

CS 306 PROJECT PROPOSAL - ONLINE SHOPPING DATABASE (OSDB)

Group Members:

Rana İşlek, 27836

Tuna Karagül, 28027

Göktuğ korkulu, 27026

Alp Civan, 28408

Hasan Kadir Uçmazoğlu, 28034

Project Description

The purpose of this project is creating an online shopping site which is a type of electronic commerce that allows customers to buy goods or services directly from a seller via the Internet using a web browser or a mobile application. The user's address, email address, and phone number will be saved on this site. The advantage of this system will be primarily; The user will be able to see the remaining stock number and price of the product directly and can buy it with the credit card which is registered on the site. From the point of view of the seller; When the seller makes the necessary actions on the site about the stock number and price of the product, the only thing the firm has to do is to deliver the product to the shipping company. The firm will use the shipment number determined by the site while delivering the product.

Entities:

Shopping website

Product

Customer

Payment

Credit Card

Manufacturer

Seller

Shipping

Attributes:

address

cost, id, stock, comment

email, id, password address

total cost, payment method, id, time

id

s_name, s_id

cost, id , company

Relationships:

Manufacturer produces product

Seller buys product

Shop website has product

Customer enters shopping website

Customer buys product

Customer pays payment

Payment purchase with credit card

Seller send to shipping

Shipping goes to customer

Credit Card: The credit card which is registered on the shopping website is the way to make a payment. “Card-ID” is the only attribute of the credit card and the only relationship associated with it is the payment entity.

Customer: Customer is the entity who purchases goods from the shopping website. The relationships of customers are as follows; **‘Buy’** is the relationship between customer and product, the customer buys the product. **‘Pays’** relationship between customer and payment means that customer pays the payment. The contributions of ‘Customer’ are customer email,id,password and address.

Manufacturer: Manufacturer is the producer who **“produces”** the product, so manufacturer is an entity and **“produces”** is a relationship between manufacturer and product. Seller **“buys/orders”** the items that s/he sells on the website from her/him, therefore **“buys”** is a relationship between manufacturer and seller.

Payment: Payment is the entity which customers pay to purchase with a credit card. Its attributes are total cost, payment method, payment-ID and time. **“Pays”** is the relationship between customer and payment and **“purchase with”** is the relationship between credit card and payment.

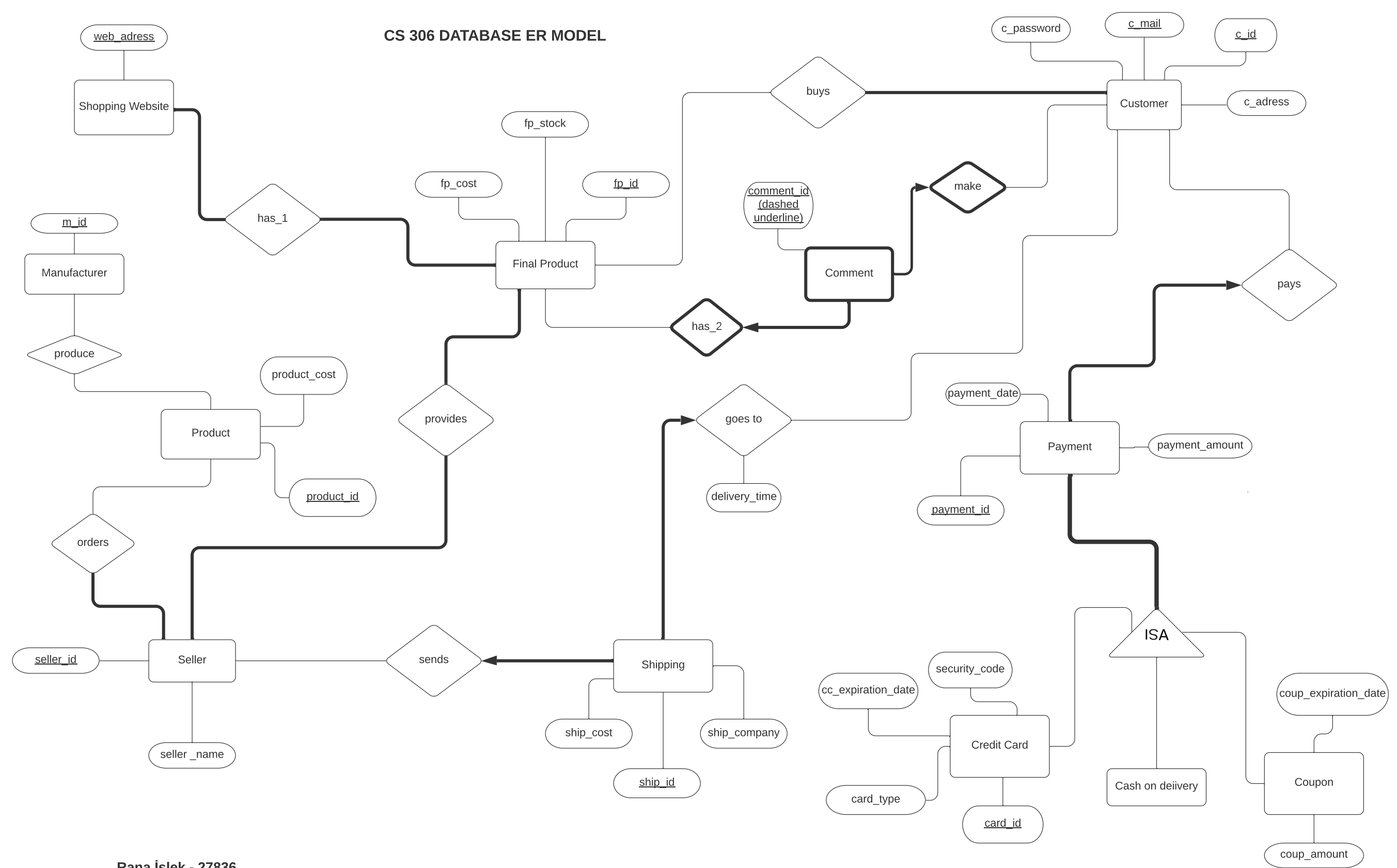
Product: Products are the items which can be digital or physical that are sold on the website. “Products” is an entity and its attributes are product cost, product id, stock and comments. It has relationships with customer, manufacturer, shop website and seller. Manufacturer **“produces”** product, then seller **“buys”** the product. Therefore shop website **“has”** product and finally customer **“buys”** product.

