**ONLINE BUS REGISTRATION SYSTEM**

A project report on a computerized system for bus registration

Online bus registration system is a database system that stores information about buses, their owners, drivers, the Sacco the operate and also the route the work on. It is built to work online and to easily keep track of all activities in public service vehicle (psv) sector. In case of any need to get information about a vehicle or employee, the system provides the information at hand easily.

ER DATA MODELS DESIGN

## Entities and attributes

The following are the entities and their attributes available in our databases;

1. Vehicles Table

* Vehicle registration number (primary Key)
* Vehicle name
* Capacity
* Owner ID
* Route ID
* Sacco ID
* Employee ID

1. Owners Table

* Owners ID (primary Key)
* Owners name
* Phone number
* Email address
* Home address

1. Sacco Table

* Sacco ID (primary key)
* Sacco name
* Description

1. Route Table

* Route ID (primary key)
* Route Name
* Description

1. Employee Table

* Employee ID (primary key)
* Employee name
* Phone number
* Email address
* Address

These tables are linked using their primary keys. The vehicles table is seen as the mother table that provide the center point of the relationship.

## Entity description

### Vehicles table

|  |  |  |
| --- | --- | --- |
| **Attribute name** | **Data type** | **Key/null** |
| Vehicle regno | Int | Primary key |
| Vehicle name | Varchar | null |
| Capacity | Int | null |
| Owner Id | Int | Not null |
| Employee ID | Int | Not null |
| Route ID | Int | Not null |
| Sacco ID | Int | Not null |

Table : vehicle table

### Owner table

|  |  |  |
| --- | --- | --- |
| **Attribute name** | **Data type** | **Key/null** |
| Owners ID | int | Primary key |
| Owners name | varchar | Null |
| Phone number | Int | Null |
| Email address | varchar | Not null |
| Addresss | varchar | Null |

Table : Owners Table

### Employees table

|  |  |  |
| --- | --- | --- |
| **Attribute name** | **Data type** | **Key/null** |
| Employee ID | Int | Primary key |
| Employee name | Varchar | Null |
| Phone number | Int | Not null |
| Email address | Varchar | Not null |
| Address | Varchar | Not null |

Table : Employees table

### Sacco table

|  |  |  |
| --- | --- | --- |
| **Attribute name** | **Data type** | **Key/null** |
| Sacco ID | Int | Primary key |
| Sacco name | Varchar | Null |
| Description | Varchar | null |

Table :Sacco table

### Route table

|  |  |  |
| --- | --- | --- |
| **Attribute name** | **Data types** | **Key/null** |
| Route ID | int | Primary key |
| Route name | Varchar | Null |
| Description | Varchar | Null |

Table : Route table

## Relationship specification

The vehicle table being the primary table in our database all the tables are linked to it through their primary keys. The figure below shows how our tables are related.

Figure : Relationship diagram

Owner ID FK

Route ID FK

Employee ID FK

Sacco ID PK

Vehicles Table

Employee ID PK

Employee Table

Sacco ID PK

Sacco Table

Owner table

Owner ID PK

Route Table

Route ID PK

|  |  |  |  |
| --- | --- | --- | --- |
| **Primary Table** | **Foreign Table** | **Degree of relationship** | **Constraints** |
| Owner table | Vehicle | 1-Many | Referential integrity   * Cascade delete * Cascade update |
| Employees table | Vehicle | 1-1 | Referential integrity   * Cascade delete * Cascade update |
| Sacco table | Vehicle | 1-Many | Referential integrity   * Cascade delete * Cascade update |
| Route table | Vehicle | 1-Many | Referential integrity   * Cascade delete * Cascade update |

