

CS 353 Database Management Systems Project

Design Report

Food Ordering and Delivery System

Group 19

Batuhan Özçömlekçi, 21703297

Yusuf Ziya Özgül, 21703158

Musa Ege Ünalın, 21803617

Mustafa Göktan GÜdükbay, 21801740

ER Diagram

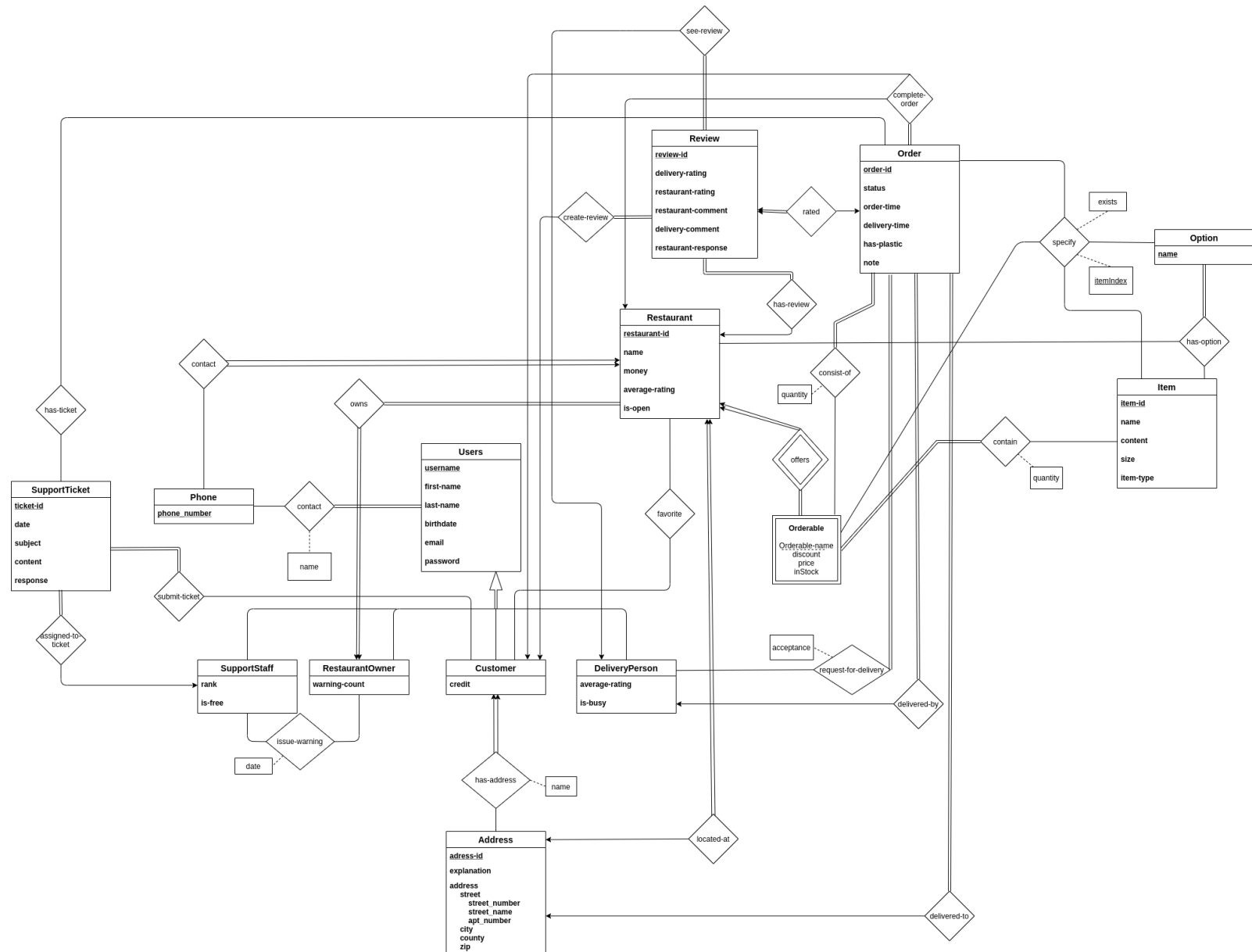


Table Schemas

Item(item-id, name content, size, itemtype)

Option(name)

Order(order-id, status, order-time, delivery-time, has-plastic, note)

Review(review-id, delivery-rating, restaurant-rating, restaurant-comment, delivery-comment, restaurant-response)

Restaurant(restaurant-id, name, money, average-rating, is-open)

Orderable(restaurant-id, orderable-name, discount, price, instock)

- The attribute restaurant-id is a foreign key to Restaurant

Address(address-id, explanation, street, street_number, street_name, apt_number, city, county, zip)

Users(username, first-name, last-name, birthdate, email, password)

DeliveryPerson(username, average-rating, is-busy)

- The attribute username is a foreign key to Users

Customer(username, credit)

- The attribute username is a foreign key to Users

RestaurantOwner(username, warning-count)

- The attribute username is a foreign key to Users

SupportStaff(username, rank, is-free)

- The attribute username is a foreign key to Users

SupportTicket(ticket-id, date, subject, content, response)

Phone(phone-number)

Contain(restaurant-id, orderable-name, item-id, quantity)

