Bilkent University

Department of Computer Engineering

Final Report

CS353 DATABASE SYSTEMS

Onat Korkmaz
Muhammed Maruf Şatır
Ümit Çivi
Erdem Ege Eroğlu

January 4, 2022

http://onat.korkmaz.ug.bilkent.edu.tr/courses/cs353.html

Introduction **3**

Final E/R Diagram **4**

Final Database Tables **5**
- User 5
- Employee 5
- Branch 5
- Customer 5
- CashTransaction 5
- Credit Card Transaction 5
- Credit Card 6
- Transaction 6
- Discount 6
- Pays 6
- Submission 6
- Package 6
- Insurance 6
- InsuranceCompany 7
- Report 7
- Transfer 7
- Evaluates 7
- Courier 7
- Clerk 7
- Pick-up Assignment 8
- Delivery Assignment 8

Implementation **8**
- Tools and Languages: 8
- Problems Encountered: 8
- Contribution of Each Team Member: 9

Advanced Database Components **9**

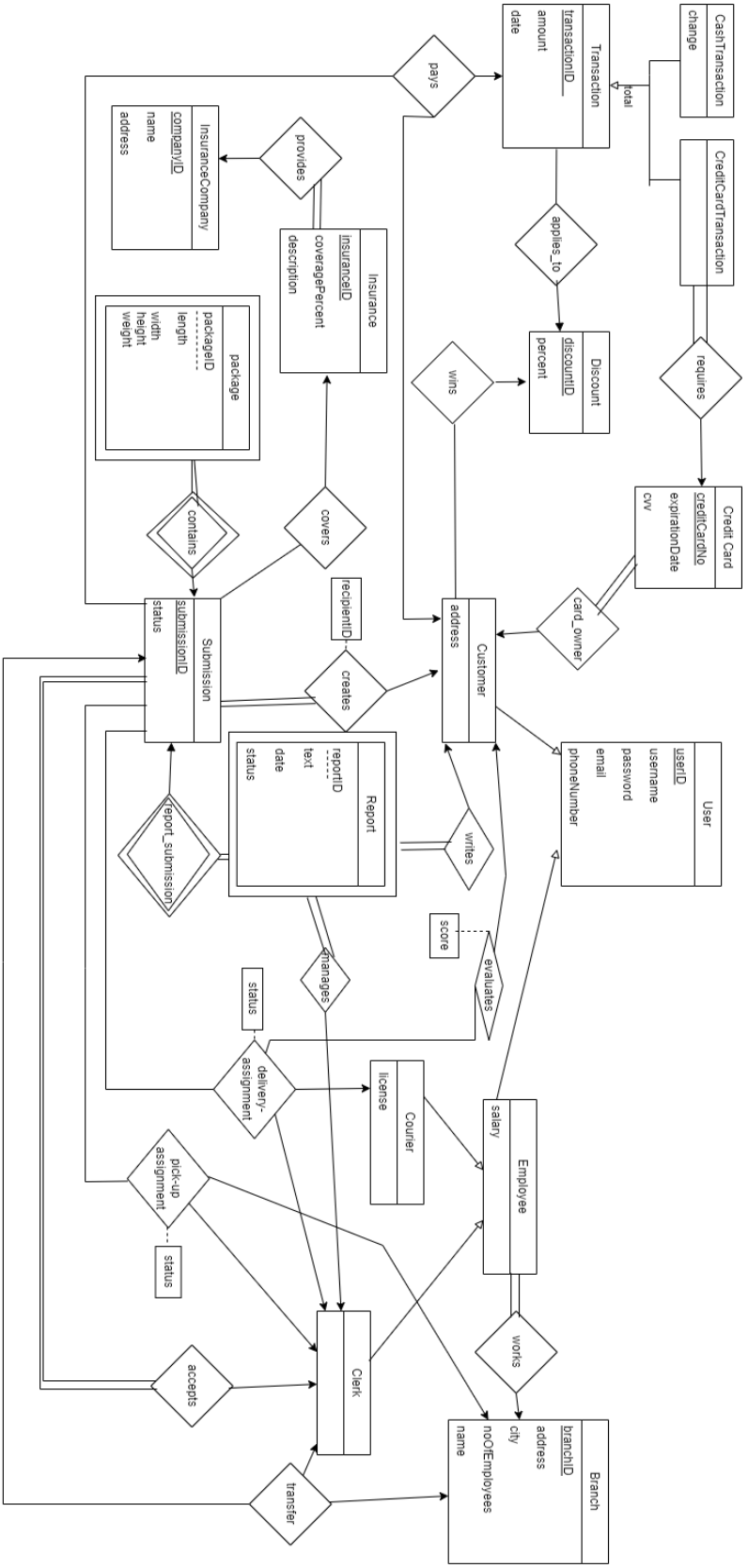
Secondary indices: **11**

User Manual **11**
- Build Instructions 11
- Employee manual 14
- Customer Manual 17

Introduction

Cargo application is a cargo management system which is a web-based project designed to allow customers and employees to use the functionalities of the cargo management system. There are two types of users which are customers and employees in the application. Customers can save their credit card and use it as a payment option for cargo payment. Customers can select the user which they want to send a cargo and select a procedure for it which is calling the courier or delivering the cargo by hand to the clerk. In addition, the courier can see the cargo history for both sent and delivered cargo and their status. Moreover, they can evaluate the courier and write a report about the process of submission and delivery. Employees of the program have the main functionalities of validating the cargo status, acceptance etc. Employees can assign a courier if the customer chooses their branch and calls a courier. They can accept the cargo by hand from customers and employees can accept the cargo if the cargo is transferred from other branches. Moreover, for the delivery to the destination location, employees can assign a courier for the delivery. Finally, they can see the evaluation from the customers and finalize the report. They can finalize the report as positive or negative.

Final E/R Diagram



Final Database Tables

User

user(userID, username, password, email, phone_number)

No foreign keys.

Employee

employee(userID, salary,branchID,username,password,email,phone_number)

userID is a foreign key to User.

branchID is a foreign key to Branch.

Branch

branch(branchID, name, address, city, noOfEmployees)

No foreign keys.

Customer

customer(userID, address, discountID, username, password, email, phone_number)

discountID is a foreign key to discount.

userID is a foreign key to user.

CashTransaction

cashTransaction(transactionID, change)

transactionID is a foreign key to transaction.

Credit Card Transaction

creditCardTransaction(transactionID, creditCardNo)

transactionID is a foreign key to transaction.

creditCardNo is a foreign key to creditCard.

Credit Card

creditCard(creditCardNo, expirationDate,cvv,customerID)

customerID is a foreign key to customer.

Transaction

transaction(transactionID, amount, date, discountID)

discountID is a foreign key to discount.

Discount

discount(discountID,percent)

No foreign keys

Pays

pays(submissionID,transactionID, customerID)

transactionID is a foreign key to transaction.

customerID is a foreign key to customer.

submissionID is a foreign key to submission.

Submission

Submission(submissionID, senderID, recipientID, insuranceID, clerkID,status)

senderID, recipinetID are foreign keys to customer.

insuranceID is a foreign key to insurance

clerkID is a foreign key to clerk.

Package

package(packageID, weight, width, length, height, submissionID)

submissionID is a foreign key to submission.

Insurance

insurance(insuranceID, coveragePercent, description, companyID)

companyID is a foreign key to InsuranceCompany

InsuranceCompany

insuranceCompany(companyID, name, address)

No foreign keys.

Report

report(reportID, submissionID, customerID, clerkID, text, date, status)

submissionID is a foreign key to submission.

customerID is a foreign key to customer.

clerkID is a foreign key to clerk.

Transfer

transfer(submissionID, clerkID, branchID)

submissionID is a foreign key to submission.

branchID is a foreign key to Branch.

clerkID is a foreign key to clerk.

