Category	Description	Reviewers Comment	Action taken by reviewed group
Build	Could you clone from Git and build using the README file?	I was able to clone the git repos for the frontend and backend, but I was not able to build either project. The backend project had no setup instructions, and we were told the project required hardware to run. The frontend project did not intially have setup instructions, but after I told them that over email, they added some setup instructions. Unfortunately the instructions required many steps (including having access to Visual Studio), and they did not provide enough time to complete the build	
Legibility	Was the flow sane and were variable names and methods easy to follow? Does the code adhere to general guidelines and code style?	The frontend was extremely messy code, and hard to follow. The project contains many complex classes with absolutely 0 comments in any of the files. There were also many hardcoded “magic” values throughout the files with no indication as to the significance of the hardcoded values. I can’t imagine any developer (except for the person that coded it) would be able to pick up the project and understand what was going on. The backend was pretty solid, the classes were formatted much more organized, and were easier to follow. There were a few places that would benefit from more comments, but in general I felt that the comments were sufficient.	
Implementation	is it shorter/easier/faster/cleaner/safer to write functionally equivalent code? Do you see useful abstractions?	I don’t know that there are any places I would suggest implementation changes on the backend, but there are certainly places on the frontend that could be optimized. I would start by refactoring some of the methods in Gauge.h to simplify the logic.	
Maintainability	Are there unit tests? Should there be? Are the test covering interesting cases? Are they readable?	There are no unit tests, but as is the case with many of the AIAA projects, its not necessarily the type of project that is especially well-suited for unit tests.The frontend has a mode that allows it to be run with fake data which is good.I would like to see a full integration test developed where they mock the data source on the backend and test frontend/backend integration.	
Requirements	Does the code fulfill the requirements?	The code seems to satisfy the requirements described in their requirements document.	
Other	Are there other things that stand out that can be improved?	The frontend really needs a lot of work. I’ve already described my recommendations in terms of what they should fix/change. I’d like to also note that the backend had descriptive commit messages to go along with each commit, which seriously increases a 3rd parties’ ability to understand the code. I would suggest they follow suit on the frontend, as they currently are not.	