

Achiziții de Date

Aplicație pentru PC de înregistrare a
datelor despre baterie cu afișaj grafic

Pantazică Viviana – Ștefania,
Grupa 324 CB

Universitatea Politehnică București
Facultatea de Automatică și Calculatoare
2021 – 2022

1. Scopul proiectului

Proiectul a constat în realizarea unei aplicații pentru PC care să poată înregistra și afișa date importante despre baterie, precum:

- Procentajul bateriei,
- Dacă PC-ul este băgat la încărcat sau nu,
- Timpul aproximativ rămas până la descărcarea completă a bateriei.

2. Prezentarea aplicației

Aplicația este relativ simplă, deoarece se bazează pe extragerea datelor generate de sistemul de operare și afișarea acestora cu ajutorul unei interfețe grafice. Am folosit pentru implementarea acesteia Visual Studio Code, cu rulare în fereastră interactivă. Limbajul ales a fost Python, datorită bibliotecilor „psutil”, care face posibilă accesarea datelor despre senzori, respectiv „tkinter”, pentru afișajul grafic.

3. Prezentarea codului

```
# python script showing battery details
from time import time
import psutil

# function returning time in hh:mm:ss
def convertTime(seconds):
    minutes, seconds = divmod(seconds, 60)
    hours, minutes = divmod(minutes, 60)
    return "%d:%02d:%02d" % (hours, minutes, seconds)

# returns a tuple
battery = psutil.sensors_battery()

percentage = "Battery percentage : ", battery.percent

isPluggedIn = "Power plugged in : ", battery.power_plugged

# converting seconds to hh:mm:ss
timeLeft = "Battery left : ", convertTime(battery.secsleft)
```

Conform documentației:

psutil.sensors.battery() returns a named tuple consisting of following values: percent, seclft, power_plugged:

If no battery is installed or metrics can't be determined None is returned.

percent: Power left in percentage.

secsleft: Approx seconds left before the power runs out.

It is set to psutil.POWER_TIME_UNLIMITED if it is on charging.

If this value can't be determined it is set to psutil.POWER_TIME_UNKNOWN

.

power_plugged: True if power is plugged in, False if it isn't charging or None if it can't be determined.

```
# GRAPHIC DISPLAY
from tkinter import *

root = Tk()

label = Label(
    text="Battery Info Display",
    foreground="white",
    background="black"
)
label.pack()

def changetext():
    a.config(text=timeLeft)

a = Label(
    root,
    text=isPluggedIn,
)

a.configure(width = 100,
            foreground = "white",
            background = "orange",
            activebackground = "#33B5E5",
            relief = FLAT)

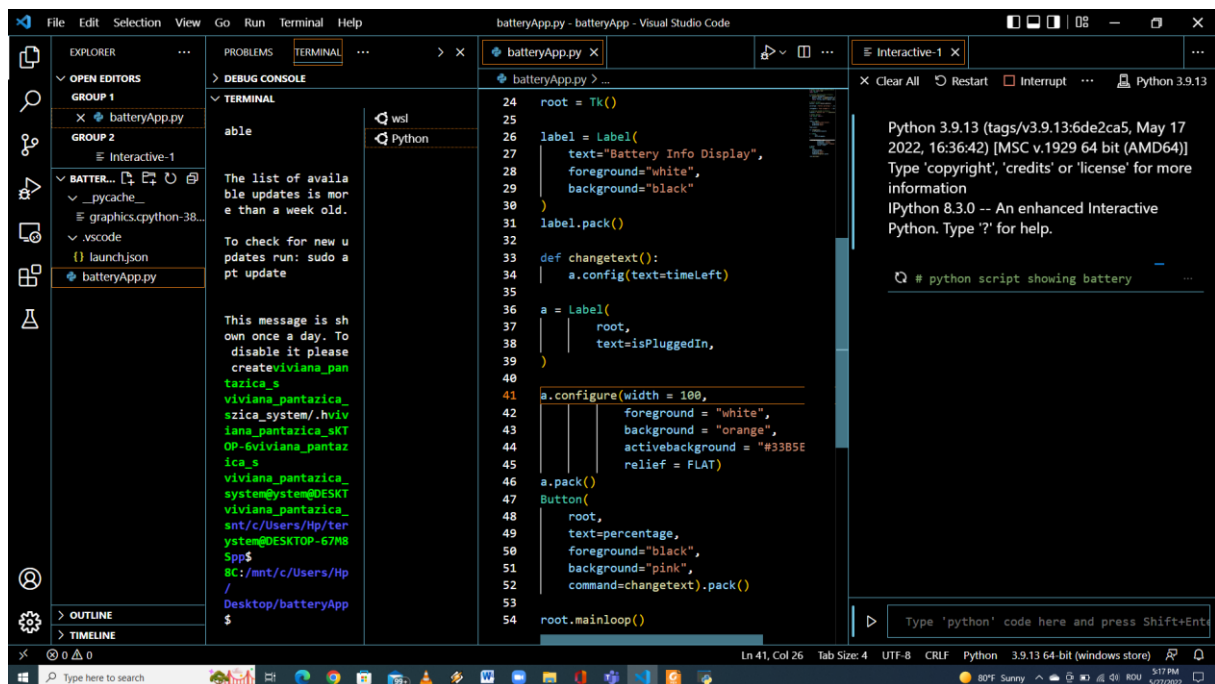
a.pack()
Button(
    root,
    text=percentage,
    foreground="black",
    background="pink",
    command=changetext).pack()
```

```
root.mainloop()
```

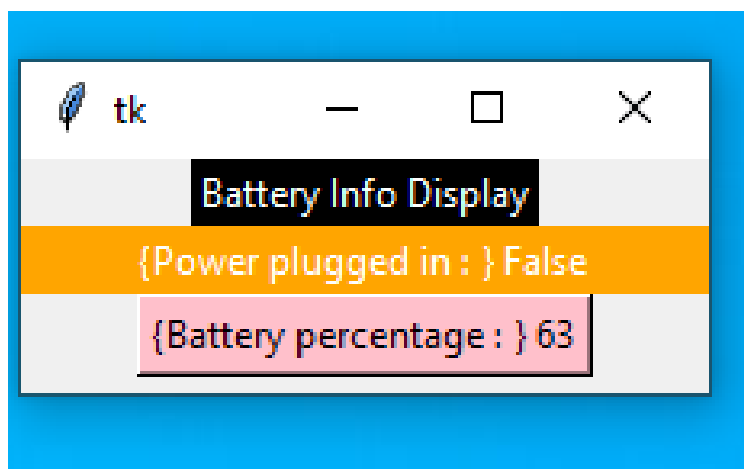
Pentru afișajul grafic:

Am folosit biblioteca Tkinter, am creat un obiect root, pe care l-am populat cu o etichetă generică ce conține numele aplicației (label), o eticheta cu valoare modificabilă (a), care fie prezintă dacă PC-ul este conectat la încărcat (Power Plugged In: True / False), fie timpul rămas până la epuizarea bateriei (Battery Left) și un buton care afișează mereu procentajul bateriei (Battery percentage), dar care, prin apăsare, modifică valoarea etichetei anterioare.

Rularea aplicației:



Înainte de apăsarea butonului:



După apăsarea butonului:

