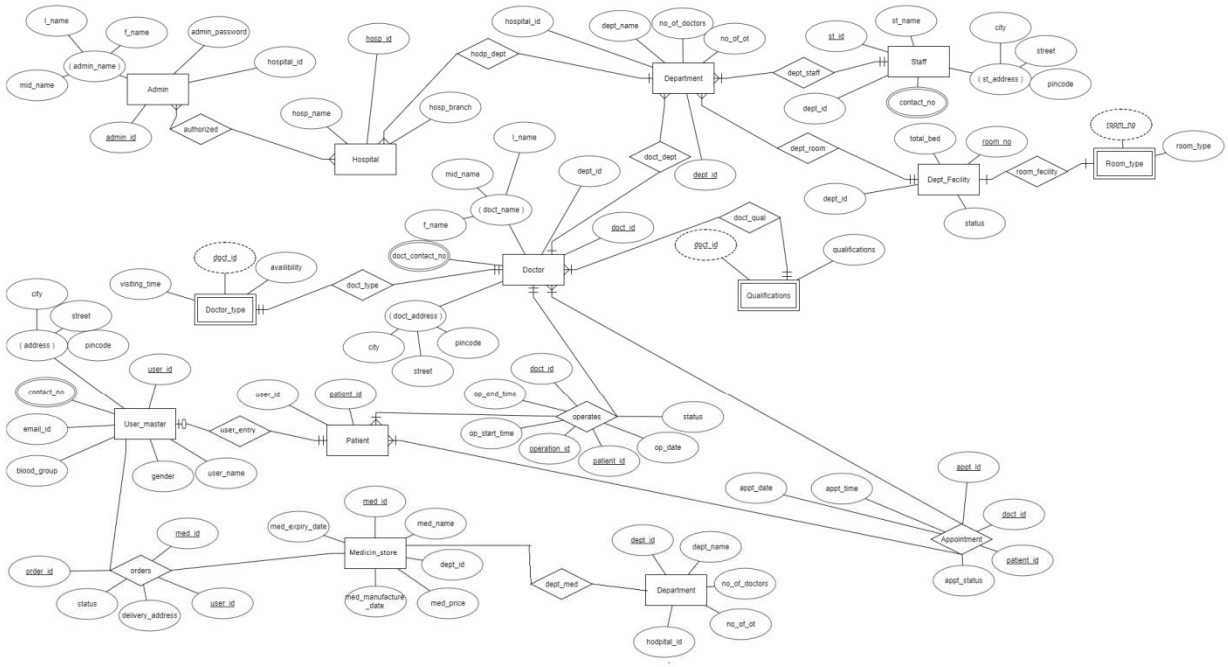
4.The system provides records of different operations.

5.The system provides record of different fields.

6.System gives availability of beds information.

7.System generates computer based invoice.

ER DIAGRAM



IMPLEMENTATION

TABLE CREATION

1.Hospital Table

create SEQUENCE hospital_id_seq start with 0001 INCREMENT BY 1 MAXVALUE 5000 NOCYCLE nocache ORDER;

CREATE TABLE hospital (hospital_id VARCHAR(10), hospital_name VARCHAR(20), hospital_branch VARCHAR(20), PRIMARY KEY(hospital_id));

alter table hospital modify (hospital_id varchar(30));

HOSPITAL				
Sr.No	Field Name	Data Type	Size	Constraint
1.	Hospital_id	Varchar	20	Primary key
2.	Hospital_name	Varchar	20	
3.	Hospital_branch	Varchar	10	

- insert into hospital(HOSPITAL_ID,HOSPITAL_NAME,hospital_branch) values (to_char(sysdate,'YYYYMMDD')||'HOSPITAL'||hospital_id_seq.nextval,'sterling','Bhavna gar');

- insert into hospital(HOSPITAL_ID,HOSPITAL_NAME,hospital_branch) values (to_char(sysdate,'YYYYMMDD')||'HOSPITAL'||hospital_id_seq.nextval,'sterling','Surat');

2.Admin Table

create SEQUENCE admin_id_seq start with 0001 INCREMENT BY 1 MAXVALUE 5000 NOCYCLE nocache ORDER;

CREATE TABLE admin_master (admin_id VARCHAR(10), admin_first_name VARCHAR(20), admin_mid_name varchar(20), admin_last_name varchar(20), admin_password VARCHAR(10), hospital_id REFERENCES hospital(hospital_id), PRIMARY KEY(admin_id));

alter table admin_master modify (admin_id varchar(30));

alter table admin_master modify (hospital_id varchar(30));

