**E-commerce**

Request for Proposal

|  |  |  |
| --- | --- | --- |
| RFP:E-commerce | Proposal due by: 09-15-2021 | E-commerce site |
| Project overview:  A company wants to enroll into the E-commerce business so we will look into its various requirements. | | |
| Project Goals:   * Design a database of the e-commerce * It may contain various attributes , their features and relationship between the attributes * Suggest the client for the best possible model. | | |
| Scope of work:  To design and develop the database system to manage the E-commerce company site. | | |

**Requirement for data models**

Firstly, we will get to know the business model of the e-commerce company by meeting with the business stockholders , managers ,and CEOs.

We will ask them various questions about the business policies and the legal rules and regulations.

We will ask them about their company strategy and what are the features they want to include and also try to suggest to them our best models with the hands- on experience of our senior developers.

As ecommerce is a very trending business , the business model could get complex with the current state of art scenario .

Meeting Notes

Date:9/8/2021

Agenda:

* Discussion on company basic requirements.
* Discussion on recent state-of-the-art e-commerce features.
* Discussion about the database entities
* Discussion on database attributes
* Discussion on preferred DBMS

Meeting Notes:

1. Main business plans
2. Appropriate database to choose
3. MySQL,Oracle,PostgreSQL , SQL server

Decision Made:

1. No prior decision made, but postponed the discussion for further meetings

Action items:

1. Create the visual representation of the database model to make it easily understandable to the client.

Discovery Questions

|  |  |
| --- | --- |
| Question/Thoughts | Answer |
| What are the basic features for your system you want? | It's an ecommerce site . |
| Which database system do you want? | Maybe some open source. |
| What are the unique features you want for your site except for some basic features? | Like automatic cart item removal after some deadlines.  Online payments from e-wallets |
| Can a user be able to track the product delivery? | Yes, the customer should be able to track its product delivery. |

So, being working as a data engineer I will only be focused on designing the database of ecommerce.So to enforce this database design , firstly I will design the conceptual model of it, which includes the listing of the possible entities of our model.

Then we will work with a logical model , which includes further breaking down the entities into its attributes and showing the relationship between the entities.

Below is the list of entities which we will include in the model.

**Possible Entities of our model**

1. Product
2. Customer
3. Shipping address
4. Payment
5. cart
6. Order
7. OrderDetails
8. Reviews

**Business rules**

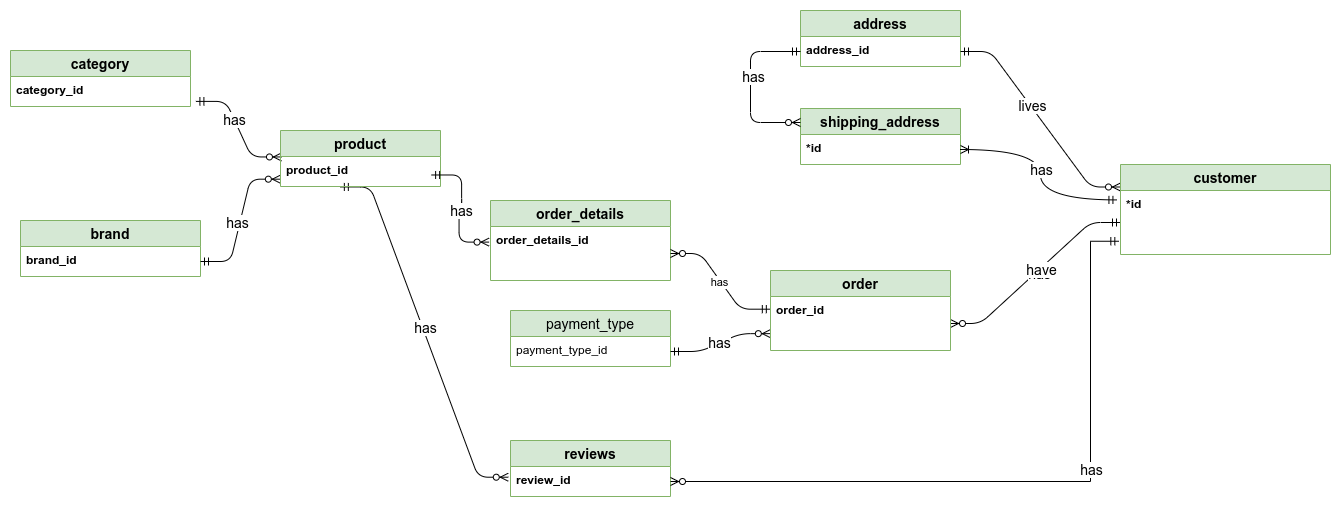
* Product should have its name , category , stock,price and brand.
* Products can only be associated with a single category and brand.While a category and brand can have multiple products .
* Payment can only be done either by ESEWA or the option cash on delivery can be selected.
* Customer should have its id, name, phone number , email address , and his/her possible address and its shipping address.The shipping address can also be provided during the checkout process.
* The customers can have only a single address while they can have multiple shipping addresses.
* Items and their quantity can be added to the cart if the product is in stock.
* Cart can be proceeded to checkouts.
* Customers can add reviews to the product

