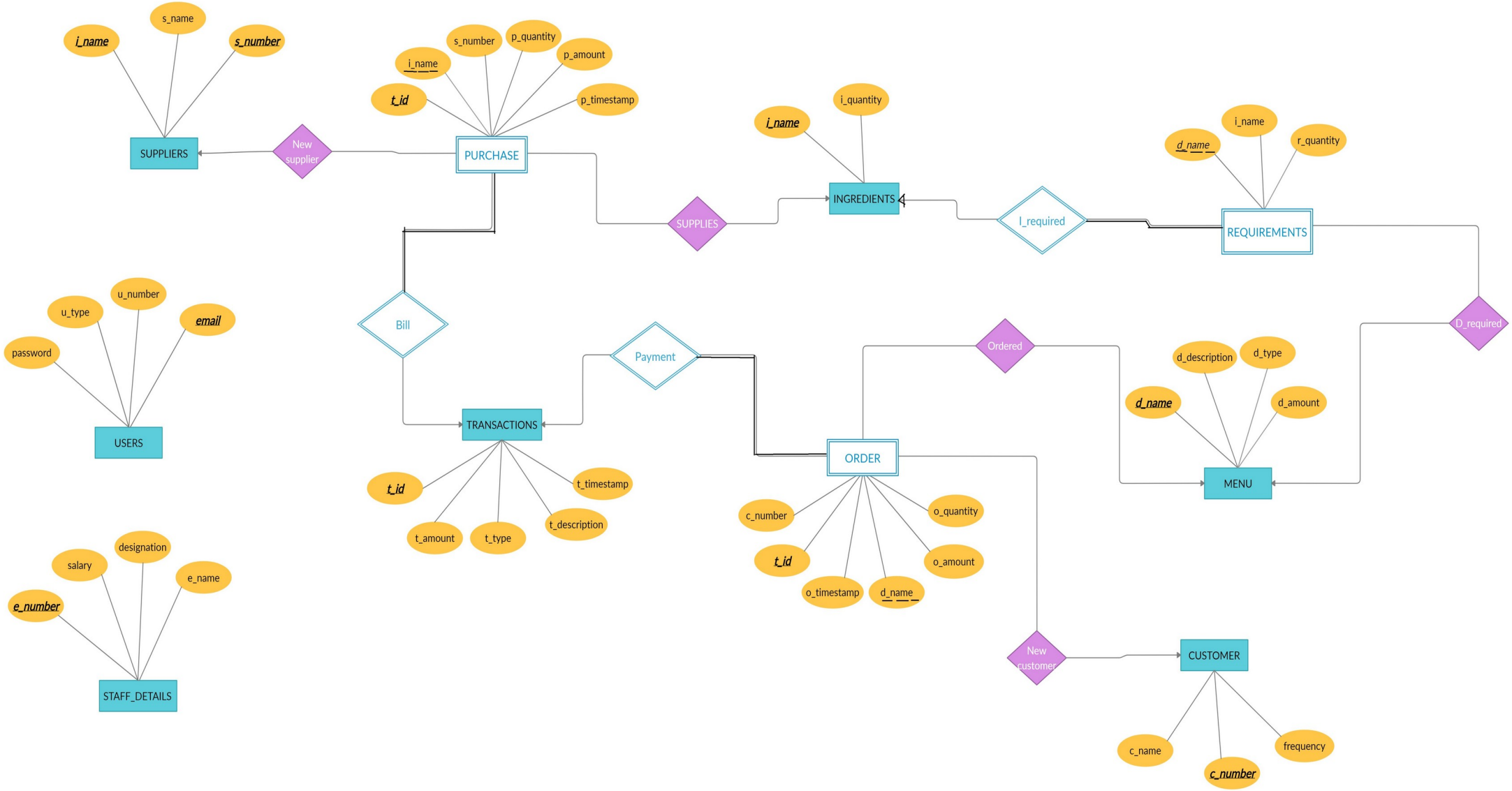
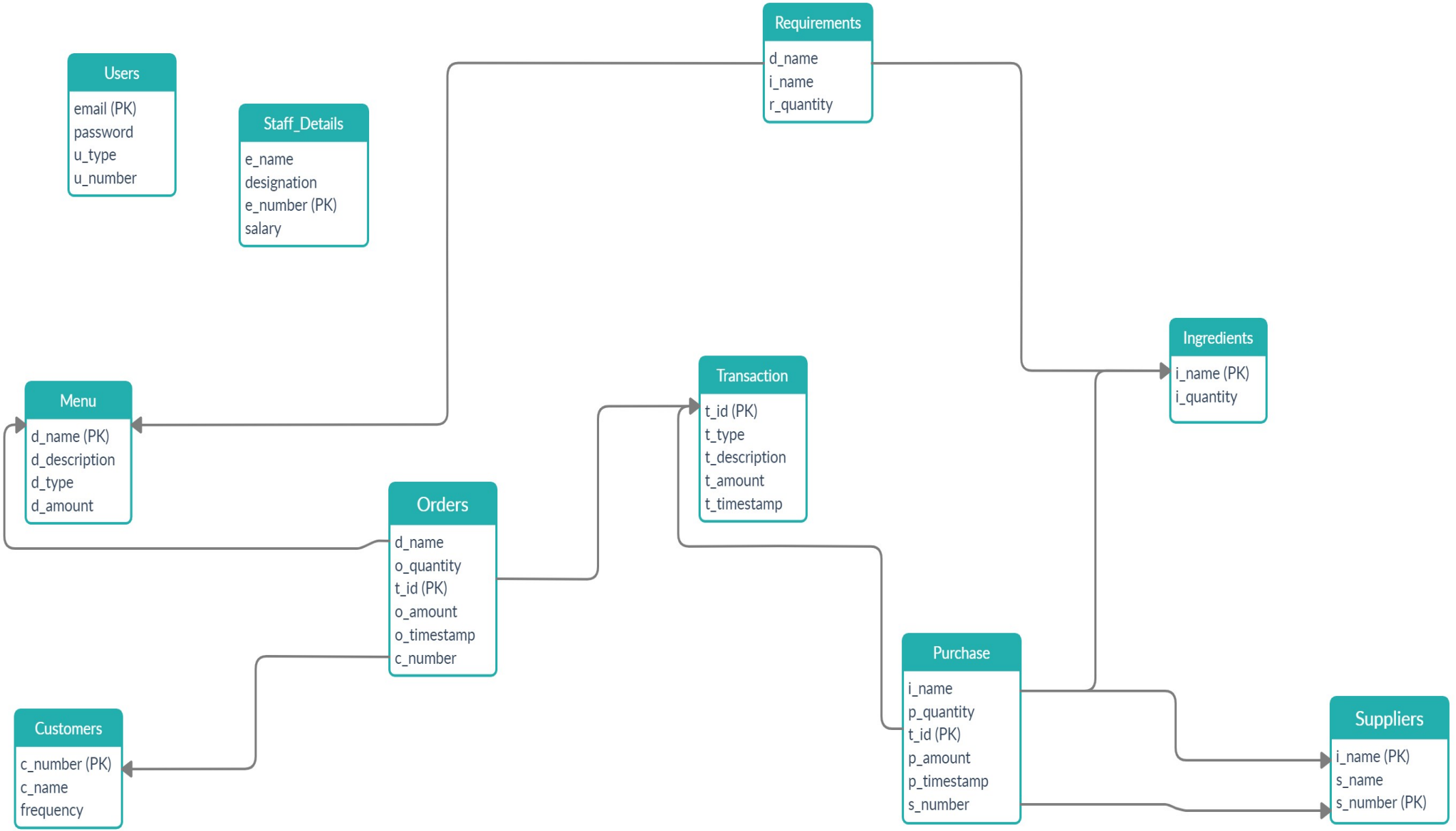
Functional Dependancies and Normalisation