- The attributes orderable-name and restaurant-id is a foreign key to Orderable
- The attribute item-id is a foreign key to Item

HasOption(restaurant-id, option-name, item-id)

- The attribute restaurant-id is a foreign key to Restaurant
- The attribute option-name is a foreign key to Option
- The attribute item-id is a foreign key to Item

Specify(item-id, option-name, order-id, restaurant-id, orderable-name, item-index, exists)

- The attribute item-id is a foreign key to Item
- The attribute option-name is a foreign key to Option
- The attribute order-id is a foreign key to Order
- The attributes orderable-name and restaurant-id is a foreign key to Orderable

Offers(restaurant-id, orderable-name)

- The attribute restaurant-id is a foreign key to Restaurant
- The attribute orderable-name is a foreign key to Orderable

ConsistOf(order-id, restaurant-id, orderable-name, quantity)

- The attribute order-id is a foreign key to Order
- The attributes orderable-name and restaurant-id is a foreign key to Orderable

HasReview(restaurant-id, review-id)

- The attribute restaurant-id is a foreign key to Restaurant
- The attribute review-id is a foreign key to Review

Rated(review-id, order-id)

- The attribute order-id is a foreign key to Rated
- The attribute review-id is a foreign key to Review

CompleteOrder(order-id, username, restaurant-id)

- The attribute username is a foreign key to Customer
- The attribute restaurant-id is a foreign key to Restaurant
- The attribute order-id is a foreign key to Order

SeeReview(review-id, username)

- The attribute username is a foreign key to DeliveryPerson
- The attribute review-id is a foreign key to Review

LocatedAt(address-id, restaurant-id)

- The attribute restaurant-id is a foreign key to Restaurant
- The attribute address-id is a foreign key to Address

HasAddress(address-id, username, name)

- The attribute username is a foreign key to Customer
- The attribute address-id is a foreign key to Address

IssueWarning(support-staff-username, restaurant-owner-username, date)

- The attribute support-staff-username is a foreign key to SupportStaff
- The attribute restaurant-owner-username is a foreign key to RestaurantOwner

AssignedToTicket(ticket-id, username)

- The attribute username is a foreign key to SupportStaff
- The attribute ticket-id is a foreign key to SupportTicket

SubmitTicket(ticket-id, username)

- The attribute username is a foreign key to Customer
- The attribute ticket-id is a foreign key to SupportTicket

HasTicket(ticket-id, order-id)

- The attribute order-id is a foreign key to Order
- The attribute ticket-id is a foreign key to SupportTicket

Favorite(username, restaurant-id)

- The attribute username is a foreign key to Customer
- The attribute restaurant-id is a foreign key to Restaurant

Contact(username, phone-number, name)

- The attribute username is a foreign key to Customer
- The attribute phone-number is a foreign key to Phone

RestaurantContact(restaurant-id, phone-number)

- The attribute restaurant-id is a foreign key to Restaurant
- The attribute phone-number is a foreign key to Phone

Owns(restaurant-id, username)

- The attribute username is a foreign key to RestaurantOwner
- The attribute restaurant-id is a foreign key to Restaurant

CreateReview(review-id, username)

- The attribute username is a foreign key to Customer
- The attribute review-id is a foreign key to Review

RequestForDelivery(username, order-id, acceptance)

- The attribute username is a foreign key to DeliveryPerson
- The attribute order-id is a foreign key to Order

DeliveredBy(order-id, username)

- The attribute username is a foreign key to DeliveryPerson
- The attribute order-id is a foreign key to Order

DeliveredTo(order-id, address-id)

- The attribute address-id is a foreign key to Address
- The attribute order-id is a foreign key to Order

