

Report

Coursework: Part 1

Car Dealership

U2026971

<https://github.com/samwright000/ccse-cardealer/tree/main/cardealershippython>

Content

Report	0
Content	1
High-level Business Requirements	2
Functional and Non-functional Requirements	3
Functional	3
Customer	3
Administrator	3
Non-functional	4
How The High-level Business Requirements Were Mapped To Functional And Non-functional Requirements	5
Explain How These Requirements Were Delivered Using The Agile Software Development Methodology	7
Software Architecture and High Level Data Flow	9
Use Case Diagram	9
Model View Controller Diagram	10
Entity Relation Diagram	11
Class Diagram	12
Evidence That The Prototype Meets The Business Requirements	13
Proving Business Requirement 1	13
Proving Business Requirement 2 and 5	14
Proving Business Requirement 3, 4 and 5	19
Proving Business Requirement 6	22
Proving Business Requirement 7 and 8	23
References	27

High-level Business Requirements

1. Customers should be able to browse a selection of cars
2. Customers should be able to purchase car then display closest dealer
3. Customers should be able to apply for finance
4. Customers should be able to upload personal documents
5. Customers should only be able to access their account
6. For admin accounts multi factor authentication should be enabled.
7. Admin should be able to assess the internal administration module via a button on webpage
8. The finance team can keep track of inventory, number of cars sold, customer finance application and customer documents.

Functional and Non-functional Requirements

Functional

Customer

1. Should be able to view cars and their details (model, series, price, picture, description.) which are in stock
2. Shouldn't be able to view cars which aren't in stock
3. Should be able to sign up for an account
4. Should be able to sign in to just their account
5. Should be able to add available cars to their baskets
6. Should be able to remove cars from their basket
7. Should be able to select a car from their basket for purchase
8. Once a car is selected for purchase can choose whether to purchase with one time payment or apply for a finance plan.
9. Should display the closest dealership when making a one time payment
10. Should be able to enter card number, CVV, card expiration date, delivery address including house number, road name, city, county and postcode when purchasing.
11. Should be able to view the price of the car when buying
12. Should be able to upload pdf and pictures to their account
13. Should be able to remove uploaded documents
14. Should be able to view all the order and its related information (order date, delivery address, price of the car, how much the car is paid off, the status of the order, car image) they have bought and are currently buying.
15. Should only be able to make payments to cars which aren't yet paid off which have been approved.

Administrator

16. Should be able to view car information such as id, series, model, description, price, image, stock, number of cars sold.
17. Should be able to add a car to the database
18. Should be able to edit the current cars
19. Should be able to remove cars

20. Should be able to view all users and their orders information
21. Should be able to edit a users details
22. Should be able to edit the status of an order from pending, ongoing and paid.
23. Admins should be able to navigate to an administration module which is only available to assigned users roles.

Non-functional

1. Multiple users should be able to access the service at one
2. Customers should only be able to access their own data.
3. The system should be available at all times except when under maintenances
4. The system should store sensitive information securely
5. To access the admin account an email two factor authentication must occur
6. The system should be able to store many types of user data from text to pdf and images.

How The High-level Business Requirements Were Mapped To Functional And Non-functional Requirements

Below is a table which shows how the business requirements map to the function and non-functional requirements.

High-level business requirement	Functional requirements	How?
1	1, 2	Function requirement 1 allows users to view the available cars for purchase. Requirement 2 results in cars which aren't available and not possible for purchase.
2	3, 4, 5, 6, 7, 8, 9, 10, 11	These functional requirements all enable business requirement 2 to be carried out. A user needs an account and needs to be able to access that account to make purchases. A user needs to be able to add cars to their basket and then select them. The users need to be able to input their bank details and upload documents for the payment to be approved.
3	8, 10, 11, 15	These function requirements are also needed for when a user wants to apply for finances. They need to be able to upload their documents which then are reviewed by the financial department.
4	12, 13	As part of business requirements to apply for finance they also need to be able to upload relevant documents. This is a partial repeat of business requirements 2 and 3, it is explicitly mentioned to do this via uploading documents.
5	14, 13	To make sure the company remains GDPR (GDPR, 2018) the system needs to make sure user data is available for them to access. They also need to be able to remove data.
7	16, 17, 18, 18, 20, 21	The administration module is a large section of the project. The admins need to be able to manage the cars in ways outlined in function requirements.
8	16, 20, 22	The finance team needs to be able to access users data so they are able to make a judgement if finance payment is allowed. They also need to be able to keep track of the orders and stock. This is outlined in the function requirements

Figure 1.1 - Mapping Business Requirements to Functional Requirements

High-level business requirement	Non Functional requirements	How?
1	1, 3,	<p>We want many customers to be able to browse the website and use the services at once, therefore one of our functional requirements is that many users should be able to access the system.</p> <p>As well as many people being able to access the website, we want people to be able to access it at any time. Not everyone looks at cars at the same time and if there is hope to grow the company internationally different countries have different time zones.</p>
2	1, 3	<i>Same as above</i>
3	1, 3	<i>Same as above</i>
4	1, 3, 4, 6	<i>Same as above.</i> Additionally, the user needs to be able to enter text for form inputs but also needs to be able to upload documents relating to their finances. Therefore the system needs to be able to store this type of data. The data the user will be uploading is going to be sensitive, private financial data as well as passwords, therefore this needs to be held securely. One example of this is hashed passwords.
5	2, 3, 4, 6	The user should only be able to access their data. This is so that good security practices are followed but also so it complies with GDPR (GDPR, 2018) making sure users information is available when they need it.
7	3,	Admins may need to access the website at all times of the day therefore it should always be available.
8	3, 5	To maintain security, the administrator users have to use two factor authentication via their email address. This email address is the one they signed up with and use to sign in with.

Figure 1.2 - Mapping Business Requirements to Non-Functional Requirements

Explain How These Requirements Were Delivered Using The Agile Software Development Methodology

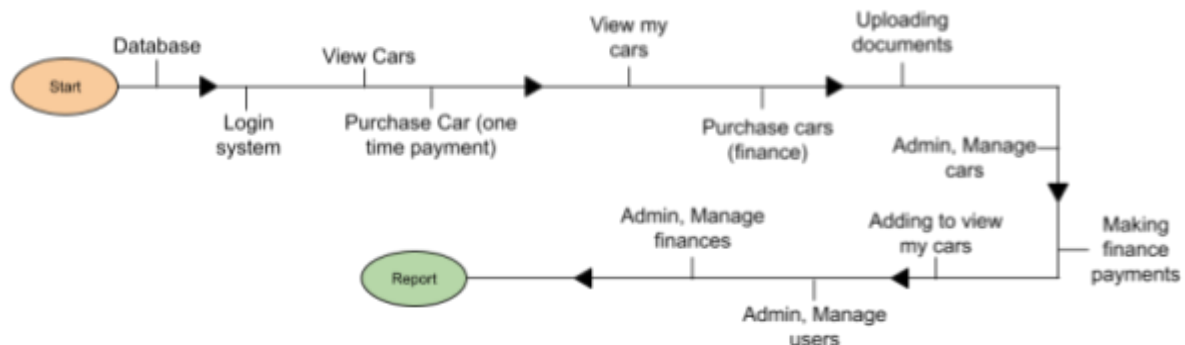


Figure 1.3 - Sprint Order Timeline

An agile software development is an iterative process that puts flexibility, collaboration and customer satisfaction as its priorities Codecademy (2019). Agile is both incremental and iterative and is made up of sprints. A sprint is a smaller, more manageable chunk of a problem (PaulPetrus, 2020). A sprint is usually a short period of time between one week and a month. For my project they were usually a few days long to a week. Each time a sprint starts it has planning, implementation then a review and testing (PaulPetrus, 2020). This is how I developed each of my sprints. Unlike a waterfall, it does this for each of its sprints rather than the whole project at large.

