**SimpleBackup Details**

**https://github.com/mattmcvay/SimpleBackup**

This service requires the user to have a SQL Server database back end for the following:

* A back up log
* Holding the source and destination values.

1. Create a database named AutoBackup.
2. In that database create the following table to hold the source and destination values:

CREATE TABLE [dbo].[Simplebackup](

[SourceDir] [varchar](max) NULL,

[Destination] [varchar](max) NULL

)

Note: After this table is complete, you can manually enter the source and destination, via an insert statement. Example:

Insert into dbo.Simplebackup

(SourceDir, Destination)

values('C:\Users\name\OneDrive\Desktop\', 'D:\BackupFolder\')

\*\*\*\* Each value has to end in a back slash or the program may not work \*\*\* => \

1. Create the backup log table:

CREATE TABLE [dbo].[BackupLog](

[FileName] [varchar](max) NOT NULL,

[BackupPath] [varchar](max) NOT NULL,

[BackupDate] [datetime] NOT NULL

)

1. Create the following stored procedure:

CREATE PROCEDURE [dbo].[insertRecord]

@FileName varchar(max),

@BackupLocation varchar(max)

AS

Begin

Insert into dbo.BackupLog

values (@FileName,@BackupLocation, GetDate())

End

GO

As of 5.4.2023, this service only backs up one folder location to another. Future enhancements are planned to add the option of multiple backup source and destination locations.

If the file is already backed up, the service simply writes over it with the current file. This way any updates to the file made by the user will automatically get backed up.

This service is currently set to run every 12 hours, but his can be adjusted by altering the Backup.cs file on line 20:

\_timer = new System.Timers.Timer(60000 \* 720) {AutoReset = true}; // Every 12 hours.

In the same Backup.cs file, on line 36, the list of file extensions to be backed up is listed. This line can be updated to add or remove file extension types by the user if needed.   
  
string[] extensions = new string[] { ".gif", ".xls", ".xlsx", ".txt", ".pdf", ".doc", ".docx", ".csv", ".ppt", ".pptx" };

Configure the connection string in the App.config file according to your specific SQL Server connection. This is on line 4.

The final thing to do is install the service:

1. Build the SimpleBackup project in Visual Studio. After it builds successfully,
2. Right click on the project and select ‘Open Folder in File Explorer’.
3. Navigate to bin > Debug
4. Inside the Debug folder you should see the following files:

SimpleBackup.exe

SimpleBackup.exe.config

SimpleBackup.pdb

Topshelf.dll

Topshelf.xml

1. Copy all of these files to where you want the service installed, normally they would go in a path such like the following:

C:\Users\name\AppData\Local\SimpleBackup

1. Within that folder open a Windows command prompt, or in a Windows command prompt change directory so that you are in that folder. Run the following command:

SimpleBackup.exe install start

Full example:

C:\Users\name\AppData\Local\SimpleBackup>SimpleBackup.exe install start

This will install and start your service.

The SimpleBackupUI project can be downloaded to give the user the following abilities outside of just having only the service.

- Turning the service on or off from the UI.

- Updating the source and destination values of the backup.

- Seeing the drives attached to the PC.

- Seeing the backup log in the UI instead of querying SQL Server.

**Found here: https://github.com/mattmcvay/SimpleBackupUI**