Evaluates

evaluates(customerID, submissionID, courierID, clerkID, score)

customerID is a foreign key to customer.

submissionID is a foreign key to submission.

clerkID is a foreign key to clerk.

courierID is a foreign key to courier.

Courier

courier(userID, license)

userID is a foreign key to employee.

Clerk

clerk(userID)

userID is a foreign key to employee.

Pick-up Assignment

pickupassignment(submissionID, branchID, courierID,status)

submissionID is a foreign key to submission.

branchID is a foreign key to Branch.

courierID is a foreign key to courier.

Delivery Assignment

deliveryassignment(clerkID, courierID, submissionID, status)

submissionID is a foreign key to submission.

clerkID is a foreign key to clerk.

courierID is a foreign key to courier.

Implementation

Tools and Languages:

In implementation phase we used the following tools and programming languages;
For Database: phpMyAdmin to manage the database components
For Back-end: PHP MySQLi library
For Front-end: HTML, CSS, Bootstrap with Javascript alerts

Problems Encountered:

During implementation, we realized that in order to create a responsive user interface, we needed a client-side language like Javascript. Since PHP is a server-side programming language we could not achieve what we wanted at first. In order to tackle this problem we used small Javascript codes in our application so that the pages become more responsive and user-friendly.

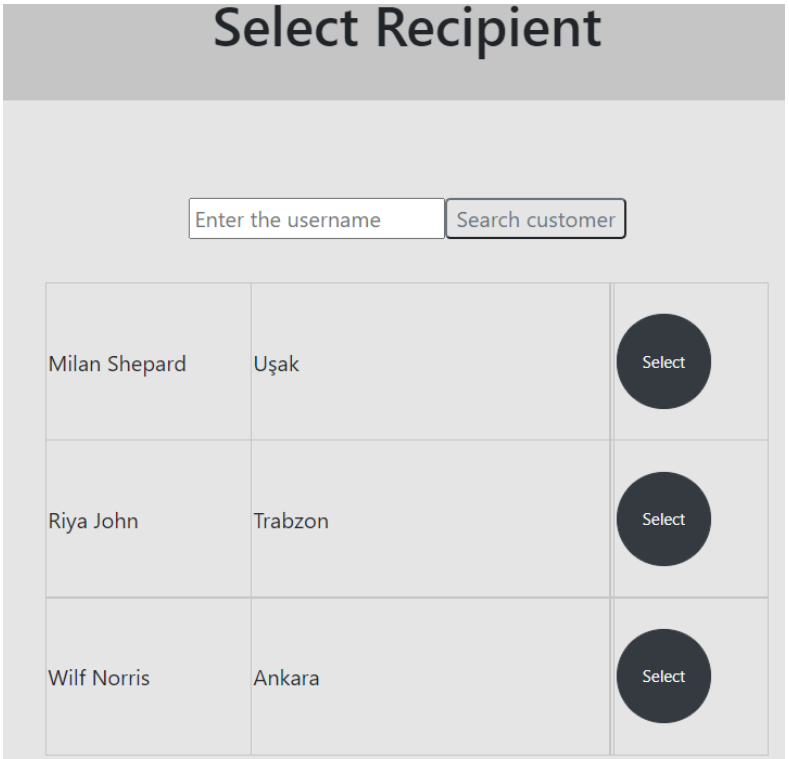
Another problem was the integration of front-end and back-end. The group members in the front-end part did not have adequate information about the back-end and vice versa. This led to problems during fetching queries and displaying data. In order to tackle this problem, each part increased their knowledge regarding the other part of the project. Then, we were able to integrate the front-end and back-end.

Contribution of Each Team Member:

Muhammed Maruf Şatır: Implementation of back-end. Created the database tables with Ümit Çivi. We shared with Ümit the implementation of the back end of the program.
Ümit Çivi: Implementation of part of the back-end. Created the database tables together with Maruf. Integration of the front-end and back-end.
Erdem Ege Eroğlu: Implementation of the front-end and worked with Onat Korkmaz. Customer side made by Erdem Ege Eroğlu
Onat Korkmaz: Wrote the initial queries of the pages and created the front-end part of the project with Ege Eroğlu. Worked on the employee functionalities on the front-end.

Advanced Database Components

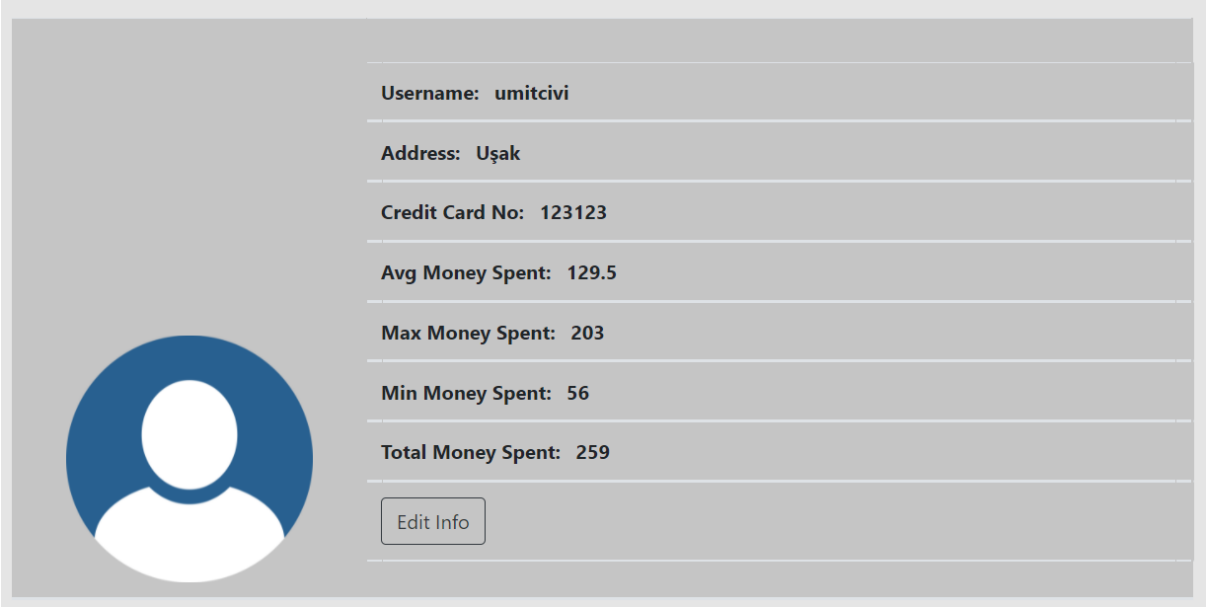
Triggers: We used a trigger while inserting the package description, i.e. weight,length etc. The trigger checks each of the dimensions whether or not they are valid. If they are not valid the default value of zero is used.
Views: We created a view for customers to see recipient name and address without showing their private data. Example output for the view:
CREATE VIEW recipient_view AS select username, address from customer



REPORT: The report of the application includes the information of the payments of the customer. The minimum, maximum, sum and average of the payments of the cargo is outputted to the screen.

```
DELIMITER $$
CREATE PROCEDURE get_report_procedure_customer(IN profileID INT)
BEGIN
select
c.userID,
c.username,
sum(m.amount),
MIN(m.amount),
MAX(m.amount),
AVG(m.amount)
from transaction m
join pays o on m.transactionID = o.transactionID
join customer c on c.userID = o.customerID
WHERE c.userID = profileID
group by c.userID, c.username;
END $$

DELIMITER ;
```



Stored Procedure:
We have used the stored procedure for storing the report written in the report section. We had use the parameter for taking the userID from the current customer. The calling procedure is:
$report = mysqli_query($conn,"CALL get_report_procedure_customer('$userID')");

Constraints:
We did not use constraints. However, we write the implementation as not to be obliged to use constraints. For instance, we take the courier assignments dynamic and the employee just adds the valid courier in their branch.