When I was developing my project, I broke it down into sprints. See figure 1.3 for the order in which the sprint was completed. Each sprint followed the processes outlined in figure 1.5 below. I also partially created user stories as seen in figure 1.4 below.

I.	Title	A.	S.	A.	Tags
9	👑 As a dealer, I want to be able to edit the configuration options that customers can pick for cars, so that the appropriate options are available.	---	👑	C	As a dealer
4	👑 As a dealer, I want to view all the customers past and future payments, so that I can keep track of them	👑	👑	C	As a dealer
3	👑 As a dealer, I want to manage the current cars being built, so that I can notify the customer	👑	👑	C	As a dealer
5	👑 As a dealer, I want to be able to add/remove cars from being viewed on the dealership website, so that old cars can be removed and new can be added	👑	👑	C	As a dealer
6	👑 As a dealer, I want to be able to edit the availability of cars, on that customers know what is and isn't in stock	👑	👑	C	As a dealer
1.	👑 As a customer, I want to be able to view the future payments for my car(s) so that I can keep track of them	👑	👑	C	As a customer
8	👑 As a customer, I want to be able to edit the options of the available cars, to make it suits me.	👑	👑	C	As a customer
7	👑 As a customer, I want to be able to view the available cars so that I can pick one I want	👑	👑	C	As a customer
2	👑 As a customer, I want to be able to make payments in line with the financial agreement.	👑	👑	C	As a customer
1	👑 As a customer, I want purchase the best car for me so that I can have it as soon as possible.	👑	👑	C	As a customer

Figure 1.4 - User stories

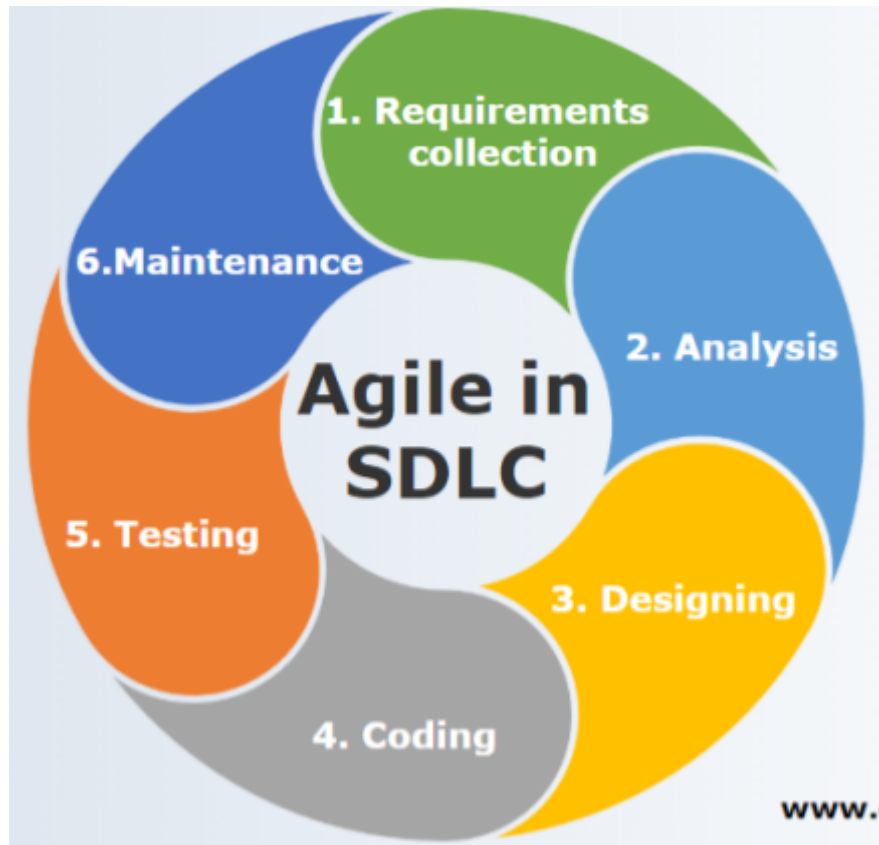


Figure 1.5 - (Chaudhary, 2019)

Software Architecture and High Level Data Flow

Use Case Diagram

See below in figure 2.1 a use case diagram. A Use Case Diagram (UCD) describes a set of actions which we call cases that a system should perform in collaboration with external users of the system they are known as actors (Kirill Fakhroutdinov, 2014). Each use case should provide valuable results to the actor or stakeholder.

These diagrams are both behaviour and structure diagrams (Kirill Fakhroutdinov, 2014).

