**Criteria 10**

Although the project was finished on time, the solution that was created by the end was still incomplete with bugs and some unfinished features. Some of the task delays experienced throughout the development may have contributed to this end result. The first that was experienced was despite the SRS being given only one day to complete on the Gantt chart, in actuality it took 2-3 lessons to complete, a minor setback that affected the entire project timeline. The design creations also took more time than expected to complete, with the scale of the task not expected. The final design required more touch ups to be completed outside of class, leading it to be completed one day after initially planned. Furthermore, during the coding stage of the project misevaluation of the scale of the solution and the coding skills required led to the process taking longer than expected. An extension was necessary to get the solution to a state which had all of the required functionalities and at least running. These delays may have contributed to the end product not being fully completed. With this in mind, for future projects it would be prudent to assign more time for the coding and design stages of the project so that sufficient time is given to complete these phases adequately.

Assumptions made about certain stages of development as well as the simplicity of particular functionalities of the solution were erroneously made leading to the project needing more time than was initially planned. For example, the creation of the designs took longer than expected with some delays and an extension needed. Furthermore, the writing to files functionality of the solution was more complicated than initially thought, leading to more time needed to figure out how to construct it and ultimately it ended up with more bugs. As such, the project cost more time than expected and presented more difficulties. To avoid this, creating a project with a smaller scope may be necessary. Additionally the project had multiple changes to the timeline which led to the initial timeline in the Gantt chart not lining up with when the project finally completed, being around a week or two later than the original set final date of development.

The prototypes – whilst fulfilling most of the necessary requirements – being filled with many bugs is most likely the result of the limited time frame of the project as well as the underestimated level of developer experience needed to complete the solution. For future projects, the developer’s level of skill needs to be considered before commencing development.

Despite the aforementioned shortcomings of the project plan, it did assist in the completion of the project. The clear outline of all of the tasks meant it was clear what and when tasks needed to be completed and what needed to be done first before the project could continue. In the face of the project not going as expected the candid nature of the plan helped steer the development forward despite difficulties experienced along the way. To further assist this, annotations were made to the Gantt chart noting down extensions and delays faced, areas of the Gantt chart which required more detail and/or sub tasks, any changes to the timeline, and the reasons behind all of these.

**Evaluation of the project plan**

**Completeness**: Were any tasks missing in the Gantt chart and/or plan?

As noted in the annotations of the Gantt chart, more detailed sub steps were needed for the criteria 6 phase of the project as it only had a few steps such as implementation of the UI and internal documentation. However the rest of the Gantt chart did not have this issue.

**Maintainability**: Could the Gantt chart be easily annotated to reflect the current situation of the project?

Annotations were able to be easily added to the Gantt chart to reflect contingencies and thus still remain helpful in guiding the development of the project.

**Accuracy**: Were the tasks set at the right time? Were the time estimates correct?

Many of the tasks were not completed at their original set date as the dates initially put reflected an outdated timeline from the previous year. The time estimates for some tasks were correct, but some needed extensions such as the final designs and the development of the solution. These were reflected in the annotations of the Gantt chart.

**Readability**: Were there any difficulties in comprehending the tasks on the Gantt chart?

The Gantt chart was fully readable and there were no problems with attempting to read the Gantt chart.

**Conclusion**

The Gantt chart’s advantages of guiding the development and giving clear, set tasks on what needs to be done on certain days helped in guiding the project to completion. However, the Gantt chart has the disadvantage of not being able to account for contingencies that may arise during development and thus it can easily become inaccurate over the course of the project.

From what was experienced from this project, to improve a future one more detailed sub-tasks would be added to the Gantt chart and more slack time would be added to account for contingencies that may arise. This will decrease the strain on time for the developer and increase the likelihood of creating a complete prototype as per the developer specifications.

Further development on the solution if there were more resources and time may result in all of the planned features being implemented such as the score feature and sorting the questions by tags. Additionally more time would be spent testing the solution and fixing present bugs that prevent it from being complete.