File Server Synchronization System

Members: (Batch B6)

Nirmit Goyal 9913103619

Varun Gambhir 13103603

Abstract:

F.S.S.S. overcomes the problem of manually updating the file server of any institute/organization. We used SMB Client module of python to access a remote Samba Server. This system will sync all the directories/files, which the user wants. The user can specify the directories which he wants to sync using check box. This system's capability also enables the user to upload his files to his registered directory on the remote server. Using crontab process of linux, this software runs after a particular interval of time.

Objectives:

1. Auto Sync. of a remote Samba server. Demonstration will be done by synchronizing fileserver2.pdc.jiit.
2. Enable user to upload files present on his local machine to his directory on the remote server and vice-versa conveniently.
3. Time interval to be set using linux's *crontab* process.
4. Used *gnome-panel* of linux's GUI to run the software using icon.

Technologies used:

1. Python 2.7

2. SMB Client module to access Samba Server
3. Tix, ttk, Tkinter module for GUI
4. Crontab process of linux
5. subprocess, soldier module to run linux commands

Division of project work:

Nirmit Goyal:

1.Synchronization of files and directories using SMB Client module of python to access a remote Samba server.

2.Download filesystem after mounting the samba fileserver. Upload local files on student's remote registered directory on the fileserver.

3.Used *gnome-panel* of linux's GUI to run the software using icon.

Varun Gambhir:

1.Used dictionary to simulate dirtree with checkboxes to display directory listing of files of the remote samba server. Saved paths of directories checked by the user in a text file on the local computer.

2.Used *tix, tkinter, ttk* module of python to build the GUI for the software.

3.Made use of crontab process in linux to run a linux command after a particular interval of time.Time interval to be set by the user.

Code:

Full code and Documentation is available @ https://github.com/varungambhir/File-Server-Sync-System-

Small piece of code:

```
import smbclient
import os
import operator
import datetime
import getpass
import subprocess
import time
import re

share = """Study Material""" # # for Study Material
# share = """9913103619"""  ## for my directory

fp=open("Details.txt",'r')
data = fp.read().splitlines()
server = data[0]
username = data[1]
password = data[2]
syspass=data[3]
domain = "pdc.jiit"
user = getpass.getuser()
attach = """/home/""" + user + """/Documents"""
smb = smbclient.SambaClient(server, share, username, password, domain)
local_file = []
local_dir = []
remote_file = []
remote_dir=[]
sorted_local_dir=[]
local_dir_dict={}

# print server,username,password
def pathtodir(path):
    if not os.path.exists(path):
        l = []
        p = "/"
        l = path.split("/")
        i = 1
        while i < len(l):
            p = p + l[i] + "/"
            i = i + 1
            if not os.path.exists(p):
                os.mkdir(p)

with open("/home/" + user + "/File-Server-Sync-System-/path_file.txt", "r+") as f2:
    all_paths = [line.rstrip('\n') for line in f2]

for item in all_paths:
    # print item
    if '.' in item:
        remote_dir.append( item.rsplit('/', 1)[0])
remote_dir=list(set(remote_dir))
# for i in remote_dir:
#     print i
```

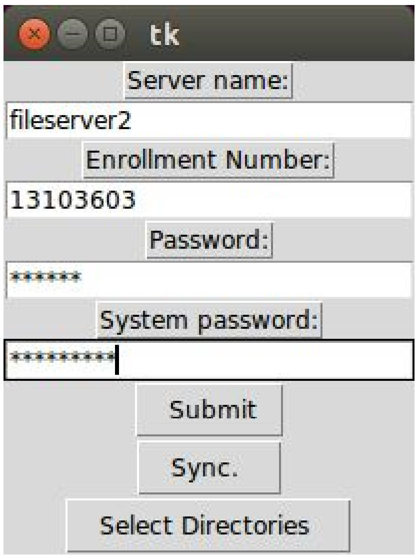
```
for item in remote_dir:
        if smb.exists(item):
                local_dir.append(attach+item)
for i in local_dir:
        # print i
        local_dir_dict[i]=i.count('/')
sorted_local_dir = sorted(local_dir_dict.items(), key=operator.itemgetter(1))
local_dir = [x[0] for x in sorted_local_dir]
pathtodir(os.path.dirname(local_dir[0]))
for i in local_dir:
        # print i
        if not os.path.exists(i):
      os.mkdir(i)

for item in remote_dir:
        if smb.exists(item):
                for file in smb.listdir(item):
                        if smb.isfile(item+"/"+file):
                                # print file
                                remote_file.append(item+"/"+file)
                                local_file.append(attach+item+"/"+file)
for i, j in zip(remote_file, local_file):
    dict_ = smb.info(i)
    str_remote = re.sub(' +', ' ', dict_["write_time"])
    remote_last_mod = str_remote.split(" ")[3]
    f = open(j, "w+")
    str_local = re.sub(' +', ' ', time.ctime(os.path.getmtime(j)))
    local_last_mod = str_local.split(" ")[3]
    if remote_last_mod != local_last_mod:
        smb.download(i, j)
        f.close()

os.system('notify-send "Your SM is now Synchronized with the remote server !!!" ')
f.close()
fp.close()
```

