COMPUTER PROJECT

PREFACE

We discovered the concept of an online shopping cart while researching the subject. An online shopping cart is a virtual tool that allows customers to easily select and save things from an online business. It functions as a digital equivalent to a physical shopping cart in a physical store, delivering a streamlined and organized shopping experience.

Users can utilize an online shopping cart to browse various products, add desired things to their cart, and continue browsing the store's offerings. The cart keeps track of the goods you've chosen, their quantities, and their pricing.

The online shopping cart then computes the total cost, including discounts, and displays an order summary for evaluation.

An online shopping cart's convenience allows consumers to simply amend their selections, remove things, or add more before finalizing their purchase.

In conclusion, an online shopping cart improves the whole online shopping experience by offering a user-friendly interface for selecting, organizing, and purchasing items from an online store. It simplifies the process, saves time, and provides a safe environment for customers to shop with confidence.

INTRODUCTION

The online shopping cart project seeks to give clients with a simple and easy-to-use platform for browsing, selecting, and purchasing products online. Because of the growing popularity of e-commerce, firms must have an effective online purchasing system in place.

Python, a powerful programming language noted for its simplicity and readability, is used to create the project. Python enables us to create a fully functional online shopping cart system. It allows us to efficiently handle user inputs, make calculations, and maintain data.

In addition, SQL is employed in our project for database management. SQL offers an organized method for storing, organizing, and managing huge amounts of data relating to products, customer information, and so on. It ensures that data is safely stored and that it can be quickly accessed and retrieved when needed.

We can construct a online shopping cart system that fits the needs of both customers and businesses by combining Python and SQL. The project's goal is to simplify the purchasing process, improve user experience, and boost overall consumer happiness.

OBJECTIVE

Our idea aims to provide users with a seamless purchasing experience. Customers may quickly add goods to their cart and view, update, and delete their cart, as well as the total amount of the things purchased.

SCOPE

Online shopping carts have a vast scope nowadays. They are essential tools for businesses to sell their products or services online. Here are some key aspects of the scope of online shopping carts:

1. Product Management: Online shopping carts allow businesses to add, organize, and manage their products or services. You can easily update product details, prices, and inventory information.

2. Payment Integration: Online shopping carts integrate with various payment gateways, enabling businesses to accept payments securely and conveniently. They support multiple payment methods like credit/debit cards, digital wallets, and even cash on delivery, depending on the business preferences.

3. Order Processing: Shopping carts automate the order processing workflow. They generate order confirmations, handle inventory updates, and display total amount.

SYSTEM REQUIREMENTS

·Hardware:
Processor- Intel(R) Core(TM) i3-10100 CPU @ 3.60GHz 3.60 GHz
Installed Ram- 8.00 GB (7.84 GB usable)
Device ID- 10961D9C-B48F-4FF2-8256-ABA1095FC996
Product ID- 00331-10000-00001-AA567
System type- 64-bit operating system, x64-based processor

·Software:
Python and MYSQL database

SYSTEM DESIGN

DATABASE DESIGN

Tables created:

- **Items**

1. **ItemNumber (PRIMARY KEY)**
2. **ItemName**
3. **In_stock**
4. **Price**
5. **discount**

```
mysql> desc items;
+------------+--------------+------+-----+---------+-------+
| Field      | Type         | Null | Key | Default | Extra |
+------------+--------------+------+-----+---------+-------+
| ItemNumber | int          | NO   | PRI | NULL    |       |
| ItemName   | varchar(255) | YES  |     | NULL    |       |
| in_stock   | int          | YES  |     | NULL    |       |
| price      | float        | YES  |     | NULL    |       |
| discount   | float        | YES  |     | NULL    |       |
+------------+--------------+------+-----+---------+-------+
5 rows in set (1.90 sec)
```

item table:

```
mysql> select * from items;
+------------+----------+----------+-------+----------+
| ItemNumber | ItemName | in_stock | price | discount |
+------------+----------+----------+-------+----------+
|          1 | Item A   |       10 | 15.99 |       20 |
|          2 | Item B   |        5 | 24.99 |       50 |
|          3 | Item C   |       20 | 10.49 |       15 |
|          4 | Item D   |        8 | 32.75 |     42.5 |
|          5 | Item E   |       15 |  7.99 |       10 |
|          6 | Item F   |       12 | 18.75 |       30 |
|          7 | Item G   |       30 |  6.49 |     27.5 |
|          8 | Item H   |        7 |  42.5 |       65 |
+------------+----------+----------+-------+----------+
```

