

# PROJECT2

## PROPOSAL FOR THE CLOTHINE INFORMATION SYSTEM DEVELOPMENT PROJECT

### [Abstract](#)

This file shows the second part of the development of CLOTHINE. It shows the Creation of databases and Web applications.

OMPHEMETSE SENNA  
omphesenna@gmail.com

## **TABLE OF CONTENT**

1. PROBLEM STATEMENT
2. DEFINITION OF SCOPE
3. GOALS OF THE PROJECTS
4. MISSION
5. BUSINESS RULES
6. USE-CASE DIAGRAM
7. CONCEPTUAL DATA MODEL
8. LOGICAL DATA MODEL
9. PHYSICAL DATA MODEL
10. DATABASE OBJECTS (CREATING A DATABASE)
11. DATABASE OBJECTS (CREATING A TABLES)
12. DATABASE OBJECTS (ADDITIONAL DATA ELEMENTS)
13. OUTPUTS AND INPUT DESIGN, AND PROTOTYPING

## **1. Problem Statement:**

There have been several problems in the company that affects its performance and reputation. The problems that the company has are as follows:

- They don't have effective, flexible, and versatile management in the company.
- There is no effective communication between the employees and the clients.
- Clients can't exactly specify the type of services they kindly want.
- record book can be misplaced.
- information about the client can be written wrongly.
- Their couriers sometimes deliver the client's item to the wrong location.

Therefore, the cause of above problems is caused by the company's lack of management and the reason for not having a system that will meet them halfway by making the clients do the requests online.

## **2. Definition of Scope:**

This project will develop and deliver a new web application for the Clothine company. The web application will enable the user to make bookings or request services at any time they wish. The web application will provide everything that the user may need. For instance, if the user wants to get in contact with the company, there will be sites in the web app that will lead the client to the company.

The web application will have different sectors that the user will go through when requesting the service.

## **3. Goals:**

This project will develop and implement the system that will reach all the goals that this company have given. The following are the goals that this project will meet.

The system will:

- Improve communication between the user and the employees.
- Keep all the records of the client.
- Give view to the client all the services that the company have.
- Help the user option to whether make a house-call which is opting the employees of the company to come and make the services at the client's house or whether services will be made at the company's workshop.
- Allow the user to choose the method of payment.
- Improve business processes by automating the request of the business. It must make all the requests happen over the internet.
- Improve business communication by providing a consistent graphical user interface to the user.

## **4. Mission:**

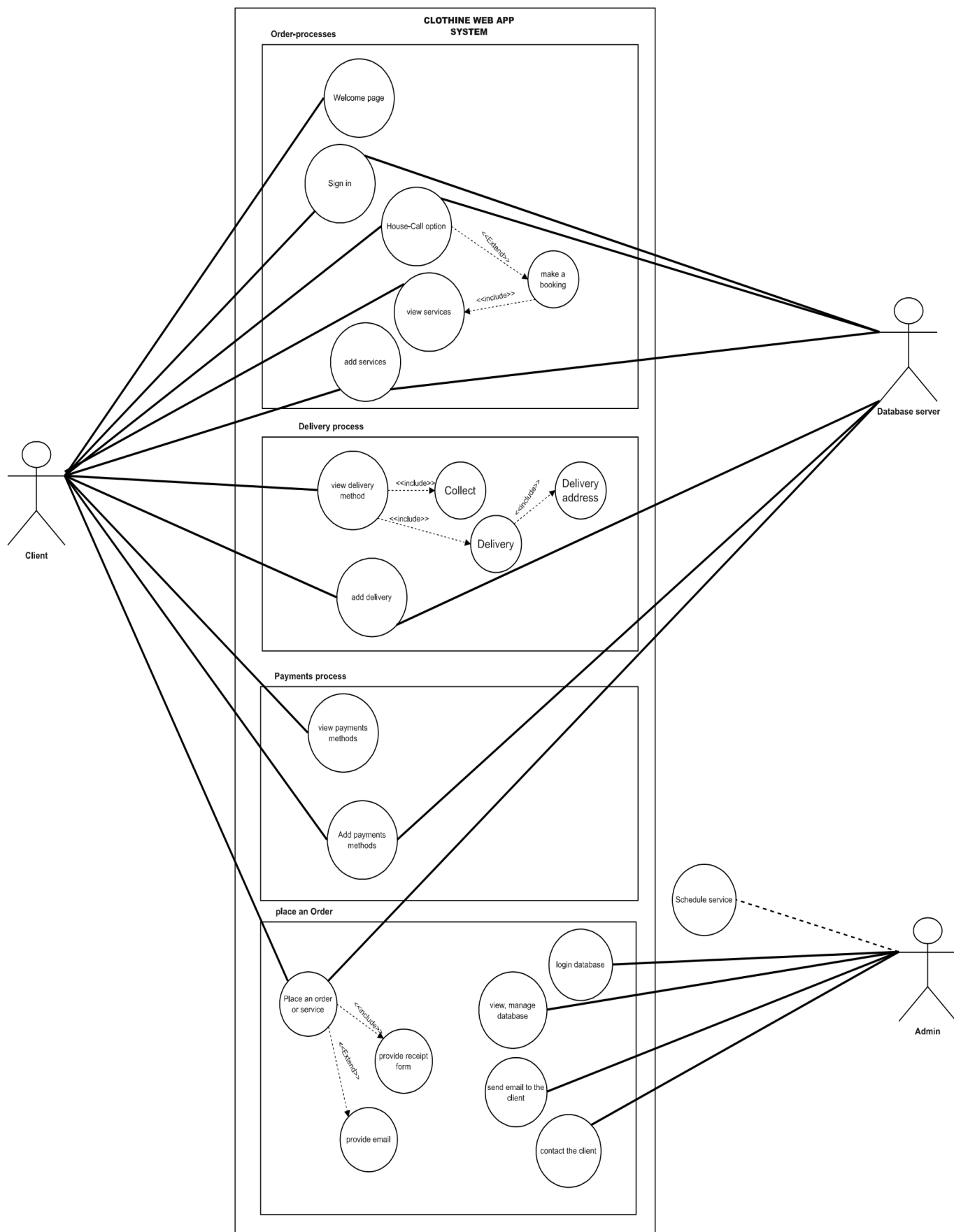
- Clothine Laundry Services is still new and to improve it, we need to create a web application that functions and that will meet the clients halfway.
- Provide a web application with different costs of services whereby clients will be able to choose or identify the kind of services they will need and to pay for those services.
- Keep the information of the client safe and protected always.