ADMIN_MASTER				
Sr.No	Field Name	Data Type	Size	Constraint
1.	Admin_id	varchar	10	Primary key

2.	Admin_first_name	Varchar	10	
3.	Admin_mid_name	Varchar	20	
4.	Admin_last_name	Varchar	20	
5.	Admin_password	Varchar	10	
6.	Hospital_id	Varchar	20	Foreign key(hospital(hospital_id))

• insert into admin_master (admin_id,ADMIN_FIRST_NAME,admin_mid_name,admin_last_name,admin_password, hospital_id) values(to_char(sysdate,'YYYYMMDD')||'ADMIN'||admin_id_seq.nextval,'Keval','Rajyag uru','Atulkumar','mypassword','20171101HOSPITAL2');

3.Department Table

create SEQUENCE dept_id_seq start with 0001

INCREMENT BY 1 MAXVALUE 5000 NOCYCLE nocache ORDER;

CREATE TABLE department_master (dept_id VARCHAR(10), dept_name VARCHAR(15), no_of_doctors INT, no_of_operation_theater INT, hospital_id REFERENCES hospital(hospital_id), PRIMARY KEY(dept_id));

alter table department_master modify (dept_name varchar(30)); alter table department_master modify (doct_id varchar(30));

alter table department_master modify (hospital_id varchar(30));

DEPARTMENT_MASTER				
Sr.No	Field Name	Data Type	Size	Constraint
1.	Dept_id	Varchar	10	Primary key
2.	Dept_name	Varchar	15	
3.	No_of_doctors	Int		
4.	No_of_doctors	Int		
5.	Hospital_id	Varchar	20	Foreign key(hospital(hospital_id))

• insert into department_master(dept_id,dept_name,no_of_doctors,no_of_operation_theater,hosp ital_id) values(to_char(sysdate,'YYYYMMDD')||'DEPT'||dept_id_seq.nextval,'Dental Department',2,1,'20171101HOSPITAL2');

• insert into department_master(dept_id,dept_name,no_of_doctors,no_of_operation_theater,hosp ital_id) values(to_char(sysdate,'YYYYMMDD')||'DEPT'||dept_id_seq.nextval,'Neuro Department',2,1,'20171101HOSPITAL2');

4.Doctor Table

create SEQUENCE doct_id_seq start with 0001 INCREMENT BY 1 MAXVALUE 5000 NOCYCLE nocache ORDER;

CREATE TABLE doctor_master (doct_id VARCHAR(10), doct_first_name VARCHAR(30), doct_mid_name varchar(30), doct_last_name varchar(30), dept_id REFERENCES department_master(dept_id), PRIMARY KEY(doct_id));

alter table doctor_master modify (doct_id varchar(30));

alter table doctor_master modify (doct_id varchar(30));

DOCTOR_MASTER				
Sr.No	Field Name	Data Type	Size	Constraint
1.	Doct_id	Varchar	10	Primary key
2.	Doct_first_name	Varchar	30	
3.	Doct_mid_name	Varchar	30	
4.	Doct_last_name	Varchar	30	
5.	Dept_id	Varchar	10	Foreign key department_master(dept_id)

• insert into doctor_master (doct_id,doct_first_name,doct_mid_name,doct_last_name,dept_id) values (to_char(sysdate,'YYYYMMDD')||'DOCT'||doct_id_seq.nextval,'Hardik','Upadhyay','Nar eshBhai','20171102DEPT3');

5.Contact for Doctor

create table doctor_contact (doct_id varchar(10) references doctor_master(doct_id), doct_contact_no int constraint cont_cons CHECK(REGEXP_LIKE(doct_contact_no,'^([1-9]{1}[0-9]{9})$')));

alter table doctor_contact modify (doct_id varchar(30));

DOCTOR_CONTACT

Sr.No	Field Name	Data Type	Size	Constraint
1.	Doctor_id	Varchar	10	Foreign key(doctor_master(doct_id))
2.	Doctor_contact_no	Int		cont_cons CHECK(REGEXP_LIKE(doct_contact_no,'^([1-9]{1}[0-9]{9})$'))

- insert into doctor_contact (doct_id,doct_contact_no) values('20171102DOCT5',9598789565);