User Interface Design and Corresponding Statements

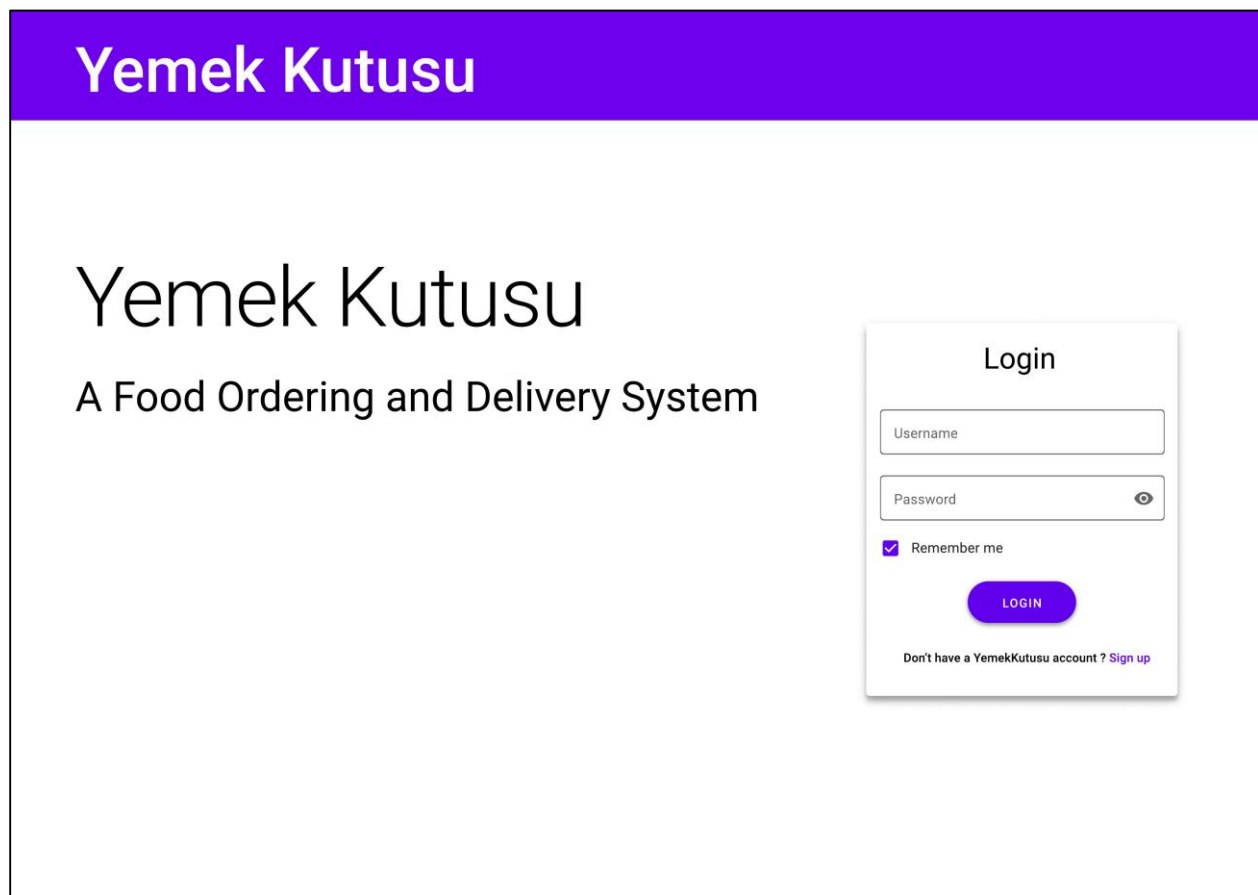
Login Page

Inputs: @username, @password

SQL Query:

```
SELECT count(*)  
FROM Users U  
WHERE U.user-name == @username AND U.password = @password;
```

User Interface:



The image displays the user interface for the login page of 'Yemek Kutusu', a food ordering and delivery system. The page features a purple header with the text 'Yemek Kutusu'. Below the header, the main content area is white. On the left side, the text 'Yemek Kutusu' is displayed in a large, black, sans-serif font, followed by the subtitle 'A Food Ordering and Delivery System' in a smaller, black, sans-serif font. On the right side, there is a login form with a white background and a subtle shadow. The form is titled 'Login' in a bold, black, sans-serif font. It contains two input fields: 'Username' and 'Password'. The 'Password' field has a small eye icon to its right, indicating a toggle for password visibility. Below the input fields, there is a checkbox labeled 'Remember me' which is checked. At the bottom of the form, there is a purple button with the text 'LOGIN' in white, uppercase letters. Below the button, there is a link that says 'Don't have a YemekKutusu account ? Sign up' in a small, black, sans-serif font, with 'Sign up' in purple.

Signup Page

SQL Query:

Inputs: @username, @first-name, @last-name, @birthdate, @email, @password

Shared for all users:

```
INSERT INTO Users VALUES(@username, @first-name, @last-name, @birthdate, @email,  
                           @password)
```

If user type is support:

```
INSERT INTO SupportStaff VALUES(0, true)
```

If user type is restaurant owner:

```
INSERT INTO RestaurantOwner VALUES(0)
```

If user type is customer:

```
INSERT INTO Customer VALUES(0)
```

If user type is delivery person:

```
INSERT INTO DeliveryPerson VALUES(NULL, false)
```

User Interface:

Yemek Kutusu

Sign up

☒ Customer
☐ Restaurant Owner
☐ Driver
☐ Support

SIGN UP

Homepage

Input: @username

SQL Query:

```
SELECT username, first-name, last-name, birthdate, email  
FROM Users  
WHERE @username = username;
```

If user type is customer:

Display Credit:

```
SELECT credit  
FROM Customer  
WHERE @username = username;
```

Display Addresses:

```
SELECT *  
FROM Address NATURAL JOIN HasAddress  
WHERE username = @username;
```

Display Phone Numbers:

```
SELECT *  
FROM Phone NATURAL JOIN Contact  
WHERE username = @username;
```

User Interface:

YemekKutusu

Item Restaurant Address

Search

ORDERSBOXSUPPORT

USER_NAMELOGOUT

Username: USER_NAME

First Name: User

Last Name: Name

Birthdate: 01/01/1999

Email: user@gmail.com

Change Password

Credit: 200 TL

Add Credit

Address 1
Bilkent Main Campus 75. Dormitory, Çankaya, Ankara

Address 2
Bilkent Main Campus 80. Dormitory, Çankaya, Ankara

NEW ADDRESS

Phone Number 1
0555 555 5555

NEW PHONE NUMBER

Search an Item

Input: @search_keyword

SQL Query:

```
SELECT R.restaurant-id, R.name, R.rating, Add.street, Add.street_number, Add.street_name,
Add.apt_number, Add.city, Add.county, Add.zip, Or.orderable-name
```


```
FROM Restaurant R, offers Of, Orderable Or, locatedAt Loc , Address Add, Item I,contain Con
```

```
WHERE R.is-open = TRUE AND R.restaurant-id = Of.restaurant-id AND Or.orderable-name =
Of.orderable-name AND I.name LIKE ‘%@search_keyword%’ AND Loc.address-id =
add.address-id AND Loc.restaurant-id = R.restaurant-id AND Or.orderable-name =
Con.orderable-name AND Con.item-id = I.item-id
```

