```
/*
This is the Code for the Cache o' Phone Geocache project.
This Project also serves as my skilled work, the rest of which you can find on
https://justricity.de under Cache o' Phone.
The Code here was written by Me. Justin. If you have any Questions, Ideas or
anything, feel free to contact me through my social media
*/
/*
Dies ist der Code für das Cache o' Phone Geocache Project.
Dieses Projekt wurde in Zusammenhang mit meiner Facharbeit erstellt, welche du
auf https://justricity.de unter Cache o' Phone finden kannst.
Dieser Code wurde von mir (Justin) geschrieben. Falls du irgendwelche Fragen,
Vorschläge oder sonstiges hast, schreib mir doch gerne über meine Social Media
Accounts
*/
//Including all needed Librarys (Intigrieren aller benötigten Bibliotheken)
#include "Arduino.h"
#include "SoftwareSerial.h"
#include "DFRobotDFPlayerMini.h"
#include "Keypad.h"
//Defining the volume to play at in % (Definieren der Lautstärke in %)
#define Volume  50
//Defining the Relay Pin (Definieren des Relay Pins)
#define Relay  12
//Defining the Busy Pin (Definieren des Busy Pins
#define Busy A0
//Defining the Code Numbers (Definieren der Code Nummern)
int LockCode[4] = {1, 2, 3, 4};
int EnteredCode[4];
int EntryCounter = 0;
#define PlayEnd  false
int Playloop = 0;
//Definitions for the Keypad (Definitionen für das Keypad
const byte ROWS = 4;
const byte COLS = 4;
const byte MaxPinCodeLength = 20;

char keys[ROWS][COLS] = {
                          {1,2,3,13},
                          {4,5,6,23},
                          {7,8,9,33},
                          {40,10,42,43}
                         };
byte rowPins[ROWS] = {5,4,3,2}; //5,6,7,8};
byte colPins[COLS] = {9,8,7,6}; // {9,10,11,12};
//Setting the Pincode (Festlegen des Pincodes)
```

```
Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins, ROWS, COLS);

//Code for the DFPlayerMini (Code für den DFPlayerMini)
SoftwareSerial mySoftwareSerial(10,11); // RX, TX
DFRobotDFPlayerMini myDFPlayer;
void printDetail(uint8_t type, int value);

void setup() {
  //Setting the Button Pin as a Input with pullup (Festlegen des Knopf Pins als
einen Eingang mit Pullup)
  pinMode(Relay, OUTPUT);

  //Setting up the DFPlayerMini (Einstellen des DFPlayerMini)
    mySoftwareSerial.begin(9600);
  Serial.begin(115200);

  Serial.println();
  Serial.println(F("DFRobot DFPlayer Mini Demo"));
  Serial.println(F("Initializing DFPlayer ... (May take 3~5 seconds)"));

  if (!myDFPlayer.begin(mySoftwareSerial)) {
    Serial.println(F("Unable to begin:"));
    Serial.println(F("1.Please recheck the connection!"));
    Serial.println(F("2.Please insert the SD card!"));
    while(true){
      delay(0);
    }
  }
  Serial.println(F("DFPlayer Mini online."));

  //Setting up the relay (Einstellen des Relays)
  pinMode(Relay, OUTPUT);
  digitalWrite(Relay, LOW);
  //Telling the debugger (probably me (Justricity/Justin)) that the Setup is done
(Informieren des Debuggers (warscheinlich ich(Justricity/Justin)), dass das setup
fertig ist.)
  Serial.println("Device is setup");

  Serial.println("The current Code is:");
  Serial.print(LockCode[0]);
  Serial.print(LockCode[1]);
  Serial.print(LockCode[2]);
  Serial.print(LockCode[3]);
}
```

```
void loop() {

  //Setting the Volume(Einstellen der Lautstärke)
  myDFPlayer.volume(Volume);
  //Waiting 5 Seconds before starting the Sound (5 Sekunden Pause bevor der Sound abgespielt wird)
  delay(5000);
 Player();
}

//Creating an extra loop for the DFPlayer for cleaner code (Erstellen einer extra schleife für einen aufgeräumteren Code)
void Player() {
  while(Playloop == 0){
    Playloop = 1;
    Serial.println(Playloop);
    myDFPlayer.play(1);
    Serial.println("Starting to play");
    delay(2000);
    Serial.println(analogRead(Busy));
    while(analogRead(Busy) <= 10){
      Serial.print("Playing");
      Serial.print(analogRead(Busy));
      delay(1000);
    }
    //Telling the Debugger that the DFPlayer is finished playing and starting the Input Sequence (Informieren des Debuggers, dass der DFPlayer fertig mit spielen ist und starten der Eingabesequenz)
      Serial.println("Finised");
      Input();
 }
}

//Extra loop for the Keypad (Extra Schleife für das Keypad)
void Input(){
  Serial.println("Getting Key");


  for(int EntryCounter = 0; EntryCounter < 4; EntryCounter++){
    int Entry = keypad.getKey();
    EnteredCode[EntryCounter] = Entry;
    Serial.println(EnteredCode[EntryCounter]);
    delay(1000);
    }
    Serial.println("Finished");
```

```
      Serial.print("You Entered:");
      Serial.print(EnteredCode[0]);
      Serial.print(EnteredCode[1]);
      Serial.print(EnteredCode[2]);
      Serial.print(EnteredCode[3]);

  if(EnteredCode[0] == LockCode[0]){
     if(EnteredCode[1] == LockCode[1]){
       if(EnteredCode[2] == LockCode[2]){
          if(EnteredCode[3] == LockCode[3]){
            Serial.println("Code correct. Opening");
       }
         else{
    Serial.println("Code incorrect. Acces denied");
  }
     }
       else{
    Serial.println("Code incorrect. Acces denied");
  }
    }
      else{
    Serial.println("Code incorrect. Acces denied");
  }
  }
  else{
    Serial.println("Code incorrect. Acces denied");
  }

  Serial.println(keypad.getKey());
  delay(500);
}
```