

Category	Description	Reviewers Comment	Action taken by reviewed group
Build	Could you clone from Git and build using the README file?	It was unexpected that the README led us to a google drive, to download the project. The setup did work very quick and simple.	Moved compiled program builds to the releases section of github.
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 code seems to be following the guidelines, and it is easy to follow overall. Maybe some reordering here and there.	No specific action taken.
Implementation	Is it shorter/easier/faster/cleaner/ safer to write functionally equivalent code? Do you see useful abstractions?	The code seems repetitive, which should be fixable with loops. Fixing this will probably reduce the total amount of code.	Repetitive code is an artifact of buttons in a WPF project being references to an object rather than sequential spaces in memory. No action.
Maintainability	Are there unit tests? Should there be? Are the test covering interesting cases? Are they readable?	No unit test, I'm not sure if there should be any.	No specific action taken. Verification functions for inputs exist in framework and file io.
Requirements	Does the code fulfill the requirements?	I think the code meets the requirements	No action needed.
Other	Are there other things that stand out that can be improved?	I would change a bit of the UI, like bigger buttons, so it's easier to press on a screen. Have the same amount of time buttons per row.	Made changes to the timeline in order to make more readable.

Category	Description	Reviewers Comment	Action taken by reviewed group
Build	Could you clone from Git and build using the README file?	Yes, setup was easy using the files from the Google Docs as directed through the README.	No action needed.
Legibility	Was the flow sane and were variable names and methods easy to follow? Does the code adhere to general guidelines and code style?	Yes, it seemed to be consistent in code style and general guidelines. It could be improved by reordering some things like the timelines.	No action needed.
Implementation	Is it shorter/easier/faster/cleaner / safer to write functionally equivalent code? Do you see useful abstractions?	It would be faster to have some of the code in a loop to reduce repetitiveness. It is generally not a good idea to have the same code repeated in case changes are needed.	Repetitive code is an artifact of buttons in a WPF project being references to an object rather than sequential spaces in memory. No action.
Maintainability	Are there unit tests? Should there be? Are the test covering interesting cases? Are they readable?	There are no unit tests. The app seems simple enough that it could be tested simply through user tests.	No specific action taken. Verification functions for inputs exist in framework and file io.
Requirements	Does the code fulfill the requirements?	Yes.	No action needed.
Other	Are there other things that stand out that can be improved?	More guidelines on required input would be good. That, or more checking that the input is correct.	No specific action taken. Verification functions for inputs exist in framework and file io.

Category	Description	Reviewers Comment	Action taken by reviewed group
Build	Could you clone from Git and build using the README file?	It was a little confusing being directed to download all the files from Google Docs, but after I did, setup was quick and simple.	Moved compiled program builds to the releases section of github.
Legibility	Was the flow sane and were variable names and methods easy to follow? Does the code adhere to general guidelines and code style?	I'd say the code followed general guidelines and code style. It was a little hard to follow through and to figure what methods were available or being used. Some things were also kind of out of order, like having Timeline 2 followed by Timeline 4, 3, then 8 in order.	Comments have been added to many areas of the code, certain pieces of code such as registry value changes have been extracted to their own files.
Implementation	Is it shorter/easier/faster/cleaner/ safer to write functionally equivalent code? Do you see useful abstractions?	A lot of the code seemed repetitive and could be contained in a loop. For example, with the Timelines objects in Form1.Designer.cs, only the drawing coordinates seem to be the unique element between each one that can't be done via increments in a loop. Adding each coordinate to an array and increment through that would be a solution to that. Changes like that would cut down the size significantly.	Repetitive code is an artifact of buttons in a WPF project being references to an object rather than sequential spaces in memory. No action. Additionally, the fix described here is the current code.
Maintainability	Are there unit tests? Should there be? Are the test covering interesting cases? Are they readable?	No unit tests.	No specific action taken. Verification functions for inputs exist in framework and file io.
Requirements	Does the code fulfill	This code definitely meets	No action needed.

	the requirements?	the requirements.	
Other	Are there other things that stand out that can be improved?	Making it more obvious how to use the app would make it a lot easier to use. Things like the recommended feeding rate not being on the time boxes and those only getting feeding rates when a break is added, and only before, just would be confusing as a new user with no other information.	Readme covered by nurse training programs. Bugs fixed by addition of a new window to ensure valid variable values.

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 and launch the program.	No action needed.
Legibility	Was the flow sane and were variable names and methods easy to follow? Does the code adhere to general guidelines and code style?	There was lots of repeated code, but things were structured in a sane way with related code being grouped together and indented nicely.	Repeated code based on api restrictions, no action taken.
Implementation	Is it shorter/easier/faster/cleaner/ safer to write functionally equivalent code? Do you see useful abstractions?	There was lots of code that seemed to be able to be put into for loops / abstracted. Such as Timeline assignments, for example.	Repetitive code is an artifact of buttons in a WPF project being references to an object rather than sequential spaces in memory. No action.
Maintainability	Are there unit tests? Should there be? Are the test covering interesting cases? Are they readable?	No unit tests. There should be unit tests for different possible inputs to the program to make sure things don't break on certain inputs.	No specific action taken. Verification functions for inputs exist in framework and file io.
Requirements	Does the code fulfill the requirements?	Meets most requirements except the not crashing requirement.	Made changes to the startup routine that prevent crashes from uninitialized variables.
Other	Are there other things that stand out that can be improved?	Still some bugs they mentioned. A readme on how to use the program would be nice.	Readme covered by nurse training programs. Bugs fixed by addition of a new window to ensure valid variable values.

Category	Description	Reviewers Comment	Action taken by reviewed group
Build	Could you clone from Git and build using the README file?	Yes	No action needed.
Legibility	Was the flow sane and were variable names and methods easy to follow? Does the code adhere to general guidelines and code style?	Yes and yes	No action needed.
Implementation	Is it shorter/easier/faster/cleaner/ safer to write functionally equivalent code? Do you see useful abstractions?	There might have been a faster way to initialize all of the time buttons, but not a high priority item.	Repetitive code is an artifact of buttons in a WPF project being references to an object rather than sequential spaces in memory. No action.
Maintainability	Are there unit tests? Should there be? Are the test covering interesting cases? Are they readable?	Yes, it's easy to test with loading/saving .txt files.	No specific action taken. Verification functions for inputs exist in framework and file io.
Requirements	Does the code fulfill the requirements?	Yes	No action needed.
Other	Are there other things that stand out that can be improved?	The -1:00AM bug when the application is first opened.	Changed the first window from the main window to a smaller pop-up which forces initialization of all variables.