User Interface:

YemekKutusu Item ORDERS BOX SUPPORT USER_NAME LOGOUT

Search Results for “burger”



Burger King
Çankaya, Bilkent 1, Ankara
★★★★☆


Hamburger

Chicken Burger

Fish Burger

Chicken Burgers Menu

Hamburgers Menu



McDonald's
Çankaya, Kizilay, Ankara
★★★★☆

Chicken Burger

Hamburger

Deluxe Hamburger Menu

Search a Restaurant by Name

Input: @search_keyword

SQL Query:

```
SELECT Res.name, Res.average-rating, Res.is-open  
FROM Restaurant Res,  
WHERE Res.name LIKE '@search_keyword%'
```

Search a Restaurant by Address

Input: @search_keyword

SQL Query:

```
SELECT Res.name, Res.average-rating, Res.is-open  
FROM Restaurant Res  
WHERE add.street LIKE '@search_keyword%' or add.street_number LIKE  
'@search_keyword%' or add.street_name LIKE '@search_keyword%' or add.city LIKE  
'@search_keyword%' or add.apt_number LIKE '@search_keyword%' or add.county LIKE  
'@search_keyword%' or add.zip LIKE '@search_keyword%'
```

View Restaurant Menu

Input: @restaurant_id

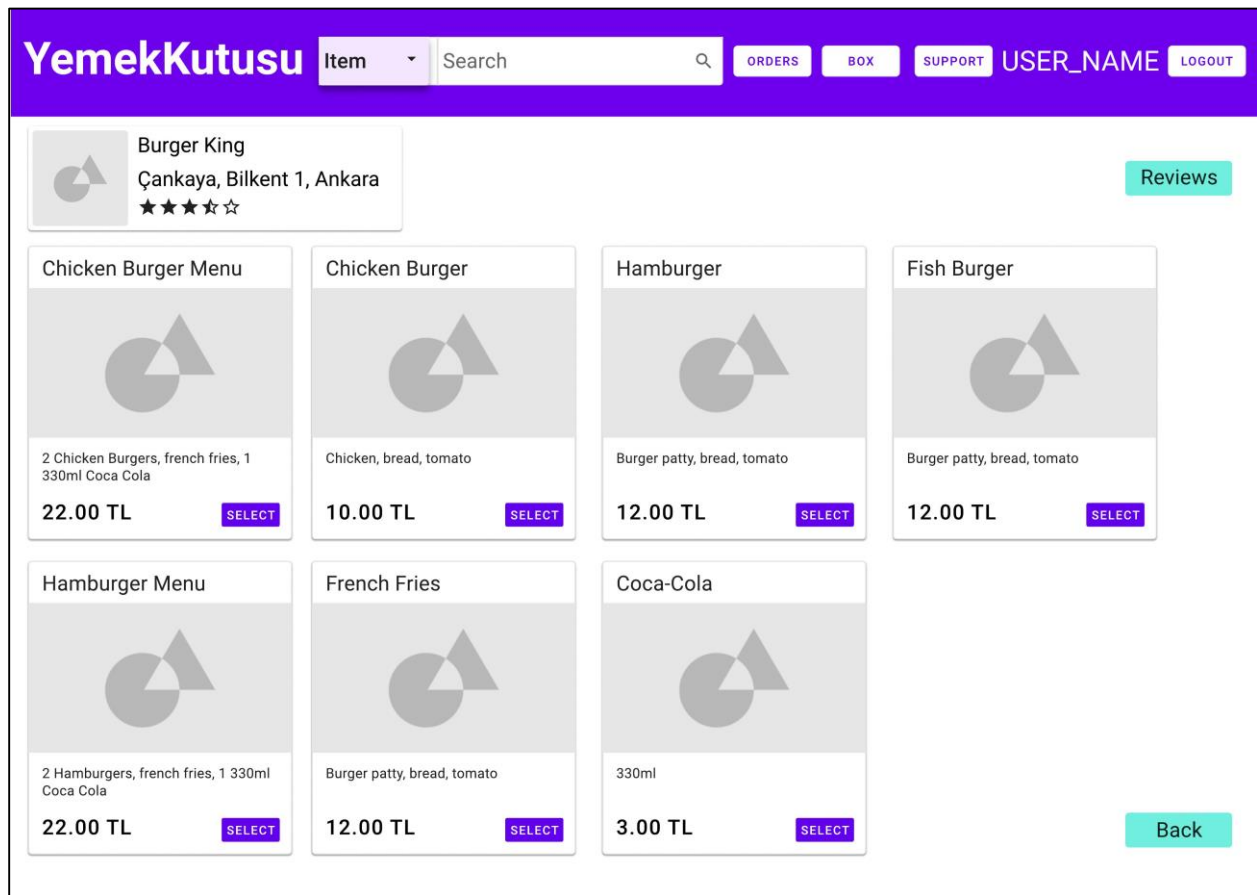
SQL Query:

SELECT O.orderable-name, O.price, Con.quantity, I.name, I.content

FROM offers Of, Orderable O, contain Con, Item I

WHERE Of.restaurant-id = "@restaurant_id" AND O.orderable-name = Of.orderable-name
AND Con.orderable-name = O.orderable-name AND I.item-id = Con.item-id

User Interface:



View Options

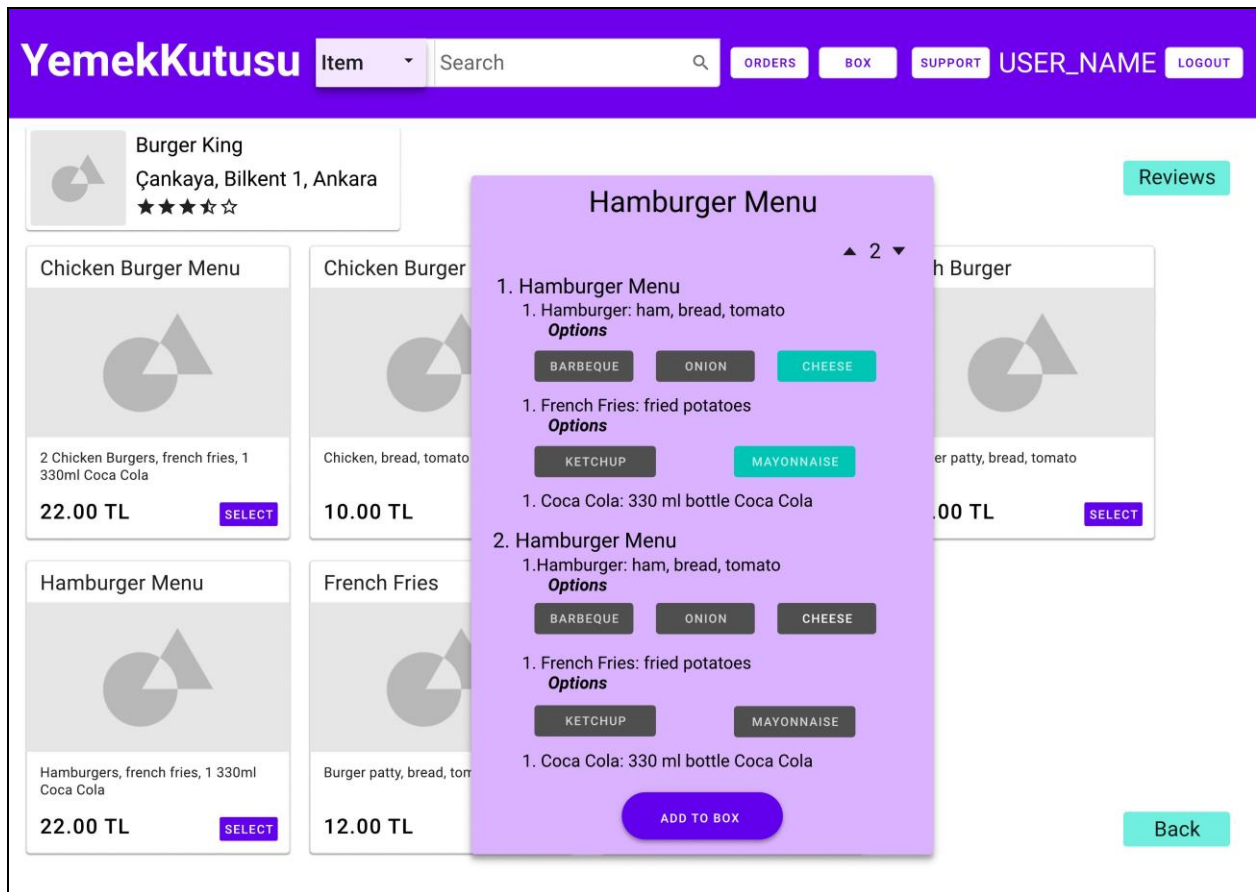
SQL Query:

SELECT I.name, Op.name

FROM Item I, Option Op, hasOption has, contain Con

WHERE Con.orderable-name = @orderable-name AND Con.item-id = I.item-id AND has.item-id = I.itemId AND Op.name = has.name