Seller: Seller is an intermediary person or company that makes financial transactions between producer/manufacturer and customer. Firstly, the seller **‘buys’** products from the manufacturer. After adding some commission to the purchase price, the seller **‘directs’** the product to the customer by using shipping. Intrinsically, the seller has an id number and name of its own as attributes.

Shopping website: This is the website which customers visit to do their shopping. The shopping website includes all of the products that are on sale which is shown by **‘has’** relationship. The only contribution of this entity is the **‘Address’**. The address is the information of the web address of the shopping website.

Shipping: Shipping is the way that the seller sends the ordered items to the customer. Shipping is the entity and its attributes are shipping id, cost and shipping company. After shipping information is received by the shipping company from the seller, they **‘ship’** the product to the customer who has already paid the corresponding amount of money to the seller.

CS 306 DATABASE ER MODEL



Rana İşlek - 27836
Tuna Karagül - 28027
Hasan Kadir Uçmazoğlu - 28034
Alp Civan - 28408
Göktaş Korkulu - 27026

ER to Relational Model Conversion

```
CREATE TABLE Shopping Website(  
    web_adress CHAR(20),  
    PRIMARY KEY(web_adress)  
)
```

```
CREATE TABLE has_1(  
    web_adress CHAR(20) NOT NULL,  
    fp_id INTEGER NOT NULL,  
    PRIMARY KEY(web_adress, fp_id),  
    FOREIGN KEY (web_adress) REFERENCING Shopping Website,  
    FOREIGN KEY (fp_id) REFERENCING Final Product,  
)
```

```
CREATE TABLE Final Product (  
    fp_cost REAL,  
    fp_id INTEGER,  
    fp_stock INTEGER,  
    PRIMARY KEY(fp_id).  
)
```

```
CREATE TABLE provides(  
    fp_id INTEGER NOT NULL,  
    seller_id INTEGER NOT NULL,  
    PRIMARY KEY (fp_id, seller_id),  
    FOREIGN KEY (fp_id) REFERENCING Final Product,  
    FOREIGN KEY (seller_id) REFERENCING Seller,  
)
```

```
CREATE TABLE Seller(  
    seller_id INTEGER,  
    seller-name CHAR(20),  
    PRIMARY KEY(seller_id),  
)
```

```
CREATE TABLE orders(  
    seller_id INTEGER,  
    product_id INTEGER NOT NULL,  
    PRIMARY KEY(seller_id, product_id),  
    FOREIGN KEY (seller_id) REFERENCING Seller,  
    FOREIGN KEY (product_id) REFERENCING Product,
```