Secondary indices:

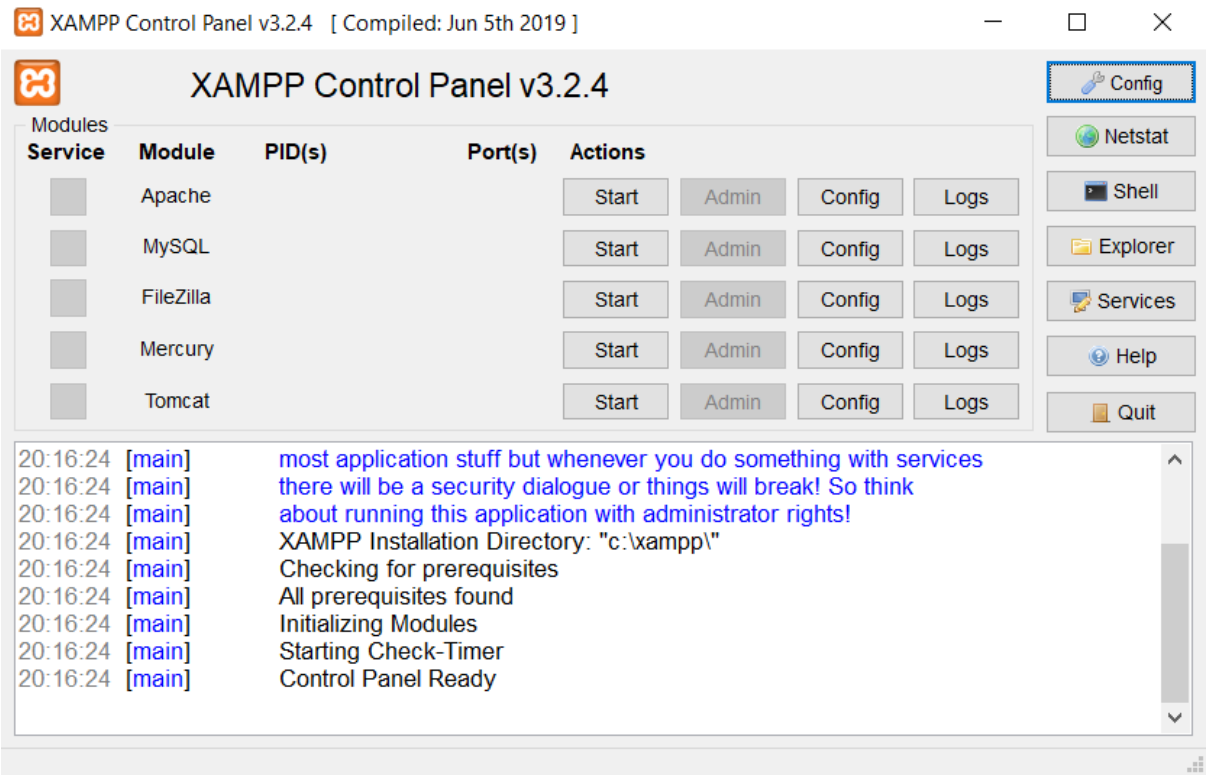
We have used secondary indices for the username of the customer table. We have used it for searching for a recipient.

User Manual

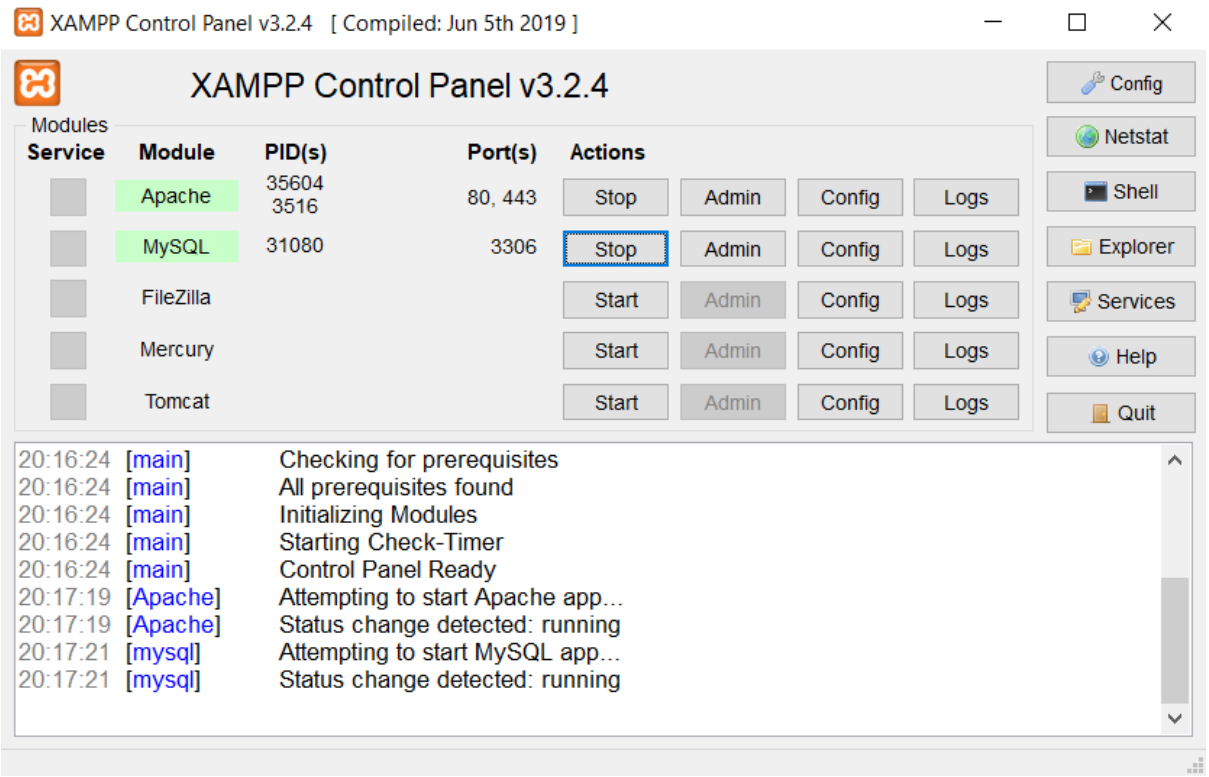
Build Instructions

- Firstly, XAMPP should be downloaded in order to run the program.

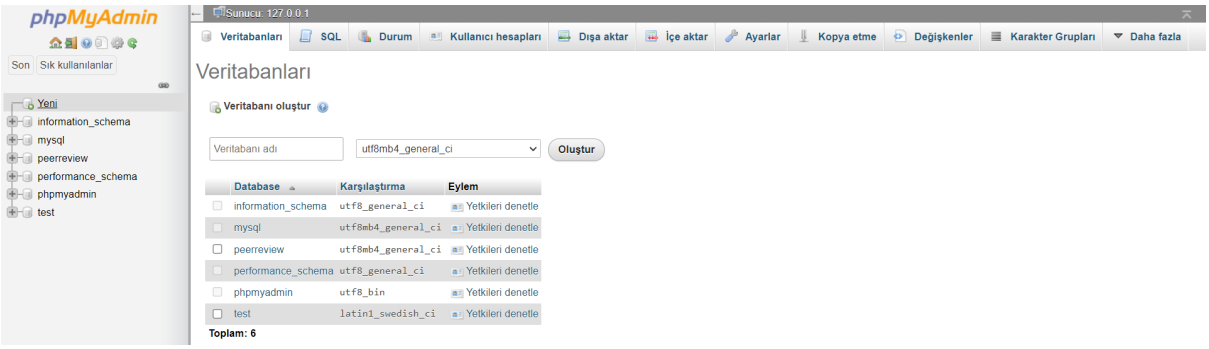
- After it is installed, XAMPP Control Panel should be opened.



