



CS166 Final Project Report

Xiyuan Wu, xwu150@ucr.edu

Julian Rodriguez, jrodr546@ucr.edu

Description

Implementation

Function Breakdown

0. Welcome UI
1. Create User
2. Log In
3. View Profile
4. Update Profile
5. View Menu
6. Place Order
7. Views all Orders
8. View Recent Orders
9. View Order Information
10. Update Order Status
11. Update Menu
12. Update User

[13. View Stores](#)

[Create Index](#)

[Problems/Findings](#)

[Contributions](#)

Description



If the text is italicized, it means it's extra work we did.

Additionally, we also did a good user interface(UI) detail see screenshot sample output.

Implementation

Function Breakdown

There are 13 functions, create an index and a report we need to be done for this project. For this part, we will talk about each function individually, provide sample output, ways to implement it, the purpose of it and extra work we did (for extra credit).

0. Welcome UI

Extra Work: We modify the welcome message that is displayed to the user, making it more friendly

```
-----  
Welcome to Pizza Store! What would you like to order today?  
-----  
  
Connecting to database...Connection URL: jdbc:postgresql://localhost:26924/xwu150_project_phase_3_DB  
  
Done  
MAIN MENU  
-----  
1. Create user  
2. Log in  
9. < EXIT  
Please make your choice: █
```

1. Create User

The purpose of this function is to create a profile for a new customer that a customer must log in to order (In this project, yes, but in the real world, customers don't have to create an account).

For this function, it will get the username, password, and phone number from the user. In the real world, the program might do additional checks, such as if the phone number is valid, but this requires additional work, such as database, so we ignore it here. *But we have to check for empty input; if it happens, we will get an error. In addition, this program also checks if the user has already registered to avoid duplicates.*

Normal Output

```
MAIN MENU
-----
1. Create user
2. Log in
9. < EXIT
Please make your choice: 1
Enter username: aa
Enter password: aa
Enter phone number: 123456123
User successfully created!
```

Already Registered

```
MAIN MENU
-----
1. Create user
2. Log in
9. < EXIT
Please make your choice: 1
Enter username: x
Enter password: x
Enter phone number: 1234567890
Error: Username already exists. Please choose a different username.
```

2. Log In

After the user is created, this function will prompt the user to log in. After creating, the user will enter their username and password to log in. If data exists, log in

successfully and display the main menu. If not, display the message "Invalid username or password. Please try again." We also do input checks for this function.

```
MAIN MENU
-----
1. Create user
2. Log in
9. < EXIT
Please make your choice: 2
Enter username: x
Enter password: x
Login successful! Welcome, x (Customer

MAIN MENU
-----
1. View Profile
2. Update Profile
3. View Menu
4. Place Order
5. View Full Order ID History
6. View Past 5 Order IDs
7. View Order Information
8. View Stores
9. Update Order Status
10. Update Menu
11. Update User
.....
20. Log out
Please make your choice: 
```

3. View Profile

User will be able to view their profile if they prefer. As the name is, this function displays the user profile. There is no check for this; once user log in successfully, they should be able to view their profile.

```
--- Viewing Profile ---
Executing query: SELECT login, favoriteItems, phoneNum FROM Users WHERE login = 'x';

--- Your Profile ---
Username: x
Favorite Items: cake
Phone Number: 1
```

The reason it has favorite item is that, we test whole program and call `updateProfile` to update user profile. Default is empty. Same thing with phone number, we updated afterwards.

4. Update Profile

This function allows user to update their profile. After the user selects it, it will prompt the user which info the user wants to update. In the code, if the user role is manager, it will also allow the manager to update a user role.

```
--- Update Profile ---
What would you like to update?
1. Favorite Items
2. Phone Number
3. Password
Enter your choice: 1
Enter new favorite items: pizza
Profile successfully updated!
```

In this function, we can see that favorite item has updated from cake to pizza.

