# XML Programming Project

# Application Introduction:

(Less than 500 words)

The Wood Stock GUI application is a Windows Forms Application using the C#.NET programming language. The application was designed for a small retail store that produces wooden toys for the general public. Due to new customers coming online it was seen prudent to add new functionality to the application by means of updating the output to a suitably formatted XML file. The application now has the option to save the output file in either .csv or .xml file format. This is achieved by selecting the desired file format from the drop down list located next to the save button. If the user selects the .xml file extension they then also have the further option of selecting either no styling or one of the two styled outputs. This is done by selecting one of the 3 radio buttons located above the save file text box. The default selection is none. Note that if .csv is selected from the drop down list the three radio buttons are still selectable but have no influence on the file output.

In order to ensure a valid document is produced via the application I created a schema document which describes the structure and content for the resulting xml file. I chose schema over DTD for a few reasons, but the biggest factor was that schema uses an XML based syntax meaning I didn’t have to learn a totally new language saving me heaps of time. Another key benefit of schema is its ability to implement strong typing. Meaning the data types of an element can be defined this ensures data stored in the xml document is accurate. In my schema document I have four simple elements with their data types set, three as a string and the currentCount element set to an integer. There data types match the variable types from the application code. I also have two complex elements defined one with an ID attribute defined as an integer. This ID attribute means every Item has a unique identifier allowing me to select certain items from inside my style sheets, when trying to produce the two different options of output. Another key point in my schema document was using the all indicator inside the stockItem complex element. This allows the elements to appear in any order which is needed due to the application being able to sort the output file into a number of different orders. The all indicator also means that every element must appear only once. Lastly to ensure that any number of stockItems could occur I set this element to unbounded. While my schema document isn’t very complex it does help to ensure that the resulting XML document is both well-formed and valid.