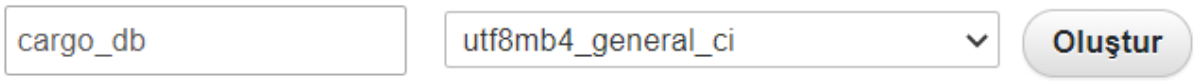
- As seen above, the Start buttons of the rows Apache and MySQL should be clicked to start the modules. The control panel should look like below after the buttons are clicked:



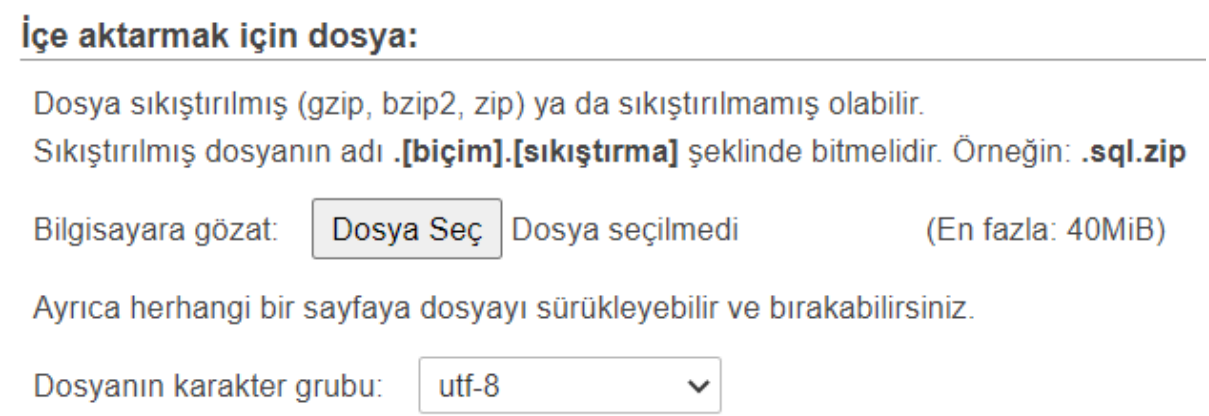
- Once this part is completed, the URL http://localhost/phpmyadmin/ should be searched in Google Chrome.



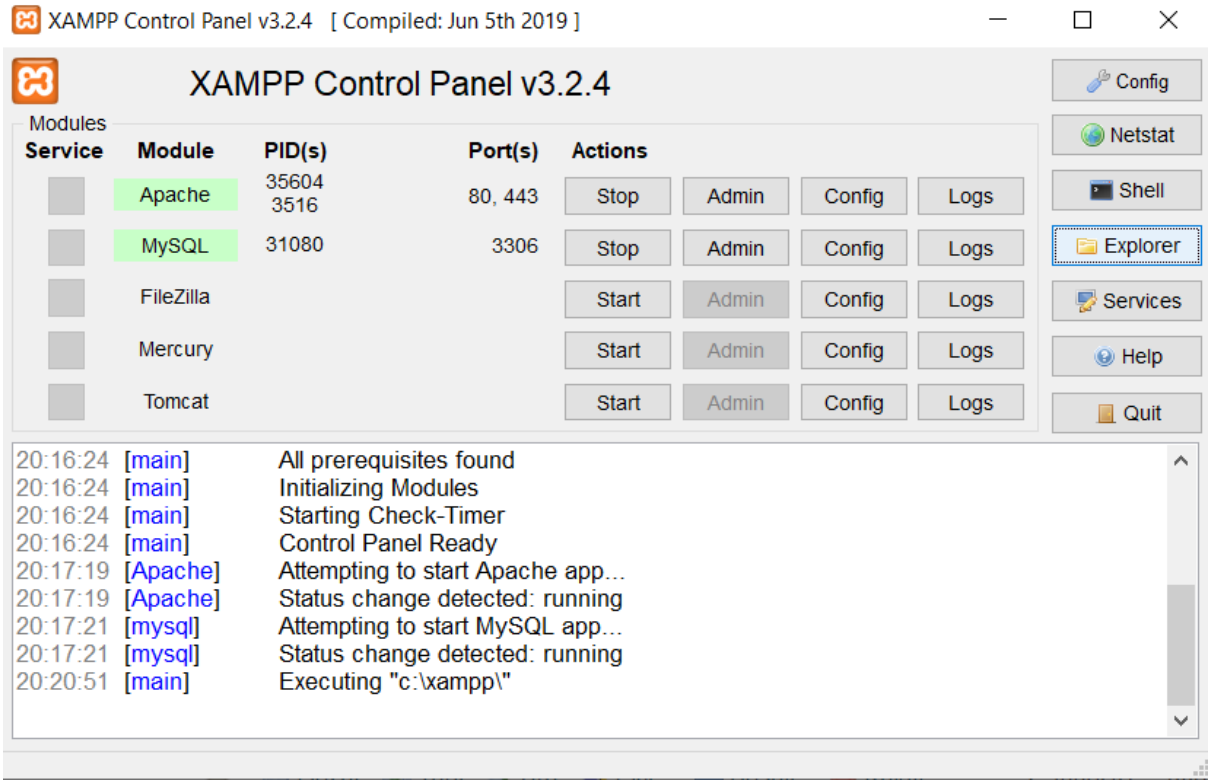
- Press “New” on the left bar to create a new database.



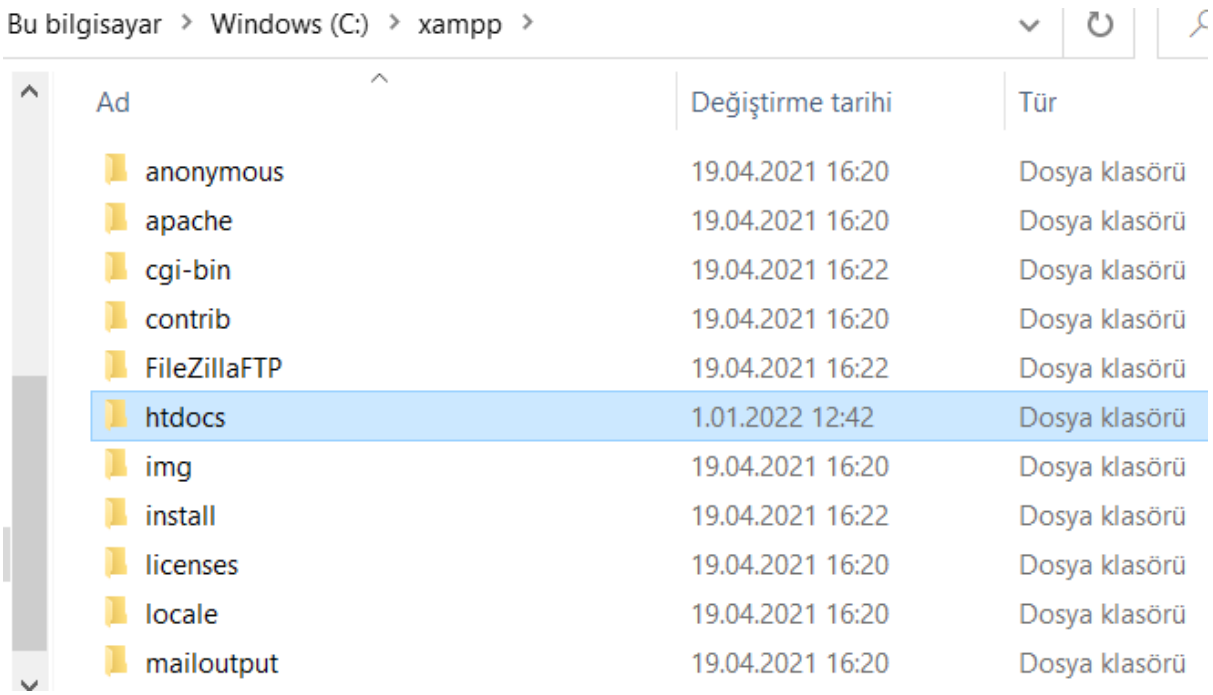
- On the field “Database name”, write “cargo_db” and click the Create button. Then click the Import button on the top bar.



