The Development of the Agricultural Sensor System has, despite a number of changes, gone smoothly. The original designs supplied to us were initially confusing due to many inconsistencies between the class diagram and sequence diagram. However as we planned and developed different parts of the system we were able to fill in the blanks using the UI designs. As the UI designs acted as an end goal we were able to develop around the functionality it displayed.

The initial planning of was undertaken when first presented with the designs. It was clear from the class diagram that there were a number of larger more important classes that would need to be coded before much of the system could take shape. In order to reduce time wastage early on in development we first split the different classes between ourselves. The outline of each class was then coded and setup so that as a team we could tackle the key classes and functions that effect the most functionality.

We continued to develop the software as a group following an Agile developed based approach. Each meeting working through a screen of the UI design and the functionality needed for it to function deciding on its perceived importance and development man hours. This development could then be continued as a group, formulating the important classes first. Potentially this may have slowed down our production as development outside of the team meetings may have helped speed up the process. However while focusing on the key underlying functionality we thought it best develop as a team in order to ensure each member understood the processes in the program.

The designs supplied for the development of the agricultural sensor system required a number of different changes in order for the system to work effectively. Initially when developing the class diagrams functions it may have been easier to start developing the sequence diagrams and the UI. Much of the class diagram lack description as to the purpose of the functions and we found ourselves trying to find a place where they could be used.

In most cases the changes made were obvious and minor. Such as the addition of a Vector<SensorData> to the Historical Data class. Without the vector the historical data would have had no data to return. Many inconsistencies were missing parameters in functions differing between the class and sequence diagrams. AddSensor in the FieldStation took 3 Strings used in the Sensor constructor however the initial constructor only took 2 strings, missing one for the units. These changes would have gone unnoticed had we worked following the sequence diagrams, however these small inconsistencies delayed production as we analysed the documents looking for a way the functions could perform without the missing attributes. Though in analysing the documents in further detail we have had to make very few major changes, instead the system matches the sequence diagrams and UI almost exactly.

However larger changes included the addition of an extra Map<> to the servers data. This changed ‘Map<String, HistoricalData> data’ to ‘Map<String, Map<String, HistoricalData>> data’. Originally this data was to contain the historical data relating to the FieldStation Id however as we found from implementing the UI there was a requirement for HistoricalData relating to individual sensors as well. This alteration changes the behaviour of different functions in the Report Class as now data could be returned from either individual Sensors or Fieldstations as a whole.

Further changes made to the original design documents were mainly additions to the features presented. The original documents lacked a description of how the users register. In order for the system to work effectively we had to design a new register page and add a button to the original login page. Further to this the original UI displayed ‘Detail’ buttons for both sensors and field stations. Without any knowledge of what to present to the user we had to design our own dialogs deciding what information to display. Potentially before starting development it would have been beneficial to first plan out each aspect of the UI, and how it relates to the back end classes. This would have allowed us to develop the UI first using a front to back development approach. However that said our used development plan of development sprints focused on individual storyboard screens did allow us to ensure the functionality for those screens worked effectively before moving on.