School of Computer Science and Engineering

Team ID: 4D01

Implementation Phase Report

Category: Application Development

Title: Supermarket Management System

List of Team Members:

1. Aditya Vikram – 01FE19BCS220
2. Rishab Jain – 01FE19BCS228
3. Yash Raj – 01FE19BCS229
4. Harshita Hiremath – 01FE19BCS235

Team Leader: Rishab Jain

Responsibilities: [express the initial responsibilities of each team member for the implementation phase]

Aditya Vikram

- Work on Django

Rishab Jain

- Work on SQL queries

Yash Raj

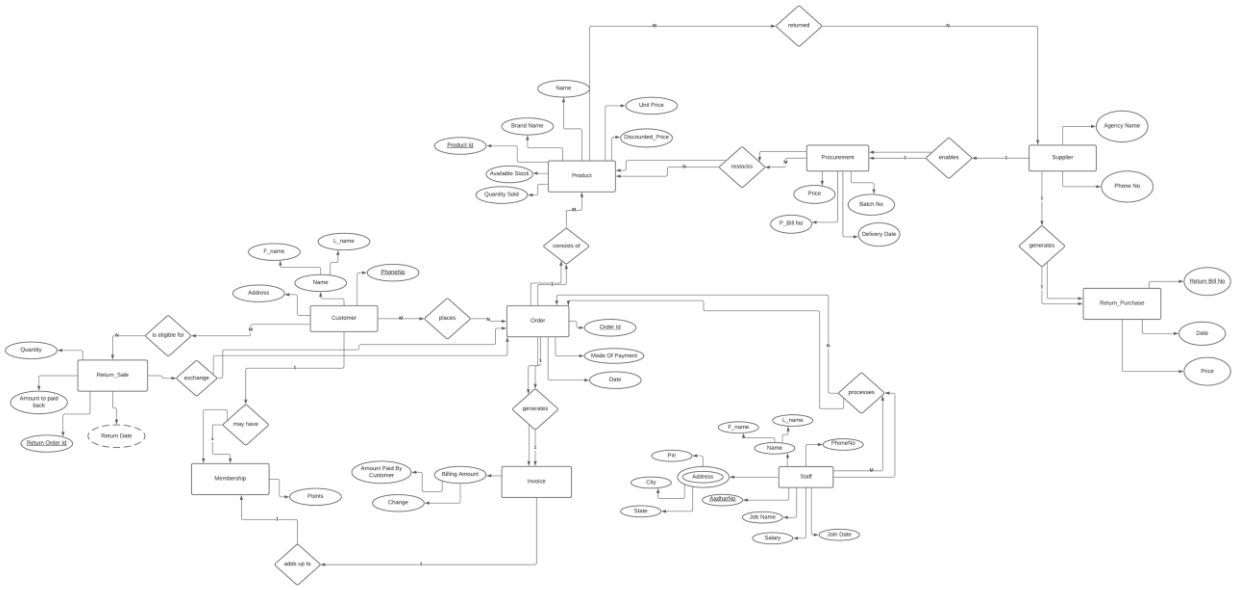
- Improvise UI

Harshita Hiremath

- Improvise ER and work on Django

Design updating: [Describe the changes (if any) that need to be made to the Entity-Relationship Diagram you have given in Design Phase before it can be implemented as tables in SQL. If you identify any changes, redraw a new E/R diagram.]

Relationship between STAFF & ORDER was established, where in STAFF processes ORDER.



Implementation Phase Questions to be answered

[Note: *Complete the application development, deployment and demonstration for the specified database. This section of the course project involves implementing the database designed in the Design Phase using a DBMS. You may use Oracle, MySQL or any other relational database package. You need to develop an application interface (front end) for the created database. You may use Django framework environment for designing the front end. Here, you need to deploy your application and demonstrate the various user views and the output of each variety of queries that the client needs. Also, you need to submit a document comprising of SQL statements, user interface screen shots, outputs, graphs. (You need to develop using any of the IDE tools that you have access to.)****]***

Normalization: [Any change in normal form as compared to design phase do include]

No changes were necessary.

