**Assessment Cover Sheet**

* Complete and attach this cover sheet to your assessment before submitting

**Assessment Title**

*Individual Project*

**Programme Title:**

*Bachelor of Information, Communication and Technology*

IT-7405

**Course No.:**

*Web-Development with Non-Relational Databases*

**Course Title:**

*Ali Hussain Khamis*

**Student Name:**

*202000483*

**Student ID:**

Sini Raj Pulari

**Tutor:**

**Date submitted:**

**Due Date:** Tuesday, 3 rd Jan 2022 by 11:55 p.m.

***By submitting this assessment for marking, either electronically or as hard copy, I confirm the following:***

* This assignment is **my own work**
* Any information used has been properly referenced.
* I understand that a copy of my work may be used for moderation.
* I have kept a copy of this assignment

Do not write below this line. For Polytechnic use only*.*

**Date of Marking:**

**Assessor:**

**Grade/Mark:**

**Comments:**

Contents

[Business Motivation and Problem Statement 3](#_Toc123032217)

[The main Functionalities are: 3](#_Toc123032218)

[Additional Functionality: 3](#_Toc123032219)

[Creating the No-SQL MongoDB Database and Data Modeling 3](#_Toc123032220)

[Running the server 3](#_Toc123032221)

[Creating Database and collection and Inserting 3](#_Toc123032222)

[Querying 4](#_Toc123032223)

[Update 9](#_Toc123032224)

[Replace 10](#_Toc123032225)

[Indexes 10](#_Toc123032226)

[Delete 10](#_Toc123032227)

[Query Diagnosis and Analysis 11](#_Toc123032228)

[Query Planner Mode 11](#_Toc123032229)

[Execution Stats Mode 11](#_Toc123032230)

[All Plans Execution Mode 12](#_Toc123032231)

[Django 13](#_Toc123032232)

[Creating car view app 16](#_Toc123032233)

[Creating Search Bar 20](#_Toc123032234)

[Add Car 22](#_Toc123032235)

[Admin 25](#_Toc123032236)

[Login 27](#_Toc123032237)

[Registration 29](#_Toc123032238)

[Reflection 32](#_Toc123032239)

[Figure 1 Running Server 5](#_Toc123045228)

[Figure 2 Running Terminal 5](#_Toc123045229)

[Figure 3 Creating Database 5](#_Toc123045230)

[Figure 4 Inserting Multiple Data 6](file:///D:\IT7405_202000483\IT7405_202000483.docx#_Toc123045231)

[Figure 5 Inserting One data 6](file:///D:\IT7405_202000483\IT7405_202000483.docx#_Toc123045232)

[Figure 6 Retrieving All Data 6](file:///D:\IT7405_202000483\IT7405_202000483.docx#_Toc123045233)

[Figure 7 Searching Database by Name 7](#_Toc123045234)

[Figure 8 Searching by IN 8](#_Toc123045235)

[Figure 9 Searching by AND 9](#_Toc123045236)

[Figure 10 Searching by OR 10](#_Toc123045237)

[Figure 11 Searching with OR and AND 10](file:///D:\IT7405_202000483\IT7405_202000483.docx#_Toc123045238)

[Figure 12 Update Many 11](#_Toc123045239)

[Figure 13 Update One 11](#_Toc123045240)

[Figure 14 Replace One 12](file:///D:\IT7405_202000483\IT7405_202000483.docx#_Toc123045241)

[Figure 15 Indexes 12](#_Toc123045242)

[Figure 16 Delete One 12](#_Toc123045243)

[Figure 17 Delete Many 12](#_Toc123045244)

[Figure 18 Query Planner Mode 13](#_Toc123045245)

[Figure 19Execution Stats Mode 13](file:///D:\IT7405_202000483\IT7405_202000483.docx#_Toc123045246)

[Figure 20 All Plans Execution Mode 14](#_Toc123045247)

[Figure 21 Virtual Environment 15](#_Toc123045248)

[Figure 22 Django Project 15](#_Toc123045249)

[Figure 23 Requirements 15](#_Toc123045250)

[Figure 24 Command 16](#_Toc123045251)

[Figure 25 Website 16](#_Toc123045252)

[Figure 26 Database Connection 16](#_Toc123045253)

