pip install kivy

from kivy.app import App

from kivy.uix.boxlayout import BoxLayout

from kivy.uix.label import Label

from kivy.uix.button import Button

from kivy.uix.timepicker import TimePicker

from plyer import notification

import time

from threading import Thread

class PhoneSilencerApp(App):

def build(self):

self.time\_picker = TimePicker()

self.status\_label = Label(text="Silencer is currently inactive", font\_size=18)

self.activate\_button = Button(text="Activate Silencer", on\_press=self.activate\_silencer)

self.layout = BoxLayout(orientation="vertical", spacing=10, padding=10)

self.layout.add\_widget(self.time\_picker)

self.layout.add\_widget(self.status\_label)

self.layout.add\_widget(self.activate\_button)

return self.layout

def activate\_silencer(self, instance):

selected\_time = self.time\_picker.time

current\_time = time.localtime(time.time())

activation\_time = time.struct\_time(

(current\_time.tm\_year, current\_time.tm\_mon, current\_time.tm\_mday,

selected\_time.hour, selected\_time.minute, 0, 0, 0, 0)

)

time\_difference = time.mktime(activation\_time) - time.mktime(current\_time)

if time\_difference <= 0:

self.status\_label.text = "Please select a future time."

return

self.status\_label.text = "Silencer activated until {}".format(time.strftime("%H:%M", activation\_time))

notification.notify(

title="Phone Silencer Activated",

message="Your phone will be silent until {}".format(time.strftime("%H:%M", activation\_time)),

)

# Run the silencer in a separate thread to avoid blocking the main UI

silencer\_thread = Thread(target=self.silence\_phone, args=(time\_difference,))

silencer\_thread.start()

def silence\_phone(self, duration):

time.sleep(duration)

notification.notify(

title="Phone Silencer Deactivated",

message="Your phone silencer has been deactivated.",

)

self.status\_label.text = "Silencer is currently inactive"

if \_\_name\_\_ == "\_\_main\_\_":

PhoneSilencerApp().run()