- **User**

1. **Slno (PRIMARY KEY)**
2. **user_id**
3. **Username**
4. **pincode**
5. **itemnumber**

```
mysql> desc user;
+------------+--------------+------+-----+---------+----------------+
| Field      | Type         | Null | Key | Default | Extra          |
+------------+--------------+------+-----+---------+----------------+
| slno       | int          | NO   | PRI | NULL    | auto_increment |
| user_id    | int          | YES  |     | NULL    |                |
| username   | varchar(255) | YES  |     | NULL    |                |
| pincode    | varchar(10)  | YES  |     | NULL    |                |
| itemnumber | int          | YES  |     | NULL    |                |
+------------+--------------+------+-----+---------+----------------+
5 rows in set (0.00 sec)
```

- **payment**

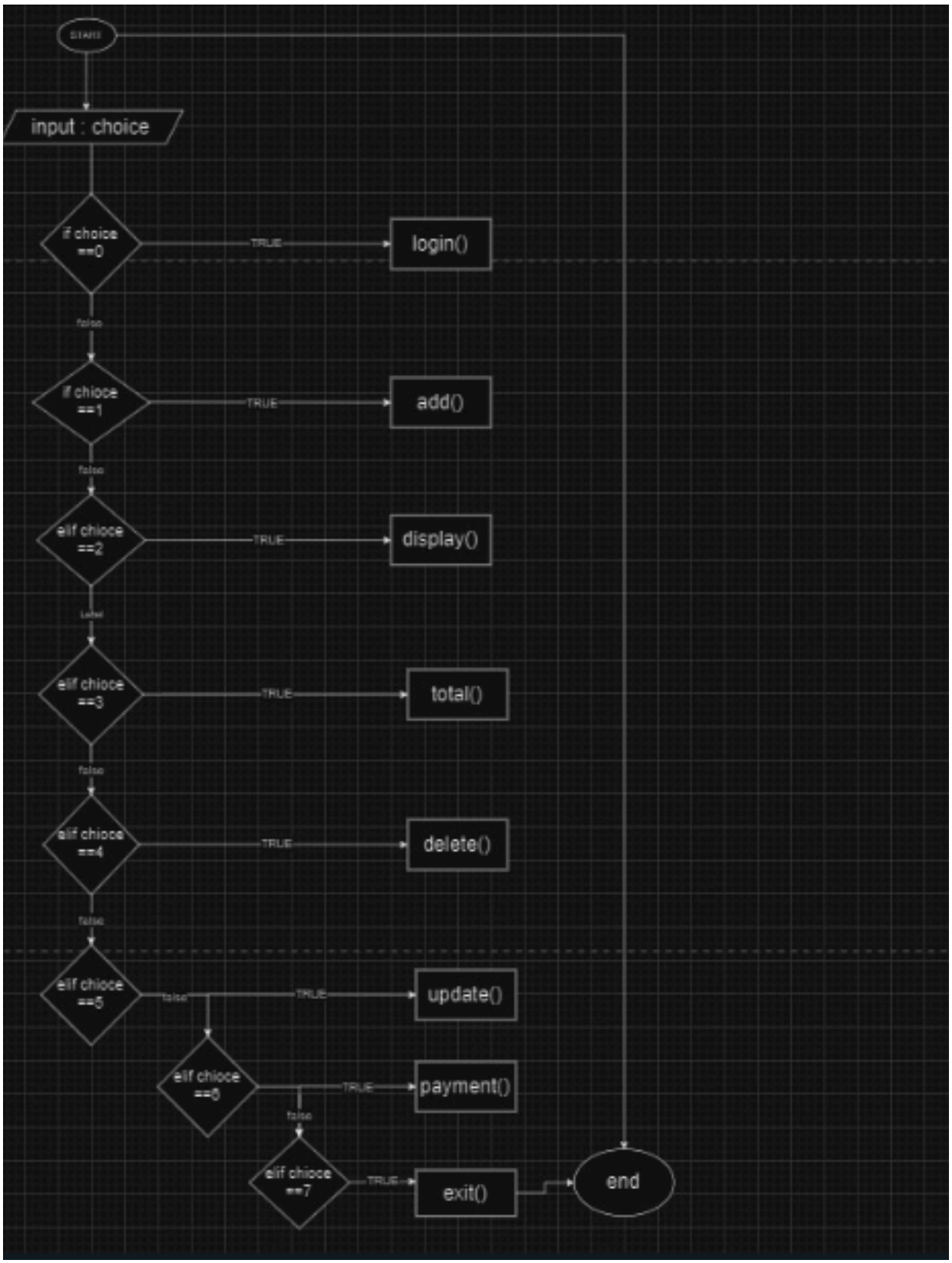
1. **payment_id (PRIMARY KEY)**
2. **pay_type**
3. **user_id**

```
mysql> desc payment;
+------------+-------------+------+-----+---------+----------------+
| Field      | Type        | Null | Key | Default | Extra          |
+------------+-------------+------+-----+---------+----------------+
| payment_id | int         | NO   | PRI | NULL    | auto_increment |
| pay_type   | varchar(20) | YES  |     | NULL    |                |
| user_id    | int         | YES  |     | NULL    |                |
+------------+-------------+------+-----+---------+----------------+
3 rows in set (0.00 sec)
```

