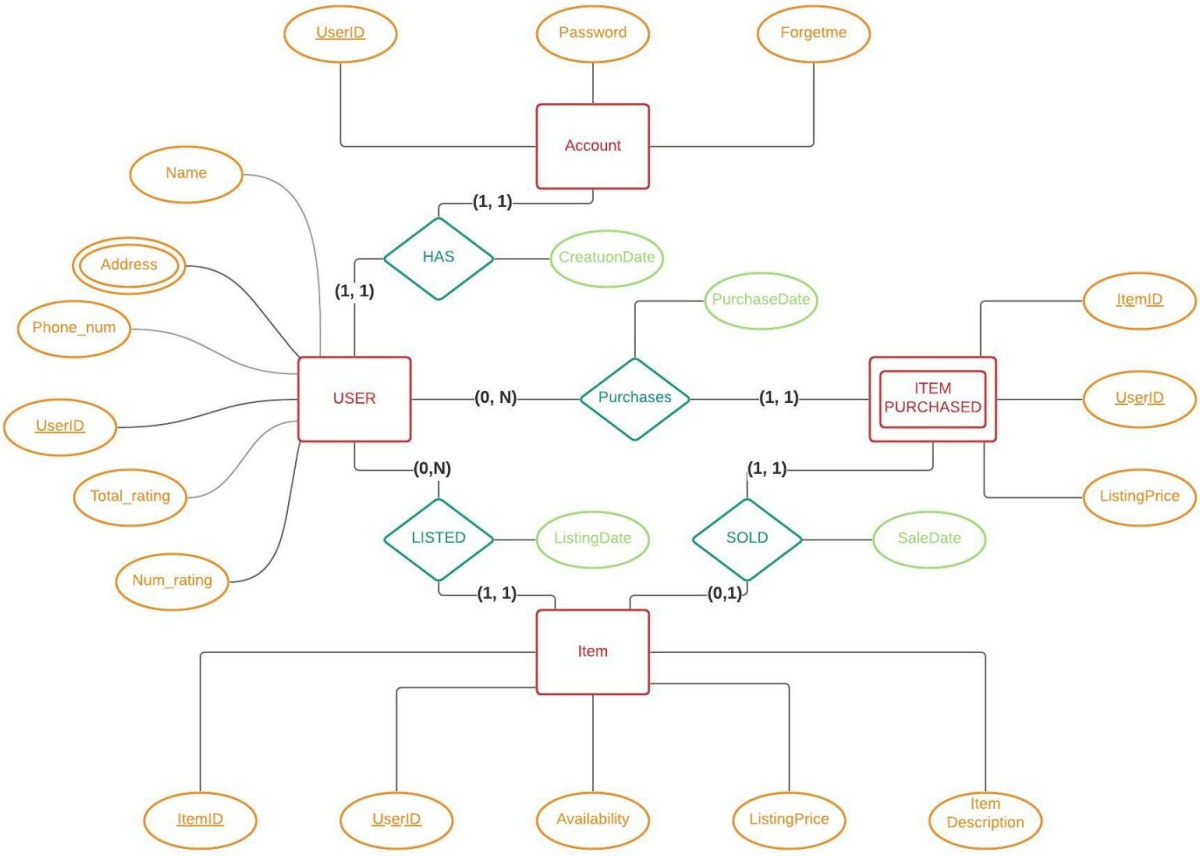
Milestone two - EasyMarket

Surya Iyer, Yangxin Fan, Moshiul Azam

Task A: ER Diagram

Four entities: USER, ITEM, ACCOUNT, and PURCHASES and four relationship types.
Relationships types include:

- HAS (between USER, ACCOUNT), 1:1 cardinality ratio
- PURCHASES (between USER, PURCHASES), 1:N cardinality ratio
- LISTED (between USER, ITEM), 1:N cardinality ratio
- SOLD (between ITEM, PURCHASES), 1:1 cardinality ratio



Cardinality:
In relationship HAS, min = max =1 for both USER and ACCOUNT.
In relationship PURCHASE, min = 0, max = N for USER, min = max = 1 for PURCHASES (purchased items).

In relationship LISTED, min = 0, max = N for USER, min = max =1 for ITEM.
In relationship SOLD, min = 0, max = 1 for ITEM and min = max = 1 for PURCHASES.

Participation and (min, max) constraints are shown in the ER diagram, derived from cardinality.

Task B: Relational Database Design Using ER-to-Relational Mapping

Part I: ER-to-Relational Mapping Algorithm

Step 1: Mapping of regular entities type
1.1: Create relations USER, ITEM, and ACCOUNT
1.2: Choose UserID, Password, and ItemID as primary keys of USER, ACCOUNT, and ITEM

Step 2: Mapping of weak entities type
2.1: Create relation PURCHASES
2.2: Include UserID and ItemID as foreign keys
2.3: Choose primary key of PURCHASES as the combination of its owners' primary keys UserID and ItemID.

