# 

**Department of Computer Science**

**Spring-2017**

**CSC 565**

Final Documentation-Medium Project

By:

Shivakrishna Bujunoori

F00442378

Contents:

[Client](#_Toc437595958) Details----------------------------------------------------------------------------------------------3

Abstract-----------------------------------------------------------------------------------------------------4

Project Title------------------------------------------------------------------------------------------------4

Requirements and Specifications--------------------------------------------------------------------5

Existing System ------------------------------------------------------------------------------------------6

Problems with Existing Solution-----------------------------------------------------------6

Proposed System------------------------------------------------------------------------------------------7

Solution over Existing Constraints: -------------------------------------------------------7

Solution Strategy-----------------------------------------------------------------------------------------8

Benefits of Proposed System: ---------------------------------------------------------------8

Prototype and Design flow-----------------------------------------------------------------------------9

Implementation------------------------------------------------------------------------------------------11

Testing and Sample test Case-----------------------------------------------------------------------15

Future Enhancements---------------------------------------------------------------------------------20

Bibliography-----------------------------------------------------------------------------------------------21

# Client Details:

  Valerie Locklair,

System Administrator, Computer Science Department

**Address:** Concordia University Wisconsin

12800 N. Lake Shore Drive Mequon, WI 53097

E-mail: [valerie.locklair@cuw.edu](mailto:valerie.locklair@cuw.edu)

Contact Number: (262) 243-5700

# Developers’ Details:

Shivakrishna Bujunoori – F00442378

Shivakrishna.bujunoori@cuw.edu

**Abstract**

This Project solution results to a feasibility for the administrator/Instructor to a particular class to track all the newly joined students who logged into the systems of Computer Science Department or the University systems with an associated Student id and a generated unique number which is a randomly generated 5 digits of Unique Number associated to username of the Student. In our case associated pair will be Foo\*\*\*\* ID and a 5 digit unique number which will be created at the first login of a student.

**Project Title**

Random Unique Number generation for new Students in Concordia University Wisconsin Computers environment

**For example:-**

Consider a student newly entered the university. Student will be given with credentials to login into the university computers by creating student’s username and password account to the network users Active Directory Groups listings (AD groups).

Without login into the university network systems, student will not be provided with a drive to store their data which is M drive in network. When a student login into the computer with credentials at first time, after the successful login it will take couple of minutes to create a personal space for the user and loads the desktop.

The requirement of this task is to create a text file with randomly generated unique number which is associated with the student and storing this text file.

* For viewing to the user in a location within the assigned M drive space.
* Keeping the associated random unique number according to student for the administrator for usage in tracking and business purposes.

**Requirements and Specifications**

The following are the potential requirements of this project are listed below

* Generating 5 digits of Unique Number to the newly logging student to the newly joined students who logged into the systems of Computer Science Department or the University systems with an associated Student id.
* After generation of the 5 digit number for a student, a record of the Student Username which is FOO\*\*\*\*\*\* ID and 5 digit pair should be visible to student and recorded to some kind of Master file in the certain location of the Server. This master file will be a checkpoint for administrator or the Instructor for further business processing over a period of time.
* User would be given a text file with the name Assignment0.txt in their personal My Documents location to view the generated Pair. ( FOO\*\*\*\*ID--5 digit number)
* For every insertion of the generated pair to the Master file, The Master file records would be sorted in an ascending order according to the 5 digit number.
* User/student would not know that the process of generating Unique 5 digit number.

**Environment Constraints:**

As per System Administrator who is the client of this project, the operating system in which students are allowed to enter is a windows operating system.

Operating System: Windows

**Existing System**

In the current system, There exists a batch file (Some\_Name.bat) and running this batch file on every login phase of the student resulting generation of a unique number. Copying Username and generated number to the Assignment0.txt file in My Documents of every student by creating Assignment0.txt file with adding FOO\*\*\*\*ID and number to it. Updating FOO\*\*\*\*ID and number to the Master file in the certain path in the Server.

**Problems with Existing Solution:**

* This Solution does not provide 5 digits of unique number.
* Does not provide Uniqueness on pair of with FOO\*\*\*\*ID and number in Master file in the certain path in the Server.
* Manual process – needs administrator to check the uniqueness of generated Pair of FOO\*\*\*\*ID and number with Master file
* Does not place the Pair of FOO\*\*\*\*ID and number within Master file by sorting it with respect to unique number.
* This process becomes a Manual Iterative process.