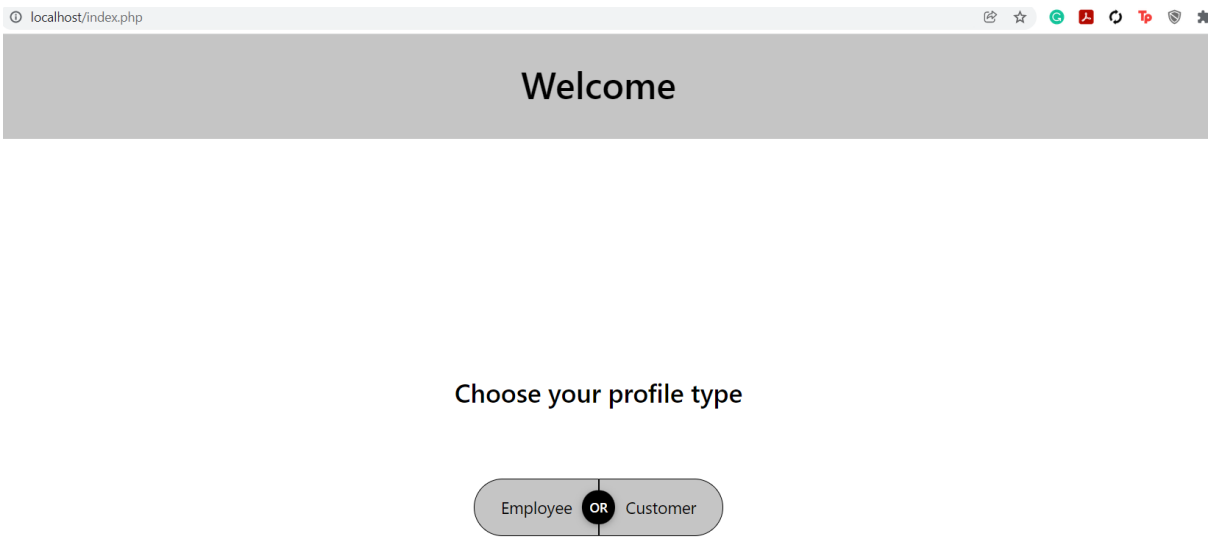
- Select the .sql file from the folder that you downloaded. After you choose the file, hit the Go button on the bottom right corner.



- coming back to the XAMPP control panel, click Explorer on the right bar. It will open the xampp folder in your computer.

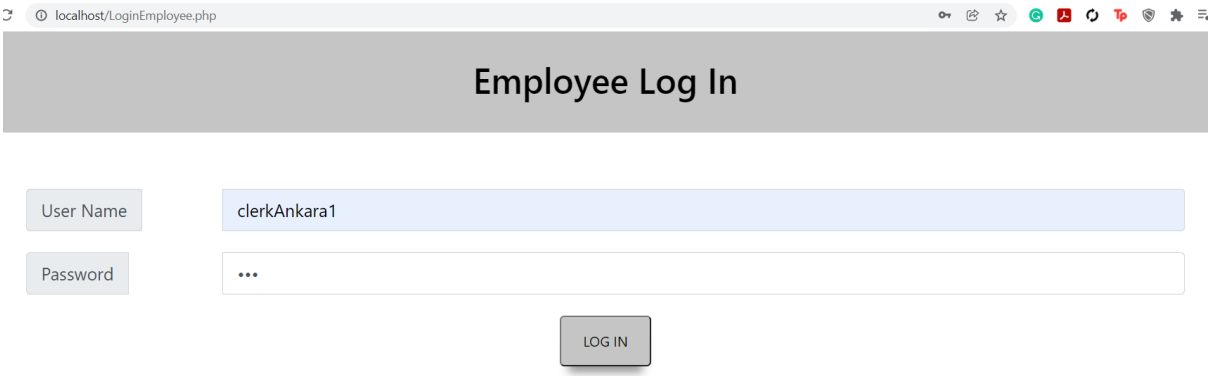


- If you have a folder named htdocs inside the xampp folder, change it with the new one. If there is no such folder as htdocs, create a new one and transfer the files in the htdocs folder that you downloaded. Once you are done, you are ready to launch the application. Go to URL http://localhost/index.php to start the website.

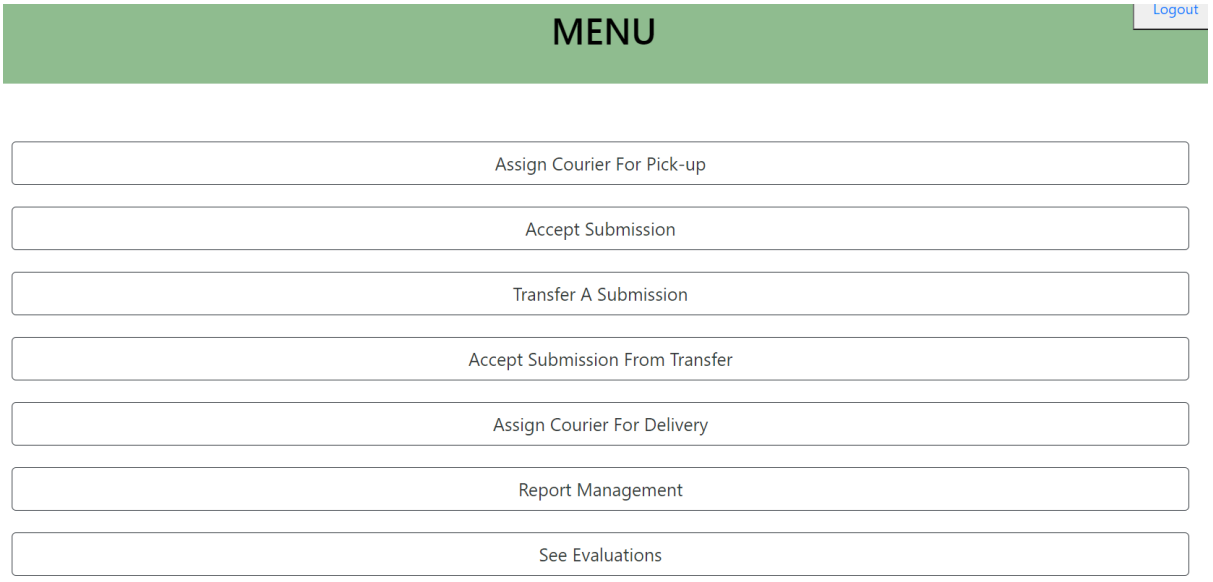


- Here you can see the welcome page. You can choose your role to login to the system.