[Figure 27 Base Files 17](#_Toc123045254)

[Figure 28 Directory Path 17](#_Toc123045255)

[Figure 29 Static 17](#_Toc123045256)

[Figure 30 Migration 17](#_Toc123045257)

[Figure 31 Images 18](#_Toc123045258)

[Figure 32 Car Model 18](#_Toc123045259)

[Figure 33 Car view 19](#_Toc123045260)

[Figure 34 Creating Files 19](#_Toc123045261)

[Figure 35 HTML Code 19](#_Toc123045262)

[Figure 36 URL 20](#_Toc123045263)

[Figure 37 car view to settings 20](#_Toc123045264)

[Figure 38 Adding car view URL pattern 20](#_Toc123045265)

[Figure 39 Inserting Data 21](#_Toc123045266)

[Figure 40 First phase of website done 21](#_Toc123045267)

[Figure 41 Search working 23](#_Toc123045268)

[Figure 42 Add Car Form 24](#_Toc123045269)

[Figure 43 Image Uploader Function 24](#_Toc123045270)

[Figure 44 Add Car view 25](#_Toc123045271)

[Figure 45 Add Car URL 25](#_Toc123045272)

[Figure 46 Add Car in installed apps 26](#_Toc123045273)

[Figure 47 URL Mapping 26](#_Toc123045274)

[Figure 48 installing crispy 26](#_Toc123045275)

[Figure 49 Add car page 26](#_Toc123045276)

[Figure 50 Final Page 27](#_Toc123045277)

[Figure 51 Super User 27](#_Toc123045278)

[Figure 52 Admin Login 28](#_Toc123045279)

[Figure 53 Admin Page 28](#_Toc123045280)

[Figure 54 Code 28](#_Toc123045281)

[Figure 55 Updated admin page 28](#_Toc123045282)

[Figure 56 added users to the installed applications 29](#_Toc123045283)

[Figure 57 Redirection Paths 29](#_Toc123045284)

[Figure 58 Adding user URL Pattern 29](#_Toc123045285)

[Figure 59 updated nav 30](#_Toc123045286)

[Figure 60 Hello to the user logged in 30](#_Toc123045287)

[Figure 61 Three pages 30](#_Toc123045288)

[Figure 62 Page one 30](#_Toc123045289)

[Figure 63 Page two 31](#_Toc123045290)

[Figure 64 Page Three 31](#_Toc123045291)

[Figure 65 Register view 32](#_Toc123045292)

[Figure 66 URL code 33](#_Toc123045293)

[Figure 67 register HTML page 33](#_Toc123045294)

[Figure 68 Registration in nav bar 33](#_Toc123045295)

[Figure 69 installed application register 34](#_Toc123045296)

[Figure 70 register URL path 34](#_Toc123045297)

# Business Motivation and Problem Statement

Finding Transport is every important because it provide us with independence and freedom of mobility. But many people have a hard time to find the correct car that suits their needs or if it is within the budget. So, to tackle this dilemma, I decided to create a Car Sales website.

## The main Functionalities are:

* Add Car.
* Login
* Registration.

## Additional Functionality:

* Search Bar
* Change password

# Creating the No-SQL MongoDB Database and Data Modeling

## Running the server

First, I ran the mongod.exe which is the mongo dB server.

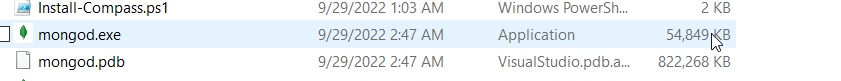


Figure Running Server

Secondly, I opened mongosh which is the terminal

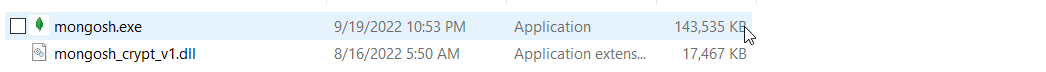


Figure Running Terminal

## Creating Database and collection and Inserting

Use keyword to create the database



Figure Creating Database

Graphical user interface

Description automatically generatedInserting into nonexistent collection will create the collection automatically

