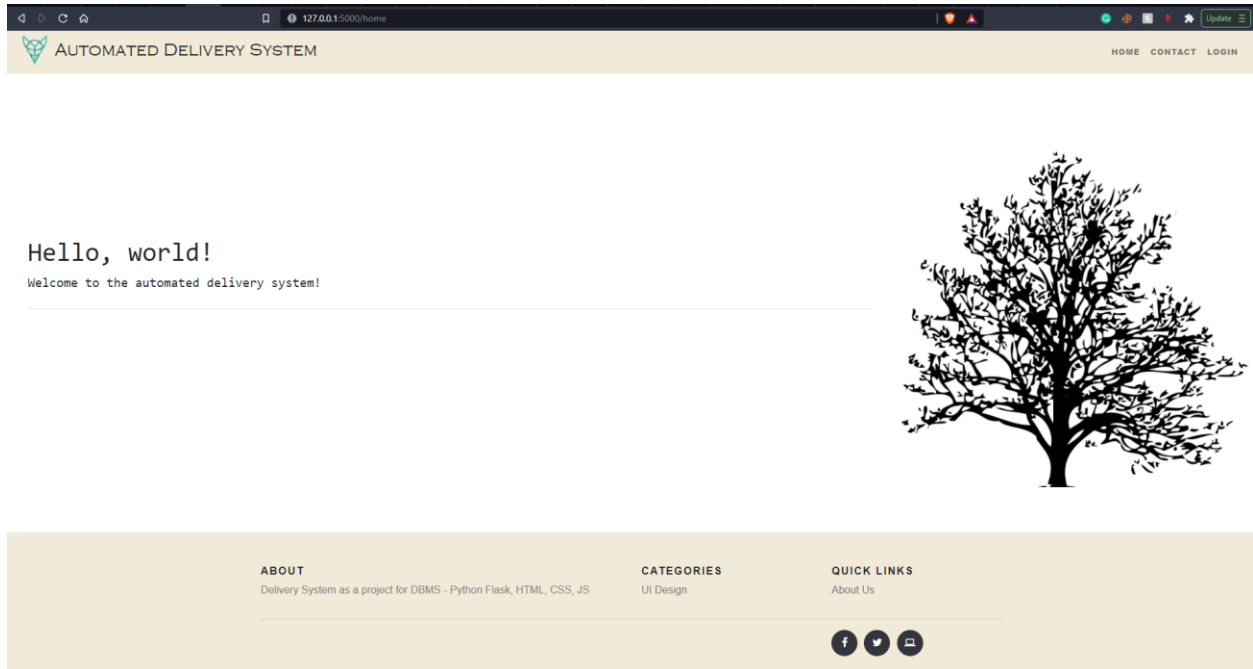
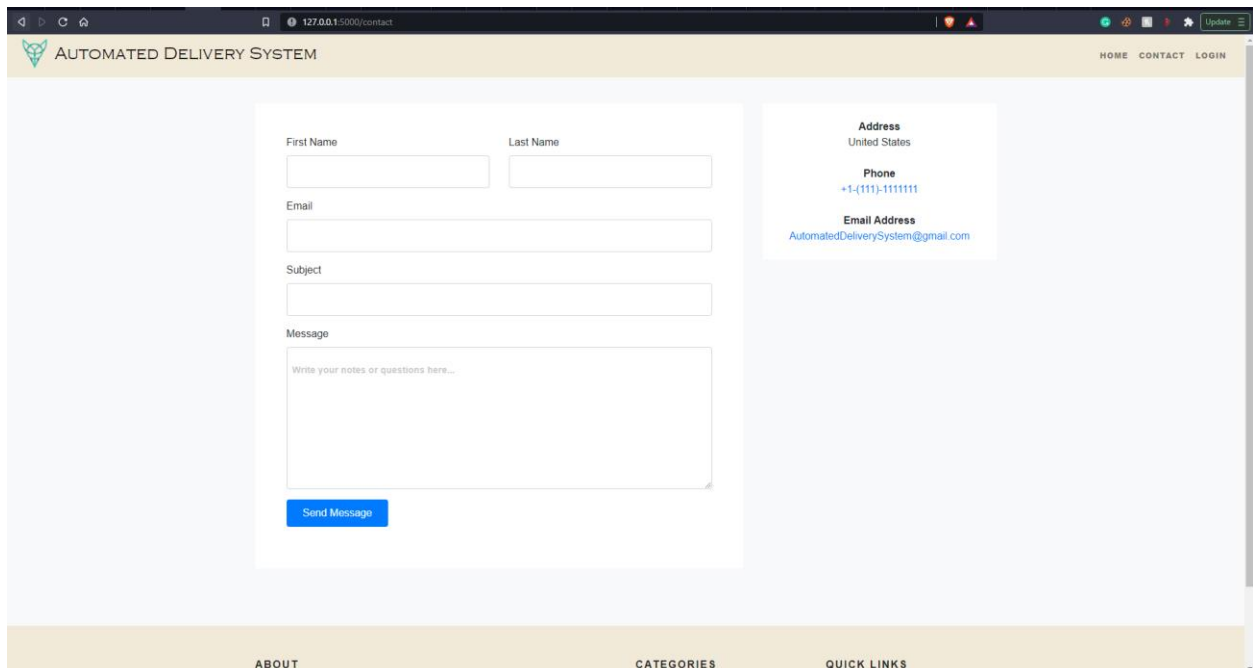


Screenshot Output

1. Home Page:



2. Contact Page:



3. Sign in page:

The screenshot shows the 'Sign In' page of the 'Automated Delivery System'. The page has a light beige header with the system name and navigation links (HOME, CONTACT, LOGIN). The main content area features a white sign-in form with a blue 'SIGN IN' button. The form includes a dropdown for account type, fields for username and password, a 'Remember password' checkbox, and a link to 'Register' for new users. The footer contains three sections: 'ABOUT' (describing the system as a project for DBMS), 'CATEGORIES' (UI Design), and 'QUICK LINKS' (About Us).

Automated Delivery System

HOME CONTACT LOGIN

Sign In

Select an account type

username

Password

☐ Remember password

SIGN IN

OR

Create an account? [Register](#)

ABOUT
Delivery System as a project for DBMS - Python Flask, HTML, CSS, JS

CATEGORIES
UI Design

QUICK LINKS
About Us

4. Register page (account types: customer, shops, bot):

The screenshot shows the 'Create an Account' page of the 'Automated Delivery System'. The page features a cartoon robot illustration on the left and a registration form on the right. The form includes fields for Name, Street, City, Zip, Phone Number, and account type, along with Username and Password fields. A blue 'Create your account' button is at the bottom of the form. The footer is identical to the sign-in page.

