	50
A: +90%	45
B: +80%	40
C: +70%	35
D: +60%	30
F: <50%	25

CS Capstone Design

Final Project Demo Grading Sheet (50 pts)

TEAM: Team Controller

Overview: The main purpose of the "Technical Demos" is to very clearly explore the extent to which the team has implemented the key functional and performance requirements for their project. Grading is based on two factors:

- **Completeness:** Have all of the key requirements been implemented. To what extent does the product have all of the functionalities and performance that was promised.
- **Quality:** Just having basic functionality is the bare minimum. What is the quality of the implementation? Is the resulting product aesthetically pleasing, easy to use, and a pleasure to work with; to what extent is it "ready for market"?

This template is fleshed out by the team, **reviewed and approved by CS mentor beforehand,** and then brought to demo in hardcopy for the mentor to use as a grading sheet.

Requirements Review

Based on our requirements acquisition work and evolution during implementation, the following are the key technical requirements driving of our product:

CR06: Installation without admin rights. The data display module shall not require admin rights to install setup or use. The user will be able to install the application without the use of administrator rights

Specific implemented functionalities that satisfy this requirement:

- The WSSS_Setup.exe installer installs without admin rights

CR02: Serial communication. The data display module shall be capable of serializing messages to send and deserializing messages received via an RS422 serial port. Additionally, the controller simulator shall be capable of serializing messages to send and deserializing messages received via an RS422 serial port.

Specific implemented functionalities that satisfy this requirement:

- The controller simulator is capable of serializing messages before sending them as well as deserializing incoming serial messages.
- The data display module (our main application) is capable of serializing messages before sending them as well as deserializing incoming serial messages.

CR04/CR05/CR07: Viewing controller data. The data display module shall display all weapon status information directly to the application's window for the duration of a session. Additionally, The data display module shall have the capability to display **only errors, cleared errors, active errors and non-error events** to the Events tab of the GUI

Specific implemented functionalities that satisfy this requirement:

- The events page as well as the status, and electrical will display data sent by the controller simulator
- A filtering option on the events page allows users to filter through the different types of events or errors

CR03: Outputting to a logfile. The data display module shall be capable of generating a log file including all known events when requested by the user. The user shall be able to determine if a log file will be automatically generated after a session. The user shall be able to determine how many auto saved log files will be kept before overwrites occur on the oldest autosaved file.

Specific implemented functionalities that satisfy this requirement:

- The application keeps a running log file that adds data to the end of the file after each data point comes in and a manual log file that contains all data up until that point. Additionally there is a user setting that can be adjusted by the end user that details the number of running log files that can exist in the system before clearing the oldest one.

Demonstration Sequences:

This section outlines the demonstration sequences prepared to prove the above functionalities. Each sequence is a coherent walk-through of some piece/area of the product, designed to highlight implementation of specific requirements/functionalities outlined in the last section.

Demonstration Sequence 1: Installation

Requirements demonstrated: **CR06**

Flight Plan for this demo sequence: Step by step plan of things to do/demo

1. First the user will obtain the installer file
2. Then then user will interact with the installer by following the steps presented
3. The user the will finalize the installation and will have an installed executable of the file on their computer

Evaluation (filled in real-time by mentor):

✔ Convincingly demo'd each of targeted requirements?

✔ Quality, aesthetics and other evaluative comments:

Demonstration Sequence 2: Serial Communication

Requirements demonstrated: **CR02**

Flight Plan for this demo sequence: Step by step plan of things to do/demo

4. The user would start the application
5. Assuming the user has the proper serial ports configured, the user would then connect to the controller (separate device simulating the controller)
6. An indicator resembling a light will turn green to notify the user that the controller is connected. (serial messages being sent from CSim will be visible through developer page)
7. Once the application disconnects from controller the indicator will turn from green to red to indicate that the connection was ceased

Evaluation (filled in real-time by mentor):

✔ Convincingly demo'd each of targeted requirements?