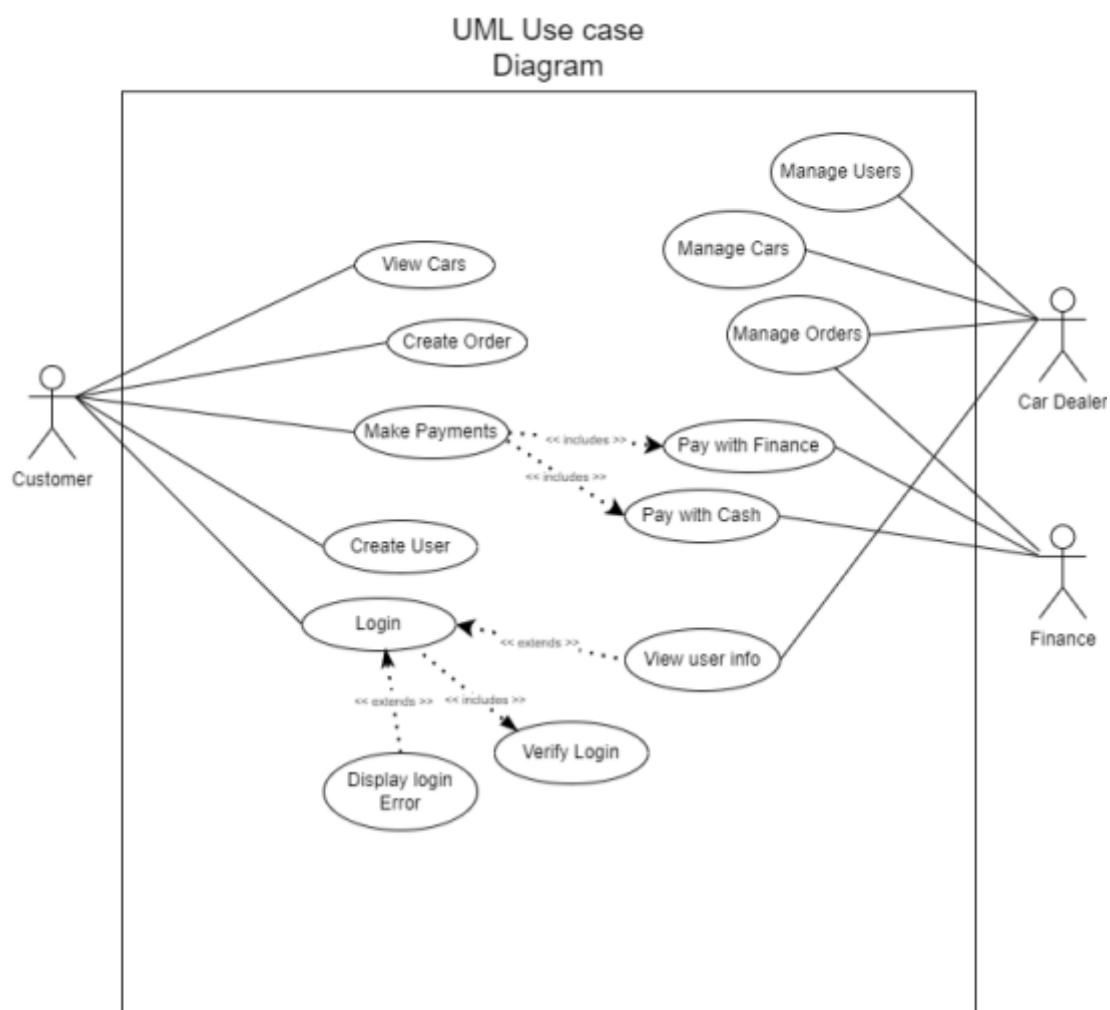


Figure 2.1 - Use case diagram for car dealer service

Model View Controller Diagram

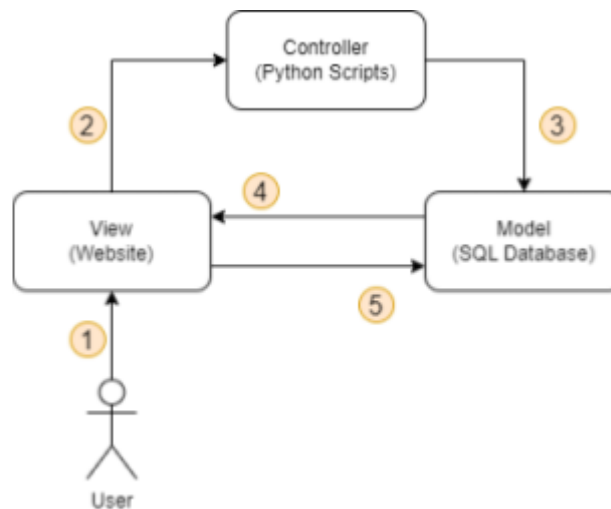


Figure 2.2 - Model View Controller Diagram of dealership service

1. The view is the website. The user interacts with the website to access the data and services the system provides
2. The view notices the controller of a particular event. In this case, the website form or buttons when pressed results in python scripts being run. Depending on the request we then move to 3.
3. The controller updates the model. An example of this is when the user adds a car to the basket, and this results in a row being entered into the users basket table.
4. The model alerts the view that there has been a change. An example of this is producing a confirmation message to the user of the action they have committed.
5. After step 4 the view may grab more data from the model to update itself. An example of this is when a user buys a car the “/mycars” page is updated to show the user’s new car.

The diagram and descriptions above have been aided by (Martin, 2019).

Entity Relation Diagram

Below is the entity relationship diagram, which shows how the tables in the database interact with each other.

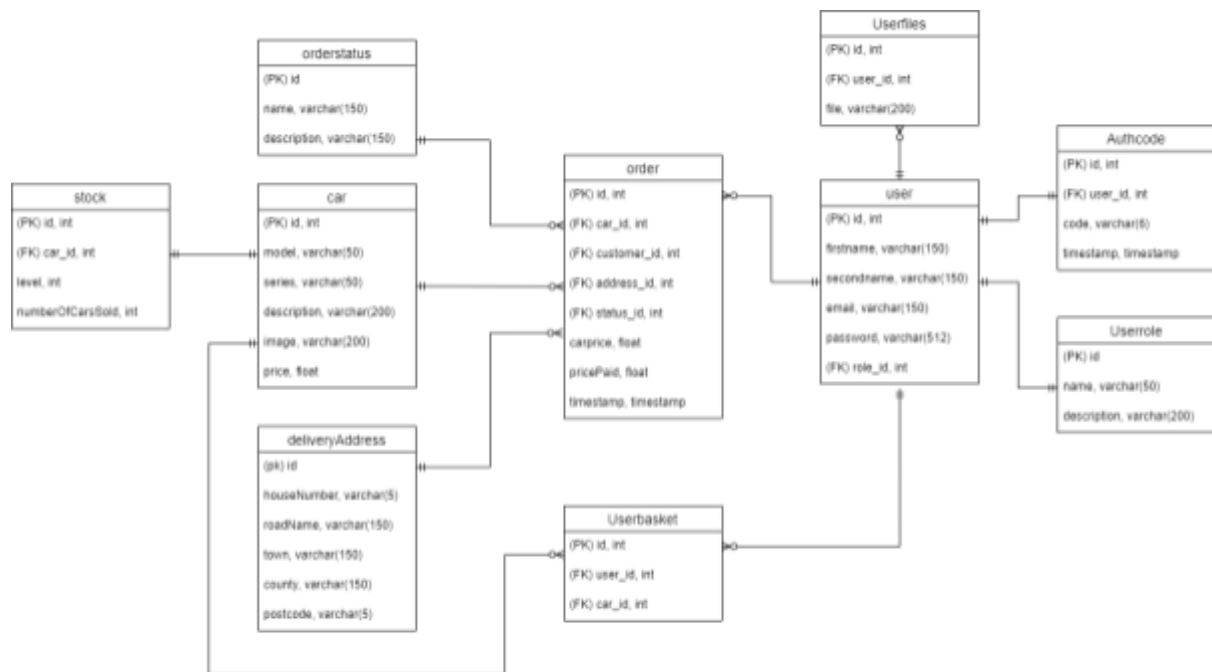


Figure 2.3 - ER Diagram for car dealer SQL database

Class Diagram

This class diagram doesn't directly show how the system is made up. This is because I used flask and SQLAlchemy which are python libraries therefore the make of the system is slightly different. This is to demonstrate what it would look like if I had more time and would have used classes more.