Automated Delivery System

HOME CONTACT LOGIN

Create an Account

Typically it takes 2 mins to register a new account

Name

Street

City

Zip

+1 Phone Number

Select an account type

Username

Password

Create your account

OR

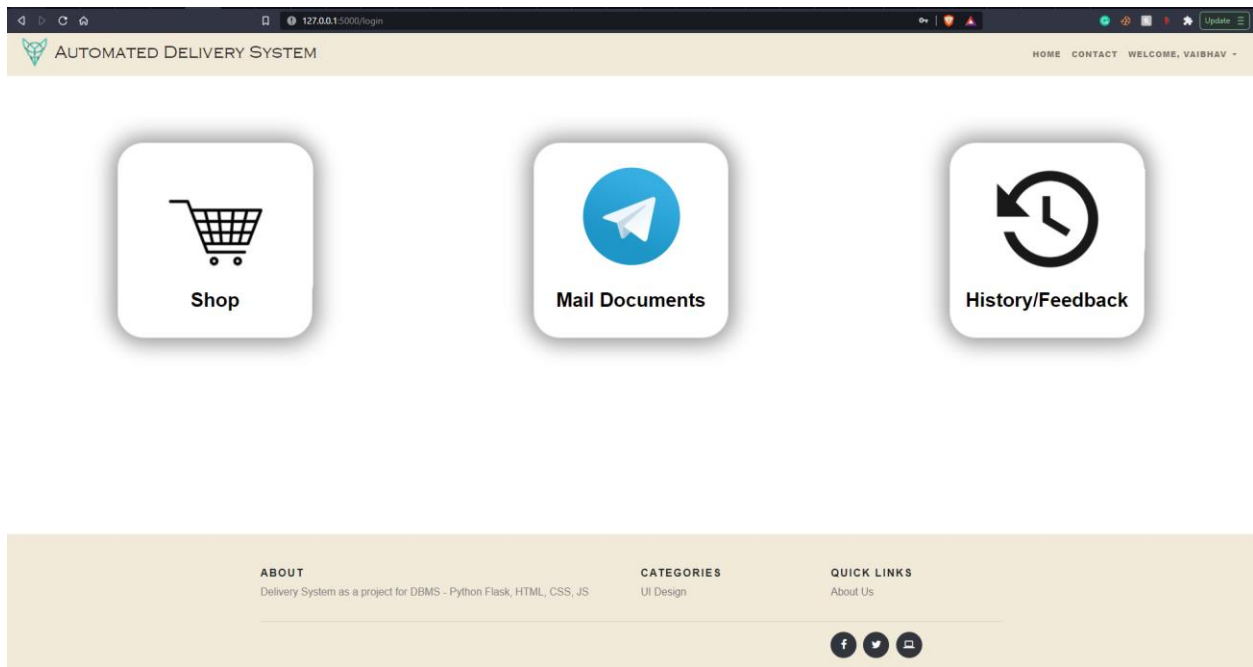
Already Registered? [Login](#)