**Proposed System**

**Solution over Existing Constraints:**

As we have an Environment constraint of dealing with Windows Operating System, It would be good to run batch file as windows has a feature of running batch files whatever present in the windows startup folder. As we know, there is a startup folder after each successful case of entry to load desktop, batch files present in the system gets executed.

* If the batch script file which will be placed in the startup folder of windows operating system gets run after loading desktop.

Using the above feature, a sophisticated Solution to accomplish all the requirements would be using batch and Powershell scripting together.

**Why not just batch scripting as exist in Existing System?**

By using just batch scripting, we are able to generate the uniqueness in an instinct but for

* an effective sorting of the Master file and writing code for looping across the Id-Number pairs present in the Master file would be easy to code
* Efficient in terms of uniqueness comparisons and runs as quick as batch file.
* In future, after performance degradation due to constraint of 5 digits in generating unique number will be resolved as quickly as a single line of code change in the Powershell script gives us 2 billion unique combinations where as in batch we get at most 2^32 combination of random numbers.
* Power shell is an advanced version of the batch programming which enable users and administrators to perform effective analysis on the system logs and the development of scripts would be mere to programming in an object oriented language such as JAVA.

**Solution Strategy**

To accomplish all the requirements of this project and concerning about security on the script from the students perspective, Solution strategy as follows.

**Step1.** A minimized run property of .bat file’s shortcut file will be placed into the startup folder of the windows systems. (Startup folder, run property will be explained in further document at **Testing and Sample test Case** section.)

**Step2.** After successful login of the Student, shortcut .bat file in the startup folder calls a master batch file which is stored somewhere in the secure server, the script written in the master batch file calls the Powershell script file which has all the mechanism scripted into it and stored in the secure server location.

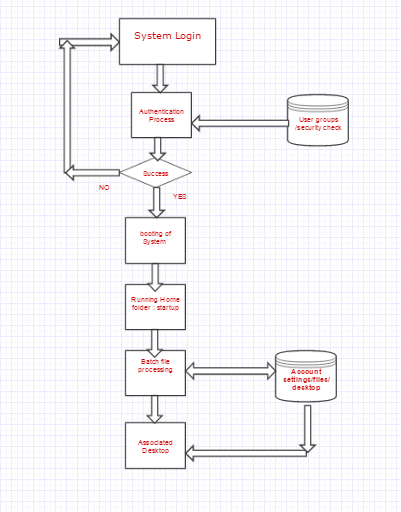
**Step3:** the script or the code present in the Powershell file will does the batch processing block functionality which listed in the below prototype section.

**Benefits of Proposed System:**

* Process is Automated not Manual
* Requirement of Randomness gained well using Powershell features.
* Requirement of 5 digits of unique number accomplished
* Uniqueness among all the FOO\*\*\*ID-5 digit random number ion the records of Master Excel file accomplished
* Efficient sorting after updating the new record into Master Excel file at the correct position based on sorting
* As Master file was chosen as Excel, for business analysis can be done easy from the admin and Instructor perspective

**Prototype and Design flow**

Below flow diagram explains the Solution strategy to accomplish all the requirements listed.

****

**In the above flow chart of Prototype- Batch Processing block functionality as follows:-**

When the windows operating system boots up, after loading Desktop with personal settings of the current user it will run the .bat files in the Startup folder of the System. As the shortcut of the .bat file is present in the Startup folder which will gets executed by calling .bat master file in the server with a minimized version of run capability of the respective .bat master file.

In the .bat master file which will call the Powershell script file present in secure place within server. The power shell script works as follows

**Step1:** checks weather the My Documents location of current User has the Assignment0.txt file or not.

**Step2:** If it is present, Process will be terminated and loads the desktop home page.

**Step3:** If the step1 condition failed, Powershell script creates the random 5 digits of unique number and reads all the Numbers present in the Master file which is Excel (.xlsx file of windows spreadsheet file) file in this solution strategy.

**Step4:** In this step, random number generated would be compared with the numbers available in the Master Excel file.

**Step5:** If the bucket of numbers read from Master Excel file contains the currently generated 5 digits number, program will loop back to generate a 5 digits of unique number and does the step4 repeated until the match not found.

**Step6:** After the Unique pair of FOO\*\*\*\*ID and 5 digit number is formed, generated pair will be appended to the correct position based on the sorting of the records according to the unique number in ascending.

