

POINT OF SALE WEB APPLICATION

GROUP FOUR

- . Jonaboina Laxmana Manesh
- . Narravula Haripriya
- . Kamidi Shruthi
- . Kaithepalli Radha Sai Durga
- . Aravind Gande
- . Sandiri Saivardhan Reddy
- . Paida Harini

Abstract:

Point of sale is a critical piece of a point of purchase, refers to the place where a customer executes the payment for goods or services and where sales taxes may become payable. It can be in a physical store, where pos terminals and systems are used to process card payments or a virtual sales point such as a computer or mobile electronic device.

Introduction:

We propose to build an point of sale system where employees will login with username and password and it will be directed to page where they will fill the mobile number of the customer and barcode of the product. In the further step the barcode respective with the product is retrieved from the database and then the employee selects the quantity of the item clicks on submit. The total bill is generated and then the employee selects checkout option with which it will be redirected to the page where he can generate other customer's bill.

Technologies used (or) Requirements:

Hardware Requirement:

1. Intel i3 and above Processor
2. 4 GB RAM and above
3. 500 GB Hard Disk

Software Requirement:

1. Mysql
2. Java
3. Html
4. CSS
5. Javascript
6. Apache Tomcat Server 8.5

MYSQL:

We used a MySQL database. The tables we created are barcodes and customers

In the barcodes table we have the fields like

- ❖ **Barcode id**- this is used to identify a particular item name
- ❖
- ❖ **Item name**- the name of the item which customer selected
- ❖
- ❖ **Price**-the total amount the customer spent on the products

In the customer table we have fields like

- ❖ **Item name**-the name of the item which customer selected
- ❖
- ❖ **Price**-the price of the item
- ❖
- ❖ **Quantity**-the quantity of the items selected by customer
- ❖
- ❖ **Total**-the total bill of the items customer purchased

JAVASCRIPT FILES USED:

Billing.jsp- Here in this file the barcode and the mobile number are entered by the employee

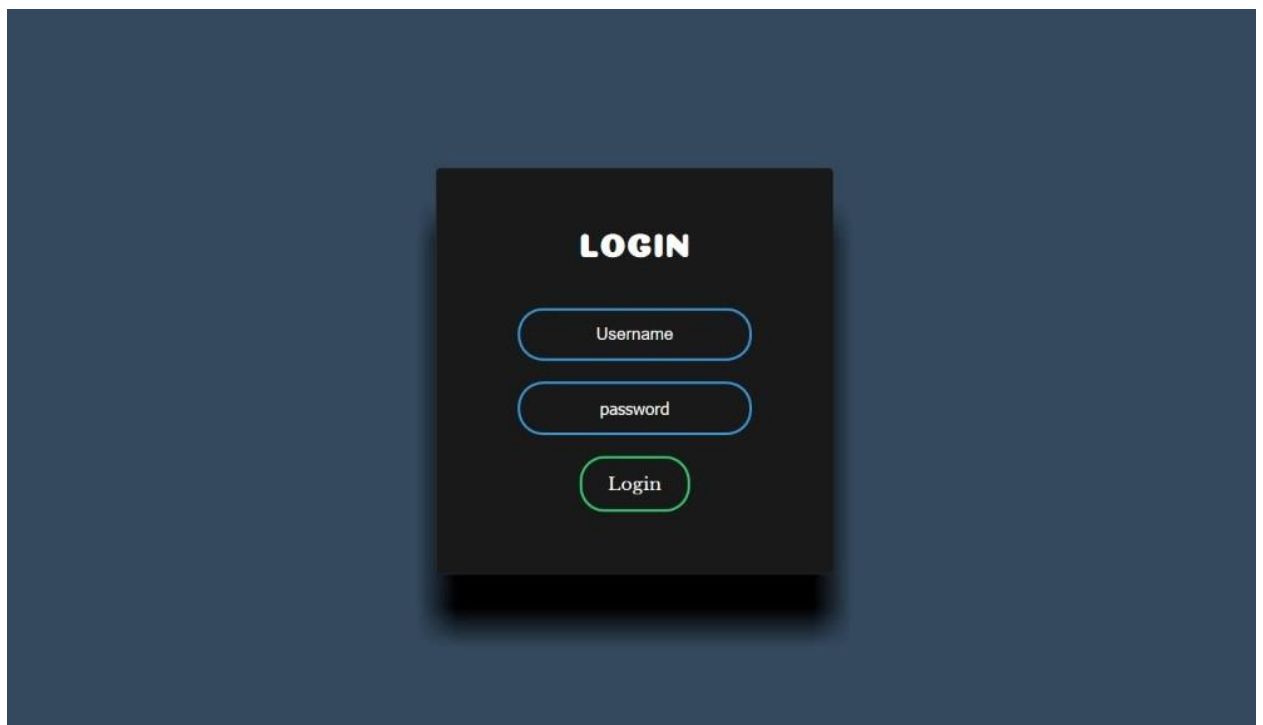
Generatebill.jsp- After giving the data the bill is generated and that is further stored in the database. This entire process is done in this file