Initial ER diagram:



Initial relational schema:



Menu:

menu
d_name
d_description
d_type
d_amount

Functional dependencies:

- d_name -> d_type
- d_name -> d_description
- d_name -> price

Normalisation:

- Already in 1NF
- Already in 2NF
- Already in 3NF

Ingredients:

ingredients
i_name
i_quantity

Functional dependencies:

- i_name -> i_quantity

Normalisation:

- Already in 1NF
- Already in 2NF

- Already in 3NF

Transactions:

transactions
t_id
t_type
t_description
t_amount
t_timestamp

Functional dependencies:

- t_id -> t_type
- t_id -> t_description
- t_id -> t_amount
- t_id -> t_timestamp

Normalisation:

- Already in NF
- Already in 2NF
- Already in 3NF

Suppliers:

suppliers
i_id
s_name
s_number

Functional dependencies:

- s_number -> s_name

Normalisation:

- Already in 1NF
- To convert to 2NF:

supplier_info
s_name
s_number

ingredient_supplier
i_name
s_number

- Already in 3NF

Customers:

customers
c_name
c_number
frequency

Functional dependencies:

- c_number -> c_name
- c_number -> c_frequency

Normalisation:

- Already in 1NF
- Already in 2NF
- Already in 3NF

Users:

users
email
password
u_type
u_number

Functional dependencies:

- email -> password
- email -> u_type
- email -> u_number

Normalisation:

- Already in 1NF
- Already in 2NF
- Already in 3NF

Staff_details:

staff_details

e_name
designation
salary
e_number

Functional dependencies:

- e_number -> e_name
- e_number -> designation
- e_number -> salary

Normalisation:

- Already in 1NF
- Already in 2NF
- Already in 3NF

Purchase:

purchase
i_name
t_id
p_quantity
p_amount
p_timestamp
s_number

Functional dependencies:

- i_name -> t_id
- i_name-> p_quantity
- i_name -> p_amount
- i_name -> p_timestamp
- i_name -> s_number

Normalisation:

- To convert to 1NF:

This table is not in 1NF because i_name is a multi valued attribute(there can be multiple ingredients in the same purchase). This can be changed by making separate entries for purchase of each different ingredient.

This causes a change in the primary key of this table(primary key becomes {i_name, t_id}), thus changing the functional dependencies as follows:

{i_name, t_id} -> p_quantity

{i_name, t_id} -> p_amount

{i_name, t_id} -> p_timestamp

{i_name, t_id} -> s_number

purchase
i_name
t_id
p_quantity
p_amount
p_timestamp
s_number

- Already in 2NF

- Already in 3NF

Requirements:

requirements
d_id
i_name
r_quantity

Functional dependencies:
- {d_name, i_name} -> r_quantity

Normalisation:
- Already in 1NF
- Already in 2NF
- Already in 3NF

Orders:

orders
d_name
t_id
o_quantity
o_amount
o_timestamp
c_number

Functional dependencies:
- t_id -> d_name

- t_id -> o_quantity
- t_id -> c_number
- t_id -> o_amount
- t_id -> o_timestamp

Normalisation:

- To convert to 1NF:

This table is not in 1NF because d_name is a multi valued attribute(there can be multiple dishes in the same order). This can be changed by making separate entries for order of each different dish by a customer.

This causes a change in the primary key of this table(primary key becomes {d_name, t_id}), thus changing the functional dependencies as follows:

- {d_name, t_id}-> d_name
- {d_name, t_id} -> o_quantity
- {d_name, t_id} -> c_number
- {d_name, t_id} -> o_amount
- {d_name, t_id} -> o_timestamp

orders
d_name
t_id
o_quantity
o_amount
o_timestamp
c_number

- Already in 2NF
- Already in 3NF

Final relational schema: