

Project Notebook

Restaurant Finder

Group 3

Rucha Nimbalkar

Section 1 >Project Proposal

1] Application Description

The application is for a website which offers customers the facility to find restaurants near them. The customers can see the average rating, hours of operation and view the menu. Users can also rate a restaurant, write a review.

2] Problem Components

The application should have a simple UI to provide following functionalities.

Functionalities:

Search for a restaurant based on parameters

The user can search a restaurant by giving parameters like price range, Zip, and cuisine.

Rate a Restaurant

Users can rate a restaurant

Write a Review/View Reviews

Users can write reviews for a restaurant they have visited.

View Menu

Users can view the Menu.

3] Extra functionalities

- 1) Web based application
- 2) Register a user
- 3) Rating by user (User can rate)
- 4) Admin will review the User Review and comments and only publish them if they are appropriate

4] Solution

To create a simple UI using java jdbc swing and connect it to mysql and implement the functionalities mentioned above. This will solve the problem.

5] Milestones: Date Plan

Come up with a proposal Nov 5 - 2015

Initial ideas were apartment management, online shopping system but finally after one of my classmates suggested to do something that I can show on my resume. Initial work in Restaurant Finder App was the motivation to come up with a restaurant finder app proposal.

Do the database Design. Draw ERD . Nov 18 -2015

Draw the initial design of the entity relationship diagram. Come up with Entities from the relations, draw the relationships .

Form the relations using ERD and normalize them in 4nf. Nov 20 -2015

Create the database using the relations in 4nf. Nov 23 - 2015

Use datasets to fill in the relations/tables Nov 24 - 2015

Finalize the technology Nov 25 - 2015

Do the Swing GUI to create simple UI.

Connect the database

Start Coding and complete at least 3 functionalities Dec 7 - 2015

Write queries to implement functionalities

Complete the working project Dec 13 - 2015

6] Test Cases

- 1] Enter cuisine and zip to find restaurants.
- 2] Add a restaurant
- 3] Update rating, address, zip or restaurant name
- 4] Delete a restaurant from the database
- 5] View Reviews
- 6] Admin can only add, update or delete . User can be a admin .

7] Example Questions

Are there any American cuisine restaurants in 98402?

Are there any Thai cuisine restaurants in 98402?

