#include <Keypad.h>

#include <Servo.h>

#include<SoftwareSerial.h>

SoftwareSerial gsm(8,23);

Servo myservo1;

Servo myservo2;

Servo myservo3;

Servo myservo4;

Servo myservo5;

Servo myservo6;

char text [50];

int flag=0;

boolean state=0;

int alphacount,i,j,l,k,m;

char ch,cha[2],letter;

char alpha[26]={'A','B','C','D','E','F','G','H','I','J','K','L','M','N','O','P','Q','R','S','T','U','V','W','X','Y','Z'};

char brailey[26][10]={{"2#"},{"25#"},{"23#"},{"236#"},{"26#"},{"235#"},{"2356#"},{"256#"},{"35#"},{"356#"},{"28#"},{"258#"},{"238#"},{"2368#"},{"268#"},{"2358#"},{"23568#"},{"2568#"},{"358#"},{"3568#"},{"289#"},{"2589#"},{"3569#"},{"2389#"},{"23689#"},{"2689#"}};

#define Password\_Lenght 10

int pos = 0;

char Data[Password\_Lenght];

char sentmessage[Password\_Lenght];

int found = 0;

int len;

byte data\_count = 0, master\_count = 0;

bool Pass\_is\_good;

char customKey;

char customKey2,customKey3;

const byte ROWS = 4;

const byte COLS = 4;

char keys[ROWS][COLS] = {

{'1', '2', '3', 'A'},

{'4', '5', '6' ,'B'},

{'7', '8', '9' ,'C'},

{'\*', '0', '#' ,'D'}

};

bool door = true;

byte rowPins[ROWS] = {1, 2, 3, 4};

byte colPins[COLS] = {5, 6, 7, A9};

Keypad customKeypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS);

char bword[7]="ABCDIK";

int word\_count=strlen(bword);

int smsg=0;

char dword[2][10]={{"AB"},{"EF"}};

char dmeaning[2][10]={{"EFGH"},{"ABCD"}};

int ddc=0;

int dictdatacount=0;

char key,dictdata[10],dictmean[10],bdictdata[10];

int wc=0,al,pwcount=0,pc;

char pword[10];

void setup()

{

myservo1.attach(9);

myservo2.attach(10);

myservo3.attach(11);

myservo4.attach(12);

myservo5.attach(13);

myservo6.attach(A0);

ServoClose();

}

void loop()

{

customKey = customKeypad.waitForKey();

if(customKey=='A')

{

dictionary();

}

if(customKey=='D')

Displaymsg();

if(customKey=='B'){

ServoOpen1();

ServoOpen2();

ServoOpen3();

ServoOpen4();

ServoOpen5();

ServoOpen6();

delay(1000);

ServoClose();

Sendmsg();

}

}

void clearData()

{

while (data\_count != 0)

{

Data[data\_count--] = 0;

}

return;

}

void ServoOpen1()

{

for (pos = 90; pos >= 0; pos -= 5) {

// in steps of 1 degree

myservo1.write(pos);

delay(0);

}

}

void ServoOpen2()

{

for (pos = 90; pos >= 0; pos -= 5) {

myservo2.write(pos);

delay(0);

}

}

void ServoOpen3()

{

for (pos = 90; pos >= 0; pos -= 5) {

myservo3.write(pos);

delay(0);

}

}

void ServoOpen4()

{

for (pos = 90; pos >= 0; pos -= 5) {

myservo4.write(pos);

delay(0);

}

}

void ServoOpen5()

{

for (pos = 90; pos >= 0; pos -= 5) { // goes from 0 degrees to 180 degrees

// in steps of 1 degree

myservo5.write(pos); // tell servo to go to position in variable 'pos'

delay(0); // waits 15ms for the servo to reach the position

}

}

void ServoOpen6()

{

for (pos = 90; pos >= 0; pos -= 5) { // goes from 0 degrees to 180 degrees

// in steps of 1 degree

myservo6.write(pos); // tell servo to go to position in variable 'pos'

delay(0); // waits 15ms for the servo to reach the position

}

}

void ServoClose()

{

for (pos = 0; pos <= 90; pos += 5) { // goes from 180 degrees to 0 degrees

myservo1.write(pos);

myservo2.write(pos);

myservo3.write(pos);

myservo4.write(pos);

myservo5.write(pos);

myservo6.write(pos);// tell servo to go to position in variable 'pos'

delay(0); // waits 15ms for the servo to reach the position

}

}

/\*void ServoClose()

{

for (pos = 0; pos <= 180; pos += 5) { // goes from 180 degrees to 0 degrees

myservo1.write(pos); // tell servo to go to position in variable 'pos'

delay(15); // waits 15ms for the servo to reach the position

}

}\*/

void Displaymsg()

{

// lcd.setCursor(0, 0);

