**Screen 1: User Login Screen**

**Task 1: Login**

select \* from user where user.Username = "in\_username" and user.Password = "in\_pass" and user.Status = "Approved");

**Screen 3: Register User**

**Task 1 - Username doesn’t already exist**

select exists(select \* from user where user.username = "$inuser"); #Must return 0

**Task 2 - Confirm Password and Password Match**

select exists(select \* from user where user.username = "$inuser"); #Must return 0

**Task 3 - Register User**

insert into user values ("$in\_user","$in\_pass","$Pending","$Fname","$LName");

insert into UserEmails values ("$in\_user","$email");

**Screen 4: Register Visitor**

**Task 1 - Username doesn’t already exist**

select exists(select \* from user where user.username = "$inuser"); #Must return 0

**Task 2 - Confirm Password and Password Match**

select exists(select \* from user where user.username = "$inuser"); #Must return 0

**Task 3 - Insert**

insert into user values ("$in\_user","$in\_pass","$Pending","$Fname","$LName");

insert into UserEmails values ("$in\_user","$email");

Insert into visitor values (“$in\_visi”);

**Screen 5: Register Employee**

**Task 1 - Username doesn’t already exist**

select exists(select \* from user where user.username = "$inuser"); #Must return 0

**Task 2 - Confirm Password and Password Match**

select exists(select \* from user where user.username = "$inuser"); #Must return 0

**Task 3 - Insert**

insert into user values ("$in\_user","$in\_pass","$Pending","$Fname","$LName");

insert into UserEmails values ("$in\_user","$email");

insert into employee values ("$inuser", max(employee.EmployeeID)+1,"$PhoneNum","$Address","$City","$State","$Zipcode","$EmployeeType");

**Screen 6 Register Employee-Visitor**

**Task 1 - Username doesn’t already exist**

select exists(select \* from user where user.username = "$inuser"); #Must return 0

**Task 2 - Confirm Password and Password Match**

select exists(select \* from user where user.username = "$inuser"); #Must return 0

**Task 3 - Insert**

insert into user values ("$in\_user","$in\_pass","$Pending","$Fname","$LName");

insert into UserEmails values ("$in\_user","$email");

insert into employee values ("$inuser", max(employee.EmployeeID)+1,"$PhoneNum","$Address","$City","$State","$Zipcode","$EmployeeType");

Insert into visitor values (“$in\_visi”);

**Screen 15: User Take Transit**

**Task 1: Selecting which transit to take**

select transit.Transit\_type,transit.route,transit.Price,count(transitconnectssites.SiteName)

from transit join transitconnectssites on transit.Transit\_type = transitconnectssites.TransitType and transitconnectssites.TransitRoute = transit.Route

#where transit.Price Between "$Price1" and "$Price2"

where transitconnectssites.SiteName = "$SiteName"

#and transit.Transit\_type = "$Tranist\_type"

#and transit.Route = "$Route"

group by transit.Transit\_type,transit.Route;

**Task 2:Updating the user take transit table**

Insert into usertaketransit values ("$Username","$Transit\_type","$Route","$TransitDate");

**Screen 16: View Transit History**

**Task 1: Display user’s Transit History**

select usertaketransit.Transit\_Date, usertaketransit.Route, usertaketransit.Transit\_type, transit.Price

from UserTakeTransit

join Transit on usertaketransit.Route = transit.Route and usertaketransit.Transit\_type = transit.Transit\_type

join TransitConnectsSites on usertaketransit.Route = transitconnectssites.TransitRoute and usertaketransit.Transit\_type transitconnectssites.TransitType

where

usertaketransit.username = "!!!username"

#utt.Transit\_Date between "$Start\_date" and "$End\_date"

#and utt.Route = "Route"

#and utt.Transit\_type = "Transit\_type"

group by transit.Route, transit.Transit\_type,usertaketransit.Transit\_Date;

**Screen 17: Manage Employee Profile**

**Task 1: Update Profile**

#Replace an email

UPDATE UserEmails SET email = "$in\_user" WHERE username = "$in\_user" and email = "$in\_user";

#Add a new email

INSERT INTO UserEmails (Email, Username) VALUES ("$in\_user", "$in\_user");