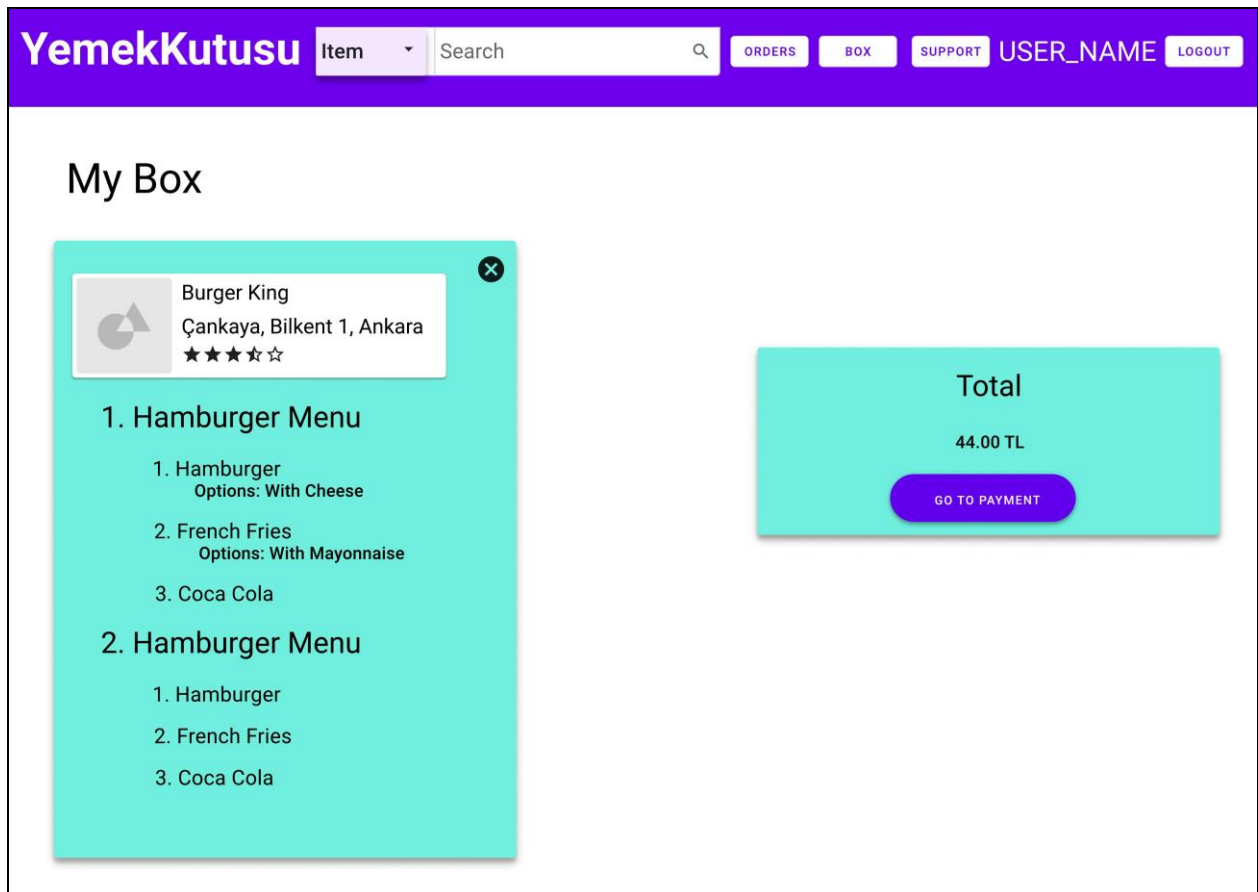
User Interface:



List Box of a Customer

No SQL query is needed for this functionality because the order is not completed yet. The customer is just listing her/his box.

User Interface:



Make Order

Inputs: @user-name, @order-id @orderable-name, @restaurant-id, @quantity, @time, @has-plastic, @note, @item-id, @option-name, @item-index, @exists

SQL Insertion Statements

These insertions will be in a loop for each ordered menu, each item, each option. Parameters for such cases are for only one menu, item or option.

INSERT INTO ConsistOf VALUES(@order-id, @orderable-name, @restaurant-id, @quantity)

This query is for getting the item-id s of an orderable menu.

SELECT C.item-id

FROM Contain C

WHERE C.restaurant-id = @restaurant-id and orderable-name = @orderable-name

INSERT INTO Specify VALUES(@item-id, @option-name, @order-id, @restaurant-id, @orderable-name, @item-index, @exists)

INSERT INTO Order VALUES(@order-id, "Prepared", @time, NULL, @has-plastic, @note)

INSERT INTO CompleteOrder VALUES(@order-id, @username, @restaurant-id)

Make Order

Inputs: @user-name, @order-id @orderable-name, @restaurant-id, @quantity, @time, @has-plastic, @note, @item-id, @option-name, @item-index, @exists

SQL Insert Statements.

These insertions will be in a loop for each ordered menu, each item, each option. Parameters for such cases are for only one menu, item or option.

INSERT INTO ConsistOf **VALUES**(@order-id, @orderable-name, @restaurant-id, @quantity)

This query is for getting the item-id s of an orderable menu.

SELECT C.item-id

FROM Contain C

WHERE C.restaurant-id = @restaurant-id and orderable-name = @orderable-name

INSERT INTO Specify **VALUES**(@item-id, @option-name, @order-id, @restaurant-id, @orderable-name, @item-index, @exists)

INSERT INTO Order **VALUES**(@order-id, "Prepared", @time, NULL, @has-plastic, @note)

INSERT INTO CompleteOrder **VALUES**(@order-id, @username, @restaurant-id)

SQL Query:

SELECT add.street, add.street_number, add.street_name, add.apartment_number, add.city,
add.county, add.zip

FROM HasAddress has, Address add

WHERE has.username = @username AND has.address-id = add.address-id

Order Confirmation

User Interface:

YemekKutusu Item Search [ORDERS](#) [BOX](#) [SUPPORT](#) USER_NAME [LOGOUT](#)