Question1: Give the SQL statement(s) used to create the Oracle/MySQL database tables needed to implement the normalized relational schema.

```
create table customer
(
customer_ph_no bigint not null
primary key,
first_name varchar(10) null,
last_name varchar(10) null,
addr_line_1 varchar(20) null,
addr_line_2 varchar(20) null
);

create table product
(
product_id int not null
primary key,
product_name varchar(15) null,
brand varchar(15) null,
price int null,
available_stock int null,
quantity_sold int null,
discounted_price int null
);

create table staff
(
aadhar_number bigint not null
primary key,
staff_id int not null,
first_name varchar(20) null,
last_name varchar(20) null,
addr_line_1 varchar(20) null,
addr_line_2 varchar(20) null,
job_designation varchar(10) null,
salary int null,
join_date date null,
phone_number bigint null,
constraint staff_staff_id_uindex
unique (staff_id)
);
```

```
create table orders
(
order_id int not null
primary key,
order_date date null,
customer_ph_no bigint null,
mode_of_payment varchar(10) null,
staff_id int null,
constraint orders_customer_customer_ph_no_fk
foreign key (customer_ph_no) references customer (customer_ph_no)
on update cascade on delete cascade,
constraint orders_staff_staff_id_fk
foreign key (staff_id) references staff (staff_id)
on update cascade on delete cascade
);

create table invoice
(
order_id int null,
amount_paid int null,
change_generated int null,
constraint invoice_orders_order_id_fk
foreign key (order_id) references orders (order_id)
on update cascade on delete cascade
);

create table membership
(
customer_ph_no bigint null,
order_id int null,
pts_added_or_redeemed int null,
constraint membership_customer_customer_ph_no_fk
foreign key (customer_ph_no) references customer (customer_ph_no)
on update cascade on delete cascade,
constraint membership_orders_order_id_fk
foreign key (order_id) references orders (order_id)
on update cascade on delete cascade
);

create table ordered_items
(
order_id int null,
product_id int null,
quantity int null,
```

```
constraint ordered_items_orders_order_id_fk
foreign key (order_id) references orders (order_id)
on update cascade on delete cascade,
constraint ordered_items_product_product_id_fk
foreign key (product_id) references product (product_id)
on update cascade on delete cascade
);

create table sales_return
(
order_id int null,
product_id int null,
quantity int null,
amount_to_pay int null,
replacement_order_id int null,
constraint sales_return_orders_order_id_fk
foreign key (order_id) references orders (order_id)
on update cascade on delete cascade,
constraint sales_return_orders_order_id_fk_2
foreign key (replacement_order_id) references orders (order_id)
on update cascade on delete cascade,
constraint sales_return_product_product_id_fk
foreign key (product_id) references product (product_id)
on update cascade on delete cascade
);

create table supplier
(
supplier_ph_no bigint not null
primary key,
agency_name varchar(20) null,
addr_line_1 varchar(20) null,
addr_line_2 varchar(20) null
);

create table procurement
(
batch_no int not null
primary key,
bill_no varchar(10) null,
amount_to_pay int null,
supplier_ph_no bigint null,
delivery_date int null,
constraint procurement_bill_no_uindex
```

```
unique (bill_no),
constraint procurement_supplier_supplier_ph_no_fk
foreign key (supplier_ph_no) references supplier (supplier_ph_no)
on update cascade on delete cascade
);

create table procured_items
(
batch_no int null,
product_id int null,
quantity int null,
constraint procured_items_procurement_batch_no_fk
foreign key (batch_no) references procurement (batch_no)
on update cascade on delete cascade,
constraint procured _items_product_product_id_fk
foreign key (product_id) references product (product_id)
on update cascade on delete cascade
);

create table purchase_return
(
batch_no int null,
date date null,
amount_returned int null,
product_id int null,
constraint return_purchase_procurement_batch_no_fk
foreign key (batch_no) references procurement (batch_no)
on update cascade on delete cascade,
constraint return_purchase_product_product_id_fk
foreign key (product_id) references product (product_id)
on update cascade on delete cascade
);
```

Question2: Give the actual data stored in each table of the database. (real sample data)

