

Relatório final My Home

JANEIRO 2022

PREPARADO POR

Gonçalo Bragança

Inês Marques

Isabella Costa

Rodrigo Barata



PARA OS DOCENTES

PEDRO ARROZ SERRA
GUILHERME SANCHES
ANDREA COSTIGLIOLA

Introdução

Como foi dito no relatório inicial, o tema escolhido foi o My Home, especificamente:

- Controlo de iluminação interior/exterior
- Controlo de temperatura
- Controlo de portas e janelas
- Controlo de outros sensores

Durante a realização do projeto debruçamo-nos sobre esses temas e tentámos aplicá-los na situação correta, tentando seguir ao máximo aquilo que tínhamos deixado descrito no relatório inicial.

Abordagem

Inicialmente, pretendíamos fazer a modelagem física, no entanto, como tal não foi possível optámos pela modelagem no simulador. Desta forma, tivemos algumas limitações no que toca a componentes físicos (ex: NFC scanner não está disponível no TinkerCad) e consequentemente modificação de lógica/código.

Os tópicos do nosso projeto atual seria:

- Controlo de iluminação por toda a casa
- Zona da garagem – controlo de iluminação e motor através de uma password
- Display LCD – exibição da temperatura da casa, assim como o controlo de iluminação de toda a casa (excepto garagem)
- O sistema de lock/unlock foi apenas aplicado na garagem

Controlo de iluminação

A iluminação de toda a casa poderá ser controlada através de um botão ou de um sensor de proximidade.

Foram adicionados 5 botões e 4 sensores de proximidade (a garagem não possui) de acordo com o número de divisões da casa.

SENSORES DE PROXIMIDADE

COMO REAGEM?

OS SENSORES DE PROXIMIDADE SÃO ATIVADOS A PARTIR DOS 30CM

BOTÕES

PARA QUE SERVEM?

OS BOTÕES TÊM O PODER DE LIGAR A LUZ INDEPENDENTEMENTE DO SENSOR

NOTA

O SENSOR DE LUMINOSIDADE DA PARTE EXTERIOR DA CASA FOI DESCARTADO DO NOSSO PLANO INICIAL

Controlo de temperatura

O controlo de temperatura foi feito através de um sensor, no entanto, para não sobrecarregar as entradas analógicas do Arduíno, apenas implementámos numa divisão da casa. Esta é constantemente exibida no LCD, servindo como medida padrão de toda a casa.

Num cenário ideal o controlo de temperatura trabalha em conjunto com o aquecimento central da casa.

SENSOR DE TEMPERATURA

COMO REAGE?

O SENSOR DE TEMPERATURA IRÁ DETETAR A MESMA E ENVIAR O SINAL PARA SER APRESENTADO NO LCD

LCD

O QUE APRESENTA?

O LCD NÃO SÓ APRESENTA A TEMPERATURA ATUAL, COMO TAMBÉM APRESENTA AS LUZES LIGADAS

Controlo de portas

EXTERIORES

Ao contrário do relatório inicial, o controlo de portas exteriores apenas é feito na zona da garagem.

A porta da garagem é aberta após a introdução de uma senha correta ("1111"), caso contrário é imitado um alarme. No nosso plano inicial, a porta da garagem poderia ser aberta por um NFC Scanner, mas devido à falta deste componente no TinkerCad alterações foram feitas para a porta da garagem abrir através de um keypad com senha pré-definida.

Após a verificação da senha, temos um motor que começa a trabalhar de modo a abrir a porta da garagem e luz da mesma será acendida, de modo a dar tempo para a pessoa estacionar, fechar a garagem e só depois a luz apagar.

Depois da luz de apagar, o motor volta a trabalhar na medida em que a porta da garagem fecha automaticamente.

KEYPAD

PARA QUE SERVE?