Figure : ER table

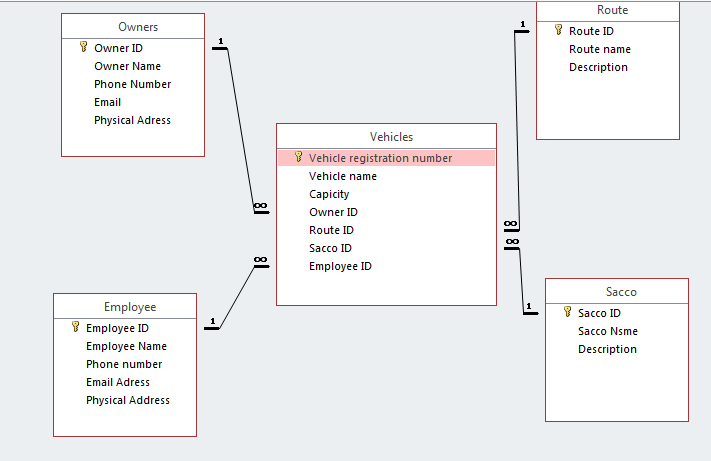


Figure : Relationship snapshot

# CHAPTER 3: IMPLEMENTATION OF DATABASE AND SQL.

## Database and Table creation

The following SQL was used to initially come up with the database before implementing it with Microsoft access.

**Vehicles table**

The following code was used.



Figure Create database sql

We added primary using the following sql,

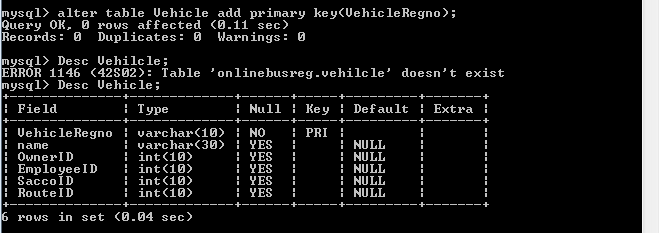


Figure Alter table sql

**Owners table**



Figure : owner table SQL

Adding primary key,

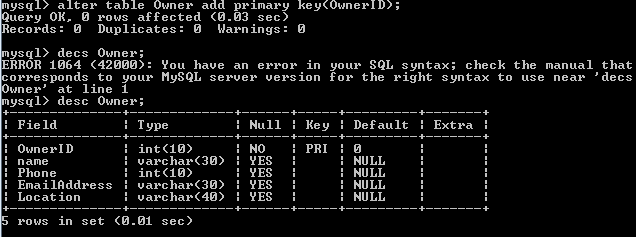


Figure : alter Owner table

Table view with test data.

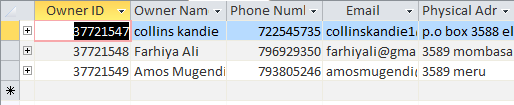


Figure : Access Tabular format

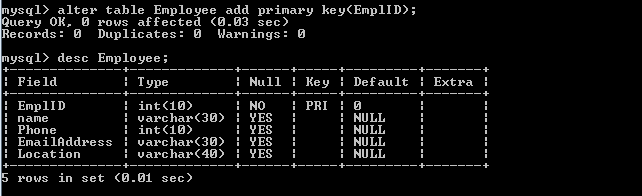
**Employees table**

Using MySQL.



Figure : Employee SQL

Adding primary key and describing it.

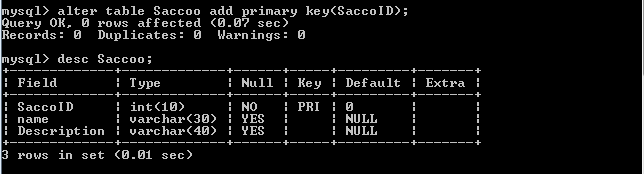


**Sacco Table**

Using MySQL.



Adding primary key then describing it.

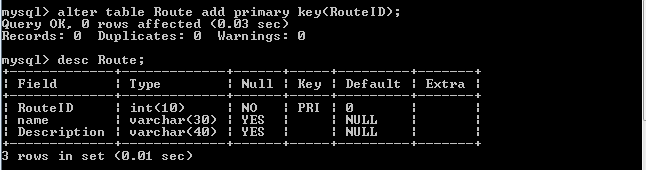


**Route Table.**

MySQL statements.



Adding primary key and describing the table;



## Querying the database

During this process, we used entirely Microsoft access for the purpose of clear and understandable reports. However, the statements used to query the database are purely SQL statements. The following are the statements and there functions.