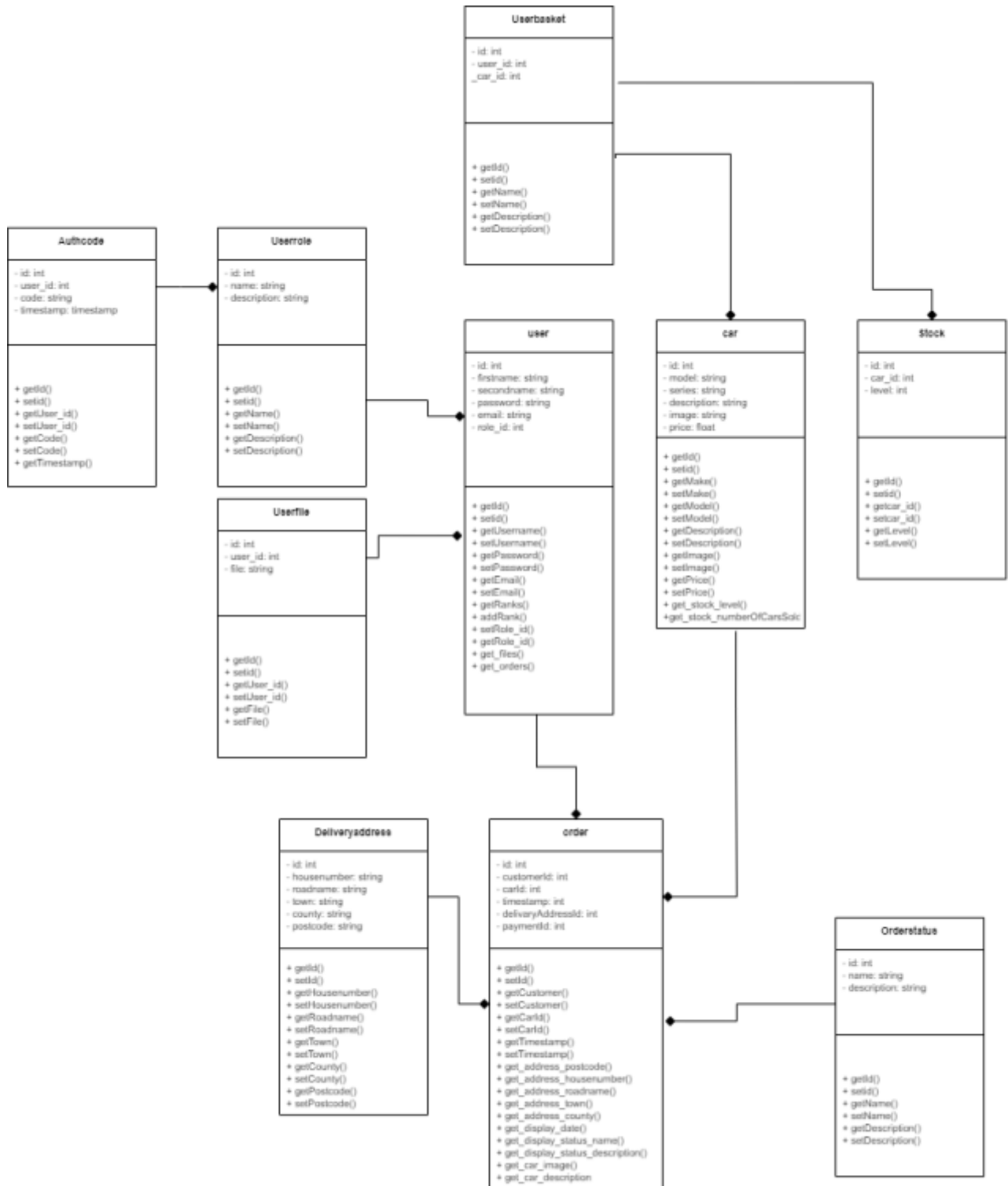


Figure 2.4 - Class Diagram for the system

Evidence That The Prototype Meets The Business Requirements

Proving Business Requirement 1

In figure 3.1 below, shows the homepage of the website, customers are able to view the available cars. You have to create an account and login to be able to add a car to the basket.

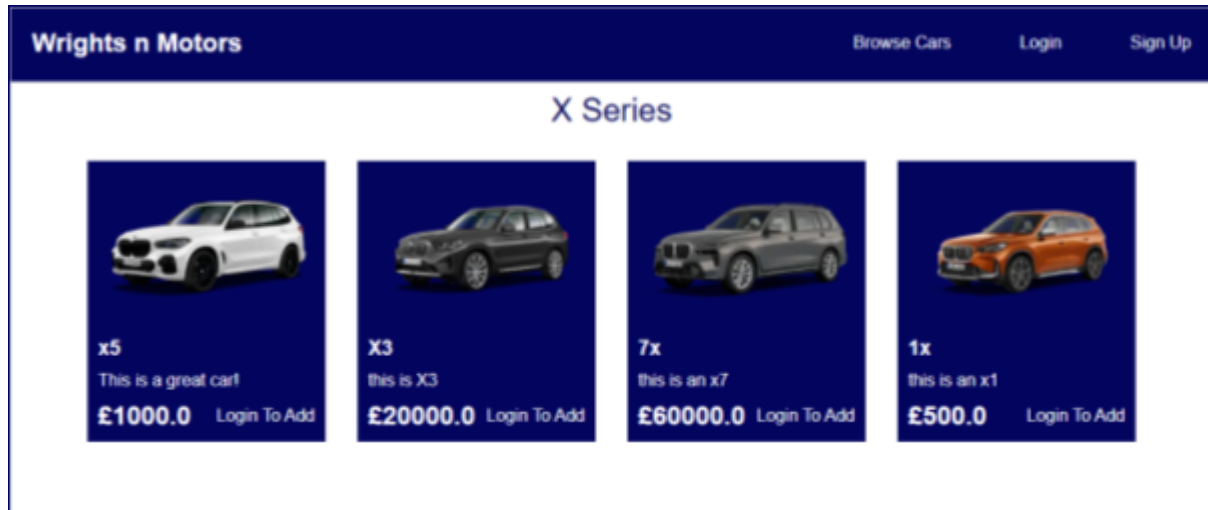
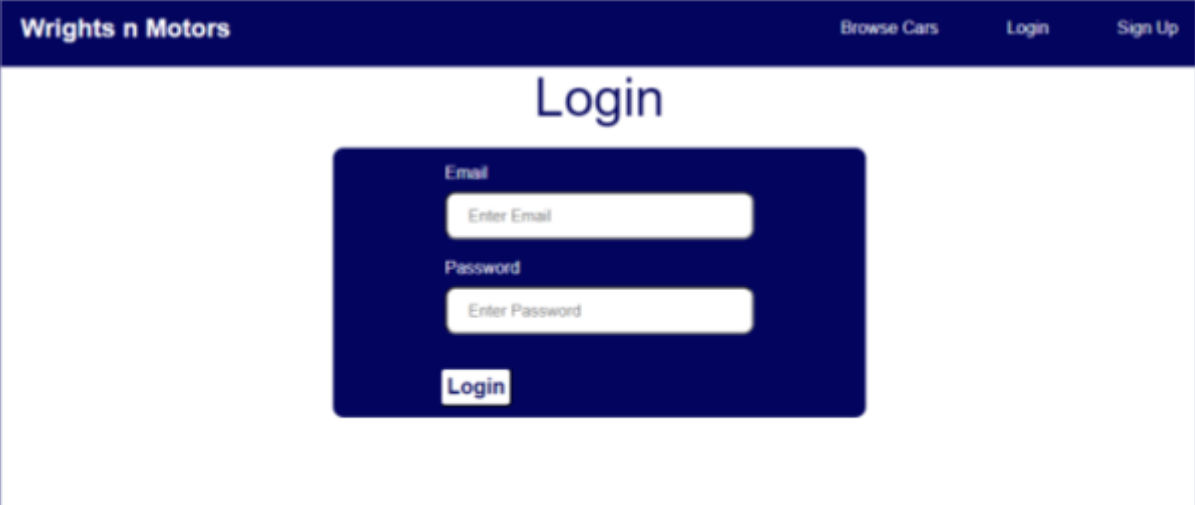


Figure 3.1 - Home page '/'

Proving Business Requirement 2 and 5

For a customer to purchase a car must first add it to their basket but to do that they need to login.



The screenshot shows the 'Wrights n Motors' website header with links for 'Browse Cars', 'Login', and 'Sign Up'. The main heading is 'Login'. Below it is a dark blue form box containing two input fields: 'Email' with the placeholder 'Enter Email' and 'Password' with the placeholder 'Enter Password'. A 'Login' button is at the bottom of the form.

Figure 3.2 - login page "/login"

If a customer doesn't have an account they are required to sign up. See in figure 3.3.