5. View Menu

This function allows the users to view all menus from the store; the menu should be the same for different stores. Here, the user has several options to view the menu:

1. View all menu
2. Enter a specific menu type they want to order. *When prompting the user, we put a list of options in parentheses ('entree', 'drinks', 'sides) instead of just asking the user the item type they want directly. Good Interface.*
3. Filter menu by max price. Users can enter the maximum price they want to pay, and the system will only display the menu under that price.
4. Filter menu prices from low to high.
5. Filter menu prices from high to low.

By default, unlike SQL tabular format, it displays like mess

```

--- Browse Menu ---
1. View all items
2. Filter by type ('entree', 'drinks', 'sides')
3. Filter by price (items under a certain price)
4. Sort by price (Low to High)
5. Sort by price (High to Low)
Enter your choice:
1
Executing query: SELECT * FROM Items
itemname      ingredients    typeofitem    price   description
Cheese Pizza   Cheese, Bread, Tomato Sauce    entree 9.99   a classic!
Pepperoni Pizza Cheese, Bread, Tomato Sauce, Pepperoni    entree 10.99  a classic!
Hawaiian Pizza Cheese, Bread, Tomato Sauce, Pepperoni, Pineapple    entree 11.99  a twist on a classic!
Vegetarian Pizza Cheese, Bread, Tomato Sauce, onions, spinach, green peppers    entree 10.99  100% vegetarian delight
BBQ Chicken Pizza Cheese, Bread, Tomato Sauce, Chicken    entree 11.99  chicken lover's delight
Deluxe Pizza   Extra Cheese, Bread, Premium Tomato Sauce    entree 12.99  delicious
Buffalo Chicken Pizza Cheese, Bread, Tomato Sauce, Chicken    entree 12.99  contains premium buffalo chicken
Pacific Veggie Pizza Cheese, Bread, Tomato Sauce, Onions, Anchovies    entree 10.99  for the fish lovers
Wisconsin 6-Cheese Pizza Cheese, Bread, Tomato Sauce    entree 12.99  all types of cheese
Spinach & Feta Pizza Cheese, Bread, Tomato Sauce, Spinach, Feta    entree 10.99  a customer favorite
Pesto Margherita Pizza Cheese, Bread, Tomato Sauce    entree 10.99  a classic!
Spicy Chicken Sausage Pizza Cheese, Bread, Tomato Sauce, Chicken    entree 10.99  a classic!
Meat Lover's Pizza Cheese, Bread, Tomato Sauce, Chicken, Bacon    entree 10.99  chicken and bacon surprise
Sprite        High Fructose Corn Syrup, Lemon, Lime    drinks 1.99   delicious!
Coca-Cola     High Fructose Corn Syrup    drinks 1.99   delicious!
Pepsi         High Fructose Corn Syrup    drinks 1.99   delicious!
7up          High Fructose Corn Syrup, Lemon, Lime    drinks 1.99   delicious!
Lemonade      Lemon    drinks 1.99   summer vibes
Water Bottle   Hydrogen, Oxygen    drinks 1.99   a classic!
Breadsticks   Bread    sides 3.99   premium quality!
Fries          Potatoes   sides 3.99   premium quality!
Nachos         Bread    sides 3.99   premium quality!
Pasta          Wheat, Water, Flour   sides 3.99   premium quality!
Caesar Salad   Lettuce, Tomatoes, Crutons   sides 3.99   premium quality!
Brownie        Chocolate Chips, Flour   sides 3.99   premium quality!
Ice Cream      Milk, Sprinkles   sides 3.99   premium quality!
Chocolate Cake Chocolate Chips, Cakemix   sides 3.99   premium quality!

