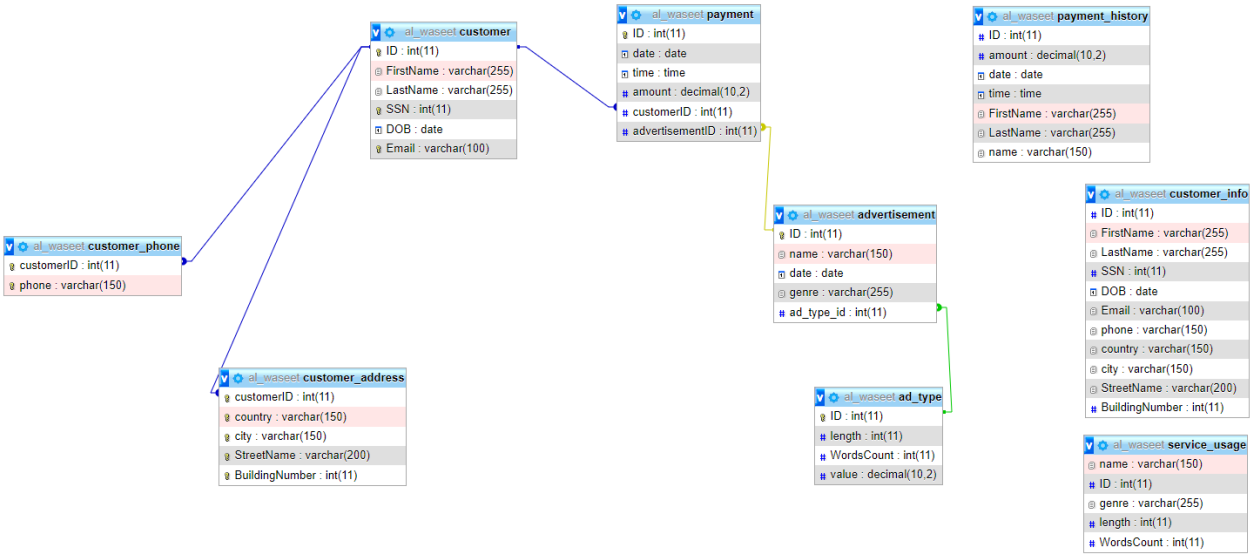
Technical Documentation

1 Introduction

This document is going to talk about how this database was created, what are the tables, relationships and constraints that we have for each entity. It will also discuss the database's views with each one's name and description on what it does, same thing for procedures each one's name is mentioned with a description about it. There is also a table for the privileges for our 2 users (employee and customer) with all the queries for the grants and descriptions on each one of them. We will also have a validation part that will discuss input, output and security validation. Input validation will discuss the constraints and how they are correct. Output validation discusses the select statement with views and procedures so we can know if we have the correct result after querying these select statement. And lastly security validation will discuss the privileges given to the users and how they are separated. And lastly this document talks about the gui's pages and their validation.

2 Physical Schema



3 Database Development

3.1 Database Overview

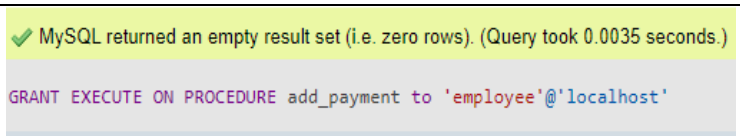
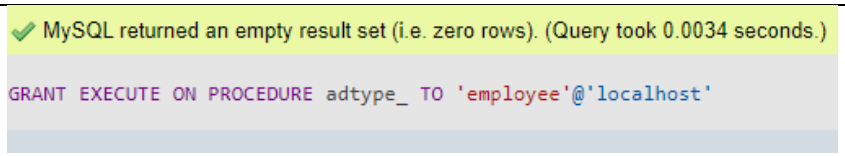
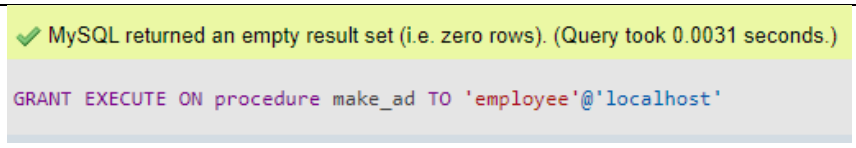
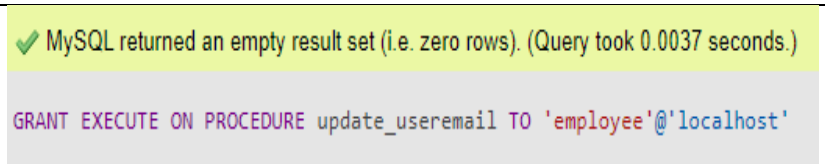
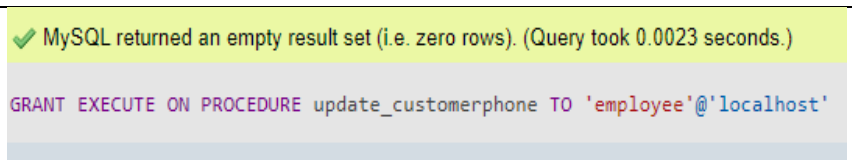
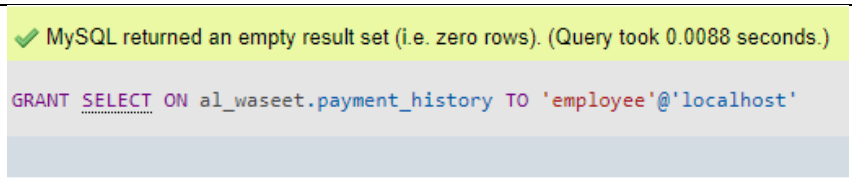
Table	Name	Description
1.	Customer	This table contains 6 attributes and has an ID as its primary key. It also has a relationship with customer address, customer phone and payment, as it gives them customerID as a foreign key.
2.	Customer_address	Customer address has 5 attributes and is related to the customer table by the customerID as a foreign key and primary key, all the other attributes are considered primary keys as well.
3.	Customer_phone	Customer phone has 2 attributes and is related to the customer table by the customerID as a foreign key and primary key, the other attribute is also considered a primary key.
4.	Advertisement	Advertisement has 5 attributes and is related to the ad_type table and payment table the primary key of this table is ID, it takes a foreign key from ad_type_id and gives one to payment (advertisement_id).
5.	Ad_type	Ad_type has 4 attributes and it's a table which decides the ad type and how its value is calculated it has an ID as its primary key and it's related to the advertisement table.
6.	payment	Payment is a table that 6 attributes and has a relationship with advertisement and customer tables, this table has ID as its primary key and is used to give the information of the payment to the customer with their ID and the ID of the advertisement they are paying for.