MENU DESIGN

- **login():for user to login**
- **Add: function(): to used to add the items to the cart**
- **Displaycart(): to display the cart**
- **total_amount(): to display the total amount of the user**
- **update: to update the item in the cart**
- **Delete: to delete the data in the cart**

FLOWCHART


START
input : choice
if choice ==0
TRUE
login()
false
if choice ==1
TRUE
add()
false
elif choice ==2
TRUE
display()
false
elif choice ==3
TRUE
total()
false
elif choice ==4
TRUE
delete()
false
elif choice ==5
false
TRUE
update()
elif choice ==6
TRUE
payment()
false
elif choice ==7
TRUE
exit()
end

CODE

```
import mysql.connector
userid = None
username = None
pincode = None

def login():
    global userid, username, pincode
    if userid is None or username is None or pincode is None:
        userid = input("Enter userid: ")
        username = input("Enter username: ")
        pincode = input("Enter pincode: ")
    return userid, username, pincode
```

```
def displaycart():
    import mysql.connector

    con = mysql.connector.connect(host='localhost',
                                  user='root', password='root123',
                                  database='shopping_cart')

    cur = con.cursor()
    userid, username, pincode = login()

    print()
    query = "select user_id,username,itemname,price,discount,qty from items,user where user.itemnum
        userid)
    cur.execute(query)
    cart = cur.fetchall()

    print('user_id',' ','username','itemname' ,'','price',' ','discount', '   ','qty')

    print()
    for i in cart:
        for j in i:
            print(j,end='\t')
        print()
    con.commit()
    con.close()
    cur.close()
```

```
def add():
    import mysql.connector
    con = mysql.connector.connect(host='localhost',
                                  user='root', password='root123',
                                  database='shopping_cart')

    cur = con.cursor()
    userid, username, pincode = login()

    print("--------------------------------------")
    print("1-itemA\n2-itemB\n3-itemC\n4-itemD\n5-itemE\n6-itemF\n7-itemG\n8-itemH")
    print("--------------------------------------")
    while True:
        sel_itemno = input("Enter the item numbers you want to add to cart ")
        qty=input("enter the quantity")

        query = "insert into user (user_id , username ,pincode,itemnumber,qty) values({},'{}','{}',{},{

        cur.execute(query)
        ch = input("input y to continue,n to stop: ")
        if ch == 'n' or ch == 'N':
            break
    con.commit()
    con.close()
    cur.close()
    displaycart()
```

```python
def total_amount():
    import mysql.connector

    con = mysql.connector.connect(host='localhost',
                                  user='root', password='root123',
                                  database='shopping_cart')
    cur = con.cursor()
    count = 0
    userid, username, pincode = login()
    query = "select username,cast(SUM(Price - (Price * Discount / 100))*qty as signed) from items,user
        userid)
    cur.execute(query)
    r = cur.fetchall()
    for i in r:
      z=i[1]
      count=count+z
    print("your total amount is :",'$',count )
    con.commit()
    con.close()
    cur.close()
    return count
```

```python
def update():
    import mysql.connector

    con = mysql.connector.connect(host='localhost',
                                  user='root', password='root123',
                                  database='shopping_cart')
    cur=con.cursor()
    userid, username, pincode = login()