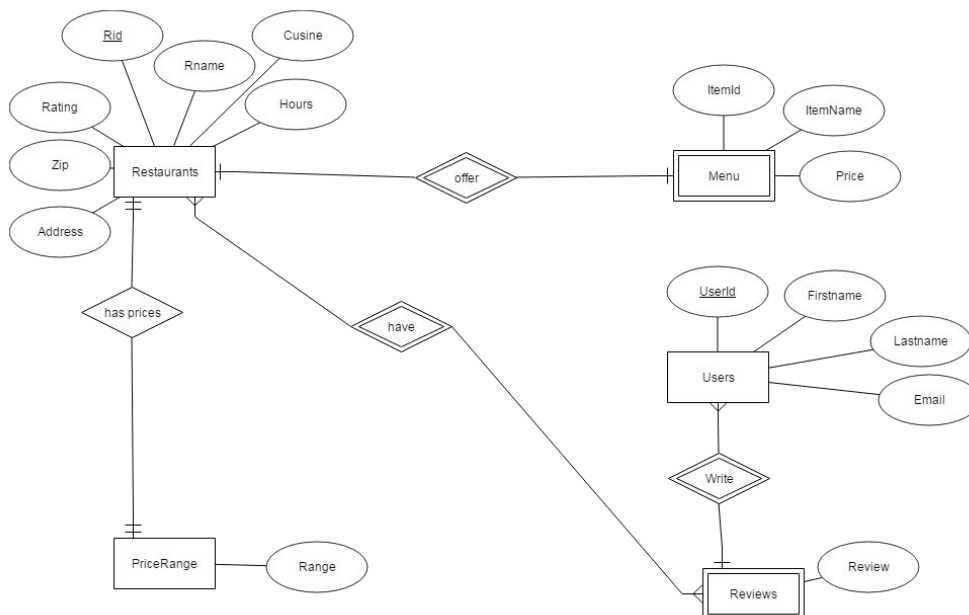
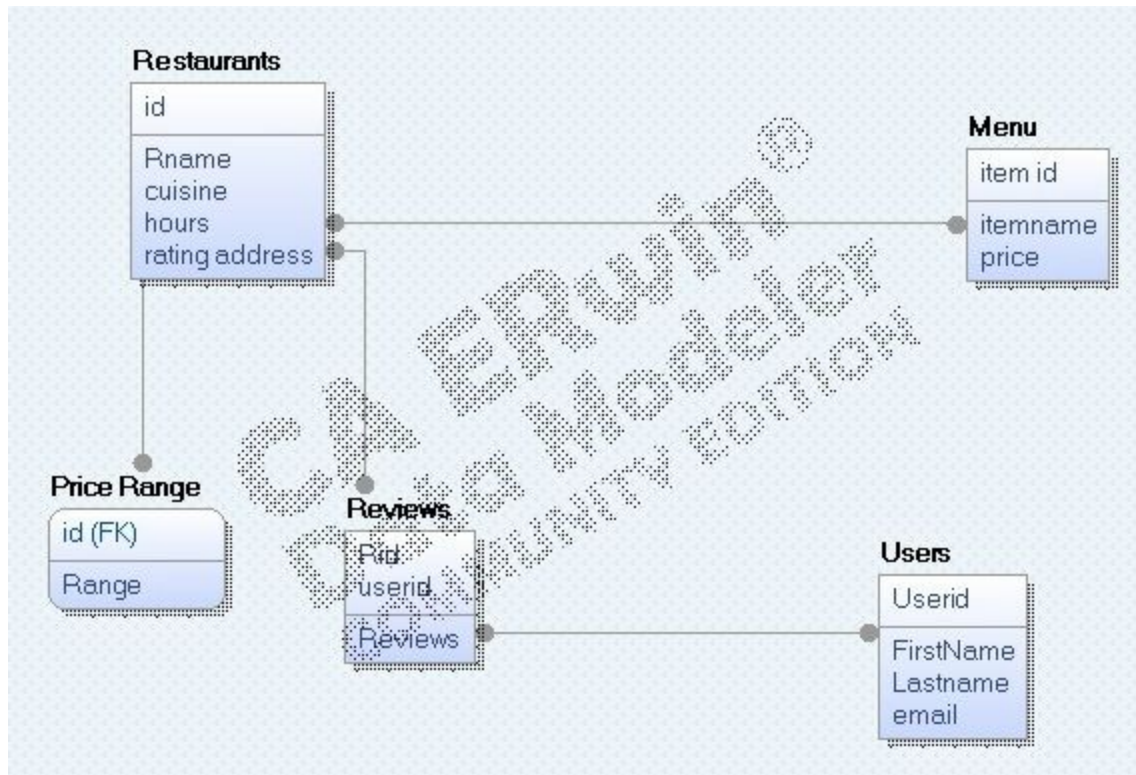
Can Users view the reviews for a restaurant?

Can admin add a restaurant?