View	Name	Description
1.	Customer_info	This view shows all the customers' information in the database with their address and phone number
2.	Payment_history	Shows payment information with the customer's name and advertisement name
3.	Service_usage	This view shows columns from 4 tables (customer, payment, advertisement and ad_type) this will give us all the needed information about the service we have.
4.		

Procedure	Name	Description
1.	Make_ad	Allows the customer to make an advertisement and add rows into advertisement. Takes 5 parameters which are; name, genre, date, ad_type_id and id.
2.	Update_useremail, Update_customerphone, Update_customeraddress	These procedures allow the customer to edit their email phone and address. They all take the customerID as a parameter then each one takes the wanted value for example update_useremail takes customerID and email. And customerphone takes customerID and Phone.
3.	Customer_payment	Allows the customer to see information about their payment and the customer id is the parameter with information about the advertisement, it takes c_id as the parameter which is customerID.
4.	Add_payment	Allows the employee to add a payment. It takes 6 parameters which are (amount, id, date, time, advertisement_id, customer_id)
5.	Adtype_	This procedure takes all the ad_type columns as attributes in order for the user to be able to insert information into the table.

3.2 Security

User name	Privilege Command	Description	Screenshot
employee	GRANT SELECT ON al_waseet.service_usage TO 'employee'@'localhost'	This privilege will give the employee access to see the table from that view about the service usage.	MySQL returned an empty result set (i.e. zero rows). (Query took 0.0034 seconds.) `GRANT SELECT ON al_waseet.service_usage TO 'employee'@'localhost'`
	GRANT SELECT,UPDATE ON al_waseet.customer_info TO 'employee'@'localhost'	This privilege will give the employee access to update and	MySQL returned an empty result set (i.e. zero rows). (Query took 0.0036 seconds.) `GRANT SELECT,UPDATE ON al_waseet.customer_info TO 'employee'@'localhost'`

		see the customer's information.	
	GRANT EXECUTE ON PROCEDURE add_payment to 'employee'@'localhost'	This will allow the employee to help the customers to add their payment	
	GRANT EXECUTE ON PROCEDURE adtype_ TO 'employee'@'localhost'	This will also allow the employee to help the customer for their adtype	
	GRANT EXECUTE ON procedure make_ad TO 'employee'@'localhost'	This will allow the employees help the customers for making an ad and inserting.	
	GRANT EXECUTE ON PROCEDURE update_useremail TO 'employee'@'localhost'	Can let the employee update a customers email.	
	GRANT EXECUTE ON PROCEDURE update_customerphone TO 'employee'@'localhost'	Can let the employee update a customer's phone	
	GRANT SELECT ON al_waseet.payment_history TO 'employee'@'localhost'	This privilege will give the employee access to see the table from that view about the payment history of the service.	

customer	GRANT EXECUTE ON PROCEDURE add_payment TO 'customer'@'localhost'	This privilege will give the user(customer) permission to execute the procedure which will make them add a payment.	MySQL returned an empty result set (i.e. zero rows). (Query took 0.0036 seconds.) `GRANT EXECUTE ON PROCEDURE add_payment TO 'customer'@'localhost'`