Order Confirmation

Choose Address

Address 1
Bilkent Main Campus 75. Dormitory, Çankaya, Ankara

Address 2
Bilkent Main Campus 80. Dormitory, Çankaya, Ankara

[NEW ADDRESS](#)

☐ Deliver Now
☒ Deliver Later

ENTER TIME
| : 00 AM
Hour Minute PM

Note

Total: 44.00 TL


[CANCEL](#) [BACK](#) [PAY](#)

List Orders of a Customer

User Interface:

YemekKutusu Item ORDERS BOX SUPPORT USER_NAME LOGOUT

Order History




Burger King
Çankaya, Bilkent 1, Ankara
★★★★☆

Being Prepared

2 Hamburger Menu

Total: 44.00 TL

MORE INFO REVIEW REPEAT



Burger King
Çankaya, Bilkent 1, Ankara
★★★★☆

Delivered

1 Chicken Burger Menu

Total: 22.00 TL

MORE INFO SEE REVIEW REPEAT

Check Details of an Order

Inputs: @username

SQL Query:

SELECT Ord.name, O.order-time,R.name, add.street, add.street_number, add.street_name, add.apt_number, add.city, add.county, add.zip, R.rating, Ord.name, I.name, Op.name, Del.username

FROM Order O, Orderable Ord, ConsistOf Cof, Contain Con, CompleteOrder Ord2, Restaurant R, LocatedAt Loc, Address add, Item I, Option Op, HasOption hasO, DeliveredBy Del

WHERE @username = Ord2.username AND Cof.order-id = O.order-id AND Ord.orderable-name = Cof.orderable-name AND R.restaurant-id = Ord2.restaurant-id AND Loc.restaurant-id = R.restaurant-id AND Loc.address-id = add.address-id AND Ord.order-id = Cof.order-id AND Del.order-id = O.order-id AND Op.name = hasO.name AND I.item-id = hasO.item-id AND O.order-id = Ord2.order-id

User Interface:

The screenshot displays the YemekKutusu web application interface. At the top, there is a purple header with the logo, a search bar, and navigation links for ORDERS, BOX, SUPPORT, USER_NAME, and LOGOUT. The main content area is divided into two sections. On the left, under 'Order History', there are two order cards. The first card shows a Burger King restaurant with a 4.5-star rating and a '2 Hamburger Menu' with buttons for MORE INFO, REVIEW, and REPEAT. The second card shows the same restaurant with a 4.5-star rating and a '1 Chicken Burger Menu' with buttons for MORE INFO, SEE REVIEW, and REPEAT. On the right, a detailed order modal is open for Order ID: 1234567. It shows the restaurant name, address, and rating. The order date is 15/09/2020 at 15:05. The menu items are listed as follows: 1. Hamburger (Options: With Cheese), 2. French Fries (Options: With Mayonnaise), and 3. Coca Cola. Below the menu items, it says 'Delivered by delivery_person_name'. The total amount is 44.00 TL.

Write Comments for an Order

Inputs: @reviewID, @restaurantComment, @deliveryComment, @restaurantRating,
@driverRating

SQL Insert Statement:

INSERT INTO Review

VALUES(@reviewID, @driverRating, @restaurantRating, @restaurantComment,
@deliveryComment, NULL)

User Interface:

The screenshot displays the YemekKutusu web application. The header is purple with the logo 'YemekKutusu', a search bar, and navigation links: 'ORDERS', 'BOX', 'SUPPORT', 'USER_NAME', and 'LOGOUT'. The main content area shows an 'Order History' section with two items from 'Burger King' at 'Çankaya, Bilkent 1, Ankara'. The first item is '2 Hamburger Menu' and the second is '1 Chicken Burger Menu'. A modal window is open over the second item, allowing the user to provide feedback. The modal contains text input fields for 'Restaurant Comment' and 'Driver Comment', and star rating sliders for 'Restaurant Rating' and 'Driver Rating'. At the bottom of the modal are 'CANCEL' and 'SEND' buttons.

See the Comment Written for an Order

Inputs: @order_id

SQL Query:

```
SELECT R.name, R.average-rating, add.street, add.street_number, add.street_name,  
       add.apt_number, add.city, add.county, add.zip, R.rating, Rev.restaurant-comment,  
       Rev.driver-comment, Rev.restaurant-rating, Rev.driver-rating, Rev.restaurant-response  
FROM Rated Rate, Restaurant R, LocatedAt Loc, Address add, Review Rev  
WHERE @order_id = Rate.order-id AND Rev.review-id = Rate.review-id AND  
       Loc.restaurant-id = R.restaurant-id AND Loc.address-id = add.address-id
```

User Interface:

The screenshot displays the YemekKutusu web application. The header is purple with the logo, a search bar, and navigation links: ORDERS, BOX, SUPPORT, USER_NAME, and LOGOUT. The main content area is titled 'Order History' and shows two orders from 'Burger King' at 'Çankaya, Bilkent 1, Ankara' with a 4.5-star rating. The first order is for '2 Hamburger Menu' and the second is for '1 Chicken Burger Menu'. A modal window is open over the second order, allowing the user to leave a review. The modal includes fields for 'Restaurant Comment' (filled with 'Good'), 'Driver Comment' (filled with 'Delivery was fast.'), 'Restaurant Rating' (set to 6), 'Driver Rating' (set to 6), and 'Restaurant Response' (filled with 'Thank you!').