    query = "select user_id,username,itemname,price,discount from items,user where user.itemnumber=ite
        userid)
    cur.execute(query)
    cart = cur.fetchall()
    print('the items your can select to update are \n')
    for i in cart:
        print(i[2])
    print()
    print("the cart before update")
    displaycart()
    print()
    print("----------------------------------------")
    print("1-itemA\n2-itemB\n3-itemC\n4-itemD\n5-itemE\n6-itemF\n7-itemG\n8-itemH")
    print("----------------------------------------")
    z = int(input("input the previous item number"))
    b = int(input("new itemnumber "))
    qtyy=input("enter the new  quantity")
    query = "update user set itemnumber= {} where itemnumber={}".format(b, z)
    q1="update user set qty= {} where itemnumber={}".format(qtyy, z)
    cur.execute(q1)
    cur.execute(query)
    con.commit()
    con.close()
    cur.close()
    print("the cart after update")
    print("\n")
    displaycart()
    print()
```

```python
def delete():
    con = mysql.connector.connect(host='localhost',
                                  user='root', password='root123',
                                  database='shopping_cart')
    cur = con.cursor()
    userid, username, pincode = login()
    query = "select user_id,username,itemname,price,discount from items,user where user.itemnumber=item
        userid)
    cur.execute(query)
    cart = cur.fetchall()
    print('the items your can select to delete are \n')
    for i in cart:
        print(i[2])
    print()
    print("the item numbers are")
    print("----------------------------------------")
    print("1-itemA\n2-itemB\n3-itemC\n4-itemD\n5-itemE\n6-itemF\n7-itemG\n8-itemH")
    print("----------------------------------------")
    z = int(input("input the item number of the item you want to delete"))
    print("the cart before deletion")
    displaycart()
    print()
    query = "delete from  user where itemnumber={};".format(z)
    cur.execute(query)
    con.commit()
    con.close()
    cur.close()
    print("after deletion")
```

```python
    displaycart()
    print()
def payment():
    import mysql.connector

    con = mysql.connector.connect(host='localhost',
                                  user='root', password='root123',
                                  database='shopping_cart')
    cur = con.cursor()
    userid, username, pincode = login()
    print("payment options are google_pay,credit_card , debit _card")
    payment = input("the type of payment:")
    q=total_amount()
    pay=int(input("enter your amount to be owed"))
    print()
    if pay==q:
        print("Transation Successful")
    else:
        print("error invalid payment")
    query = "insert into payment (pay_type,user_id) values('{}',{});".format(payment, userid)
    z = cur.execute(query)
    con.commit()
    con.close()
    cur.close()
```

```python
while True:
    print("----------------------------------------")
    chioce = input("0-login\n1- add to cart\n2- display cart\n3- show total price of items bought\n4-de
    print("----------------------------------------")
    if chioce =="0":
        print()
        login()
        print()
    elif chioce == "1":
        add()
        print("\n")
    elif chioce == "2":
        print("your cart is:")
        displaycart()
        print("\n")
    elif chioce == "3":
        total_amount()
        print("\n")
    elif chioce == "4":
        delete()
    elif chioce == "5":
        update()
    elif chioce == "6":
        print("\n")
        payment()
        print("\n")
    elif chioce=="7":
        break
```