**Conceptual Modelling**

In this section we will list our entities , attributes and their relationship , now we will show the ER diagram by taking only entities and attributes into consideration.

|  |  |  |
| --- | --- | --- |
| Entity | Attributes | Relationship |
| Product | id,name,category,price,brand\_id,stock | categoty\_id,brand\_id |
| Category | name , description,id |  |
| Brand | product\_id,name,description |  |
| Customer | id,name,phone\_number,email,address\_id | address\_id |
| Address | country,state,city,landmark,id |  |
| Shipping address | id,country,state,city,house\_number,landmark,customer\_id | customer\_id |
| Payment methods | cash\_on\_delivery,esewa,id | order\_id |
| Cart | product\_id,product\_quantity,cart\_id,customer\_id | product\_id,customer\_id |
| Order | id,customer\_id,order\_date | customer\_id |
| Order\_details | Order\_id,product\_id,quantity | order\_id,product\_id,payment\_method |
| Reviews | review\_id,product\_id,customer\_id | product\_id,customer\_id |

*Fig: Table listing the entities , attributes and their relationship keys*



*Fig: ER diagram of E-comm*

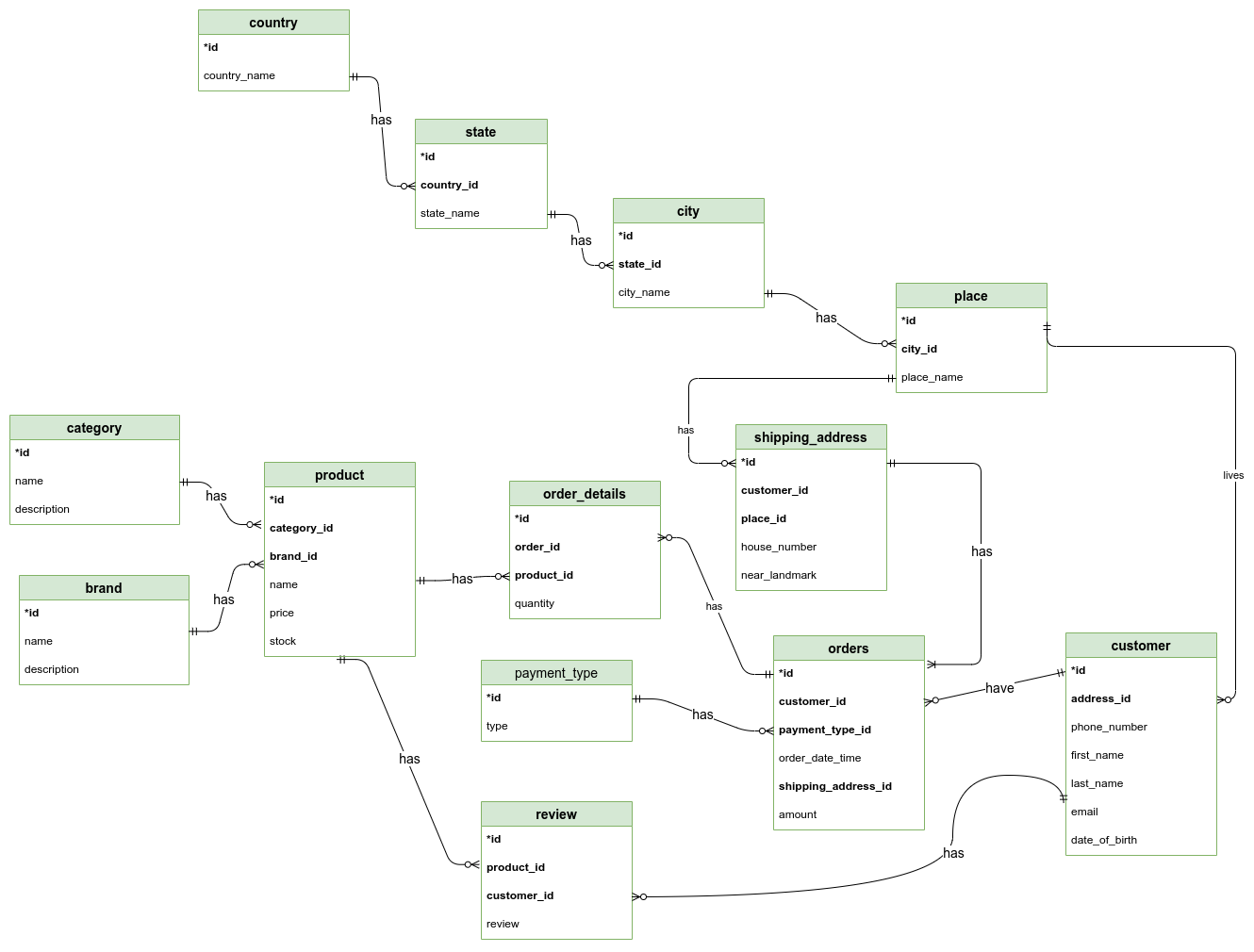
**Logical Modelling**

We already found the entities , attributes and their basic relationship before in the requirement analysis.Now we will give the description of the entities , description of their attributes and the attributes domain.

|  |  |  |
| --- | --- | --- |
| Entity | Description | Domain |
| customer | Clients of the e-commerce |  |
| Attributes:  id  name  address\_id  phone\_number  email  date\_of\_birth | Identifier for user entity,SK,PK  First Name +Last Name  Address of the user , FK  Customer can have more than one phone number  Valid email address of the customer  Valid date of birth | Auto generated,Serial  Text  FK reference  Valid phone number  Valid email address  Valid Date stamp |
| product | The material which the e-commerce sells |  |
| Attributes:  id  category\_id  brand\_id  name  Price  stock | Identifier for the product entity,SK,PK  FK which references the category entity.  FK which references the brand entity.  Name of the product  The price of the product.  It shows whether the product is in stock. | Auto generate,Serial  FK  FK  Text  Numerical value  Boolean value or fixed numeric |
| brand | The brand of the product | originals |
| Attributes:  id  name  description | Identifier of the brand entity , SK,PK  The name of the brand  The description of the brand. | Auto generated,Serial  Text  Text |
| category | The various categories to which the product belongs |  |
| Attributes:  Id  name  description | Identifier of the category entity, SK,PK  Name of the entity category  Brief description about the category | Auto generated,Serial  Text  Text |
| place | The local street where the customer lives |  |