**Owner Information.**

This query is used to select data from the table Owners and Vehicle, it displays information like Owner ID, Name, Phone Number, Email, physical address, vehicle registration number and vehicle name base on the values input in the two tables. The relationship between these tables enables this query to extract this data.

***SQL statement***

*SELECT Owners.[Owner ID], Owners.[Owner Name], Owners.[Phone Number], Owners.Email, Owners.[Physical Adress], Vehicles.[Vehicle registration number], Vehicles.[Vehicle name]*

*FROM Owners INNER JOIN Vehicles ON Owners.[Owner ID] = Vehicles.[Owner ID];*

**Route and Sacco**

This query uses the table Sacco, Route and vehicles to extract data about Sacco’s and routes that share the same vehicle.

***SQL statement***

*SELECT Route.[Route ID], Route.[Route name], Sacco.[Sacco ID], Sacco.[Sacco Nsme], Sacco.Description*

*FROM Sacco INNER JOIN (Route INNER JOIN Vehicles ON Route.[Route ID] = Vehicles.[Route ID]) ON Sacco.[Sacco ID] = Vehicles.[Sacco ID];*

**Route and its vehicle**

This query extracts data from the table vehicle and Route based on the relationship between the tables. The data from this query are displayed in a report that lists all the routs and the vehicles that belong to these routes.

***SQL statement***

*SELECT Route.[Route ID], Route.[Route name], Route.Description, Vehicles.[Vehicle registration number], Vehicles.[Vehicle name], Vehicles.Capicity*

*FROM Route INNER JOIN Vehicles ON Route.[Route ID] = Vehicles.[Route ID];*

**Vehicle Info**

This query show data in a report about the vehicle, its owner, the route it operates, the sacco it belongs to and the driver information. The tables involved are vehicles, sacco, route, owners and employee table and drivers table.

***SQL statement***

*SELECT Vehicles.[Vehicle registration number], Vehicles.[Vehicle name], Vehicles.Capicity, Owners.[Owner ID] AS [Owners\_Owner ID], Owners.[Owner Name], Owners.[Phone Number], Owners.Email, Route.[Route ID] AS [Route\_Route ID], Route.[Route name], Route.Description AS Route\_Description, Sacco.[Sacco ID] AS [Sacco\_Sacco ID], Sacco.[Sacco Nsme], Sacco.Description AS Sacco\_Description*

*FROM Employee INNER JOIN (Sacco INNER JOIN (Route INNER JOIN (Owners INNER JOIN Vehicles ON Owners.[Owner ID] = Vehicles.[Owner ID]) ON Route.[Route ID] = Vehicles.[Route ID]) ON Sacco.[Sacco ID] = Vehicles.[Sacco ID]) ON Employee.[Employee ID] = Vehicles.[Employee ID];*

**Vehicle and Sacco**

Displays information about vehicles and the sacco they belong to specifically. It only uses the table vehicle and sacco.

***SQL statement***

*SELECT Sacco.[Sacco ID], Sacco.[Sacco Nsme], Vehicles.[Vehicle registration number], Vehicles.[Vehicle name], Vehicles.Capicity*

*FROM Sacco INNER JOIN Vehicles ON Sacco.[Sacco ID] = Vehicles.[Sacco ID];*

# CHAPTER 4: APPLICATION DESIGN

## Description of the system

The system is purely driven by GUI (graphical User interface) this makes it easier for anyone using the system. Its uses user friendly colors that do not irritate the user.

It is a secure system that allow only registered user to log into the system.