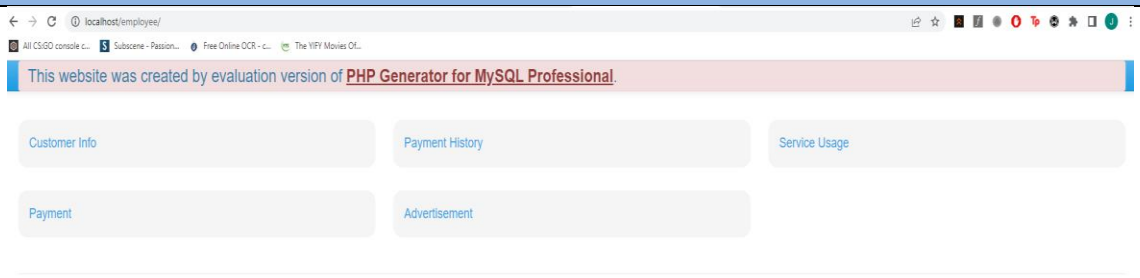
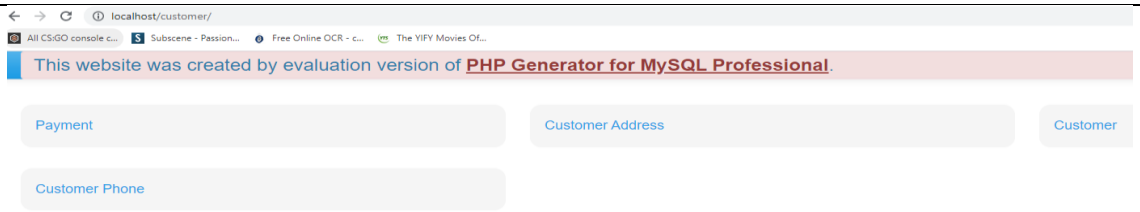
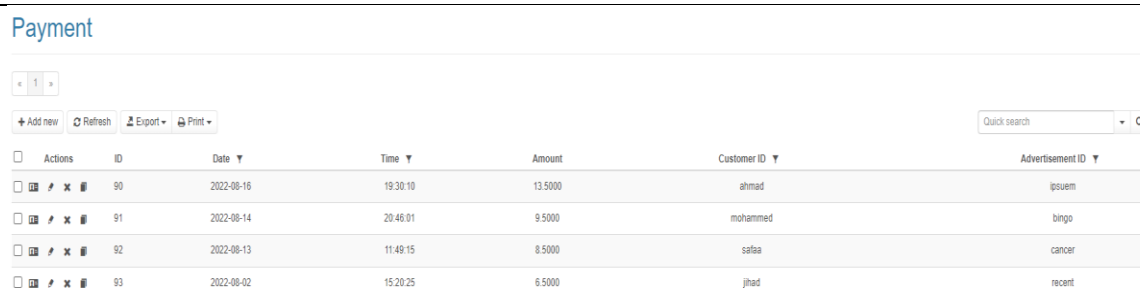
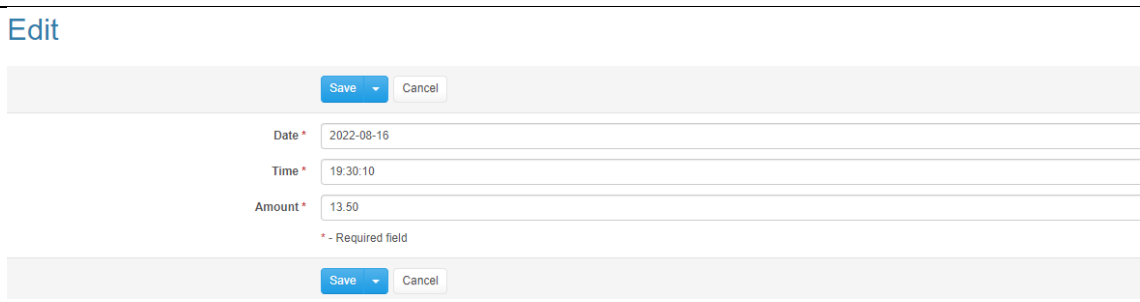
	GRANT EXECUTE ON PROCEDURE update_useremail to 'customer'@'localhost'
	
	GRANT EXECUTE ON PROCEDURE update_customeraddress to 'customer'@'localhost'
	
	GRANT EXECUTE ON PROCEDURE update_customerphone to 'customer'@'localhost'
		This will give the customer the privilege to update their email, address and number using this procedure.	MySQL returned an empty result set (i.e. zero rows). (Query took 0.0029 seconds.)
			`GRANT EXECUTE ON PROCEDURE update_useremail to 'customer'@'localhost'`
			
			MySQL returned an empty result set (i.e. zero rows). (Query took 0.0036 seconds.)
			`GRANT EXECUTE ON PROCEDURE update_customeraddress to 'customer'@'localhost`
			
			MySQL returned an empty result set (i.e. zero rows). (Query took 0.0036 seconds.)
			`GRANT EXECUTE ON PROCEDURE update_customerphone to 'customer'@'localhost'`

	GRANT EXECUTE ON PROCEDURE customer_payment TO 'customer'@'localhost'	This privilege will allow the customer to execute the procedure to see their payments	MySQL returned an empty result set (i.e. zero rows). (Query took 0.0036 seconds.) `GRANT EXECUTE ON PROCEDURE customer_payment TO 'customer'@'localhost'`
customer	GRANT EXECUTE ON PROCEDURE make_ad TO 'customer'@'localhost'	This privilege will allow the customer to insert into the advertisement table for their ad.	MySQL returned an empty result set (i.e. zero rows). (Query took 0.0032 seconds.) `GRANT EXECUTE ON PROCEDURE make_ad TO 'customer'@'localhost'`