ABOUT
Delivery System as a project for DBMS - Python Flask, HTML, CSS, JS

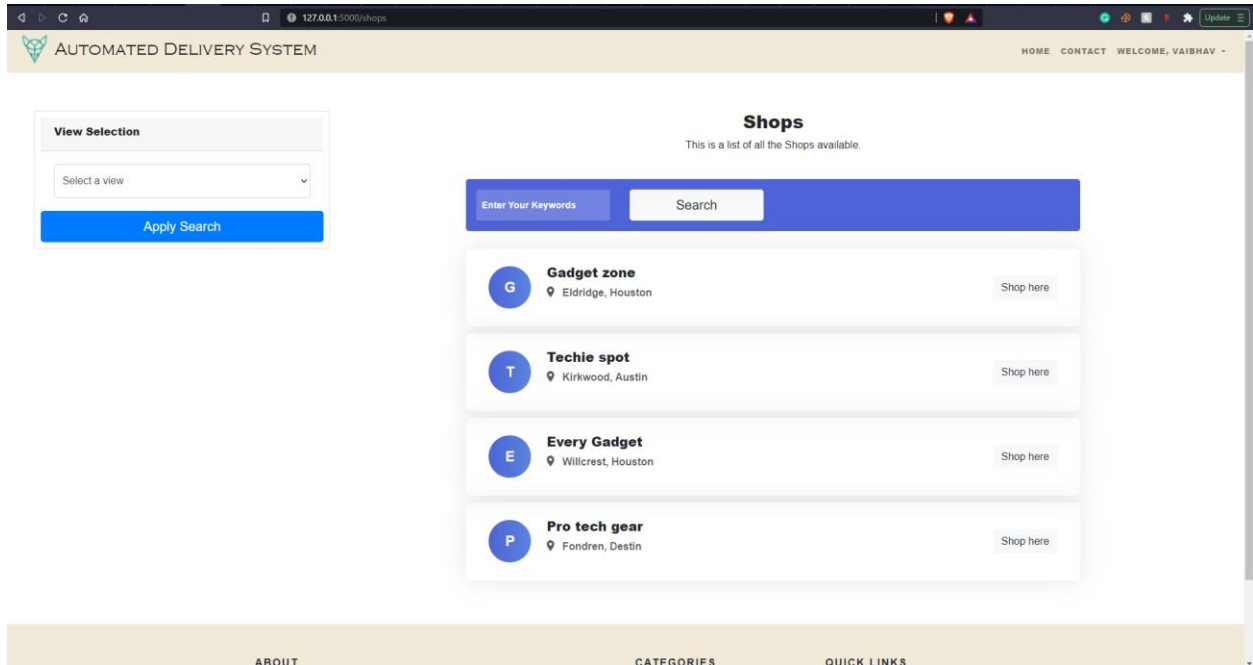
CATEGORIES
UI Design

QUICK LINKS
About Us

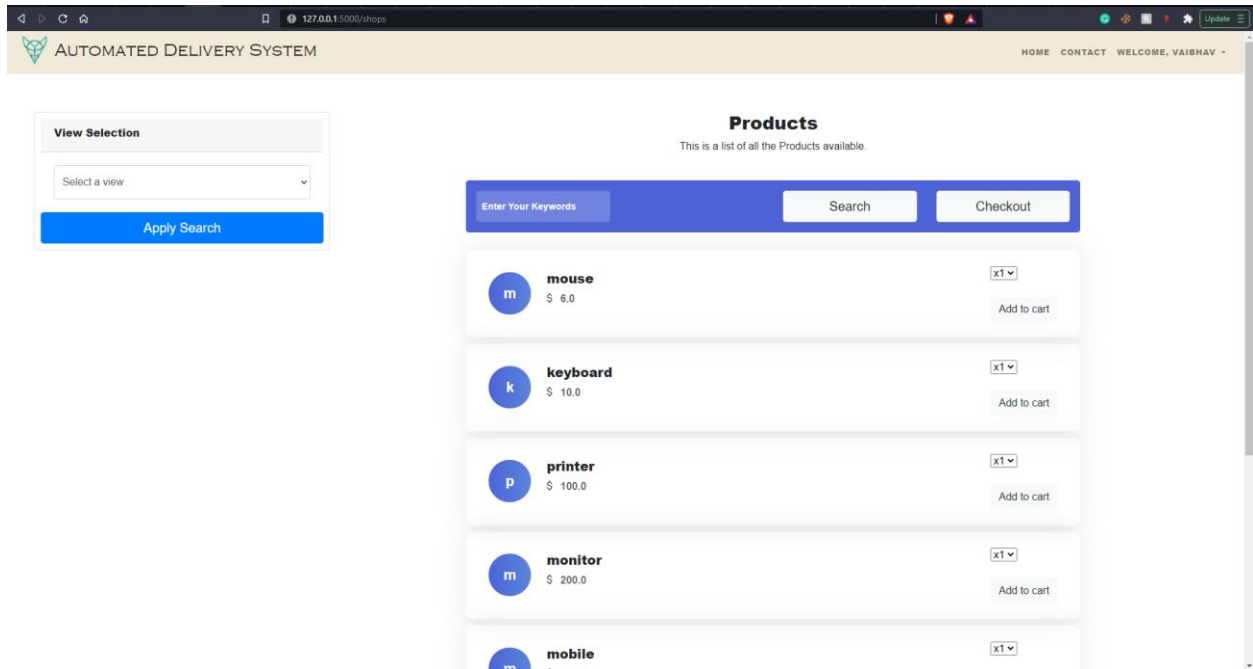
5. Customer home page:



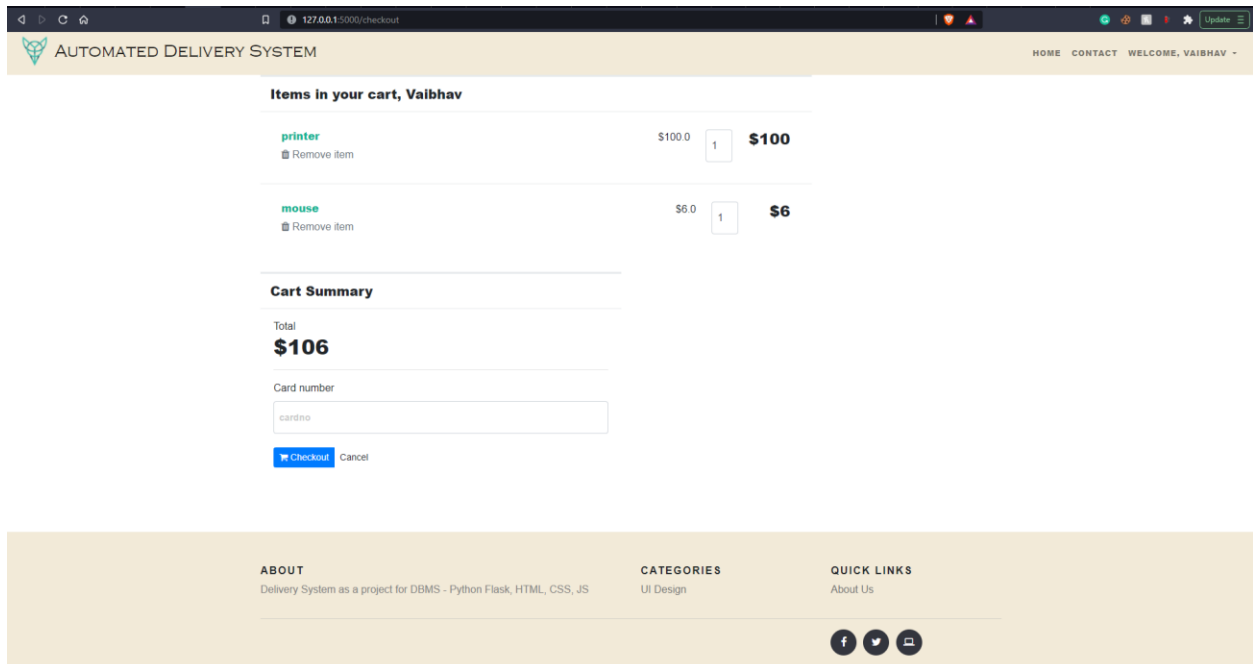
6. Customer shops view:



7. Customer products view (filtered by shops or all products):



8. Checkout and payment for customers:



9. Mailing services by customer to other customers:

The screenshot shows the 'Courier Service' page of the 'Automated Delivery System'. The page header includes the system name and navigation links: HOME, CONTACT, and WELCOME, VAIBHAV. The main heading is 'Courier Service' with a subtitle: 'This is a list of all the customers available to send couriers to'. Below this is a search bar with a blue 'Enter Your Keywords' button and a white 'Search' input field. The results are displayed in a list of four customer cards, each with a blue circular profile picture containing a white letter, the customer's name, their location, and a 'Schedule' button.

Profile	Name	Location	Action
V	Vaibhav	Houston	Schedule
J	Juhi	Houston	Schedule
J	John	Austin	Schedule
S	Smith	Houston	Schedule


The footer contains three links: ABOUT, CATEGORIES, and QUICK LINKS.

10. Purchase history of customer:

The screenshot shows the 'History of Vaibhav' page of the 'Automated Delivery System'. The page header includes the system name and navigation links: HOME, CONTACT, and WELCOME, VAIBHAV. The main heading is 'History of Vaibhav' with a subtitle: 'This is a list of all the recent orders'. Below this is a search bar with a blue 'Enter Order ID' button and a white 'Search' input field. The results are displayed in a list of three order cards, each with a blue circular profile picture containing a white letter, the item name, the order ID, the price, and buttons for 'Details' and 'Feedback'.

