
Programming Assignment 1

Date Posted: June 7
Preliminary Class Diagram: June 16
Class Outline: June 27
Functionality Outline: June 30
Final Project Due: Thursday, July 7

Programming Assignment 1

Background Information

Drilling for oil in the United States has become a high-tech industry. The oil production companies rely on a complex array of sensors on each drilling rig to monitor the state of all operations. These sensors feed data to a server mounted on the rig. The server is connected to a network which enables workers on the rig as well as managers back at the company offices to monitor all aspects of the operations.

Your Assignment

You are to create a program that simulates the sensors on oil rigs. Following is a list of requirements based on a preliminary discussion with representatives of Oilfield Instrumentation-USA (OI-USA)*. (Note: Your list of requirements in the documentation shall expand on the list below. **Just copying this list and calling it your requirements is NOT acceptable.**)

1. The application shall allow the user to select any number of wells from a list of active wells. At any time while the application is running the user shall be able to add or remove wells from the selected list.
2. The application shall allow the user to select any number of sensors to monitor on each selected well. At any time while the application is running the user shall be able to add or remove sensors being monitored for each selected well.
3. All data defining the list of active wells and the sensors on each shall be read from a data file in a modified XML format. A data parser class will be provided to read and parse the data file.
4. Sensor readings for all selected wells shall be printed on the screen at 5 second intervals.
5. In this version of the software, wells shall be identified by:
 - The name of the company operating the well.
 - A well ID in the format "nn-nnnn" where n is any digit in the range 0..9.
6. In this version of the software, possible sensors for each well shall include:
 - **Hole depth** - Current depth of the drill hole measured in feet.
 - **Bit depth** - Current location of the drill bit in the hole measured in feet. This is the same as hole depth unless the bit is traveling, i.e. being removed or inserted.
 - **ROP (Rate of Penetration)** - Rate at which the bit is drilling. This is measured in feet per hour of drilling except when the bit is traveling when the reading displayed for this sensor is "Off Bottom", i.e. is not currently drilling.

- **Pump pressure** - This is the pressure in the pump which is pumping the mud (lubricant) down into and out of the hole. This is measured in PSI (Pounds per Square Inch).
 - **Casing pressure** - Pressure in the bit casing. This is measured in PSI.
 - **Flow out** - Mud flow out of the bit casing. This is measured in percentage (%) of the maximum possible flow.
 - **Torque Max** - Maximum torque that can be applied to the bit. This is measured in kilo-footpounds (kflb).
 - **Mud Pit Volume** - Volume of mud (used as a lubricant) available. This is measured in barrels abbreviated as **BBL**. FYI: BBL as an abbreviation for barrel(s) is an old symbol. The double 'b' is thought to either represent the plural "barrels" or to avoid confusion with the abbreviation "bl" which stood for "bales."
7. At 5 second intervals a report shall be printed on the screen giving the identifying information on each well being monitored and the current readings of all selected sensors. This display shall continue until the application is terminated unless the user is selecting wells to add to or remove from the list of monitored wells or selecting sensors to add to or remove from the list of sensors for any monitored well.
 8. This project shall be created as a Win-32 Console Application using the Microsoft Visual Studio compiler It shall **NOT** use precompiled headers.

Deliverables

These products as specified below shall be delivered electronically via e-mail to the instructor.

Preliminary Class Diagram -- The class diagram shall be drawn using standard UML notation and shall show all of the classes to be implemented in the software and their relationships (dependencies, associations, generalizations, realizations, etc.) The PCD shall be submitted for instructor approval NLT (Not Later Than) Thursday, June 16.

Class Outline -- The class outline shall list all proposed variables and functions in each proposed class with a brief description of what each does. The class outline shall be submitted for instructor approval NLT (Not Later Than) Tuesday, June 27.

Functionality Outline -- The functionality outline shall be an outline which will show the step-by-step functionality of the program. This should be taken out to a fair amount of detail. The functionality outline shall be submitted for instructor approval NLT (Not Later Than) Thursday, June 30.

Final Project -- The entire Visual Studio solution (compatible with **Microsoft Visual Studio 2012 or later**) shall be compressed into a zip file and submitted for instructor approval NLT Thursday, July 7. Just turning in your source files is not acceptable.

We will have several class periods in which we will meet as a team to discuss and plan this project. We will be doing a lot of brainstorming and planning together, but remember that each person is responsible for implementing the final design on their own.

To download a sample executable as well as a data file and a parser for the data, find them in the files sections on canvas. To see a number of hints that may be useful for implementing this assignment, find them in the files sections of the canvas page.

This programming assignment is based on a real-life situation and a software project that a former CS professor worked on while employed at SAIC. To see information on this project check out the [Oil Field Instrumentation - USA \(OI-USA\)](#) site and click on the link to "Drilling Instrumentation Systems" in the lower portion of the display.