Products Table

product_id	product_name	brand	Price	available_stock	quantity_sold	discounted_price
1	Jimjam	brittania	10	100	0	0
2	Cream & Onion	lays	20	100	0	0
3	Toothbrush	Pepso	40	100	0	0

		dent				
4	Chakki Atta	Aashirvad	300	100	0	0
5	Hairdye	Garnier	180	100	0	0
6	Maggi	Nestle	12	100	0	0
7	Dishwash	Vim	19	100	0	0
8	Toothpaste	Colgate	84	100	0	0
9	HealthDrink	Boost	730	100	0	0
10	Ketchup	Nestle	120	100	0	0
11	Lotion	Himalaya	215	100	0	0
12	Atta	Ashirvaad	60	100	0	0
13	Handwash	Lifebuoy	99	100	0	0
14	Toothbrush	Colgate	101	100	0	0
15	Biscuits	DarkFantasy	82	100	0	0
16	Biscuits	Oreo	58	100	0	0
17	Biscuits	Nice	24	100	0	0
18	Bhujia	Haldiram	175	100	0	0
19	PotatoChips	Bingo	20	100	0	0
20	Coffee	Bru	150	100	0	0
21	Tea	Red Label	265	100	0	0
22	Noodles	Sunfeast	123	100	0	0
23	Vermicelli	Mtr	59	100	0	0
24	Pasta	Disano	99	100	0	0
25	Honey	Dabur	250	100	0	0
26	Choco Spread	Disano	269	100	0	0
27	Hair Oil	Parachute	120	100	0	0
28	Shampoo	Dove	430	100	0	0
29	Shampoo	Nyle	256	100	0	0
30	Hair Dye	Godrej	120	100	0	0

31	Lotion	Nivea	192	100	0	0
32	Deo	Nivea	156	100	0	0
33	Deo	Axe	193	100	0	0
34	Sambar masala	Aashirvad	62	100	0	0
35	Salt	Tata	18	100	0	0
36	Oil	Safola	168	100	0	0
37	Cold drink	Minute maid	69	100	0	0
38	Cold drink	Pepsi	35	100	0	0