3.3 User Interface

3.3.1 Flowchart and Data Movement Diagrams (DFD)

3.3.2 Interfaces Development

Page ID	Title	Description	Screenshot
1.	Employee homepage	This page is the first page the employee will see when he opens the system	
2.	Customer homepage	This page is the first page the customer will see when he opens the system	
3.	tables	When the user clicks on one of the tables they will see this screen with an option of actions	
4.	Editing	This is what happens when the user wants to edit on a table this is the screen that shows up	

4 Maintenance

4.1 Database recovery & backups

4.2 Database maintenance in general

5 Testing

5.1 Data Validation

Number	Type	Description	screenshot

1.	All cases of PK	Here I tried to insert data into the customer table with a primary key that has been already used. (unique)	**Error**
			SQL query: Copy
			`INSERT INTO `customer` (`ID`, `FirstName`, `LastName`, `SSN`, `DOB`, `Email`) VALUES (11, 'malak', 'ahmad', 101025, '2001-09-09', 'malak.ahmad@gmail.com')`
			MySQL said:
			`#1062 - Duplicate entry '11' for key 'PRIMARY'`

		Here I was validating that a primary key cannot be a null value in another table which was ad_type. (not null validation)	**Error**
			SQL query: Copy
			`INSERT INTO `ad_type` (`ID`, `length`, `WordsCount`, `value`) VALUES (null, 12, 150, '15.50')`
			MySQL said:
			`#1048 - Column 'ID' cannot be null`

<table>
<tr><td rowspan="2">2.</td><td rowspan="2">All cases of FK</td><td>Update cascade. Here I changed an existing id which was 11 and updated to 15, I then checked in the customer address table to see if it updated.</td><td>Here I used this query to update in the table customer


```
UPDATE `customer` SET `ID` = '15' WHERE `customer`.`ID` = 11
```


+ Options

	customerID	country	city	StreetName	BuildingNumber
Edit Copy Delete	12	Jordan	Amman	gardens st.	23
Edit Copy Delete	12	Jordan	Zarqa	army street	19
Edit Copy Delete	13	Jordan	Amman	university st.	6
Edit Copy Delete	14	Jordan	Aqaba	atallah suhaimat st	90
Edit Copy Delete	15	Jordan	Amman	mecca st.	12
Edit Copy Delete	15	Jordan	Irbid	madina st.	15

You can also edit most values by double-clicking directly on them.</td></tr>
<tr><td>Delete cascade. Here I will delete a row in the parent table and then see if it's deleted in the child table.</td><td>Query: DELETE FROM `ad_type` WHERE `ad_type`.`ID` = 115

	ID	name	date	genre	ad_type_id
Edit Copy Delete	1005	ipsuem	2022-08-01	political	101
Edit Copy Delete	1006	bingo	2022-08-06	sport	102
Edit Copy Delete	1007	cancer	2022-08-15	health	103
Edit Copy Delete	1008	recent	2022-08-09	news	104

</td></tr>
<tr><td>3.</td><td>Unique</td><td>SSN in the table customer is given a unique constraint</td><td>Error

SQL query: Copy


```
INSERT INTO customer (ssn) VALUES (101011)
```


MySQL said:


```
#1062 - Duplicate entry '101011' for key 'SSN'
```

</td></tr>
<tr><td>4.</td><td>int</td><td>Here I will insert a string in a cell that has an int</td><td>

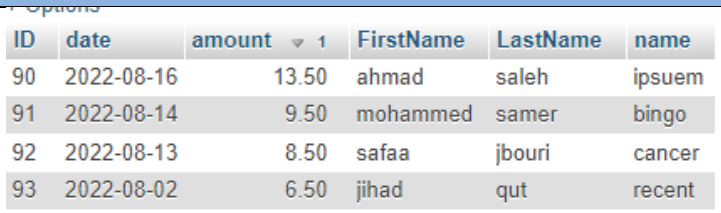
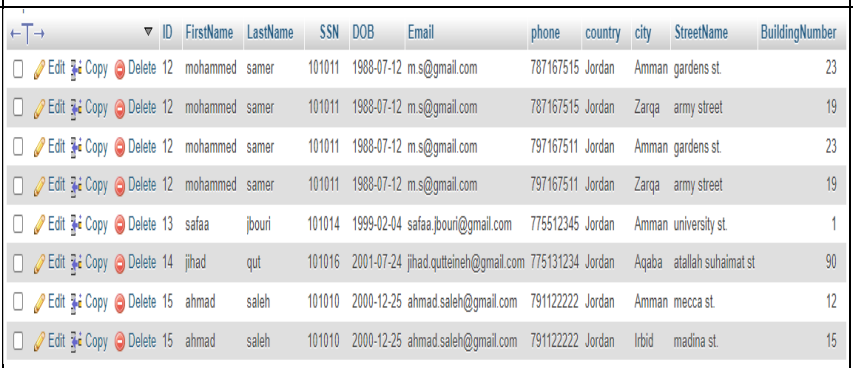
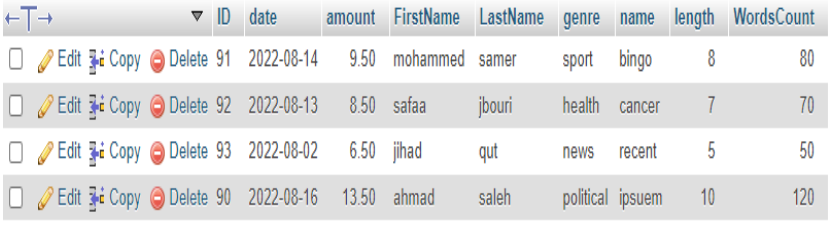
```
UPDATE `customer_address` SET `BuildingNumber` = 'ahmad' WHERE `customer_address`.`customerID` = 13
```

