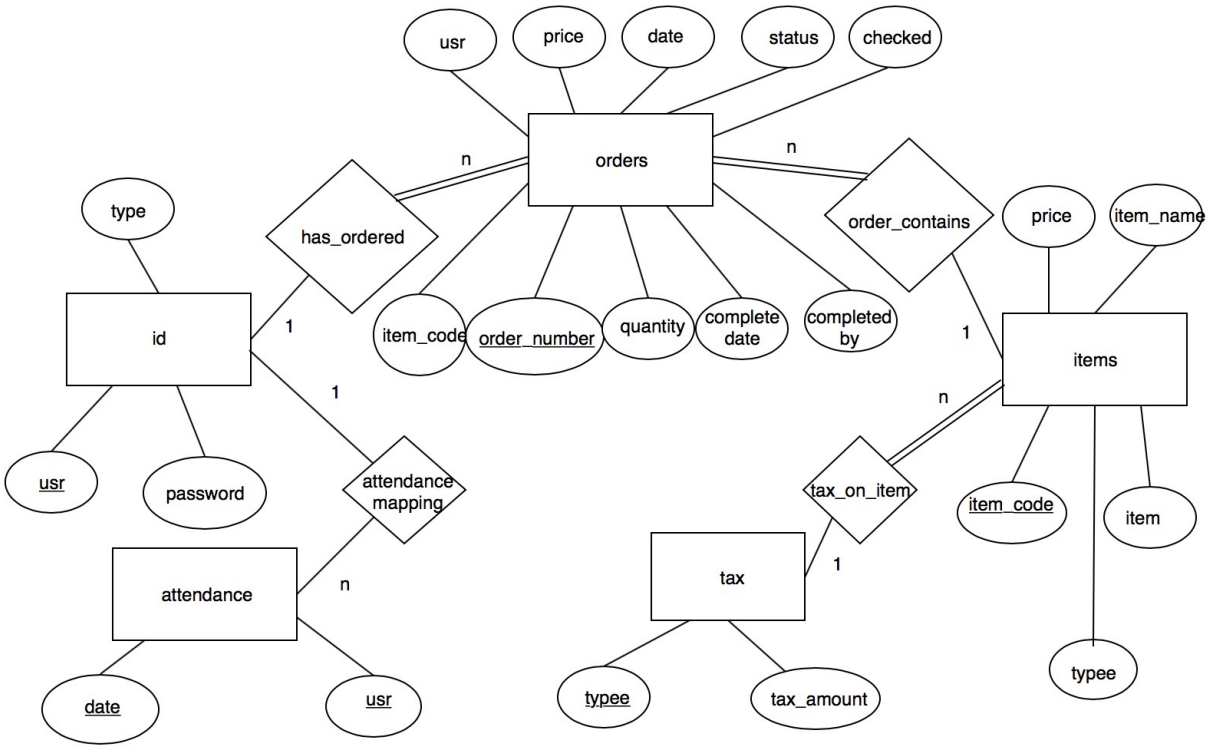
DATABASE DESIGN
Inventory Management Project

Varun Jain - 2014170
Siddharth Singh - 2014105

ER Diagram



Functional Dependencies

1. Table Name : id
Functional Dependencies
- usr->password
- usr->type

2. Table : items
Functional Dependencies
- item_code->item_name
- item_code->price

3. Table: orders
Functional Dependencies
- order_number->usr
- order_number->item_code

4. Table : tax
No Functional Dependencies

5. Table : attendance
No Functional Dependencies

- Checking Table ‘id’ for 2NF and 3NF violation

Key -> usr

2NF satisfying condition: There must not be any non-key attribute in the relation that is functionally determined by only a subset of a key

1. FD : usr->password
The non-key attribute ‘password’ is not functionally determined by any subset of the key, hence it doesn’t violates 2NF.

2. FD : usr->type
The non-key attribute ‘type’ is not functionally determined by any subset of the key, hence it doesn’t violates 2NF.

3**NF satisfying condition:**
In relation R, one of the following must be true for a functional dependency X -> B:
- *X* is a superkey
- *B* is a *key* attribute

1. FD : usr->password
usr is a super key and password is not a key attribute, hence it is in 3NF.

2. FD : usr->type
usr is a super key and type is not a key attribute, hence it is in 3NF.

Hence table ‘id’ is in 3NF

- Checking Table ‘items’ for 2NF and 3NF violation

Key -> item_code

2NF satisfying condition: There must not be any non-key attribute in the relation that is functionally determined by only a subset of a key

1. FD : item_code->item_name
The non-key attribute ‘item_name’ is not functionally determined by any subset of the key, hence it doesn’t violates 2NF.

2. FD : item_code->price
The non-key attribute ‘price’ is not functionally determined by any subset of the key, hence it doesn’t violates 2NF.

3**NF satisfying condition:**
In relation R, one of the following must be true for a functional dependency X -> B:

- *X* is a superkey
- *B* is a *key* attribute

1. FD : item_code->item_name
item_code is a super key and item_name is not a key attribute, hence it is in 3NF.

2. FD : item_code->price
item_code is a super key and price is not a key attribute, hence it is in 3NF.

Hence table 'items' is in 3NF

- Checking Table 'orders' for 2NF and 3NF violation

Key -> order_number

2NF satisfying condition: There must not be any non-key attribute in the relation that is functionally determined by only a subset of a key

1. FD : order_number->usr
The non-key attribute 'usr' is not functionally determined by any subset of the key, hence it doesn't violates 2NF.

2. FD : order_number->item_code
The non-key attribute 'item_code' is not functionally determined by any subset of the key, hence it doesn't violates 2NF.

3**NF satisfying condition:**
In relation R, one of the following must be true for a functional dependency X -> B:
- *X* is a superkey
- *B* is a *key* attribute

1. FD : order_number->usr
order_number is a super key and usr is not a key attribute, hence it is in 3NF.

2. FD : order_number->item_code
order_number is a super key and item_code is not a key attribute, hence it is in 3NF.

Hence table 'items' is in 3NF.

Minimal Relational Schema Design

Functional Dependencies:

H={usr->password, usr->type, item_code->item_name, item_code->price, order_number->usr, order_number->item_code}

usr^+ = (password, type)
$item\_code^+$ = (item_name, price)
$order\_number^+$ = (usr, password, type, item_code, item_name, price)

Checking for minimal FDs

1. usr->password : can't be removed, because for J = H - {usr->password}, usr^+ = (type)

2. usr->type : can't be removed, because for J = H - {usr->type}, usr^+ = (password)

3. item_code->item_name : can't be removed, because for J = H - {item_code->item_name), $item\_code^+$ = (price)

4. item_code->price : can't be removed, because for J = H - {item_code->price), $item\_code^+$ = (item_name)

5. order_number->usr : can't be removed, because for J = H - {order_number->usr), $order\_number^+$ = (item_code, item_name, price)

6. order_number->item_code : can't be removed, because for J = H - {order_number->item_code), $order\_number^+$ = (usr, password, type)

Hence our functional dependencies are minimal.