

# Lector / Clonador RFID

Autor: Iñaki Abadía Osta

Tutor: Enrique Torres Moreno

# Motivación

## RFID



## MIFARE



- Classic
- Ultralight
- DESFire
- Plus
- ...

# Motivación

## MIFARE CLASSIC

1KB (ó 4)

Habitualmente solo se usa el UID



# Motivación

## MIFARE CLASSIC 1K

		Byte Number within a Block																
Sector	Block	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Description
15	3	Key A					Access Bits			GPB	Key B					Sector Trailer 15		
	2																	Data
	1																	Data
	0																	Data
14	3	Key A					Access Bits			GPB	Key B					Sector Trailer 14		
	2																	Data
	1																	Data
	0																	Data
:	:																	
:	:																	
:	:																	
1	3	Key A					Access Bits			GPB	Key B					Sector Trailer 1		
	2																	Data
	1																	Data
	0																	Data
0	3	Key A					Access Bits			GPB	Key B					Sector Trailer 0		
	2																	Data
	1																	Data
	0									GPB						Manufacturer Block		

- 64 Bloques de 16 Bytes
- 16 Sectores de 4 Bloques
  - 3 Usables
  - 1 Bits de acceso, clave
- 1024 Bytes
  - 768 usables
- Bloque 0
  - UID
  - **NO ESCRIBIBLE**

EN TEORIA...

# Motivación

MIFARE CLASSIC 1K = 0 SEGURIDAD

Key A	Access Bits	Key B
FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF

**MFOC**

2008

<https://github.com/nfc-tools/mfoc>

**MFCUK**

2010

<https://github.com/nfc-tools/mfcuk>



# Objetivo

Dado que:

Habitualmente solo se usa el UID

Al no ser escribible el bloque 0, clave por defecto

Existen tarjetas donde se puede escribir el UID

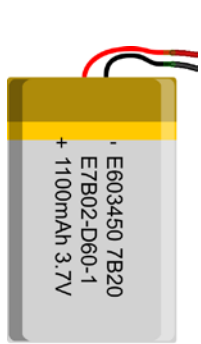
Voy a:

**Demostrar lo facil  
que es robar una  
identidad**

# Objetivo

## Lector / Escritor / Clonador RFID

- Autónomo
- Portable
- No volatil
- Control/Interacción remota



# Objetivo

## Lector / Escritor / Clonador RFID

Autónomo + Portable + No volátil

### ESP8266 (NodeMCU Dev Kit)



- Bajo coste (<3€)
- Tamaño reducido
- Wifi
- SPI
- Sistema de ficheros (No volátil)
- Existen cantidad de librerías de la comunidad