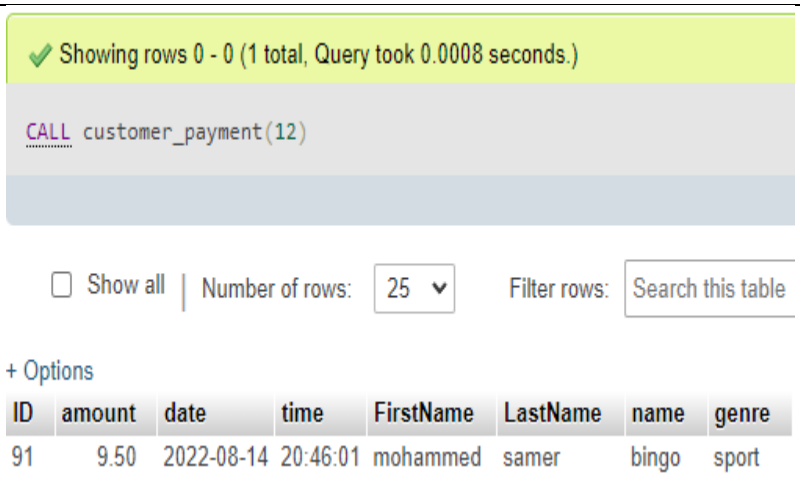

⚠ Warning: #1366 Incorrect integer value: 'ahmad' for column `al_waseet`.`customer_address`.`BuildingNumber` at row 1</td></tr>
</table>

		datatype. We can see that it will take it but it's giving us a warning, mysql uses casting in this situation to try to give the cell a value.	

5.	Not null	Here we have a not null constraint on length from ad_type	**Error**
			SQL query: Copy
			`INSERT INTO `ad_type` (`ID`, `length`, `WordsCount`, `value`) VALUES (120, null , 120, '12.50')`
			MySQL said:
			`#1048 - Column 'length' cannot be null`

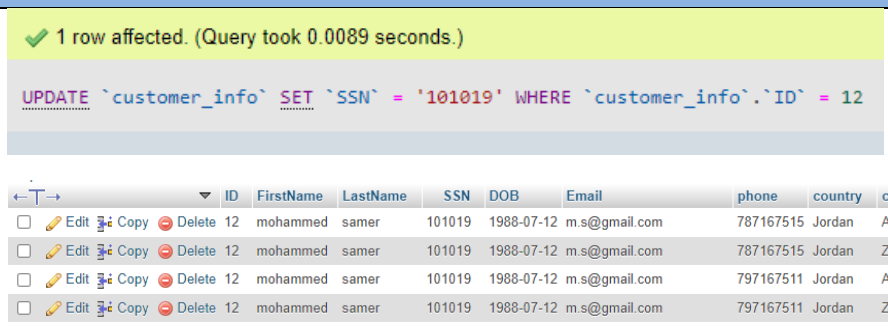
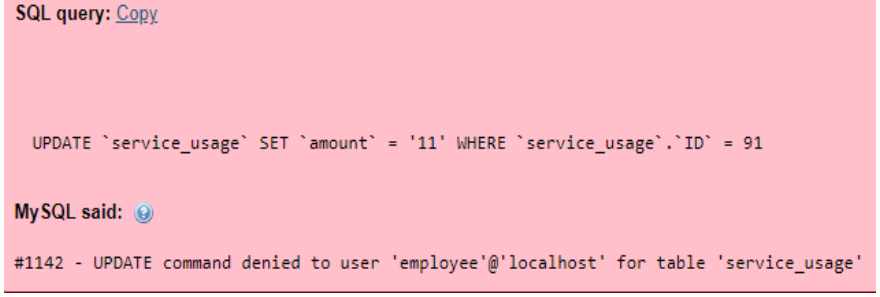
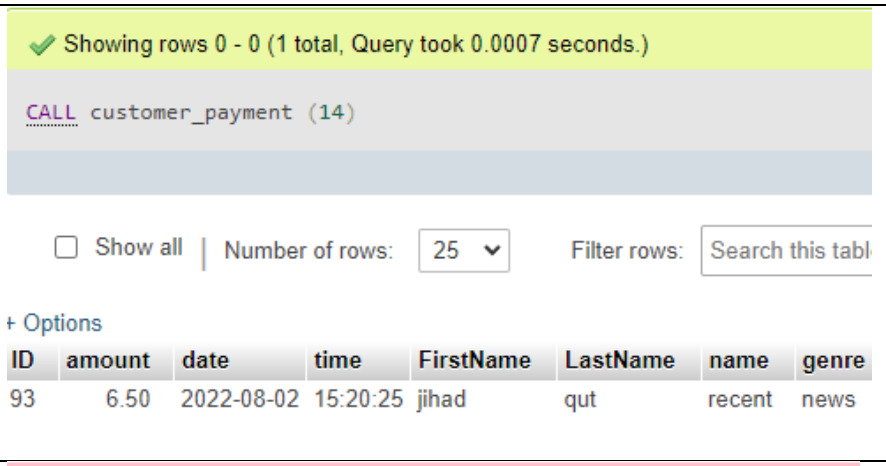
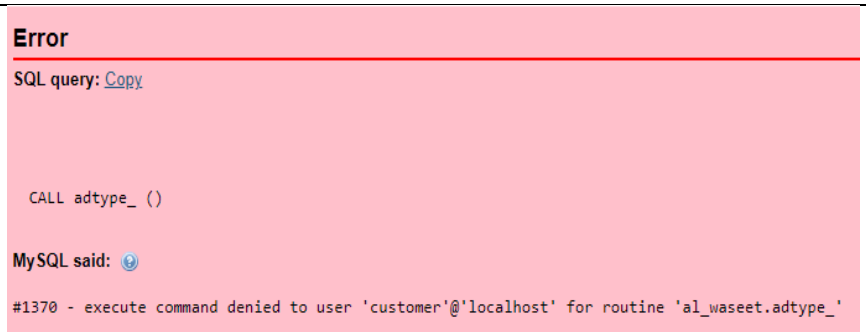
5.2 Output Validation

