

Co-creation technological challenge:

# Pictogram board with emotions



01

## What is this activity about?



The pictogram board with emotions is a tool designed to help people with communication difficulties.

It works as follows:

- Each pictogram represents an emotion.
- When it is placed on the base, an electronic system recognises the NFC card and **plays** the name of that emotion **out loud**.
- The pictograms are designed with an **adapted grip** to make them easier to use.

02

## What will be built?

The challenge consists of **three main elements**:

### A. Adapted pictograms

Original design based on **ARASAAC** pictograms.

They represent emotions related to videoconferencing.

3D printed in bright colours.

They include an **NFC card** inside.

They incorporate an **easy-grip** support.

### B. Board with four slots

Space to place up to **4 pictograms**.

Designed and 3D printed.

### C. Electronic box

Contains the circuit that makes it possible to play the audio:

Arduino

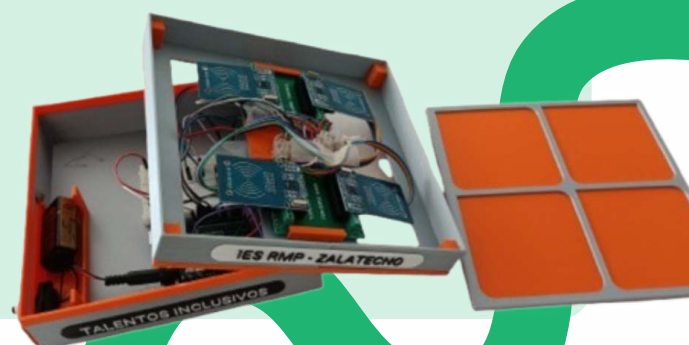
NFC readers

MP3 player

Speaker

Battery and switch

Indicator LED





## 03 Step-by-step challenge

STEP 1: Topic selection	STEP 2: 3D design	STEP 3: 3D printing	
Chosen topic: <b>Emotions.</b>	Design of the adapted grip support.	Printing of pictograms and supports in coloured PLA.	
	Design of the board with 4 slots.		
Selection of base pictograms from <a href="#">ARASAAC</a> .	Design of the circuit box (including a battery drawer, cable openings and a “second level” to organise components).	Slicing with colour changes to highlight the emotions.	
STEP 4: Circuit assembly			
Placement of the Arduino, NFC readers, MP3 player, speaker and LED.	Connection using jumpers and a breadboard.	Soldering of the necessary components.	Adjustments and redesigns of the box to improve assembly.
STEP 5: Audio preparation	STEP 6: Programming	STEP 7: Final assembly	
Recording of the pictogram audio by the students.	Reading the codes of each NFC card.	Insertion of the NFC cards into the pictograms.	
Sound editing with Audacity.	Adding those codes to the Arduino program.	Assembly of the board and the box.	
Uploading the audio files to the MP3 player.	Functional testing.	Final tests with the target groups.	

## 04 Materials needed

3D printing	Electronics		
Coloured PLA filament	• Arduino	• LED + 100 Ω resistor	• Breadboard
Glue	• NFC readers	• Jumpers and cables	• Connection sheet
NFC cards or stickers	• MP3 player	• 9V battery (or USB rechargeable battery)	• Magnets (optional)
	• Speaker	• Switch	• Soldering tin

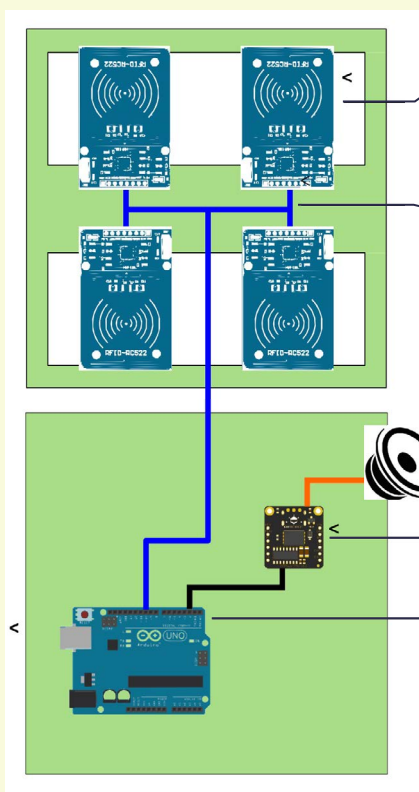
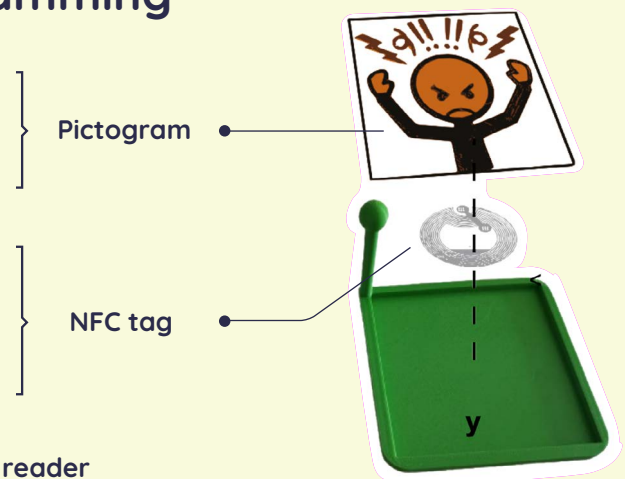
## 05 Tools needed