Snapshots and Testing:

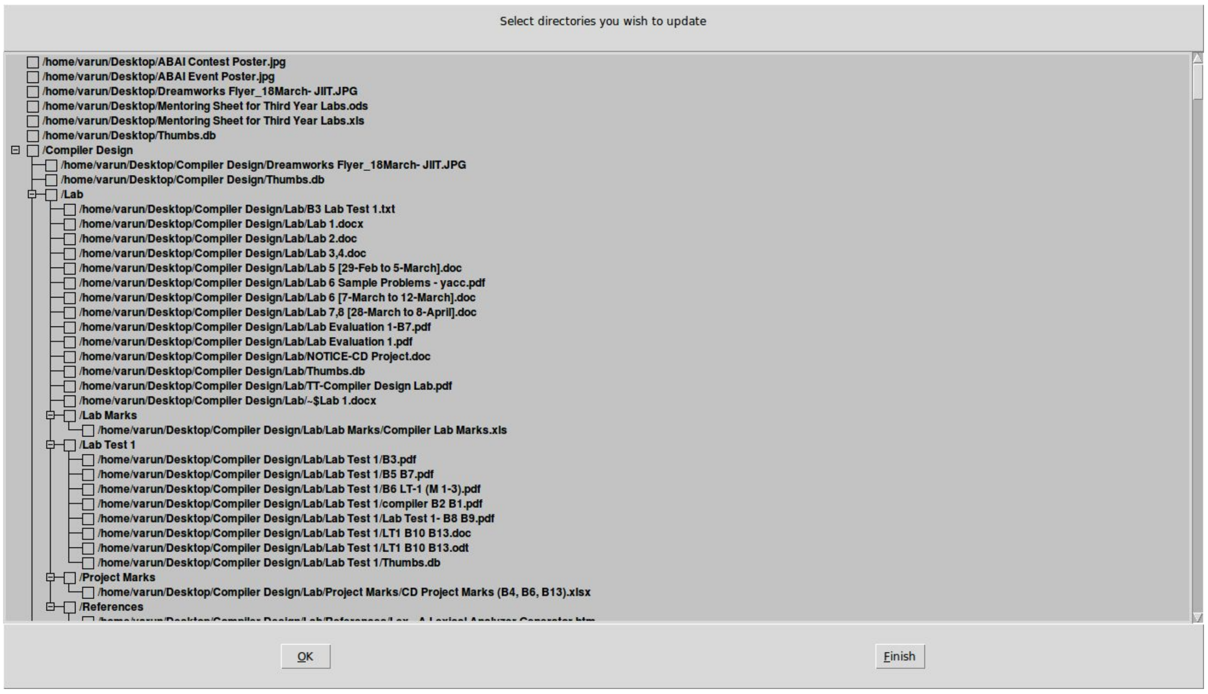
1) Index page(index.py)



tk

Server name:
fileserver2
Enrollment Number:
13103603
Password:
******
System password:
****