Number	Query Description	Screenshot (query + result)	Result validation
1.	This query is used to give us information about the payment with the customer's name and the advertisement name and then we ordered them from the highest amount to lowest	 SELECT payment.ID, payment.date, payment.amount, customer.FirstName, customer.LastName, advertisement.name FROM payment INNER JOIN customer ON payment.customerID = customer.ID INNER JOIN advertisement ON payment.advertisementID = advertisement.ID ORDER BY amount DESC	We can see that I used inner join twice in order to get the advertisement name and the customer's name, and also used order by to order the data by their amount descending
2.	This query was created in a view and it will enable the employee to view all the customer's information from 3 tables (customer, customer_address customer_phone)	 *query and full screenshot at the last pages of the report	We can see that we collected the needed information from all 3 tables with all the columns needed. We can see 4 columns for the first customer because he has 2 adresses and 2 phone numbers.
3.	This query was created as a view that will give the employee a table that has information from 4 tables that are about the service usage.	 *query and full screenshot at the last pages of the report	We can see that we are able to retrieve the needed columns from these tables correctly to get the needed information about the service usage.

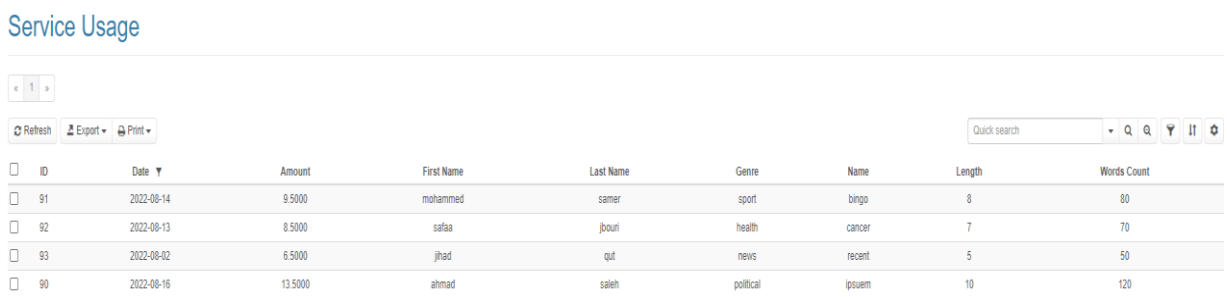
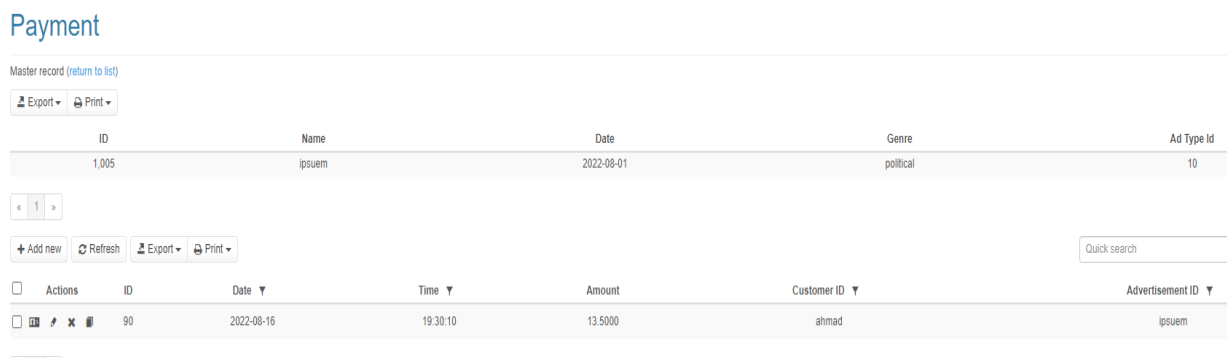
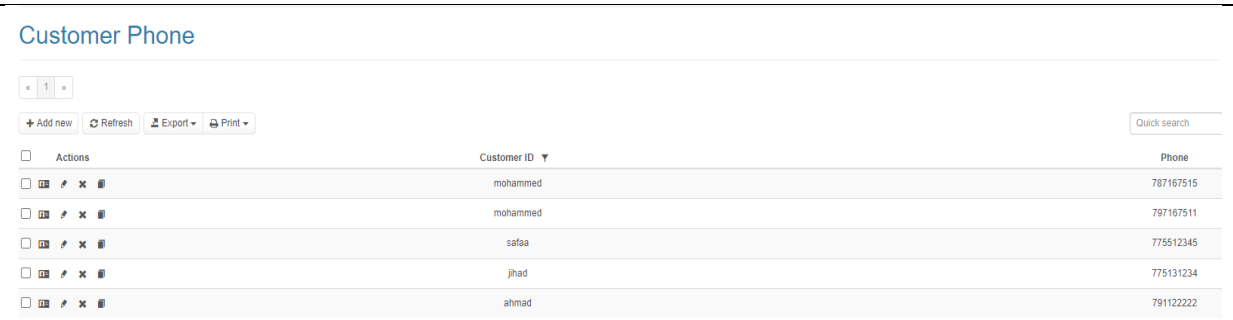
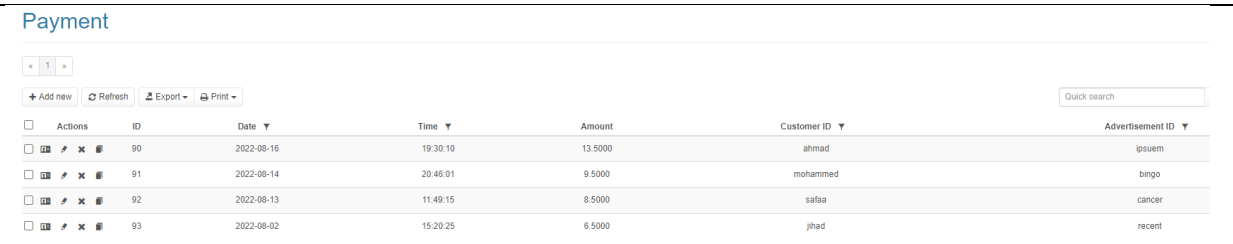
4.	This query is from a procedure that takes the customer id as a parameter and shows him/her information about their payment		We can see that the output we have is only showing us the information to the customer that entered his/her id.

5.3 Security Validation

Note: you need to test the given and not given privileges.

Number	User Name	Description of privilege/no privilege	Screenshot (query + result)
1.	Employee	The customer was given permission to update and select on the view customer_info.	