#Update Phone Number

UPDATE Employee Set Phone = "$in\_user" WHERE username = "$in\_user";

#Update First Name

UPDATE User Set FirstName = "$in\_user" WHERE username = "$in\_user";

#Update Last Name

UPDATE User Set LastName = "$in\_user" WHERE username = "$in\_user";

#Drop Visitor Account

DELETE FROM Visitor WHERE Username = "$in\_user";

**Screen 18: Administrator Manage User**

**Task 1: Display user details**

select useremails.username, count(\*), user.status, employee.EmployeeType as 'UserType' from useremails

natural join user natural join employee

#where username = 'username\_input'

#where status = 'status\_input'

where employee.EmployeeType = 'User\_Type\_input'

group by username;

**Task 2: Approve Status**

update user

set user.status = "Approved"

where (username = "$input\_username") and status not in ("Approved");

**Task 3: Decline Status**

update user

set user.status = "Declined"

where username = "$input\_username" and status not in ("Approved");

**Screen 19: Administrator Manage Site**

**Task 1: Filter site details**

Select Site.Name, CONCAT(User.FirstName, ' ', User.LastName) AS Manager, Site.OpenEveryday FROM Site JOIN User ON user.username = site.ManagerUsername where "$in\_user" = "$in\_user";

**Task 2: Delete site**

DELETE FROM Site Where Name = "$in\_user";

**Screen 20: Administrator Edit Site**

**Task 1: Update Site Information**

UPDATE Site SET ManagerUsername = "$in\_user", OpenEveryday = "$in\_user", Zipcode = "$in\_user", Name = "$in\_user", Address = "$in\_user" WHERE Name = "$in\_user";

**Screen 21: Administrator Create Site**

**Task 1: Create Site**

select e.username from Employee as E left outer join Site as S on E.Username = S.ManagerUsername

where e.username not in (select managerusername from site)

and e.employeetype = 'Manager';

#where e.username not in (select managerusername from site);

insert into site

values ("Name","Address","Zipcode","OpenEveryday","ManagerUsername")

**Screen 22: Administrator Manage Transit**

**Task 1: Select Transit**

create or replace view `Transit\_Logged` as Select transit.route, transit.transit\_type, transit.price,count(\*) as Transit\_Logged from usertaketransit

join transit on transit.Route = usertaketransit.Route and transit.Transit\_type = usertaketransit.Transit\_type

group by transit.Route,transit.Transit\_type;

create or replace view `Connected Sites` as select transit.route, transit.Transit\_type, count(\*) as Connected\_Sites from transitconnectssites

join transit on transitconnectssites.TransitRoute = transit.route and transit.Transit\_type = transitconnectssites.TransitType

group by transit.route, transit.Transit\_type;

select `Transit\_Logged`.route,`Transit\_Logged`.transit\_type,`Transit\_Logged`.price,`Connected Sites`.Connected\_Sites,`Transit\_Logged`.Transit\_Logged from `Transit\_Logged`

join `Connected Sites` on `Transit\_Logged`.route = `Connected Sites`.route and `Transit\_Logged`.transit\_type = `Connected Sites`.transit\_type

join transitconnectssites as tcs on tcs.TransitType = `Transit\_Logged`.transit\_type and tcs.TransitRoute = `Transit\_Logged`.route

#where

#tcs.SiteName = "$input SiteName"

#and `Transit\_Logged`.price between "$Low Pricec" and "$High Price"

#and `Transit\_Logged`.route = "$input\_route"

#and `Transit\_Logged`.transit\_type = "$input\_Transittype"

group by `Transit\_Logged`.route,`Transit\_Logged`.transit\_type;

**Task 2: Update Transit**

**update transit**

set #input what we want to change

where transit.transit\_type = "$input\_type" and transit.route = "$input\_route"

**Task 3: Delete Transit**

Delete from transit where transit.transit\_type = "$input\_type" and transit.route = "$input\_route"

**Screen 23: Administrator Edit Transit:**

**Task 1: Edit Route Information**

select sitename from transitconnectssites where transittype = "MARTA" and transitroute = "Blue";

update transit

set Route = "Route\_Input", Price = "Price\_Input"

where Transit\_type = "Transit\_type\_Input";

insert into TransitConnectsSites

values ("SiteName\_Input","Transport\_type","Transit\_Route");

delete from transitconnectssites

where SiteName = "SiteName\_Input";