O KEYPAD É UTILIZADO DE FORMA AO UTILIZADOR PODER DIGITAR A SENHA DE ENTRADA

MOTOR

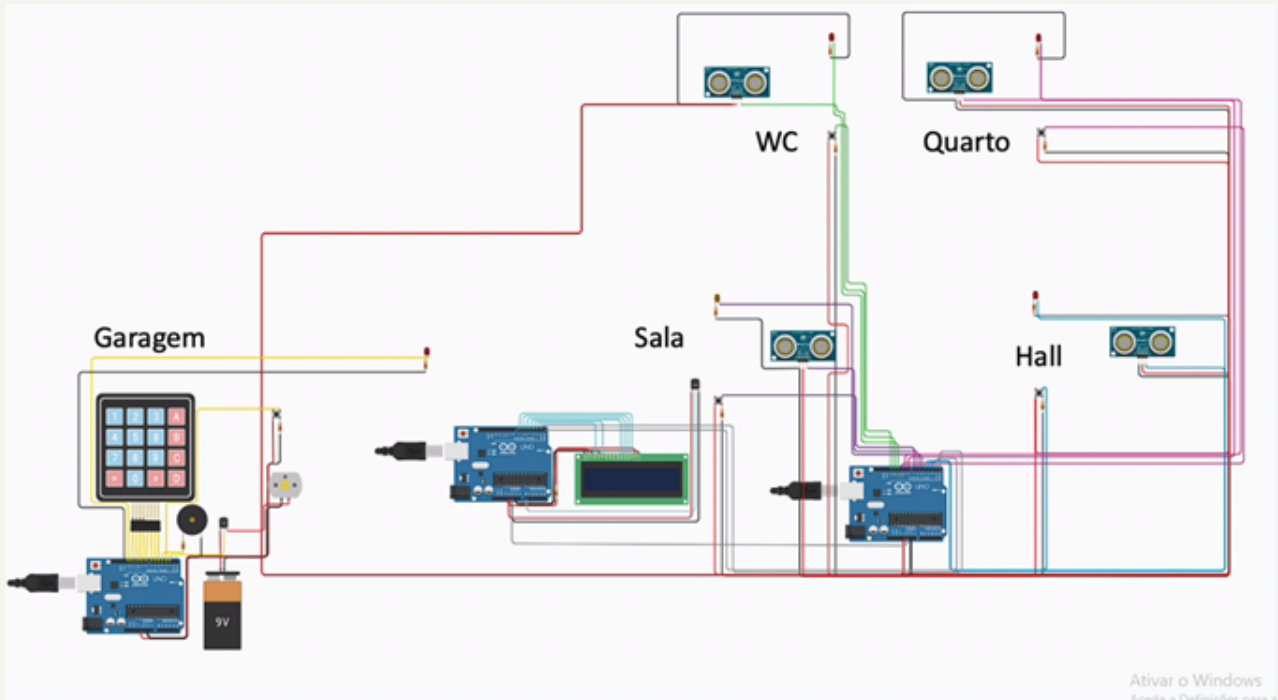
QUAL O SEU USO?

NESTA SIMULAÇÃO ASSUMIMOS QUE O MOTOR ESTÁ A ABRIR/FECHAR A PORTA DA GARAGEM APÓS VERIFICAÇÃO DA SENHA

Listagem de componentes

LED	5 UNIDADES TODAS AS DIVISÕES
BOTÃO	5 UNIDADES TODAS AS DIVISÕES
RESISTOR	10 UNIDADES TODAS AS DIVISÕES
BATERIA 9V	1 UNIDADE GARAGEM
TRANSISTOR NPN	1 UNIDADE GARAGEM
SENSORES DE PROXIMIDADE	4 UNIDADES TODAS AS DIVISÕES, EXCETO GARAGEM
KEYPAD (4X4)	1 UNIDADE GARAGEM
MOTOR CC	1 UNIDADE GARAGEM
PIEZO	1 UNIDADE GARAGEM
LCD	1 UNIDADE TODAS AS DIVISÕES, EXCETO GARAGEM
ARDUINO	3 UNIDADES DEBAIXO DA CASA

Arquitetura da Casa



AO DETALHE

O ARDUÍNO CÓMODO REALIZA O CONTROLO DE ILUMINAÇÃO DAS DIVISÕES DA CASA, EXCETO GARAGEM, POSTERIORMENTE COMUNICA COM O ARDUÍNO LCD PARA INDICAR QUAIS DAS LUZES ESTÃO ACESAS.

SIMULAÇÃO

PODERÁ CLICAR [AQUI](#) PARA ACEDER AO TINKERCAD OU COPIAR O LINK NO FIM DESTE RELATÓRIO.

