

1. Mp3Recorder, Python demo project

Android 11 Python app to record and email mp3 files.

2. Design and component review

- Android KivyMD framework app gui interface
- User buttons drive Record, Email, EmailFile and Exit.
- Timed log activity updates.

===== Mp3Recorder Main Screen [Wifi UP] [Info log] =====

[Mp3Recorder Thu Aug 3 15:24:07 2023] [Wifi * UP *]

START Recording

Email

Email File

Exit

[===== Thu Aug 3 15:23:28 2023 =====]

[===== Thu Aug 3 15:23:51 2023 =====]

[Email File : Mp3Recorder_25Jul2023_174306.mp3]

===== Mp3Recorder Main Screen [Wifi DN] =====

[Mp3Recorder Thu Aug 3 15:32:26 2023] [Wifi - DN -]

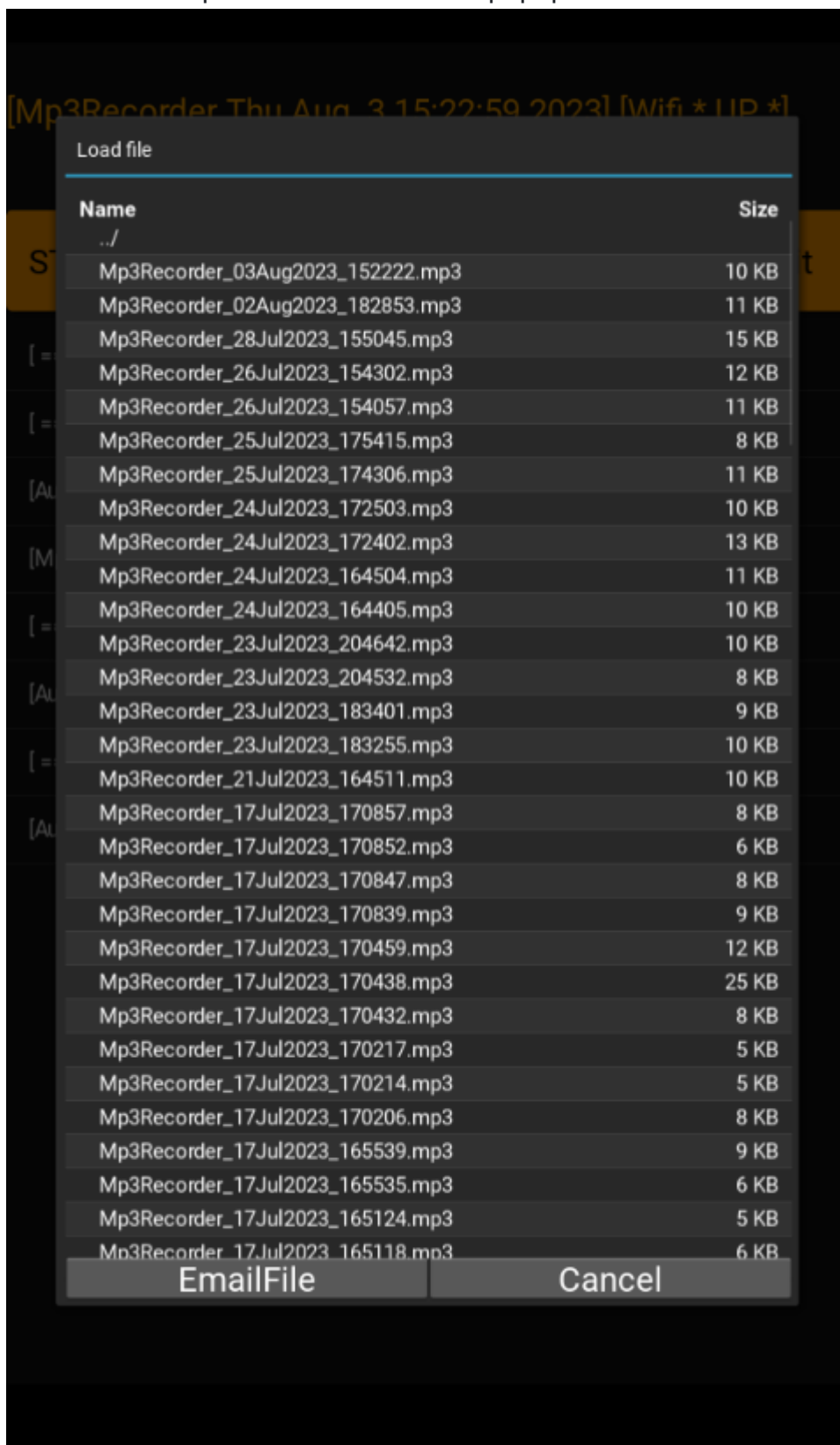
START Recording

Email

Email File

Exit

===== Mp3Recorder Email File popup screen =====



3. Code Review

- README.md docs:

<<https://code.visualstudio.com/Docs/languages/markdown>>

<<https://www.youtube.com/watch?v=jeOfS90Flf8&t=515s>>

<<https://marketplace.visualstudio.com/items?itemName=manuth.markdown-converter>>

- Python classes:

main.py

- buildozer requirement 'main.py' entry
- class Mp3RecorderApp(MDApp):
 - package name, lowercase, in buildozer.spec.
 - added kivymd color scheme
- class Mp3Recorder(MDBoxLayout):
 - apk mainline class linked to mp3recorder.ky <Mp3Recorder> function
 - def __init__
 - initiallize record state, scroll view, permissions, timer.
 - def timer()
 - Update current time label, with added wifi indication.
 - def record()
 - Applies start and stop recording BUTTON, calls Recorder() class
 - def email()
 - Email the latest recorded file, via Recorder().email().
 - def emailfile()
 - File chooser popup for mp3 file selection, mp3recorder.ky <LoadFile>
 - def exit()
 - Call Recorder() to close app

recorder.py

- Configured at Recorder() class initalization via the Documents/Mp3Recorder.ky
- def __init__
 - android MediaRecorder assignment
 - def configInit()
 - App configuration via Mp3Recorder.csv, called after permissions have been granted
 - def record_start()
 - Called from <Mp3Recorder> initial 'START Recording' button, checks if recording is possible
 - def record_stop()
 - 'STOP Recording' button hit, save current recorded filename, stop recording
 - def email()
 - Called from Mp3Recorder().email(filename), call Recorder().send_email() to actually email the file.
 - def exit()
 - Called by Mp3Recorder().exit() to actually exit the app.

sharedstorage.py

- source: <<https://github.com/Android-for-Python/androidstorage4kivy>>
Exposes Android 11 shared storage files, copies the default configuration

- Kivy configuration:

mp3recorder.kv

- Support files:

Mp3Recorder.csv
Mp3Recorder.png
README.md

- Build files:

buildozer.spec

Installation: <<https://pypi.org/project/buildozer/>>

Default 'buildozer.spec' changes:

title = Mp3Recorder
header of apk filename

```

package.name = mp3recorderapp
package.domain = mp3recorderapp
# dir included in apk build
source.include_exts = py,png,jpg,kv,atlas,svg,mp3
# dir excluded in apk build
source.exclude_dirs = tests, bin, venv, .venv, save*
# apk version number
version = 4.5
# package requirements, during Android build
requirements = python3,kivymd,jnius,android,plyer,pydub,ffmpeg,pin
# Android app orientation
orientation = portrait
# Android app fullscreen
fullscreen = 0
# Android app permissions
# MANAGE_EXTERNAL_STORAGE not supported, requires manual enable, F
android.permissions = INTERNET, RECORD_AUDIO, READ_EXTERNAL_STORAGE
# Minimum versions work with tested tablet.
android.api = 30
android.minapi = 21
android.archs = armeabi-v7a
ios.ios_deploy_url = https://github.com/phonegap/ios-deploy
ios.ios_deploy_branch = 1.7.0

requirements.txt
Visual Studio python build environment generated from working virtualenv
$ pip freeze > requirements.txt
New environment generated withing a virtualenv:
$ pip install -r requirements.txt
.gitignore
gitHub files to ignore.

```

4. Usage

- [START Recording] changes to [STOP Recording], say something to record.
- [STOP Recording] stops the recording and changes back to [START Recording] for
- [Email] will email the current recorded file to the Mp3Recorder.csv 'Receiver'
- [Email File] brings up a popup list of previously recorded files, tap/highlight
- [Exit] quit the app.