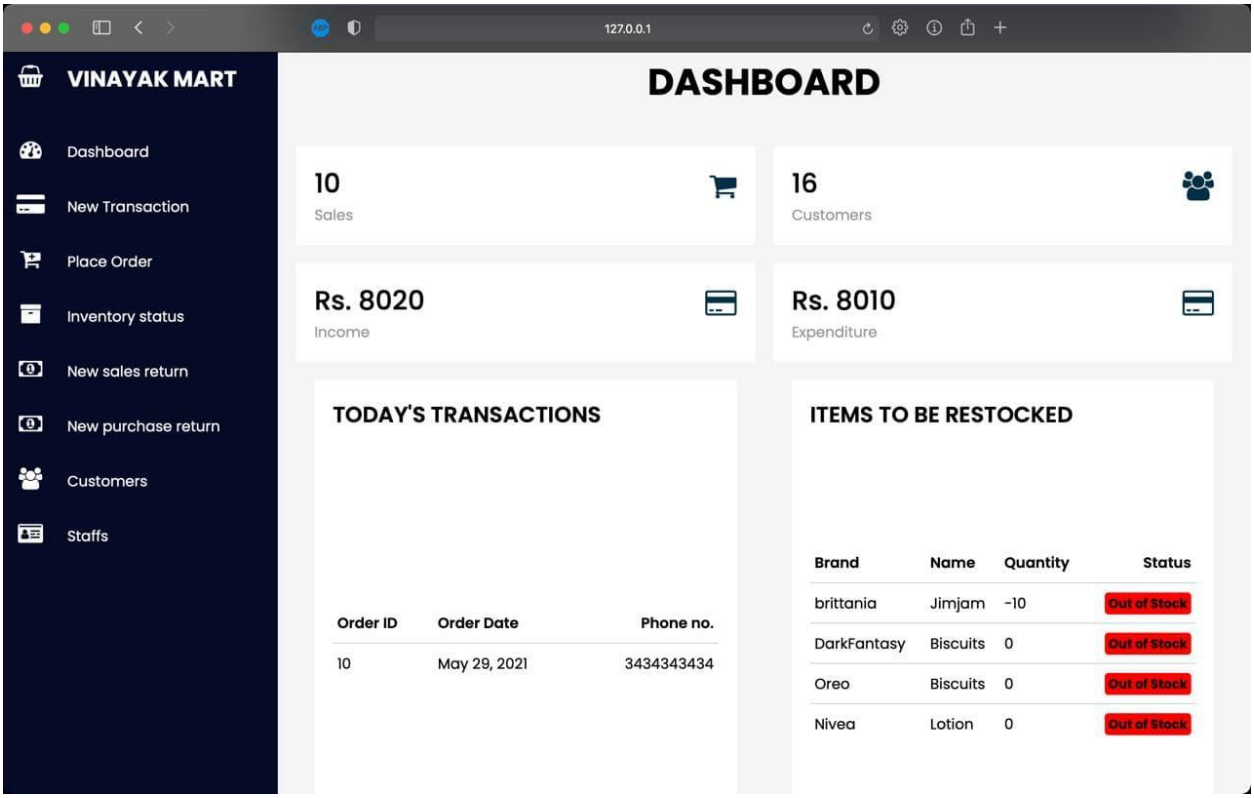
Supplier Table

supplier_ph_no	agency_name	addr_line_1	addr_line_2
8383020334	Sri Sai Goods	Dharwad	Arera Road
8939292923	Agarwal Goods	Dharwad	Nabad Nagar
9288628265	Sudarshan Entp	Dharwad	Teacher's Colony
9701001117	Raja wholesale	Hubli	Vidyanagar
9837382223	Kishore wholesale	Hubli	KIMS

Staff Table

Aadhar_no	staff_id	first_name	last_name	addr_line1	addr_line2	job_designation	salary	join_date	phone_no
121114849301	2	Rahul	Sharma	Shirupark	Hubli	cashier	10000	2019-09-24	8939949397
129823838491	3	Vidhi	Sanghvi	Gokul road	Hubli	cashier	10000	2019-05-29	9038381157
145861663728	1	Vishal	Singh	Vidyanagar	Hubli	cashier	15000	2017-04-13	8320193123
346513483739	4	Shlok	Kumar	KIMS	Dharwad	staff	12000	2018-08-12	9828282137
452314567382	5	Preeti	Sharma	Station road	Dharwad	staff	9000	2020-03-01	8339202121

Question3: Give the snapshots, description and SQL queries for each of the user interface forms for your application. (Create the front end using Django and hook it up to the SQL.)



Total no.of sales :

select count(order_id) from invoice;

Income generated :

select total_sales AS (sum(amount_paid) – sum(change_generated)) from invoice;

Today's transactions:

select order_id, amount as (amount_paid – change generated), customer_ph_no from order where order_date = curdate( );

Total no.of customers :

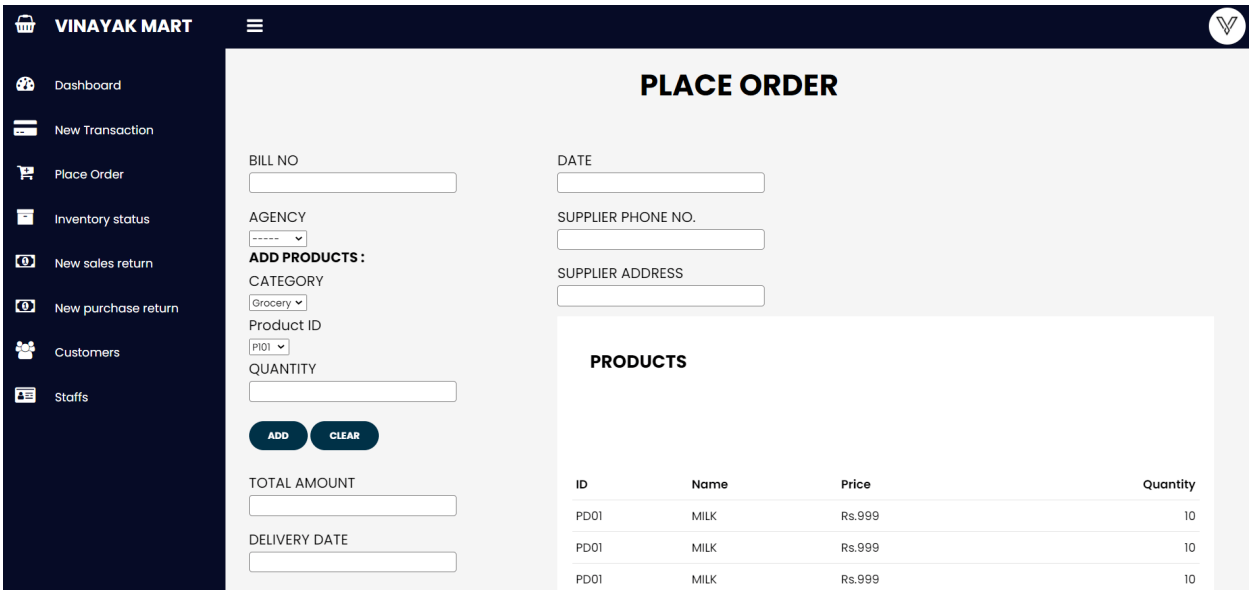
select count(distinct customer_ph_no) from order;

