Before writing any code, we decided to all agree on a common design for our website and determine our workflow so we could collaborate and synchronise between each other. Our previous module incorporated Miro as a way to gather ideas and visualise them, so we all agreed it would be a good tool to settle on a core idea for the website. As per the specification our website should feature;

- animation

- validation

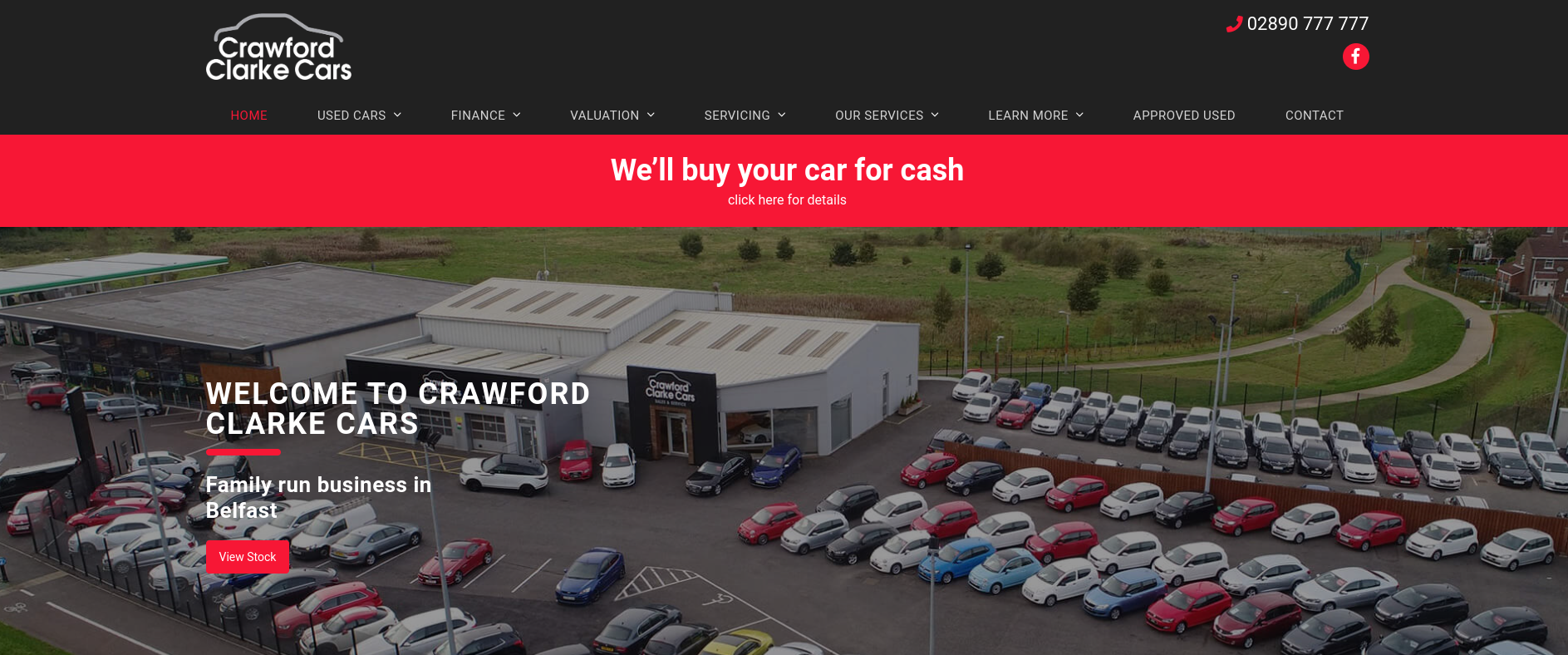
- cookies/LocalStorage

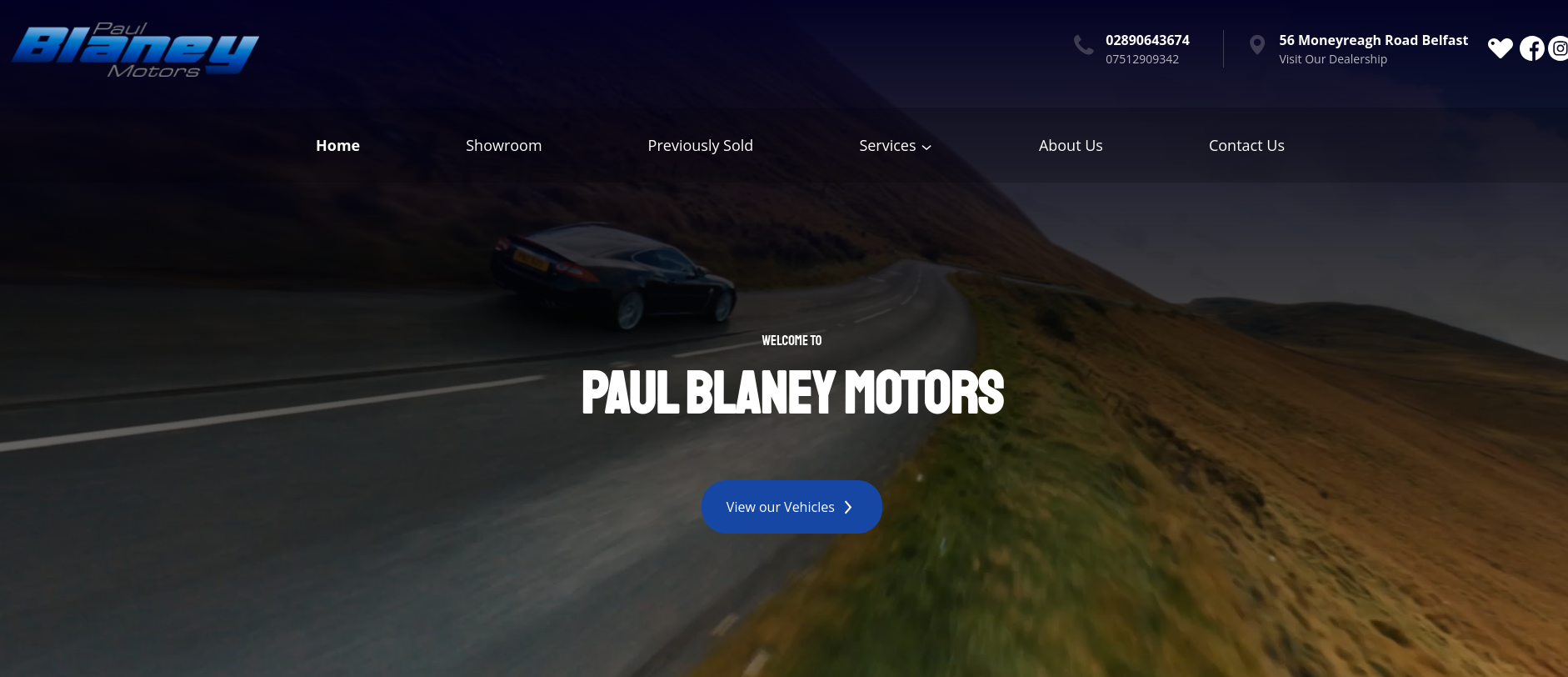
What types of websites make use of these elements? We began researching and determined the best use of these elements would be featured in an online marketplace, a storefront that would be responsible for handling customer details such as location, email or phone numbers. It would also be expected to store and process payment information. Broadly speaking, we can feature animation on the homepage, validation on the payment processor and cookies on the login page.

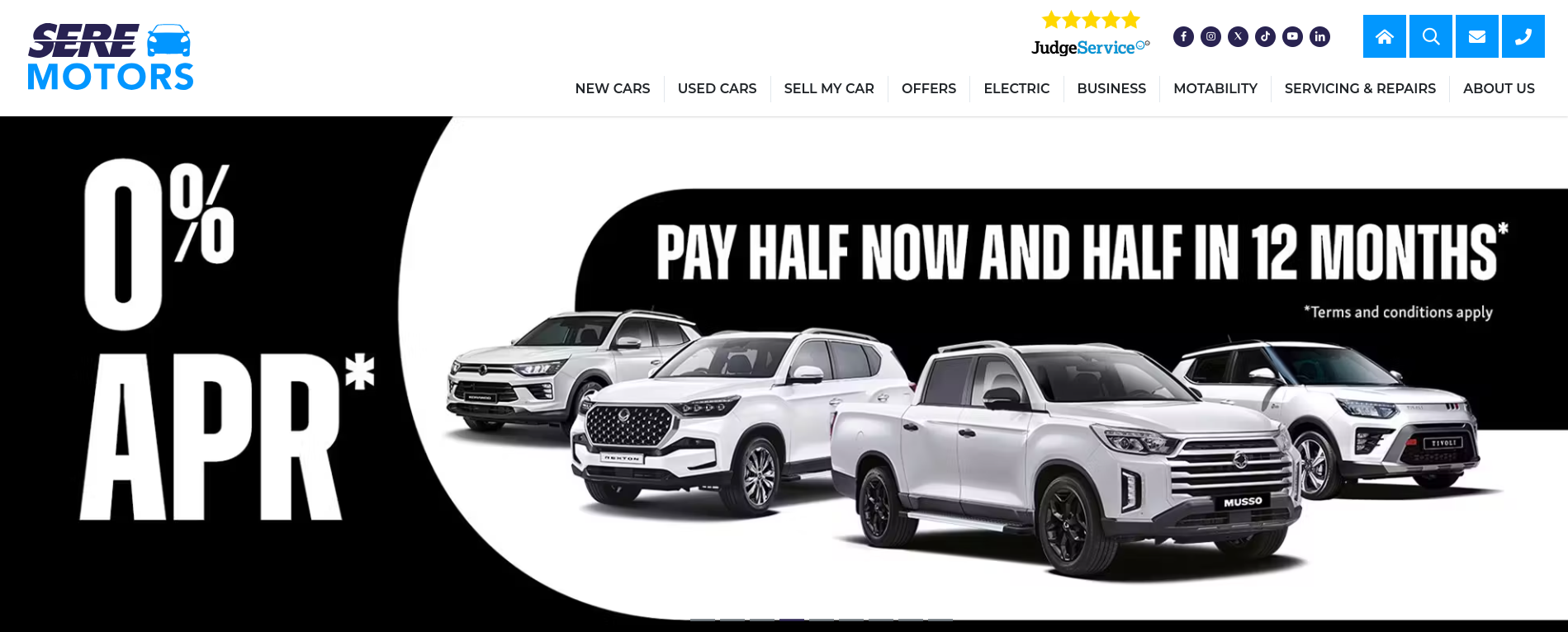
With the backend conceptualised, what online marketplace would be expected to make good use of a modern, friendly user interface? We decided to focus on something that caters to a specific market; having a general web store (ie. Amazon) would be too broad and complex for this assignment, and would involve too much time spent on gathering assets and setting categories when only a few of each should be efficient. Abhi settled on the idea of a used car dealership – the scope of content is kept small while allowing us to focus on modern elements such as animations and accessibility. His knowledge of cars also make our job of setting categories and gathering car images/details easier.

I set out to research common elements that would be found on car dealership websites and how they are navigated. Screenshots were gathered of a few dealership websites, preferably ones with a clean look and low complexity as we aren’t planning to incorporate every feature. Local dealerships had a common design language with categories kept at the top of the page at all times, with the content kept below. Animations are simple but effective which would make navigation and accessibility easier.

Here are the websites we referenced;

https://www.crawfordclarkecars.co.uk/

[](https://www.paulblaneymotors.com/)https://www.paulblaneymotors.com/

https://www.seremotors.com/

When gathering assets for the website we aimed to keep everything consistent – colour schemes, the website’s general layout, and size and style of images. Images were somewhat difficult – every car is taken from a different angle and not all are transparent. This would pose an issue if we decide to change the background colour. I opted to search for images with consistent angles and lighting, and manually removing the background if necessary.

I also designed the website’s logo. A few drafts were made but tended to look somewhat simplistic and didn’t match the look other websites were going for.

In the end, this is the logo we settled on.



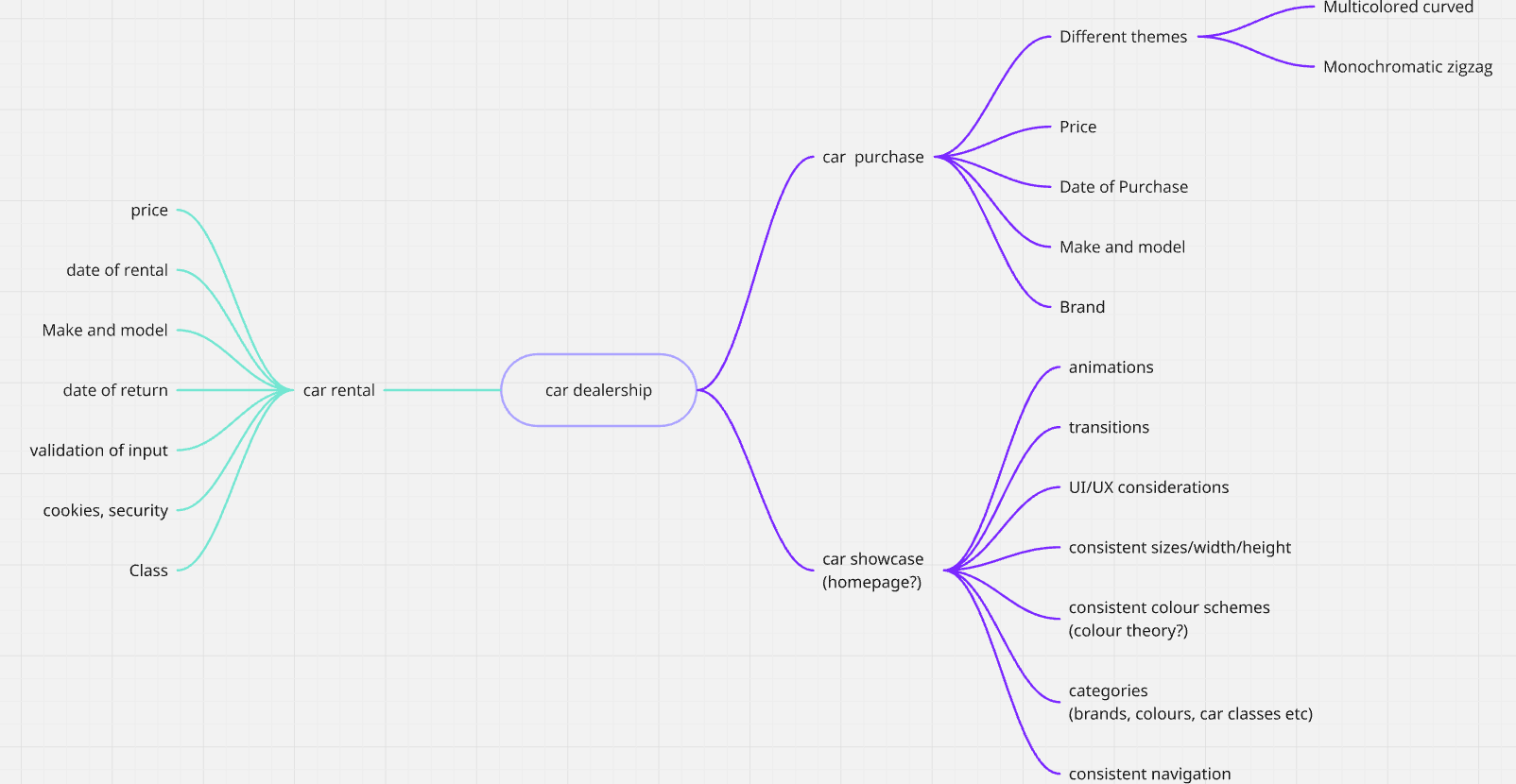
It has clear imagery to symbolise what the website is and stands out on most backgrounds we tested.