✔ Quality, aesthetics and other evaluative comments:

Demonstration Sequence 3: Viewing controller data

Requirements demonstrated: **CR04/CR05/CR07**

Flight Plan for this demo sequence: Step by step plan of things to do/demo

8. Once the controller connection has been established, the user can navigate to events, status and electrical to view information from the controller's session.
    a. Events will show the weapon's events and errors
    b. Status will show the current status of the weapon
    c. Electrical will show the name, voltage, and amps for each of the weapon's electrical components
9. The user can then look in depth at the events page with the filtering options that include:
    a. default
    b. all events
    c. all errors
    d. cleared errors
    e. active errors
10. Additionally, a search feature allows the user to search any keyword inside the GUI

Evaluation (filled in real-time by mentor):

✔ Convincingly demo'd each of targeted requirements?

✔ Quality, aesthetics and other evaluative comments:

Demonstration Sequence 4: Outputting to log file

Requirements demonstrated: **CR03**

Flight Plan for this demo sequence: Step by step plan of things to do/demo

11. After a successful connection to another laptop with the controller simulator running, the user can open the 'Open Log File Folder' button to view all log files saved. There are two types of log files:
    a. Automatic log files: These log files are automatically generated and added to after each data point
    b. Manual log files: These are manually downloaded files that contain all of the event and error data
12. Additionally these log files include a user setting where the user can enable an advanced log file. These advanced log files include status updates as well as electrical data and session statistics.

Evaluation (filled in real-time by mentor):

✔ Convincingly demo'd each of targeted requirements?

✔ Quality, aesthetics and other evaluative comments:

Demonstration Sequence 5: User settings

Requirements demonstrated: **NA**

Flight Plan for this demo sequence: Step by step plan of things to do/demo

13. The user can change and save their serial connection and log file settings in the connection tab and events tab respectively.
14. Once the user closes the application, the user will then re-open the program and it will retain the modified settings from their last session.
15. Additionally, the user can enable or disable the following settings that are application specific:
    a. Colored events output - affects the appearance of the events page
    b. Advanced log file - includes additional information on the log files
    c. Notify on error cleared - provides a notification when the controller simulator cleared an error
    d. connection timeout duration - timeout duration for when a physical connection does not receive data. Will disconnect the session if data has not been received in that time
    e. auto saved log file limit - the number of autosave log files that will be saved before the oldest log file is deleted

f. ram clearing and max number of nodes- optimization that clears the GUI when the max number of nodes are met

Evaluation (filled in real-time by mentor):

✔ Convincingly demo'd each of targeted requirements?

✔ Quality, aesthetics and other evaluative comments:

Demonstration Sequence 6: Developer settings

Requirements demonstrated: **CR02**

Flight Plan for this demo sequence: Step by step plan of things to do/demo

16. Once enabled, the developer user will be able to control the information going into the product for testing. There will be a new tab called "Developer" that the user can navigate to.
17. The developer user will be able to start a simulated controller session running in the background.
18. The developer user will be able to view all randomly generated messages in the UI by the simulated controller.
19. The developer user will be able to send their own custom messages via serial communication from the simulated controller.
20. The developer user will be able to clear any non-cleared errors from the current simulated session.
21. The developer will be able to change the generation rate of the incoming data

Evaluation (filled in real-time by mentor):

✔ Convincingly demo'd each of targeted requirements?

✔ Quality, aesthetics and other evaluative comments:

Demonstration Sequence 6: Notifications

Requirements demonstrated: **NA - Nice to haves**

Flight Plan for this demo sequence: Step by step plan of things to do/demo

22. Before connecting, a notification that a saved serial port has opened will be displayed in the top right corner of the application as well as the notification page.
23. Once the connection is established, a notification message will appear notifying users that a session has been started and the auto save log file has been created.
24. After disconnecting a notification will be displayed telling the user that the session has ended and statistics about that session.
25. Upon an error or issue with the application's function, a red dot will be displayed on the notification icon along with a warning message.

Evaluation (filled in real-time by mentor):

- ✔ Convincingly demo'd each of targeted requirements?

- ✔ Quality, aesthetics and other evaluative comments:

Other challenges recognized by not addressed by demo:

If there were requirements that you listed earlier that were *not* covered by a demo, list here. This will hopefully be a short list…but better to be clear about what is done and not. If there are items here, one could provide explanation of why they weren't done.