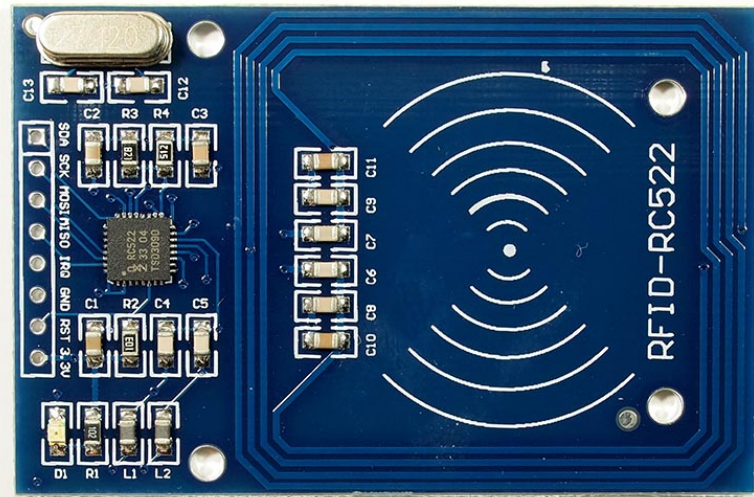


# Objetivo

## Lector / Escritor / Clonador RFID

Autónomo + Portable + No volátil

### MFRC522 (Lector/Escritor RFID)



- Bajo coste (<2€)
- Tamaño reducido
- Comunicación SPI

# *Objetivo*

## Lector / Escritor / Clonador RFID

Interacción remota

### Smartphone Android

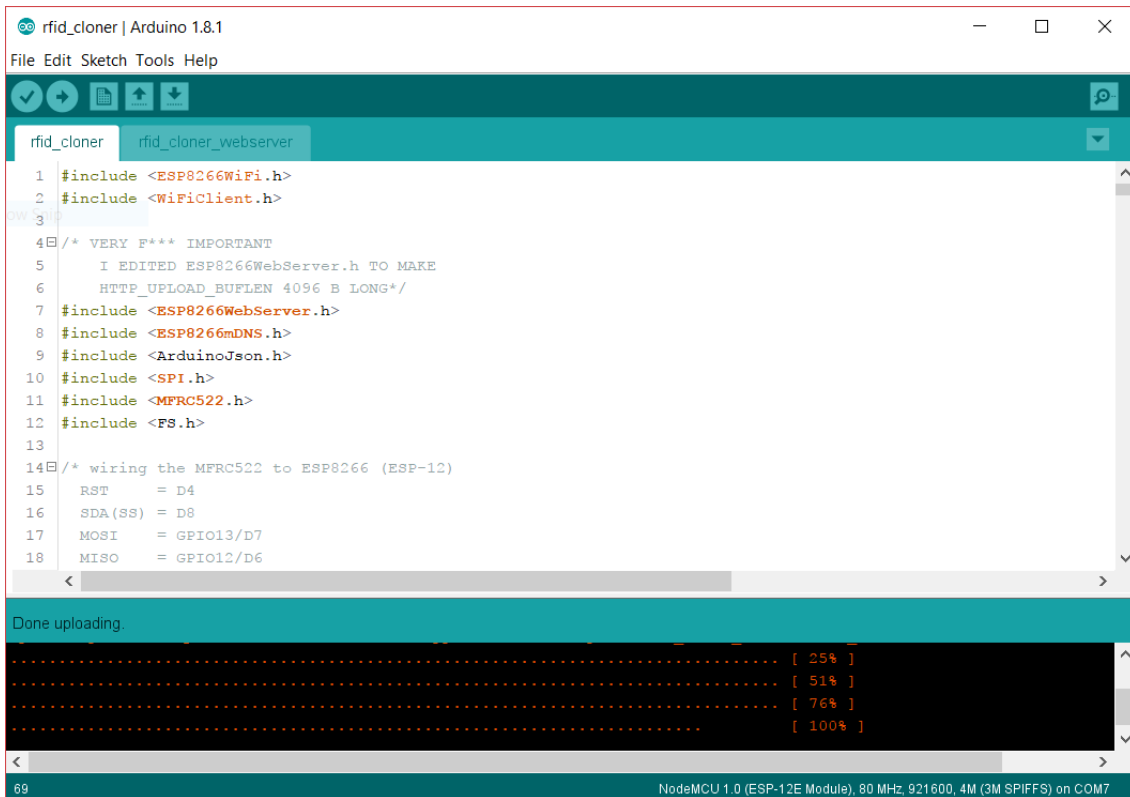


- Bajo coste
- Wifi/Bluetooth
- Interfaz táctil
- Fácil de programar
- ¿Quién no tiene uno?

# Implementación

## ESP8266 + RC522

### Arduino IDE



```
1 #include <ESP8266WiFi.h>
2 #include <WiFiClient.h>
3
4 /* VERY F*** IMPORTANT
5    I EDITED ESP8266WebServer.h TO MAKE
6    HTTP_UPLOAD_BUFLen 4096 B LONG*/
7 #include <ESP8266WebServer.h>
8 #include <ESP8266DNS.h>
9 #include <ArduinoJson.h>
10 #include <SPI.h>
11 #include <MFRC522.h>
12 #include <FS.h>
13
14 /* wiring the MFRC522 to ESP8266 (ESP-12)
15    RST    = D4
16    SDA(SS) = D8
17    MOSI   = GPIO13/D7
18    MISO   = GPIO12/D6
```