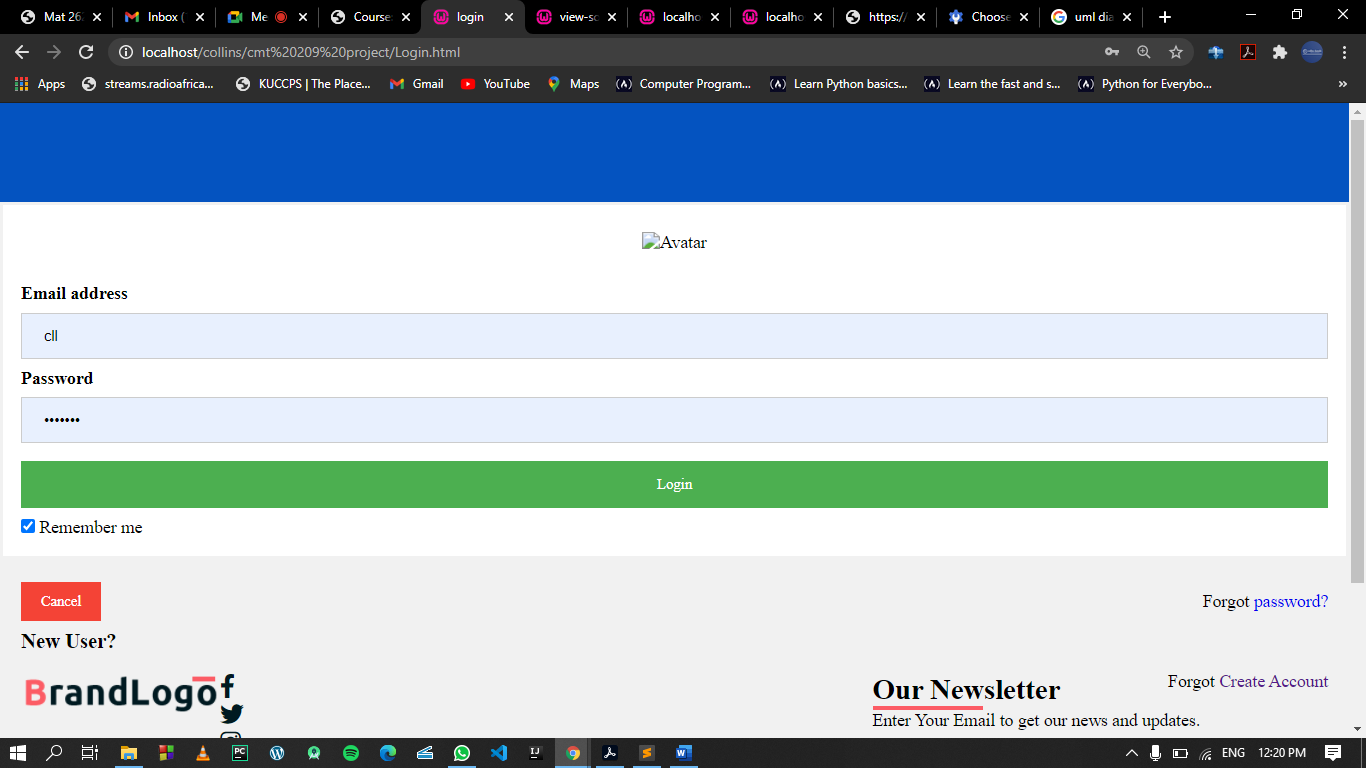


Figure : Login interface

User that are using the system for the first time are required to create new account.

Online bus registration system accepts inputs in forms then stores the data in tables. User are able to enter as many data as possible. Retrial of information from the system is based on reports, which the user queries, directly by selecting the desired report from the report tab.

The design of user interface and buttons in the system we used Visual Basic and Microsoft macros.

Examples of the source codes;