| Attributes:  id  city\_id  place\_name | Identifier of the street entity, SK,PK  FK which references the city.  The name of the place inside the city where the customer lives. | Auto generated,Serial  FK  FK |
| city | The name of the city. |  |
| Attributes:  id  state\_id  city\_name | Identifier of the city entity, SK,PK  FK which references the state of the city.  The name of the city | Auto generated, Serial  FK  Text |
| state | The name of the state. |  |
| Attributes:  id  country\_id  state\_name | Identifier of the state entity, SK,PK.  FK which references the country of the state.  The name of the state | Auto generated,Serial  FK  Text |
| country | The name of the country. |  |
| Attributes:  id  country\_name | Identifier of the country entity, SK,PK  The name of the country | Auto generated,Serial  Text |
| shipping\_address | The address to which the customer wants to ship the product. |  |
| Attributes:  id  customer\_id  street\_id  house\_number  near\_landmark | Identifier of the shipping\_address entity, SK,PK  FK which references the customer.  FK which references the street of customers.  It is the house number to deliver the product.  Near landmark at which the house is located | Auto generated,Serial  FK  FK  Text+Number  Text |
| order\_details | The details of the orders made by the customer. |  |
| Attributes:  id  order\_id  product\_id  quantity | Identifier of the country entity, SK,PK.  FK which references the order.  FK which references the product bought by customer  The number products bought by customer | Auto generated  FK  FK  Numeric Field |
| orders | The orders of the customer. |  |
| Attributes:  id  shipping\_id  payment\_id  order\_date  amount | Identifier of the order entity, SK,PK.  FK which references the shipping address of customer  FK which references the payment\_type mentioned by the customer during order confirmation.  The date and time at which the order was confirmed.  The total amount of the order by customer | Auto generated  Valid id from shipping\_address table  Valid id from the payment table  Time Stamp  Numeric type |
| payment\_type | The method of payment listed by the customer. |  |
| Attributes:  id  type | Identifier of the payment\_type entity, SK,PK.  The type of payment chosen by the customer during order confirmation, which could be either payment through e-wallets or cash on delivery. | Auto generated  Options to choose |
| reviews: | The reviews given by customers to the product. |  |
| Attributes:  id  Product\_id  customer\_id  reviews | Identifier of the reviews entity, SK,PK.  FK which references the product to which the customer has reviewed  It is the customer who has reviews the product.FK  The reviews given by the customer on the product. | Auto generated  Valid id from product table,FK  Valid id from customer table ,FK  Text |