The screenshot shows the 'Wrights n Motors' website header with links for 'Browse Cars', 'Login', and 'Sign Up'. The main heading is 'Sign Up'. Below it is a dark blue form box containing five input fields: 'First name' (placeholder 'Enter First Name'), 'Second name' (placeholder 'Enter Second Name'), 'Email' (placeholder 'Enter Email'), 'Password' (placeholder 'Enter Password'), and another 'Password' field (placeholder 'Confirm Password'). A 'Submit' button is at the bottom of the form.

Figure 3.3 - sign up page "/sign-up"

Once a user has signed up, they will automatically be logged in.

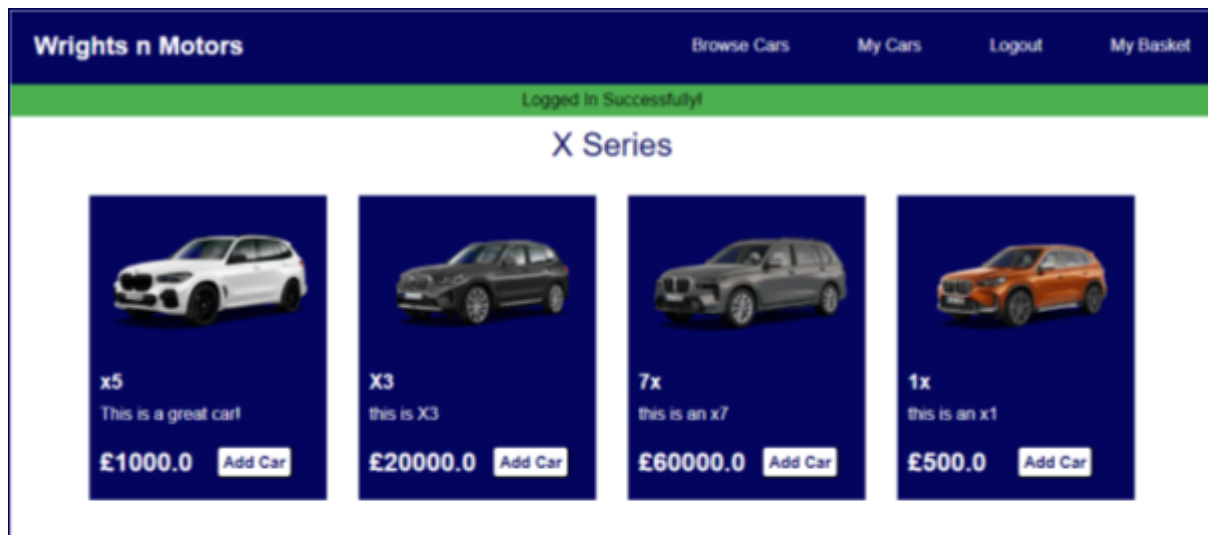


Figure 3.4 - Home page once user logged in

When the user logs in they are displayed with a successful message, they are now able to add cars to their basket.



Figure 3.5 - Adding car to basket

When the user adds a car to their basket they are displayed with a confirmation message. The user can then click the "my basket" on the taskbar and see their basket.



Figure 3.6 - Users basket

Once an item is in the users basket, they can remove it or then proceed to checkout.

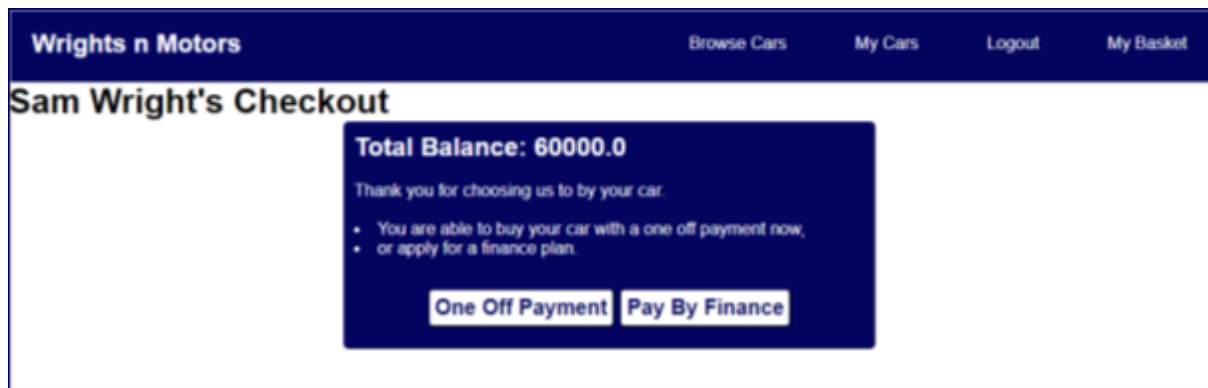


Figure 3.7 - Checking out car selected

A customer is then able to choose between one time payment or apply for finance. We will first do one off payment.

Wrights n Motors

[Browse Cars](#)

[My Cars](#)

[Logout](#)

[My Basket](#)

Sam Wright's Payment

Total: 60000.0

Bank Information

Bank Card

Enter bank card

CW

CVV

Card Expire Month

MM

Card Expire Year

YY

Delivery Address

House Number

Enter House Number

Road Name

Enter Road Name

City

Enter city

County

Enter county

Postcode

Enter postcode

Make Payment

Figure 3.8 - Payment interface

The customer enters their bank details.

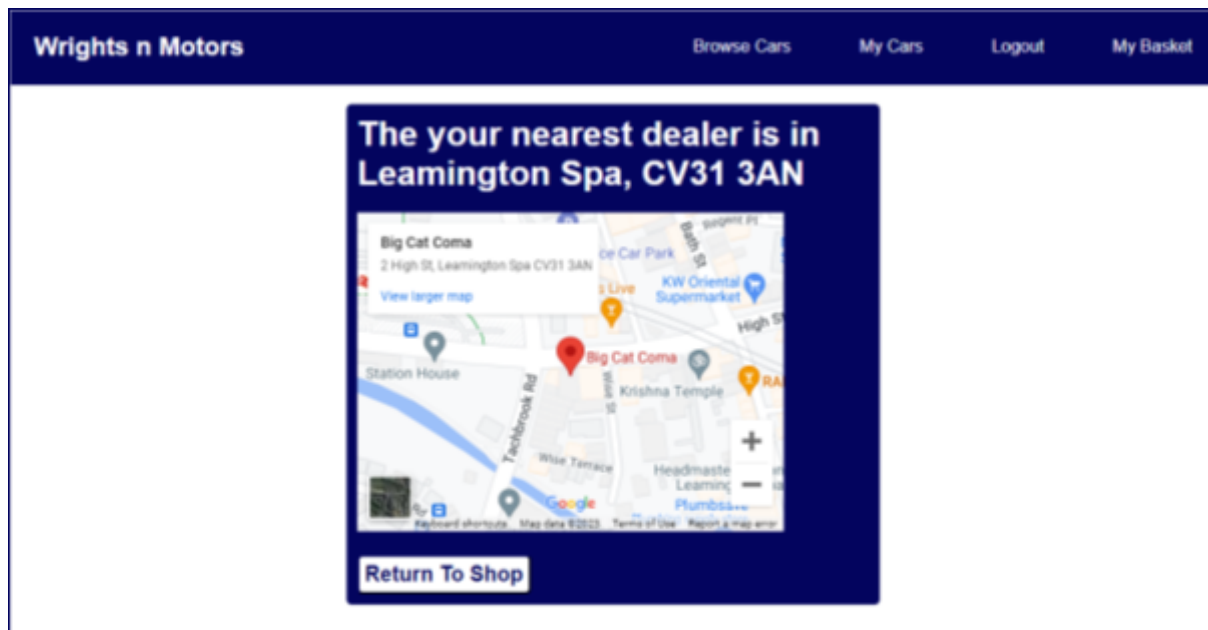


Figure 3.9 - Closest car dealership

The customer is then displayed the closest dealer. We can then go to the “my cars” tab and view our order