6.Qualifications for perticutlar Doctor

CREATE TABLE qualification (doct_id varchar(10) REFERENCES doctor_master(doct_id), qualification VARCHAR(25));

alter table qualification modify (doct_id varchar(30));

QUALIFICATION				
Sr.No	Field Name	Data Type	Size	Constraint
1.	Doct_id	Varchar	10	Foreign key (doctor_master(doct_id))
2.	Qualification	Varchar	25	

• insert into qualification (doct_id,QUALIFICATION) values('20171102DOCT5','MBBS'); • insert into qualification (doct_id,QUALIFICATION) values('20171102DOCT5','MD');

7.Class Of Doctor

CREATE TABLE doctor_class

(doct_id REFERENCES doctor_master(doct_id), visiting_time VARCHAR2(25), availability VARCHAR2(25) check in('temporary','full service','special service')), avail_count int, PRIMARY KEY(doct_id));

alter table doctor_class modify (doct_id varchar(30));

alter table doctor_class modify (visiting_time varchar(100));

DOCTOR_CLASS

Sr.No	Field Name	Data Type	Size	Constraint
1.	Doct_id	Varchar	10	Foreign key (doctor_master(doct_id))
2.	Visiting_time	Varchar	25	
3.	Availability	Varchar	25	check in('temporary','full service','special service'))
4.	Avail_count	Int		

- insert into doctor_class (doct_id,visiting_time,availability) values('20171102DOCT5','8a.m. to 1p.m. and 4 to 7p.m.','every day');

8.User Table

create SEQUENCE user_id_seq start with 0001 INCREMENT BY 1 MAXVALUE 5000 NOCYCLE nocache ORDER;

CREATE TABLE user_master (user_id VARCHAR(10) primary key, user_first_name VARCHAR(20), user_mid_name varchar(20), user_last_name varchar(20), user_password varchar(30) CHECK (REGEXP_LIKE(user_password,'^([a-zA-Z0-9@*#] {7,14})$')), user_age INT, user_gender VARCHAR(2) CHECK (user_gender IN ('M','F','O')), user_blood_grp VARCHAR(10), user_description VARCHAR(30), user_city VARCHAR(30), user_street varchar(30), user_pincode int check(REGEXP_LIKE(user_pincode,'^([0-6){6}')), user_email varchar(50) check (REGEXP_LIKE(user_email,'^([a-zA-Z0-9_ \-\.]+)@((gmail.com)|(yahoo.com))$')));

alter table user_master modify (user_id varchar(30));

alter table user_master modify (user_pincode int check(REGEXP_LIKE(user_pincode,'^([0-6] {6})$')));

ALTER TABLE user_master MODIFY (user_email VARCHAR(50) CHECK (REGEXP_LIKE(user_email,'^([a-zA-Z0-9_\-\.]+)@((gmail.com)|(yahoo.com))$')));

USER_MASTER				
Sr.No	Field Name	Data Type	Size	Constraint
1.	User_id	Varchar	10	Primary key
2.	Use_first_name	Varchar	20	
3.	User_mid_name	Varchar	20	
4.	User_last_name	Varchar	20	
5.	user_password	Varchar	30	check(REGEXP_LIKE(user_password,'^([a-zA-Z0-9@*#]{7,14})$'))
6.	User_age	Int		

7.	User_gender	Varchar	2	(user_gender IN ('M','F','O'))
8.	User_blood_grp	Varchar	10	
9.	User_description	Varchar	30	
10.	User_city	Varchar	30	
11.	User_street	Varchar	30	
12.	User_pincode	Int		check(REGEXP_LIKE(user_pincode,'^([0-
13.	User_email	Varchar	50	check (REGEXP_LIKE(user_email,'^([a-zA-

• insert into user_master(USER_ID,USER_FIRST_NAME,USER_MID_NAME,USER_LAST_NAME,USER_PASSWORD,USER_AGE,USER_GENDER,USER_BLOOD_GRP,USER_DESCRIPTION,USER_CITY,USER_STREET,USER_PINCODE,USER_EMAIL)
values(to_char(sysdate,'YYYYMMDD')||'USER'||user_id_seq.nextval,'Sandhya','Suraj','Rathi','ascvdgfjdks',27,'F','B+',null,'Pushkar','Hanuman Gali',436036,'diyaaurbati@gmail.com');