*Fig:Table showing the entities,attributes,attributes description and attributes domain of the entity.*

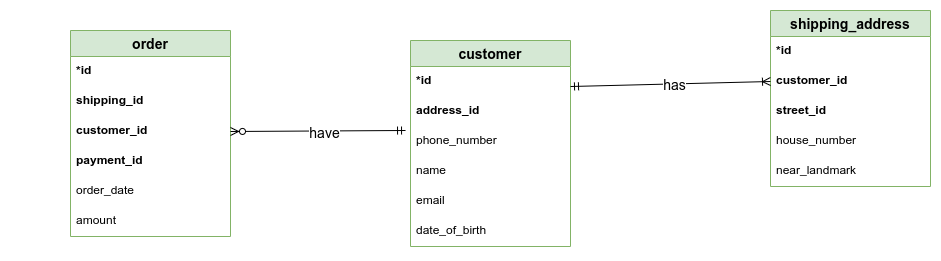
Now we will also draw ER diagrams of the entities, attributes and their relationship which is an entity-relationship diagram like we drew above but now we will add its attributes and show their relationships.



*Fig: ER diagram of E-comm*

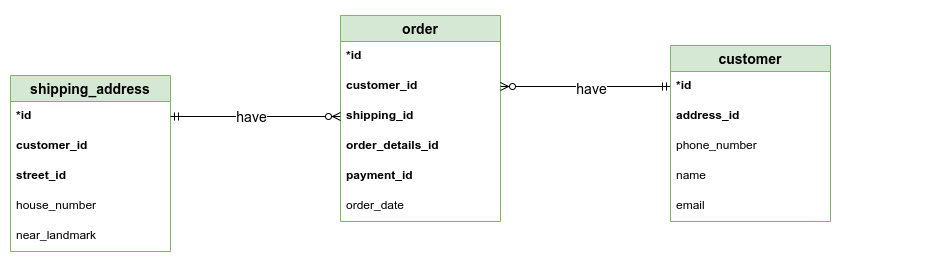
**Verification**

After building the ER, now I tried to verify it and going through it I found the Fan-Trap problem.