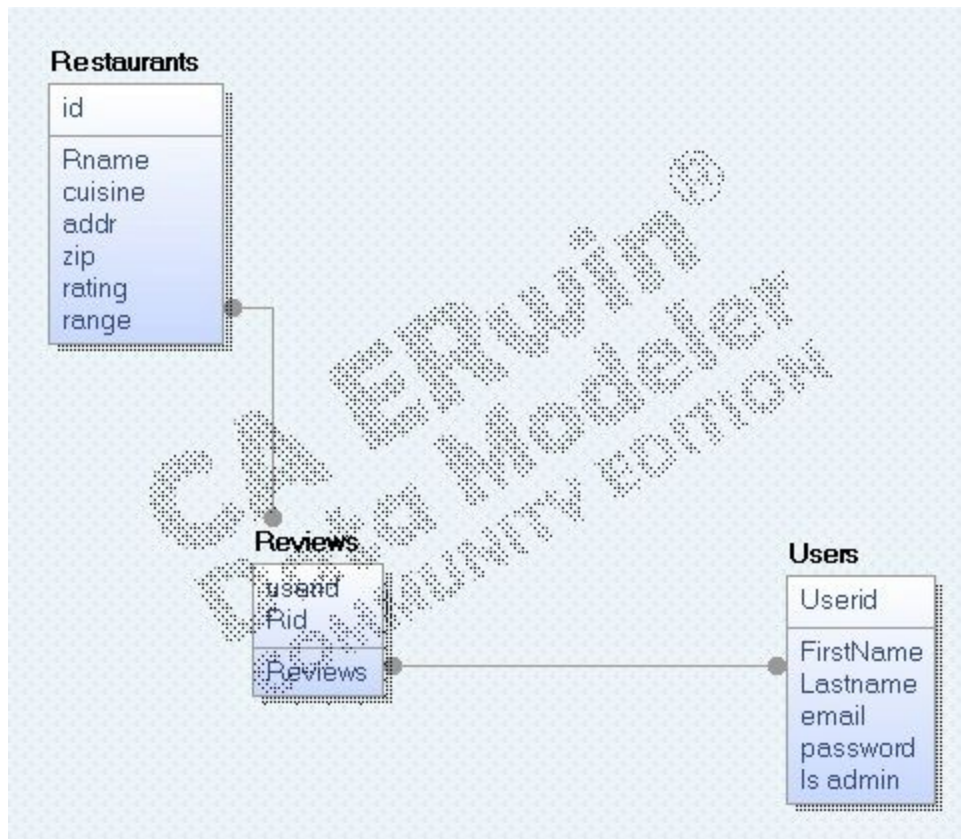
Can admin delete a restaurant?

Can admin update a restaurant?

Section 2> Data Model Section:



Final ERD:



Section 3> Database Schema Section:

Users (UserID, First Name, Last Name, Email)

Restaurants (Rid, R_Name, Cuisine, Hours, Address, Zip, Rating)

Menu (Itemid, ItemName, Price)

Reviews (UserId, Rid, Review)

PriceRange (Rid, Range)

TableName	Table Description
Users	UserId, Fname, Lname, Email
Restaurants	Rid, R_Name, Cuisine, Hours, Address, Zip, Rating
Reviews	UserId, Rid, Review
Menu	ItemId, Itemname, Price
Userdata	Userid, Pwd
PriceRange	Rid,Range

After, some research and findings, I found out that the Hours attribute can be of variable length hence, it cannot be included in the entity Restaurants.

The entity Menu has attributes Item Id, Item Name and Price.

As per our assumption each restaurant offers one menu but a menu has multiple items.

This can be implemented as Menu (Rid, Item Id, Name, Price), but then it will have multiple elements for each item and same Rid. There is another way of implementing it, wherein we will use multiple tables of menu with menuid or having rid and menu containing description of entire menu but we would lose attribute price and it will also have redundant data of Rid.

So the menu is dropped and it will be displayed using some other method.

As for now, the objective is to display Restaurants as per the requirements.

The functionalities or operations like add, update and delete can be done only by admin . A user can be either admin or not an admin. User will have to log in to perform this operations and hence he should have a password.

Final Relations:

Restaurants(Rid,Rname,cuisine,address,zip,rating,range)

Users(Userid,Firstname,Lastname,password, Email,Is admin)

Reviews(Rid,UserId,Review)

Section 4> Populate the database:

```
CREATE DATABASE IF NOT EXISTS `restaurant_finder_database` ;
```

```
USE `restaurant_finder_database`;
```

