# Module Manager Spec

Purpose:

The goal is to display information about dll files using the dll itself and the associated xml file. Each dll will be displayed as a TreeView in a window in the UI. Each module (dll file) will have its own enable/disable check box to indicate whether the module is being used or not. It will have a “load module location” button to allow the user to select the location of the dll’s. This information will be able to be saved to a config file. On startup of the application, this config file, if it exists, will be loaded and the enable/disable status of each module located in the previously specified file path will be displayed.

Breakdown:

1.) A load button will allow the user to select a directory where dll files are contained.

2.) The TreeView area will display each module (dll file) and its methods in child nodes. Each module will have an associated enable/disable check box.

3.) The user can select any of the methods from the previously stated TreeView area. The information about the selected method will be displayed in a list box next to the TreeView area. Alternatively, this information can be displayed in a “hover over” help text way.

4.) There will be a “save configuration” button that will store the loaded file location, as well as the enable/disable status for each of the modules. If this file exists on startup, the application will load with this saved data as it exists.

Implementation:

1. Allow the user to select a director using FolderBrowserDialog
2. This information will be stored in an ObservableCollection of custom Module objects that will have an overridden ToString method to organize and format the data from the .dll and .xml files.
3. Using this resource,