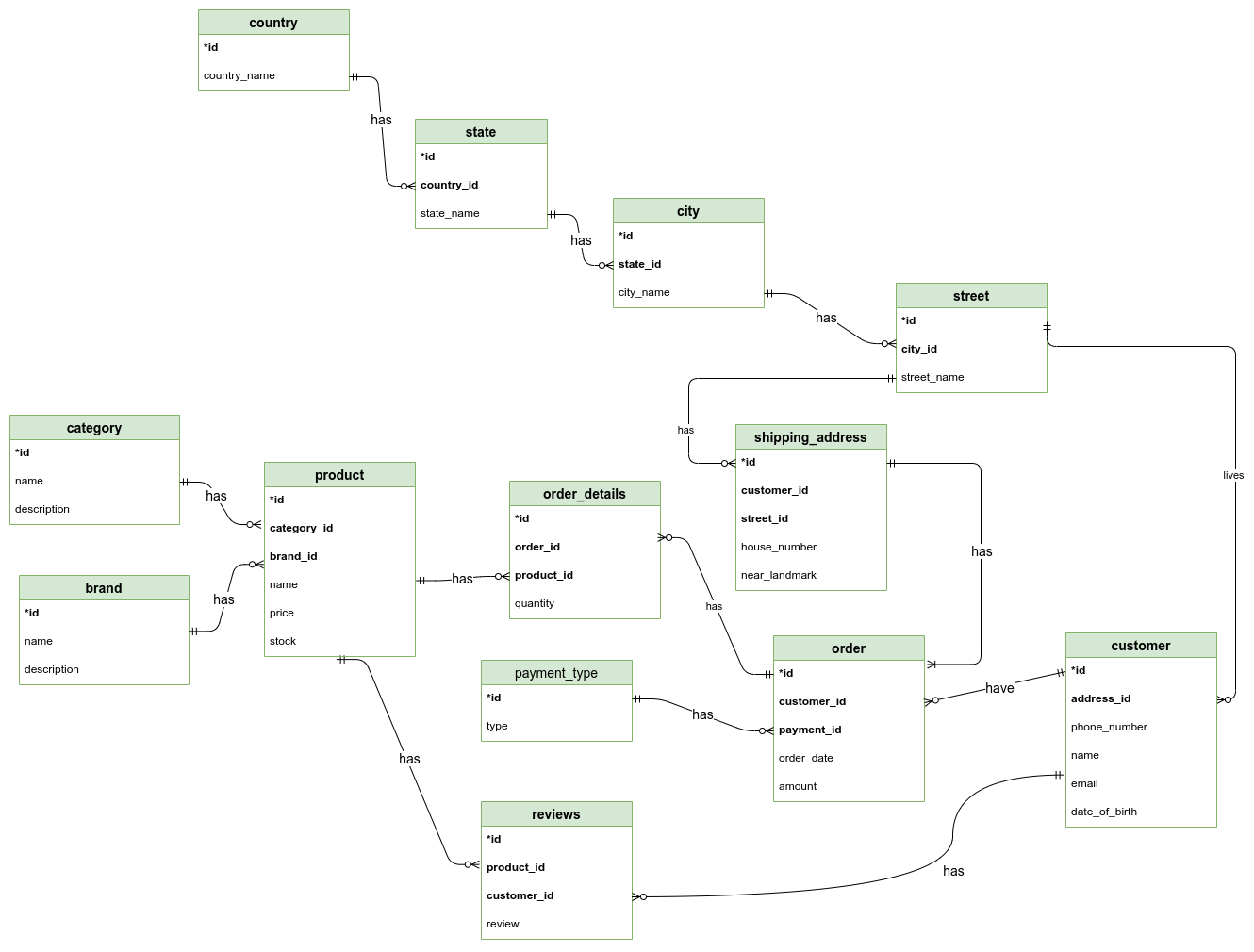
*Fig: Fan-Trap of above ER model of E-comm*

So, we can solve these problems by reconstructing the ER model to represent the correct association among the entities.



*Fig: Fan-Trap solution of above ER model of E-comm*

Now , we have solved the fan trap problem, The ER diagram is as below.



*Fig: ER diagram after solving the Fan-Trap problem*

Now , I am going to check whether each table satisfies a different Normalization Form(NF).

Here,All of the tables are already in 1NF as it has atomic data and no repeating groups.

So, if it satisfies the 1NF , I am going to check for the 2NF.

As every table has its own unique id as its primary key which is also a surrogate key, it already satisfies the condition of 2NF.

As the table is already in 2NF form, we can now check for 3NF conditions.

For the table not to satisfy the 3NF condition , there should be the transitive dependency . But looking at all of the mentioned tables I don’t see any non-prime attributes determining the other non-prime attributes.So,we can say there is no transitive dependency.

Hence, the tables are already in 3NF form.

**Physical modelling**

This is the part where I will physically implement the model into the database.For that purpose we are gonna use PostgreSQL relational database.It is an open source DBMS.

It uses the Standard Query Language(SQL) to create,update,alter ,delete the tables or any rows in the database.

Firstly , I made the dummy datas for each table with some tuples in it in the Excel sheet.

Then , I have used python and the driver of PostgreSQL which is Psycopg2 to push the

Data of .xlsx file into the Postgre DBMS.I have use Jupyter Notebook to write all of the code .

The physical implementation code of the E-comm is presented below in my github repository link:

LINK: <https://bit.ly/3hezrpl>

**References:**

1. *Draw.io (*[*https://app.diagrams.net/*](https://app.diagrams.net/) *)*
2. PostgreSQL ( <https://www.postgresql.org/> )
3. Psycopg2 ( <https://www.psycopg.org/docs/> )
4. Python Programming language (<https://www.python.org/> )
5. Jupyter Notebook (<https://jupyter.org/> )