### Librerías

#### ESP8266

*Comunity*

<https://github.com/esp8266/Arduino>

*WebServer*

*SPIFFS*

*mDNS*

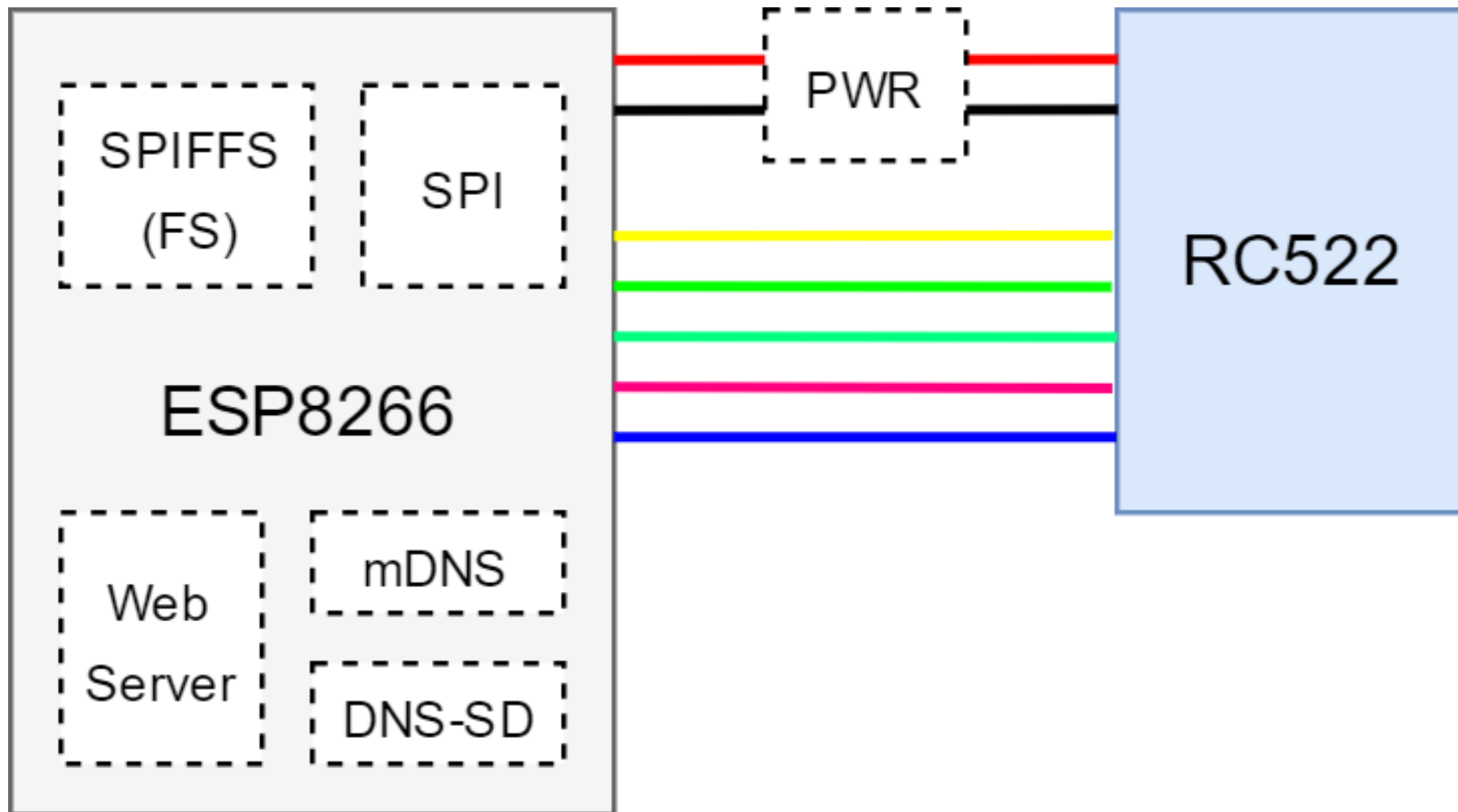
#### RC522

*Miguel Balboa*

<https://github.com/miguelbalboa/rfid>

# Implementación

## ESP8266 + RC522



# Implementación

## ESP8266 + RC522

### Modo promiscuo

Lee y guarda todas las tarjetas  
en el rango

**SPIFFS** *(lib)*  
3M

<https://github.com/pellepl/spiffs>