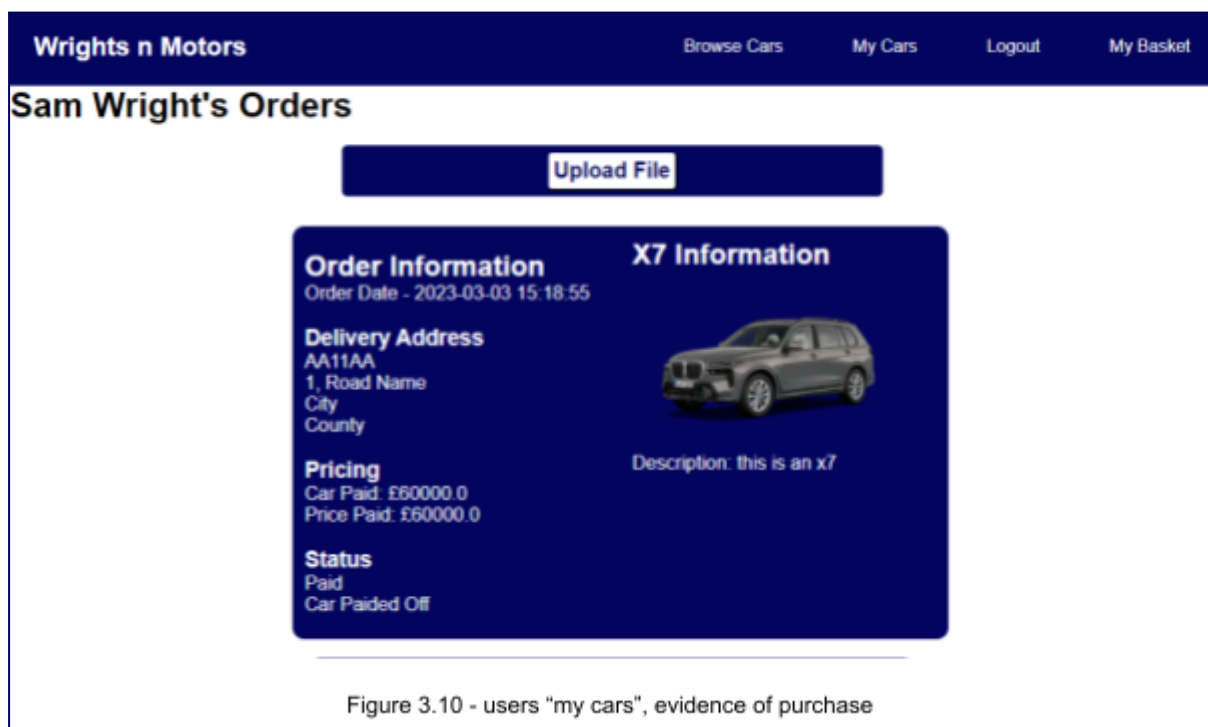


Figure 3.10 - users "my cars", evidence of purchase

Proving Business Requirement 3, 4 and 5

If the user wants to apply for finance instead of clicking one off payment in figure 3.7 they can instead click “Pay By Finance”. They then processed to enter bank details as in figure 3.8. Once that is done they are then displayed to enter the personal documents

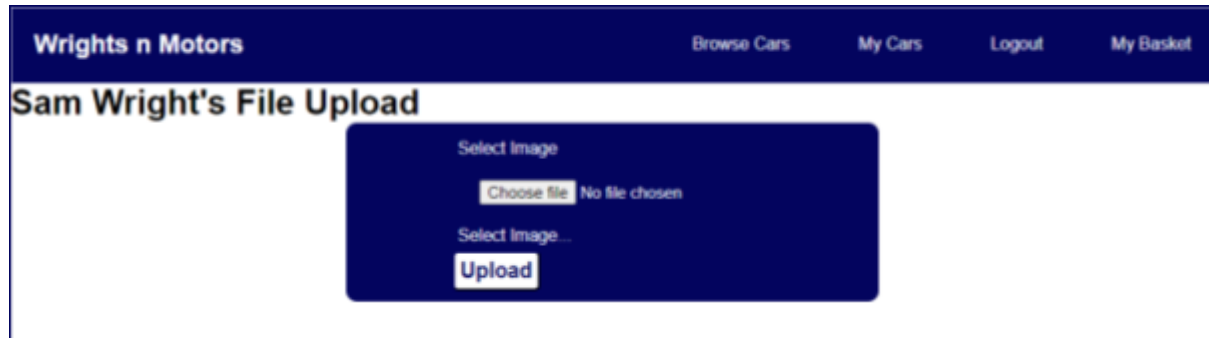


Figure 3.11 - “/uploadfiles”



Figure 3.12 - Confirmation of upload

The user can then view and remove their uploaded documents on the “my cars” page

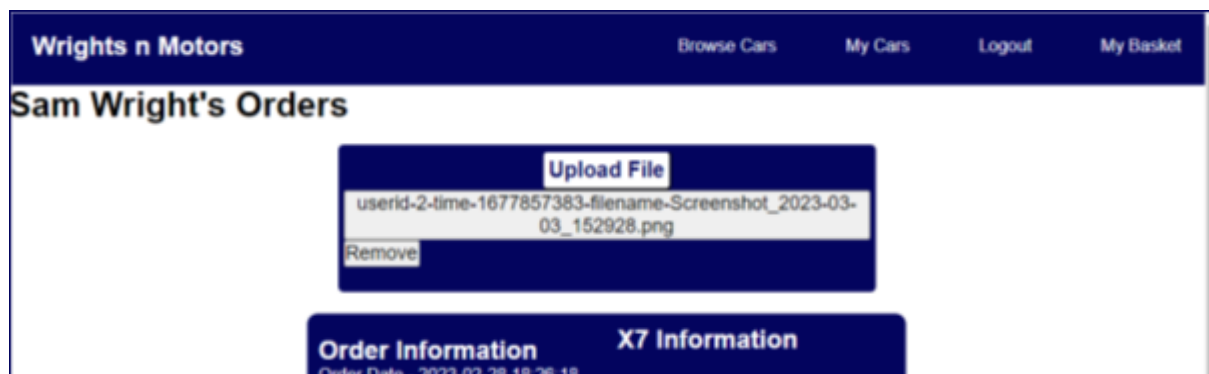


Figure 3.13 - Users uploaded documents

The car will then be awaiting approval.