Restaurants:

```
INSERT INTO restaurants (id, r_name, cuisine, addr, zip, rating, r_range) VALUES
(984021,'Pacific Grill','American','1502 Pacific Ave, Tacoma, WA','98402',4.4,'high'),
(984022,'IndoChine Asian Dining Hall','Indonesian','1924 Pacific Ave, Tacoma,
WA','98402',4.0999999904632568,'High'),
(984023,'The Old Spaghetti Factory','Italian','1735 Jefferson Ave, Tacoma, WA
','98402',3.799999952316284,'Med'),
(984024,'Abella Pizzeria','Italian','1946 Pacific Ave, Tacoma, WA
','98402',4.199999809265137,'Low'),
(984025,'Galanga Thai Cuisine','Thai','1129 Broadway, Tacoma,
WA','98402',4.0999999904632568,'Med'),
(984026,'The Rock Wood Fired Kitchen','Italian','1920 Jefferson Ave, Tacoma, WA
','98402',4.1,'High'),
(984061,'Gateway To India','Indian','2603 6th Ave, Tacoma,
WA','98406',4.199999809265137,'High'),
(984091,'Chili Thai Restaurant','Thai','3213 S 38th St, Tacoma, WA','98409',4.7,'High'),
(984661,'India Mahal','Indian','1905 Bridgeport Way W #110, University
Place','98466',4.000000003,'Med'),
(984991,'Great India Cuisine','Indian','9518 Gravelly Lake Dr SW, Lakewood,
WA','98499',3.8,'Med'),
```

(984992,'Cham Garden Korean BBQ','Korean','10518 S Tacoma Way, Lakewood, WA','98499',4.6,'Med'),

(984993,'Bubble Grill','American','Tacoma Ave','98499',4.2,'low'),

(984995,'Swiss Pub','Swiss','19th Jefferson St','98402',4.2,'high'),

(984996,'Teriyaki','Japanese','tacoma Ave','98499',4.9,'low'),

(984997,'Bombay Garden','Indian','4th, Jackson','98499',3.2,'low'),

(984998,'Panda Express','Thai','Tacoma Ave','98499',4.5,'high'),

(984999,'AppleBees','American','Cedar St','98499',4.9,'med');

<u>id</u>	<u>r_name</u>	<u>cuisine</u>	<u>addr</u>	<u>zip</u>	<u>rating</u>	<u>r_range</u>
<u>984021</u>	<u>Pacific Grill</u>	<u>American</u>	<u>1502 Pacific Ave, Tacoma, WA</u>	<u>98402</u>	<u>4.4</u>	<u>high</u>

<u>984022</u>	<u>IndoChine Asian Dining Hall</u>	<u>Indonesian</u>	<u>1924 Pacific Ave, Tacoma, WA</u>	<u>98402</u>	<u>4.1</u>	<u>High</u>
<u>984023</u>	<u>The Old Spaghetti Factory</u>	<u>Italian</u>	<u>1735 Jefferson Ave, Tacoma, WA</u>	<u>98402</u>	<u>3.8</u>	<u>Med</u>
<u>984024</u>	<u>Abella Pizzeria</u>	<u>Italian</u>	<u>1946 Pacific Ave, Tacoma, WA</u>	<u>98402</u>	<u>4.2</u>	<u>Low</u>
<u>984025</u>	<u>Galanga Thai Cuisine</u>	<u>Thai</u>	<u>1129 Broadway, Tacoma, WA</u>	<u>98402</u>	<u>4.1</u>	<u>Med</u>
<u>984026</u>	<u>The Rock Wood Fired Kitchen</u>	<u>Italian</u>	<u>1920 Jefferson Ave, Tacoma, WA</u>	<u>98402</u>	<u>4.1</u>	<u>High</u>
<u>984061</u>	<u>Gateway To India</u>	<u>Indian</u>	<u>2603 6th Ave, Tacoma, WA</u>	<u>98406</u>	<u>4.2</u>	<u>High</u>
<u>984091</u>	<u>Chili Thai Restaurant</u>	<u>Thai</u>	<u>3213 S 38th St, Tacoma, WA</u>	<u>98409</u>	<u>4.7</u>	<u>High</u>
<u>984661</u>	<u>India Mahal</u>	<u>Indian</u>	<u>1905 Bridgeport Way W #110, University Place</u>	<u>98466</u>	<u>4</u>	<u>Med</u>

