

Liquor's Castle

You are hot, come we will give you a shot

In the present, quickly changing business conditions when everything is being digitalized, be it shopping, banking, all kinds of online money transactions, entrance exam preparation, to make one's business more profitable, they need to come online as it makes your products more accessible to customers.

Online shopping is a trend these days, and our project revolves around that. Our project provides the registered-verified (verified =legal age) users a catalog of liquor and its availability around them and, most importantly, helps them save their precious time. People can view whether a particular item is available or not in their region by using our web app.

Consider an example, you are visiting a new state in India on occasion, and you want an XYZ particular brand, So instead of going to every shop in search of your brand, you can use the web app to find the shop where you can get your brand.

Community plays an integral role in our project. We as a community believe in smart -responsible drinking and avoid drunk driving at all costs.

Our target audience is mainly people of legal age who consume alcohol (legal age of alcohol consumption varies across states in India). Our project also allows the supplier(the shop owner) to regulate his stocks and sales. We let the owner decide if the consumer is in the state to consume more alcohol or not. We want to make it easier for both the consumer and the seller, So, for better security, we give the seller the responsibility of verifying the age.

The database can also be used to do competitive analysis with other suppliers and with oneself to see the percentage change in sales from the last period. We are not focusing on this for now.

Stakeholders

Owner of Shop (Shopkeeper)

A shopkeeper can use our application in many ways, like managing supplies, customers, demand, stock, etc. The shopkeeper has its login as a shopkeeper in which he can change/alter the database for his shop. So, there are various queries given below for the owner.

- ❑ He can check the stock of his shop, which is a little bit easier for him by using our application rather than the pen-paper tradition.
- ❑ He can order stock from the suppliers.
- ❑ He can check its recent order.
- ❑ Suppliers can update the price of the item
- ❑ Suppliers can update availability status and details of availability.
- ❑ He can check the information of all the items in his shop.

Customer

A customer can make an account in our application through which he can perform many queries, and some of them are listed below: -

- ❑ He can search for his/her specific brand's availability.
- ❑ He can check nearby shops from where he can get his brand.
- ❑ He can place an order.
- ❑ He can view item availability status.
- ❑ He can search for items according to his budget

Supplier

Supplier plays a vital role in supplying goods, so to maintain proper supply, he can use our application. He also has an account in our database through which he can access his supplies in a better way and can serve him to society in a better way :-)

- ❑ He can add/remove brands to his stock.
- ❑ He can change the price of his stock.
- ❑ He can add/remove products or alter the quantity of his stock.
- ❑ He can check the availability of his stock
- ❑ He can do edit/update in "area of serving" attribute.