HTML:

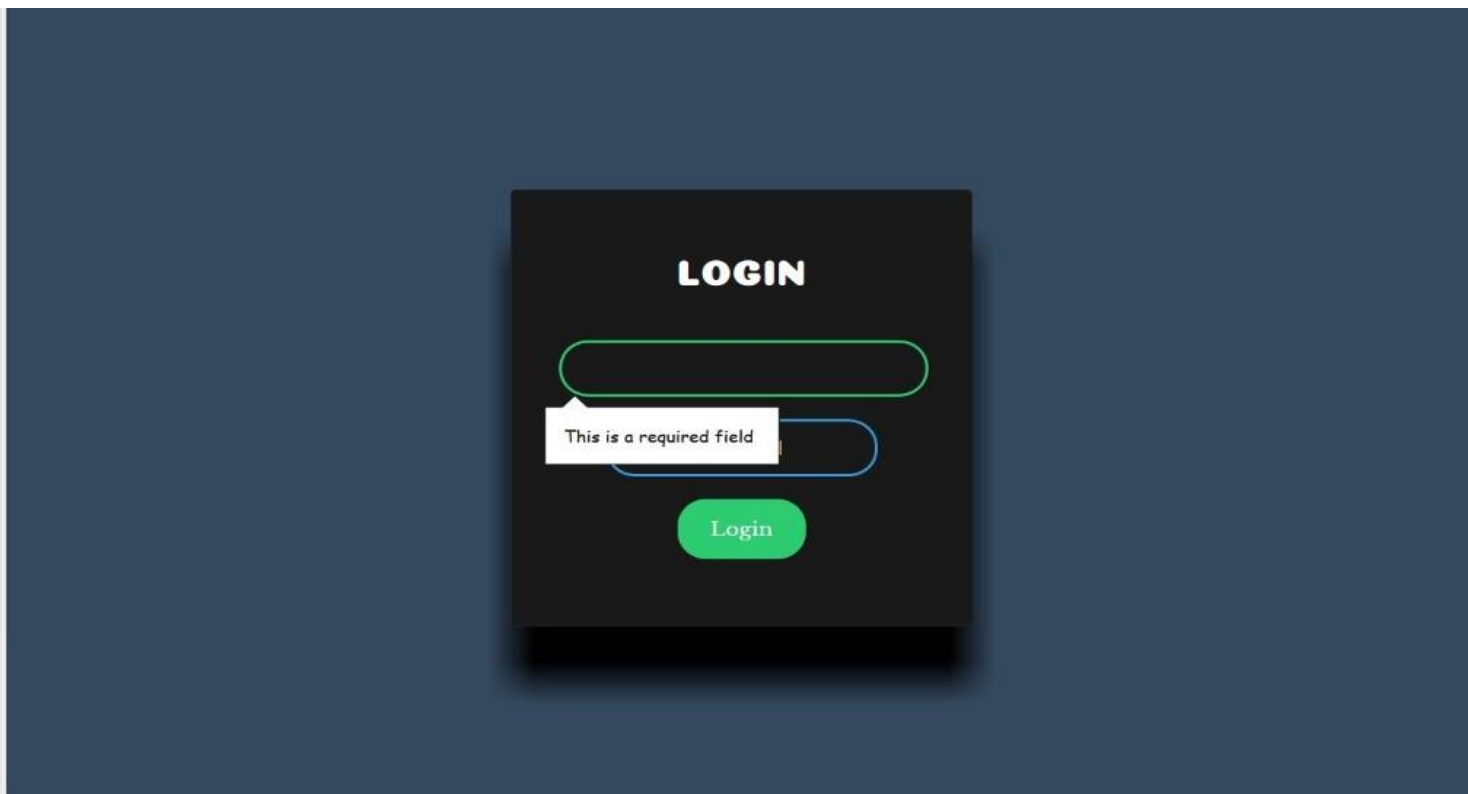
Login.html - this is a html file where the login and password are generated with a login button which are to be filled by the employee