• insert into user_master(USER_ID,USER_FIRST_NAME,USER_MID_NAME,USER_LAST_NAME,USER_PASSWORD,USER_AGE,USER_GENDER,USER_BLOOD_GRP,USER_DESCRIPTION,USER_CITY,USER_STREET,USER_PINCODE,USER_EMAIL)
values(to_char(sysdate,'YYYYMMDD')||'USER'||user_id_seq.nextval,'Sandhya','Suraj','Rathi','ascvdgfjdks',27,'F','B+',null,'Pushkar','Hanuman Gali',436036,null);

9.Contact_no of user

create table user_contact (user_id varchar(10) references user_master(user_id), user_contact_no int, constraint cont1_cons CHECK(REGEXP_LIKE(user_contact_no,'^([1-9]{1}[0-9]{9})$')));

alter table user_contact modify (user_id varchar(30));

USER_CONTACT				
Sr.No	Field Name	Data Type	Size	Constraint
1.	User_id	Varchar	10	Foreign key user_master(user_id)
2.	User_contact_no	Int		constraint cont1_cons CHECK(REGEXP_LIKE(user_contact_no,'^([1-9]{1}[0-9]{9})$'))

• insert into user_contact (user_id,user_contact_no) values('20171103USER59',8248648624);

10.Patient Table

create SEQUENCE pt_id_seq start with 0001 INCREMENT BY 1 MAXVALUE 5000 NOCYCLE nocache ORDER;

CREATE TABLE patient_master (pt_id VARCHAR(10) PRIMARY KEY, user_id REFERENCES user_master(user_id));

alter table patient_master modify(pt_id varchar(30));

alter table patient_master modify(user_id varchar(30));

PATIENT_MASTER				
Sr.No	Field Name	Data Type	Size	Constraint
1.	Pt_id	Varchar	10	Primary key
2.	User_id	Varchar	10	Foreign key user_master(user_id)

insert into patient_master(pt_id,user_id)
values(to_char(sysdate,'YYYYMMDD')||'PT'||pt_id_seq.nextval,'20171103USER59');

11.Appointment Table

create SEQUENCE apt_id_seq start with 0001 INCREMENT BY 1 MAXVALUE 5000 NOCYCLE nocache ORDER;

CREATE TABLE appointment (apt_id NUMBER(10), doct_id REFERENCES doctor_master(doct_id), pt_id REFERENCES patient_master(pt_id), apt_status VARCHAR(20), apt_date DATE, apt_time VARCHAR(10), PRIMARY KEY(apt_id));

alter table appointment modify(pt_id varchar(30));

alter table appointment modify(doct_id varchar(30));

alter table appointment modify(apt_id varchar(30));

alter table appointment modify(apt_status varchar(20) check(apt_status in('appointed','in progress','completed','cancelled')));

APPOINTMENT				
Sr.No	Field Name	Data Type	Size	Constraint
1.	Apt_id	Number	10	Primary key
2.	Doct_id	Varchar	10	doctor_master(doct_id),
3.	Pt_id	Varchar	10	patient_master(pt_id)
4.	Apt_status	Varchar	20	
5.	Apt_date	Date		
6.	Apt_time	Varchar	10	

• insert into appointment (APT_ID,DOCT_ID,PT_ID,APT_STATUS,APT_DATE,APT_TIME) values(to_char(sysdate,'YYYYMMDD')||'APT'||apt_id_seq.nextval,'20171102DOCT5', '20171103PT90','appointed','03-Nov-2017','03 p.m.');

12.Staff Table

create sequence staff_id_seq start with 0001 increment by 1 maxvalue 5000 nocycle nocache order;

CREATE TABLE staff (staff_id VARCHAR(10), staff_first_name VARCHAR(30), staff_last_name varchar(30), staff_mid_name varchar(30), dept_id REFERENCES department_master(dept_id), staff_address VARCHAR(35), PRIMARY KEY(staff_id));

alter table staff modify(staff_id varchar(30)); alter table staff modify(dept_id varchar(30));

STAFF				
Sr.No	Field Name	Data Type	Size	Constraint
1.	Staff_id	Varchar	10	Primary key
2.	Staff_first_name	Varchar	30	
3.	Staff_last_name	Varchar	30	
4.	Staff_mid_name	Varchar	30	
5.	Dept_id	Varchar	10	Foreign key
6.	Staff_address	Varchar	35	