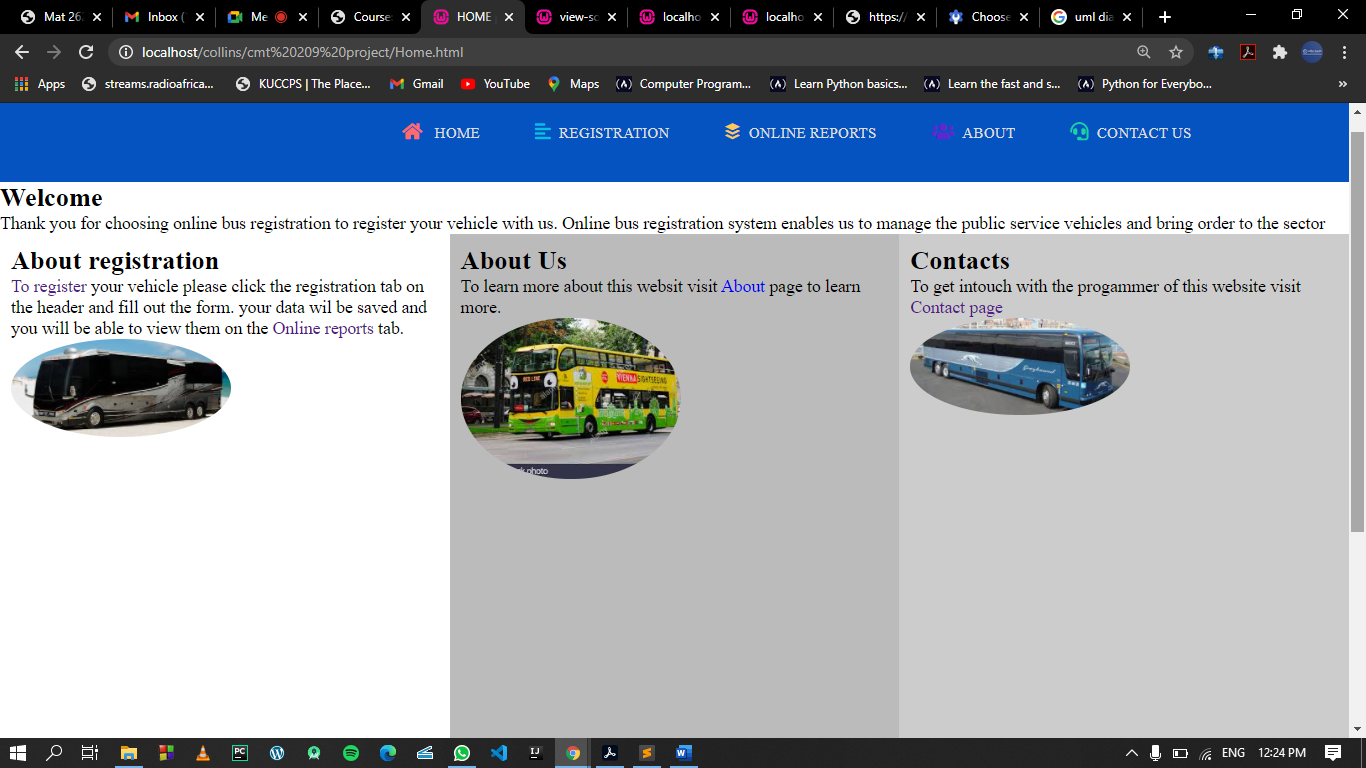
*Login page*

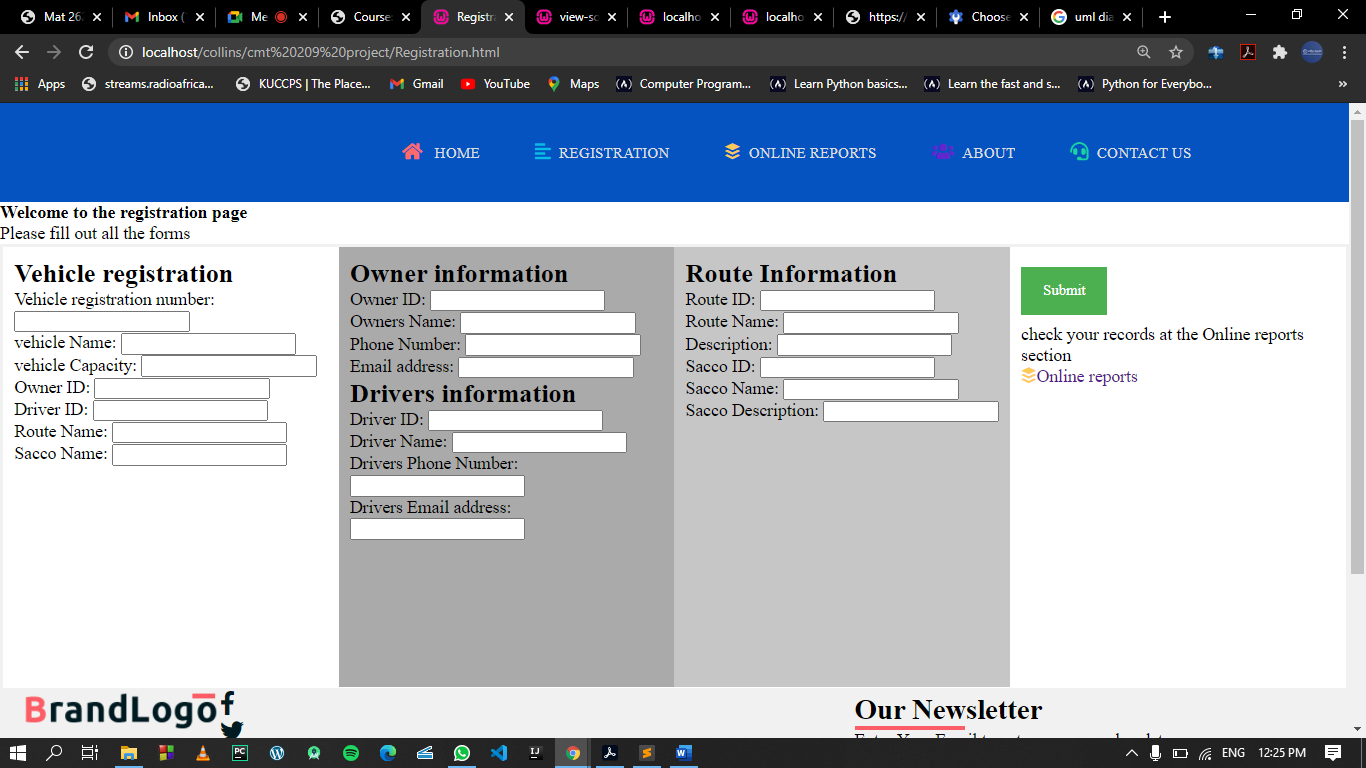
## User Guide

Ideally the system was to be accessed online. To access our database you are supposed to enter the following