Expenditure :

select sum(amount_to_pay) from procurement;

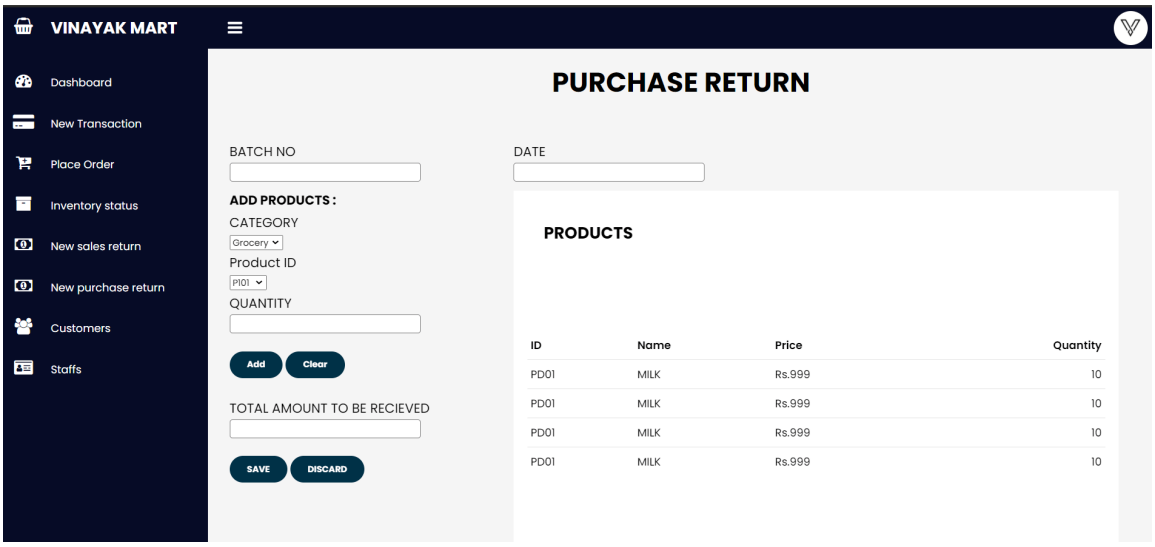
Products soon to be out of stock :

select product_name,product_id,available_stock AS quantity from products where available_stock<10;