Design and printing		Electronics and programming	
Computer with FreeCAD	3D printer	Computer with Arduino IDE	Microphone
PrusaSlicer	Pliers, file, scissors	Audacity	Soldering iron
		Screwdrivers, pliers, scissors	

## 06 Board assembly and programming

It follows the **ARASAAC** system, one of the most widely used systems in Spain and Latin America. It offers free, open pictograms, from which a set was selected to express emotions and moods.

**NFC** (Near Field Communication) is a short-range wireless communication mechanism using radio waves, up to 20 cm, and high frequency, 13.56 MHz. It is used in access control, contactless card payments and pictograms.



- RFID-RC522 reader**  
**RFID** (Radio Frequency Identification) is a technology that makes it possible to identify and connect different devices by emitting and reading radio waves.
- Communication bus**  
 Communication channel between the readers and the Arduino. It uses the SPI (Serial Peripheral Interface) protocol. This communication protocol uses information transmission lines shared among the four devices.
- 3W Speaker, DFROBOT**
- DFPlayer Pro**  
 Music player containing the audio files associated with the pictograms. It can be used autonomously, but in this case it is controlled by the Arduino.
- Arduino UNO**  
 An easy-to-use and affordable microcontroller. It is the brain of the board, programmed to periodically check the RFID readers to detect when a new pictogram is placed, and to send the DFPlayer Pro the signal to play the corresponding audio.

This is a simplified version of the program that controls the pictogram board.

It is written in C and uses specific software libraries for the RFID readers and the MP3 player.

The code has a register of the NFC tags placed in the pictograms, and its operation is governed by two functions:

- The **setup()** function runs only once when the board starts. It configures the components.
- The **loop()** function runs immediately afterwards and repeats continuously. It contains the main logic.

```
#include <SPI.h> #include <MFRC522.h>
#include <DFRobot DF12015.h> #include
<SoftwareSerial.h>
// Register of pictograms by the associated
NFC tag.
// Each one corresponds to an audio file.
RegistroRFID registro[]
{
  {{0xFF, 0x0F, 0xCS, 0x11}}, 1, "/1.mp3"},
  {{0xFF, 0x0F, 0xC6, 0x11}}, 2, "/2.mp3"},
  {{0xFF, 0x0F, 0xC7, 0x11}}, 3, "/3.mp3"},
  // ...
  {{0xFF, 0x0F, 0xA6, 0x24}}, 27, "/27.mp3"}
};

void setup() {
  // Protocol initialization
  Serial.begin(9600); // Serial communication
  SPI.begin(); // SPI bus

  // Simultaneous reset of the readers
  pinMode(PIN_RST, OUTPUT);
  digitalWrite(PIN_RST, LOW);
  delay(10);
  digitalWrite(PIN_RST, HIGH);
  delay(10);

  // Sequential initialization of the readers
  for (uint8_t reader = 0;
       reader < LECTORES_NFC;
       reader++)
  {
    mfr522[reader].PCD_Init(ssPins[reader],
                           pin rst);

  }

  // MP3 player initialization
  DF12015Serial.begin(MP3_BAUD_RATE);
  DF12015.setVol(MP3_VOLUMEN);
  DF12015.switchFunction(DF12015.MUSIC);
  DF12015.setPlayMode(DF12015.SINGLE);
  DF12015.setBaudRate(MP3_BAUD_RATE);
}
```

```
void loop() {
  // Sequential check of the readers
  for (uint8_t reader = 0;
       reader < LECTORES_NFC;
       reader++)
  {

    // A transaction is started on the SPI bus
    SPI.beginTransaction(mfr522SPI);

    // The bus is enabled for the reader
    digitalWrite(ssPins[reader], LOW);

    // Check whether there is a new NFC tag
    if (mfr522[reader].PICC_IsNewCardPresent())
    {

      // The tag identifier is obtained
      if (mfr522[reader].PICC_ReadCardSerial())
      {

        lee uid(mfr522[reader],
                uid_leido, &uid_long);

        mfr522[reader].PICC_HaltA();

        // The tag is processed to play the
        audio
        procesa_tag(uid_leido)
      }

    }

    // The transaction ends and continues with
    the next reader
    digitalWrite(ssPins[reader], HIGH);
    SPI.endTransaction();

    // ... and continues accordingly reader
  }
}
```