Step 3: Mapping of 1:1 relationship types
There are two 1:1 relationship types: USER has ACCOUNT and ITEM is purchased and listed in PURCHASES.
3.1: Choose UserID as the foreign key in ACCOUNT that refers to primary key UserID in USER; choose ItemID as the foreign key in PURCHASES that refers to primary key ItemID in ITEM
3.2: Include CreationDate as an attribute of ACCOUNT and SaleDate as an attribute of ITEM

Step 4: Mapping of 1:N relationship types
There are two 1:N relationships types: USER listed ITEM and USER purchased PURCHASES (purchased items).
3.1: Choose UserID as the foreign key of PURCHASES that refers to primary key UserID in USER
3.2: Include PurchaseDate as an attribute of PURCHASES and ListingDate as an attribute of ITEM

Step 5: Mapping of M:N relationship types (not applicable)
There are no M:N relationships.

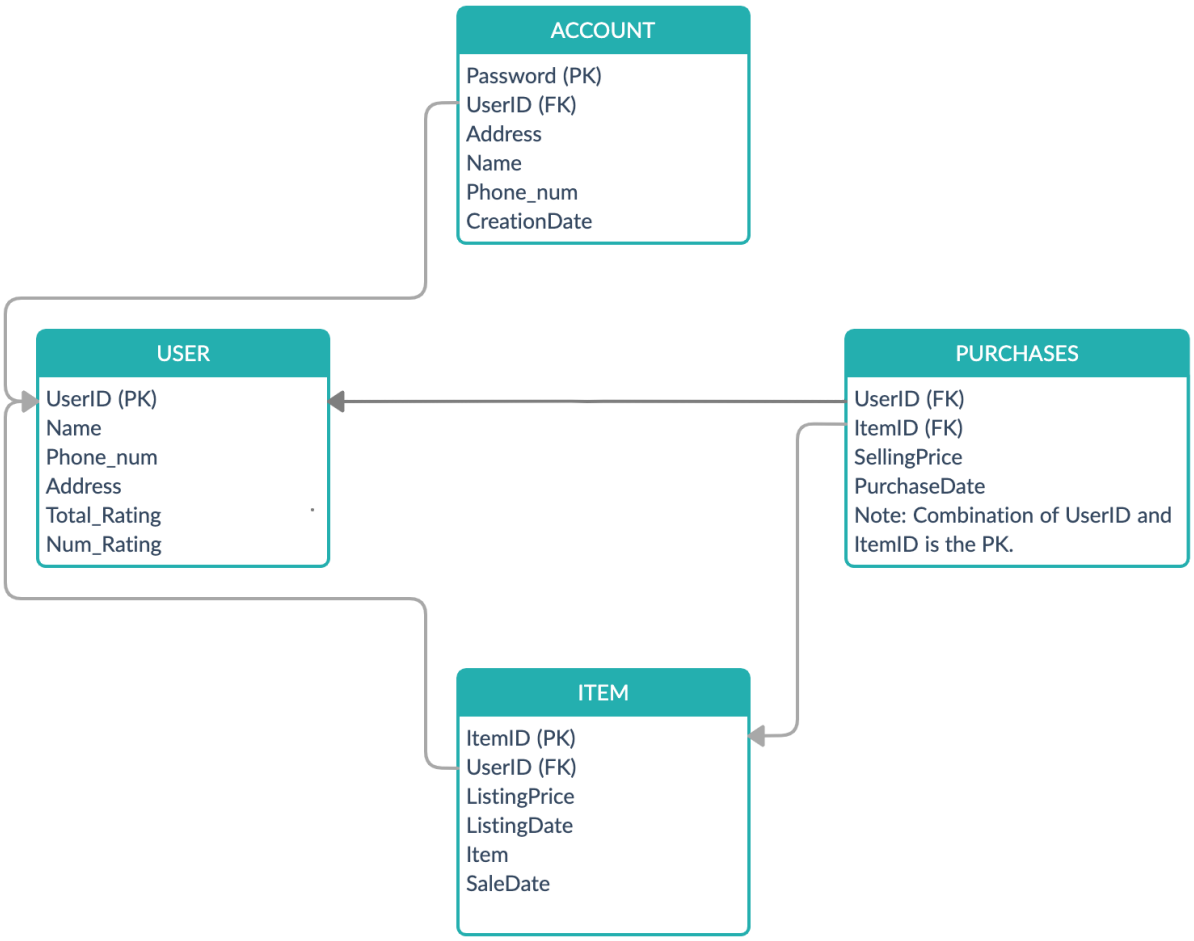
Step 6: Mapping of multivalued attributes (not applicable)

Step 7: Mapping of *N*-ary Relationship Types (not applicable)

Step 8: Options for Mapping Specialization or Generalization (not applicable)

Step 9: Mapping of Union Types (not applicable)