Figure Inserting Multiple Data

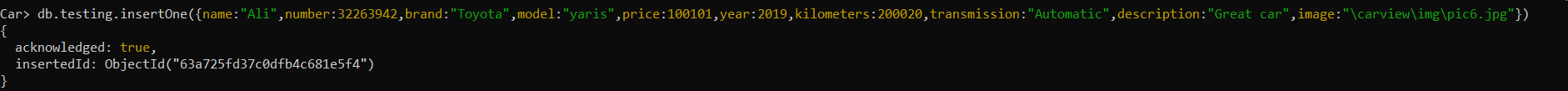
Another way of inserting is insert one

Figure Inserting One data

## Querying

Find all retrieves everything in the database

Text

Description automatically generated

Figure Retrieving All Data

Finding with specific fields

Figure Searching Database by Name

Search by in

****

Figure Searching by IN

Searching with AND

Text

Description automatically generated

Figure Searching by AND

Specify OR condition



Figure Searching by OR

Specify OR as well as AND condition

Figure Searching with OR and AND

## Update

Update Many

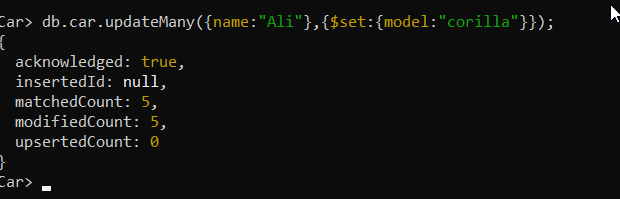


Figure Update Many

Update One

Text

Description automatically generated

Figure Update One

## Replace

Figure Replace One

## Indexes

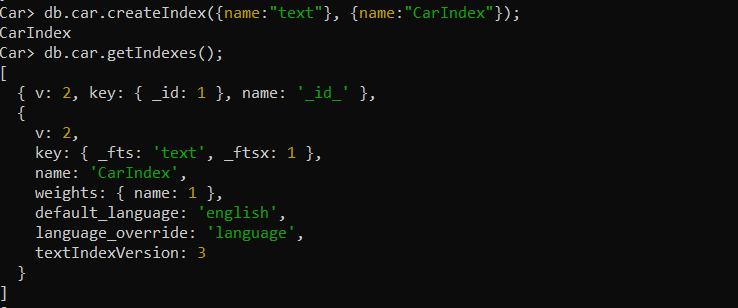


Figure Indexes

## Delete

Delete One

Text

Description automatically generated

Figure Delete One

Delete Two



Figure Delete Many

## Query Diagnosis and Analysis

### Query Planner Mode

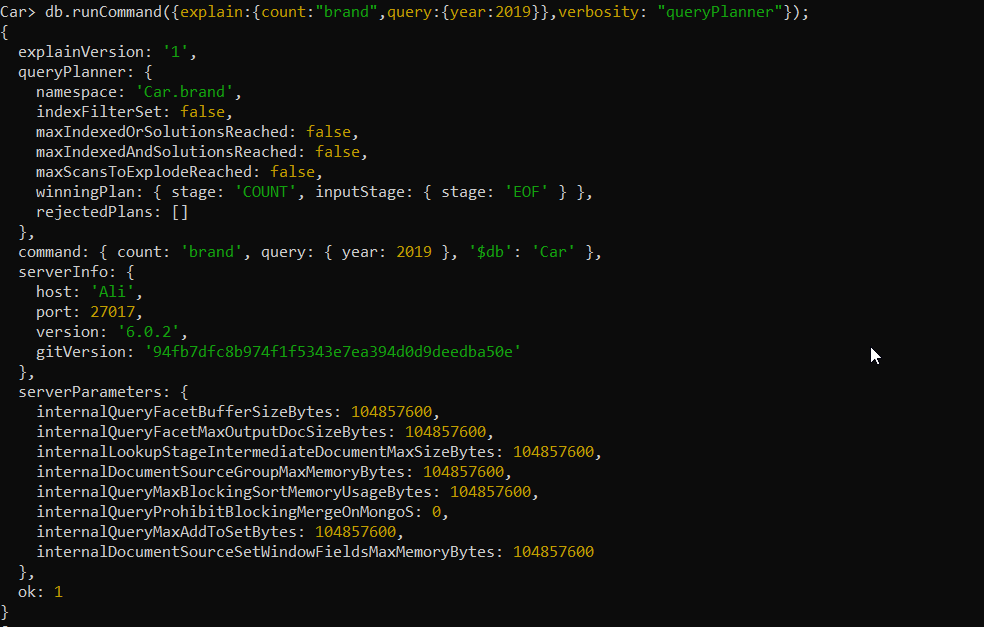


Figure Query Planner Mode

### Execution Stats Mode

Figure Execution Stats Mode

### All Plans Execution Mode



Figure All Plans Execution Mode

# Django

Creating a virtual environment

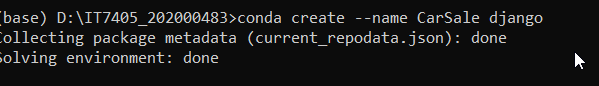


Figure Virtual Environment

Activated and created a Django project

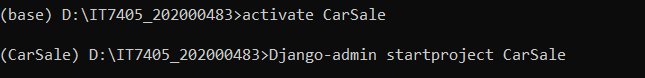


Figure Django Project

Downloaded the requirements

Text

Description automatically generated

Figure Requirements

Running Server

Text

Description automatically generated

Figure Command

Graphical user interface, text

Description automatically generated

Figure Website

Connect to Database

Text

Description automatically generated

Figure Database Connection

In the main project I created a templates folder where I will store my base and navbar html and I added the directory path to the settings template and static URL.