Sin directories, FS plano, carpetas = filtros

### Servidor Web

Wifi *(lib)*

mDNS responder *(lib)*

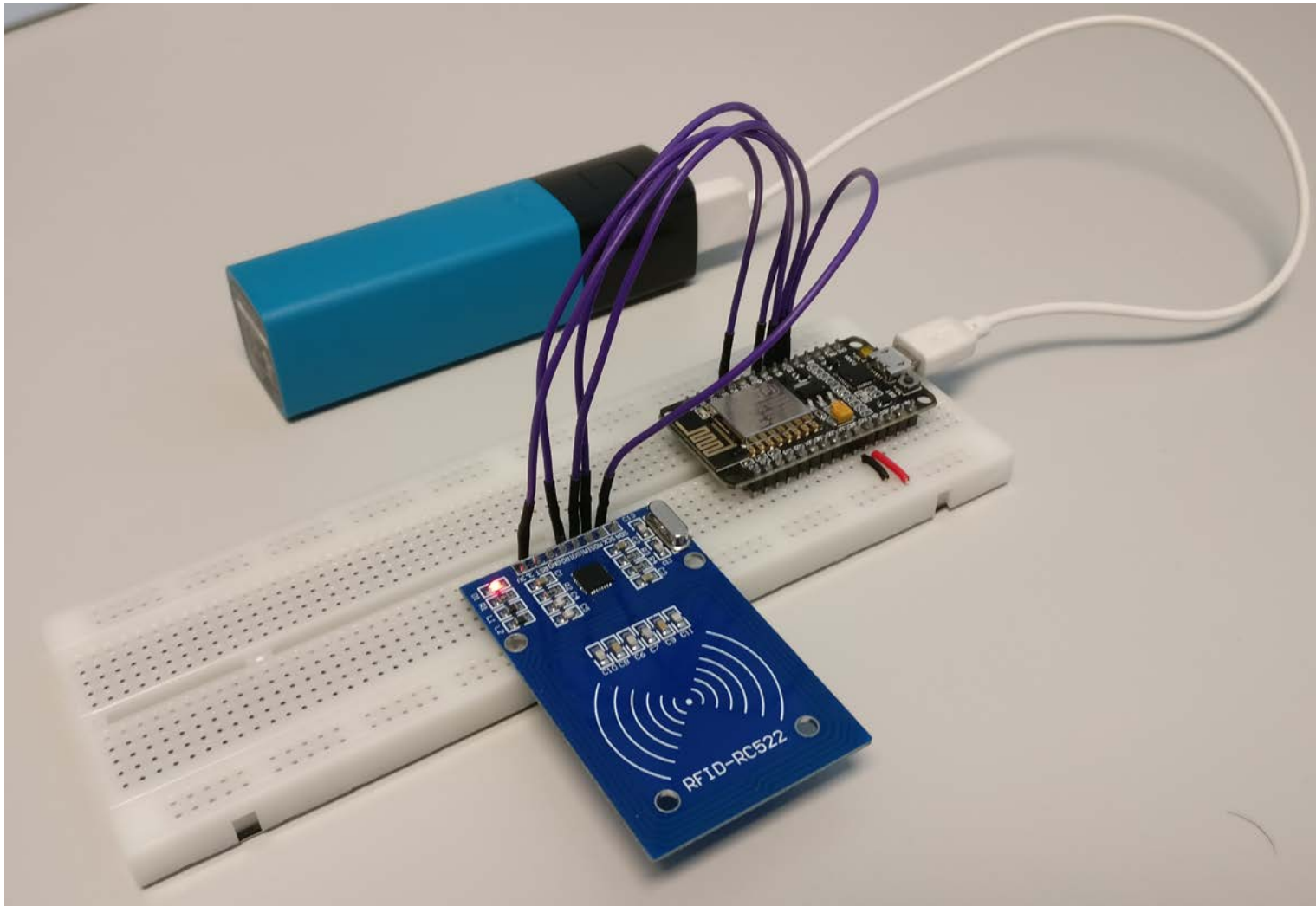
DNS-SD service advertiser *(lib)*

WebServer *(lib)* HTTP API

METHOD	URL	Descripción
GET	/cardslist	JSON con listado de tarjetas en SPIFFS
GET	/card	JSON con contenido de tarjeta
DELETE	/card	Borra tarjeta
PUT	/card	Manda tarjeta, param <i>write</i> para escribirla