**Step7:** A text file with a name Assignment0.txt will be created under My Documents of the current user. The Key value pair will be written to the created text file and saved.

**Implementation**

Master bat file which calls the power shell content is as follows.

**1. Master.bat**

@echo off

Powershell.exe -executionpolicy remotesigned -File  [\\CHEMNITZ\Applications\a\CSC150\Master\_powershell.ps1](file:///\\CHEMNITZ\Applications\a\CSC150\Master_powershell.ps1)

When the above batch file runs at the startup of an operating system it calls the below Powershell which location is added as a file argument as shown above. Contents of Powershell as follows

2. **Master\_Powershell.ps1**

write-host " Hello $env:UserName "

$localFilepath="\\CHEMNITZ\User Data\$env:UserName\Documents\Assignment0.txt"

$masterFile="\\CHEMNITZ\Applications\a\CSC150\Master.xlsx"

$localUserName=$env:UserName

write-host " local username is $localUserName and path is $localFilepath and the test path is $testpath"

If (Test-Path $localFilepath){

# file with path $path doesn't exist

write-host " local drive already contains Assignment0.txt"

}

else{

$checker = $FALSE

$arrExcelValues = @()

write-host " our checker initial value is $checker"

$objExcel = New-Object -ComObject Excel.Application

#$objExcel = new-object -comobject excel.application

$objExcel.Visible = $False

$xlLastCell = [Microsoft.Office.Interop.Excel.Constants]::xlLastCell

$objWorkbook = $objExcel.Workbooks.Open($masterFile)

$objWorksheet = $objWorkbook.Worksheets.Item(1)

$i = 2

Do {

$arrExcelValues += $objWorksheet.Cells.Item($i, 2).Value()

$i++

}

While ($objWorksheet.Cells.Item($i,2).Value() -ne $null)

Do{

#$Number = get-random

$Number = get-random -Maximum 99999 -Minimum 10000

write-host " random unique number is $Number"

foreach ($objItem in $arrExcelValues) {

if($Number -eq $objItem){

$checker = $TRUE

write-host " our checker present inside value is $checker"

break

}

write-host $objItem

}

write-host " our checker final value is $checker"

}

While ($checker -eq $TRUE)

write-host " writing to network Master File ==start==> "

$objWorksheet.activate()

$objRange = $objWorksheet.UsedRange

$lastRow = $objRange.SpecialCells($xlLastCell).Row+1

$objWorksheet.Cells.Item($lastRow,1) = $localUserName

$objWorksheet.Cells.Item($lastRow,2) = $Number

write-host " writing to network Master File ==completed==> "

#sort begins

$xlSortOnValues = $xlSortNormal = 0

$xlTopToBottom = 1

$xlAscending = 1

$last = $objWorksheet.UsedRange.SpecialCells(11).Address($False,$false)

#$xlCellTypeLastCell = 11

$range1 = $objWorksheet.range("A2:$last" )

$range2 = $objWorksheet.range("B2")

$range3 = $objWorksheet.range("A2")

#two-column sort ---> works

$objWorksheet.sort.sortFields.clear()

$objWorksheet.sort.sortFields.add($range2, $xlSortOnValues, $xlAscending,`

$xlSortNormal)

$objWorksheet.sort.sortFields.add($range3, $xlSortOnValues, $xlAscending,`

$xlSortNormal)

$objWorksheet.sort.setRange($range1)

$objWorksheet.sort.header = $xlNo

$objWorksheet.sort.orientation = $xlTopToBottom

$objWorksheet.sort.apply()

$objWorkbook.Save()

#sort end

$objWorkbook.Close()

$null =$objExcel.Quit()

$null =[System.Runtime.Interopservices.Marshal]::ReleaseComObject($objExcel)

#Stop-Process -Name objExcel -Force

write-host " writing result to local Assignment0.txt start==>"

$result = "$env:UserName - $Number"

write-output $result | add-content $localFilepath

write-host " writing result to local Assignment0.txt done"

write-host " this is the end"

}

**Testing and Sample test Case**

For test case sample, New Student credentials will be

**Username:** cscsc313

**Password:** will be obtained from System administrator.

**Step1:** The content of Master Batch file is updated with the Powershell file location as follows

@echo off

Powershell.exe -executionpolicy remotesigned -File  \\CHEMNITZ\Applications\a\CSC150\Shiva\_powershell2.ps1

pause

-pause is to hold the terminal to observe the content written to the terminal ans asks user to press any key to close it and above script is saved into file location: [\\CHEMNITZ\User Data\csc313\Documents\Medium\_Shiva\test.bat](file:///\\CHEMNITZ\User%20Data\csc313\Documents\Medium_Shiva\test.bat)

