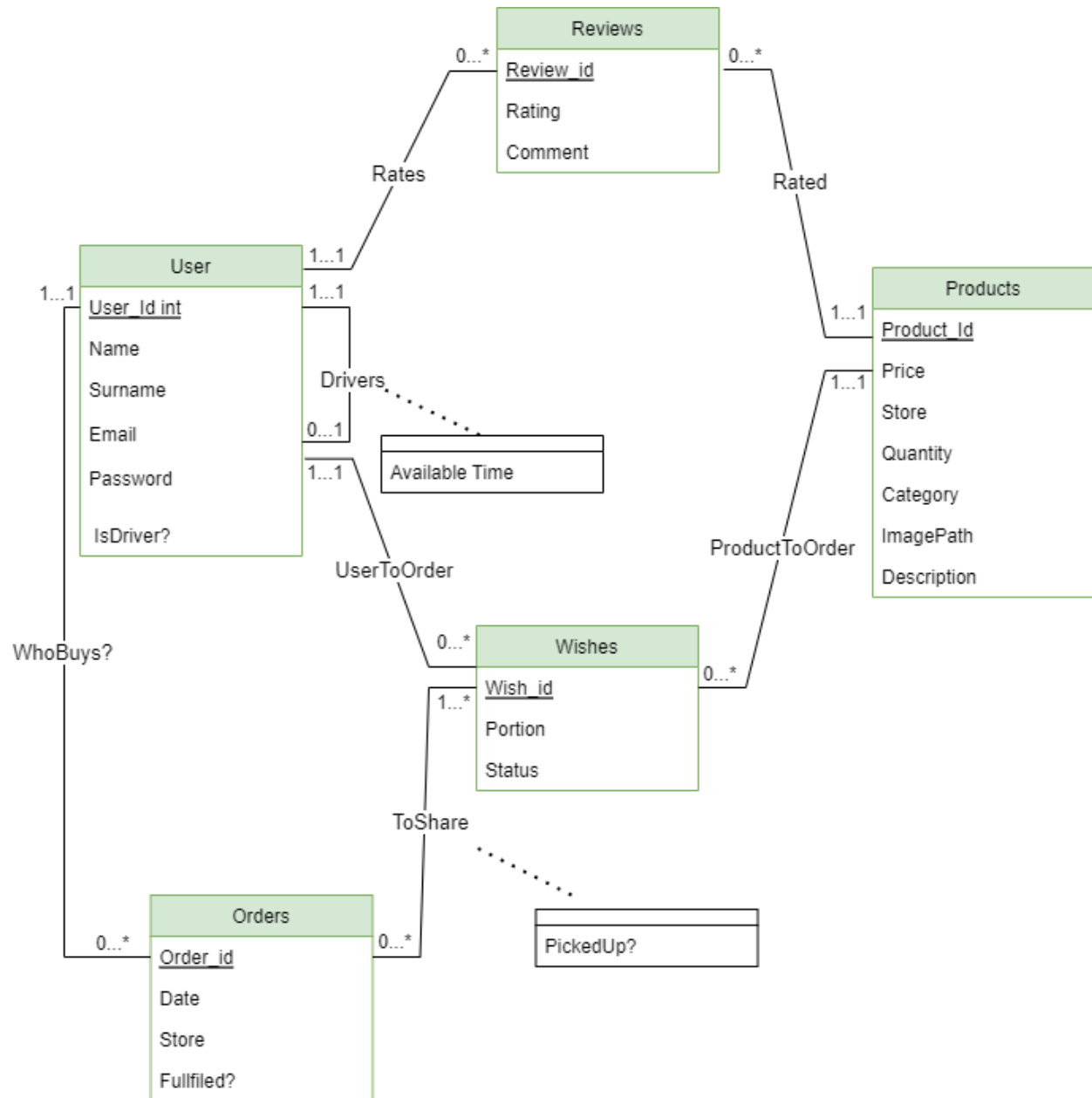


CS 411 – Team 029 OurSQL Stage 2

1. UML diagram



2. Relational Schema

```
TABLE Products (  
  productId int [PK],  
  category varchar(225),  
  price decimal,  
  image varchar(200),  
  store varchar(225),  
  description varchar(2000)  
);
```

```
TABLE Users (  
  userId int [PK],  
  name varchar(225),  
  surname varchar(225),  
  email varchar(50),  
  password varchar(225),  
  isDriver boolean);
```

```
TABLE Driver (  
  driverId int [PK] [FK to Users.userId],  
  availableTime varchar(225) );
```

```
TABLE Orders (  
  orderId int [PK],  
  driver int [FK to Users.userId],  
  date date,  
  store varchar(225),  
  fulfilled boolean );
```

```
TABLE Wishes (  
  wishId int [PK],  
  productId int [FK to Products.productId],  
  userId int [FK to Users.userId],  
  portion decimal,  
  status boolean );
```

```
TABLE ToShare(  
  wishId int [FK to Wishes.wishId],  
  orderId int [FK to Orders.orderId],  
  pickedUp boolean);
```

```
TABLE Reviews(  
  reviewId int [PK],  
  userId int [FK to User.userId],  
  productId int [FK to Products.productId],  
  rating int ,  
  comments varchar(1000));
```

3. Entity descriptions and assumptions

Users: The table represents the users of our application using their personal data and login information.

Products: The products are uniquely identified by the product_id. The products have attributes as store where product is sold, the quantity of items in a product (1 bread, 1 bottle of water, 1 pack of eggs, or 12 bottles of soda a single pack), price of a product, category of product (dairy, produce etc.), imagePath which is path to the image in the servers file system. Description which is a short description of a product provided by a store.

Reviews: Uniquely represented by Review_id, other attributes include the rating (1-5) and the comment from user regarding that product.

Wishes: The wish in our table represents some product and its corresponding portion which each user wants. Each wish in our application is uniquely represented by the wish_id. The user_id and product_id represents the user and product of a wish respectively. The portion is an integer value; it shows the amount of wanted products which can be shared. Our assumption is that any product which has quantity more than 1 is sharable. The status attribute indicates whether this wish is active or not (i.e., whether this wish should be accounted when forming the orders for a driver.) Otherwise, we assume that this wish completed.