# *Implementación*

ESP8266 + RC522



# *Implementación*

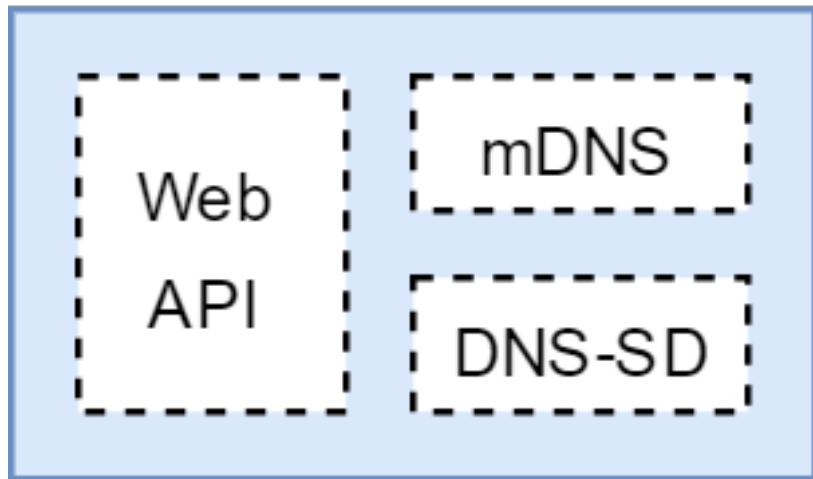
## Android APP



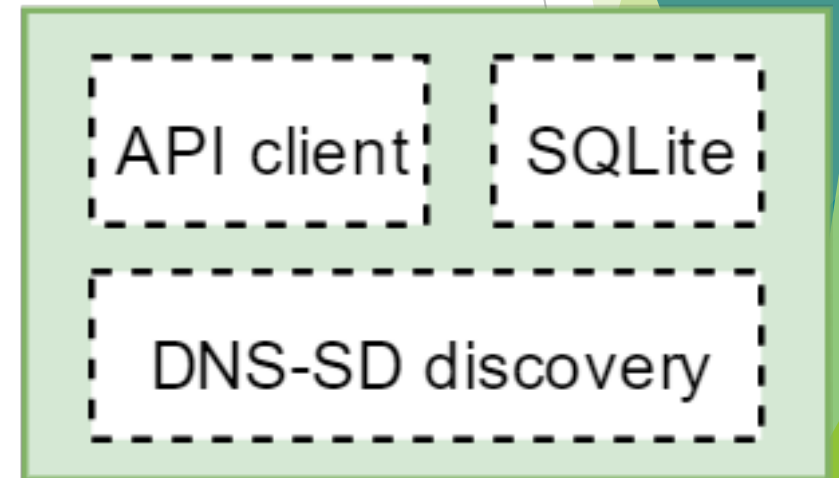
# *Implementación*

## Android APP

### ESP8266



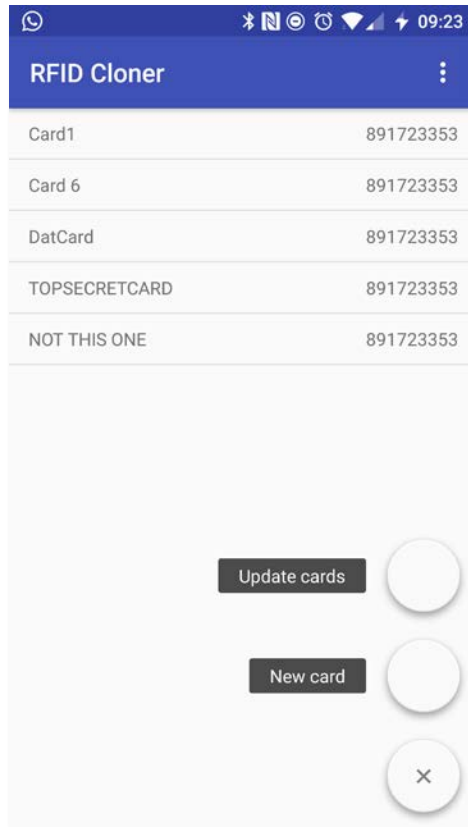