• insert into staff(STAFF_ID,STAFF_FIRST_NAME,STAFF_LAST_NAME,STAFF_MID_NAME,DEPT_ID,ST AFF_ADDRESS) values
(to_char(sysdate,'YYYYMMDD')||'STAFF'||staff_id_seq.nextval,'Bhargav','Rathod','N','2 0171102DEPT2','add1');

13.Contact_no.s of staff members

create table staff_contact (staff_id varchar(10) references staff(staff_id), staff_contact_no int, constraint cont_cons2 CHECK(REGEXP_LIKE(staff_contact_no,'^([1-9]{1}[0-9]{9})$'))

);

alter table staff_contact modify(staff_id varchar(30));

STAFF_CONTACT				
Sr.No	Field Name	Data Type	Size	Constraint
1.	Staff_id	Varchar	10	Foreign key staff(staff_id)

2.	Staff_contact_no	Int		constraint cont_cons2 CHECK(REGEXP_LIKE(staff_contact_no,'^([1-9]{1}[0-9]{9})$'))

• insert into staff_contact (staff_id,staff_contact_no) values('20171103STAFF2',1123456789);

14.Medicine Information data

create sequence med_id_seq start with 0001 maxvalue 5000 increment by 1 nocycle nocache order;

CREATE TABLE medicine_store (med_id VARCHAR(10), med_name VARCHAR(25), dept_id REFERENCES department_master(dept_id), company_name VARCHAR(30), med_price NUMBER(8,3), med_mfd DATE, med_expd DATE, PRIMARY KEY(med_id));

alter table medicine_store modify(med_id varchar(30));

alter table medicine_store modify(dept_id varchar(30));

MEDICINE_STORE				
Sr.No	Field Name	Data Type	Size	Constraint
1.	Med_id	Varchar	10	Primary key
2.	Med_name	Varchar	25	
3.	Dept_id	Varchar	10	Foreign key (department_master(dept_id))
4.	Company_name	Varchar	30	
5.	Med_price	Number	(8,3)	
6.	Med_mfd	Date		
7.	Med_expd	date		

• insert into medicine_store(MED_ID,MED_NAME,DEPT_ID,COMPANY_NAME,MED_PRICE,MED_MFD,MED_EXPD) values(to_char(sysdate,'YYMMDD')||'MED'||med_id_seq.nextval,'med1','20171102DEPT2','comp1',123,'01-Jan-2015','01-Jan-2017');

15.Order Information

create sequence order_id_seq start with 0001 maxvalue 5000 increment by 1 nocycle nocache order;

CREATE TABLE order_master (order_id VARCHAR2(15), med_id REFERENCES medicine_store(med_id), user_id REFERENCES user_master(user_id), deli_address VARCHAR2(30), status VARCHAR2(30), PRIMARY KEY(order_id));

alter table order_master modify(med_id varchar(30));

alter table order_master modify(user_id varchar(30));

alter table order_master modify(order_id varchar(30));

ORDER_MASTER				
Sr.No	Field Name	Data Type	Size	Constraint
1.	Order_id	Varchar	15	Primary key
2.	Med_id	Varchar	10	Foreign key medicine_store(med_id)
3.	User_id	Varchar	10	Foreign_key user_master(user_id)
4.	Deli_address	Varchar	30	
5.	Status	Varchar	30	

- insert into order_master(ORDER_ID,MED_ID,USER_ID,DELI_ADDRESS,STATUS)values (to_char(sysdate,'YYMMDD')||'ORDER'||order_id_seq.nextval,'171103MED1','20171103USER59','add1', 'completed');

16.Room and Bed Information

CREATE TABLE dept_facility (room_no INT, dept_id REFERENCES department_master(dept_id), tot_bed INT, PRIMARY KEY(room_no));

alter table dept_facility modify(dept_id varchar(30));

DEPT_FACILITY				
Sr.No	Field Name	Data Type	Size	Constraint
1.	Room_no	Int		Primary key
2.	Dept_id	Varchar	10	Foreign key department_master(dept_id)

3.	Tot_bed	int		

- insert into dept_facility(ROOM_NO,DEPT_ID,TOT_BED) values(1,'20171102DEPT2',10);

17.Current Room and Bed Status

CREATE TABLE room_status (room_no REFERENCES dept_facility(room_no), room_type_ac VARCHAR(10) default ('AC'), room_type_delux varchar(10) default ('NON-DELUX'), available_bed int, constraint roomtype1cons check(room_type_ac in('AC','NON-AC')), constraint roomtype2cons check(room_type_delux in('TV-REFRIGERATOR','NON-DELUX')));

ROOM_STATUS				
Sr.No	Field Name	Data Type	Size	Constraint
1.	Room_no	Int		Foreign key dept_facility(room_no)
2.	Room_type_ac	varchar	10	Default('AC')+ roomtype1cons check(room_type_ac in('AC','NON-AC')),
3.	Room_type_delux	Varchar	10	Default('NON-DELUX')+ roomtype2cons check(room_type_delux in('TV-REFRIGERATOR','NON-DELUX'))
4.	Available_bed	Int		

- insert into room_status(ROOM_NO,AVAILABLE_BED)values (1,7);

18.Operation Table

create sequence op_id_seq start with 0001 maxvalue 5000 increment by 1 nocycle nocache order; CREATE TABLE operates (doct_id REFERENCES doctor_master(doct_id), pt_id REFERENCES patient_master(pt_id), op_id VARCHAR2(20), op_date DATE , op_start_time VARCHAR2(20), op_end_time VARCHAR2(20), status VARCHAR2(15) CHECK (status IN ('Success','Failed','In Progress')), PRIMARY KEY(op_id));

alter table operates modify(doct_id varchar(30));

alter table operates modify(pt_id varchar(30));

alter table operates modify(op_id varchar(30));

OPERATES				
Sr.No	Field Name	Data Type	Size	Constraint
1.	Doct_id	Varchar	10	Foreign key doctor_master(doct_id),
2.	Pt_id	Varchar	10	patient_master(pt_id)
3.	Op_id	Varchar	20	Primary key
4.	Op_date	Date		
5.	Op_start_time	Varchar	20	
6.	Op_end_time	Varchar	20	
7.	Status	Varchar	15	CHECK (status IN ('Success','Failed','In Progress'))

- insert into operates(DOCT_ID,PT_ID,OP_ID,OP_DATE,OP_START_TIME,OP_END_TIME,STATUS) values('20171102DOCT5','20171103PT1',to_char(sysdate,'YYMMDD')||'OP'||op_id_seq .nextval,'21-Oct-2017',null,null,null);

Indexes And Views And Sequences

--1.Hospital Table

```
 create SEQUENCE hospital_id_seq start with 0001   INCREMENT BY 1 MAXVALUE 5000
NOCYCLE nocache ORDER;

--2.Admin Table

 create SEQUENCE admin_id_seq start with 0001   INCREMENT BY 1 MAXVALUE 5000 NOCYCLE
nocache ORDER;

--3.Department Table

 create SEQUENCE dept_id_seq start with 0001 INCREMENT BY 1 MAXVALUE 5000 NOCYCLE
nocache ORDER;

--4.Doctor Table

 create SEQUENCE doct_id_seq start with 0001   INCREMENT BY 1 MAXVALUE 5000 NOCYCLE
nocache ORDER;

--5.User Table

 create SEQUENCE user_id_seq start with 0001   INCREMENT BY 1 MAXVALUE 5000 NOCYCLE
nocache ORDER;

--6.Patient Table

 create SEQUENCE pt_id_seq start with 0001   INCREMENT BY 1 MAXVALUE 5000 NOCYCLE
nocache ORDER;

--7.Appointment Table

 create SEQUENCE apt_id_seq start with 0001   INCREMENT BY 1 MAXVALUE 5000 NOCYCLE
nocache ORDER;

--8.Staff Table

 create sequence staff_id_seq start with 0001  increment by 1 maxvalue 5000 nocycle nocache
order;

--9.Medicine Information data

 create sequence med_id_seq start with 0001  maxvalue 5000 increment by 1 nocycle nocache
order;

--10.Order Information
```

create sequence order_id_seq start with 0001 maxvalue 5000 increment by 1 nocycle nocache order;

--11.Operation Table

create sequence op_id_seq start with 0001 maxvalue 5000 increment by 1 nocycle nocache order;

1. create view user_credential as select user_id,user_password from user_master;

2.create index appointment_check_index on appointment

(apt_status,apt_date,apt_time);

SRS Queries

4.1.1 The system takes username and password for authentication

Input: Username and Password

Output: Valid or not

Description:If username and password are valid then system will allow to enter into software.

Queries:

create view user_credential as select user_id,user_password from user_master;

Select user_id,user_password from user_credential

where user_id=:uid and password=:pwd;

4.1.2 The system checks user is blend or admin

Input:Username and Password

Output:User or admin

Description:If entered username is for admin then system allows to log in as admin.