URL in your web browser (preferably google crome) [www.onlinebusreg.ac.ke](http://www.onlinebusreg.ac.ke).

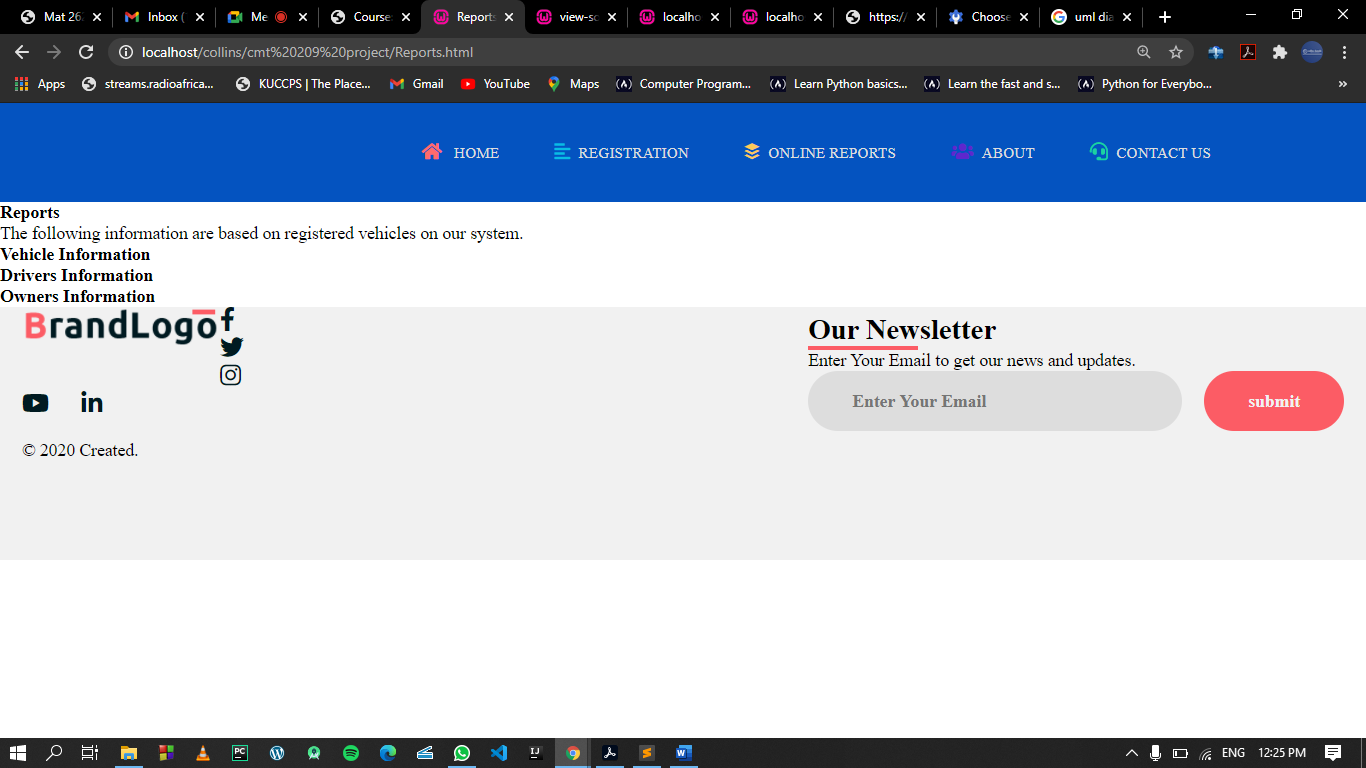
You will be required to create a new account so that you can log in.