Profile	Item	Order ID	Price	Actions
V	keyboard	O-801	\$ 25	Details Feedback
V	mobile	O-802	\$ 25	Details Feedback
V	mouse	O-803	\$ 25	Details Feedback

11. Feedbacks on purchased products by customers:



Feedback

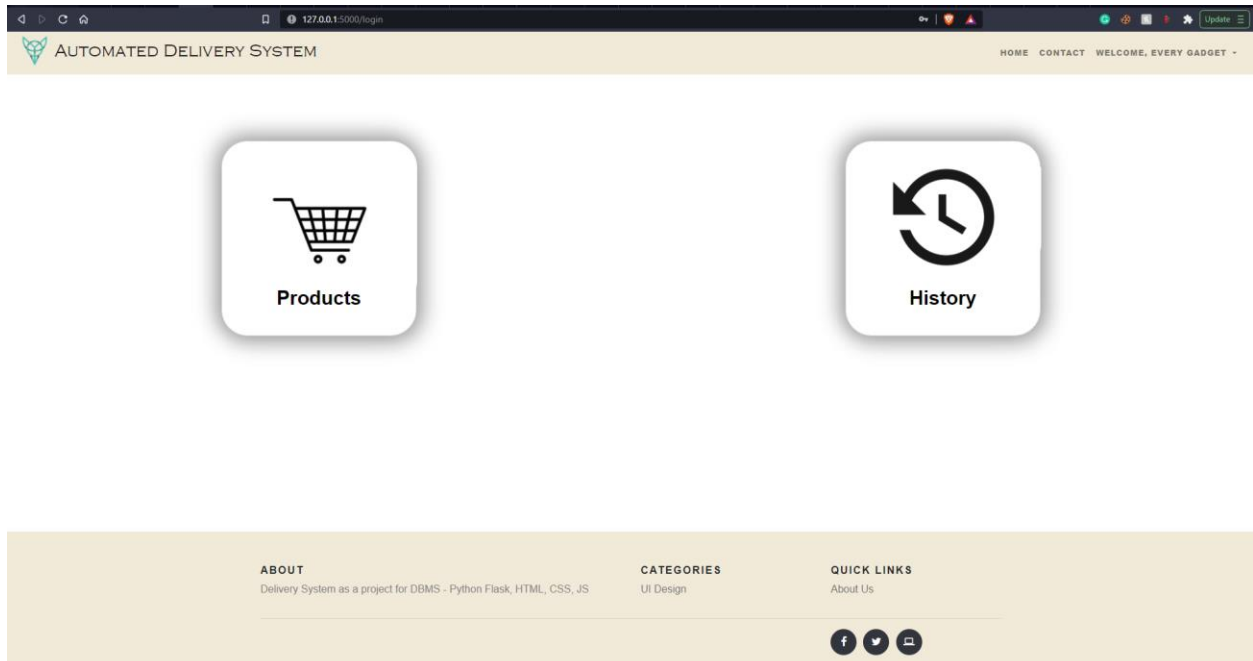
Typically it takes 2 mins to write a feedback

Order ID: O-801

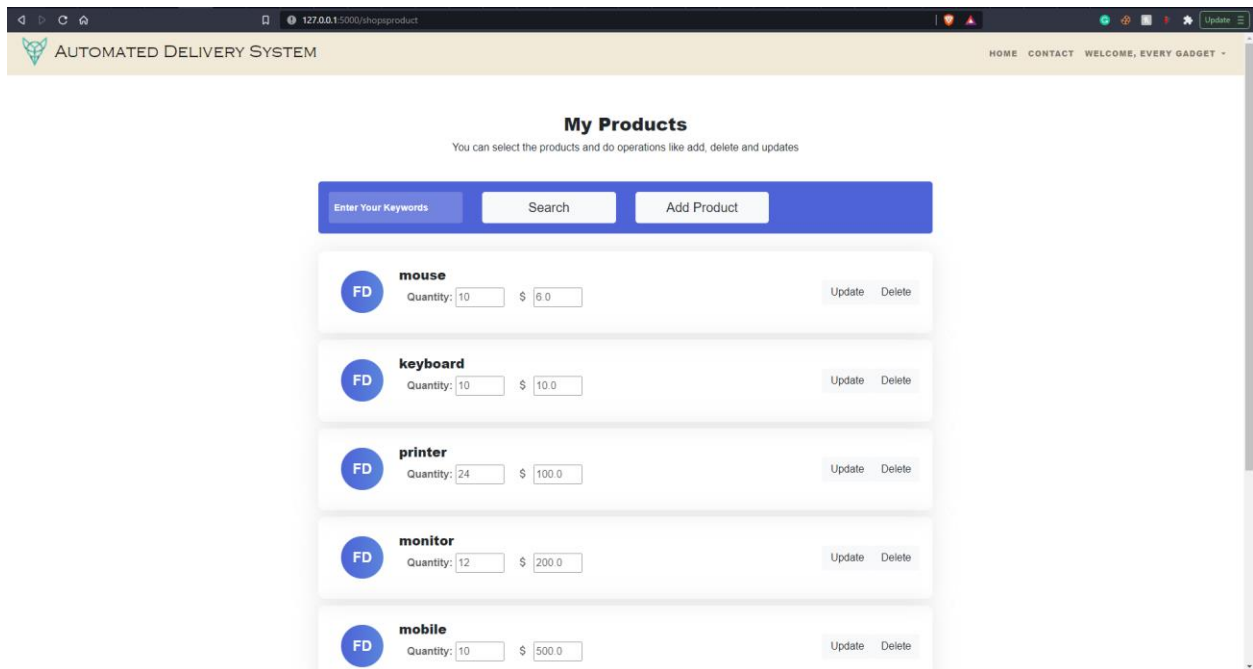
Message:

[Send Feedback](#)

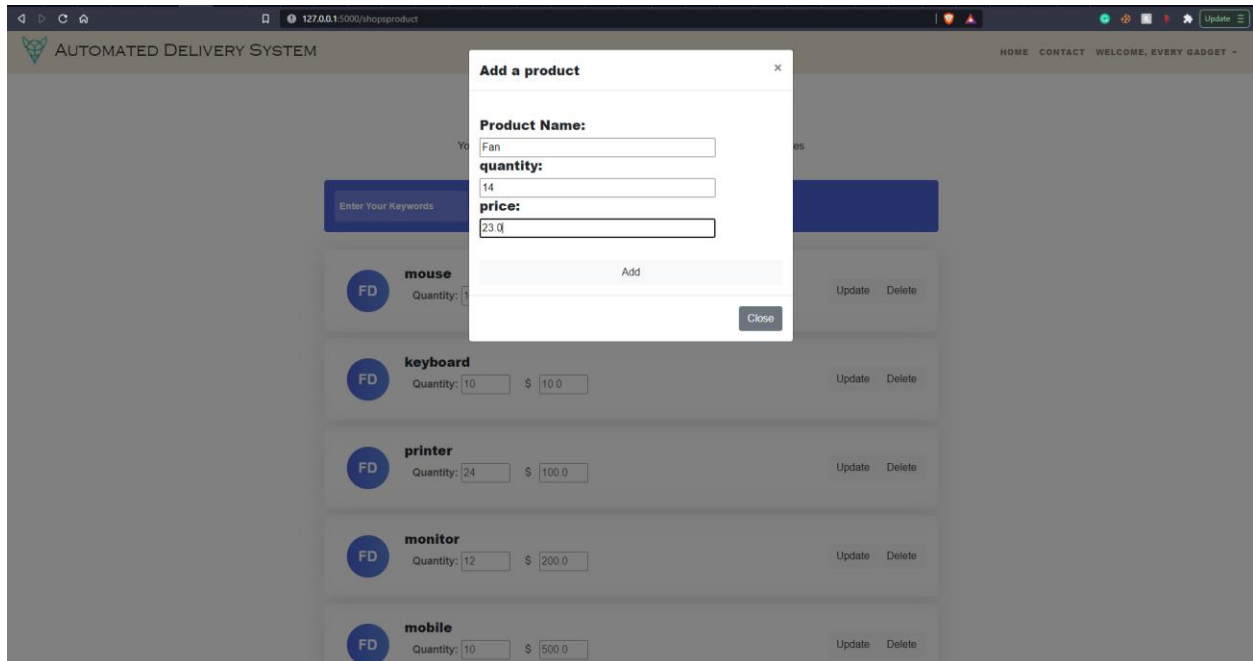
12. Shops home page:



