# Project Management 3.7 Final Reflection Abdurrahman Obaidullah

During the entirety of the project, I used Trello for project management as well as Microsoft To-Do. Using the project management tool allowed me to organise and keep the process I needed to do and complete them. One thing that was particularly good about using Trello for the developmental processes in creating the project is that I was able to share a certain card to the client or other stakeholders. This meant that I was able to get feedback from the client/ stakeholders without meeting them in person. In the card that was shared for feedback, an image or a link was provided to the webpage, so they knew what was always going on. From this, I created checklist on both Microsoft To-Do as well as on Trello. The checklists made remembering to complete certain tasks a lot easier which meant that I was able to get more feedback for further development. The use of the checklist allowed me to do the project in small chunks instead of a few large chunks. This meant that I could break my working time into small parts, so I did not lose interest. As a result, the workflow was a lot more efficient.
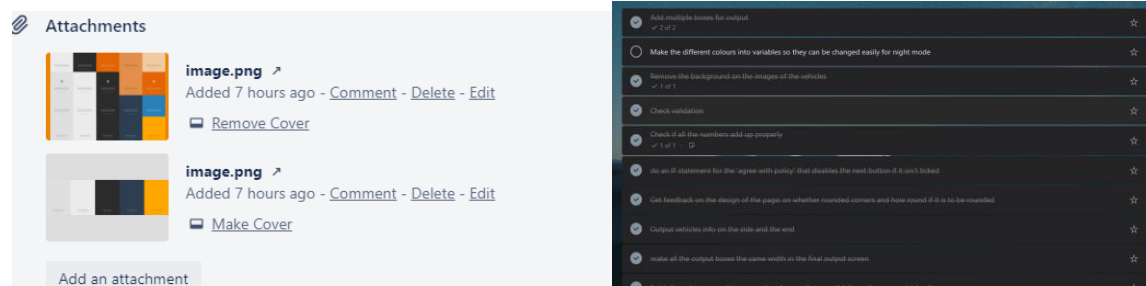
This helped complete more complex tasks such as validating inputs as well scaling for mobile easier. I also wrote down what I was doing with the page before I left, so I knew exactly what to do when I came back. I think this by itself saved a lot time and made the developmental process more efficient.

With the help of Microsoft To-Do, I was able to come back to things that I needed to complete but had not yet done so. If I was stuck with a certain part of the program or required feedback to carry on, I added to the to-do check list so that I could come back to it.

The reason that I chose to use Microsoft TO-DO for creating checklists for project management were a few reasons. One of the reasons was that I decided to use MTD was that it is available on nearly all devices that can download any software as well as it being online. This meant that I was able to access it through my phone if I got an idea or remembered something, I had previously forgot on to put into the checklist. MTD has alarm and push notifications to both PC and mobile which meant that I could set timers on when a checklist should be complete by. However, the largest reason I picked MTD was that it is my current daily driver for keeping my life in order. This meant that I was able to integrate this project management into my daily life without creating too many problems. MTD also has a cleaner design that Trello as well as being faster in terms of speed, but that didn't play as big of a role as the others did.

For the design methodology, I used agile design methodology. Agile design meant that I was basically in a positive loop of prototyping, testing, receiving feedback and then to improvement from feedback. Using agile design, I was able to go back to the client/stakeholder for feedback which in turn made final product more appealing to both the user and stakeholders/client. An example is when a stakeholder gave feedback that the page was very basic and not colourful. After this feedback, I went researching again for different designs and landed on something like the current one. After that, I received more feedback which all went together and made the page a lot better. This as a result meant that I could change the code as I went instead of changing it all at once because the client did not like it.

While trialling my website, some of the feedback I received was that selected age with numbers was not a good idea. Leaving just a blank box that tells you to enter the number of days you want to hire for was not very efficient. It would raise problems that the user could enter more than maximum date, which would cause the page to crash. This was good feedback and lead me to add addition and subtraction buttons to increase/decrease how many days you are hiring for.

One of the key decisions that I made during the project was restarting the code from basically scratch. This was an important moment as the main reason I did this was that I noticed that I had a lot of code that I did not need yet I had it. I lost code that I needed and code that I didn't need which was a good decision. I was able to restart fresh with new design ideas as well as coding ideas that would make the code more efficient. I managed to do more research which pushed me with feedback to basically add more to the page both in terms of content and colour.

Another change I made during the project was splitting the entire site into small pages with the user unable to proceed unless they have completed their tasks/inputs. I also did research on other purchasing sites and a lot of them had multiple pages to split the code into multiple chunks, so it was easier for the user to digest.

One of the most important tools that helped massively in the development of the webpage was the Visual Studio Code IDE and its extensions. These extensions included extensions such as live servers, closing code shower as well as code beautifier (not the actual names of extensions). The first extension was the Live server. The live server meant that the page updated automatically on the web and I did not have to open or refresh the tab every single time I updated something. I ended up saving a lot of time since the page wasn't reloading, it was just updating changed content live.
Another extension I used was the code beautifier. The purpose of the code beautifier is to clean the code every time the page was saved. Having the code cleaned before, while and after I was coding was a massive relief as cleaner code is easier to navigate and as a result, I was able to code more efficiently. This ensured that my code was always clean and was easy to navigate.

I tested hiring a vehicle and noticed that I did not know what I had booked until the very end. One of the largest changes I made during the creation of the webpage was an addition of an output of the prices and vehicle details on the right of the page. I realised that while I was

booking, I did not know what I have already booked. For this reason, I added an output screen on the side.

One of the changes that I made to the interface was the progress bar on the left. This progress bar on the left tells the user where and how far through the booking they are. This was inspired by when I am ordering online, I would like to know how much there is to go.

Aesthetics

There is a logo of relating to Ricky's Rides at the top of the page in the header. There is clear contrast between the font colour and the colour of the background of the page. The font colour is a dark blue whereas the background more or less white
The design is minimal that includes that there is white space around boxes and the design is not cluttered. The footer is located at the bottom of the page and contains a footer with a copyright info and links to relevant information about Ricky's ride location, phone and email. The design should still look nice when scaled down to mobile or scaled up to a larger device.
The font is readable no matter what the size of the screen.

Future proofing and sustainability

There are code comments across the entire code, across all pages to help future coders that may upgrade/ update the page. I also used DIV tags instead of using HTML5 tags which behave differently in different environment (i.e. browsers). This also ensures consistency and precision in my design for many years to come. On top of this, I have avoided using too much hard-coded JavaScript, so the code doesn't break in the future.
Future proofing also meant using the latest version of coding platforms such as CSS3. Using the latest versions means that they still have a longer lifetime left that an older version. Newer versions are also more secure. Plugins were also avoided in order to really make the page be sustainable for years to come.
The information of the page at this point is up to date. This is so the page is still relevant many years from now.

Usability

when I designed the site, I considered the 10 usability Heuristics; which meant that the site adheres to consistency and standards.
The footer is also located at the very bottom of the page so the user may access any relevant information that they may need.
On top of this, I kept the fonts and colours on the page the right amount of vibrancy as well as keeping the design uncluttered and simple. The user is able to tell a button apart and there should also be hover effects over anything that can be clicked.
Error messages should also be relevant to the input the user is inputting and easy to understand and recover from.

Refinements I made to my interface where I added bold text on the title/ announces in the outputs and vehicle details. This meant that readability on the page was a lot better since I could distinguish between information that I really need to pay attention to and things that I didn't need to as much.