A picture containing text

Description automatically generated

Figure Base Files

Text

Description automatically generated

Figure Directory Path

A screenshot of a computer screen

Description automatically generated with medium confidence

Figure Static

Migrating to the database

Text

Description automatically generated

Figure Migration

## Creating car view app



Creating static folder for the images

Text

Description automatically generated

Figure Images

Creating a car model

Text

Description automatically generated

Figure Car Model

Created a view for car view

Text

Description automatically generated

Figure Car view

Created a car view html page and URLs file

Text

Description automatically generated

Figure Creating Files

Code html

Text

Description automatically generated

Figure HTML Code

URL mapping

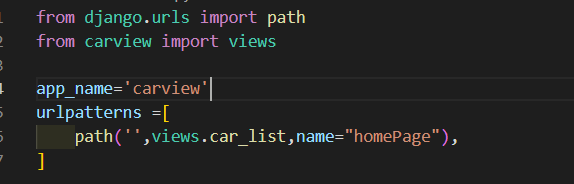


Figure URL

Adding car view to settings

Text

Description automatically generated

Figure car view to settings

Imported include and added car view URLs to URL patterns

Text

Description automatically generated

Figure Adding car view URL pattern

Populating database with python shell

Text

Description automatically generated

Figure Inserting Data

Loading website

Graphical user interface, website

Description automatically generated

Figure First phase of website done

## Creating Search Bar

Creating Search bar view

Graphical user interface, text

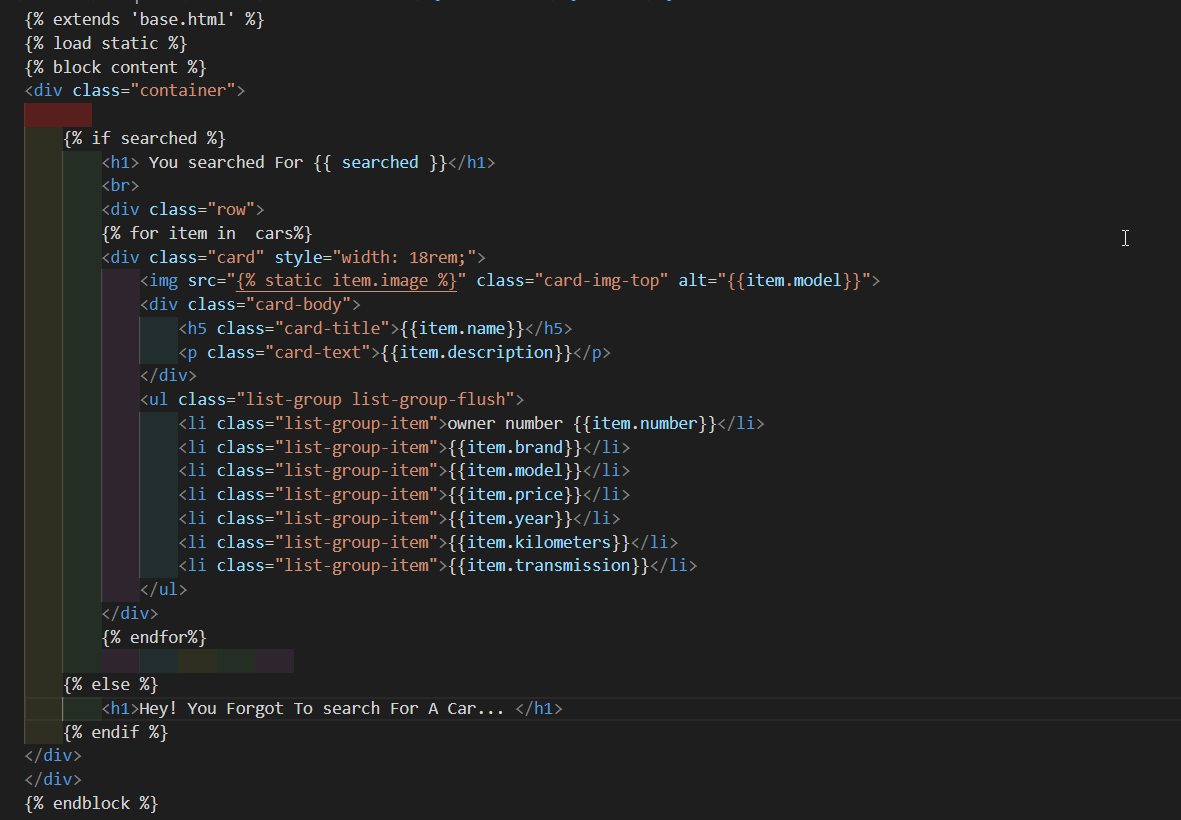
Description automatically generated

Mapped search bar

Text

Description automatically generated

Creating search bar html



Added search bar for navigation

Text

Description automatically generated

Website phase two done

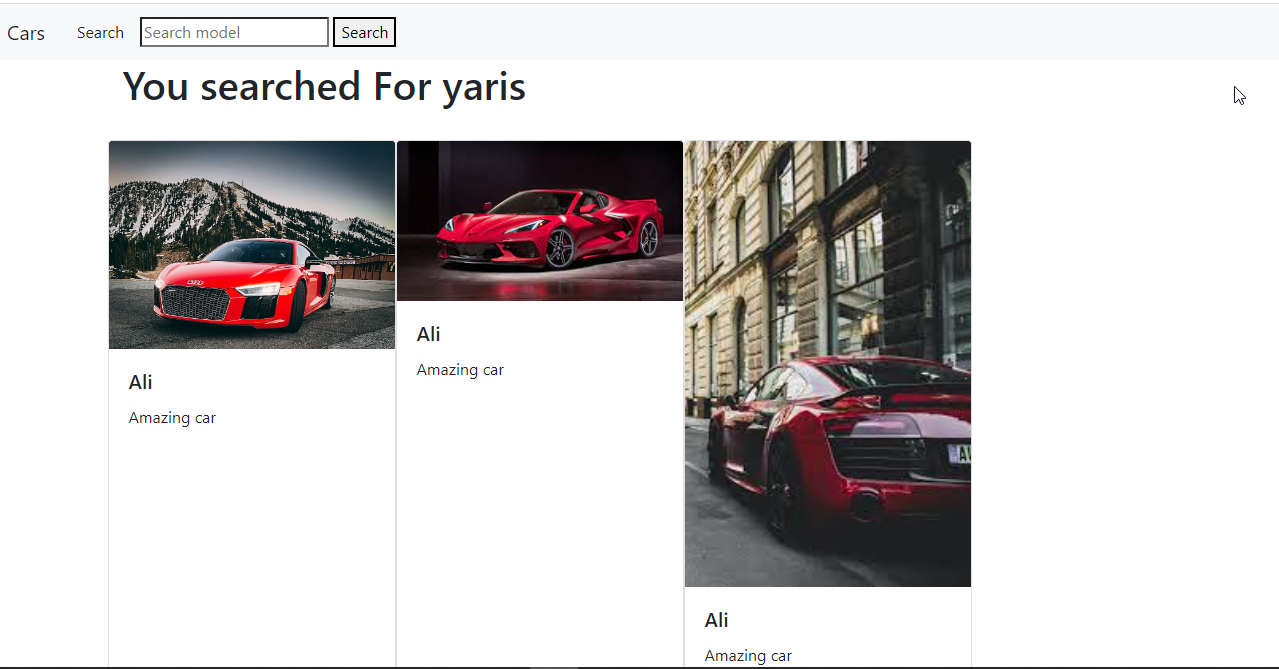


Figure Search working

## Add Car

Created forms file and created forms class

Text