<u>984991</u>	<u>Great India Cuisine</u>	<u>Indian</u>	<u>9518 Gravelly Lake Dr SW, Lakewood, WA</u>	<u>98499</u>	<u>3.8</u>	<u>Med</u>
<u>984992</u>	<u>Cham Garden Korean BBQ</u>	<u>Korean</u>	<u>10518 S Tacoma Way, Lakewood, WA</u>	<u>98499</u>	<u>4.6</u>	<u>Med</u>
<u>984993</u>	<u>Bubble Grill</u>	<u>American</u>	<u>Tacoma Ave</u>	<u>98499</u>	<u>4.2</u>	<u>low</u>
<u>984995</u>	<u>Swiss Pub</u>	<u>Swiss</u>	<u>19th Jefferson St</u>	<u>98402</u>	<u>4.2</u>	<u>high</u>
<u>984996</u>	<u>Teriyaki</u>	<u>Japanese</u>	<u>tacoma Ave</u>	<u>98499</u>	<u>4.9</u>	<u>low</u>
<u>984997</u>	<u>Bombay Garden</u>	<u>Indian</u>	<u>4th, Jackson</u>	<u>98499</u>	<u>3.2</u>	<u>low</u>
<u>984998</u>	<u>Panda Express</u>	<u>Thai</u>	<u>Tacoma Ave</u>	<u>98499</u>	<u>4.5</u>	<u>high</u>
<u>984999</u>	<u>AppleBees</u>	<u>American</u>	<u>Cedar St</u>	<u>98499</u>	<u>4.9</u>	<u>med</u>

Reviews:

INSERT INTO reviews (UserId, RestaurantId, Reviewscol)

VALUES

('asina',984021,'Great Place, Great food'),

('chuckb',984021,'A little lower to my tastes.'),

('shelco',984021,'Rude Host. Food is great though'),

('zaynm',984021,'Will never visit again.'),
 ('shelco',984022,'Awesome food'),
 ('zaynm',984022,'A little pricey, but worth it.'),
 ('cmassina',984023,'You call it Italian. bad'),
 ('cmassina',984024,'Simple and economic. Crowded sometimes'),
 ('ellew',984025,'Tasty food, great price.'),
 ('noel',984025,'Yummy Thai food. Quantity is good'),
 ('hanmarin',984026,'Very Good Hospitality.'),
 ('ksweet',984026,'Finger licking good food. Warm host'),
 ('joel',984061,'Worth the wait and price'),
 ('joshel',984061,'not good'),
 ('ksweet',984061,'They did not play soft kitty. Food was okay.'),
 ('matd',984061,'Nice'),
 ('chuckb',984091,'good place to hangout'),
 ('matd',984091,'Good. Will visit soon'),
 ('mkelly',984091,'Yummy Indian Food'),
 ('noel',984091,'Loved it');

<u>UserId</u>	<u>RestaurantId</u>	<u>Reviewscol</u>
<u>asina</u>	<u>984021</u>	<u>Great Place, Great food</u>
<u>chuckb</u>	<u>984021</u>	<u>A little lower to my tastes.</u>
<u>shelco</u>	<u>984021</u>	<u>Rude Host. Food is great though</u>
<u>zaynm</u>	<u>984021</u>	<u>Will never visit again.</u>

<u>shelco</u>	<u>984022</u>	<u>Awesome food</u>
<u>zaynm</u>	<u>984022</u>	<u>A little pricey, but worth it.</u>
<u>cmassina</u>	<u>984023</u>	<u>You call it Italian. bad</u>
<u>cmassina</u>	<u>984024</u>	<u>Simple and economic.</u> <u>Crowded sometimes</u>
<u>ellew</u>	<u>984025</u>	<u>Tasty food, great price.</u>
<u>noel</u>	<u>984025</u>	<u>Yummy Thai food.</u> <u>Quantity is good</u>
<u>hanmarin</u>	<u>984026</u>	<u>Very Good Hospitality.</u>
<u>ksweet</u>	<u>984026</u>	<u>Finger licking good food.</u> <u>Warm host</u>
<u>joel</u>	<u>984061</u>	<u>Worth the wait and price</u>
<u>joshel</u>	<u>984061</u>	<u>not good</u>
<u>ksweet</u>	<u>984061</u>	<u>They did not play soft kitty.</u> <u>Food was okay.</u>
<u>matd</u>	<u>984061</u>	<u>Nice</u>
<u>chuckb</u>	<u>984091</u>	<u>good place to hangout</u>
<u>matd</u>	<u>984091</u>	<u>Good. Will visit soon</u>
<u>mkelly</u>	<u>984091</u>	<u>Yummy Indian Food</u>
<u>noel</u>	<u>984091</u>	<u>Loved it</u>