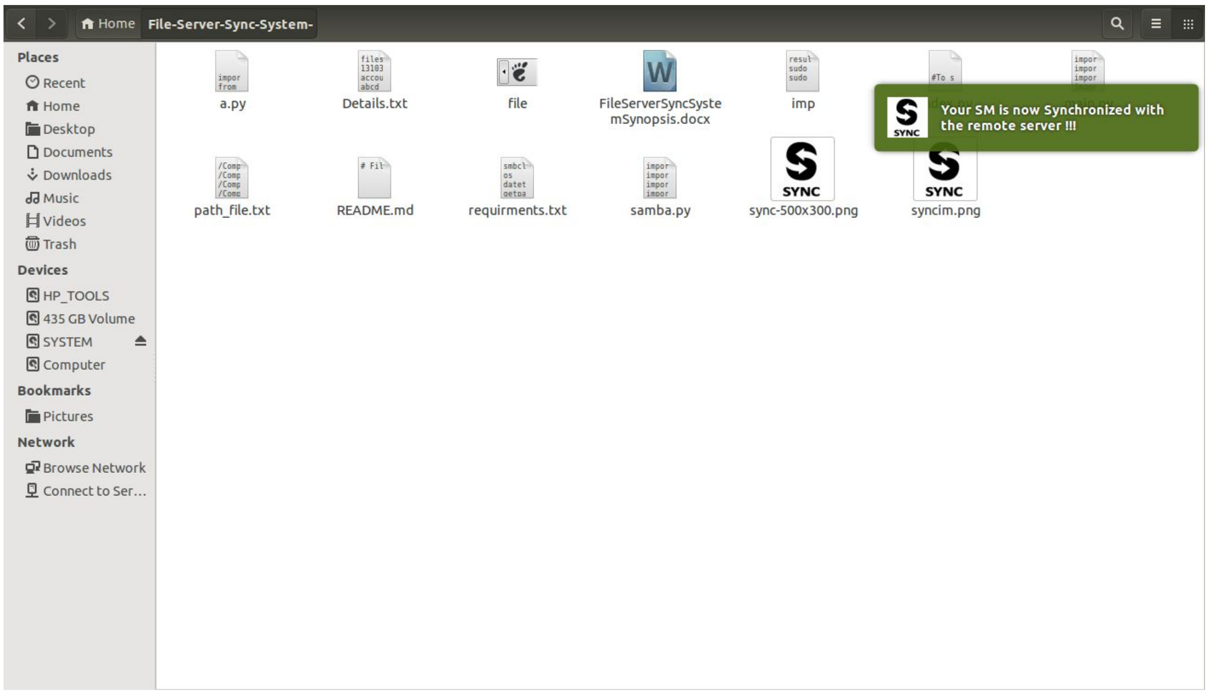
2) Directory tree view(main.py)



Select directories you wish to update

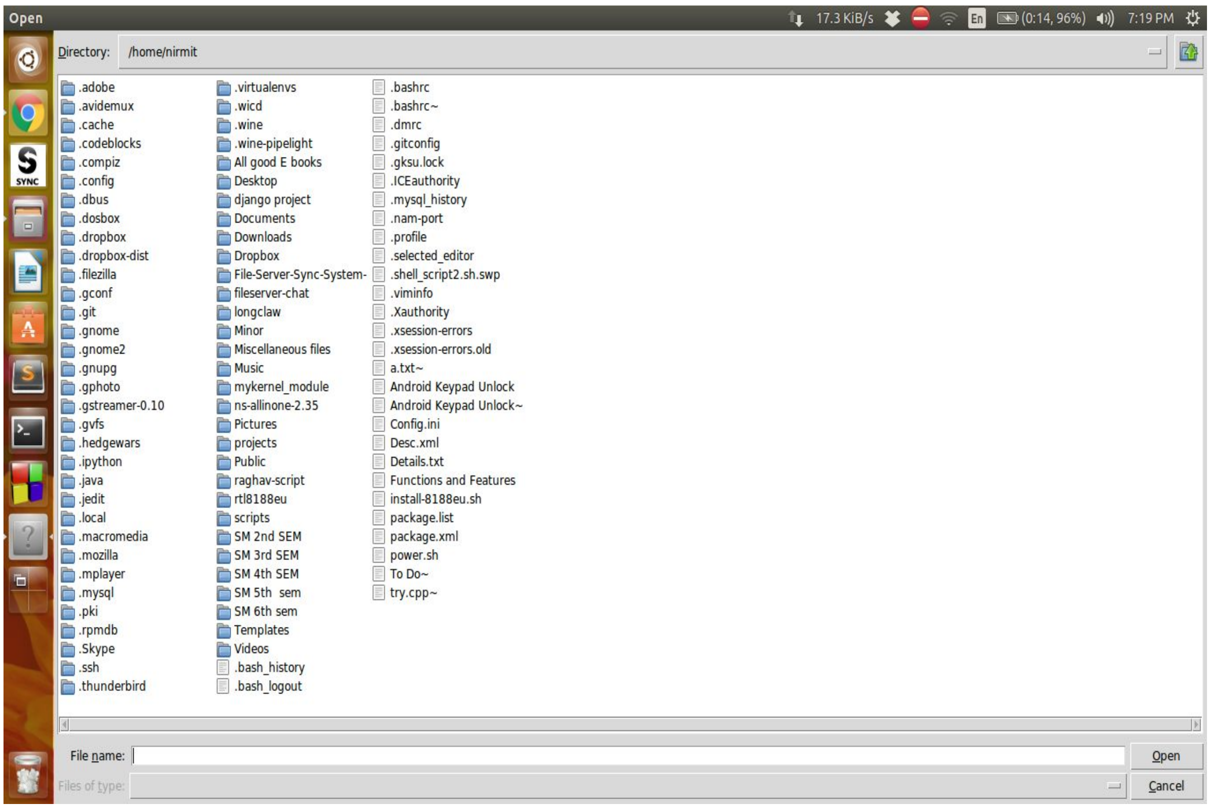
```
/home/varun/Desktop/ABAI Contest Poster.jpg
/home/varun/Desktop/ABAI Event Poster.jpg
/home/varun/Desktop/Dreamworks Flyer_18March- JIIT.JPG
/home/varun/Desktop/Mentoring Sheet for Third Year Labs.ods
/home/varun/Desktop/Mentoring Sheet for Third Year Labs.xls
/home/varun/Desktop/Thumbs.db
/Compiler Design
  /home/varun/Desktop/Compiler Design/Dreamworks Flyer_18March- JIIT.JPG
  /home/varun/Desktop/Compiler Design/Thumbs.db
  /Lab
    /home/varun/Desktop/Compiler Design/Lab/B3 Lab Test 1.txt
    /home/varun/Desktop/Compiler Design/Lab/Lab 1.docx
    /home/varun/Desktop/Compiler Design/Lab/Lab 2.doc
    /home/varun/Desktop/Compiler Design/Lab/Lab 3,4.doc
    /home/varun/Desktop/Compiler Design/Lab/Lab 5 [29-Feb to 5-March].doc
    /home/varun/Desktop/Compiler Design/Lab/Lab 6 Sample Problems - yacc.pdf
    /home/varun/Desktop/Compiler Design/Lab/Lab 6 [7-March to 12-March].doc
    /home/varun/Desktop/Compiler Design/Lab/Lab 7,8 [28-March to 8-April].doc
    /home/varun/Desktop/Compiler Design/Lab/Lab Evaluation 1-B7.pdf
    /home/varun/Desktop/Compiler Design/Lab/Lab Evaluation 1.pdf
    /home/varun/Desktop/Compiler Design/Lab/NOTICE-CD Project.doc
    /home/varun/Desktop/Compiler Design/Lab/Thumbs.db
    /home/varun/Desktop/Compiler Design/Lab/TT-Compiler Design Lab.pdf
    /home/varun/Desktop/Compiler Design/Lab/~$Lab 1.docx
    /Lab Marks
      /home/varun/Desktop/Compiler Design/Lab/Lab Marks/Compiler Lab Marks.xls
    /Lab Test 1
      /home/varun/Desktop/Compiler Design/Lab/Lab Test 1/B3.pdf
      /home/varun/Desktop/Compiler Design/Lab/Lab Test 1/B5 B7.pdf
      /home/varun/Desktop/Compiler Design/Lab/Lab Test 1/B6 LT-1 (M 1-3).pdf
      /home/varun/Desktop/Compiler Design/Lab/Lab Test 1/compiler B2 B1.pdf
      /home/varun/Desktop/Compiler Design/Lab/Lab Test 1/Lab Test 1- B8 B9.pdf
      /home/varun/Desktop/Compiler Design/Lab/Lab Test 1/LT1 B10 B13.doc
      /home/varun/Desktop/Compiler Design/Lab/Lab Test 1/LT1 B10 B13.odt
      /home/varun/Desktop/Compiler Design/Lab/Lab Test 1/Thumbs.db
    /Project Marks
      /home/varun/Desktop/Compiler Design/Lab/Project Marks/CD Project Marks (B4, B6, B13).xlsx
    /References
```