13. Shops product listing page (add, update, delete products from listing):



14. Adding products to the listings by shop:



15. Shops' history of sold products:

The screenshot shows a web browser window with the URL `127.0.0.1:5000/shophistory`. The page header for the 'Automated Delivery System' includes a logo, navigation links (HOME, CONTACT, WELCOME, EVERY GADGET), and an 'Update' button. The main heading is 'History of Every Gadget' with the subtitle 'This is a list of all the recent orders'. Below this is a search bar with a blue 'Enter Order ID' button and a 'Search' input field. The results show two identical entries for a 'mouse' with Order ID: O-803. Each entry has a circular icon with a 'V' and a 'Details' button. The footer contains three columns: 'ABOUT' (Delivery System as a project for DBMS - Python Flask, HTML, CSS, JS), 'CATEGORIES' (UI Design), and 'QUICK LINKS' (About Us). Social media icons for Facebook, Twitter, and LinkedIn are at the bottom right.

History of Every Gadget
This is a list of all the recent orders

Enter Order ID Search

V **mouse**
Order ID: O-803 Details

V **mouse**
Order ID: O-809 Details

ABOUT
Delivery System as a project for DBMS - Python Flask, HTML, CSS, JS

CATEGORIES
UI Design

QUICK LINKS
About Us

f t in

16. Live updates on delivery status view by shop's account:

The screenshot shows a web browser window with the URL `127.0.0.1:5000/shophistory`. The page header is identical to the previous screenshot. The main heading is 'Tracking Order Status' with the subtitle 'This is a delivery schedule for Delivery ID: D-203'. Below this is a search bar with a blue 'Enter Tracking ID' button and a 'Search' input field. The results show a single entry for 'Delivery ID: D-203'. It includes a circular icon with 'FD', a dropdown menu showing 'Delivered', and a table with columns 'TrackingID', 'Source', and 'Destination'. The table data shows 'T-502', 'Destin', and 'Houston'. The footer is identical to the previous screenshot.

Tracking Order Status
This is a delivery schedule for Delivery ID: D-203

Enter Tracking ID Search

FD **Delivery ID: D-203**
TrackingID: T-502 Source: Destin Destination: Houston
Delivered

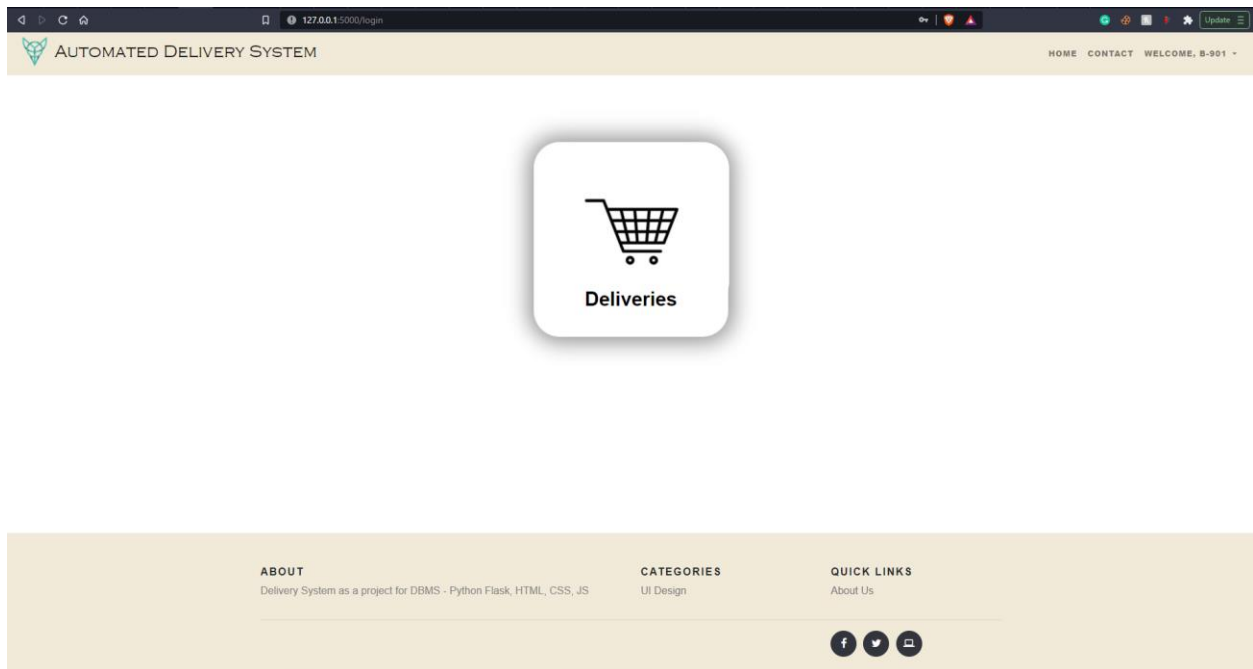
ABOUT
Delivery System as a project for DBMS - Python Flask, HTML, CSS, JS