Query:--

4.1.3 System allows to sign up for new user

Input:All required credentials:name,password,address,number etc.

Output:Successfully registered or not

Description:the system takes all information for user and information valid then registered user successfully

Query: insert into
user_master(USER_ID,USER_FIRST_NAME,USER_MID_NAME,USER_LAST_NAME,USER_PASSWORD,USER_AGE,USER_GENDER,USER_BLOOD_GRP,USER_DESCRIPTION,USER_CITY,USER_STREET,USER_PINCODE,USER_EMAIL)
values(to_char(sysdate,'YYYYMMDD')||'USER'||user_id_seq.nextval,'Sandhya','Suraj','R athi','ascvdgfjdks',27,'F','B+',null,'Pushkar','Hanuman Gali',436036,'diyaaurbati@gmail.com');

insert into
user_master(USER_ID,USER_FIRST_NAME,USER_MID_NAME,USER_LAST_NAME,USER_PASSWORD,USER_AGE,USER_GENDER,USER_BLOOD_GRP,USER_DESCRIPTION,USER_CITY,USER_STREET,USER_PINCODE,USER_EMAIL)
values(to_char(sysdate,'YYYYMMDD')||'USER'||user_id_seq.nextval,'Sandhya','Suraj','R athi','ascvdgfjdks',27,'F','B+',null,'Pushkar','Hanuman Gali',436036,null);

4.1.4 System provides user to update them registered information

Input:Update value

Output:Updated successfully or not

Description:system takes updated values from user if valid then update.

Query:update user_master set user_street='galaxy road' where user_id=:uid; ---20171103USER1

;4.2 The system provides appointment generation to user

4.2.1 System will take information date,time for appointment

Input:Info,date,time,place

Output:appointment_flag

Description:System set appointment_flag true if all data are valid.

Query:insert into appointment

(APT_ID,DOCT_ID,PT_ID,APT_STATUS,APT_DATE,APT_TIME)
values(to_char(sysdate,'YYYYMMDD')||'APT'||apt_id_seq.nextval,

'20171102DOCT5', '20171103PT90','appointed','03-Nov-2017','03 p.m.');

```
create index appointment_check_index on appointment
                    (apt_status,apt_date,apt_time);

select apt_id,apt_status from appointment
 where apt_date =:dat;--03-NOV-17
```

4.2.2 System will notify time and reply to user

Input:Checking information

Output:Appointment_answer_flag

Description:The system will set appointment_answer_flag true, if appointment possible and it will notify to come for appointment.

Query::insert into appointment

(APT_ID,DOCT_ID,PT_ID,APT_STATUS,APT_DATE,APT_TIME)
values(to_char(sysdate,'YYYYMMDD')||'APT'||apt_id_seq.nextval,

'20171102DOCT5', '20171103PT90','appointed','03-Nov-2017','03 p.m.');

4.2.2.1 System also allow to user to cancel appointment

Input : Reason,appointment_id

output :appointment_cancel_answer_flag

description:System takes info for cancellation if possible and set appointment_cancel_answer_flag equals to true.

Query:update APPOINTMENT set

apt_status='cancelled' where

APT_ID=:aid;--20171103APT1

4.3 The system provides user to show current scenario of hospital

4.3.1 System checks and gives current scenario by hospital management

Input: scenario_name

Output: scenario_details

Description:System gives appropriate scenario and user will be showed it easily

4.3.1.1 The System gives current Doctor Details

Input: doctor_id,name

Output: doctor_details

Description:system gives information about doctor

Query: create index doc_search_index on doctor_master(doct_id,doct_first_name,doct_mid_name,

doct_last_name);

select distinct(dept_id),visiting_time,availability from doctor_master natural join doctor_class

where doct_first_name='Hardik' and doct_last_name='NareshBhai' and doct_mid_name='Upadhyay';

```
select distinct(dm.dept_id),dc.visiting_time,dc.availability
        from doctor_class dc ,doctor_master dm where dm.doct_id in
        (select doct_id from doctor_master where
dm.doct_first_name='Hardik' and
        dm.doct_mid_name='Upadhyay' and
dm.doct_last_name='NareshBhai');
```

