# Software architecture report

The program was written in three layers. Database, Code and UI. The database was provided, whilst the code and UI was written by us.

The program we have created is based around the database and .dll’s that we were provided for us when we started. The program works by using linked built in methods to create, edit and delete data from the database while providing a platform to view it easily. Our end goal was to create some software that would alleviate the pressures of using a paper system to manage the system.

Each separate entity eg. Customer, tour or ticket are treated separately within the database each having their own list and being linked using a foreign keys, for example each ticket has its own unique primary key in the form of its ID but it is linked to a unique instance of a tour by the foreign key of the tours ID. This means that all the separate lists are linked together meaning we could edit related data easily.

The UI is made using windows forms with the standard toolbox buttons, text boxes and list boxes using events to call the built in methods to edit the database.

The methods were pre-written within the database and from the use case diagram we were able to implement the right methods at the right points in the program (e.g. all ticket methods were contained under the ticket tab in the UI). The goal to deliver a usable and reliable piece of software, we believe, was met. This was as a result of our testing and a clear plan to begin with.

All methods were encased within try, catch braces. This allowed the team to attempt the methods, then if issues occurred, read what exception was thrown through a message box.

NB: We would have included the USE CASE diagram however it was on Connors computer and he has left due to mitigating circumstances so we were not able to retrieve it for submission.