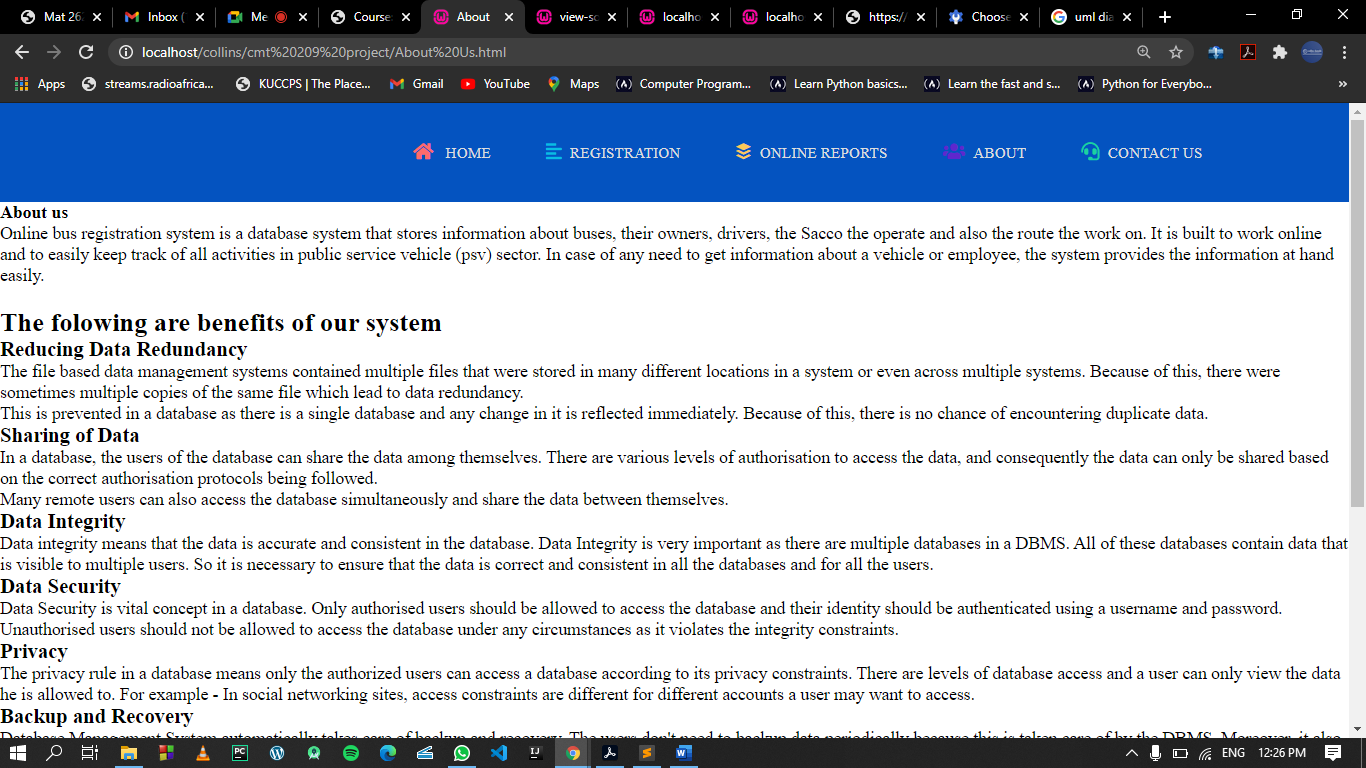
After logging in you will be prompted by our home/