FINAL OUTPUT(screenshots)

- This is how the login page is generated with a login.html file with username, password and login button.



- If the employee clicks on login without entering username or password then a warning is popped as “This is a required field” that means it’s compulsory to fill the required details to go to the next step



- Here the barcode of the item, mobile number of the customer and quantity of the item which customer needs are entered and then the employee selects “Add Data” if he wants to enter another item and in the following step the item name, price, quantity and the total price is generated.

Barcode:

Mobile:

Qty: 1

Add Data

Item	Price	Qty	Total
------	-------	-----	-------

Submit Data

- In this way the employee adds the items and finally selects on submit data if the customer shopping is done

http://localhost:8080/POSwebapp/billing.jsp?manesh&=root

Barcode: 20210524 x

Mobile:

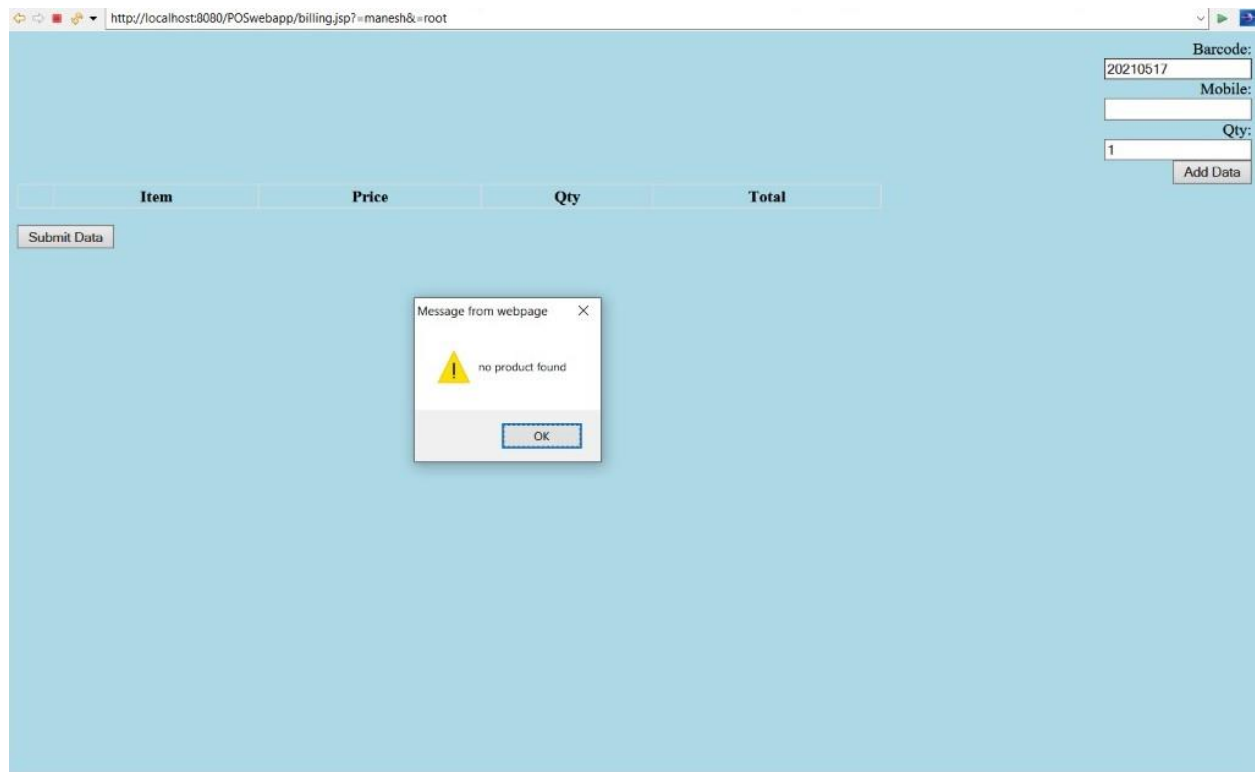
Qty: 1

Add Data

	Item	Price	Qty	Total
Remove	Rice	50	1	50
Remove	Shampoo	2	1	2
Remove	trimmer	1000	1	1000
Remove	scrub	199	1	199
Remove	fivestar	20	1	20

Submit Data

- If the barcode is entered incorrectly then “No product is found” warning is popped on the screen.



- The mobile number is also a required field here, if the mobile number is left blank then a warning is shown as that it can't be blank and must be filled

The screenshot shows a web application interface with a shopping cart and a warning message. The cart table is as follows:

	Item	Price	Qty	Total
Remove	Rice	50	1	50
Remove	Shampoo	2	1	2
Remove	trimmer	1000	1	1000
Remove	scrub	199		199
Remove	livestart	20		20

Below the table is a "Submit Data" button. To the right, there are input fields for "Barcode:" (20210524), "Mobile:" (blank), and "Qty:" (1), along with an "Add Data" button. A warning message box is displayed in the center, titled "Message from webpage", with a yellow warning icon and the text "mobile number cant be blank". The box has an "OK" button.

- This is an ideal way of entering total details like barcode, phone number and quantity by the employee.

testdb.jsp test1.jsp Bill.java test2.jsp Javamysqhte... Sqlinsertse... billing.jsp generatebill... Dynamically... login.html Dynamically... 2

http://localhost:8080/POSwebapp/billing.jsp?manesh&root

Barcode
20210524
Mobile
9182451235 x
Qty
1
Add Data

	Item	Price	Qty	Total
Remove	Rice	50	1	50
Remove	Shampoo	2	1	2
Remove	trimmer	1000	1	1000
Remove	scrub	199	1	199
Remove	fivestar	20	1	20

Submit Data

- This is how a bill is generated for the total items the customer has shopped and there is a “go back” option where the employee can go back and add items if the customer changes his mind and wants to add or remove something.

Product Name	Price	Qty	Total
Rice	50	1	50
Shampoo	2	1	2
trimmer	1000	1	1000
scrub	199	1	199
fivestar	20	1	20
Total			1271.0

[go back](#)

- The entire list of items with barcodes and prices are stored in our database.