Users:

```
INSERT INTO users (userid, firstname, lastname, password, email, is_admin) VALUES
('asina','Adrian','Sina','java','asina@hmail.com',0),
('chuckb','Chuck','Bass','java','chuckb@hmail.com',1),
('cmassina','Chris','Massina','lava','cmass@mproj.com',0),
('ellew','Elle','Woods','java','elwoods@hmail.com',1),
('eval','Eva','Longaria','lava','elong@gmail.com',0),
('hanmarin','Hanna','Marin','java','hmarin@hmail.com',0),
('joel','Joel','Martin','java','joemartin@hmail.com',0),
('joshel','Josh','Hazel','java','Jhazel@hmail.com',0),
('ksweet','Kaley','Sweet','java','kalsweet@hmail.com',0),
('matd','Matt','Damon','lava','md@hmail.com',0),
('mkelly','Mindy','Kelly','lava','mk@theoffice.com',1),
('noel','Noel','Khan','java','nkhan@hmail.com',0),
('rickmrin','Ricky','Marin','java','rickm@hmail.com',0),
('shelco','Shelly','Cooper','java','shelcoop@hmail.com',0),
('zaynm','Zayn','Malik','java','zmalik@hmail.com',1);
```

<u>userid</u>	<u>firstname</u>	<u>lastname</u>	<u>password</u>	<u>email</u>	<u>is_admin</u>
---------------	------------------	-----------------	-----------------	--------------	-----------------

<u>asina</u>	<u>Adrian</u>	<u>Sina</u>	<u>java</u>	<u>asina@hmail.com</u>	<u>0</u>
<u>chuckb</u>	<u>Chuck</u>	<u>Bass</u>	<u>java</u>	<u>chuckb@hmail.com</u>	<u>1</u>
<u>cmassina</u>	<u>Chris</u>	<u>Massina</u>	<u>lava</u>	<u>cmass@mproj.com</u>	<u>0</u>
<u>ellew</u>	<u>Elle</u>	<u>Woods</u>	<u>java</u>	<u>elwoods@hmail.com</u>	<u>1</u>
<u>eval</u>	<u>Eva</u>	<u>Longaria</u>	<u>lava</u>	<u>elong@gmail.com</u>	<u>0</u>
<u>hanmarin</u>	<u>Hanna</u>	<u>Marin</u>	<u>java</u>	<u>hmarin@hmail.com</u>	<u>0</u>
<u>joel</u>	<u>Joel</u>	<u>Martin</u>	<u>java</u>	<u>joemartin@hmail.com</u>	<u>0</u>
<u>joshel</u>	<u>Josh</u>	<u>Hazel</u>	<u>java</u>	<u>Jhazel@hmail.com</u>	<u>0</u>
<u>ksweet</u>	<u>Kaley</u>	<u>Sweet</u>	<u>java</u>	<u>kalsweet@hmail.com</u>	<u>0</u>
<u>matd</u>	<u>Matt</u>	<u>Damon</u>	<u>lava</u>	<u>md@hmail.com</u>	<u>0</u>
<u>mkelly</u>	<u>Mindy</u>	<u>Kelly</u>	<u>lava</u>	<u>mk@theoffice.com</u>	<u>1</u>
<u>noel</u>	<u>Noel</u>	<u>Khan</u>	<u>java</u>	<u>nkhan@hmail.com</u>	<u>0</u>
<u>rickmrin</u>	<u>Ricky</u>	<u>Marin</u>	<u>java</u>	<u>rickm@hmail.com</u>	<u>0</u>
<u>shelco</u>	<u>Shelly</u>	<u>Cooper</u>	<u>java</u>	<u>shelcoop@hmail.com</u>	<u>0</u>
<u>zaynm</u>	<u>Zayn</u>	<u>Malik</u>	<u>java</u>	<u>zmalik@hmail.com</u>	<u>1</u>