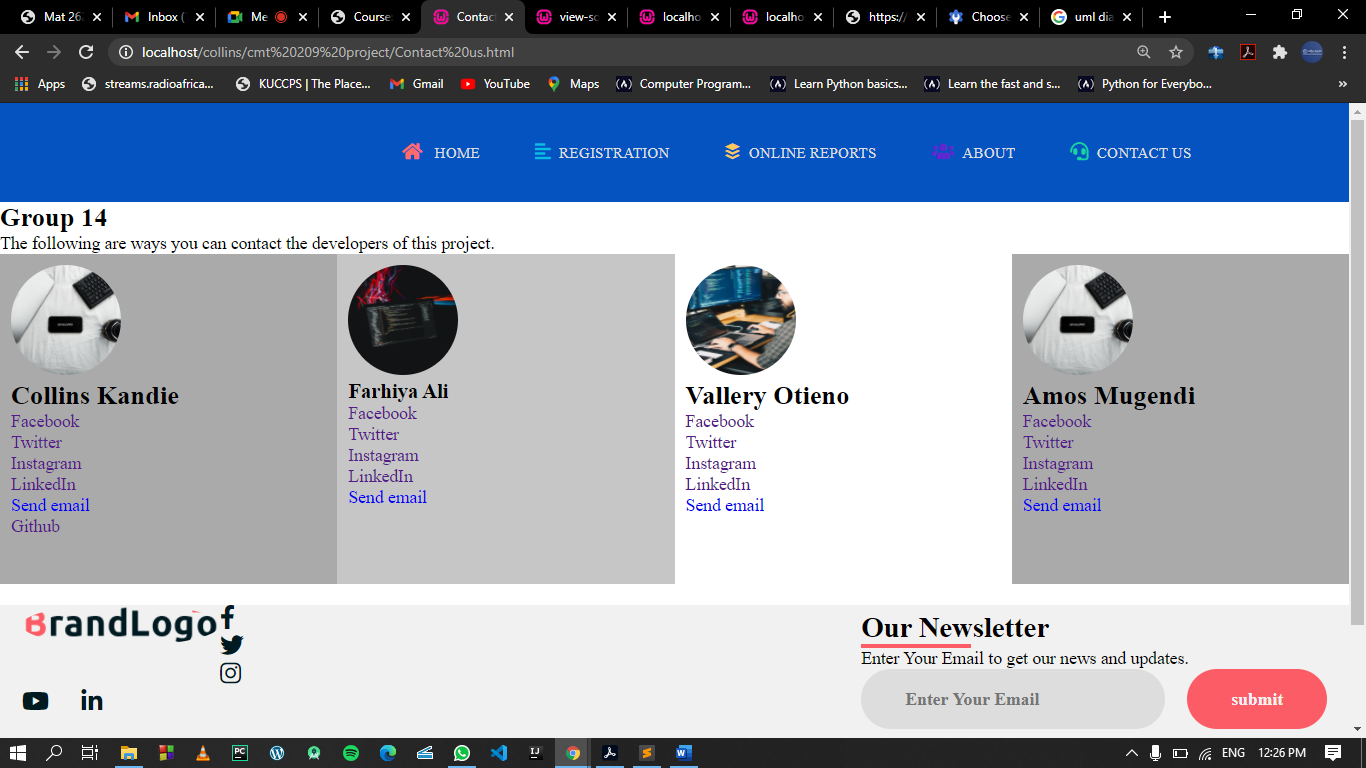


From here you will be able to choose your preferred choice of action.





About..



How to get in touch.