## **5. Business rules:**

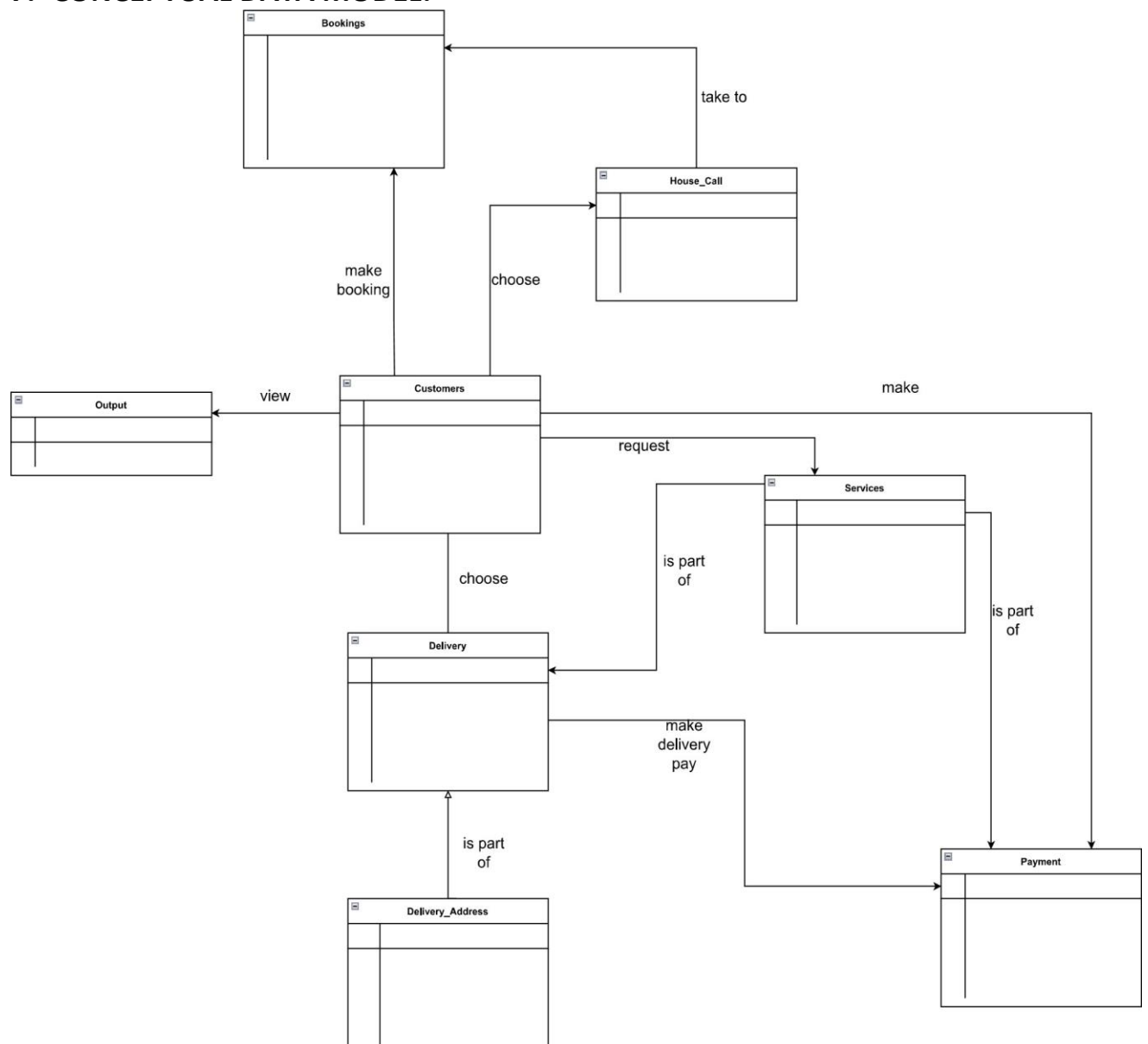
The following are the business rules that will be adhered to when the systems and the database are being designed:

- Customer can make zero or make a booking.
- A booking can be made by a customer or no customer.
- Customer can choose to make a house-call or not.
- House-call can be made by a customer or no customer.
- Customer can choose zero or one delivery.
- Delivery can be chosen by zero or one customer.
- Customer can request zero or many services.
- Services can be requested by one and only one customer.
- Customer can make zero or many payments.
- Payments can be made by one and only one customer.
- Customer can view one and only one output.
- Output can be viewed by one and only one customer.