CATEGORIES
UI Design

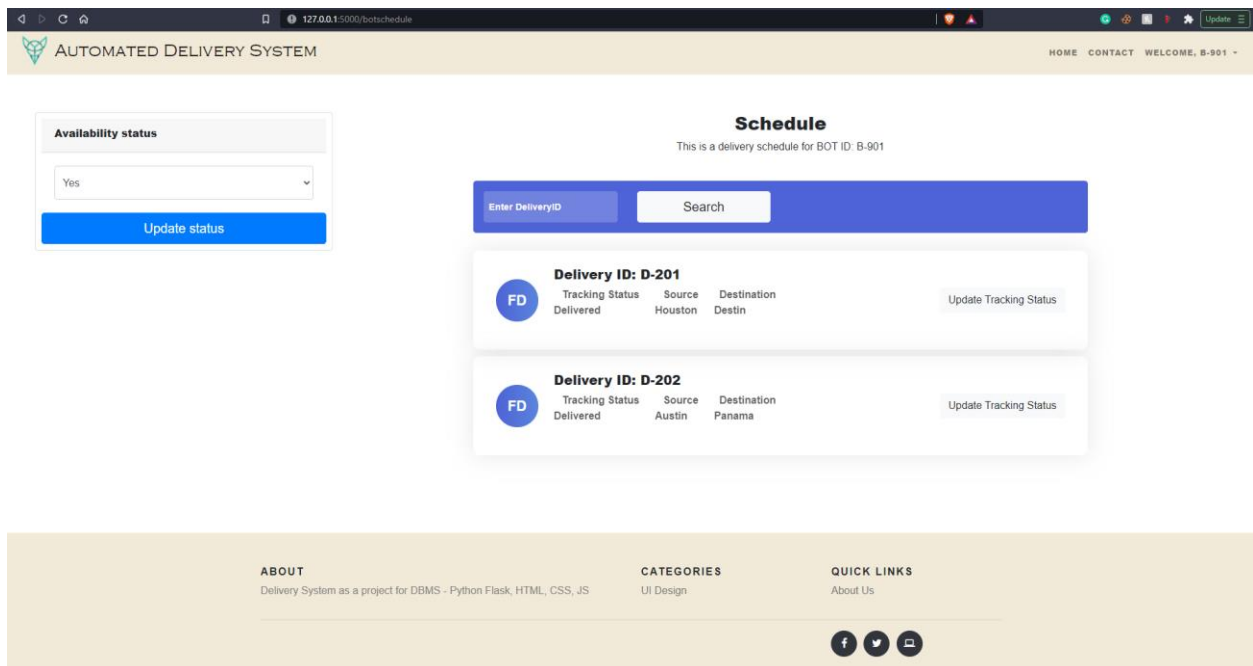
QUICK LINKS
About Us

f t in

17. Delivery person/bot's home page:



18. Scheduling when any order is placed to available bots, and an options to change the availability of it:



19. Updating the tracking status of the deliveries by the delivery person/bot:

127.0.0.1:5000/botschedule

AUTOMATED DELIVERY SYSTEM

HOME CONTACT WELCOME, B-901

Tracking Order Status

This is a delivery schedule for Delivery ID: D-201

Enter Tracking ID Search

Delivery ID: D-201

TrackingID: FD Source: Houston Destination: Destin

Delivered Update

ABOUT
Delivery System as a project for DBMS - Python Flask, HTML, CSS, JS

CATEGORIES
UI Design

QUICK LINKS
About Us

f t

Databases:

1. Botlogin:

	botID	username	password
▶	B-901	roger	that
	B-902	walter	white
	B-903	mrX	123
	B-904	wat	son
*	NULL	NULL	NULL

2. Courier scheduling table:

	customerID	deliveryID	targetID
	C-101	D-205	C-102
	C-102	D-206	C-104
...	C-104	D-207	C-101
*	NULL	NULL	NULL

3. Custlogin:

	customerID	username	password
▶	C-101	user1	root
	C-102	john1	welcome
	C-103	juhi	hello
	C-104	smith	james
*	NULL	NULL	NULL

4. Customer:

	customerID	Name	Street	City	Zip
▶	C-101	Vaibhav	Blvd	Houston	77057
	C-102	Juhi	Woodville	Houston	77077
	C-103	John	Eldridge	Austin	77023
	C-104	Smith	Willcrest	Houston	77067
*	NULL	NULL	NULL	NULL	NULL

5. Delivery:

	deliveryID	time	deliverytype	source	destination
	D-201	12:34:00	mail	Houston	Destin
	D-202	10:22:00	mail	Austin	Panama
	D-203	10:44:00	Shop	Destin	Houston
	D-204	12:34:00	Shop	Houston	Austin
	D-205	12:34:00	shop	Houston	Clear Lake
	D-207	20:06:03	shop	Dallas	Houston
	D-208	20:06:33	shop	Houston	Arlington
	D-210	20:07:57	shop	Austin	Dayton
	D-212	20:08:33	shop	Galvest...	Clear Lake
...	D-213	00:15:45	shop	Clear L...	Austin
*	NULL	NULL	NULL	NULL	NULL

6. Delivery_schedule:

10. Orderitems_delivery:

Result Grid			Filter Rows:
	OID	deliveryID	
▶	O-801	D-201	
	O-802	D-202	
	O-803	D-203	
	O-804	D-204	
*	NULL	NULL	

11. Payment:

Result Grid					Filter Rows:
	paymentID	customerID	Amount	Cardno	
▶	P-601	C-102	100	123234	
	P-602	C-101	50	214214	
	P-603	C-104	10	123424	
	P-604	C-103	200	421321	
	P-605	C-101	200	9999	
	P-608	C-101	200	2	
	P-609	C-101	100	2	
	P-610	C-101	200	2	
	P-611	C-101	100	5	
	P-612	C-101	100	4	
	P-614	C-101	106	45	
	P-616	C-101	100	2	
	P-617	C-101	0	2	
	P-618	C-101	200	2	
	P-620	C-101	100	2	
	P-622	C-101	200	2	
	P-623	C-101	106	2134	
*	NULL	NULL	NULL	NULL	

12. Phone_customer:

Result Grid			Filter Rows:
	phone	customerID	
▶	8326785566	C-101	
	8328896754	C-102	
	6782289878	C-103	
	2026673456	C-104	
*	NULL	NULL	

13. Product_shops:

Result Grid			Filter R
	shopID	productID	
▶	S-401	P-303	
	S-401	P-304	
	S-401	P-305	
	S-402	P-303	
	S-402	P-305	
	S-403	P-301	
	S-404	P-302	
*	NULL	NULL	

14. Products:

Result Grid					Filter Rows:
	productid	pname	quantity	price	
▶	P-301	mouse	10	6	
	P-302	keyboard	10	10	
	P-303	printer	24	100	
	P-304	monitor	12	200	
	P-305	mobile	10	500	
	P-306	Washing Machine	15	24	
*	NULL	NULL	NULL	NULL	

15. Shops:

Result Grid						Filter Rows:	Edit:
	shopID	name	street	city	zip		
▶	S-401	Gadget zone	Eldridge	Houston	77077		
	S-402	Techie spot	Kirkwood	Austin	77051		
	S-403	Every Gadget	Willcrest	Houston	77045		
	S-404	Pro tech gear	Fondren	Destin	77031		
*	NULL	NULL	NULL	NULL	NULL		

16. Shopslogin:

Result Grid				Filter Rows:
	ShopID	username	password	
▶	S-401	electro	hello	
	S-402	techie	welcome	
	S-403	gadget	dnd	
	S-404	techgear	gear123	
*	NULL	NULL	NULL	

17. Tracking:

Result Grid			
Filter Rows:			
	trackingID	status	location
▶	T-501	Delivered	Houston
	T-502	Pending	Austin
	T-503	Dispatched	California
	T-504	Delivered	Houston
•	NULL	NULL	NULL