Description automatically generated

Figure Add Car Form

Create a function file and created a image uploader function

Graphical user interface

Description automatically generated with medium confidence

Figure Image Uploader Function

Add Car views

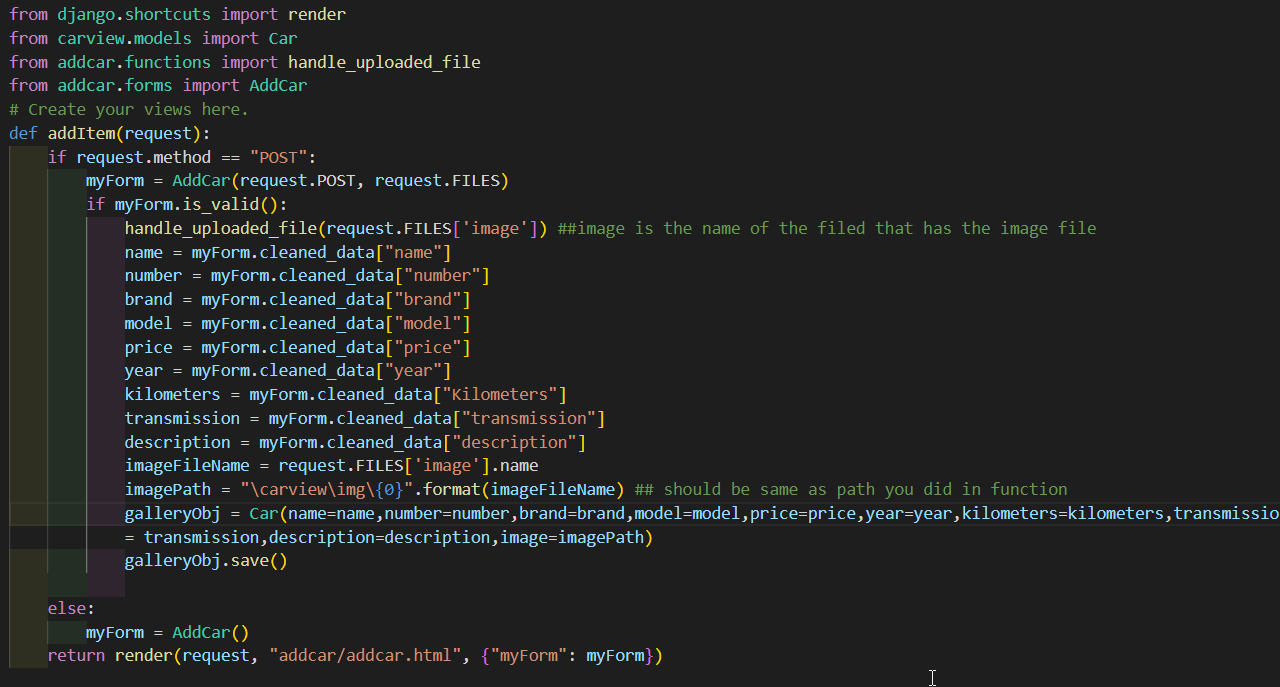


Figure Add Car view

Add Car URL

Text

Description automatically generated

Figure Add Car URL

Add Car to settings and mapped in URLS pattern

Text

Description automatically generated

Figure Add Car in installed apps

Text

Description automatically generated

Figure URL Mapping

Installed crispy

Text

Description automatically generated

Figure installing crispy

Add car page

Text

Description automatically generated

Figure Add car page

Add car Page

Graphical user interface, application, Word

Description automatically generated

Figure Final Page

I create a page for search because the buttons functions were overlapping.