## **6. USE-CASE DIAGRAM:**

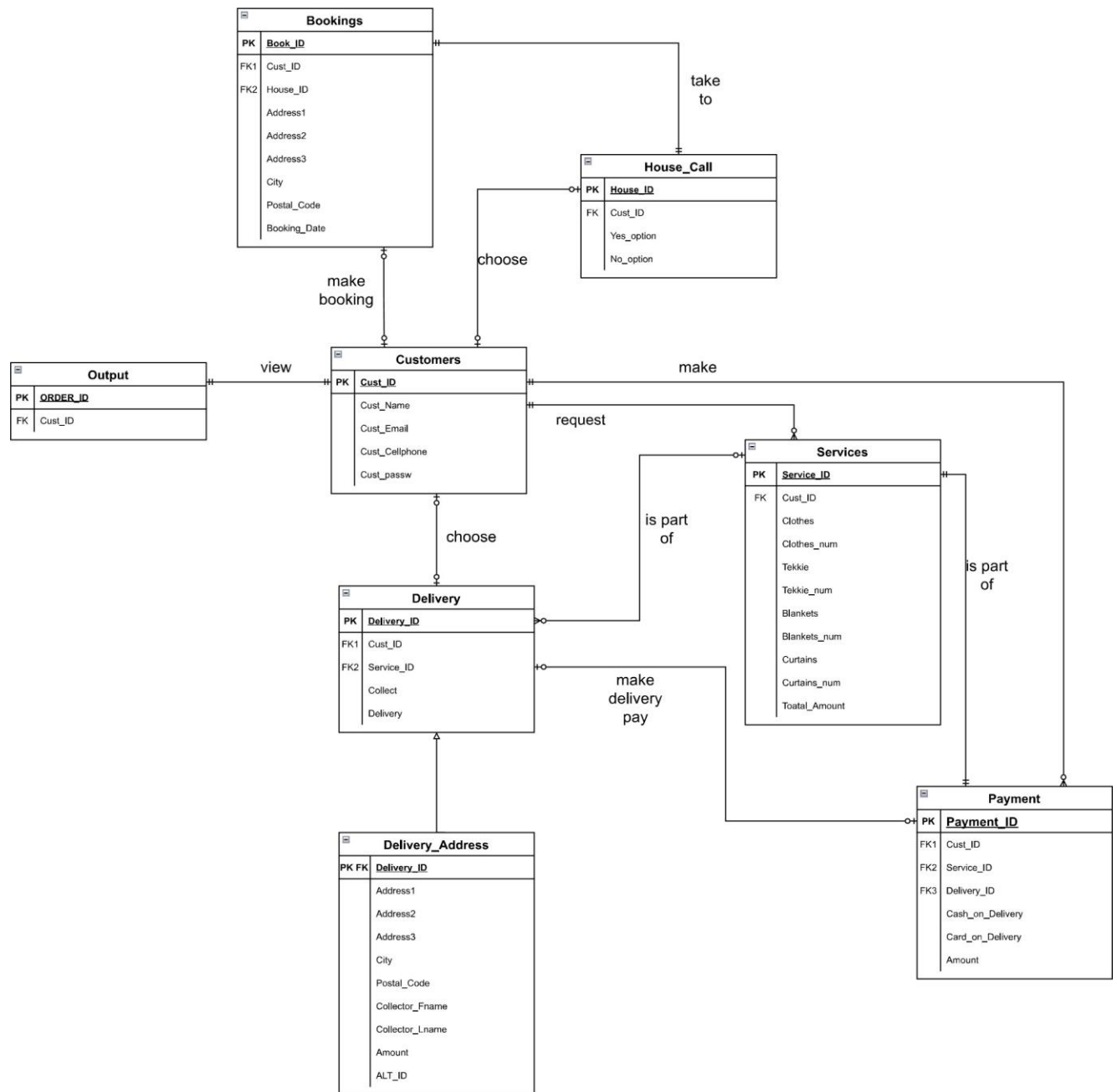


## 7. CONCEPTUAL DATA MODEL:



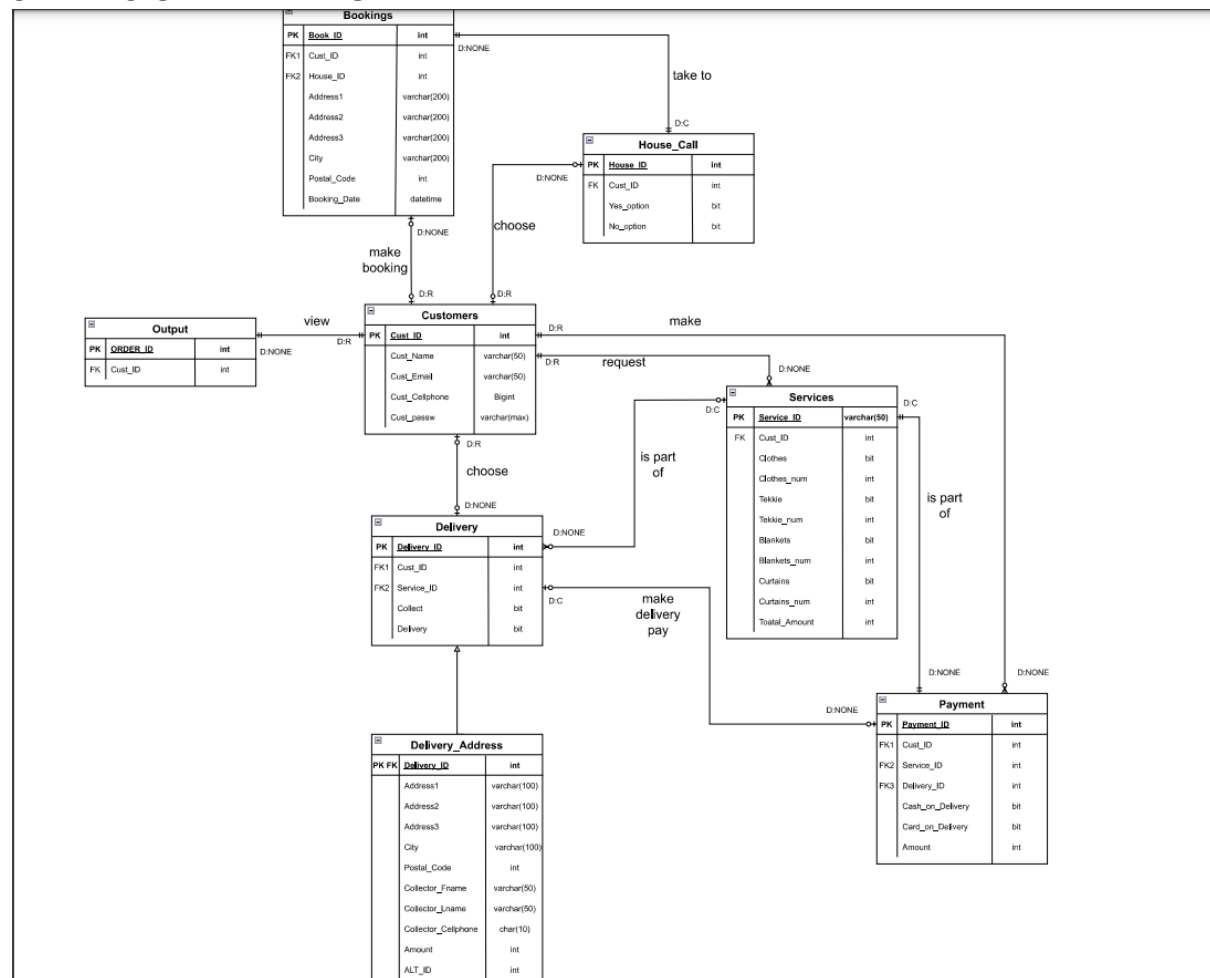
The above diagram visualizes the ideal design and some basics about how the database will look based on the business rules provided. The entity tables are linked to one another to show the flow and their relationships.

## 8. LOGICAL DATA MODEL:



The above diagram shows the Entity tables containing their attributes. It also shows the relationships between the tables.

## 9. PHYSICAL DATA MODEL:



The above diagram shows more details about the design. It shows the data types of each attribute and the referential integrity of the entity tables. This diagram will be used to design and implement the SQL database.

## 10. Database Objects (Creating a Database):

### Creating Laundry database:

- The laundry database will store the client's information from the web application after the client has an interface with the web application. This database has eight (8) attributes that will be created.

```
--create database--
Create database Laundry;
```



## 11.Database Objects (Creating tables):

- In this section we will be creating the database using the Microsoft SQL Server. The Entity tables will be created together with their relationships and relationships.

### 11.1 Customer Table:

A customer table is used to capture the details of new users or customers. It consists of 5 fields and one of those is Cust\_ID which is a primary key that will uniquely identify the row in the table. All the fields are restricted to not being null and in this table, there are no foreign keys. Cust\_Name, which is the customer's name, will only accept 50 characters from the user other than that it will not accept it. This table will be also for the client to sign on the website. The fields that will be used are Cust\_Email and Cust\_pass.

```
--Customer Entity--
Create Table Customers(
    Cust_ID int identity(1,1) primary key not null,
    Cust_Name varchar(50) not null,
    Cust_Email varchar(max) not null,
    Cust_Cellphone BIGINT not null,
    Cust_passw varchar(max) not null
);
```

### 11.2 HouseCall Table and Bookings:

A houseCall table will be created to store the options of the client that will determine whether the client wants to do the booking or not. This table will use the foreign key Cust\_ID from the customer table to link the two tables. It also has the primary key called House\_ID which will be used to uniquely identifies the row. The other two attributes are bits data type and can allow null as a value.

When the Yes\_option attribute is filled with true value the No\_option will have to be null. This will lead us to the Booking table. But when the No\_option attribute has the value true, it won't take us to the Booking table.