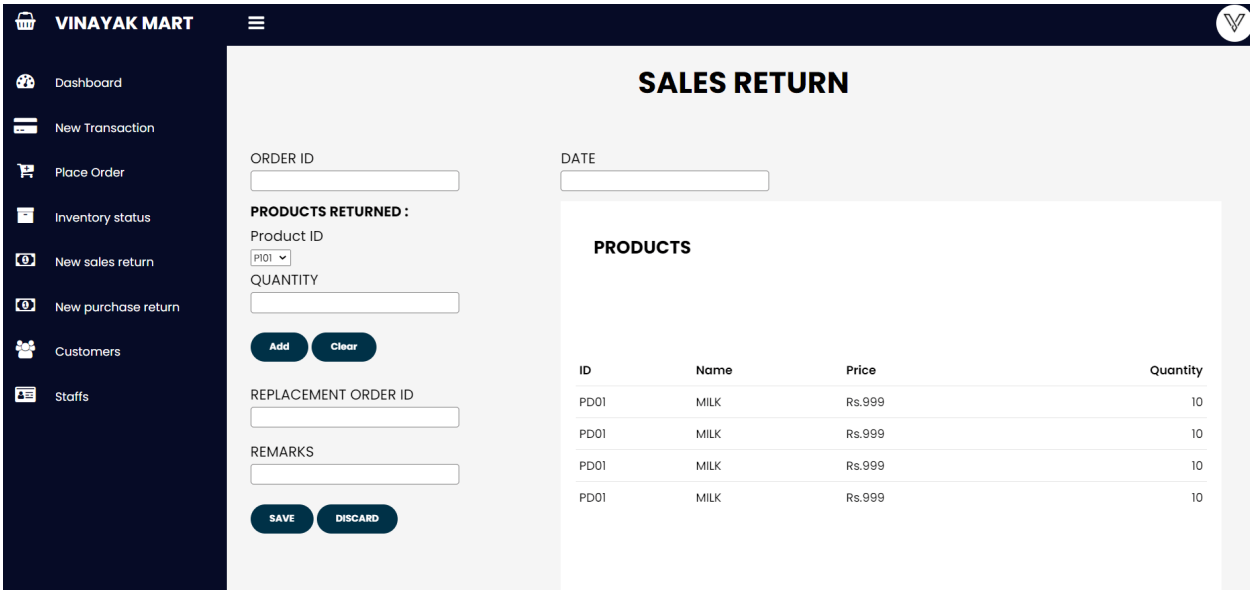
Place Order :

Once the products run out of stock, the client wishes to place an order and store the order details. All the entries will be typed in by the user and saved to the database with the help of Django framework.



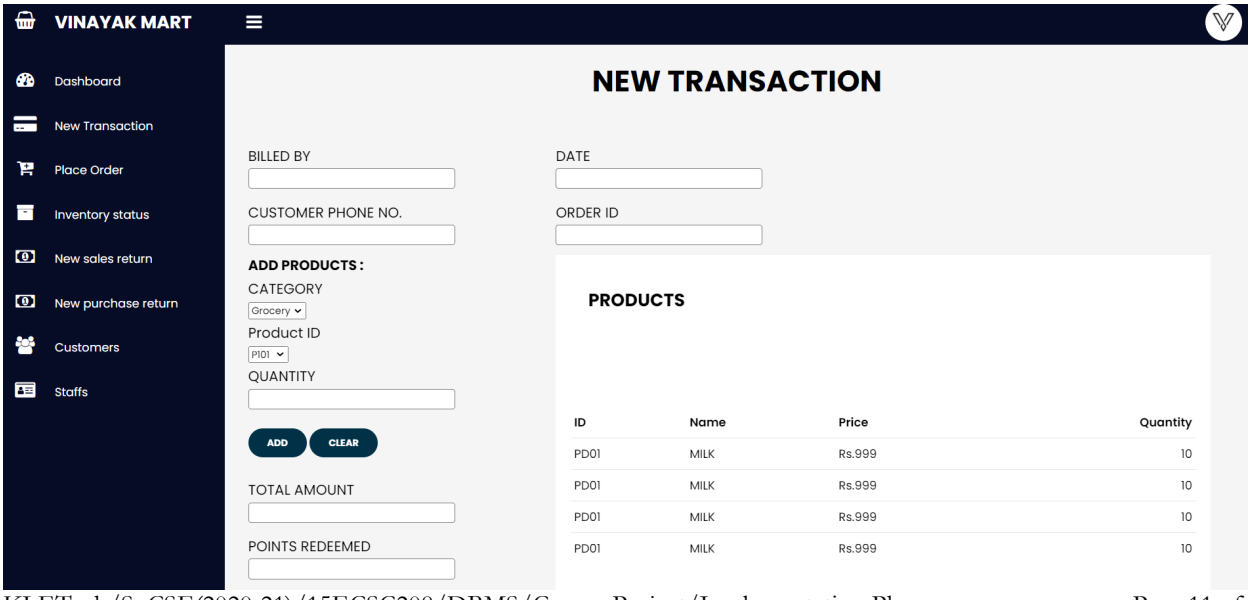
Purchase Return :

If at all the sales of a particular product doesn't cross the threshold, the client wishes to return the products back to the supplier. The receipt generated by the supplier will be filled into the UI form by the user and saved to the database with the help of Django framework.