5. Buid Mp3Recorder.apk

- Android app, Mp3Recorder.apk, is created by Buildozer on Linux
- Install Unbutu Linux on Windows:
<https://ubuntu.com/tutorials/install-ubuntu-on-wsl2-on-windows-11-with-gui>
<https://stackoverflow.com/questions/28210637/unable-to-ssh-localhost-permission-denied>

NOTE this Windows WSL configuration requirement:
<https://docs.microsoft.com/en-us/windows/wsl/install>.

If you're running Ubuntu on Windows Subsystem for Linux, there will not be
 If you don't already have openssh-server installed:

1. `sudo apt-get upgrade`
2. `sudo apt-get update`
3. `sudo apt-get install openssh-server`
4. `sudo service ssh start`

Then take the following steps to enable sshing to localhost:

1. `cd ~/.ssh`
2. `ssh-keygen` to generate a public/private rsa key pair; use the default
3. `cat id_rsa.pub >> authorized_keys` to append the key to the authorized_keys file
4. `chmod 640 authorized_keys` to set restricted permissions
5. `sudo service ssh restart` to pickup recent changes
6. `ssh localhost`

- Install Visual Code,
<https://linuxize.com/post/how-to-install-visual-studio-code-on-ubuntu-20-04/>
- Install Python
<https://linuxize.com/post/how-to-install-python-on-ubuntu-22-04/>
- Install PIP
<https://linuxize.com/post/how-to-install-pip-on-ubuntu-20.04/>
- Install Buildozer
<https://buildozer.readthedocs.io/en/latest/installation.html>
 Setup a virtual environment within Visual Code, which containerizes the build process.
 Visual Code->Terminal->New Terminal.

```
$ virtualenv .venv
dovczitter@DESKTOP-N4F7Q65:~/Mp3Recorder$ source .venv/bin/activate
(.venv) dovczitter@DESKTOP-N4F7Q65:~/Mp3Recorder$
To create a new buildozer.spec file, run
$ buildozer init
Otherwise, use the one buildozer.spec in gitHub, or edit the new one line by line
```

Assuming *.py, *.csv, *.png and .gitignore are installed, to generate the buildozer.spec file

```
$ buildozer -v android debug
```

First build will take many minutes, must see the ending 'BUILD SUCCESSFUL'
 Next build should be less than a minute, to fully rebuild, 'rm -rf .bin'.

6. Installation

- I used this tablet:
 YQSAVIOR Tablet Android 11 Tablets 7 inch
 Available on Amazon, \$42.99
https://www.amazon.com/gp/product/B09LHMMJ2Q/ref=pe_2313390_735257480_em_
- Install PlayStore 'File Manager Plus', free
- Install PlayStore 'OutLook' free, create and save your new email username and password
- Enable tablet for debug mode, which allows for direct usb

<<https://www.youtube.com/watch?v=CCFMai4JmeM>>

- File transfer the Mp3Recorder.apk to the Android 'Downloads' folder
- NOTE - Windows explorer to the tablet's 'Internal shared storage' on connect
- Enable the tablet's Wifi.
- Suggest that you edit the Mp3Recorder.csv file and copy via explore to the tablet
- Requires a tablet 'outlook' account, install via PlayStore.
- Mp3Recorder.csv configuration, <key>,<value>:
 - Outlook host and port:
 - Host,smtp-mail.outlook.com
 - Port,587
 - Outlook login username:
 - Username,myusername@outlook.com
 - Outlook login password:
 - Password,@Mypassword
 - Outlook email 'From':
 - Sender,myusername@outlook.com
 - Outlook email 'To' list:
 - Receiver,myusername@outlook.com,buddy1@addr1.com,buddy2@addr2.com,
 - Recorded base filename:
 - Mp3FileRoot,Mp3RecorderTest
- Run the app via 'File Manager +', then 'Downloads', tap the mp3recorderapp.apk
- Choose 'Install'
- Choose 'Open'
- If all works, the app comes up, top line shows per-second update, and [Wifi]
- Nice to do, would be the enable premission 'Allow management of all files'.

Two configuration choices:

- builddozer with modified Mp3Recorder.csv
 - delete 'Main storage > Documents > Mp3Recorder > Mp3Recorder.csv'
 - Uninstall current [File Manager > Apps > Mp3Recorder | UNINSTALL]
 - Install new Mp3Recorder.apk via 'File Manager +'
- or-
- 'Allow management of all files', see above
 - USB copy Windows edited Mp3Recorder.csv to the tablet's 'Internal shared storage'