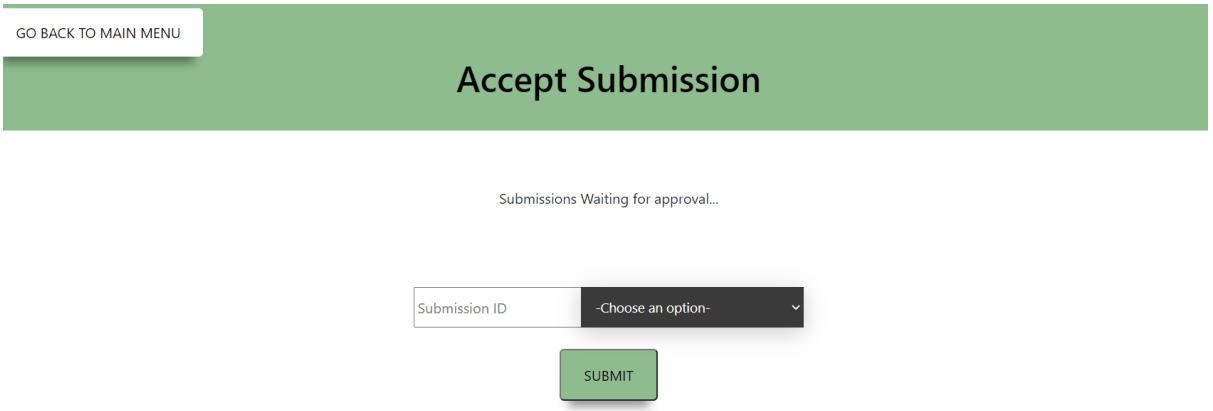
Employee manual



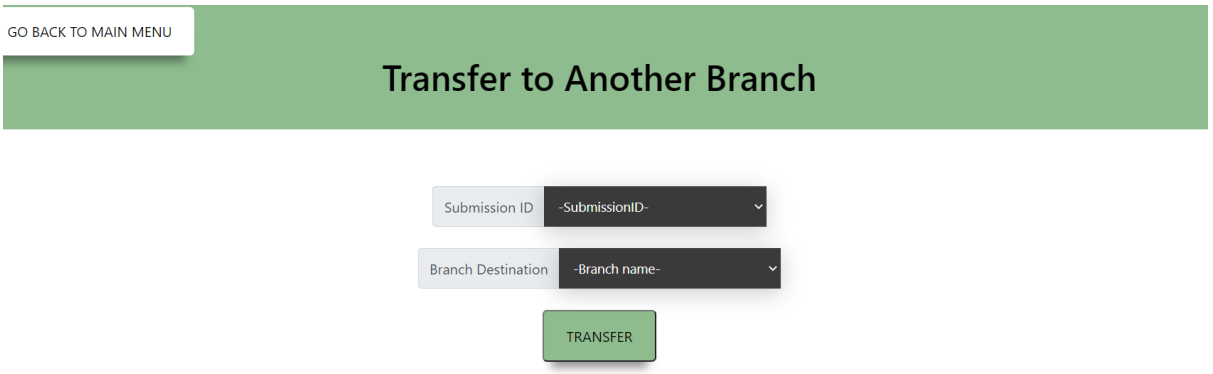
- After you choose your role as Employee, a specialized login page should pop up. You can enter your user name and password to login.



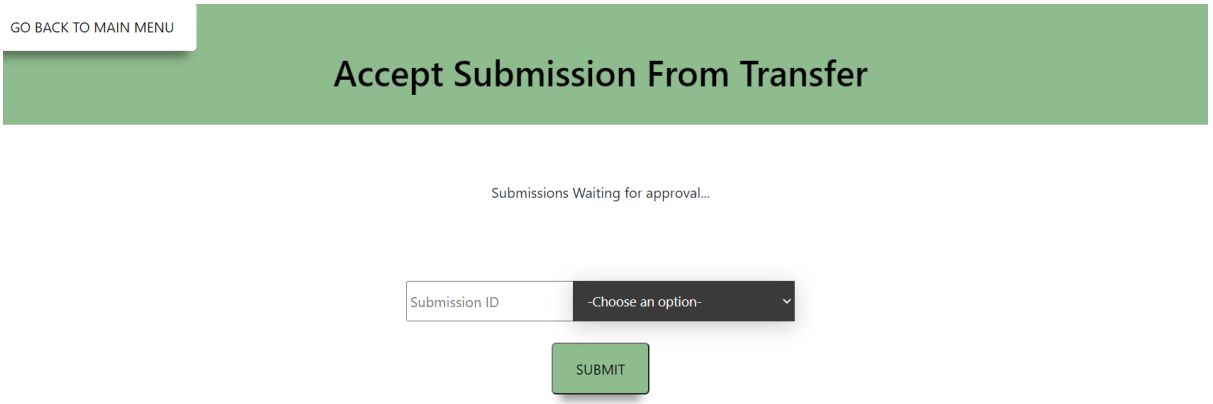
- The employee main menu page shows all the functions that the employee is supposed to do. You can navigate to the pages using the buttons indicating the tasks.



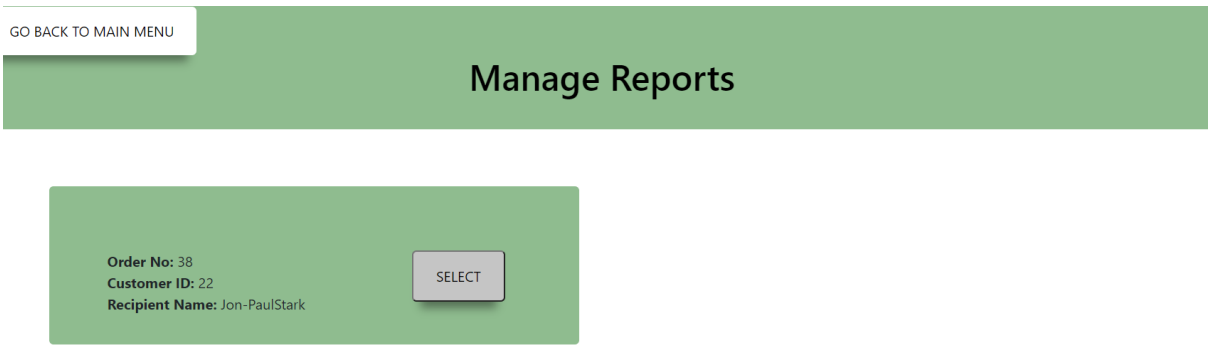
- The “accept submission” page is for employees. They can accept or decline the submission with the specified submission id.



- The transfer to another branch page is for employees. The employee can choose the submission's id. Then, choose a destination branch to transfer.



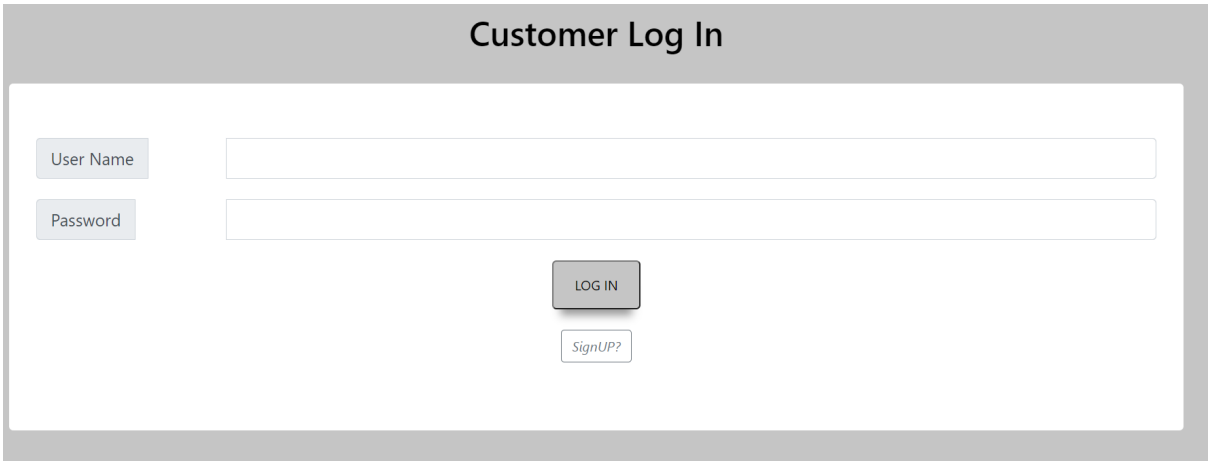
- “The accept submission from transfer” page is for employees. They can accept or send back the submission to the its old branch.