Figure similar to 9.2



Summary of mapping

Relation Name	ER Diagram Components
USER	E(USER)
ITEM	E(ITEM) + R(LISTED)

PURCHASES	E(PURCHASES) + R(Purchases) + R(SOLD)
ACCOUNT	E(ACCOUNT) + R(HAS)

Part II: Schema of your database

Table #1 BOOK (store basic information for each book)

Attribute	Default Value	Datatype	Description
BookID	Not null	INT	Unique bookID for each book; used to identify books; **primary key**
Publisher	Null	VARCHAR(255)	Publisher for each book
Title	Null	VARCHAR(255)	Book's title
ISBN	Null	INT(13)	Book's ISBN
Genre	Null	VARCHAR(255)	Genre of the book
Publication Year	Null	YEAR	Publication year
Price	Null	DECIMAL(6,2)	Selling price

Table #2 AUTHOR (store information of authors)

Attribute	Default Value	Datatype	Description
AuthorID	Not null	INT	Unique AuthorID for each author
BookID	Not null	INT	Unique bookID for each book written by the author.
FirstName	null	VARCHAR(255)	First name of the author
LastName	null	VARCHAR(255)	Last name of the author

Table #3 WAREHOUSE (store information of warehouse)

Attribute	Default Value	Datatype	Description
WarehouseID	Not null	INT	Unique id for each warehouse

Location	Null	VARCHAR(255)	Warehouse's location
Manager	Null	VARCHAR(255)	Warehouse's manager
Contact	Null	INT	Contact information such as phone number of this warehouse

Table #4 STORES (store and track the stock of each book)

Attribute	Default Value	Datatype	Description
BookID	Not null	INT	Unique bookID for each book
WarehouseID	Not null	INT	Unique id for each warehouse
AvailableQuantity	Not null	INT	Quantity of remaining books in warehouse

Table #5 ORDER (store information of each order)

Attribute	Default Value	Datatype	Description
OrderID	Not null	INT	Unique id for each order
Username	Not null	INT	Unique username of customers who makes the order
OrderDate	Not null	INT	The date of the order made
TotalPrice	Not null	DECIMAL(6,2)	Total price of books in one order
ShipName	Null	VARCHAR(255)	Shipping details such as an order's destination, the zip code, and the phone number and email customer notifies in the order.
Street	Null	VARCHAR(255)	
PostalCode	Null	INT(10)	
City	Null	VARCHAR(255)	
State	Null	VARCHAR(255)	
Country	Null	VARCHAR(255)	
PhoneNum	Null	INT(10)	
Email	Null	VARCHAR(255)	

Table #7 ORDER_CONTAINS (store each order's content)

Attribute	Default Value	Datatype	Description
OrderID	Not null	INT	Unique id for each order
BookID	Not null	INT	Unique id for each book
Quantity	Not null	INT	Each book's quantity in an order. We enforce customer to specify quantity when buying books.

Table #8 CUSTOMER (store customer's information)

Attribute	Default Value	Datatype	Description
Username	Not null	INT	Username of the customer who makes the order
FisrtName	Null	VARCHAR(255)	First Name
LastName	Null	VARCHAR(255)	Last Name
Street	Not null	VARCHAR(255)	Address and communication information of the customer. We enforce each customer provide specific address and phone number, otherwise he or she cannot create account.
PostalCode	Null	INT	
City	Not null	VARCHAR(255)	
State	Not null	VARCHAR(255)	
Country	Not null	VARCHAR(255)	
PhoneNum	Not null	INT(10)	
Email	Null	VARCHAR(255)	
Password	Not null	VARCHAR(255)	Login Password

Table #9 DELIVERS (track the status of each order)

Attribute	Default Value	Datatype	Description

OrderID	Not null	INT	Unique id for each order
CourierID	Not null	INT	Unique id for each courier
ShipStatus	Not null	VARCHAR(255)	Status of the package

Table #10 COURIER (store information of courier)

Attribute	Default Value	Datatype	Description
CourierID	Not null	INT	Unique ID for each courier
CourierName	Not null	VARCHAR(255)	Courier's Name
PhoneNum	Not null	INT(10)	Courier's phone number

Table #11 REVIEWS (store customer's review on each book)

Attribute	Default Value	Datatype	Description
Username	Not null	INT	Username of who makes the comment
BookID	Not null	INT	Unique id for each book
CommentID	Not null	INT	Unique id for each comment
CommentTime	Not null	DATETIME	When the comment is made
Rating	Not null	INT	Customer's rating for this book
Comment	Not null	TEXT	Customer's comment on this book

Table #10 COMMENT (store each comment's information)

Attribute	Default Value	Datatype	Description
CommentID	Not null	INT	Unique id for each comment
CommentTime	Not null	DATETIME	When the comment is made
Rating	Not null	INT	Customer's rating for this book
Comment	Not null	TEXT	Customer's comment on this book