**Screen 24: Administrator Create Transit:**

**Task 1: Create Transit**

#TransportType must = "MARTA", "Bus", or "Bike

insert into TransitConnectsSites(SiteName,TransitType,TransitRoute)

values

("SiteName\_Input","Transport\_type","Transit\_Route"),

("SiteName\_Input","Transport\_type","Transit\_Route");

#("SiteName\_Input","Transport\_type","Transit\_Route");

**Screen 25: Manager Manage Event:**

**Task 1: Filter Event Information**

CREATE OR REPLACE VIEW StaffCount AS

SELECT EventName, CONCAT(EventName, EventStartDate), COUNT(StaffUsername) AS StaffCount FROM EventStaff GROUP BY CONCAT(EventName, EventStartDate), EventName;

#Create View 2

CREATE OR REPLACE VIEW TotalVisits AS

SELECT EventName, CONCAT(EventName, EventStartDate) AS VKey, COUNT(VisitorUsername) AS VisitCount FROM VisitEvent GROUP BY CONCAT(EventName, EventStartDate), EventName;

#Create View 3

CREATE OR REPLACE VIEW EventV AS

SELECT \*, CONCAT(Name, StartDate) AS EKey FROM Event;

#Filter Events

SELECT DISTINCT eventV.Name, (EndDate-StartDate) AS Duration, (VisitCount \* Price) AS Revenue, VisitCount, StaffCount

FROM EventV JOIN TotalVisits On TotalVisits.Vkey= EventV.EKey

JOIN Site ON Site.Name = EventV.SiteName

JOIN StaffCount on StaffCount.EventName = EventV.Name

where Site.ManagerUsername = '$in\_user'

#and EventV.Name like '%tour%'

#and EventV.Description like '%ADA%'

#and EventV.StartDate = "$in\_user"

#and EventV.EndDate = "$in\_user"

#and EventV.EndDate - EventV.StartDate between "$in\_user" and "$in\_user"

#and TotalVisits.VisitCount between "$in\_user" and "$in\_user"

#and EventV.VisitCount\* EventV.Price between "$in\_user" and "$in\_user";

**Screen 26: Manager View/Edit Event:**

**Task 1: Create the Bottom View Box**

set @StartDate = "INPUT\_DATE";

set @EventName = "EVENT\_NAME";

set @LowDailyVisit = "INPUT LOW";

set @HighDailyVisit = "Input High";

set @LowDailyRev = "INPUT LOW";

set @HighDailyRev = "INPUT HIGH";

Select

Eventstartdate,

count(\*) as Daily\_Visits,

count(\*) \* (select price from event where name = @EventName and startdate = @StartDate) as Daily\_Revenue\_$

from visitevent

where visitevent.eventname = @EventName

and Eventstartdate = @StartDate

group by eventstartdate;

#having count(\*) between @LowDailyVisit and @HighDailyVisit;

#and count(\*) \* (select price from event where name = "Bus Tour" and startdate = '2019-02-01') between @LowDailyRev and @HighDailyRev;

**Task 2: View Event Info**

select event.name,event.price,event.startdate,event.enddate,event.MinStaffReq,Capacity,description from event

where name = @EventName and startdate = @StartDate;

**Task 3: Display staff working event**

select staffusername from eventstaff

where eventname = @EventName and eventstartdate = @StartDate;

**Task 4: Display Staff not working event**

select username from employee where username not in (select staffusername from eventstaff

where eventname = @EventName and eventstartdate = @StartDate)

and EmployeeType = "staff";

**Task 5: Update Staff Working event**

insert into eventstaff values ("$StaffUSername","EventNAme","EventStartDate","SiteName");

**Task 6: Update Event Description**

update event

set description = "$New Description"

where name = @EventName and startdate = @StartDate;

**Screen 27: Manager Create Event:**

**Task 1: Create Event**

Insert into Event

values ("$Name","$Start\_Date","Manager'sAssignedSite","$End\_Date","$Price","$Capacity","$MinStaffRequired","$Description");

select u.FirstName, u.LastName from user u left outer join EventStaff es on u.Username = es.StaffUsername

where u.Username not in

(select es.StaffUsername from EventStaff es join Event e on es.EventName = e.Name

where e.EndDate > "$Start\_Date"

group by es.StaffUsername);