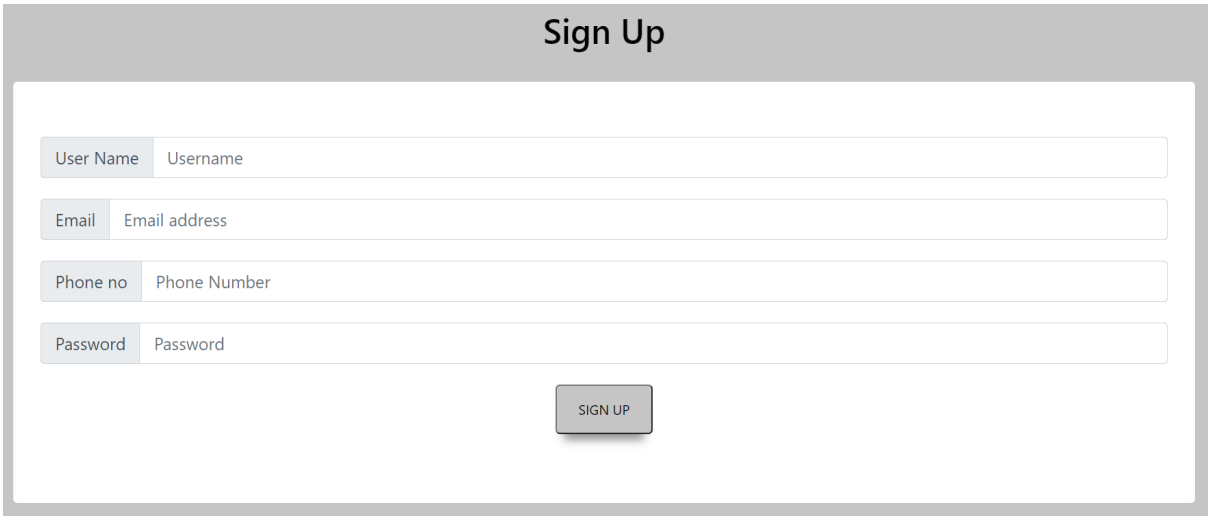
- The manager reports page is for employees. Employees can manage completed reports by customers. Each report has a select button to be managed.

- The evaluation page is for employees. They are able to see evaluations about the complicated delivery.

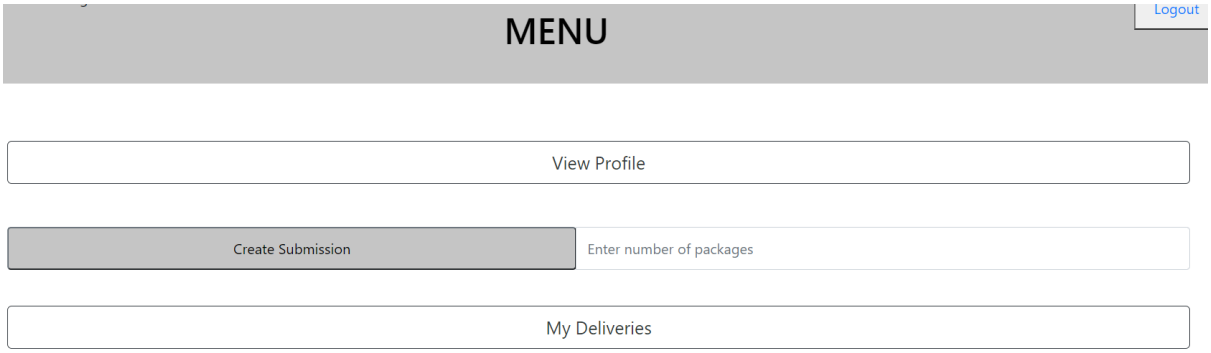
Customer Manual



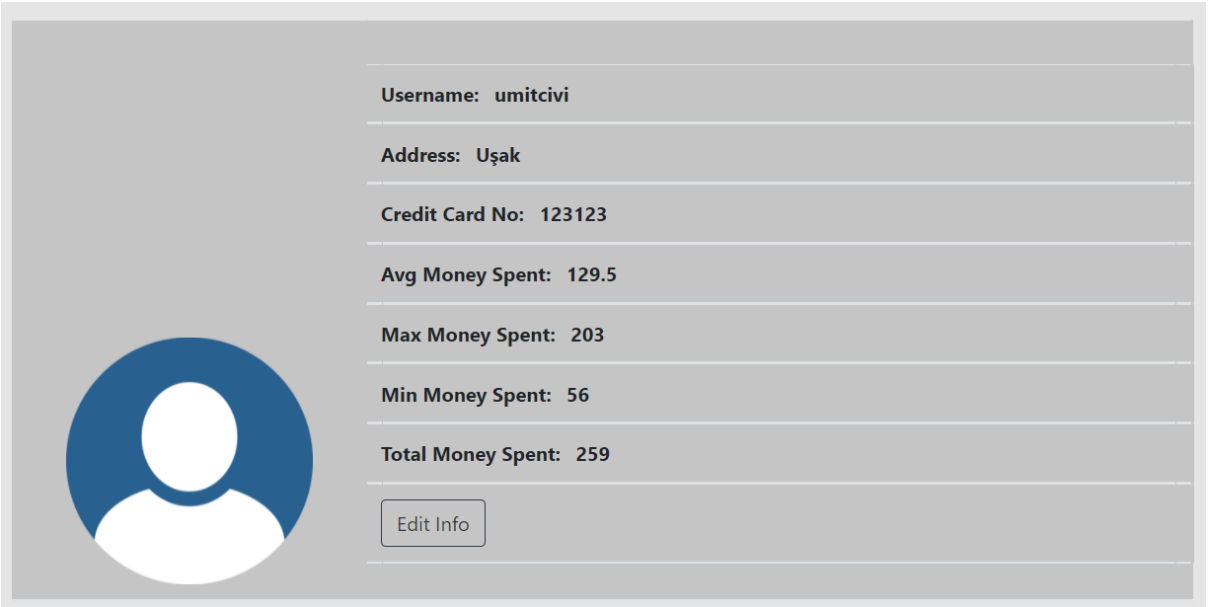
- The customer login page is obviously for customers. They can start using the application after they write down his/her username and password.



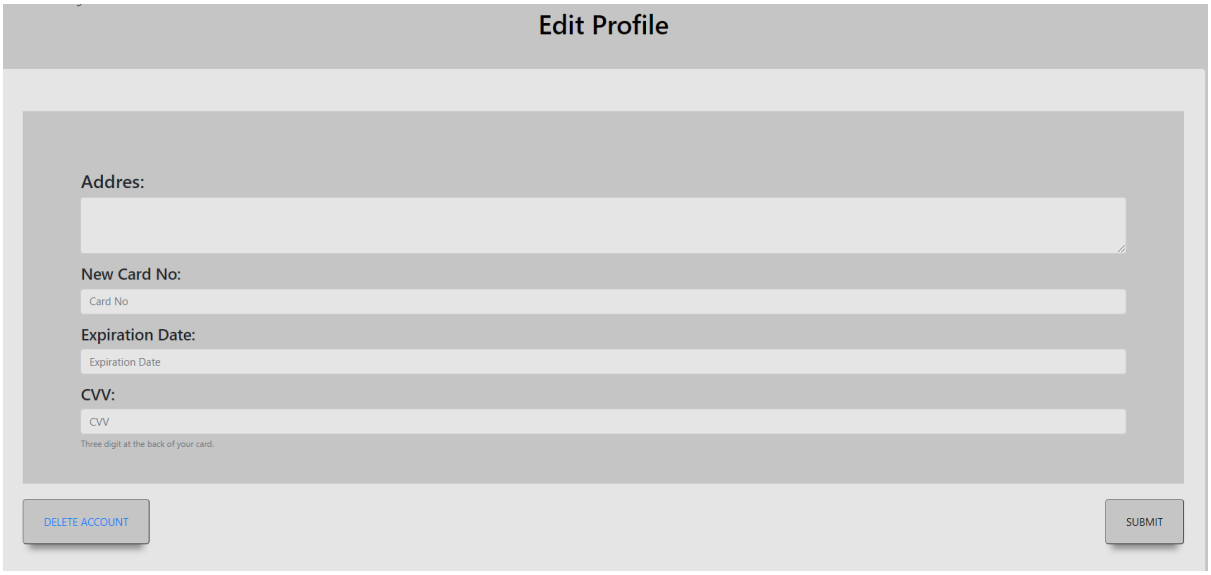
- The sign up page is for customers. They can be assigned to the system with personal information.