3)Synchronization of remote samba server successful



Your SM is now Synchronized with the remote server !!!

4) Upload file using GUI



Directory: /home/nirmit

```
.adobe            .virtualenvs              .bashrc
.avidemux         .wicd                     .bashrc~
.cache            .wine                     .dmrc
.codeblocks       .wine-pipelight           .gitconfig
.compiz           All good E books          .gksu.lock
.config           Desktop                   .ICEauthority
.dbus             django project            .mysql_history
.dosbox           Documents                 .nam-port
.dropbox          Downloads                 .profile
.dropbox-dist     Dropbox                   .selected_editor
.filezilla        File-Server-Sync-System-  .shell_script2.sh.swp
.gconf            fileserver-chat           .viminfo
.git              longclaw                  .Xauthority
.gnome            Minor                     .xsession-errors
.gnome2           Miscellaneous files       .xsession-errors.old
.gnupg            Music                     a.txt~
.gphoto           mykernel_module           Android Keypad Unlock
.gstreamer-0.10   ns-allinone-2.35          Android Keypad Unlock~
.gvfs             Pictures                  Config.ini
.hedgewars        projects                  Desc.xml
.ipython          Public                    Details.txt
.java             raghav-script             Functions and Features
.jedit            rtl8188eu                 install-8188eu.sh
.local            scripts                   package.list
.macromedia       SM 2nd SEM                package.xml
.mozilla          SM 3rd SEM                power.sh
.mplayer          SM 4th SEM                To Do~
.mysql            SM 5th  sem               try.cpp~
.pki              SM 6th sem
.rpmdb            Templates
.Skype            Videos
.ssh              .bash_history
.thunderbird      .bash_logout
```