```

After we update in code, it now displays a well-format

```

--- Browse Menu ---
1. View all items
2. Filter by type ('entree', 'drinks', 'sides')
3. Filter by price (items under a certain price)
4. Sort by price (Low to High)
5. Sort by price (High to Low)
Enter your choice:
1
Item Name           Ingredients          Type       Price   Description
Cheese Pizza        Cheese, Bread, Tomato Sauce    entree    9.99   a classic!
Pepperoni Pizza     Cheese, Bread, Tomato Sauce, Pepperoni    entree   10.99  a classic!
Hawaiian Pizza      Cheese, Bread, Tomato Sauce, Pepperoni, Pineapple    entree   11.99  a twist on a classic!
Vegetarian Pizza    Cheese, Bread, Tomato Sauce, onions, spinach, green peppers    entree   10.99  100% vegetarian delight
BBQ Chicken Pizza   Cheese, Bread, Tomato Sauce, Chicken    entree   11.99  chicken lover's delight
Deluxe Pizza        Extra Cheese, Bread, Premium Tomato Sauce    entree   12.99  delicious
Buffalo Chicken Pizza Cheese, Bread, Tomato Sauce, Chicken    entree   12.99  contains premium buffalo chicken
Pacific Veggie Pizza Cheese, Bread, Tomato Sauce, Onions, Anchovies    entree   10.99  for the fish lovers
Wisconsin 6-Cheese Pizza Cheese, Bread, Tomato Sauce    entree   12.99  all types of cheese
Spinach & Feta Pizza Cheese, Bread, Tomato Sauce, Spinach, Feta    entree   10.99  a customer favorite
Pesto Margherita Pizza Cheese, Bread, Tomato Sauce    entree   10.99  a classic!
Spicy Chicken Sausage Pizza Cheese, Bread, Tomato Sauce, Chicken    entree   10.99  a classic!
Meat Lover's Pizza  Cheese, Bread, Tomato Sauce, Chicken, Bacon    entree   10.99  chicken and bacon surprise
Sprite              High Fructose Corn Syrup, Lemon, Lime    drinks  1.99   delicious!
Coca-Cola           High Fructose Corn Syrup    drinks  1.99   delicious!
Pepsi               High Fructose Corn Syrup    drinks  1.99   delicious!
7up                High Fructose Corn Syrup, Lemon, Lime    drinks  1.99   delicious!
Lemonade            Lemon    drinks  1.99   summer vibes
Water Bottle        Hydrogen, Oxygen    drinks  1.99   a classic!
Breadsticks         Bread    sides  3.99   premium quality!
Fries               Potatoes   sides  3.99   premium quality!
Nachos              Bread    sides  3.99   premium quality!
Pasta               Wheat, Water, Flour   sides  3.99   premium quality!
Caesar Salad         Lettuce, Tomatoes, Crutons   sides  3.99   premium quality!
Brownie             Chocolate Chips, Flour   sides  3.99   premium quality!
Ice Cream            Milk, Sprinkles   sides  3.99   premium quality!
Chocolate Cake      Chocolate Chips, Cakemix   sides  3.99   premium quality!

```

6. Place Order

After browsing the menu, the user can place an order. There are 6 steps:

1. Ask users which store they want to place. *We also tell user store ID are from 1-1000, not just ask for store ID*
2. Generate a unique order ID

3. Ask the user for the name and quantities they want. *It will check if the user enters the correct item name. If incorrect, it will prompt the user to enter again.*
4. Calculate total price
5. Insert new order into Food Order
6. Insert items into Items In Order
7. Display total price

```
--- Place Order ---
Enter the Store ID (1-1000) where you want to place your order: 1
Enter item name (or type 'done' to finish): Pizza
Error: Item not found in menu. Please enter a valid item.
Enter item name (or type 'done' to finish): 7up
Enter quantity: 1
Enter item name (or type 'done' to finish): done
Order placed successfully! Your total price is: $1.99
```

7. Views all Orders

This function allows both customers and managers to see order history. The function first checks the role of the user and then it retrieves order history based on the role. Customers can see their own orders while managers will be able to see all customers' orders.

--- Order History ---						
OrderID	Username	StoreID	Total Price	Timestamp	Status	
<hr/>						
10002	istitts	362	10.99	2024-11-15 17:40:36	incomplete	
10003	jbalaamc	123	25.98	2018-02-12 13:11:23	complete	
10001	jbalaamc	123	1.99	2016-08-24 14:03:20	complete	
10000	lbeldom3	967	32.96	2014-05-07 07:01:29	complete	

8. View Recent Orders

This function allows users to view their five most recent orders. The orders are retrieved in descending order by timestamp with the most recent order coming first.

