

# Memòria del Puzzle1: Lectura de Targetes NFC amb Raspberry Pi i ACR122U

## 1. Configuració de la Raspberry Pi

Per aquest projecte, he utilitzat una Raspberry Pi 4 Model B i he instal·lat el sistema operatiu Raspberry Pi OS de 32 bits. Durant la instal·lació, he activat l'SSH per poder accedir-hi remotament sense necessitat d'un monitor. A més, he configurat la meva Raspberry Pi perquè es connecti a la xarxa que comparteixo des del meu portàtil, permetent així tenir connexió a internet sense necessitat d'un cable Ethernet.

```
Unset  
ssh pi@<IP_RASPBERRY>
```

## 2. Instal·lació de les Dependències a la Raspberry Pi

Per gestionar les dependències del projecte, he creat un entorn virtual de Python per aïllar-les del sistema principal. He fet això executant la comanda:

```
Unset  
python3 -m venv ~/Desktop/acr122u
```

Després, he activat l'entorn virtual amb:

```
Unset  
cd ~/Desktop/  
  
source acr122u/bin/activate
```

Un cop dins l'entorn virtual, he instal·lat la llibreria `nfcpy` amb:

```
Unset  
pip install nfcpy
```

### 3. La llibreria `nfcpy`: Elecció i Ús

Per aquest projecte, he utilitzat la llibreria `nfcpy`, que permet comunicar-se amb dispositius NFC des de Python. La vaig trobar buscant llibreries NFC i vaig veure que era una de les més populars i ben documentades.

Vaig escollir `nfcpy` perquè és fàcil d'utilitzar i compatible amb molts dispositius NFC, inclòs el lector ACR122U. No cal instal·lar programari addicional, i facilita la comunicació amb el lector per extreure informació com el número d'identificació únic (UID) de les targetes. Gràcies a `nfcpy`, he pogut fer un programa en Python que detecta targetes NFC i n'extreu l'UID de manera senzilla.

### 4. Explicació del Codi en Python

Abans d'escriure el codi, he creat un fitxer anomenat `puzzle1.py` dins la carpeta `Desktop/` per organitzar millor el projecte. Ho he fet utilitzant el següent comandament:

```
Unset  
nano puzzle1.py
```

Un cop obert l'editor `nano`, he afegit el codi següent.

El codi permet comunicar-se amb el lector NFC ACR122U per detectar targetes i obtenir-ne l'UID. Primer, importo la llibreria `nfc`, necessària per establir la connexió. Després, creo la classe `Rfid`, que conté tota la lògica del lector. En el mètode `__init__`, estableixo la connexió amb el lector NFC i inicialitzo una variable per guardar la informació de la targeta.

Quan es detecta una targeta, la funció `on_connect` desa la seva informació. La funció `read_uid` s'encarrega de buscar una targeta NFC i retornar el seu UID en format hexadecimal. Si no en troba cap, retorna `None`. També hi ha la funció `close`, que tanca la connexió amb el lector per alliberar recursos.

Finalment, en la part principal del codi, es crea una instància de la classe `Rfid`, es crida la funció `read_uid` i, si es detecta una targeta, es mostra el seu UID per pantalla abans de tancar la connexió amb el lector NFC.

Python

```
import nfc
```

```
class Rfid:
```

```
    def __init__(self):
```

```
        self.clf = nfc.ContactlessFrontend('usb')
```

```
        self.tag = None
```

```
    def on_connect(self, tag):
```

```
        self.tag = tag
```

```
        return True
```

```
    def read_uid(self):
```

```
        try:
```

```
            print("Esperant targeta NFC...")
```

```
            self.clf.connect(rdwr={'on-connect': self.on_connect})
```

```
            if self.tag:
```

```
                return self.tag.identifier.hex()
```

```
            return None
```

```
        except Exception as e:
```

```
            print(f"Error en llegir UID: {e}")
```

```
            return None
```

```
    def close(self):
```

```
        self.clf.close()
```

```
if __name__ == "__main__":  
    rf = Rfid()  
    uid = rf.read_uid()  
    if uid:  
        print(f"UID de la targeta: {uid}")  
    rf.close()
```

## 5. Execució del Codi

Per executar el programa, primer he connectat la Raspberry Pi a la xarxa Wi-Fi compartida del meu ordinador. Un cop connectada, he endollat el lector NFC ACR122U a un port USB de la Raspberry Pi i he accedit a la Raspberry des del terminal del meu ordinador amb:

```
Unset  
ssh pi@<IP_RASPBERRY>
```

Després, he accedit a l'escriptori on tinc el projecte amb:

```
Unset  
cd Desktop/
```

A continuació, he activat l'entorn virtual:

```
Unset  
source ~/Desktop/acr122u/bin/activate
```

Finalment, he executat el codi amb:

```
Unset  
python3 puzzle1.py
```

Quan he apropat una targeta NFC al lector, el programa ha detectat la targeta i ha mostrat el seu UID en hexadecimal a la pantalla.