**FileSync Dev Notes**

**Things TODO:**

1. (DONE-Dan) There is a conflict with multiple console.readkey()s being called. The normal UI vs the "do you want to copy this file?" UI…
2. The Full Sync flag needs to be implemented to inform the logger to only record useful information when a Full Sync occurs as opposed to a massive list of all filenames that exist…  
   Example full sync log should include: Full Sync occurred, when, how many files, folders, and size, and who said to do it
3. Package repeated code and strings into methods…
4. Fix the directory path assignments at the beginning of DirSync()
5. Report Skipped items to the log including who skipped them and when  
   Example skipped item Log: The source object, the intended target, operation desired, time of occurrence, who said no
6. Implement Logging operation to report to the user which files in the source directory, before a full sync, do not exist in the target directory.
7. Implement sync ‘preview mode’ toggle function (not for user toggle)  
     
   maybe pass a preview mode bool to the sync initialization
8. Add a sync filter to ignore files with the extension “\*.metadata”  
   After the filter declaration...

Add this line→ filter.FileNameExcludes.Add("\*.metadata"); // Exclude all \*.metedata files

**Features to add:**

1. Add an auto approve toggle for sync changes? (Cole- I have code for that if we want to try’n copy it in.)(Dan- Why don’t we make that into a function which sets a bool in the sync class for implicitly accepting changes)
2. Allow the user to specify a specific Auto sync interval
3. Allow the user to flag a specific file or folder to be synced on a specific date or after a specific interval of time
4. Allow for user selection of directories.
5. Setup multiple auto syncs between different pairs of directories

**Stuff to fix:**

The Main menu. Juggling which Console.ReadKey() is active when the sync function asks for input is bad. Also, starting a sync function should prevent the ability to begin more syncs (hitting the ‘s’ key repeatedly)

If a sync is running, you might be able to delete the metadata file by pressing f

The sync function does not log it’s errors nor does it alert functions higher up the call chain that an error has occurred. RIght now it prints the error to console.

If the user presses ‘k’ and stops the auto sync timer then presses ‘s’ or ‘f’ to run a manual sync, the functions tired to ‘s’ and ’f’ reset the sync timer before they return.

Logging…

**Notes to Remember:**

**Auto approve thing:**

**we probably don’t still need this here.. eh**

I suppose the only glaring issue if any would be that anything that isn’t the ‘y’ key is implicitly the same thing as saying no.

This is the main part:

/// <summary>

/// Asks the user what they want Auto Approve to be set to.

/// </summary>

/// <returns></returns>

private static void SetAutoApprove()

{

StopTimer();

Console.WriteLine("");

Console.WriteLine("");

Console.WriteLine(

"Auto Approve is currently set to " + AutoApprove + ".");

Console.WriteLine(

"Would you like to change it to " + !AutoApprove + "? (y/n)");

if(Console.ReadKey().KeyChar.ToString().ToLower() == "y"){

AutoApprove = !AutoApprove;

Console.WriteLine("");

Console.WriteLine("");

Console.WriteLine("Auto Approve has been changed to " + AutoApprove);

Console.WriteLine("");

}

PrintNextSync();

ResetTimer();

}

Also don’t forget to add this:

private static bool AutoApprove = false;

And this to the switch statement for the Menu:

case "a":

SetAutoApprove();//sets auto-approve

break;

And this to the actual menu Console.WriteLine (aka PrintMenu() )

"\nPress the 'a' key at any time to change Auto Approve settings"

And basically wherever the getApproval() function is, the whole thing needs to be in an if/else statement:

if (AutoApprove == true)

{

Console.WriteLine("Auto Approve is on, synced file");

return true;

}

else{

(Do all the normal stuff)

}