## Admin

Creating a super user

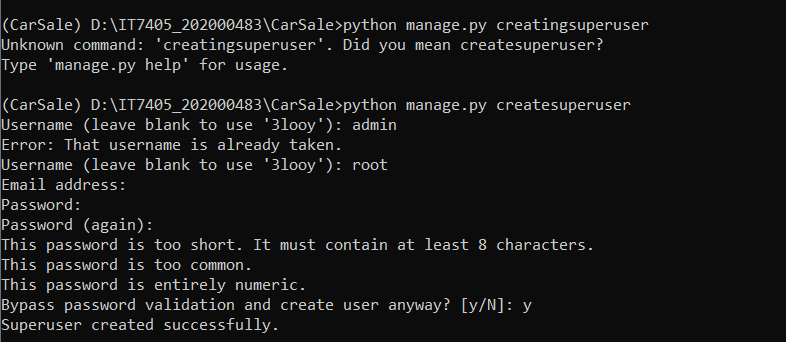


Figure Super User

Super user login and page

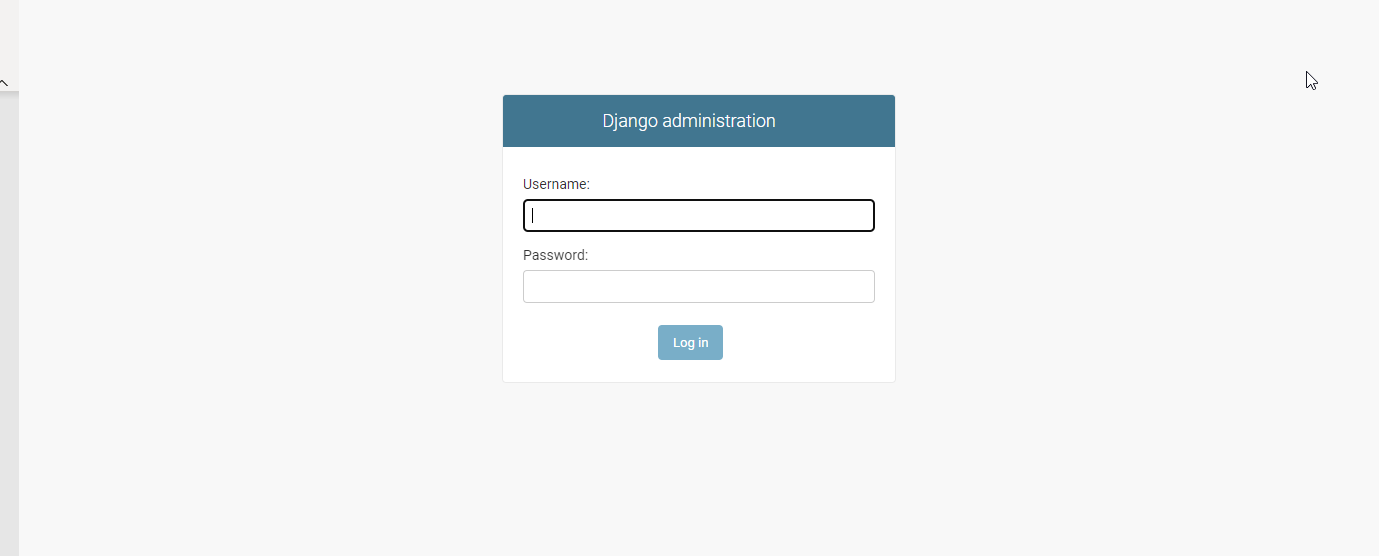


Figure Admin Login

User is root

Password is 1234

Graphical user interface, application, website

Description automatically generated

Figure Admin Page

Car Data management

Text

Description automatically generated

Figure Code

Graphical user interface, application, chat or text message

Description automatically generated

Figure Updated admin page

## Login

Creating a user application

Text

Description automatically generated

Settings

Text

Description automatically generated

Figure added users to the installed applications

Text

Description automatically generated

Figure Redirection Paths

URL patterns

Text

Description automatically generated

Figure Adding user URL Pattern

Adding user pages which will be created soon to the navigation bar

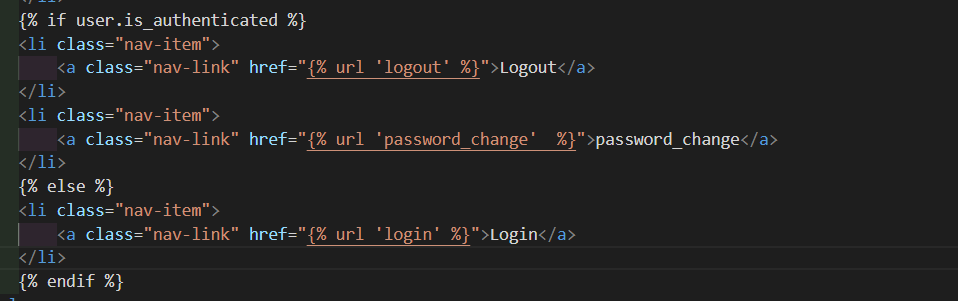


Figure updated nav

Designing the base

Graphical user interface, text

Description automatically generated

Figure Hello to the user logged in

HTML Pages

First, I created a Template folder inside it a sub folder registration which contain 3 different html pages

Graphical user interface, text

Description automatically generated with medium confidence

Figure Three pages

Text

Description automatically generated

Figure Page one

Text

Description automatically generated

Figure Page two

Text

Description automatically generated

Figure Page Three

## Registration

Creating a register application

Text

Description automatically generated with medium confidence

Figure 65 Register application

Register view

Text