Government and Investigation departments

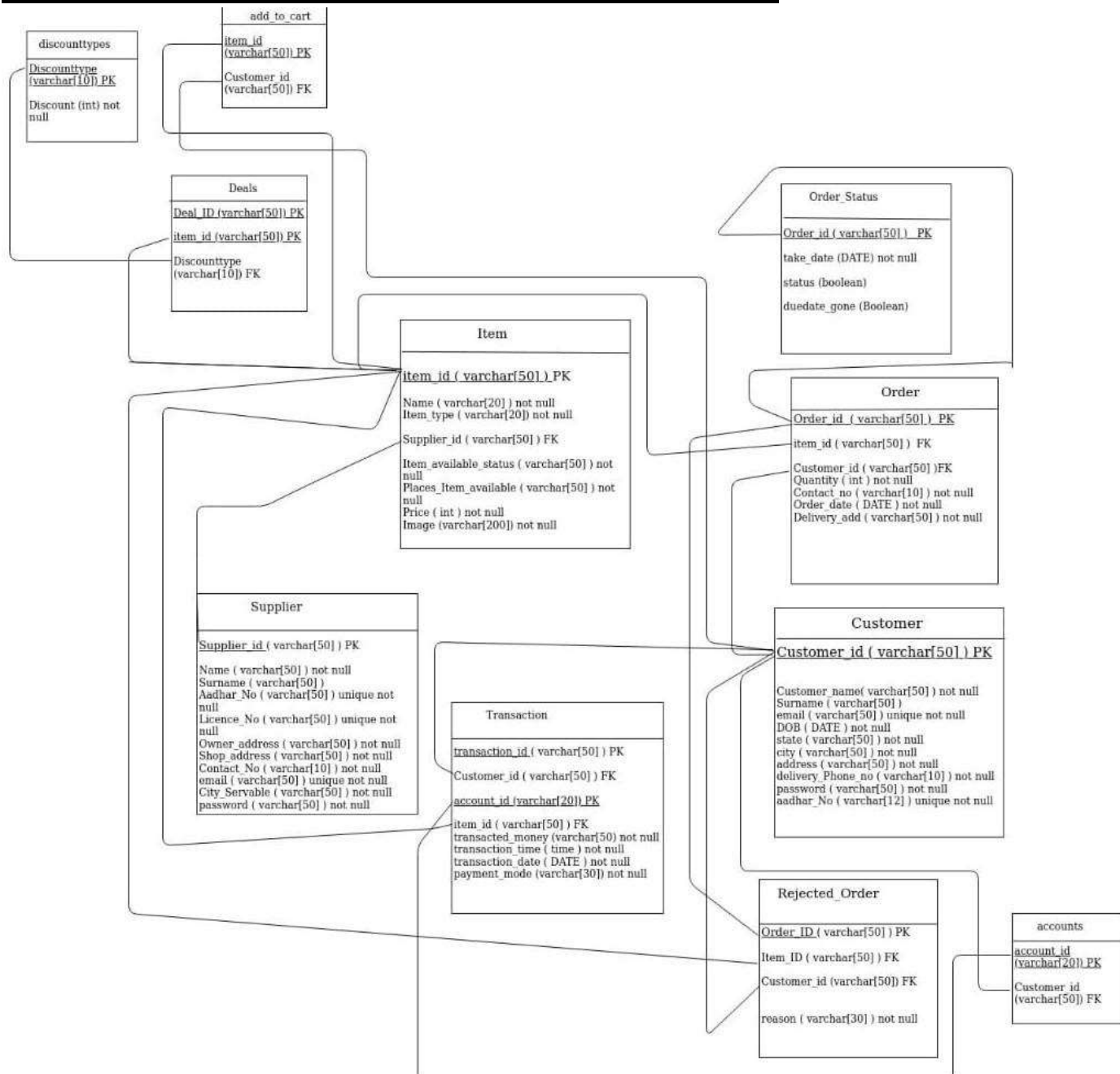
The government can also use our application to keep track of supplies on liquors, and it will help them to control smuggling to these items.

- 🔍 They can check the selling price of liquors.
- 🔍 They can ensure, dry states are not using this app
- 🔍 They can check Aadhar card No. or any id proof given by the customer.
- 🔍 They can access the licence id of all suppliers.
- 🔍 They can check any supplier's stock to control illegal activities.