Insert into EventStaff

values ("$StaffUsername","$EventName","$EventStartDate","Manager'sAssignedSite");

**Screen 28: Manager Manage Staff:**

**Task 1: Display Staff**

select concat(u.Firstname,' ', u.LastName) as 'Staff Name', count(\*) as '# Event Shifts' from user u join EventStaff es on u.username = es.staffusername

where es.EventStartDate between '2019-02-07' and '2019-02-09'

#and u.Firstname = 'Michael'

#and u.LastName = 'Smith'

group by u.Firstname, u.LastName;

**Screen 29: Manager Site Report:**

**Task 1: Calculate summary information for each date**

**Task 2: Filter by date and other criterion**

**Screen 30: Manager Daily Detail:**

**Task 1: View Daily Detail**

#SCREEN30

#View 1

CREATE OR REPLACE VIEW DayView AS

SELECT Name FROM Event WHERE '2019-02-01' between StartDate and EndDate;

#View 2

CREATE OR REPLACE VIEW TotalVisits AS

SELECT EventName, CONCAT(EventName, EventStartDate) AS VKey, COUNT(VisitorUsername) AS VisitCount FROM VisitEvent GROUP BY CONCAT(EventName, EventStartDate), EventName;

#View 3

CREATE OR REPLACE VIEW EventV AS

SELECT \*, CONCAT(Name, StartDate) AS EKey FROM Event;

#View 4

CREATE OR REPLACE VIEW Staff AS

SELECT CONCAT(User.FirstName, ' ', User.LastName) AS StaffName, CONCAT(EventStaff.EventName, EventStaff.EventStartDate) As SKey FROM EventStaff join User ON User.Username = EventStaff.StaffUsername GROUP BY CONCAT(EventStaff.EventName, EventStaff.EventStartDate), CONCAT(User.FirstName, ' ', User.LastName);

SELECT EventV.Name, VisitCount, (VisitCount \* Price) AS Revenue, GROUP\_CONCAT(Staff.StaffName) AS StaffNames

FROM EventV JOIN DayView ON DayView.Name = EventV.Name

JOIN TotalVisits on EventV.EKey = TotalVisits.VKey

JOIN Staff on Staff.SKey = EventV.EKey

JOIN Site on Site.Name = EventV.SiteName

WHERE Site.ManagerUsername = '$in\_user'

GROUP BY EventV.Name, VisitCount, Revenue;

**Screen 31: Staff View Schedule:**

**Task 1: Filter/View Staff View**

select e.name as 'Event Name',e.SiteName as 'Site Name', e.StartDate as 'Start Date',e.EndDate as 'End Date', count(es.StaffUsername) as 'Staff Count'

from event as e join eventstaff as es

on e.startdate = es.EventStartDate and e.name = es.EventName and e.SiteName = es.SiteName

#where e.Description like '%KEYWORD%'

#where e.name = 'EVENT NAME'

#where e.StartDate = 'STARTDATE'

#where e.EndDate = 'ENDDATE'

group by es.eventname, es.eventstartdate, es.sitename;

**Screen 32: Staff Event Detail:**

**Task 1: View Staff Event detail**

select e.name, e.SiteName, e.StartDate, e.EndDate, e.EndDate - e.StartDate as 'Duration days', es.StaffUsername as 'Staff Name', e.Description, e.capacity, e.Price

from event as e join eventstaff as es on e.Name = es.EventName

and e.SiteName = es.SiteName

and e.StartDate = es.EventStartDate

order by es.StaffUsername;

**Screen 33: Visitor Explore Event:**

**Task 1: Visitor view filter table**

create or replace view `EventVisits` as select event.name, event.SiteName, event.price, count(\*) as NumVisits, event.capacity - count(\*) as TicketRem,event.startdate from event

left join visitevent on event.name = visitevent.eventname and event.startdate = visitevent.EventStartDate

#where visitevent.VisitDate between "low" and "high"

group by event.name,event.startdate;

create or replace view `MyVisitsToEvent` as select VisitorUsername, eventname, eventstartdate, SiteName, count(\*) as MyVisits from VisitEvent