Section 5> Example Queries Section:

```
use restaurant_finder_database;
```

```
/*This is the sql query for the search button*/
```

```
/* To get all restaurants*/
```

```
select * from restaurants;
```

```
/* To get restaurants from zip and cuisine*/
```

```
select * from restaurants where zip = 98402 and cuisine = "American";
```

```
/*This is the sql query for the delete button*/
```

```
delete from restaurants where id= 984999;
```

```
select * from restaurants where id = 984999;
```

```
/*This is the sql query for the add button*/
```

```
INSERT INTO restaurants (id,r_name,cuisine,addr, zip, rating, r_range)
```

```
VALUES (984999, "Dawgden", "American", "19th commerce", 98499, 3.1,"low");
```

```
select * from restaurants where id = 984999;
```

/*This is the sql query for the update button*/

update restaurants

set r_name= "Raj", cuisine= "Indian", addr= "4th Pike", zip= 98499, rating= 3.5, r_range= "low"

where id = 984999;

select * from restaurants where id = 984999;

/*This is the sql query for the view reviews button*/

select Reviewscol from reviews where RestaurantId = 984021;

/*Get all Users*/

select * from users order by lastname;

/*Get the current user*/

select * from users where userid="asina" order by lastname;

/*To match the password from the database to entered password*/

select password from users where userid="asina";

Section 6> Example Reports Section:

use restaurant_finder_database;

/*This is the sql query for the search button*/

/* To get all restaurants*/

select * from restaurants;

id	r_name	cuisine	addr	zip	rating	r_range
984021	Pacific Grill	American	1502 Pacific Ave, Tacoma, WA	98402	4.4	high
984022	IndoChine Asian Dining Hall	Indonesian	1924 Pacific Ave, Tacoma, WA	98402	4.1	High
984023	The Old Spaghetti Factory	Italian	1735 Jefferson Ave, Tacoma, WA	98402	3.8	Med
984024	Abella Pizzeria	Italian	1946 Pacific Ave, Tacoma, WA	98402	4.2	Low
984025	Galanga Thai Cuisine	Thai	1129 Broadway, Tacoma, WA	98402	4.1	Med
984026	The Rock Wood Fired Kitchen	Italian	1920 Jefferson Ave, Tacoma, WA	98402	4.1	High
984061	Gateway To India	Indian	2603 6th Ave, Tacoma, WA	98406	4.2	High
984091	Chili Thai Restaurant	Thai	3213 S 38th St, Tacoma, WA	98409	4.7	High
984661	India Mahal	Indian	1905 Bridgeport Way W #110, University Place	98466	4	Med
984991	Great India	Indian	9518	98499	3.8	Med

	Cuisine		Gravelly Lake Dr SW, Lakewood, WA			
984992	Cham Garden Korean BBQ	Korean	10518 S Tacoma Way, Lakewood, WA	98499	4.6	Med
984993	Bubble Grill	American	Tacoma Ave	98499	4.2	low
984995	Swiss Pub	Swiss	19th Jefferson St	98402	4.2	high
984996	Teriyaki	Japanese	tacoma Ave	98499	4.9	low
984997	Bombay Garden	Indian	4th, Jackson	98499	3.2	low
984998	Panda Express	Thai	Tacoma Ave	98499	4.5	high
984999	AppleBees	American	Cedar St	98499	4.9	med

/* To get restaurants from zip and cuisine*/

select * from restaurants where zip = 98402 and cuisine = "American";