The Bookings table has its primary key which will uniquely identify the row of the table, and it also has referential keys that will also build a relationship among the tables. All the fields cannot be null except the Address3 attribute which is optional to the user.

```
--HouseCall Entity--
create table HouseCall (
    House_ID int identity(1,1) primary key not null,
    Cust_ID int not null
    foreign key references Customers(Cust_ID),
    Yes_option bit null,
    No_option bit null
);
--Bookings Entity--
create table Bookings (
    Book_ID int identity(1,1) primary key not null,
    Cust_ID int not null
    foreign key references Customers(Cust_ID),
    House_ID int not null
    foreign key references HouseCall(House_ID),
    Address1 varchar(200) not null,
    Address2 varchar(200) not null,
    Address3 varchar(200) null,
    City varchar(200) not null,
    Postal_Code int not null
);
```

### 11.3 Services Table:

This table is created to save the services option made by the client and the quantities of that services chosen. This table has its identifier which is service\_ID that will uniquely identify the row and it also has Cust\_ID from the Customer table that is used to link up the tables. This foreign key together with the primary key cannot go null whilst other can go null because the client can choose not to request all the services that the organization have.

```
--Services Entity--
create table Services (
    Service_ID varchar(50) primary key not null,
    Cust_ID int not null
    foreign key references Customers(Cust_ID),
    Clothes bit,
    Clothes_num int,
    Tekkie bit,
    Tekkie_num int,
    Blankets bit,
    Blankets_num int,
    Curtains bit,
    Curtains_num int,
    Total_Amount int
);
```

### 11.4 Delivery and Delivery\_Address Table:

The Delivery table is a parent of the Delivery\_Address table. The delivery table will pass the primary key which is Delivery\_ID to the child table and the child table will use that key as both primary and foreign key. The Delivery\_Address, which is the child, will use that primary key to uniquely identifies rows.

The Delivery table will also contain two foreign keys from the Customer table and Services table to make a relationship. It will have two bits and one of the bits will be a key to open the child, that is why they are left null. The attribute Delivery will be the key to the child table. If it has the value True, this will open the Delivery\_Address table.

```

--Delivery Entity--
create table Delivery (
    Delivery_ID int identity(1,1) primary key not null,
    Cust_ID int not null
    foreign key references Customers(Cust_ID),
    Service_ID varchar(50) not null
    foreign key references Services(Service_ID),
    Collect bit null,
    Delivery bit null
);
--Delivery_Address Entity--
Create table Delivery_Address(
    Delivery_ID int not null,
    CONSTRAINT PK_ID PRIMARY KEY (Delivery_ID),
    CONSTRAINT FK_ID FOREIGN KEY (Delivery_ID)
    REFERENCES Delivery(Delivery_ID),
    Address1 varchar(100) not null,
    Address2 varchar(100) not null,
    Address3 varchar(100),
    City varchar not null,
    Postal_Code int not null,
    Collector_Fname varchar(50) not null,
    Collector_Lname varchar(50) not null,
    Collector_Cellphone char(10) not null,
    Amount smallmoney not null
);

```

### 11.5 Payments Table:

This table's main purpose is to store the payment methods of the client and this table will only function when the client opts not to choose to book as well as collect. The client will have to choose the delivery option as a type of delivery and will therefore choose whether they want to pay cash on delivery or card on delivery. The two fields are of a bit data type and are not null because once one of them has a value the other one will have to be null. Payment\_ID will be the primary key of this table with a datatype integer. There are three foreign keys in this table which are the primary keys from the Customer table, the Services table, and the Delivery table.

```

--Payment Entity--
create table Payment (
    Payment_ID int identity(1,1) primary key not null,
    Cust_ID int not null
    foreign key references Customers(Cust_ID),
    Service_ID varchar(50) not null
    foreign key references Services(Service_ID),
    Delivery_ID int not null,
    Constraint FK_Delivery1 foreign key (Delivery_ID) references Delivery(Delivery_ID),
    Cash_on_Delivery bit null,
    Card_on_Delivery bit null,
);

```

### 11.6 Output Table:

Output Table is used to confirm the client has finished requesting services. After the client is done requesting services, this table will store the foreign key from the Customer table to link up the two tables. This table will also help the customer to continue from where they left off if they did not complete the process of requesting.

```

-----Create Output Table-----
Create Table Output(
    ORDER_ID INT PRIMARY KEY NOT NULL,
    Cust_ID int,
    Constraint Fk_custome foreign key (Cust_ID) References Customers(Cust_ID)
);

```

## 12.Database Objects (additional data elements that were not included when creating tables):

### i. Adding Amount attribute to Payment Table:

This attribute will combine the amount of the service from the Service table and will combine it with the delivery amount that the company offers.

```
-----Add attribute to Payment table-----  
ALTER TABLE Payment  
ADD Amount int;
```

### ii. Adding Booking\_date attribute to Booking Table:

This Attribute belongs to the Booking table and will store the date and time that the client chooses for the company to come and do the laundry services in their houses.

```
-----Add attribute to Bookings table-----  
ALTER TABLE Bookings ADD Booking_Date datetime;
```

### iii. Changing the datatype of the Amount attribute in the Delivery\_Address table:

The data type of the Amount is changed from small money data type to int.

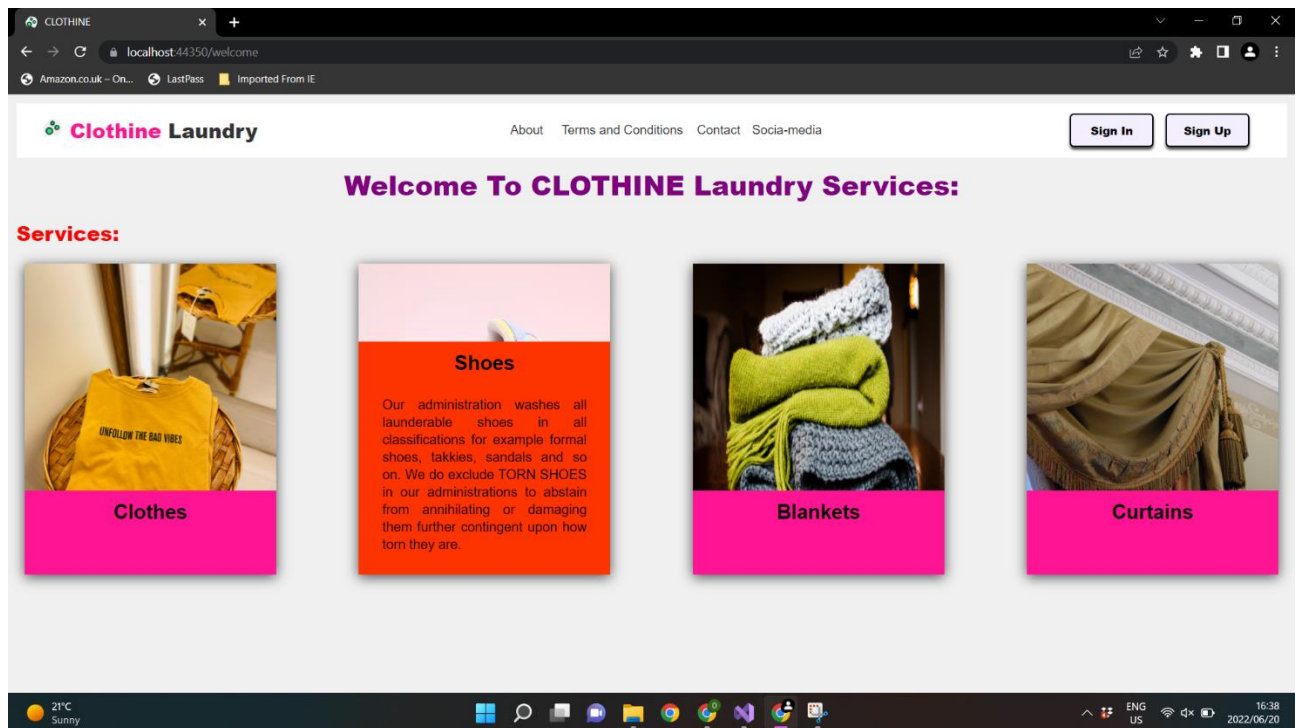
```
ALTER TABLE Delivery_Address  
alter column Amount int;
```