2.	Employee	The employee has only the privilege to select at service_usage.	
3.	Customer	The customer was given permission to execute the procedure customer payment which has a select statement.	
4.	Customer	The customer wasn't given permission to execute the procedure adtype_	

5.4 GUI Validation

Number	Description	screenshot
1.	The employee is only allowed to select the service_usage table without being able to make any actions on it.	
2.	The employee should be able to see the payment with its advertisement like this. And he can do actions on the payment	
3.	The customer should be able to edit and delete a phone number for example like this	
4.	The customer should be able to see his payment without editing on it like this.	

5.5 Assess whether meaningful data has been extracted
5.6 Assess the effectiveness of testing

6 Evaluation of database solution
6.1 Effectiveness of the database solution based on user and system requirement
6.2 Suggested improvements
6.3 Evaluation based on improvements needed

Output validation 1:

Showing rows 0 - 3 (4 total, Query took 0.0041 seconds.)

```
SELECT payment.ID, payment.date, payment.amount, customer.FirstName, customer.LastName, advertisement.name FROM payment INNER JOIN customer ON payment.customerID = customer.ID INNER JOIN advertisement ON payment.advertisementID = advertisement.ID ORDER BY amount DESC
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PH

Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

Options

ID	date	amount	FirstName	LastName	name
90	2022-08-16	13.50	ahmad	saleh	ipsuem
91	2022-08-14	9.50	mohammed	samer	bingo
92	2022-08-13	8.50	safaa	jbouri	cancer
93	2022-08-02	6.50	jihad	qut	recent

Output validation 2:

SELECT customer.ID, customer.FirstName, customer.LastName, customer.SSN, customer.DOB, customer.Email, customer_phone.phone, customer_address.country, customer_address.city, customer_address.StreetName, customer_address.BuildingNumber FROM customer

LEFT JOIN customer_phone ON customer_phone.customerID = customer.ID

LEFT JOIN customer_address ON customer_address.customerID = customer.ID

Showing rows 0 - 7 (8 total, Query took 0.0046 seconds.)