**Step2:**

The content of the Shiva\_Powershell2.ps1 is changed according to the test case. Those changes are

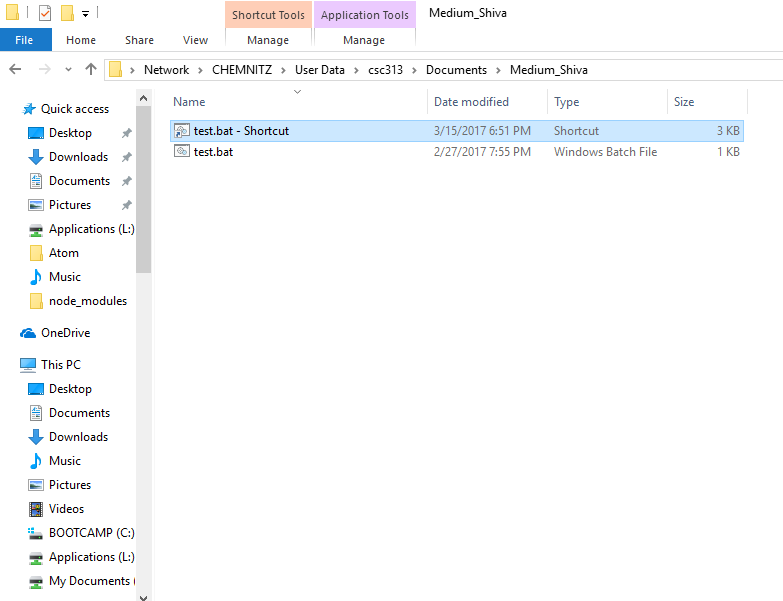
**At line2:** $localFilepath=”[\\CHEMNITZ\User Data\csc313\Documents\Medium\_Shiva\Assignment0.txt](file:///\\CHEMNITZ\User%20Data\csc313\Documents\Medium_Shiva\Assignment0.txt)”

**At line 3:** $testpath="\\CHEMNITZ\User Data\$env:UserName\Documents\Medium\_Shiva\Assignment0.txt "

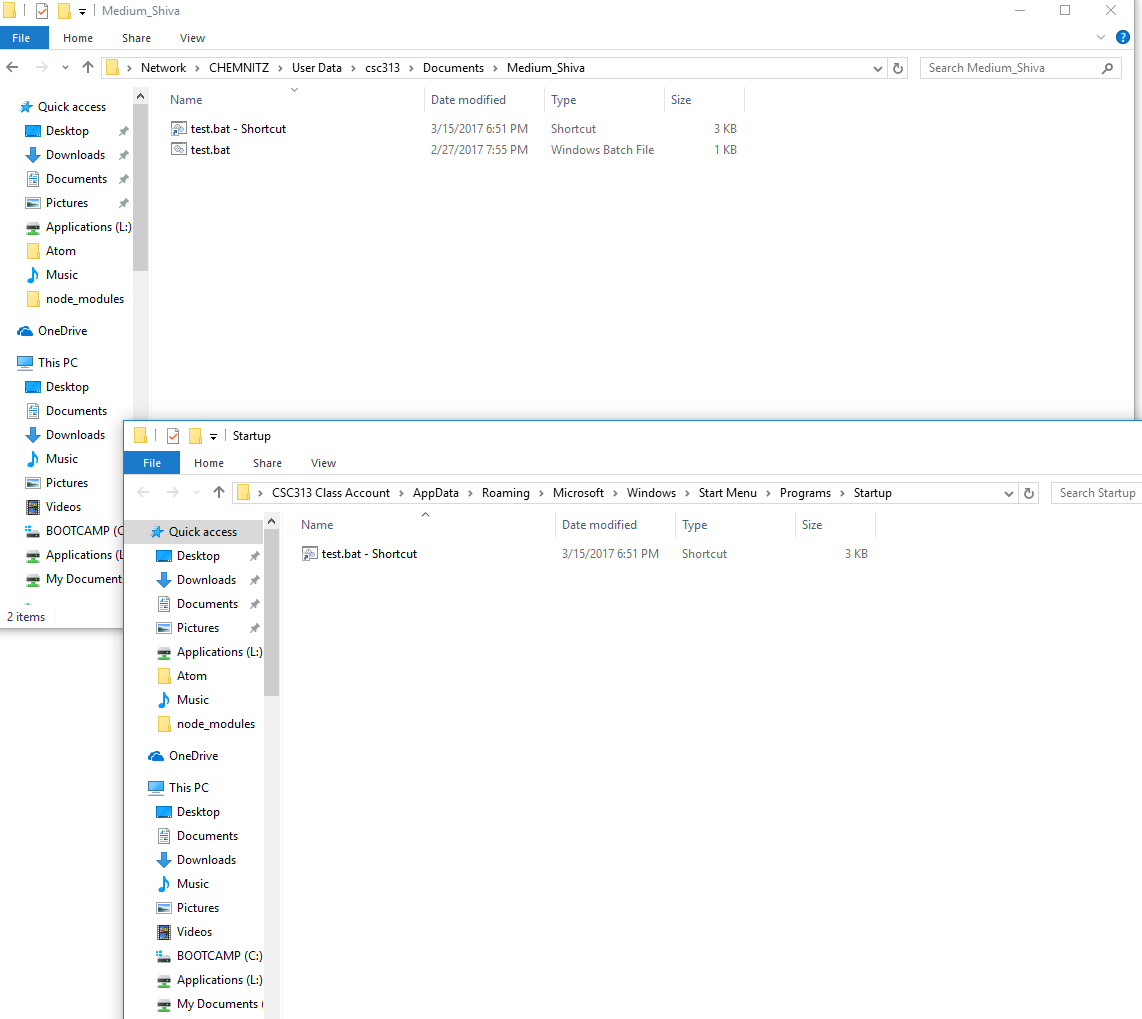
**At line 4:**$masterFile=[\\CHEMNITZ\Applications\a\CSC150\Master\_Shiva.xlsx](file:///\\CHEMNITZ\Applications\a\CSC150\Master_Shiva.xlsx) and rest of the script is unchanged and shown in the implementation section.