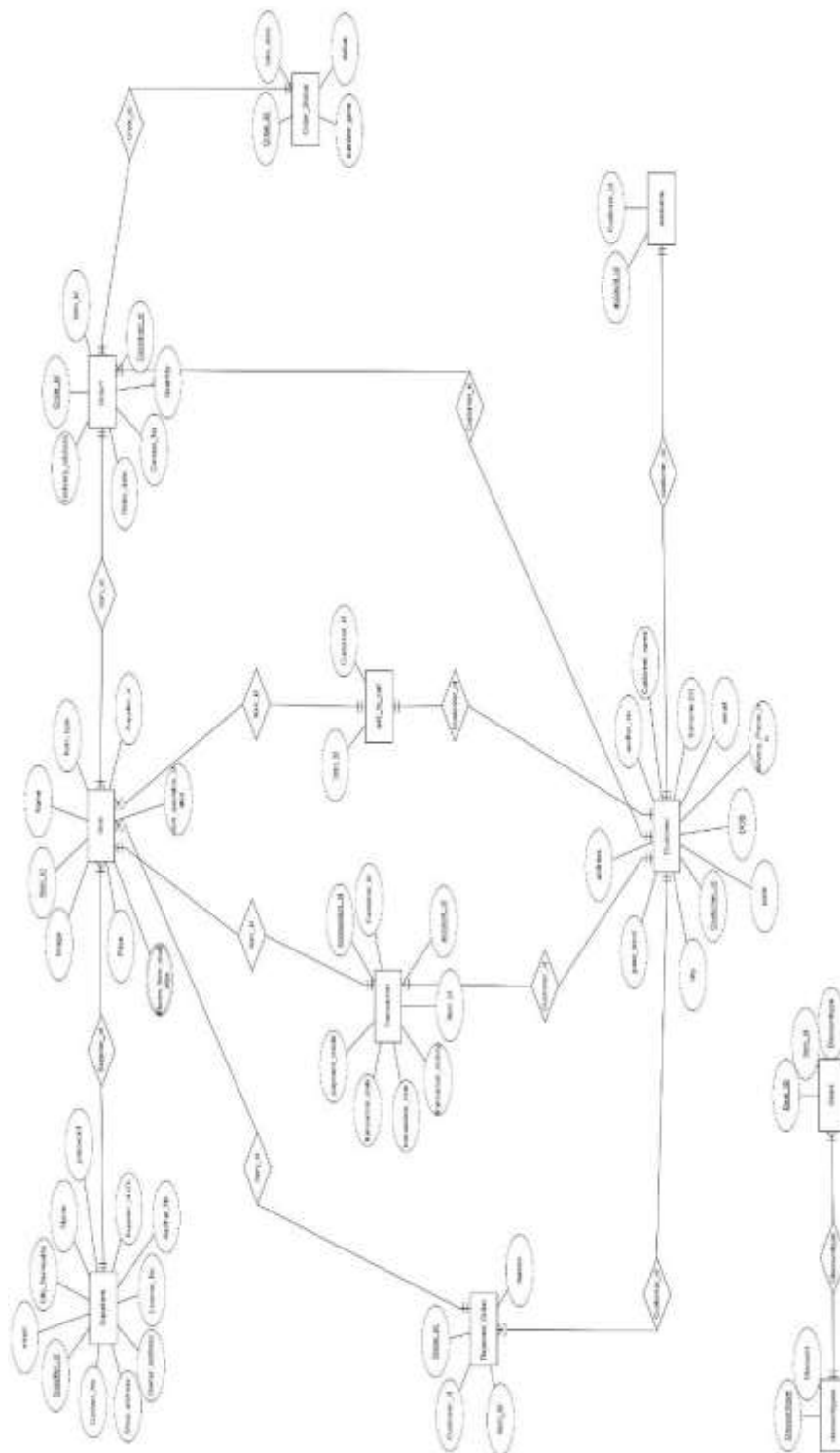
Essential Features

We had performance issues related to the speed of accessing the desired data. To solve this issue, we used "indexing." We used "views" for security purposes because they provide encapsulation. To maintain the referential integrity of data by systematically changing the data, we used "triggers" in SQL. We have used several "transactions" with all the commands like "commit," "roll back," and "savepoint" to perform several tasks. While trying to modify the database, we faced many anomalies. To avoid these anomalies, we used "data normalization." This also helped us reduce redundant values of tuples which saved storage space. To altogether avoid data redundancy, we applied normalization from "1 NF" form to "2 NF" form, then "3 NF" form till "BCNF".

Schema diagram of our database



ER diagram of our database



Database access and manipulation

1. Access all the items.

```
select * from item;
```

=> σ (item)

2. Search items by their item type.

```
select * from item where Item_type = 'beer';
```

=> σ Item_type='beer' (item)

3. Search items with price between 100 and 200.

```
Select * from item where Price between 100 and 200;
```

=> σ 100 < Price < 200 (item)

4. Search items by name where name is “Martini”.

```
select * from item where Name = 'Martini';
```

=> σ Name = 'Martini' (item)

5. To find all suppliers available in “Dwarka”.

```
select * from Suppliers where City_Serveable = 'Dwarka';
```

=> σ City_Serveable = 'Dwarka' (Suppliers)

6. Adding an items by supplier.

```
insert into Item (Item_id, Name, Item_type, Supplier_ID, Item_available_status,  
Places_item_available, Price , Available_Date, Item_Container,Image) values ('063216', 'Corona  
Extra', 'Beer', '63-7092090', 'Available', 'Delhi', 1802, '2020-09-02',  
'Bottle', 'https://products2.imgix.drizly.com/ci-corona-extra-  
2b48031ca2c738b1.jpeg?auto=format%2Ccompress&fm=jpg&q=20');
```

=> Item \leftarrow Item \cup { ('063216', 'Corona Extra', 'Beer', '63-7092090', 'Available', 'Delhi', 1802, '2020-09-02', 'Bottle', <https://products2.imgix.drizly.com/ci-corona-extra-2b48031ca2c738b1.jpeg?auto=format%2Ccompress&fm=jpg&q=20>) }

7. Add a deal for the customers.

```
insert into Deals (Deal_ID, item_id, Discounttype) values ('2T3BFREV2DW871421', '25-  
4705406', 'a');
```