Order Information	X3 Information
Order Date - 2023-03-03 15:27:34	
Delivery Address AA22AA HOUSE, Road City County	
Pricing Car Paid: £20000.0 Price Paid: £0.0	Description: this is X3
Status Pending Pending Approval	

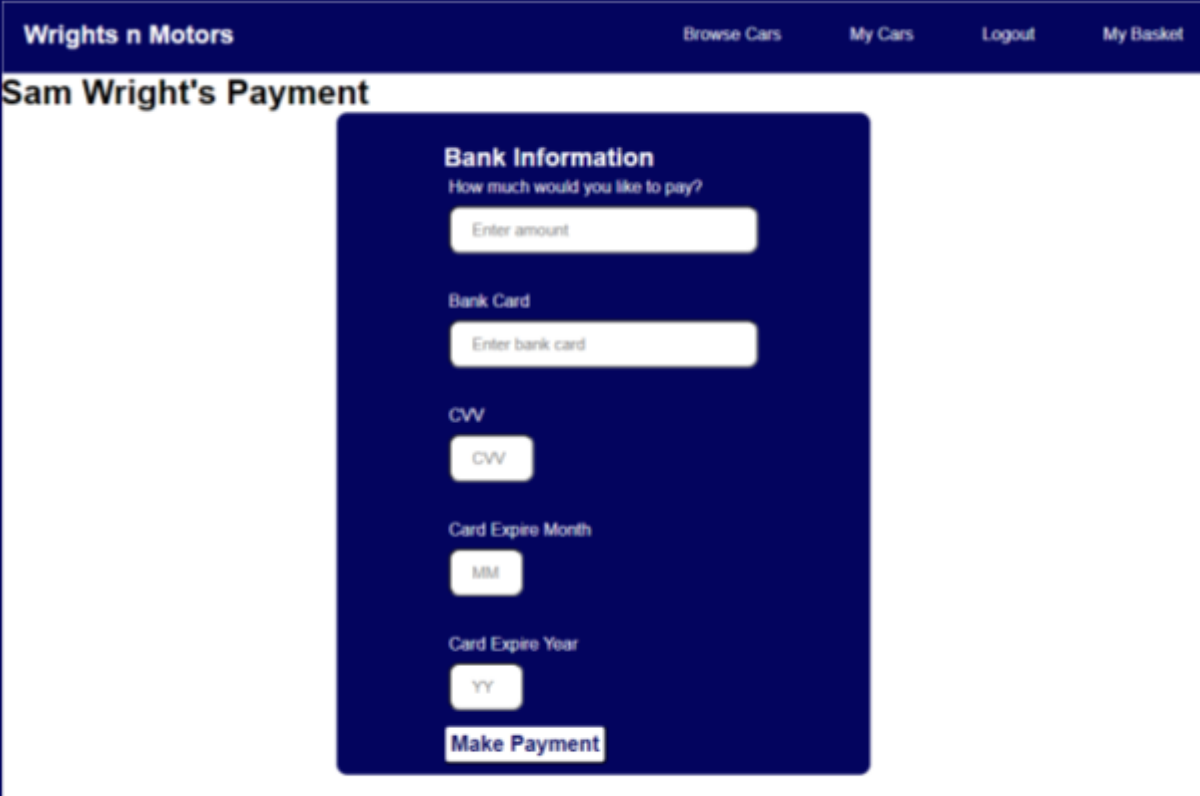
Figure 3.14 - Car Order

Once the finance team has then approved the car the status will change and then the option for payment will be allowed.

Order Information	X3 Information
Order Date - 2023-03-03 15:27:34	
Delivery Address AA22AA HOUSE, Road City County	
Pricing Car Paid: £20000.0 Price Paid: £0.0	Description: this is X3
Status Ongoing Car Currently Being Paid Off	
Make Payment	

Figure 3.15 - Ongoing order

The customer is then able to make a payment.



Wrights n Motors Browse Cars My Cars Logout My Basket

Sam Wright's Payment

Bank Information

How much would you like to pay?

Bank Card

CVV

Card Expire Month

Card Expire Year

Figure 3.16 - Making a payment for finance

The user is then able to choose how much to pay and enters the details.



Order Information

Order Date - 2023-03-03 15:27:34

Delivery Address

AA22AA
HOUSE, Road
City
County

Pricing

Car Paid: £20000.0
Price Paid: £20000.0

Status

Paid
Car Paid Off

X3 Information



Description: this is X3

Figure 3.17 - car paid off

Once a customer has paid off the car, they can no longer make payments and the status is updated.

Proving Business Requirement 6

When a user logs in using an administrator account they are also required to login using an code which is sent to their email.



The screenshot shows a web interface for 'Wrights n Motors'. At the top, there is a dark blue header with the text 'Wrights n Motors' in white. Below the header, the main content area is white and features the title 'Auth User' in a large, dark blue font. Centered below the title is a dark blue rectangular box. Inside this box, the word 'Auth' is written in small white text above a white input field containing the placeholder text 'Enter code'. Below the input field is a white button with the text 'Auth' in dark blue.

Figure 3.18 - Authorization code entering interface

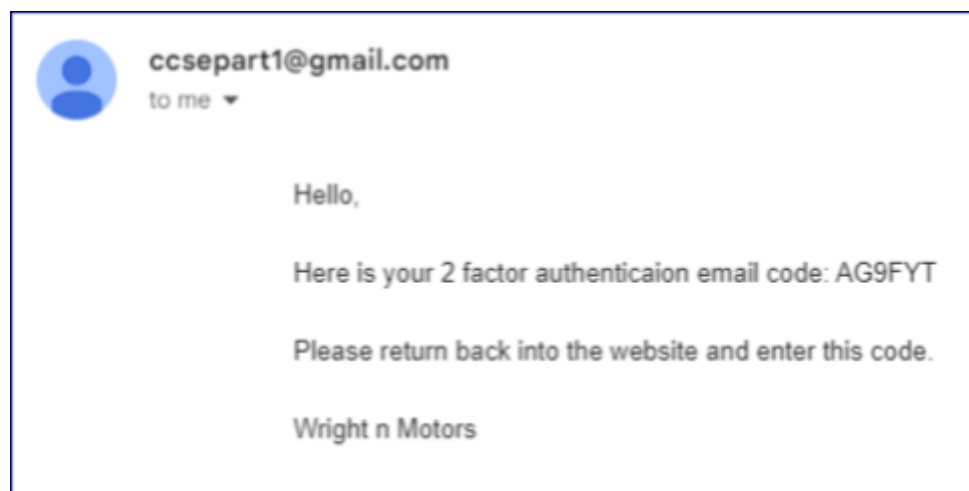


Figure 3.19 - Email for 2FA code

Proving Business Requirement 7 and 8

Once admin logs in they will be displayed an additional admin button on their taskbar.

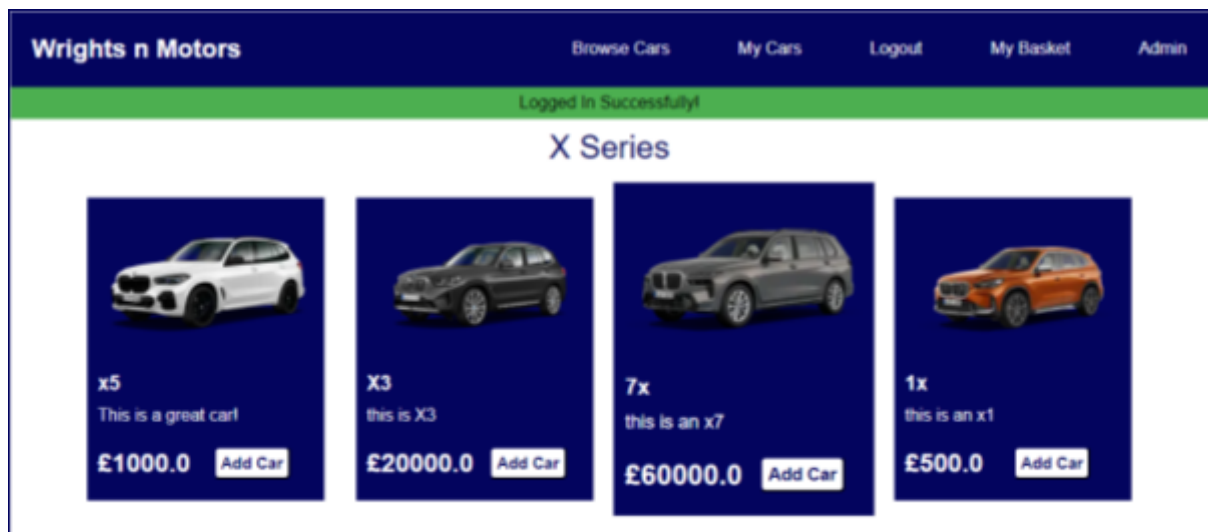


Figure 3.20 - Admin logged in

Once you click on the admin portal you will then be displayed with the admin dashboard.