## 13.OUTPUTS AND INPUT DESIGN, AND PROTOTYPING:

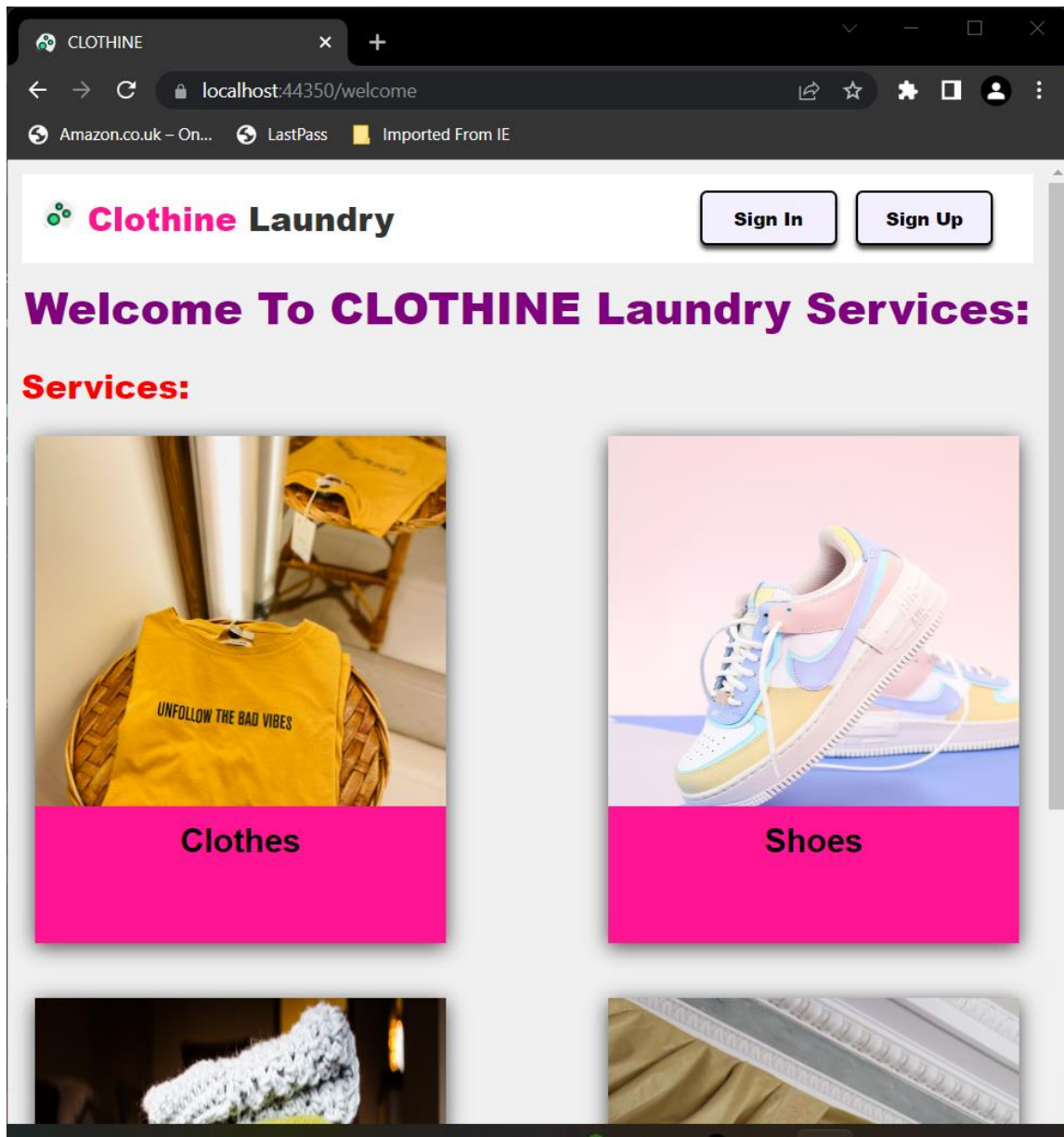
### a) WELCOME FORM:

This is the first form that the user will meet. It contains many functionalities that the client can choose from. Each Container in the middle container has information about the services we have in the web application and when the client points it with a cursor it will show information.

The form in full screen:

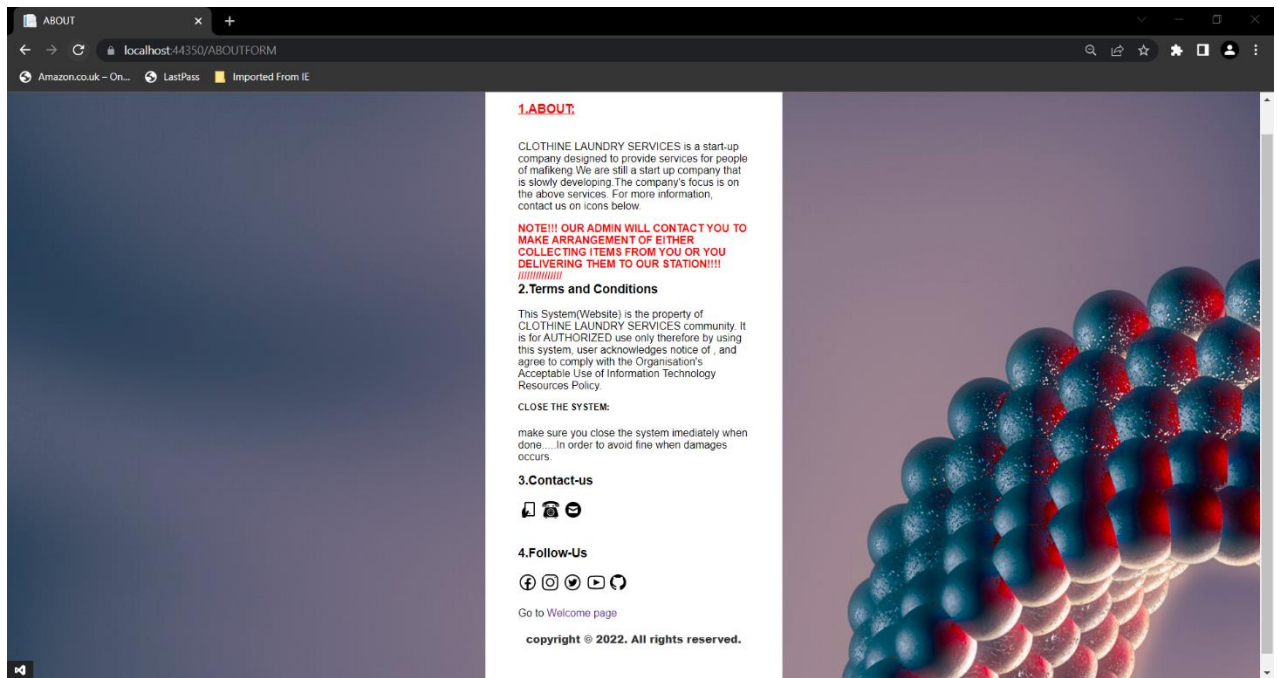


The form in half screen:



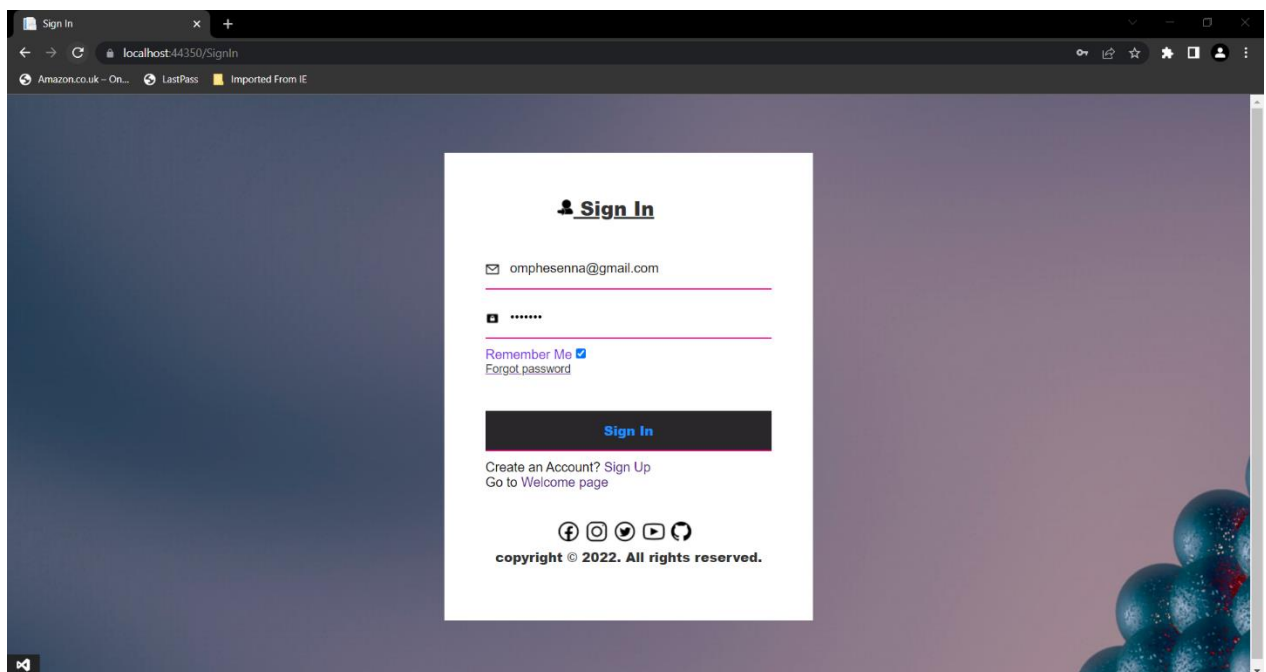
**b) ABOUT FORM:**

This form will show up when the user clicks on the About, Terms and Conditions, Contact and social media. It is a one form that contains all the options mentioned. If for instance, the user clicks on the About button, the About form will be shown and the About heading in that form will be highlighted to show the client to focus on that part.



### c) SIGN IN FORM:

This form will be shown when the client clicks on the “Sign In” button in the Welcome Form. This form will let the user inside the web application when the user’s credentials are correct. If the users forgot the password, they could click on the forgot password link that will take them to the forgot password page as well as other links.





**d) SIGN UP FORM:**

This form will capture the details of the client after they have been entered. New clients will have to register before signing into the system. The user will get to this form Welcome form or Sign in form using a link.

REGISTER FORM

localhost:44350/Signup

Amazon.co.uk - On... LastPass Imported From IE

### Sign-Up

OMPHEMETSE SENNA

omphesenna@gmail.com

Please include an '@' in the email address. 'omphesenna@gmail.com' is missing an '@'.

Password

☐ I agree to this site's [User Agreement](#) and [Privacy Policy](#)

[Sign Up](#)

Already have an Account? [Login page](#)  
[Go to Welcome page](#)

[Facebook](#) [Instagram](#) [Twitter](#) [YouTube](#) [LinkedIn](#)

copyright © 2022. All rights reserved.

21°C Sunny

ENG US 16:34 2022/06/20

**e) FORGOT PASSWORD FORM:**

This form will be used by the client to request for the password went forgotten. The client will have to confirm the email, if they are both correct and available in our database, the password will be sent via email to the client.

REQUEST PASSWORD

localhost:44350/forgotpassword

Amazon.co.uk - On... LastPass Imported From IE

### Recover password

Enter your Email

Confirm your Email

Your password will be sent to your Email

[Sign In](#)

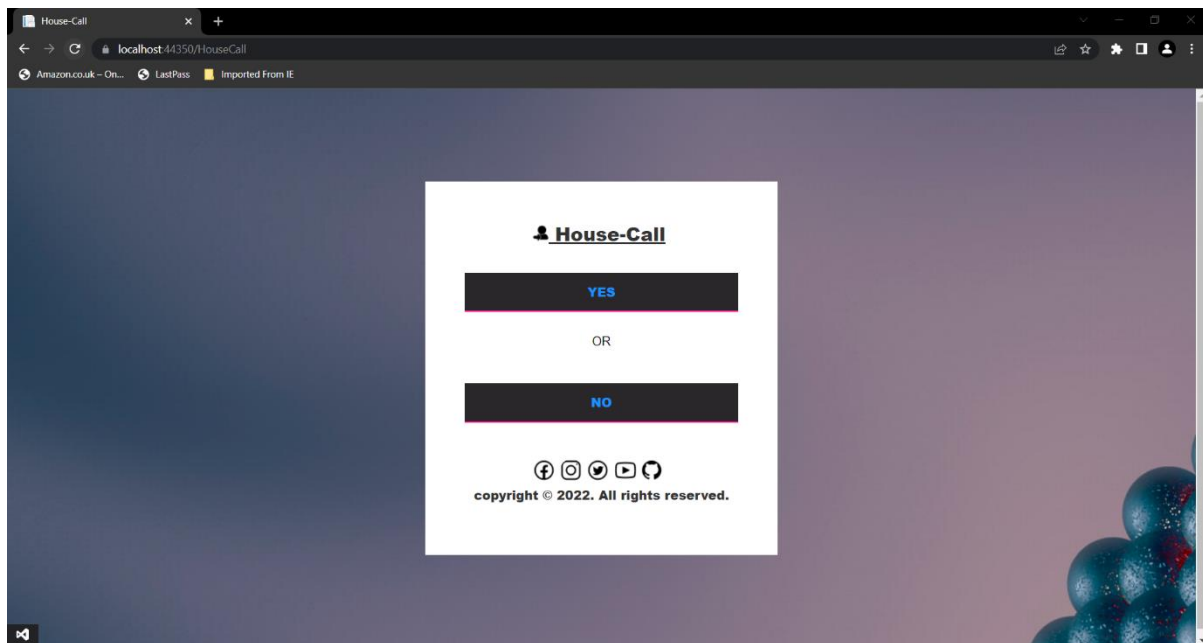
Cancel and return to login page or to signup page