#where VisitDate between @StartDate and @EndDate

where visitorUsername = "mary.smith"

group by visitevent.eventname, visitevent.EventStartDate;

create or replace view `EventsInDateRange` as select `EventVisits`.name, `EventVisits`.Sitename, `EventVisits`.price, `EventVisits`.NumVisits, `EventVisits`.TicketRem, ifnull(`MyVisitsToEvent`.MyVisits,0) as MyVisits

from `EventVisits`

left join `MyVisitsToEvent` on `EventVisits`.name = `MyVisitsToEvent`.eventname and `EventVisits`.startdate = `MyVisitsToEvent`.eventstartdate;

select \* from `EventsInDateRange`;

**Screen 34: Visitor Event Detail:**

**Task 1: Visitor View Event Detail**

select event.name,event.SiteName,event.startdate,event.enddate,event.price,`EventsInDateRange`.Ticketrem from event

join `EventsInDateRange` on event.name = `EventsInDateRange`.name

where Event.Name = "$EventName" and event.startdate = "$Startdate";

**Task 2: Log Visit**

insert into visitevent

values ("VisitorUsername","EventNAme","EventStartDate","SiteNAme","VisitDate");

**Screen 35: Visitor Explore Site:**

**Task 1: List out sites and summary statistics for each site**

create or replace view `EventVisits` as select SiteName, count(\*) as NumVisits from visitevent

#where VisitDate between "StartDAte" and "EndDate"

group by Sitename;

create or replace view `SiteVisits` as select SiteName,count(\*) as SiteVisits from visitSite

#where VisitDate between "StartDAte" and "EndDate"

group by siteName;

create or replace view `EventCount` as select Site.Name,count(\*) as EventCount from event

join Site on Site.Name = Event.SiteName

#where VisitDate between "StartDAte" and "EndDate"

group by SiteName;

create or replace view `MyVisitsToSite` as Select VisitorUsername,SiteName,count(\*) as MyVisits from visitsite

#where VisitDate between "StartDAte" and "EndDate"

where visitorusername = "visitor1"

group by VisitorUsername, SiteName;

create or replace view `MyVisitsToEvent` as select VisitorUsername,SiteName, count(\*) as MyVisits from VisitEvent

#where VisitDate between "StartDAte" and "EndDate"

where visitorUsername = "visitor1"

group by VisitorUsername, SiteName;

select Site.Name, ifnull(`EventCount`.EventCount,0) as 'Event Count',ifnull(`EventVisits`.numvisits,0)+ifnull(`SiteVisits`.SiteVisits,0) as 'Total Visits',ifnull(`MyVisitsToEvent`.MyVisits,0)+ifnull(`MyVisitsToSite`.MyVisits,0) as 'My Visits' from Site

left join `MyVisitsToEvent` on `MyVisitsToEvent`.SiteName = Site.Name

left join `MyVisitsToSite` on `MyVisitsToSite`.SiteName = Site.Name

left join `EventCount` on Site.name = `EventCount`.Name

left join `EventVisits` on Site.name = `EventVisits`.SiteName

left join `SiteVisits` on Site.name = `SiteVisits`.SiteName;

**Screen 36: Visitor Transit Detail:**

**Task 1: List transits connected to current site**

select t.Route,t.Transit\_type,t.Price,count(tcs.SiteName)

from transit as t

join transitconnectssites as tcs on t.Transit\_type = tcs.TransitType and t.Route = tcs.TransitRoute

#where tcs.TransitRoute = 'blah'

**Task 2: Log Transit**

#Insert into usertaketransit values ('user','transit type', 'route','transitdate')

**Screen 37: Visitor Site Detail:**

**Task 1: Log visit**

insert into VisitSite

values ("\$Username"\,"SiteName",\"$Visit\_Date"\);

**Screen 38: Visitor Visit History:**

**Task 1: Visitor Filter Visit History**

select vs.VisitDate as 'Date',ve.EventName as 'Event',vs.SiteName as 'Site',t.price as 'Price' from visitevent as ve, visitsite as vs,

transit as t join transitconnectssites as tcs

where

#tcs.SiteName =

#vs.VisitDate between "Start\_date" and "End\_date"

#group by vs.VisitDate,ve.EventName,vs.SiteName,t.price