<u>id</u>	<u>r_name</u>	<u>cuisine</u>	<u>addr</u>	<u>zip</u>	<u>rating</u>	<u>r_range</u>
<u>984021</u>	<u>Pacific Grill</u>	<u>American</u>	<u>1502 Pacific Ave, Tacoma, WA</u>	<u>98402</u>	<u>4.4</u>	<u>high</u>

/*This is the sql query for the delete button*/

delete from restaurants where id= 984999;

select * from restaurants where id = 984999; /*see the table to check the deletion*/

id	r_name	cuisine	addr	zip	rating	r_range

/*This is the sql query for the add button*/

INSERT INTO restaurants (id,r_name,cuisine,addr, zip, rating, r_range)

VALUES (984999, "Dawgden", "American", "19th commerce", 98499, 3.1,"low");

select * from restaurants where id = 984999;

<u>id</u>	<u>r_name</u>	<u>cuisine</u>	<u>addr</u>	<u>zip</u>	<u>rating</u>	<u>r_range</u>
<u>984999</u>	<u>Dawgden</u>	<u>American</u>	<u>19th commerce</u>	<u>98499</u>	<u>3.1</u>	<u>low</u>

/*This is the sql query for the update button*/

update restaurants

set r_name= "Raj", cuisine= "Indian", addr= "4th Pike", zip= 98499, rating= 3.5, r_range= "low"

where id = 984999;

select * from restaurants where id = 984999;

id	r_name	cuisine	addr	zip	rating	r_range
984999	Raj	Indian	4th Pike	98499	3.5	low

/*This is the sql query for the view reviews button*/

```
select Reviewscol from reviews where RestaurantId = 984021;
```

Reviewscol
Great Place, Great food
A little lower to my tastes.
Rude Host. Food is great though
Will never visit again.

/*Get all Users*/

```
select * from users order by lastname;
```

userid	firstname	lastname	password	email	is_admin
chuckb	Chuck	Bass	java	chuckb@hmail.com	1
shelco	Shelly	Cooper	java	shelcoop@hmail.com	0
matd	Matt	Damon	lava	md@hmail.com	0
joshel	Josh	Hazel	java	Jhazel@hmail.com	0
mkelly	Mindy	Kelly	lava	mk@theoffice.com	1
noel	Noel	Khan	java	nkhan@hmail.com	0
eval	Eva	Longaria	lava	elong@gmail.com	0
zaynm	Zayn	Malik	java	zmalik@hmail.com	1

hanmarin	Hanna	Marin	java	hmarin@hmail.com	0
rickmrin	Ricky	Marin	java	rickm@hmail.com	0
joel	Joel	Martin	java	joemartin@hmail.com	0
cmassina	Chris	Massina	lava	cmass@mproj.com	0
asina	Adrian	Sina	java	asina@hmail.com	0
ksweet	Kaley	Sweet	java	kalsweet@hmail.com	0
ellew	Elle	Woods	java	elwoods@hmail.com	1

/*Get the current user*/

select * from users where userid="asina" order by lastname;

userid	firstname	lastname	password	email	is_admin
asina	Adrian	Sina	java	asina@hmail.com	0

/*To match the password from the database to entered password*/

select password from users where userid="asina";

password
java

Section 7> Discussion Section:

The Project was a huge idea and due to time limitations I could not implement it as a web based app where users can log in. I realized this while working and how challenging it was to keep the basic things running. Basic functionalities are maintained and this stands as windows based app. I wanted to improve it by making it a handheld app and cover all functionalities.

The app is works for add, update delete,search and view reviews. Only admin can log in and add, update and delete the restaurant data. Users can search using zip and cuisine type.

Section Bibliography:

I have been comfortable with C and C++ and some knowledge of Java and JDBC. I used the help of online tutorials to work on my project and they were really helpful to design the swing gui.

1] The Java Black Book

Java Black Book Second Edition by Steve Holzner (Author), Steven Holzner (Author)

2] Database Systems The Complete Book

Database Systems: The Complete Book 2nd Edition

by Hector Garcia-Molina (Author), Jeffrey D. Ullman (Author), Jennifer Widom (Author)

3] Luv2code.com