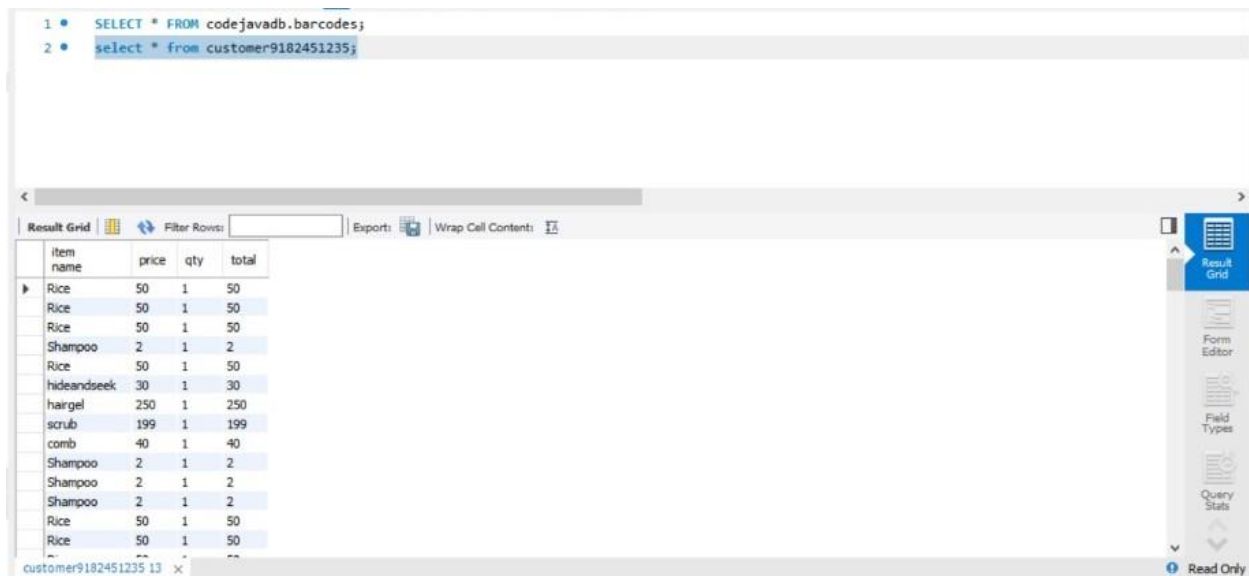
1 • `SELECT * FROM codejavadb.barcodes;`

id	item name	price
20210518	Rice	50
20210519	Shampoo	2
20210520	Soap	69
20210521	Brush	16
20210522	Bodylotion	200
20210523	hairoil	59
20210524	Srcub	199
20210525	Comb	25
20210526	Perfume	250
20210527	Biscuit	15
20210528	Fivestar	20
20210529	Barone	25
20210530	perk	10
20210531	Hairdbs	25
20210532	Lipbalm	30
20210533	Haigel	100
20210534	Trimmer	1000

barcodes 1 x

Apply Revert

- By giving the phone number of the customer in MySQL command the entire bill generated can be seen here and stored in the database for future reference.



The screenshot shows a MySQL query tool interface. At the top, a query editor contains two SQL statements:

```
1 • SELECT * FROM codejavadb.barcodes;  
2 • select * from customer9182451235;
```

Below the query editor, a toolbar includes options for 'Result Grid', 'Filter Rows', 'Export', and 'Wrap Cell Content'. The main area displays a table of results for the second query. The table has four columns: 'item name', 'price', 'qty', and 'total'. The results are as follows:

item name	price	qty	total
Rice	50	1	50
Rice	50	1	50
Rice	50	1	50
Shampoo	2	1	2
Rice	50	1	50
hideandseek	30	1	30
hargel	250	1	250
scrub	199	1	199
comb	40	1	40
Shampoo	2	1	2
Shampoo	2	1	2
Shampoo	2	1	2
Rice	50	1	50
Rice	50	1	50

At the bottom left, a tab is labeled 'customer9182451235 13'. On the right side, a vertical toolbar contains icons for 'Result Grid', 'Form Editor', 'Field Types', and 'Query Stats'. A 'Read Only' status indicator is visible at the bottom right.

DIVISION of work:

Backend coding :

- LAXMANA MANESH
- HARIPRIYA

Frontend coding :

- Aravindh
- Sai vardhan

Database and Documentation:

- Radha sai durga
- Shruthi