Queries

- DDL and DML:

Domains:

1. Create domain price
constraint value-test check(value>0)
2. create domain amount
constraint value-test
check(amount>0)
3. create domain quantity
constraint test
check(quantity>0)

- DDL and DML:

```
4. CREATE TABLE `botlogin` (
  `botID` varchar(45) NOT NULL,
  `username` varchar(45) NOT NULL,
  `password` varchar(45) NOT NULL,
  PRIMARY KEY (`botID`,`username`,`password`),
  CONSTRAINT `botID` FOREIGN KEY (`botID`) REFERENCES `deliverybot` (`botID`) ON
  DELETE CASCADE ON UPDATE CASCADE
)
```

5. INSERT INTO `botlogin` VALUES ('B-901','roger','that'),('B-902','walter','white'),('B-903','mrX','123'),('B-904','wat','son');
6. CREATE TABLE `courier` (
 `customerID` varchar(45) NOT NULL,
 `deliveryID` varchar(45) NOT NULL,
 `targetID` varchar(45) NOT NULL,
 PRIMARY KEY (`customerID`,`deliveryID`,`targetID`),
 KEY `deliverycourier_idx` (`deliveryID`),
 KEY `targetcustomercourier_idx` (`targetID`),
 CONSTRAINT `customercourier` FOREIGN KEY (`customerID`) REFERENCES `customer` (`customerID`) ON DELETE CASCADE ON UPDATE CASCADE,
 CONSTRAINT `deliverycourier` FOREIGN KEY (`deliveryID`) REFERENCES `delivery` (`deliveryID`) ON DELETE CASCADE ON UPDATE CASCADE,
 CONSTRAINT `targetcustomercourier` FOREIGN KEY (`targetID`) REFERENCES `customer` (`customerID`) ON DELETE CASCADE ON UPDATE CASCADE
)
7. CREATE TABLE `custlogin` (
 `customerID` varchar(45) NOT NULL,
 `username` varchar(45) NOT NULL,
 `password` varchar(45) NOT NULL,
 PRIMARY KEY (`customerID`,`username`,`password`),
 CONSTRAINT `custlogin` FOREIGN KEY (`customerID`) REFERENCES `customer` (`customerID`) ON DELETE CASCADE ON UPDATE CASCADE
)
8. INSERT INTO `custlogin` VALUES ('C-101','user1','root'),('C-102','john1','welcome'),('C-103','juhi','hello'),('C-104','smith','james');
9. CREATE TABLE `customer` (


```

`customerID` varchar(45) NOT NULL,
`Name` varchar(45) NOT NULL,
`Street` varchar(45) NOT NULL,
`City` varchar(45) NOT NULL,
`Zip` int NOT NULL,
PRIMARY KEY (`customerID`)
)

```

10. INSERT INTO `customer` VALUES ('C-101','Vaibhav','Blvd','Houston',77057),('C-102','Juhi','Woodville','Houston',77077),('C-103','John','Eldridge','Austin',77023),('C-104','Smith','Willcrest','Houston',77067);

11. CREATE TABLE `delivery` (
`deliveryID` varchar(45) NOT NULL,
`time` time NOT NULL,
`deliverytype` varchar(45) NOT NULL,
`source` varchar(45) NOT NULL,
`destination` varchar(45) NOT NULL,
PRIMARY KEY (`deliveryID`)
)

12. INSERT INTO `delivery` VALUES ('D-201','12:34:00','mail','Houston','Destin'),('D-202','10:22:00','mail','Austin','Panama'),('D-203','10:44:00','Shop','Destin','Houston'),('D-204','12:34:00','Shop','Houston','Austin'),('D-205','12:34:00','shop','Houston','Clear Lake'),('D-207','20:06:03','shop','C-101','Houston'),('D-208','20:06:33','shop','C-101','Houston'),('D-210','20:07:57','shop','C-101','Houston'),('D-212','20:08:33','shop','C-101','Houston'),('D-213','00:15:45','shop','C-101','Houston');

13. CREATE TABLE `delivery_schedule` (
`deliveryID` varchar(45) NOT NULL,
`trackingID` varchar(45) NOT NULL,
`botID` varchar(45) NOT NULL,
PRIMARY KEY (`deliveryID`,`trackingID`,`botID`),
KEY `deliverschedule_idx` (`deliveryID`),
KEY `deliverytracking_idx` (`trackingID`),
KEY `deliverybot_idx` (`botID`),

```
CONSTRAINT `deliverschedule` FOREIGN KEY (`deliveryID`) REFERENCES  
`delivery` (`deliveryID`) ON DELETE CASCADE ON UPDATE CASCADE,  
CONSTRAINT `deliverybot` FOREIGN KEY (`botID`) REFERENCES `deliverybot`  
(`botID`) ON DELETE CASCADE ON UPDATE CASCADE,  
CONSTRAINT `deliverytracking` FOREIGN KEY (`trackingID`) REFERENCES  
`tracking` (`trackingID`) ON DELETE CASCADE ON UPDATE CASCADE  
)
```

14. INSERT INTO `delivery_schedule` VALUES ('D-201','T-501','B-902'),('D-202','T-501','B-901'),('D-203','T-502','B-903'),('D-204','T-503','B-904');

15. CREATE TABLE `deliverybot` (
 `botID` varchar(45) NOT NULL,
 `availability` varchar(45) NOT NULL,
 `location` varchar(45) NOT NULL,
 PRIMARY KEY (`botID`)
)

16. INSERT INTO `deliverybot` VALUES ('B-901','Yes','Houston'),('B-902','Yes','Houston'),('B-903','No','Austin'),('B-904','No','Austin');

17. CREATE TABLE `feedback` (
 `FID` varchar(45) NOT NULL,
 `OID` varchar(45) NOT NULL,
 `comment` varchar(45) NOT NULL,
 PRIMARY KEY (`FID`,`OID`),
 KEY `feedback_idx` (`OID`)
)

18. INSERT INTO `feedback` VALUES ('F-331','D-203','Good'),('F-332','D-201','Excellent'),('F-333','D-204','Average'),('F-334','D-202','Poor');

19. CREATE TABLE `orderitems` (
 `OID` varchar(45) NOT NULL,
 `customerID` varchar(45) NOT NULL,
 `productID` varchar(45) NOT NULL,
 `name` varchar(45) NOT NULL,

```

`quantity` varchar(45) NOT NULL,
`datetime` datetime NOT NULL,
`paymentstatus` varchar(45) NOT NULL,
PRIMARY KEY (`OID`,`customerID`,`productID`),
KEY `customerorder_idx` (`customerID`),
KEY `productorder_idx` (`productID`),
CONSTRAINT `customerorder` FOREIGN KEY (`customerID`) REFERENCES
`customer` (`customerID`) ON DELETE CASCADE ON UPDATE CASCADE,
CONSTRAINT `productorder` FOREIGN KEY (`productID`) REFERENCES `products`
(`productid`) ON DELETE CASCADE ON UPDATE CASCADE
)