4.3.1.2 The system gives current staff details

Input: staff_id,name

Output: staff_details

Description:system takes input as asked for and provide details about staff for their current working status

```
Query:select dept_name,staff_address,staff_contact_no from staff
natural join staff_contact
natural join department_master
where staff_first_name='fnm' and staff_last_name='lnm'
 and staff_mid_name='mnm';

select dept_name,staff_address,staff_contact_no
from staff st,department_master dm,staff_contact sc
          where st.dept_id in (select dept_id from staff where
          st.staff_first_name='fnm' and st.staff_last_name='lnm'
and st.staff_mid_name='mnm');
```

4.4 The system allows user to medicine shopping

4.4.1 The system allows user to choose any specific medicine for shopping to user

Input: medicine_name,medicine_id

Output: medicine_availability_flag

Description: system checks availability and gives response true by medicine_shopping flag

Query:create index med_nm on medicine_store(med_name);

select med_id from medicine_store where med_name='abc';

4.4.2 The system allows user to check quantity of medicines

Input: medicine_name,medicine_id

Output: medicine_quantity_no

Description:system gives maximum no. of quantity that can be available to you from store

Query: select count(med_name) from medicine_store group by(med_name) having med_name=:mnm;

4.4.3 The system allows user to check quality of medicines

Input: medicine_name,medicine_id

Output: medicine_quality

Description:system will get and give details about expiry dates and ingredients and another all with inputs

Query:select company_name,med_price,med_mfd,med_expd
from medicine_store where med_name=:mnm;

4.5 The system tracks in and out staff and laboratory data.

4.5.1 The System checks available staff and track them.

Input: staff_name,staff_id

Output: staff_details

Description:system can allow to share tracking details of staff to management

Query:select dept_name,staff_address,staff_contact_no

from staff st,department_master dm,staff_contact sc

```
where st.dept_id in (select dept_id from staff where
st.staff_first_name='fnm' and st.staff_last_name='lnm'
and st.staff_mid_name='mnm');
```

4.5.2 The system free or occupied bed details

Input: Room_no

Output: free_bed_no

Description:system gets count from management entries and provide details to naive or and user(including reception)

Query:

```
select available_bed,room_no,room_type_ac,
room_type_delux,tot_bed from dept_facility
natural join room_status;
```

4.6 The system generates invoice using user's profile,and shopping details

Input: User_information,Item_information

Output: Total_amount,Invoice_file

Description: System takes User_information and prescription_id or item_details and generates invoice in particular format

Query:

```
select sum(med_price) from order_master natural join medicine_store
where user_id=:uid1 group by(order_id);--20171103USER59
```

Report

To see Doctor Qualification for particular Doctor from Perticular Department and Hospital

Report:See Hospital List:select * from Hospital

See Department List For Selected Hospital id:select * from Department_master where hospital_id=:h_id

See Doctors Details from Department id:select * from doctor_master where dept_id=:de_id

see Doctor Qualification from Doctor id:select * from qualification where doct_id =:d_id

Deployment Steps

1. For created and permitted user First Drop all tables if already exists else Create in given Order:
    a. Hospital
    b. Admin
    c. Department
    d. Doctor
    e. Doctor_contact
    f. Doctor_Qualifications
    g. User_master
    h. User_Contact
    i. Patient_master
    j. Appointment
    k. Staff
    l. Staff_contact
    m. Medicine

n. Order_master

o. Fecility

p. Status

q. Operation

2. Now Create Sequences for Ids of Tables

a.hospital_id_seq

b.admin_id_seq

c.dept_id_seq

d.doct_id_seq

e.user_id_seq

f.pt_id_seq

g.apt_id_seq

h.staff_id_seq

i.med_id_seq

j.order_id_seq

h.op_id_seq

3. Create Appropriate User Defined Reports

hospitalList->DrillDown:Dept_Hosp by department

department->DrillDown:Doct_Dept by doctor

doctor->DrillDown:Doct_Qual by doctor_qualifications

doctor_qualifications->To see qualifications by doctor id

4. Create View User_credential and Indexes app_check_index and doct_search_index if need.

Insert Values in Tables in Valid Order and alter table column for size or constraint if required.

5. Commit all Changes

Oracle Features Summary

Oracle Features	Used in Project
Triggers	No
Sequences	Yes
Cursor	No
Report	Yes
Date	Yes
Conversion Function	No
Join and Cartesian Product	Yes
Views	Yes
Index	Yes
Aggregation and Group By	Yes
Regular Expression	Yes
Constraints	Yes

SubQuery	Yes
Order By	No
Distinct	Yes