)

```
CREATE TABLE Product(  
    product_cost REAL,  
    Product-id INTEGER,  
    PRIMARK KEY (product_id),  
)
```

```
CREATE TABLE produce(  
    m_id INTEGER,  
    product_id INTEGER,  
    PRIMARY KEY (m_id, product_id),  
    FOREIGN KEY (m_id) REFERENCING Manufacturer,  
    FOREIGN KEY (product_id) REFERENCING Product,  
)
```

```
CREATE TABLE manufacturer(  
    m_id INTEGER,  
    PRIMARY KEY(m_id),  
)
```

```
CREATE TABLE Shipping_Sends(  
    ship_cost REAL,  
    ship_id INTEGER,  
    ship_company CHAR(20),  
    Seller_id INTEGER NOT NULL,  
    PRIMARY KEY (ship_id),  
    FOREIGN KEY (seller_id) REFERENCING Seller,  
)
```

```
CREATE TABLE sends(      --ALTERNATIVE TO ABOVE TABLE w/ combining the below table  
    ship_id INTEGER,  
    seller_id INTEGER NOT NULL,  
    PRIMARY KEY (ship_id),  
    FOREIGN KEY (ship_id) REFERENCING Shipping,  
    FOREIGN KEY (seller_id) REFERENCING Seller,  
)
```

```
CREATE TABLE Shipping(  
    ship_cost REAL,  
    ship_id INTEGER,  
    ship_company CHAR (20),  
    PRIMARY KEY (ship_id),  
)
```

```
CREATE TABLE goes_to(  
    Ship_id INTEGER,  
    delivery_time DATE,  
    c_id INTEGER NOT NULL,  
    c_mail CHAR(20) NOT NULL,  
    PRIMARY KEY (ship_id)  
    FOREIGN KEY (c_mail) REFERENCES Customer,  
    FOREIGN KEY (c_id) REFERENCES Customer,  
)
```

```
CREATE TABLE Customer(  
    c_password CHAR(20),  
    c_mail CHAR(20),  
    c_id INTEGER,  
    c_adress CHAR(200),  
    PRIMARY KEY (c_mail),  
    PRIMARY KEY (c_id),  
)
```

```
CREATE TABLE buys(  
    fp_id INTEGER NOT NULL,  
    c_mail CHAR(20),  
    c_id CHAR(20),  
    PRIMARY KEY (fp_id, c_mail, c_id),  
    FOREIGN KEY (fp_id) REFERENCES Final Product.  
    FOREIGN KEY (c_mail) REFERENCES Customer,  
    FOREIGN KEY (c_id) REFERENCES Customer,  
)
```

```
CREATE TABLE comments_have(  
    comment_id INTEGER,  
    Fp_id INTEGER NOT NULL  
    PRIMARY KEY (comment_id, fp_id),  
    FOREIGN KEY (fp_id) REFERENCES Final Product,  
    ON DELETE CASCADE  
)
```

```
CREATE TABLE comments_made_by(  
    comment_id INTEGER,  
    c_password CHAR(20),  
    c_mail CHAR(20) NOT NULL,  
    c_id INTEGER NOT NULL,  
    c_adress CHAR(20),
```

```
        PRIMARY KEY(comment_id, c_id),
        FOREIGN KEY (c_id) REFERENCES Customer,
        ON DELETE CASCADE
    )
```

```
CREATE TABLE pays(
    c_id INTEGER NOT NULL,
    c_mail CHAR(20) NOT NULL,
    payment_id INTEGER,
    PRIMARY KEY (payment_id),
    FOREIGN KEY (c_id) REFERENCES Customer,
    FOREIGN KEY (c_mail) REFERENCES Customer,
)
```

```
CREATE TABLE Payment(
    payment_date DATE,
    payment_amount REAL,
    Payment_id INTEGER,
    PRIMARY KEY(payment_id),
)
```

```
CREATE TABLE payment_paying(           --alternative for above two tables
    payment_date DATE,
    payment_amount REAL,
    payment_id INTEGER,
    c_id INTEGER NOT NULL,
    PRIMARY KEY (payment_id),
    FOREIGN KEY (c_id) REFERENCES Customer,
)
```

```
CREATE TABLE Credit Card(
    card_type CHAR(10),
    cc_expiration_date DATE,
    security_code INTEGER(3),
    card_id INTEGER(16),
    payment_date DATE,
    payment_id INTEGER,
    payment_amount REAL,
    PRIMARY KEY(payment_id, card_id)
)
```

```
CREATE TABLE Cash on Delivery(
    payment_date DATE,
```

```
    payment_id INTEGER,  
    payment_amount REAL,  
    PRIMARY KEY(payment_id),  
)
```

```
CREATE TABLE Coupon(  
    coup_expiration_date DATE,  
    coup_amount REAL,  
    payment_amount REAL,  
    payment_date DATE,  
    paymet_id INTEGER,  
    PRIMARY KEY (payment_id),  
)
```