**Step3:**

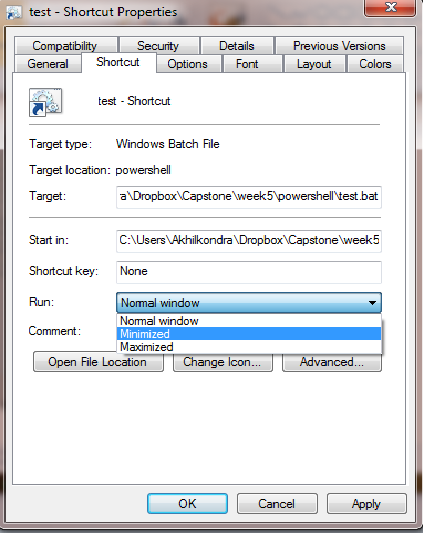
From the test.bat file location, By right clicking on the test.bat file , selecting create short cut results the shortcut file of the test.bat into the same location as shown below.



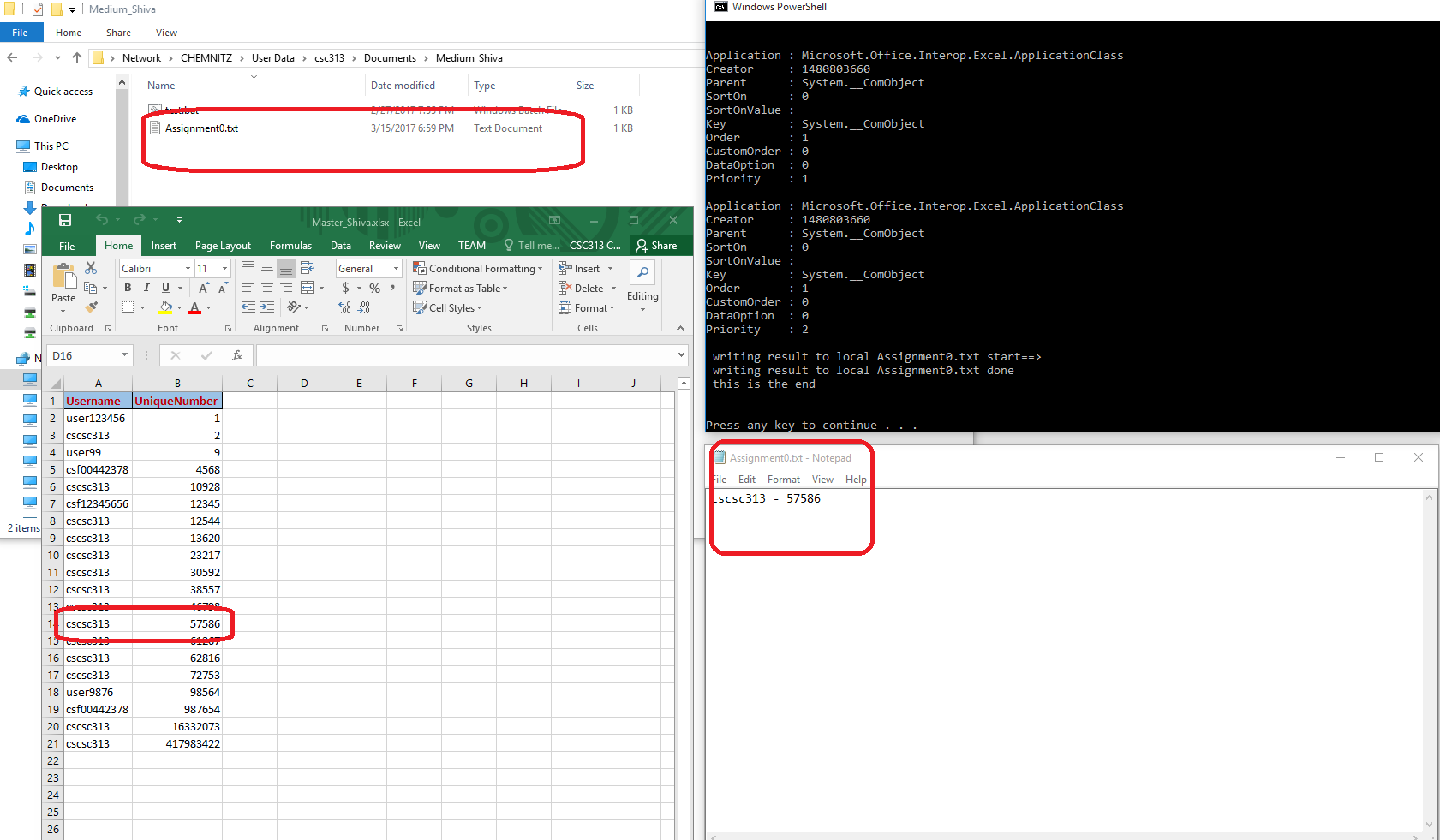
**Step4:** By running “run” command in the search file options on clicking start button in windows opens the Run terminal, and typed shell:startup which will open the startup folder of the current windows system and copy the short cut test.bat file into it as shown below.



**Step5:** After copying shortcut file to the startup folder, Right click on the shortcut test bat file, and select minimized option in RUN. By doing this, a command prompt will be opened as a taskbar item and does the functionality of the batch file and closes within short span of time. By doing this we can avoid User feeling that something went in the command prompt. Running .bat file using minimized phase can be accomplished as explained above and a sample screenshot as follows.



**Sample Output:** By login into the system with the credentials available, a sample output comparison can be seen in the below screenshot



By observing above screen, we can see text file named Assignment0.txt is stored into the destined location of the Local Drive of User and the random 5 digit unique number is stored into the Master Excel file in the Server location.

**Future Enhancements**

* Performance improvements in terms of effective algorithm for looping all the records available in the Master excel file towards checking the Uniqueness of the generated Unique 5 digit number.
* Implementation of Efficient algorithm for sorting of records in Master excel using Powershell

# Bibliography

Jofre, J. (2016, 12 14). *Getting Started with Windows PowerShell*. Retrieved from https://msdn.microsoft.com: https://msdn.microsoft.com/powershell/scripting/getting-started/getting-started-with-windows-powershell

Marufuzzaman, M. (2010, 8 21). *A Quick Guideline for Microsoft Windows PowerShell*. Retrieved from https://www.codeproject.com: https://www.codeproject.com/kb/powershell/powershell\_guideline\_p3.aspx

microsoft. (2017). *Windows PowerShell Reference*. Retrieved from https://msdn.microsoft.com: https://msdn.microsoft.com/en-us/library/ms714469(v=vs.85).aspx

Urkec. (2011, 1 15). *Why Use PowerShell in Excel?* Retrieved from http://excelposh.codeplex.com: http://excelposh.codeplex.com/documentation