Attendance System using RFID
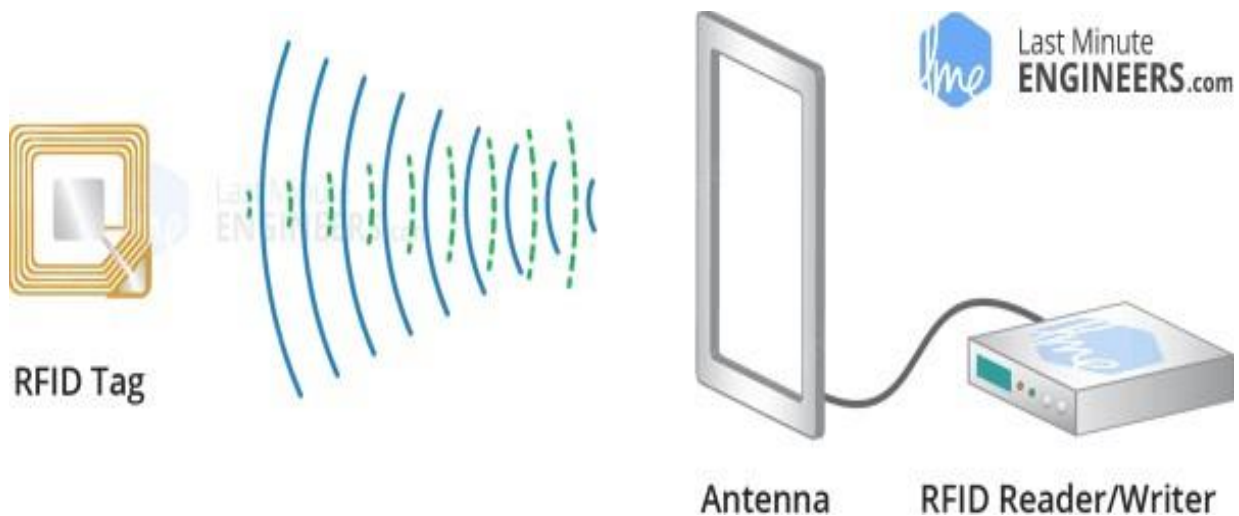# (RADIO-FREQUENCY WAVES ACCESS)

Hardware Requirements:

1. Arduino UNO
2. RFID MIRF
3. RFID Tags
4. Bread Board

Software Requirements:

1. Python 3.10 (at least) IDE and software.
2. Arduino IDE/modules.
3. Microsoft Excel (preferably)

Circuit Diagram:

RFID Tag

Antenna   RFID Reader/Writer

Project Description:

RFID Cards:

1) An RFID consists of a RFID Tag and a RFID Reader,
2) A RFID Reader consists a radio frequency module and antenna which produces high frequency electromagnetic field the tag only contains microchip for information and antenna for transmitting and receiving signals.
3) When the tag is brought close to the reader, the reader generates an electromagnetic field. This causes electrons to move through the tag's antenna and subsequently powers the chip.
4) The chip then responds by sending its stored information back to the reader in the form of another radio signal. This is called a backscatter. The reader detects and interprets this backscatter and sends the data to a computer or microcontroller.
5) The RC522 RFID module based on the MFRC522 IC from NXP is one of the cheapest RFID options you can get online for less than $4 (INR325 as of 9/11/22). It usually comes with an RFID card tag and a key fob tag with 1KB of memory. And the best part is that it can write a tag that means you can store any message in it.

```
class Courselist(tk.Frame):
    def __init__(self,parent,controller):
        tk.Frame.__init__(self,parent)

        bt_sns = tk.Button(self,text="Signals And Systems",font=("Caslon",12),height=1,width=25,bg="green" ,command=lambda:controller.show_frame(Courselogin))
        bt_edc = tk.Button(self,text="Electronic Devices",font=("Caslon",12),height=1,width=25,bg="green" ,command=lambda:controller.show_frame(Courselogin))
        bt_mal = tk.Button(self,text="Linear Algebra",font=("Caslon",12),height=1,width=25,bg="green" ,command=lambda:controller.show_frame(Courselogin))
        bt_oop = tk.Button(self,text="Object Oriented Programming",font=("Caslon",12),height=1,width=25,bg="green" ,command=lambda:controller.show_frame(Courselogin))
        bt_lnt = tk.Button(self,text="Linear Network Theory",font=("Caslon",12),height=1,width=25,bg="green" ,command=lambda:controller.show_frame(Courselogin))


        bt_sns.pack()
        bt_edc.pack()
        bt_mal.pack()
        bt_oop.pack()
        bt_lnt.pack()
```

Python:

1) While RFID Cards and Arduino UNO form the base of Hardware side of this Project, Python has been primarily used to make the frontend part of the program.
2) Python is chosen as the preferred language over others not only because it's the focus of the syllabus, but also the reason that it also supports modules/packages of Arduino and that it can be used to make readable Databases in Excel.
3) The tkinter library has been used to make the GUI.
4) A key takeaway from this project is that it uses all the knowledge that is within the syllabus, from the very beginning to the PIL Library. All as Object-Oriented.

Database:

1) The backend part of this program is the strong database that's been made on Microsoft Excel as a spreadsheet.
2) There is no need of worrying about the attendance to the professors/students after the day, as the continuously updated Excel Sheet will show all the records till that day.
3) Spreadsheet usage also means that we can convert it to CSV File and utilize it fully.

	A	B	C	D	E	F	G	H	I	J	K
	Brent	Rivera	98908016	BT21ECE0	Today	Present	Present	Present	Present	Present	Present
	Lexi	Rivera	98908016:	BT21ECE0	Today						
	Dom	Brack	86001946:	BT21ECE0	Today						
	Andrew	Davilla	13161581(	BT21ECE0	Today	Present	Present	Present	Present		
	Ben	Azelart	97678984:	BT21ECE0	Today						
	Pierson	Wodzy	19540108:	BT21ECE0	Today						
	Lexi	Hensler	66388267:	BT21ECE0	Today						

This data is stored in Excel spreadsheets, which are created in the database.

Conclusion and Future Work:

Thus, a system suitable for recording attendance without hassles was developed as a result of the project. During this process, a great deal of teamwork and coordination was achieved. Some more functionalities can be added, which weren't possible due to the unavailable resources like sending a SMS to absent students, informing about low attendance. Also, some statistical analysis of the attendance can also be made, so that the days where most students remain absent can be chosen to teach the optional aspects of the syllabus, etc.

The acquaintance with the tkinter library, databases, Arduino aspects unlocked three new dimensions of programs that can be created using Python. Various apps and executable programs on Windows are based of these three. Some of them are:

- Creating GUIs for various apps/software.
- Using Arduino as a connector between hardware/software.
- Establishing databases in companies/software requiring large data handling.

We look forward to completing similar challenging projects in the future. Thank you.