Código

ARDUÍNO CÓMODO

```
// C++ code
//

#define LED_HALL 2
#define BUTTON_HALL 4
#define SENSOR_HALL 3

#define LED_SALA 5
#define BUTTON_SALA 7
#define SENSOR_SALA 6

#define LED_QUARTO 8
#define BUTTON_QUARTO 10
#define SENSOR_QUARTO 9

#define LED_WC 11
#define BUTTON_WC 13
#define SENSOR_WC 12
char luzInfo = '0';

int sensor, led, button = 0;

void setup()
{
  Serial.begin(9600);
  pinMode(LED_HALL, OUTPUT);
  pinMode(BUTTON_HALL, INPUT);
}

float calculateDistance(int SENSOR)
{
  float duration, cm;
  pinMode(SENSOR, OUTPUT);
  digitalWrite(SENSOR, LOW);
  delayMicroseconds(2);
  digitalWrite(SENSOR, HIGH);
  delayMicroseconds(5);
  digitalWrite(SENSOR, LOW);
  pinMode(SENSOR, INPUT);
  duration = pulseIn(SENSOR, HIGH);
  cm = (duration / 29.4) / 2;
  return cm;
}
```


Código

ARDUÍNO CÓMODO

CONTINUAÇÃO

```
int divisao(int dv) {  
  switch(dv) {  
    case 1: sensor = SENSOR_HALL;  
            led = LED_HALL;  
            button = BUTTON_HALL;  
  
            break;  
    case 2: sensor = SENSOR_SALA;  
            led = LED_SALA;  
            button = BUTTON_SALA;  
  
            break;  
    case 3: sensor = SENSOR_QUARTO;  
            led = LED_QUARTO;  
            button = BUTTON_QUARTO;  
  
            break;  
    case 4: sensor = SENSOR_WC;  
            led = LED_WC;  
            button = BUTTON_WC;  
  
            break;  
    default: break;  
  }  
  
  float distancia = calculateDistance(sensor);  
  
  if (distancia <= 30 && digitalRead(button) == LOW) {  
    digitalWrite(led, HIGH);  
    return 1;  
  } else if (digitalRead(button) == HIGH && distancia <= 30) {  
    digitalWrite(led, LOW);  
    return 0;  
  } else if (digitalRead(button) == HIGH) {  
    digitalWrite(led, HIGH);  
    return 1;  
  } else {  
    digitalWrite(led, LOW);  
    digitalWrite(led, LOW);  
    return 0;  
  }  
}
```

Código

ARDUÍNO CÓMODO

CONTINUAÇÃO

```
void loop()
{
    //1 -> Hall de entrada
    //2 -> Sala de estar
    //3 -> Quarto
    //4 -> WC
    luzInfo = '0';

    if(divisao(1)){
        luzInfo = '1';
        Serial.write(luzInfo);
    }

    if(divisao(2)){
        luzInfo = '2';
        Serial.write(luzInfo);
    }

    if(divisao(3)){
        luzInfo = '3';
        Serial.write(luzInfo);
    }

    if(divisao(4)){
        luzInfo = '4';
        Serial.write(luzInfo);
    }

    if(luzInfo == '0') {
        Serial.write(luzInfo);
    }
}
```

Código

ARDUÍNO LCD

O Arduino LCD exibe as informações passadas pelo Arduino Cómodo e faz o controlo de temperatura individualmente.

```
// C++ code
//

#include <LiquidCrystal.h>
#define tempSensor A0

LiquidCrystal lcd(2, 3, 4, 5, 6, 7);

void temperatura(){
  int temperatura_raw = analogRead(A0);
  // converting the reading to voltage (millivolts)
  float temperatura_volt = (temperatura_raw / 1024.0f)*5000;

  // Convert to °C

  float temperaturaC = (temperatura_volt - 500)/10;
  lcd.setCursor(0,0);
  lcd.print(temperaturaC);
  lcd.print("C");
}

void setup()
{
  pinMode(LED_BUILTIN, OUTPUT);
  lcd.begin(16, 2);
  Serial.begin(9600);
}
```

Código

ARDUÍNO LCD

CONTINUAÇÃO

```
void loop()
{
    char luz = '\0';

    if(Serial.available() > 0){
        luz = Serial.read();
    }

    temperatura();

    if(luz == '0'){
        lcd.setCursor(0,1);
        lcd.print("                ");
    }

    if(luz == '1'){
        lcd.setCursor(0,1);
        lcd.print("Hall: ON");
    }

    if(luz == '2'){
        lcd.setCursor(0,1);
        lcd.print("Sala: ON");
    }

    if(luz == '3'){
        lcd.setCursor(0,1);
        lcd.print("Quarto: ON");
    }

    if(luz == '4'){
        lcd.setCursor(0,1);
        lcd.print("WC: ON");
    }

    Serial.print(luz);
}
```