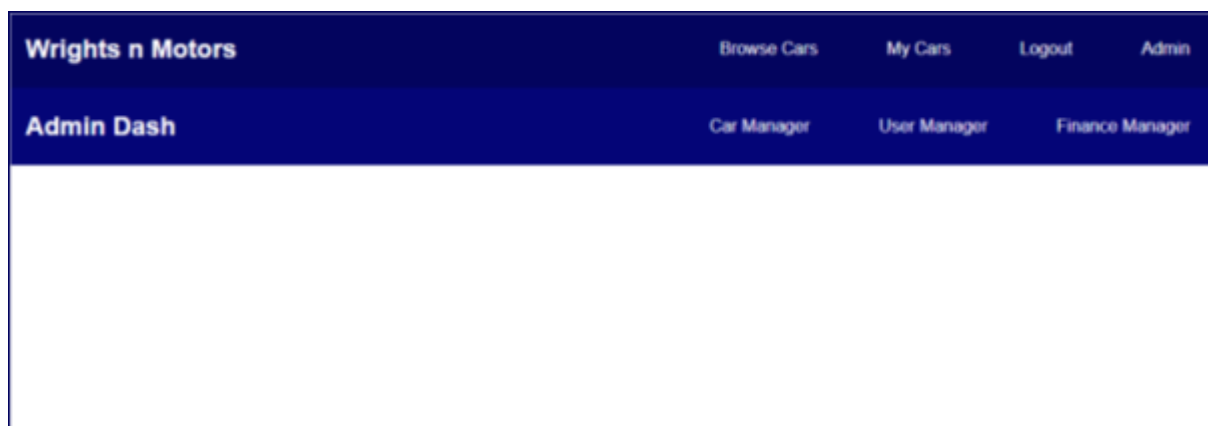


Figure 3.21 - Admin dashboard

The admin then clicked on the car manager, user manager or finance manager. Starting with cars.

Wrights n Motors
Browse Cars
My Cars
Logout
Admin

Admin Dash
Car Manager
User Manager
Finance Manager

CAR MANAGER



ID: 1
Series: x
Model: 5
Description: This is a great car!
Price £: 1000.0
Image: x5
Stock: 50
Number Of Cars Sold: 99



ID: 2
Series: X
Model: 3
Description: this is X3
Price £: 20000.0
Image: x3
Stock: 94
Number Of Cars Sold: 7



ID: 3
Series: 7
Model: x
Description: this is an x7
Price £: 60000.0
Image: x7
Stock: 53
Number Of Cars Sold: 8



ID: 4
Series: 1
Model: x
Description: this is an x1
Price £: 500.0
Image: x1
Stock: 196
Number Of Cars Sold: 4

CREATE CAR

Model

Series

Description

Image Link

Price

Stock

EDIT CAR

ID

Model

Series

Description

Image Link

Price

Stock

Amount Sold

REMOVE CAR

ID

Figure 3.22 - Car manager page “/carmanager”

On the User Manager you can see all the user uploaded files as well as all the users orders.

ID: 2 Name: Sam Wright
s@w.com

User's Files
userid-2-time-1677857383-filename-Screenshot_2023-03-03_152928.png

Order Information
ID: 8
Order Date - 2023-02-28 18:26:18

Delivery Address
CV33HS
70, Close Road
Leamington
Warwickshire

Pricing
Car Paid: £60000.0
Price Paid: £800.0

Status
Ongoing
Car Currently Being Paid Off

X7 Information

Description: this is an x7

Figure 3.23 - Users information

Edit User
ID

First name

Second name

Email

Password

Role

Figure 3.24 - At the bottom of the user manager you can edit the user information

The screenshot displays the 'Wrights n Motors' Admin Dash. The top navigation bar includes links for 'Browse Cars', 'My Cars', 'Logout', and 'Admin'. The secondary navigation bar shows 'Car Manager', 'User Manager', and 'Finance Manager'. The main content area is titled 'FINANCE MANAGER' and features a 'Statuses' section. This section contains a list of three statuses: ID 1 (pending), ID 2 (paid), and ID 3 (ongoing). Below this list is an 'Edit Status' form with two input fields: 'Order ID' and 'New Status ID', each with a placeholder text 'Enter order id' and 'Enter new status id' respectively. A 'Submit' button is located at the bottom of the form.

ID	NAME	DESCRIPTION
1	pending	pending approval
2	paid	Car paid off
3	ongoing	Car currently being paid off

Edit Status

Order ID

New Status ID

Figure 3.25 - Updating order status finance manager

The finance module allows the finance team to upload the status of an order changing a customer's access.

References

The videos which helped me to build the projects are:

Main website: <https://www.youtube.com/watch?v=dam0GPOAvVI>

Navbar: <https://www.youtube.com/watch?v=oLgtucwjVII&t=61s>

Using files: <https://www.youtube.com/watch?v=RYOQOfdN2qg&t=332s>

Using files: <https://www.youtube.com/watch?v=pPSZpCVRbvQ&t=1s>

Using files: <https://www.youtube.com/watch?v=sy1MNWt7om4>

Chaudhary, L. (2019). *Agile in SDLC | Characteristics and Functionality of Agile in SDLC*. [online] EDUCBA. Available at: <https://www.educba.com/agile-in-sdlc/> [Accessed 3 Mar. 2023].

Codecademy (2019). *Software Development Methodology: What is Agile? YouTube*. Available at: <https://www.youtube.com/watch?v=GzzkpAOxHXs> [Accessed 3 Mar. 2023].

GDPR (2018). *General Data Protection Regulation (GDPR)*. [online] General Data Protection Regulation (GDPR). Available at: <https://gdpr-info.eu/> [Accessed 2 Mar. 2023].

Kirill Fakhroutdinov (2014). *Use case diagrams are UML diagrams describing units of useful functionality (use cases) performed by a system in collaboration with external users (actors)*. [online] Uml-diagrams.org. Available at: <https://www.uml-diagrams.org/use-case-diagrams.html> [Accessed 2 Mar. 2023].

Martin, M. (2019). *MVC Tutorial for Beginners: What is, Architecture & Example*. [online] Guru99.com. Available at: <https://www.guru99.com/mvc-tutorial.html> [Accessed 2 Mar. 2023].

PaulPetrus (2020). *What Is The Difference Between Scrum and Sprint? | Long Version*. [online] www.youtube.com. Available at: <https://www.youtube.com/watch?v=UVx0oj32NVU&t=151s> [Accessed 3 Mar. 2023].