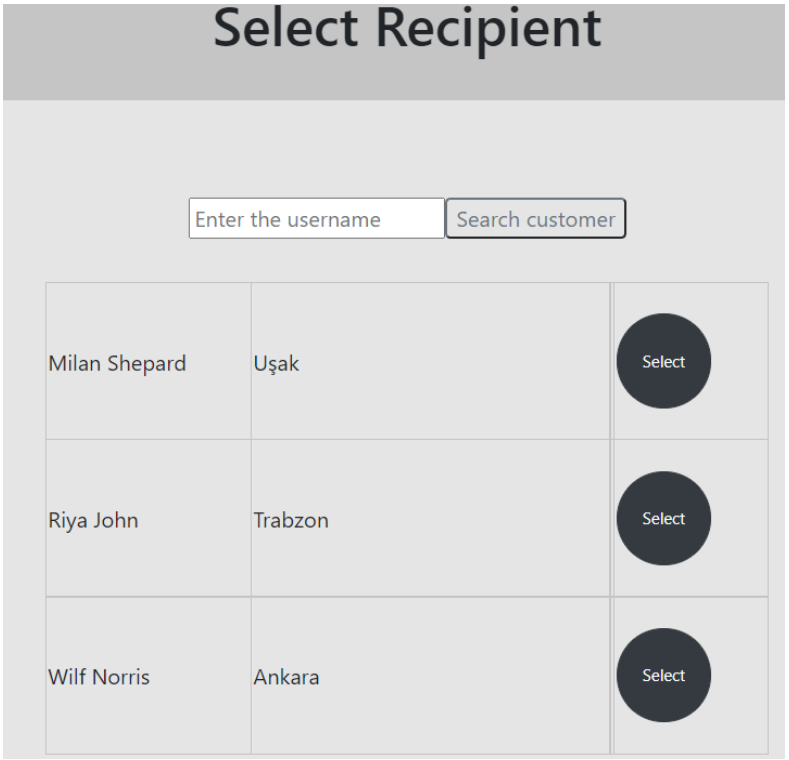
- This is a menu page for customers.They can navigate via these three buttons. Those buttons can satisfy all the functionalities of customers.



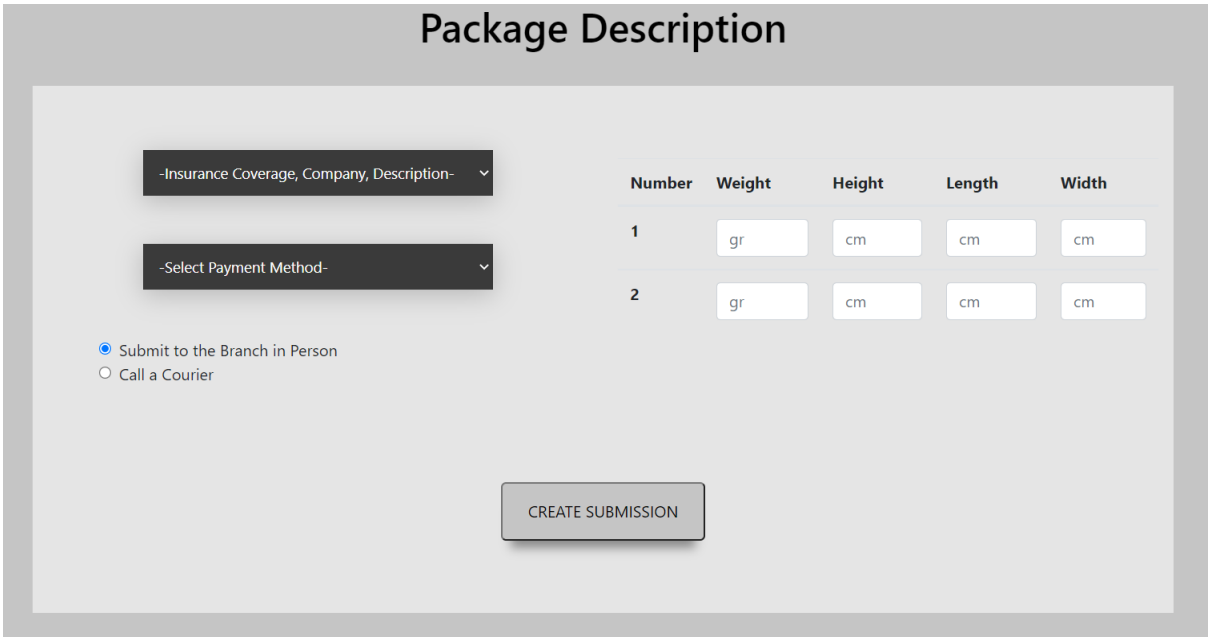
- “View profile” page is for customers. They can see all information about them self that given to the system and they can edit those.



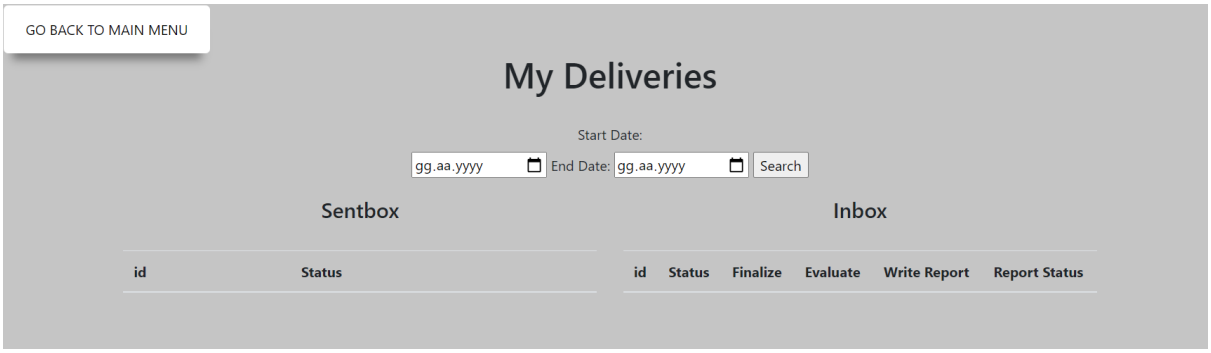
- This is “edit profile” page for the customers. They can change their address and credit card information.



- The “select recipient” page is for customers. They are navigated to this page after they want to send a cargo to select a recipient. List of recipients can be seeable after name search.



- “Package description” page is for customers. They need to specify their package’s size information. Also, if they want to call a courier they need to ask to the system on this page.



- “My deliveries” page is for customers. They can see all packages that they sent or they got in sentbox and inbox successively. They can search according to date. Finally, they are able to consider the status of submissions.