The favicon was created next – a simple vector greyscale graphic of car keys. Detail is kept to a minimum due to its small size.



We also needed a way to synchronise assets and code between each other. Originally we settled on a Visual Studio plugin called LiveCode but had issues getting it to work on everyone’s machine. We finally settled on using GitHub due to its version control and commit history, allowing us to track who contributed to which piece of code or assets. Any unwanted changes could also be reverted if needed. The only concern with this approach was making sure everyone would commit their changes manually so no work would be lost or replaced.

We could now conceptualise the layout of the website. Again, we used Miro to determine how the site will be navigated and which categories would be present.

  
After this we were ready to start coding. Nathan was not present during our first meet but still coded his own index layout while Abhi and I conceptualised. Abhi and Nathan coded in 6 categories – Sedan, SUV, Truck, New, Used and Luxury. Each had an Electric, Hybrid and Petrol subcategory. The New, Used and Luxury subcategories would belong in a main “Car Purchase” category. Each one would be populated with a set of 5-6 cars with each having their own unique photo and a list of details and specifications, which Nathan and I researched and implemented in HTML. This was a somewhat repetitive task – The final layout code of one car detail page was created by Abhi, while Nathan and I replicated the code for every car while substituting the detail text and images. This took a while but kept the look and design of the website consistent.

With some of the content completed we focused on the top index bar. Abhi had already implemented the car categories and dropdowns so we were able to link them to the finished webpages easily. We initially conceptualised the index to have;

- Logo/Homepage

- Car Rental

- Car Purchase

- Contact Us

- Sign Up/Log In

The homepage was coded by Abhi, Nathan and I to have a simple showcase of the cars available. Abhi added a scrolling text marquee to display current news/discounts. Nathan and I modified the HTML to make each image a hyperlink to the corresponding car’s detail page. The Contact Us page was skipped as it didn’t showcase any unique functionality.

We coded the Sign In page next. Nathan and I researched how such a page would function and how form validation would be implemented. We settled on these restrictions;

- Email must have an @ to be valid

- Passwords must be over 6 characters

- All fields must be present

Nathan and I implemented the fields, while Abhi implemented the validation. We assumed the login page would make for a good use of cookies, however I understood that websites usually handle credentials differently. Credentials should usually be saved in a cookie as an encrypted hash of the original string, a session ID/token, or server-side. If our website was deployed live this would be a concern but we simply have to demonstrate the functionality of a cookie. As such it simply stores the login state (“LoggedIn = Yes”) and raw strings of the details the user’s input. I would aim to emphasise security in a future project but we simply have to demonstrate functionality here.

The same approach would be replicated for the purchase screen. I updated each car detail page with a purchase button which would link to the same page that contains purchase details. Again, Abhi implemented the fields while Nathan and I implemented the validation, such as card details requiring 16 characters in numerics only.