```

```

20. INSERT INTO `orderitems` VALUES ('O-801','C-102','P-302','keyboard','4','2020-
07-06 12:20:00','complete'),('O-802','C-102','P-305','mobile','2','2018-03-02
10:50:19','complete'),('O-803','C-103','P-301','mouse','2','2019-05-04
11:34:22','complete'),('O-804','C-101','P-303','printer','3','2018-03-02
10:45:09','Complete'),('O-806','C-104','P-304','monitor','1','2020-07-06
12:20:00','Complete'),('O-807','C-101','P-303','printer','1','2021-07-26
19:19:17','completed'),('O-808','C-101','P-303','printer','1','2021-07-26
23:54:35','completed'),('O-809','C-101','P-301','mouse','1','2021-07-27
00:15:15','completed');

```

```

21. CREATE TABLE `orderitems_delivery` (
  `OID` varchar(45) NOT NULL,
  `deliveryID` varchar(45) NOT NULL,
  PRIMARY KEY (`OID`,`deliveryID`),
  KEY `orderdelivery_idx` (`deliveryID`),
  CONSTRAINT `OID` FOREIGN KEY (`OID`) REFERENCES `orderitems` (`OID`) ON
DELETE CASCADE ON UPDATE CASCADE,
  CONSTRAINT `orderdelivery` FOREIGN KEY (`deliveryID`) REFERENCES `delivery`
(`deliveryID`) ON DELETE CASCADE ON UPDATE CASCADE
)

```

```

22. INSERT INTO `orderitems_delivery` VALUES ('O-801','D-201'),('O-802','D-
202'),('O-803','D-203'),('O-804','D-204');

```

```

23. CREATE TABLE `payment` (
  `paymentID` varchar(45) NOT NULL,

```

```

`customerID` varchar(45) NOT NULL,
`Amount` int NOT NULL,
`Cardno` int DEFAULT NULL,
PRIMARY KEY (`paymentID`,`customerID`),
KEY `paymentcustomer_idx` (`customerID`),
CONSTRAINT `paymentcustomer` FOREIGN KEY (`customerID`) REFERENCES
`customer` (`customerID`) ON DELETE CASCADE ON UPDATE CASCADE
)

```

24. INSERT INTO `payment` VALUES ('P-601','C-102',100,123234),('P-602','C-101',50,214214),('P-603','C-104',10,123424),('P-604','C-103',200,421321),('P-605','C-101',200,9999),('P-608','C-101',200,2),('P-609','C-101',100,2),('P-610','C-101',200,2),('P-611','C-101',100,5),('P-612','C-101',100,4),('P-614','C-101',106,45),('P-616','C-101',100,2),('P-617','C-101',0,2),('P-618','C-101',200,2),('P-620','C-101',100,2),('P-622','C-101',200,2),('P-623','C-101',106,2134);

25. CREATE TABLE `phone_customer` (
 `phone` bigint NOT NULL,
 `customerID` varchar(45) NOT NULL,
 PRIMARY KEY (`phone`,`customerID`),
 KEY `custphone_idx` (`customerID`),
 CONSTRAINT `custphone` FOREIGN KEY (`customerID`) REFERENCES `customer`
 (`customerID`) ON DELETE CASCADE ON UPDATE CASCADE
)

26. INSERT INTO `phone_customer` VALUES (8326785566,'C-101'),(8328896754,'C-102'),(6782289878,'C-103'),(2026673456,'C-104');

27. CREATE TABLE `product_shops` (
 `shopID` varchar(45) NOT NULL,
 `productID` varchar(45) NOT NULL,
 PRIMARY KEY (`shopID`,`productID`),
 KEY `shops_product_idx` (`shopID`),
 KEY `product_shops_idx` (`productID`),
 CONSTRAINT `product_shops` FOREIGN KEY (`productID`) REFERENCES
`products` (`productid`),
 CONSTRAINT `shops_product` FOREIGN KEY (`shopID`) REFERENCES `shops`
 (`shopID`) ON DELETE CASCADE ON UPDATE CASCADE

)

28. INSERT INTO `product_shops` VALUES ('S-401','P-303'),('S-401','P-304'),('S-401','P-305'),('S-402','P-303'),('S-402','P-305'),('S-403','P-301'),('S-404','P-302');

29. CREATE TABLE `products` (
 `productid` varchar(45) NOT NULL,
 `pname` varchar(45) NOT NULL,
 `quantity` int NOT NULL,
 `price` double NOT NULL,
 PRIMARY KEY (`productid`)
)

30. INSERT INTO `products` VALUES ('P-301','mouse',10,6),('P-302','keyboard',10,10),('P-303','printer',24,100),('P-304','monitor',12,200),('P-305','mobile',10,500),('P-306','Washing Machine',15,24);

31. CREATE TABLE `shops` (
 `shopID` varchar(45) NOT NULL,
 `name` varchar(45) NOT NULL,
 `street` varchar(45) NOT NULL,
 `city` varchar(45) NOT NULL,
 `zip` varchar(45) NOT NULL,
 PRIMARY KEY (`shopID`)
)

32. INSERT INTO `shops` VALUES ('S-401','Gadget zone','Eldridge','Houston','77077'),('S-402','Techie spot','Kirkwood','Austin','77051'),('S-403','Every Gadget','Willcrest','Houston','77045'),('S-404','Pro tech gear','Fondren','Destin','77031');

33. CREATE TABLE `shopslogin` (
 `ShopID` varchar(45) NOT NULL,
 `username` varchar(45) NOT NULL,
 `password` varchar(45) NOT NULL,
 PRIMARY KEY (`ShopID`,`username`,`password`),

```
CONSTRAINT `shopslogin` FOREIGN KEY (`ShopID`) REFERENCES `shops`  
(`shopID`) ON DELETE CASCADE ON UPDATE CASCADE  
)
```

```
34. INSERT INTO `shopslogin` VALUES ('S-401','electro','hello'),('S-  
402','techie','welcome'),('S-403','gadget','dnd'),('S-404','techgear','gear123');
```

```
35. CREATE TABLE `tracking` (  
    `trackingID` varchar(45) NOT NULL,  
    `status` varchar(45) NOT NULL,  
    `location` varchar(45) NOT NULL,  
    PRIMARY KEY (`trackingID`)  
)
```

```
36. INSERT INTO `tracking` VALUES ('T-501','Delivered','Houston'),('T-  
502','Pending','Austin'),('T-503','Dispatched','California'),('T-  
504','Delivered','Houston');
```