Orders: This entity represents the orders of our application. The order is uniquely identified by the Order_id, yet every order has several wish_ids associated with it. Other attributes also include Date (date and time when the order was created) , Store (the store name where the driver will buy the products for order), and a boolean attribute to mark whether the order has been fulfilled. Our assumptions related to orders are:

- 1) the new order is created only when there are certain number of wishes with corresponding products, the portions of which in total will be equal to the quantity of sold product (e.g., when N number of users wanted the same water brand in the amount which will give 24 bottles, which is the number of bottles in a single pack sold in a store.)
- 2) The driver can be assigned to many orders which can be from different stores.
- 3) The driver will deliver the products to a single hub where customers can take their portions. The users will be notified after order is fulfilled.

4. Relations descriptions and assumptions

WhoBuys - a connection between aggregated orderId and a user who 1) opted to be a driver; 2) is assigned to buy this order from the store. Assumption: each order is for sure assigned to one driver (user) and a driver can have 0 or many assigned orders.

ToShare - is a many-to-many relationship between Wishes and Orders. It is the main goal of our web-application – to have a table of orders with products shared among several people.

Assumptions: each of these wishes can be in several orders (cases when a user orders the same stuff again). Each Order usually corresponds to several wishes. Ex: one order of 12-pack water bottles can be shared by several people. Also has an attribute "PickedUp?" showing if the user picked up the products they ordered.

ProductToOrder - is a 1-to-many relation between Products and Wishes entities. Indicates which product_id is wanted by the user. Assumptions: each wish corresponds to exactly one product and products can be listed in several wishes.

UserToOrder - is a 1-to-many relation between Users and Wishes entities. Indicates who owns this wish. Assumptions: each wish corresponds to exactly one user and users can have wishes.

Drivers - is a relation of users to themselves. It shows which users have agreed to be drivers (IsDriver attribute). In case the user can be a driver, they also put down their availability time. Assumptions: each user can be driver or not, each driver is a user for sure.

Rates/Rated - is a relation indicating who posted a review and on what product. Assumptions:
Every review corresponds to exactly one user and product. Users and products can have many
(or zero) reviews.