In this report I will be acknowledging the process which the team consisting of Haroon Sadiq, Hassan Nisar, and Mauro Nunes took to complete the Implementation Segment of the Assignment. The Implementation is built up of 6 different portions from diagrams, coding and detailed annotations and the team had spent days and nights completing these portions.  
  
The team started off with the Statement of Requirements as we knew if we had a base of requirements and needs of the Customer it would be easier to focus on what the program would need to do, as if we did not have any Customer requirements we would not have any boundaries for the program it could either not fulfil the needs of the Customer or go totally over and beyond what they need. We first looked at the Use Cases from the Design Section as this entailed what everyone (Customer, Venue Manager and Ticket Agent) would need doing so this let us have around enough requirements to get started as we knew everyone needed to login with a password and username, the Customer needed to buy a ticket and we could simply see from the Use Cases what we needed to have for everyone. The Statement of Requirements was generated in a word document.  
  
The Class Diagram was completed after we had finished the Implementation as in the Implemented Code was used Inheritance, generalization and much more and all of this was done in BlueJ, we …………………………………………………………………  
and out came a class diagram with Inheritance, primary keys and foreign keys included so making a Class Diagram before an Implementation would have been another way to do it but also would have been time consuming, with this process we were able to kill two birds with one stone as produce the implemented code first which definitely takes longer than a class diagram and with that code we generated the Class Diagram. This was generated from BlueJ into Visual Paradigm.

TALK ABOUT SEQUENCE MODEL

TALK ABOUT PSEUDO – CODE

The Pseudo – Code was mainly taken from the Statement of Requirements and as you can see, we kept focusing on the requirements as we knew our program would be based on the requirements we first listed. As a result, after looking at the Requirements we knew what we wanted the code to do but we had to not make code but a translated version and detailed. We first looked at the Customer functions such as Logging in, Holding a Ticket, Purchasing a Ticket and Inputting Details, then the Venue Manager functions and lastly the Ticket Agent, the reason for that order was because we went from the Individual who we thought had the most functions required to the least. This was generated in a word document.

TALK ABOUT IMPLEMENTATION  
  
The Testing was the last thing to be completed, the reason for this was because we needed fully completed and working code for this to work. We used Black Box testing layout to do the Testing as we started of with the process or task we were testing and a screenshot if the test worked. We got our Black Box testing for the State of Requirements and Pseudo – Code, we needed the State of Requirements as have we fulfilled the actual purpose of the program or not, and the Pseudo – Code was needed as did we get the Code to what we expected it to be. This was generated in a word documents.  
The team had many more meetings with the Implementation than the Design this was due to the fact that they had found the Implementation to be much more challenging than the Design and also they had to balance with the other assignments. The team usually had one hour meetings for three days a week ( Monday, Thursday, and Friday)