### Android APP



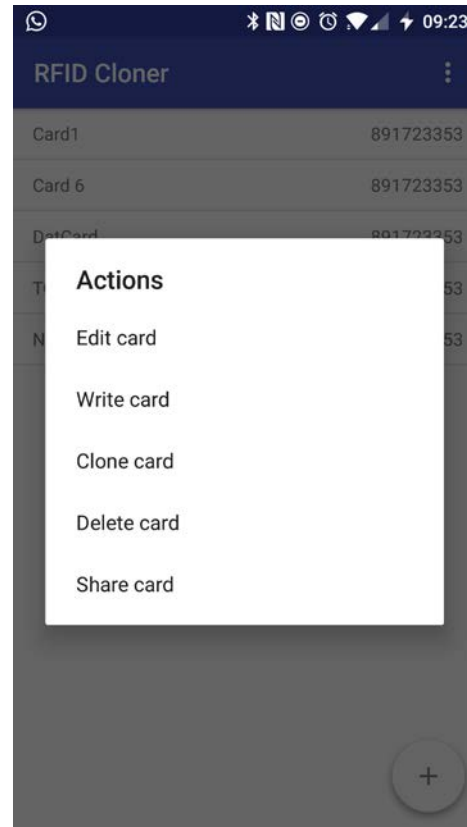


# Implementación

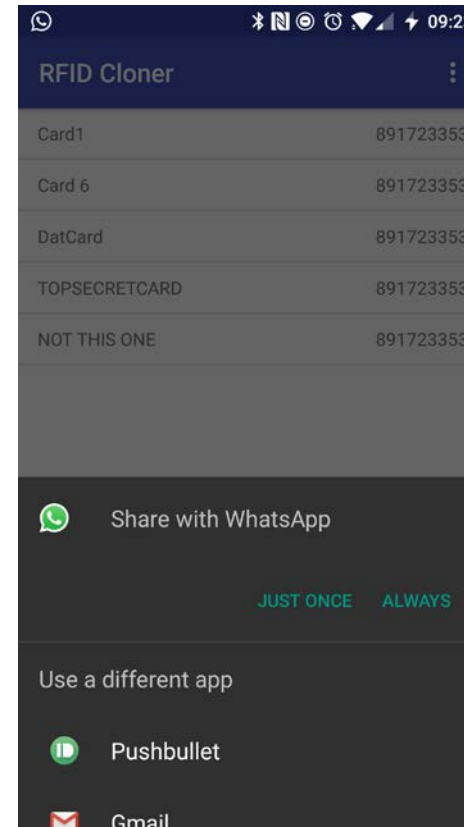
## Android APP



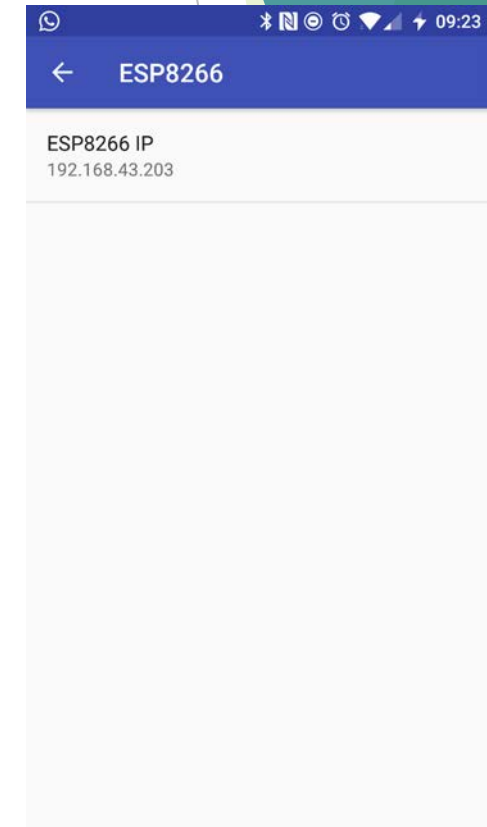
Listado de  
Tarjetas



Acciones sobre  
tarjetas



Compartir  
tarjetas

