DBMS Project : Online Retail Store System

Group 36
Jahanvi Bakshi - 2020069
Vaibhav Joshi - 2020145
Vaibhav Rajpal - 2020146
Vibhu Jain - 2020151

Problem Statement :

A local retail store owner was unable to sell products during Covid lockdown and wanted to manage the store on a computer and sell products online. So we design a Database Management System for managing Retail Stores.

Scope of Project :

Online Retail Store Systems are used widely in today's tech driven world. Some well known examples are Grofers', Big Bazaar, Reliance, and to some extent, giants like Amazon. The way technology is getting more affordable, small-scale retail stores may want to make their management systems tech driven, because it makes their tasks much easier, simpler, faster, with protected and preserved data.

Stakeholders:

1. Customers of the retail store
2. Employees of the retail store(like delivery people)
3. Suppliers to the retail store

Relationships Established:

- Items are *Added_to* Cart
- A Customer *has_a* Cart
- Suppliers *Add* Items
- Delivery-person(Employee) *Delivers* to the Customer

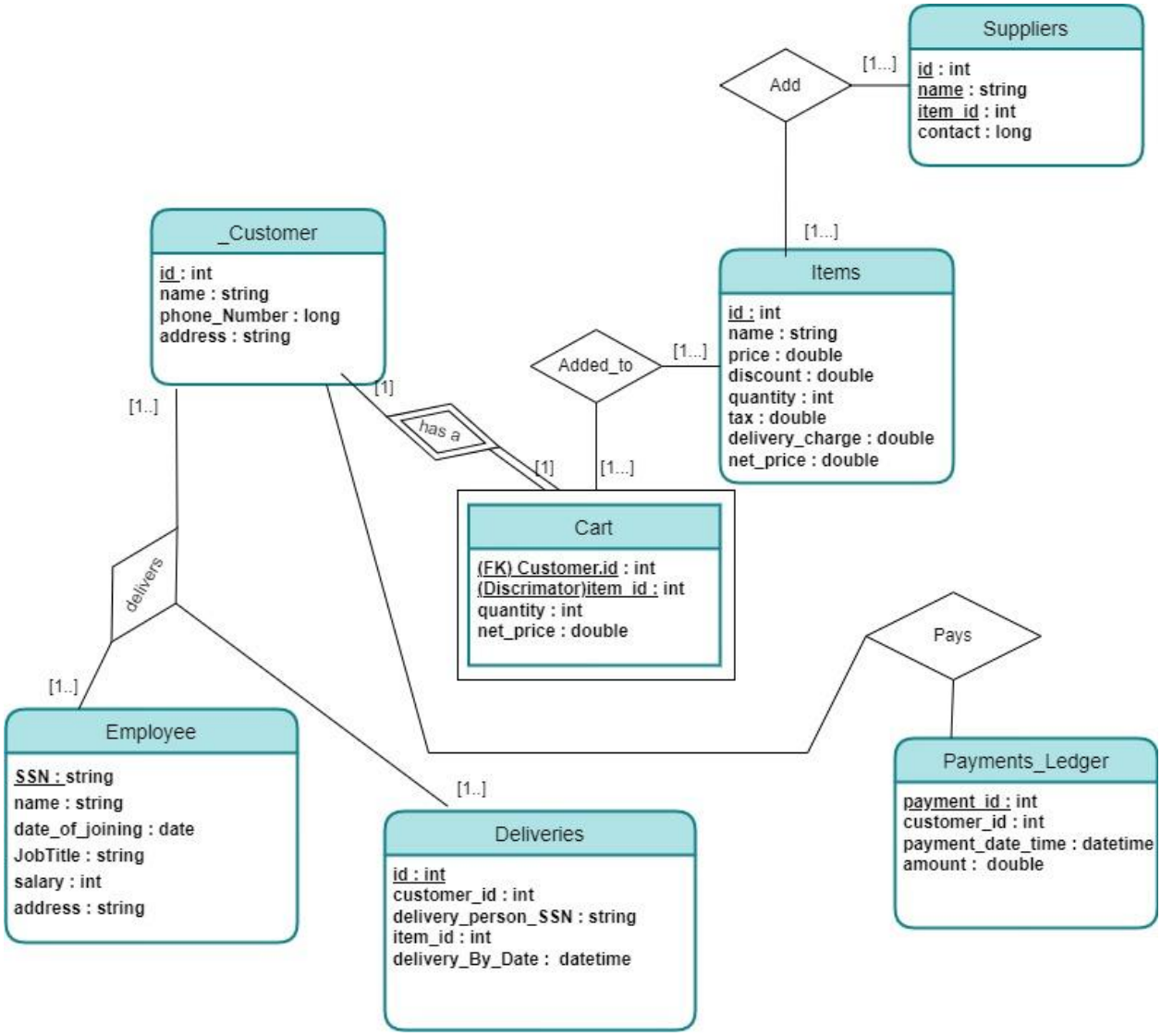
Weak Entity:

Cart is a weak entity, which has its identifying entity as Customer, with item_id as the discriminator. A cart has no meaning without a customer using it.

Ternary Relationship:

- Between Customer, Employee, and Deliveries

ER DIAGRAM:



Relational Schema:

_Customer = (id, name, phone_number, address)
Employee = (SSN, name, date_of_joining, JobTitle, salary, address)
Deliveries = (id, customer_id, delivery_person_SSN, item_id, delivery_by_date)
Items = (id, name, type, price, discount, quantity, delivery_charge, tax, net_price)
Suppliers = (id, name, item_id, contact)
Cart = ((FK)Customer.id, (Discriminator)*item_id*, quantity, net_price)
Payments_Ledger = (payment_id, customer_id, payment_date_time, amount)

Mapping Constraints & Integrity Constraints:

- One to one relationship : between Customer and Cart
- Many to many relationship : Employee and Customer; Suppliers and Items;
- Integrity Constraints : id, age are always integer, name is always string, dates are of type datetime, payment amounts are of type double(or float).

Grants and views are in file *Grants and Views.sql*
The embedded sql queries are present in file embeddedsql_queries.py
Indexing queries are present in file *indexing.sql.*
Triggers are in file *triggers.sql*
UI code in file *UI.py*