--- Recent 5 Orders ---					
OrderID	StoreID	Total Price	Timestamp	status	
<hr/>					
10003	123	25.98	2018-02-12 13:11:23	complete	
10001	123	1.99	2016-08-24 14:03:20	complete	

9. View Order Information

This function allows users to view all the information associated with an order including: order ID, store ID, total price, timestamp, status, and the items in the order. The user has to first input the order number so the function can retrieve it from the database. Customers can only view their own orders while managers and drivers can view the information on any order.

```
Enter Order ID to view details: 10003
```

```
Order ID: 10003
Store ID: 123
Total Price: 25.98
Timestamp: 2018-02-12 13:11:23
Status: complete
```

```
Items in Order:
- Buffalo Chicken Pizza (x2)
```

10. Update Order Status

This function allows managers and drivers to update the status of an order. The user has to first input the order number and then the new status of the order. If a customer tries to do this, they will be met with an error message.

```
Enter Order ID to update status: 10003
Enter new status (Pending, In Progress, Completed, Cancelled): Pending
Error: Only Managers or Drivers can update order status.
```

However, if a manager or driver tries to update the status, they will be met with a success message and the status will be changed in the database.

```
Enter Order ID to update status: 10003
Enter new status (Pending, In Progress, Completed, Cancelled): Cancelled
Order status updated successfully!
```

11. Update Menu

This function allows managers to update the menu. They are presented with the choice to either update an existing item or add a new item. If they choose to add a new item, they are presented with the screen below. The user has to enter the item name, ingredients, type, price, and description of the item.

```
--- Update Menu ---
1. Update an existing item
2. Add a new item
Enter your choice: 2
Enter new item name: Item
Enter ingredients: Apple
Enter type (entree, drinks, sides): sides
Enter price: 100
Enter description: Description
New menu item added successfully!
```

If the user chooses to update an existing item, they are asked to input the name of the item. The user can then choose which field of the item to update.

```
--- Update Menu ---
1. Update an existing item
2. Add a new item
Enter your choice: 1
Enter item name to update: Item
Which field do you want to update?
1. Ingredients
2. Type
3. Price
4. Description
Enter your choice: 2
Enter new type (entree, drinks, sides): drinks
Menu item updated successfully!
```

12. Update User

This function allows managers to update the information regarding a user. If a driver or customer tries to access this command, they will be met with an error message. Once a manager inputs the username of a user, they are presented with the option to update any field of the user. The manager can change a users' username, password, role, favorite items, and phone number.

```
--- Update User ---
Enter the username of the user to update: gblackfordm
What would you like to update?
1. Change login username
2. Change password
3. Change role (Customer/Manager/Driver)
4. Update favorite items
5. Update phone number
Enter your choice: 3
Enter new role (Customer/Manager/Driver): Customer
User updated successfully!
```

13. View Stores

This function displays all of the stores in the database. It includes the stores' id, address, city, state, open status, and review score.

1	83 Hallows Center	San Diego	California	yes	4
2	92096 Bunker Hill Avenue	Spring	Texas	yes	4
3	8229 Jenifer Terrace	Portland	Oregon	yes	5
4	4 Golf Parkway	Portland	Oregon	no	5
5	8383 Westend Terrace	Omaha	Nebraska	no	5
6	37131 Miller Terrace	Honolulu	Hawaii	no	5
7	73424 Straubel Parkway	Charlotte	North Carolina	no	5
8	65 Service Crossing	Indianapolis	Indiana	yes	4
9	0 Independence Circle	Sioux Falls	South Dakota	yes	5
10	04547 School Alley	New York City	New York	yes	4
11	6 Fordem Court	Alhambra	California	no	4
12	5 Monterey Junction	Knoxville	Tennessee	no	5
13	78634 Ludington Terrace	Decatur	Georgia	no	5
14	4102 Donald Terrace	Spokane	Washington	no	4
15	7 Debra Trail	Memphis	Tennessee	yes	4
16	7765 Roxbury Lane	Des Moines	Iowa	no	4

In addition, we changed the UI for the table display so it looks cleaner as shown below.