Description automatically generated

Figure Register view

Register URL and creating python url.py file

Text

Description automatically generated

Figure URL code

Created a register html page

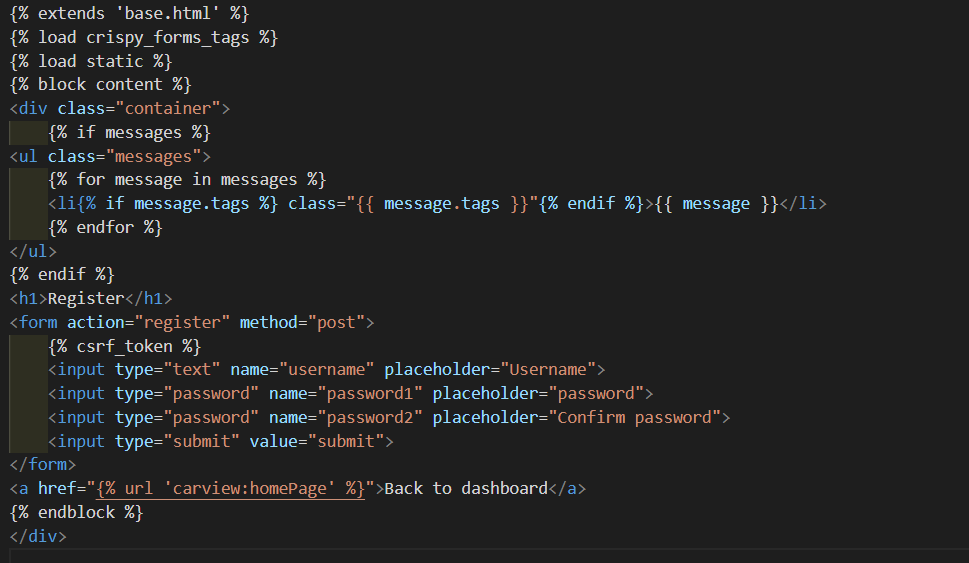


Figure register HTML page

Added registration to the nav bar

Text

Description automatically generated

Figure Registration in nav bar

Add registration to settings and URLs pattern

Text

Description automatically generated

Figure installed application register

Graphical user interface, text

Description automatically generated

Figure register URL path

# Reflection

In this project I learned a lot of valuable things. We started by finding a problem to create a solution for using Django framework and mongo dB as a database. I started the project by creating the database and running CRUD operations to populate it. After that I started working my website which include the following functions search, add, login, change password and register. Working on the project I faced a lot of challenges such as time manage due to working on multiple projects at the same time, which made me stress a lot. Furthermore, there are not a lot of documentations about Django and mongo dB working hand to hand together, so when ever I ran into a problem, I had to debug it which was very time consuming but improved my debugging skills. Designing the website was challenging because the focus of this course is backend not frontend, so learning boot strap and implementing it was very challenging. What I would do better is better organize my time and manage my stress levels by taking breaks, having walks. Finally, I learned a lot from this project that will help me in the long run.