What went well:

In this project we did a lot of things positively and ensured that we worked as a team throughout the process. One of the positives of this task was splitting up the coursework documentation early and looking at our strengths and weaknesses from previous modules and that shaped who did what. For example, Joe was good at Gantt charts and have done them for every single module that required them in the past which led to an easy decision in who was being allocated that task. Also, there was a lack of selfishness in how tasks were allocated and everyone did their fair share throughout the tasks, which helped team dynamics and morale to get things done where required.

Another positive was the use of Github for the code development as well as the organisation of coursework files. This acted as a centralised location for all the work to be submitted with the safety of using the backups if things went wrong, as well as having a safe place to keep the files in case of memory loss. By using this we were also able to have issues generated, which could be assigned to different people and then labelled appropriately which helped organisation and led everyone to understand what they had to do.

The code itself mimics a lifelike problem that has been solved, which helps the development as rational scenarios could be thought up and then this changed the code accordingly. This also allowed us to research how airport systems work and what elements have been used before and which ones should be included in our project.

What went wrong:

One of the drawbacks of the program is the fact that it doesn’t have a graphical interface which takes away from the user experience slightly and makes it a little outdated in it’s aesthetic. Despite this, it still has the required functionality for what we set out for, but this would be added later if we had more time. This is discussed below.

Another issue was meeting deadlines that we had pre-defined to try and set up a structure to follow. This became hard when taking into account different projects and coursework hand-ins, which resulted in a lack of communication at certain points of the module and a sudden kick into action about a week before the final hand in. Although the work was eventually done, more time should’ve been given to allow issues to be sorted in the final week rather than a rush of work that could’ve been a lot better. Despite this, the group worked efficiently when it was required and there was an equal amount of work done by all members and it resulted in a very positive working environment to develop a product in.

More Time Ideas:

* A GUI that allowed the user to log in and register and have a booking list on the front page. It would also hold information such as flights and online check-in and then would allow the user to book the seat and see a diagram of the plane, rather than 0 and X. This would improve the user experience which is integral for a today’s-world product.
* Instead of using a static data file like XML, something like a more significant database could be used, for example SQL, which would allow easier changes within tables and data and would also be easier to write to, as SQL parsing would be a difficult implementation but would then only have to be called remotely from that point onwards.