12 December 2022

DSB001 Digital Skills Bootcamp: Technical retrospective report
Project: Theatre Royal
Group: JUME
Members: Jo Middleton, Joe Mint, Ursula Zalewska, Mazen Srari, Milan Palmer, and Elliot Ip

Which bits of the exercise did you find easy/hard and why?
The easier part was doing the planning and breaking down the work. We worked well as a team to break down each 'must' task and brainstorm between us what we felt we needed for it to work. The harder part was starting the work and moving from planning to writing the code. Whilst we had a good idea of our plan of attack and how we wanted to approach the work, it was difficult on finding the starting point and being confident our plan is right and we could commence the actual coding.

What went well? Why do you think that was?
The team working in our group went very well. We held daily scrums to catch up on assigned tasks, discuss outstanding issues and brainstorm what solutions we need to move forward. Our teamwork and communication were crucial to our project progressing at a good pace, and ensuring we were all aware of what each member of the team is working on, as well as not waiting for a longer time to sort out errors, but being able to discuss and action any errors every day we received them.

What went badly? Why do you think that was?
As mentioned in other parts, a key part that went bad was the design phase being too long and this then meant we spent less time coding than we should have done. This then further impacts the endpoint of the project, where we then have less time to plan and prepare for the presentation, and instead, we are finishing up the coding and fixing bugs, rather than creating the presentation.

What would you do differently next time, in terms of process?
The next time we were doing this, the focus would be more on moving on from planning in an earlier stage. It is good to spend 1-2 days breaking down the problem and the solution but spending any more time than this and not doing any coding starts to lead to delays and can bring up more planned work than is needed, as well as exceeding what of the team capacity is. It is good to keep scope creep in order, and excessive planning only meant we leant toward excessive scope and functionalities for inclusion, rather than focusing more on planning and writing the essentials first, than at a later date, looking at musts and coulds planning after you have a minimum viable product.

Given more time what would you do to improve your solution?
If given more time, a big improvement to the application would be creating a user interface that interacts with the software backend. It would improve the useability, as well as look more professional to a user with a refined interface, allowing them to search/book this way, rather than a text-based interface. We found a solution on how we could do this using Java, but unfortunately, due to time constraints, we were not able to implement this at this time.

What skills did you develop or improve during the project?
A key skill learnt from doing this project is working in software development as a team. The projects so far have been working solo on the projects and working on the solutions. This project allowed us to work in a collaborative way, bringing with this the need for open communication between those involved, compromising on solutions, and agreeing on the best way forward which may not be the solution you would take if taking on the project in a solo capacity. The project as well allowed us to have the confidence to work with a team of software developers.