=> Deals \leftarrow Deals \cup { ('2T3BFREV2DW871421', '25-4705406', 'a') }

8. Supplier updating the price of item with the item_id = “131489” to 1800

```
update item set price = 1800 where Item_id = 131489;
```

=> item \leftarrow \prod Item_id, Name, Item_type, Supplier_ID, Item_available_status, Places_item_available, Price=1800 (σ Item_id='131489')

9. Customer updating their address.

```
update Customer set address = '598 Delladonna Avenue' where Customer_id = '10-1234561';
```

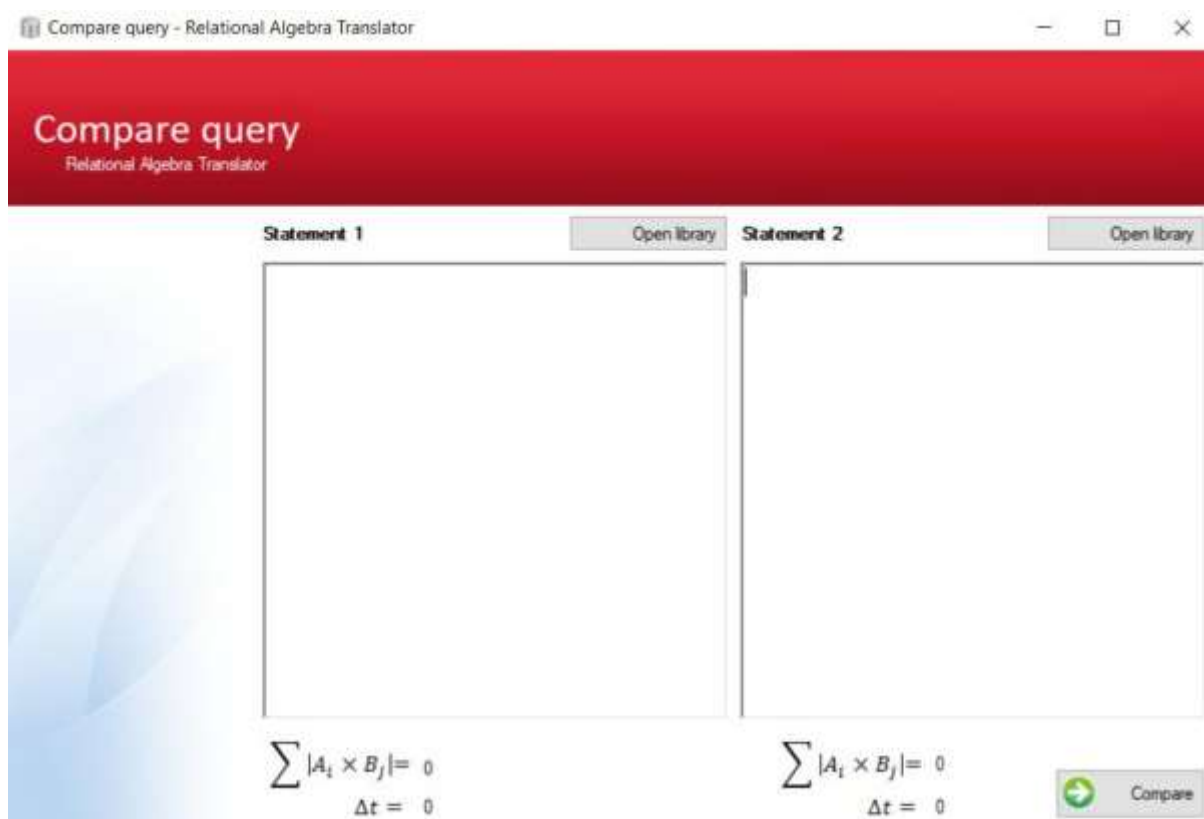
=> Customer \leftarrow \prod Customer_name, Surname, email, DOB, state, city, address= '598 Delladonna Avenue' (

σ Customer_id='10-1234561')

10. Access all the orders done by the Customer with Customer_id = "10-1234561".
select * from Order1 where Customer_id = "10-1234561";

=> $\sigma_{\text{Customer_id}='10-1234561'}(\text{Order1})$

Query Optimization



Group members-

Ankit Kumar (2018218)

Aditi Soni (2018326)

Yash Raj (2018422)

Kaiman Yadav (2018288)

Rahul Kumar (2018077)