Código

ARDUÍNO GARAGEM

O Arduíno Garagem é responsável por todo o controlo de iluminação, sensores e motor da garagem. Como este elemento não precisa de comunicar com o resto da casa, este é um Arduíno próprio.

```
// C++ code
//
#include <Keypad.h>
#include <string.h>

#define buzzer 4
#define motor 2
#define button 3
#define luz 13
const int tamanhoSenha = 4;
char senha[] = {'1','1','1','1'}; //def senha
char codigo[tamanhoSenha];
char letra;
int count = 0;

short a = 0, i= 0, s= 0, j= 0;

//      CONFIGURAÇÃO E FUNÇÕES DO KEYPAD
const int COLUNA = 4;
const int LINHA = 4;
#define MOTORGARAGEM 4;

char teclas[LINHA][COLUNA] =
{
  {'1', '2', '3', 'A'},
  {'4', '5', '6', 'B'},
  {'7', '8', '9', 'C'},
  {'*', '0', '#', 'D'}
};

byte linhas[LINHA] = {12,11,10,9};

byte colunas[COLUNA] = {8,7,6,5};

Keypad keypad = Keypad(makeKeymap(teclas), linhas, colunas, LINHA, COLUNA);

void setup()
{
  Serial.begin(9600);
  pinMode(buzzer, OUTPUT);
  pinMode(motor, OUTPUT);
  pinMode(button, INPUT);
  pinMode(luz, OUTPUT);
}
```

Código

ARDUÍNO GARAGEM

CONTINUAÇÃO

```
void abrirGaragem(){
    digitalWrite(motor, HIGH);
    ligarLuz();
    delay(1000);
    digitalWrite(motor, LOW);
}

void fecharGaragemB(){
    digitalWrite(motor, HIGH);
    ligarLuz();
    delay(1000);
    digitalWrite(motor, LOW);
}

void ligarLuz(){
    digitalWrite(luz, HIGH);
    delay(3000);
    digitalWrite(luz, LOW);
}

void loop(){
    int value = 0;
    char tecla = keypad.getKey();

    while(count <= tamanhoSenha-1){ //verificação tamanho
        tecla = keypad.getKey();
        if (tecla != NO_KEY){
            Serial.print(tecla);
            codigo[count] = tecla;
            count++;
        }
    }
    for(int i = 0; i < tamanhoSenha; i++){//verificação char
        if(codigo[i] == senha[i]){
            continue;
        }else{
            value = 1;
            break;
        }
    }
}
```

Código

ARDUÍNO GARAGEM

CONTINUAÇÃO

```
Serial.println("");  
if(value){  
    Serial.println("Senha Incorreta");  
    tone(buzzer,200, 100);  
}else{  
    Serial.println("Senha Correta");  
    abrirGaragem();  
    while(digitalRead(button) == LOW){  
        continue;  
    }  
    fecharGaragemB();  
}  
count = 0;
```

Conclusão

Em suma, conseguimos aprofundar o nosso conhecimento nesta área e idealizar o que tínhamos pensado no relatório inicial, no entanto, devido a diversos fatores externos infelizmente acabámos por não conseguir realizar a maquete que iria incluir a montagem física do circuito, tendo este ficado apenas pelo TinkerCad.

Realização

GONÇALO BRAGANÇA
INÊS MARQUES
ISABELLA COSTA
RODRIGO BARATA

LINK COMPLETO PARA O TINKERCAD

[HTTPS://WWW.TINKERCAD.COM/LOGIN?NEXT=%2FTHINGS%2F4HAZAVW37BW-COPY-OF-PROJETO-FINAL-EXPERIMENTAL%2FEDITEL%3FSHARECODE%3DZX5OXZ0DID-NGQQOMKPB49U6BHRRUEEJAX2D7XRDWOU](https://www.tinkercad.com/login?next=%2Fthings%2F4HAZAVW37BW-COPY-OF-PROJETO-FINAL-EXPERIMENTAL%2FEDITEL%3FSHARECODE%3DZX5OXZ0DID-NGQQOMKPB49U6BHRRUEEJAX2D7XRDWOU)