



Great Speed for Great Needs.

Milestone 4(a): Implementation

Course: CMPT 354

Group: 41

July 2nd

#	Student Name (founders)	Student Number	Email Address
1	Tomi Lui	301426310	tomi_lui@sfu.ca
2	Benjamin Smus	301369665	ben_smus@sfu.ca
3	Rithik Ramamurthy	301389571	rramamur@sfu.ca

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above.

In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by Simon Fraser University.

A short description of the final project, and what it accomplished:

In milestone 4, we have built a complete application that has food delivery functionalities.

Technology:

Our team built a full stack application with html, python's Flask API framework and mysql database.

Usage:

- Users can:
 - register accounts by entering their email, address and credit card information
 - Place orders from a restaurant
 - Rate restaurants
- Restaurant owners can
 - register their companies with our app
 - Add dishes to their restaurants
- The app allows you to can view all users in database
- Search for item from a restaurant
- Data validation on client and server side

A description of how your final schema differed from the schema you turned in.

- **Rubric: Differences between the final schema and the proposed schemas are clearly and thoughtfully explained.**

Throughout our planning and development process, we have mostly used the schema that we turned in except for minor changes.

Some changes that we've made:

- Separated Zip from Address into its own table since it can be a repeated value
- Separated city_name from Address into its own table since it can be a repeated value
- Added the auto keyword to timestamp and primary keys. This allows MySQL to handle indexing and creating time more efficiently than inserting these values programmatically.
- Removed functional dependencies from credit card number Bank Identification Number (BIN). BIN is the first 6 values of every credit card number and this value contains information about the credit cards bank provider, credit card type (visa, mastercard), and whether or not it is debit or credit. Since this value can be shared by multiple different credit cards, we have separated Card_BIN6 into its own table from card_all.

- Previously, restaurants were a simple entity but we have changed it so that it must be owned/managed by a manager. For every restaurant in Restaurant table, it now has a foreign key that points to a manager in the User table. This value is provided in the restaurant sign up form.
- At first, we wanted to create tables to contain items (dishes with price) specifically for a restaurant (restaurant menu for users to order from), but we have improved it by creating a single Item table to contain all Items from all restaurants. Each item in Item table has a foreign key that points to a restaurant. Restaurant Items/Menu can be retrieved by simply writing a join query to find which restaurant the item belongs to.

Overall we kept the original schema form. Our improvements led to creating even tables, which made the coding implementation harder because we have to write more checks and insert data to more tables.

Examples of queries used

Selection:

(in structure get_restaurant_id_by_name with parameter restaurant_name)

```
SELECT restaurant_id
FROM Restaurant
WHERE restaurant_name = '{restaurant_name}';
```

Projection:

```
SELECT *
FROM Restaurant;
```

Join query:

(structure view_restaurant_items(restaurant_name))

```
SELECT *
FROM Restaurant, Item
WHERE restaurant_name = '{restaurant_name}' AND Restaurant.restaurant_id = Item.restaurant_id;
```

Aggregation query:

```
SELECT avg(price)
FROM Item;
```

Nested aggregation with group-by:

```
SELECT restaurant_name, MIN(price)
FROM Item, Restaurant
WHERE restaurant_name IN
(SELECT restaurant_name
FROM Restaurant
WHERE Restaurant.restaurant_id = Item.restaurant_id)
GROUP BY restaurant_name;
```

Division query:

```
SELECT u.first_name, u.last_name
```

```
FROM User u
```

```
WHERE NOT EXISTS
```

```
(SELECT * from Restaurant r
```

```
WHERE NOT EXISTS
```

```
(SELECT o.consumer_id
```

```
FROM _Order o
```

```
WHERE u.user_id=o.consumer_id AND r.restaurant_id=o.restaurant_id));
```

SCREENSHOTS:

Before inserting Table:

View Users

user_id	first_name	last_name	email	phone	type
1	Annabel	Downs	new_emailll@gmail.com	1346599700	2
2	King	Frank	KingFrank3@outlook.com	9324886303	0
5	Jazlynn	Bauer	JazlynnBauer3@outlook.com	6477606880	1
6	Marcelo	Hubbard	MarceloHubbard7@yahoo.com	3571954712	2
7	Jaidyn	Spears	JaidynSpears0@hotmail.com	5219063808	2
8	Ivy	Graham	IvyGraham1@outlook.com	1612343608	1
9	Manuel	Li	ManuelLi4@gmail.com	8081255811	2
10	Kenzie	West	KenzieWest2@gmail.com	4136017496	1
11	Raelynn	Mccullough	RaelynnMccullough7@outlook.com	7273566322	0
12	Kamden	Carney	KamdenCarney0@outlook.com	5857598098	1
13	Angie	Riley	AngieRiley8@outlook.com	9783650491	2
14	Francis	Floyd	FrancisFloyd2@gmail.com	4759030626	0
15	Jayvon	Shea	JayvonShea3@gmail.com	4290219093	2
16	Jayvon	Shea	JayvonShea0@hotmail.com	1538105325	2
17	Rayna	Delgado	RaynaDelgado1@outlook.com	9732670738	2
18	Brian	Dalton	BrianDalton2@gmail.com	3576006017	1
19	Junior	Carson	JuniorCarson7@yahoo.com	8810902483	2
20	Neil	Haney	NeilHaney2@gmail.com	1262345595	1
21	newuser	new	new@gmail.com	7787133333	0

Inserting data values:

Register User

User Details

First Name

tonny

Last name

smith

Email

tonnysmith@newwww.com

Phone

7787133333

User type

Consumer ☒

Restaurant Manager ☐

Speeder ☐

Address Info

Zip

V6R 1Y1

City

Vancouver

Province

BC

Building Number

111

Unit Number

11

Street Name

cambie street

After inserting Values

View Users

user_id	first_name	last_name	email	phone	type
1	Annabel	Downs	new_emailll@gmail.com	1346599700	2
2	King	Frank	KingFrank3@outlook.com	9324886303	0
5	Jazlynn	Bauer	JazlynnBauer3@outlook.com	6477606880	1
6	Marcelo	Hubbard	MarceloHubbard7@yahoo.com	3571954712	2
7	Jaidyn	Spears	JaidynSpears0@hotmail.com	5219063808	2
8	Ivy	Graham	IvyGraham1@outlook.com	1612343608	1
9	Manuel	Li	ManuelLi4@gmail.com	8081255811	2
10	Kenzie	West	KenzieWest2@gmail.com	4136017496	1
11	Raelynn	Mccullough	RaelynnMccullough7@outlook.com	7273566322	0
12	Kamden	Carney	KamdenCarney0@outlook.com	5857598098	1
13	Angie	Riley	AngieRiley8@outlook.com	9783650491	2
14	Francis	Floyd	FrancisFloyd2@gmail.com	4759030626	0
15	Jayvon	Shea	JayvonShea3@gmail.com	4290219093	2
16	Jayvon	Shea	JayvonShea0@hotmail.com	1538105325	2
17	Rayna	Delgado	RaynaDelgado1@outlook.com	9732670738	2
18	Brian	Dalton	BrianDalton2@gmail.com	3576006017	1
19	Junior	Carson	JuniorCarson7@yahoo.com	8810902483	2
20	Neil	Haney	NeilHaney2@gmail.com	1262345595	1
21	newuser	new	new@gmail.com	7787133333	0
23	tonny	smith	tonnysmith@gmail.com	7787133333	0
25	tonny	smith	tonnysmith@newwww.com	7787133333	0