List Reviews for a Restaurant


Inputs: @restaurant-id

SQL Query:

```
SELECT Rev.review-id, Rev.delivery-rating, Rev.restaurant-rating, Rev.restaurant-comment,  
       Rev.delivery-comment, Rev.restaurant-response  
FROM HasReview HRev NATURAL JOIN Review Rev  
WHERE @restaurant-id = HasReview.restaurant-id
```

User Interface:

YemekKutusu Item USER_NAME



Burger King
Çankaya, Bilkent 1, Ankara
★★★★☆

Restaurant Reviews

Restaurant Rating: 6
Restaurant Comment: Good
Restaurant Response: Thank you

Restaurant Rating: 10
Restaurant Comment: Very Good

Back

Additional Requirements

Additionally, customers of the food delivery system can submit a support ticket for any purpose which is specified by the “SupportTicket”. Therefore, two new entity types are defined which are “SupportStaff” and “SupportTicket”. Customers will be able to submit their “SupportTicket”’s for any purpose. When the latter is submitted, the submitted tickets will be assigned to an available (free) “SupportStaff” and they will be responded by the “SupportStaff”. In case there is a violation of the rights of the customers is detected by “SupportStaff”’s, “SupportStaff”’s will be able to send a warning to the restaurant-owner. The system will take certain measures against the malicious behaviour of restaurants by inspecting their owners’ warning counts.

Send a Support Ticket

Inputs: @ticket-id, @date, @subject, @content

SQL Insert Statement:

INSERT INTO SupportTickets

VALUES(@ticket-id, @date, @subject, @content, NULL)

User Interface:

YemekKutusu

Item

Search

ORDERS

BOX

SUPPORT

USER_NAME


LOGOUT

Support Tickets

Support Ticket ID: 1223
Related to Restaurant Burg
Date: 12/01/2019
Subject: Food Poisoning
Content: The food I ordered
Response: Your complaint


Support Ticket

☒ Order
☐ Other

Burger King
Çankaya, Bilkent 1, Ankara
★★★★☆


Delivered
Date: 24/02/2021
Time: 15:05
Total: 44.00 TL

2 Hamburger Menu

Burger King
Çankaya, Bilkent 1, Ankara
★★★★☆

Delivered
Date: 15/01/2021
Time: 20:21
Total: 22.00 TL

1 Chicken Burger Menu

McDonald's
Çankaya, Kızılay, Ankara
★★★★☆

Delivered
Date: 03/01/2021
Time: 18:34

Subject:

Content:

CANCEL

SEND

New Ticket

Respond to Support Tickets

Inputs: @response, @staff-username

SQL Update Statement:

UPDATE SupportTickets

SET response = @response

WHERE EXISTS(SELECT *

FROM AssignedToTicket NATURAL JOIN SupportTicket

WHERE response = NULL AND username = @staff-username)

User Interface:

The screenshot displays the YemekKutusu application interface. At the top, a purple header bar contains the logo "YemekKutusu" on the left and the text "SUPPORT:USER_NAME" followed by a "LOGOUT" button on the right. The main content area is white. On the left, there is a section titled "Assigned Support" with a list of support tickets. One ticket is visible: "Support Ticket ID: 1223", "Related to Restaurant Burg", "Date: 12/01/2019", "Subject: Food Poisoning", "Content: The food I ordered", and "Response: Your complaint". A modal window is open in the center, titled "Support Ticket (ID: 1223)". It has a light purple background. At the top of the modal, there is a card for "Burger King (ID: 123345)" with the address "Çankaya, Bilkent 1, Ankara" and a rating of "★★★★☆". To the right of this card is a purple button labeled "WARN RESTAURANT". Below the card, there is a text box containing the ticket details: "Date: 12/01/2019", "Subject: Food Poisoning", and "Content: The food I ordered caused food poisoning." At the bottom of the modal, there is a label "Response:" followed by a text input field. At the very bottom of the modal, there are two purple buttons: "CANCEL" and "SEND".

List Support Tickets from Customers Account

Inputs: @username

SELECT *

FROM SubmitTicket NATURAL JOIN SupportTicket

WHERE username = @username

User Interface:

YemekKutusu Item ORDERS BOX SUPPORT USER_NAME LOGOUT

Support Tickets

New Ticket

Support Ticket ID: 1223
Related to Restaurant Burger King (Restaurant ID: 123345)
Date: 12/01/2019
Subject: Food Poisoning
Content: The food I ordered caused food poisoning.
Response: Your complaint has been reviewed and the restaurant has given the necessary warning.

List Tickets Assigned To Support Staff

Inputs: @stafft-username

```
SELECT *  
FROM AssignedToTicket NATURAL JOIN SupportTicket  
WHERE username = @staff-username
```

Send Warning

Inputs: @owner-username

SQL Update Statement:

```
UPDATE RestaurantOwner  
SET warning-count = warning-count+1  
WHERE username = @owner-username
```

User Interface:

YemekKutusuSUPPORT:USER_NAMELOGOUT

Assigned Support Tickets

Support Ticket ID: 1223
Related to Restaurant Burger King (Restaurant ID: 123345)
Date: 12/01/2019
Subject: Food Poisoning
Content: The food I ordered caused food poisoning.