--- List of Stores ---						
Store ID	Address	City	State	Open	Review	
1	83 Hallows Center	San Diego	California	yes	4	
2	92096 Bunker Hill Avenue	Spring	Texas	yes	4	
3	8229 Jenifer Terrace	Portland	Oregon	yes	5	
4	4 Golf Parkway	Portland	Oregon	no	5	
5	8383 Westend Terrace	Omaha	Nebraska	no	5	
6	37131 Miller Terrace	Honolulu	Hawaii	no	5	
7	73424 Straubel Parkway	Charlotte	North Carolina	no	5	
8	65 Service Crossing	Indianapolis	Indiana	yes	4	
9	0 Independence Circle	Sioux Falls	South Dakota	yes	5	
10	04547 School Alley	New York City	New York	yes	4	
11	6 Fordem Court	Alhambra	California	no	4	
12	5 Monterey Junction	Knoxville	Tennessee	no	5	
13	78634 Ludington Terrace	Decatur	Georgia	no	5	
14	4102 Donald Terrace	Spokane	Washington	no	4	
15	7 Debra Trail	Memphis	Tennessee	yes	4	

Create Index

For this project, we decided to use B+ tree indexing because it's a range query. We used indexing to speed up the SELECT process for our database. This is

especially noticeable when we are fetching all stores since the function prints out all 1000 stores much faster when using indexes.

```
CREATE INDEX idx_users_login ON Users USING btree(login);  
CREATE INDEX idx_users_role ON Users USING btree(role);
```

```
CREATE INDEX idx_items_name ON Items USING btree(itemName);  
CREATE INDEX idx_items_ingredients ON Items USING btree(ingredients);  
CREATE INDEX idx_items_typeofitem ON Items USING btree(typeOfItem);  
CREATE INDEX idx_items_price ON Items USING btree(price);
```

```
CREATE INDEX idx_store_id ON Store USING btree(storeID);  
CREATE INDEX idx_store_city ON Store USING btree(city);  
CREATE INDEX idx_store_state ON Store USING btree(state);  
CREATE INDEX idx_store_reviewscore ON Store USING btree(reviewScore);  
CREATE INDEX idx_store_isOpen ON Store USING btree(isOpen);
```

```
CREATE INDEX idx_foodorder_login ON FoodOrder USING btree(login);  
CREATE INDEX idx_foodorder_storeid ON FoodOrder USING btree(storeID);  
CREATE INDEX idx_foodorder_orderid ON FoodOrder USING btree(orderID);  
CREATE INDEX idx_foodorder_timestamp ON FoodOrder USING btree(orderTime);  
CREATE INDEX idx_foodorder_totalprice ON FoodOrder USING btree(totalPrice);  
CREATE INDEX idx_foodorder_status ON FoodOrder USING btree(orderStatus);
```

```
CREATE INDEX idx_itemsinorder_orderid ON ItemsInOrder USING btree(orderID);  
CREATE INDEX idx_itemsinorder_itemname ON ItemsInOrder USING btree(itemName);  
CREATE INDEX idx_itemsinorder_quantity ON ItemsInOrder USING btree(quantity);
```

Problems/Findings

Some problems we had encountered:

1. Unfamiliar with Java.
2. When implementing a function like `viewProfile`, it needs user login info to display. So there is an issue for given templates; it can't get user info because

it doesn't pass in that parameter. Since the `authoriseuser` store is in the main, we need to pass this in as a parameter, and the problem is solved.

3. There was also a problem where the "manager" role was passed incorrectly so the query had to be trimmed. To fix this, we added TRIM(BOTH FROM role) to every select query that selects the users' role. We also added a `.trim()` to the Java role variable to make sure that the role is trimmed correctly.

Contributions

Xiyuan Wu — Function 0 to 6, create the index

Julian Rodriguez — Function 7-13, improve the index

Report done together.