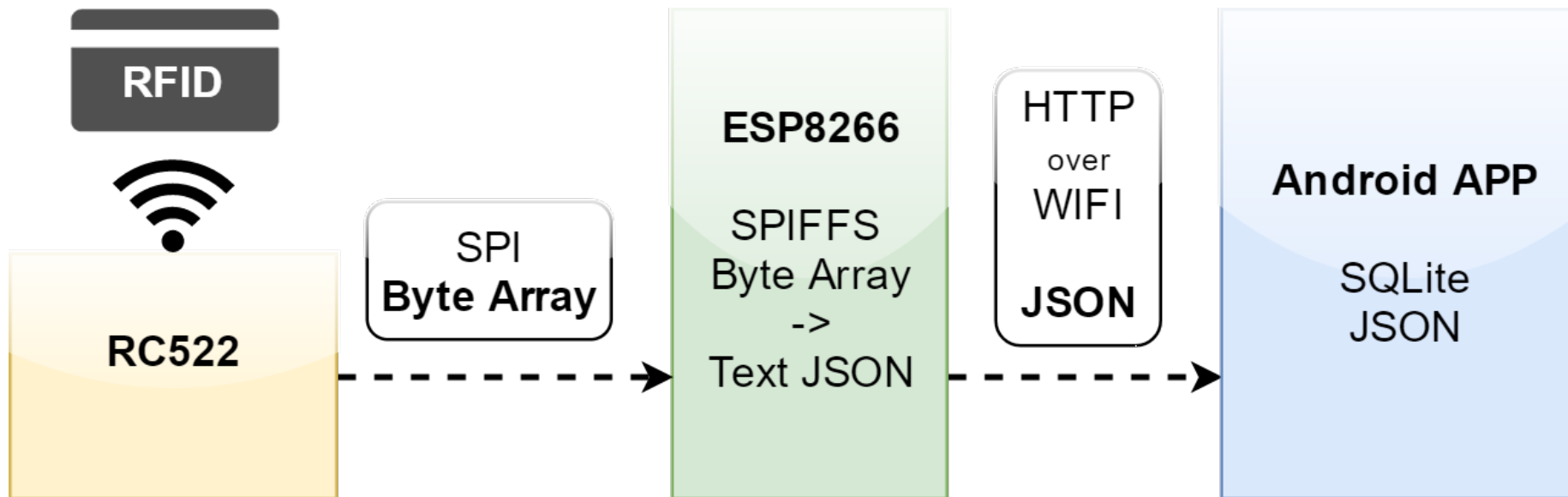


mDNS/DNS-SD  
fallback

# *Implementación*

ESP8266 + RC522 + Android APP

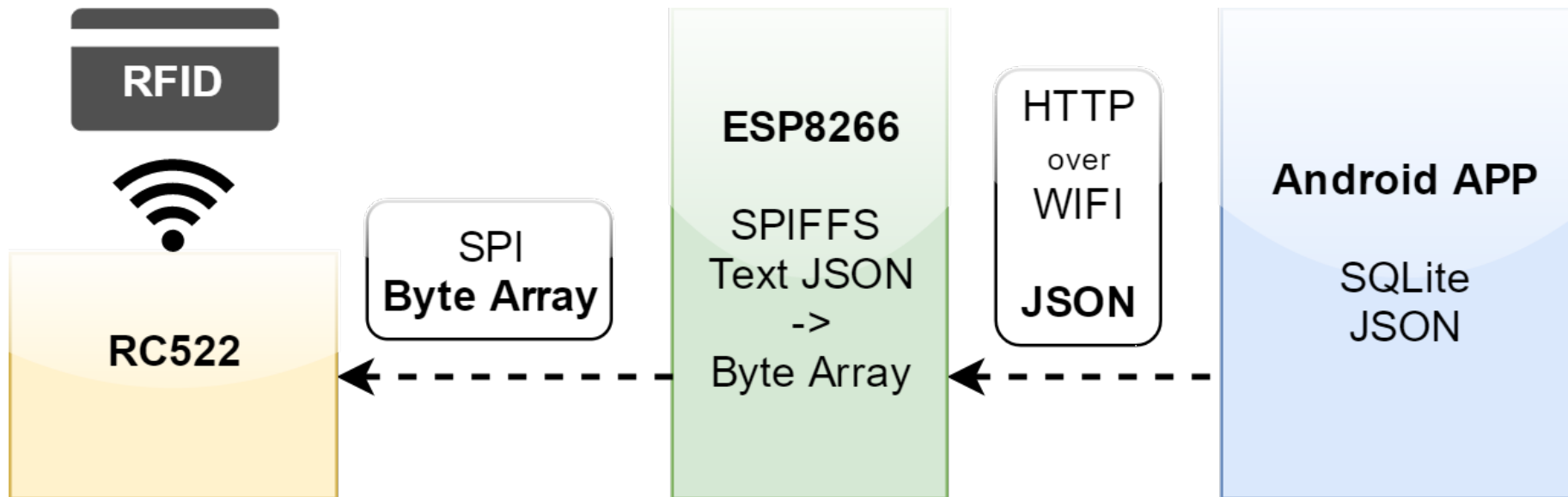
## LECTURA



# *Implementación*

ESP8266 + RC522 + Android APP

## ESCRITURA



SHOW



TIME!

Q&A

*Y VALORACIÓN*