CZ2007 Lab 1 submission

Lab group: MACS2; Group number: 3
Group member: Tang Yi Qwan (GL), Khor Sze Chong, Chen Peilin, Chan Joshua Juan Yin, Zhou Yiqi, Huang Wei

In our design, we interpreted a “Product” entity as simply a listing created by a Shop; Each entry is independent of any physical item. As a result, the same physical item (eg. an iPhone 13 Pro Max 256GB) being sold by different stores will have different entries in the Product entity set. This interpretation is also reflected by the relation between Shops and Products, “List”.

Because Product listings cannot exist without a Shop to create them, there is a referential integrity constraint in the “List” relationship; every Product must have exactly one Shop. Additionally, since Products from different Shops can have the same IDs, name, etc., the Products entity set has to be weak, and it has to be associated with the Shop entity set. In particular, a ProductID together with the associated Shop’s name must be provided in order to identify a unique Product.

Our implementation of the Price Record is within the “List” relation. The List relation contains further attributes which are used to determine the price associated with a Product at various date/times desired (date/time with Shop name and ProductID being a key). The current price of a product will simply be defined as the price associated with the most recent entry by date/time.

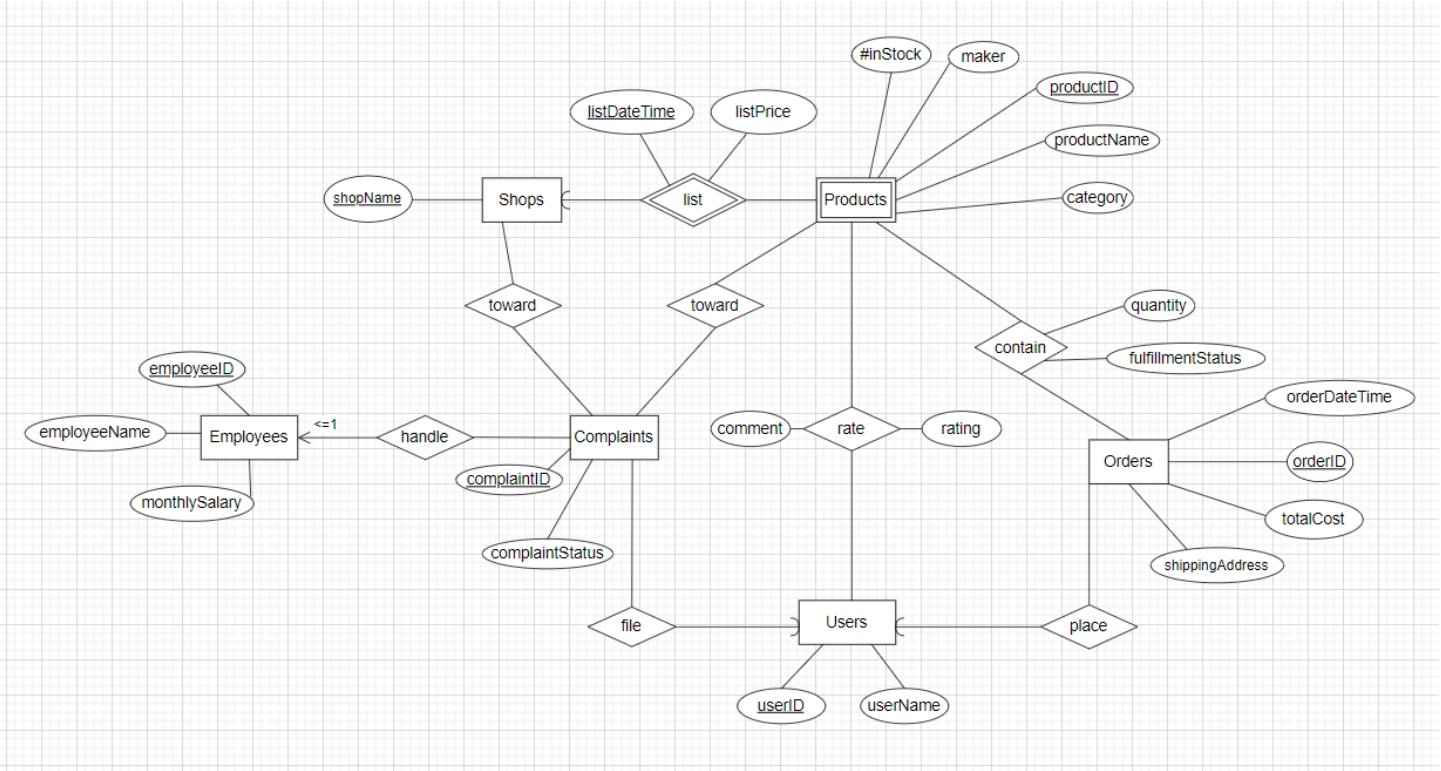
Users are the entities that rate Products, place/track Orders, and file/check Complaints, and can be identified by a unique UserID, since Users may have the same name. In particular, referential integrity constraints are needed from Orders and from Complaints, since those entities must be created by exactly one User.

“Orders” must be an entity set because it needs to be related to at least one Product and User via the “Contain” and “Place” relationships respectively. The “Contain” relationship further tracks the quantity of the corresponding Product ordered, as well as its fulfillment status. Since a User can Place several Orders, while each Order may Contain more than one Products, we cannot base the reference of each Order on a specific User or Product. As such, each Order is identified by a unique OrderID.

“Complaints” can be made towards both Shops and Products. This was implemented in our design by the “Towards” relation to the corresponding entity sets. Since a User can File several Complaints, while a Complaint may have no associated Employee, we cannot base the reference of each Complaint on a specific User or Employee. As such, each Complaint is identified by a unique ComplaintID.

Lastly, an Employee has to “Handle” Complaints. Because a Complaint can have no Employee currently handling it (eg. when it is new), there is no referential integrity constraint. In addition, since Complaints can have at most one Employee Handling it, while each Employee can Handle zero Complaints, one or many Complaints, the Handle relation is a One-to-Many relation from Employee to Complaint.

ER diagram



Name	Individual Contribution to Submission 1 (Lab 1)	Percentage of Contribution	Signature
Tang Yi Quan	ER diagram explanation, drawing	16.7	
Khor Sze Chong	ER diagram explanation drawing	16.7	
Joshua Chan Tuan Yin	ER diagram explanation, drawing	16.7	
Zhou Yigi	ER diagram explanation	16.7	
Huang Wei	ER diagram explanation	16.7	
Chan Peiln	ER diagram drawing	16.7	