[Facebook](#) [Instagram](#) [Twitter](#) [YouTube](#) [LinkedIn](#)



**f) HOUSE-CALL FORM:**

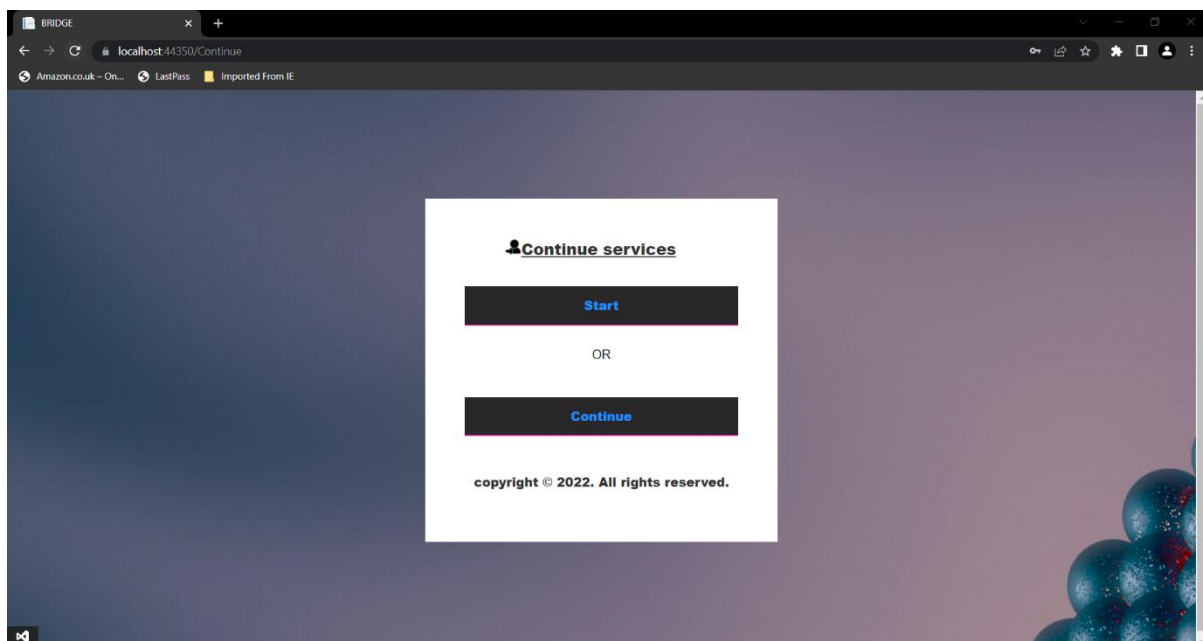
After the client has logged in, this page will ask the client to choose one of the options if they want to do the booking to allow our employee to do the laundry at the client's house.



A screenshot of a web browser displaying the 'House-Call' form. The browser's address bar shows 'localhost:44350/HouseCall'. The form is centered on a dark purple background with a decorative blue and red pattern in the bottom right corner. The form itself is a white box with a dark header containing a person icon and the text 'House-Call'. Below the header, there are two large, dark buttons with blue text: 'YES' and 'NO', separated by the word 'OR'. At the bottom of the form, there are social media icons for Facebook, Instagram, Twitter, YouTube, and LinkedIn, followed by the text 'copyright © 2022. All rights reserved.'

**g) CONTINUE FORM (optional):**

This form will only appear if the client did not finish applying for services. It will only show after the client has logged in. If the client chooses yes, the system will take the client to the form where they left off else the system will delete the record that the client had and start afresh from the house-call form.



A screenshot of a web browser displaying the 'Continue services' form. The browser's address bar shows 'localhost:44350/Continue'. The form is centered on a dark purple background with a decorative blue and red pattern in the bottom right corner. The form itself is a white box with a dark header containing a person icon and the text 'Continue services'. Below the header, there are two large, dark buttons with blue text: 'Start' and 'Continue', separated by the word 'OR'. At the bottom of the form, there is the text 'copyright © 2022. All rights reserved.'

#### h) BOOKING FORM:

The Booking form will only show up when the client chooses the “Yes” option in the House-call form. The client will have to fill all the mandatory spaces provided. When the client clicks on the display Calendar the calendar will pop up on the screen to allow the client to choose the date. Then the Client clicks on the time textbox the time will also pop up in the screen. Clear button will clear all the filled textboxes.