Sales Return :

All the details except date and replacement order id will be filled in by the user. Date will be auto filled whereas replacement order id will be auto incremented and filled accordingly.



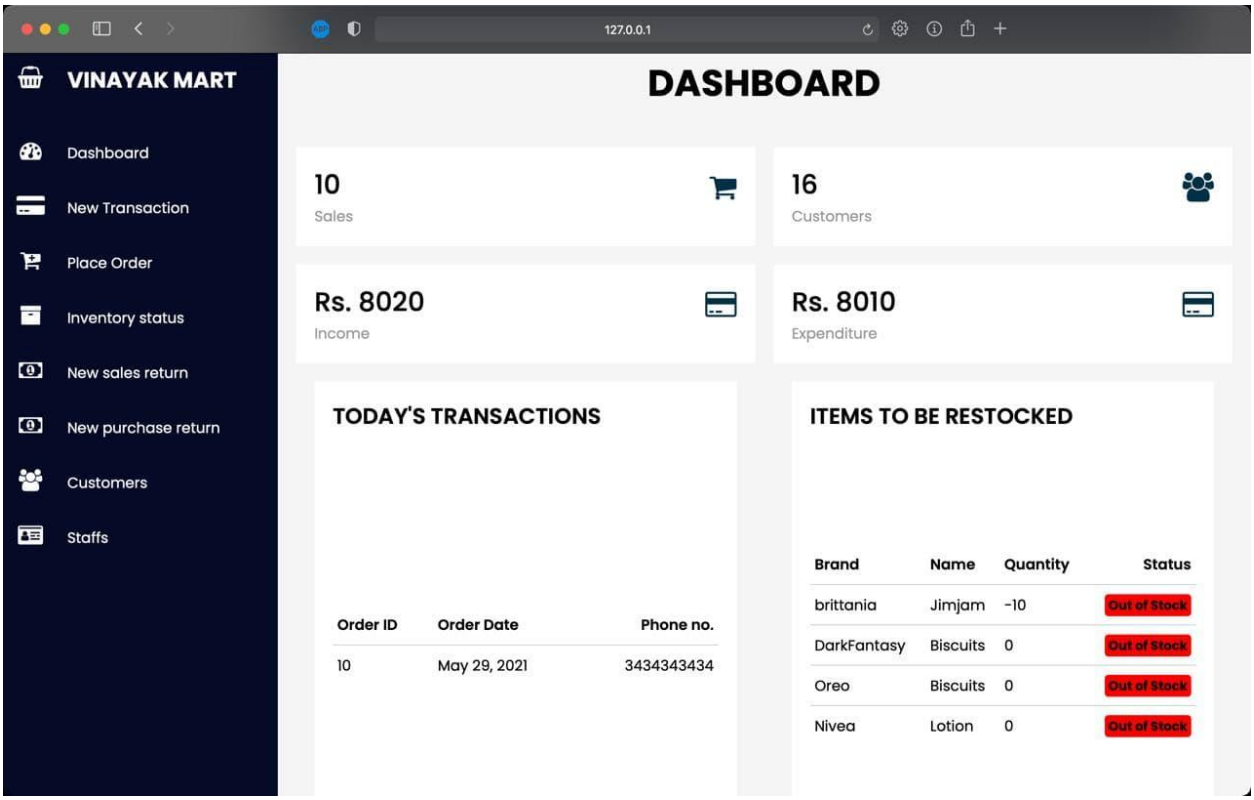
New Transaction :

All the details except order id, date and total amount will be filled in by the user. Date will be auto filled whereas order id will be auto incremented and filled accordingly. Total amount to be paid will be generated by an SQL query.

Total Amount :

select sum(o.amount*o.qty) from products p, order o where p.product_id=o.product_id;

Question4: Give all possible final bill reports/other forms of ledger reports summarized etc and graphs obtained by your application.



Submission Date: 29th May 2021