OUTPUT

```
----------------------------------------
0-login
1- add to cart
2- display cart
3- show total price of items bought
4-delete an item from cart
5- update an item from cart
6 to select payment method
7 to exit
enter your chioce0
-------------------------------------

Enter userid: 190
Enter username: katlin
Enter pincode: 1213
```

```
-------------------------------------
0-login
1- add to cart
2- display cart
3- show total price of items bought
4-delete an item from cart
5- update an item from cart
6 to select payment method
7 to exit
enter your chioce1
-------------------------------------
-------------------------------------
1-itemA
2-itemB
3-itemC
4-itemD
5-itemE
6-itemF
7-itemG
8-itemH
-------------------------------------
Enter the item numbers you want to add to cart 1
enter the quantity4
input y to continue,n to stop: y
Enter the item numbers you want to add to cart 2
enter the quantity3
input y to continue,n to stop: n

user_id   username itemname  price   discount    qty

190       katlin   Item A   15.99   20.0        4
190       katlin   Item B   24.99   50.0        3
```

```
0-login
1- add to cart
2- display cart
3- show total price of items bought
4-delete an item from cart
5- update an item from cart
6 to select payment method
7 to exit
enter your chioce2
-----------------------------------
your cart is:

user_id   username itemname  price   discount    qty

190       katlin     Item A    15.99     20.0      4
190       katlin     Item B    24.99     50.0      3
```

```
0-login
1- add to cart
2- display cart
3- show total price of items bought
4-delete an item from cart
5- update an item from cart
6 to select payment method
7 to exit
enter your chioce3
-----------------------------------
your total amount is : $ 88
```

```
0-login
1- add to cart
2- display cart
3- show total price of items bought
4-delete an item from cart
5- update an item from cart
6 to select payment method
7 to exit
enter your chioce4
-----------------------------------
the items your can select to delete are

Item A
Item B

the item numbers are
-----------------------------------
1-itemA
2-itemB
3-itemC
4-itemD
5-itemE
6-itemF
7-itemG
8-itemH
-----------------------------------
input the item number of the item you want to delete1
the cart before deletion

user_id   username itemname  price   discount    qty

190       katlin     Item A    15.99     20.0      4
190       katlin     Item B    24.99     50.0      3
```

```
after deletion

user_id   username itemname  price   discount    qty

190        katlin      Item B     24.99      50.0       3
```

```
0-login
1- add to cart
2- display cart
3- show total price of items bought
4-delete an item from cart
5- update an item from cart
6 to select payment method
7 to exit
enter your chioce5
-------------------------------------
the items your can select to update are

Item B

the cart before update

user_id   username itemname  price   discount    qty

190        katlin      Item B     24.99      50.0       3

-------------------------------------
1-itemA
2-itemB
3-itemC
4-itemD
5-itemE
6-itemF
7-itemG
8-itemH
-------------------------------------
input the previous item number2
new itemnumber 4
enter the new  quantity3
```

```
enter the new  quantity3
the cart after update



user_id   username itemname  price   discount    qty

190        katlin      Item D     32.75      42.5       3
```

```
0-login
1- add to cart
2- display cart
3- show total price of items bought
4-delete an item from cart
5- update an item from cart
6 to select payment method
7 to exit
enter your chioce6
-------------------------------------


payment options are google_pay,credit_card , debit_card
the type of payment:google_pay
your total amount is : $ 56
enter your amount to be owed56

Transation Successful


-------------------------------------
0-login
1- add to cart
2- display cart
3- show total price of items bought
4-delete an item from cart
5- update an item from cart
6 to select payment method
7 to exit
enter your chioce7
```

CONCLUSION

The project successfully incorporates key features such as user authentication, product management, order processing, and transaction tracking. Through SQL databases, we ensure data integrity, security, and efficient query performance, providing a solid foundation for the application. Python's capabilities have been harnessed for building a user-friendly interface, implementing business logic, and connecting seamlessly with the underlying database.

The use of SQL databases has played a pivotal role in managing and organizing our product inventory, customer information, and order history. Leveraging the power of SQL queries has allowed us to retrieve and manipulate data with efficiency, ensuring quick response times and optimal performance. The relational structure of the database has facilitated seamless connections between different entities, contributing to a cohesive and interconnected system.

One of the key achievements of this project is the development of a well-structured database that efficiently manages product information, user data, and order details. The SQL database has proven to be a powerful tool in handling complex queries, ensuring data integrity.

To conclude here are some features to improvise this project

- **Product Recommendations:Implement a recommendation system based on user history.**
- **Search and Filtering Options:Advanced search functionality with filters for efficient product discovery.**

BIBLIOGRAPHY

- https://github.com/relations-dil/python-relations-sql
- https://youtu.be/hlGoQC332VM?si=cY_7jPGeHsVClFhn
- ncert computer science textbook
- sumita arora computer science textbook