MAKE A BOOKING

localhost:44350/Booking

Amazon.co.uk - On... LastPass Imported From IE

### Make-Booking:

my address 114 Address3 Optional

vaal City

Postal/ZipCode

**Choose House-Date And Time Below:**

Choose Time:  
20:50

2022/07/06 20:50 Display calander

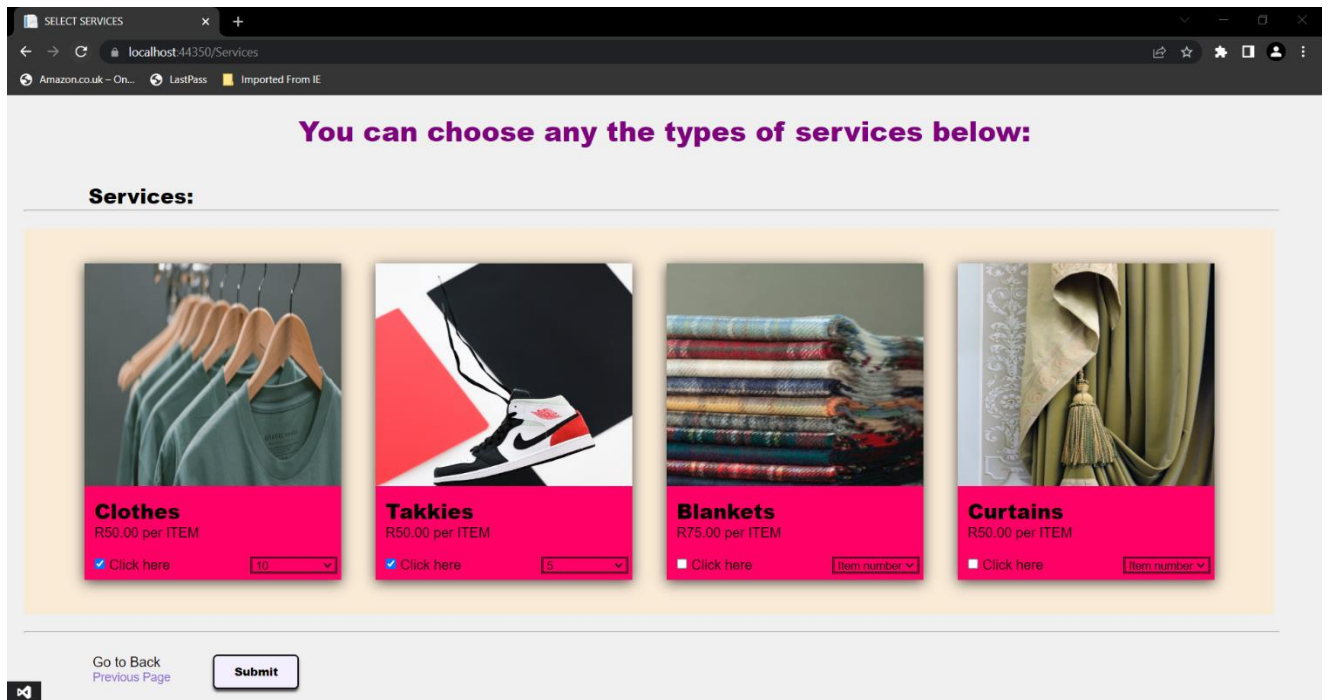
**Submit**

Go to Back  
Previous Page

Clear

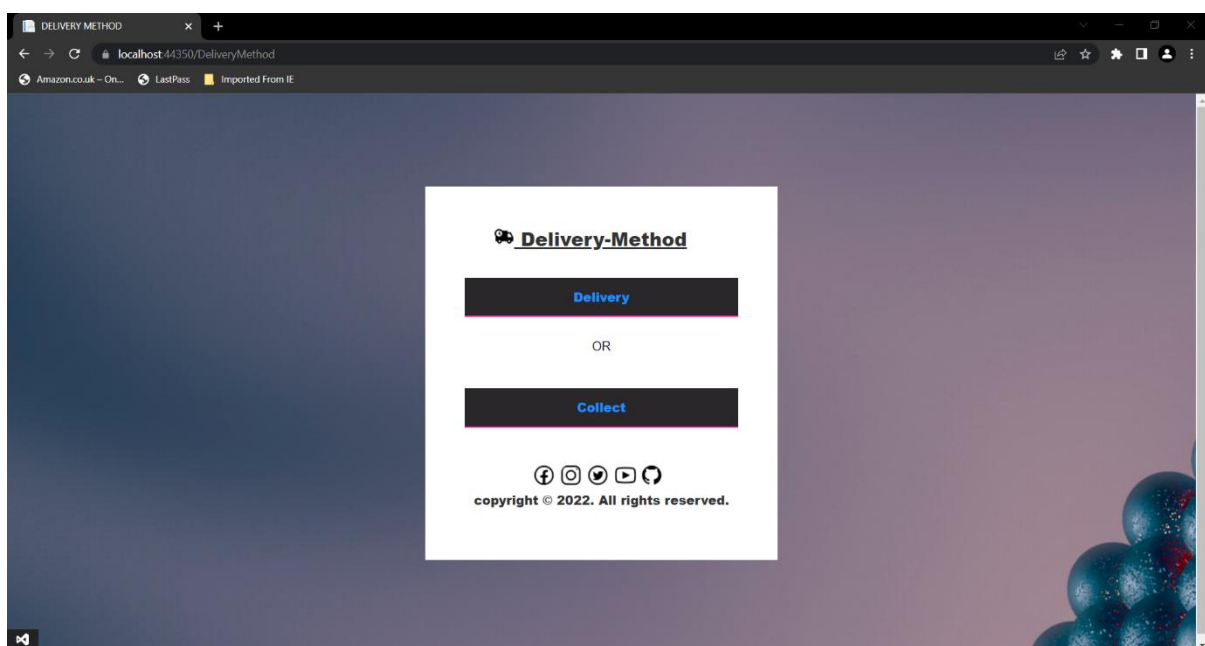
#### i) SERVICES FORM:

The services form will show all the services the company provides, and the client will have to choose from them and the quantity of the services they have chosen.



#### j) DELIVERY FORM:

The delivery form will allow the client to choose if they want to collect their items after service or want them delivered to their location. If the client chooses to collect the system will take them to the Output(results) form else to the delivery address form.



#### k) DELIVERY ADDRESS FORM:

This form will have to be filled by the client after the client has clicked or pressed on the submit button.

**DELIVERY ADDRESS**

localhost:44350/Deliv\_Address

**Location to Deliver Items:**

Address1 Address2

Address3 Optional City

Postal/ZipCode

Collectors Name Collectors surname

Collectors Cellphone number

**Submit**

[Go to Back Previous Page](#) [Clear](#)

#### l) PAYMENT FORM:

In this form, the client will choose if they pay cash during delivery, or they will use card during delivery. After choosing the client will be directed to the Output(result) form.

**PAYMENT OPTION**

localhost:44350/PaymentsMethods

**Choose method of payment:**  
The Distance costs at the bottom area for Delivery options.

**Card on Delivery** **Cash on Delivery**

**Distance Costs:**

Delivery Price = R75.00

#### m) OUTPUT FORM:

When the client is done with the above forms, the Output form will appear with the data of the client to confirm if they are correct or not. If they client find them accurate, he or she can press confirm button. After pressing the button, the email will be sent to the client and the admin.

YOUR SERVICES

localhost:44350/OutputForm

Amazon.co.uk - On... LastPass Imported From IE

## THANK YOU FOR CHOOSING OUR ORGANISATION..

Welcome, Omphe. Please confirm if the following are your chosen options..

**Services:**

PRODUCT ID: 3685

Tekkie	Tekkie_num	Clothes	Clothes_num	Blankets	Blankets_num	Curtains	Curtains_num
<input type="checkbox"/>	0	<input type="checkbox"/>	8	<input type="checkbox"/>	0	<input type="checkbox"/>	0

**Payments:**

SERVICE PRICE R:400  
Total price to pay is R: 475 Method of payment will be cash on Delivery

**Booking Address:**

No Booking made

**Delivery Method:**

Method chosen for item distribution is delivery.

Address1	Address2	Address3	City	Postal_Code
ad	awd		adwd	1747

//Your records will be saved after confirming!!!!.....  
Thank you!!!!!!

[Confirm](#)

**n) CLOSE FORM (optional):**

This form will appear after the client has confirmed on the Output form. It will take three (3) seconds after opening and redirect the user back to welcome form.