```
SELECT customer.ID, customer.FirstName, customer.LastName, customer.SSN, customer.DOB, customer.Email, customer_phone.phone, customer_address.country, customer_address.city, customer_address.StreetName, customer_address.BuildingNumber FROM customer LEFT JOIN customer_phone ON customer_phone.customerID = customer.ID LEFT JOIN customer_address ON customer_address.customerID = customer.ID
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create

Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

Options

ID	FirstName	LastName	SSN	DOB	Email	phone	country	city	StreetName	BuildingNumber
12	mohammed	samer	101011	1988-07-12	m.s@gmail.com	787167515	Jordan	Amman	gardens st.	23
12	mohammed	samer	101011	1988-07-12	m.s@gmail.com	787167515	Jordan	Zarqa	army street	19
12	mohammed	samer	101011	1988-07-12	m.s@gmail.com	797167511	Jordan	Amman	gardens st.	23
12	mohammed	samer	101011	1988-07-12	m.s@gmail.com	797167511	Jordan	Zarqa	army street	19
13	safaa	jbouri	101014	1999-02-04	safaa.jbouri@gmail.com	775512345	Jordan	Amman	university st.	1
14	jihad	qut	101016	2001-07-24	jihad.qutteineh@gmail.com	775131234	Jordan	Aqaba	atallah suhaimat st	90
15	ahmad	saleh	101010	2000-12-25	ahmad.saleh@gmail.com	791122222	Jordan	Amman	mecca st.	12
15	ahmad	saleh	101010	2000-12-25	ahmad.saleh@gmail.com	791122222	Jordan	Irbid	madina st.	15

Output validation 3:

CREATE or REPLACE VIEW service_usage AS

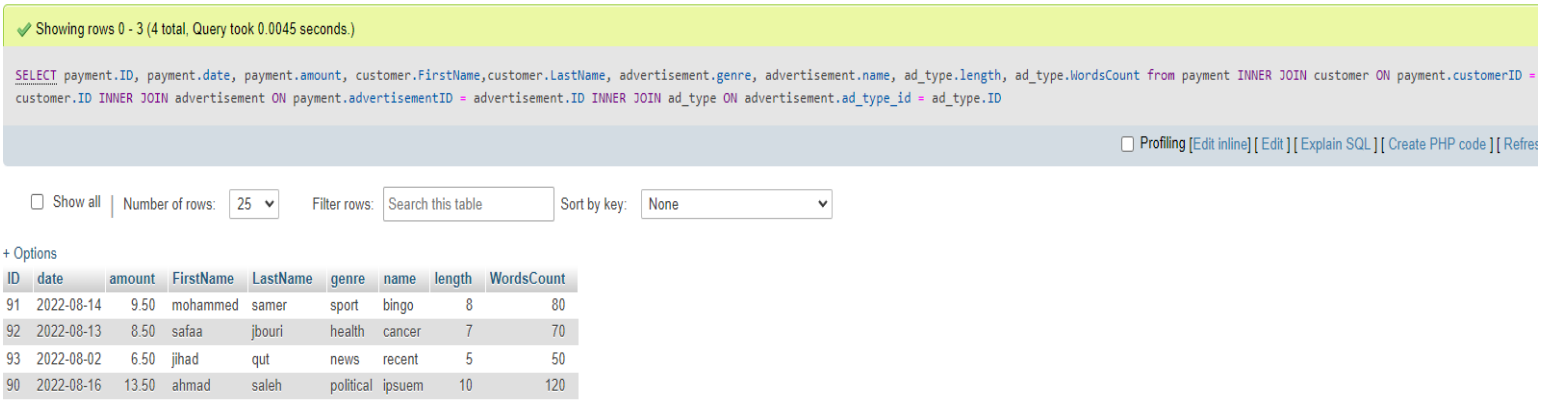
SELECT payment.ID, payment.date, payment.amount, customer.FirstName,customer.LastName, advertisement.genre, advertisement.name, ad_type.length, ad_type.WordsCount

from payment

INNER JOIN customer ON payment.customerID = customer.ID

INNER JOIN advertisement ON payment.advertisementID = advertisement.ID

INNER JOIN ad_type ON advertisement.ad_type_id = ad_type.ID



ID	date	amount	FirstName	LastName	genre	name	length	WordsCount
91	2022-08-14	9.50	mohammed	samer	sport	bingo	8	80
92	2022-08-13	8.50	safaa	jbouri	health	cancer	7	70
93	2022-08-02	6.50	jihad	qut	news	recent	5	50
90	2022-08-16	13.50	ahmad	saleh	political	ipsuem	10	120

Output validation 4: (procedure select statement)

SELECT payment.ID,payment.amount, payment.date, payment.time, customer.FirstName, customer.LastName

FROM payment

INNER JOIN customer ON payment.customerID = customer.ID

WHERE customerID = c_id