//lcd.print(" Enter Password");

for(m=0;m<word\_count;m++){

//customKey = customKeypad.getKey();

for(i=0;i<10;i++){

Data[i]='\0';

}

letter=bword[m];

for(i=0;i<26;i++){

if(alpha[i]==letter){

alphacount=i;

found=1;

}

}

if(found==0){

printf("\n");

continue;

}

//

len=strlen(brailey[alphacount]);

for(i=0;i<len;i++){

Data[i]=brailey[alphacount][i];

}

for(i=0;i<26;i++){

if(!strcmp(Data,brailey[i])){

//lcd.clear();

for(k=0;k<len-1;k++){

if(Data[k]=='2'){

ServoOpen1();

}

if(Data[k]=='3'){

ServoOpen2();

}

if(Data[k]=='5'){

ServoOpen3();

}

if(Data[k]=='6'){

ServoOpen4();

}

if(Data[k]=='8'){

ServoOpen5();

}

if(Data[k]=='9'){

ServoOpen6();

}

}

state = !state;

//digitalWrite(ledPin, state);

//text2speech.setPitch(6); //higher values = lower voice pitch

cha[0]=alpha[i];

strcpy(text,cha);

//text2speech.sayText(text);

//ServoOpen6();

//ServoOpen1();

// lcd.print(" Door is Open");

door = 0;

clearData();

flag=1;

found=0;

}

}

if(flag==0){

ServoOpen1();

ServoOpen2();

ServoOpen3();

ServoOpen4();

ServoOpen5();

ServoOpen6();

door=0;

clearData();

}

clearData();

delay(2000);

ServoClose();

}

return;

}

void Sendmsg()

{

// lcd.setCursor(0, 0);

//lcd.print(" Enter Password");

for(i=0;i<10;i++){

Data[i]='\0';

}

data\_count=0;

while(1){

//ServoOpen1();

customKey2 = customKeypad.waitForKey();

if (customKey2) // makes sure a key is actually pressed, equal to (customKey != NO\_KEY)

{

Data[data\_count] = customKey2; // store char into data array

//lcd.setCursor(data\_count, 1); // move cursor to show each new char

//lcd.print(Data[data\_count]); // print char at said cursor

data\_count++; // increment data array by 1 to store new char, also keep track of the number of chars entered

}

// if (data\_count == Password\_Lenght - 1) // if the array index is equal to the number of expected chars, compare data to master

if (Data[data\_count-1] == '#')

{

l=data\_count-1;

for(i=1;i<l;i++){

for(j=0;j<l-i;j++){

if(Data[j]>Data[j+1]){

ch=Data[j];

Data[j] = Data[j+1];

Data[j+1]=ch;

}

}

}

for(i=0;i<26;i++){

if(!strcmp(Data,brailey[i])){

//lcd.clear();

for(k=0;k<data\_count-1;k++){

if(Data[k]=='2'){

ServoOpen1();

}

if(Data[k]=='3'){

ServoOpen2();

}

if(Data[k]=='5'){

ServoOpen3();

}

if(Data[k]=='6'){

ServoOpen4();

}

if(Data[k]=='8'){

ServoOpen5();

}

if(Data[k]=='9'){

ServoOpen6();

}

}

delay(1000);

state = !state;

//digitalWrite(ledPin, state);

// text2speech.setPitch(6); //higher values = lower voice pitch

cha[0]=alpha[i];

strcpy(text,cha);

// text2speech.sayText(text);

//ServoOpen6();

//ServoOpen1();

// lcd.print(" Door is Open");

sentmessage[smsg]=alpha[i];

smsg++;

door = 0;

//clearData();

flag=1;

for(i=0;i<10;i++){

Data[i]='\0';

}

break;

}

}

if(flag==0){

ServoOpen1();

ServoOpen2();

ServoOpen3();

ServoOpen4();

ServoOpen5();

ServoOpen6();

//door=0;

//clearData();

}

/\* if (!strcmp(Data, Master)) // equal to (strcmp(Data, Master) == 0)

{

//lcd.clear();

ServoOpen();

// lcd.print(" Door is Open");

door = 0;

clearData();

}

else if (!strcmp(Data, Master1)) // equal to (strcmp(Data, Master) == 0)

{

//lcd.clear();

ServoOpen();

// lcd.print(" Door is Open");

door = 0;

clearData();

}

else

{

// lcd.clear();

//lcd.print(" Wrong Password");

delay(1000);

door = 1;

clearData();

}\*/

customKey3 = customKeypad.waitForKey();

if(customKey3=='\*'){

ServoClose();

/\* gsm.println("AT+CMGF=1");

gsm.println("AT+CMGS=\"+919441688480\"\r"); //replace x by your number

gsm.println(sentmessage);

gsm.println((char)26);\*/

//psm();

for(i=0;i<10;i++){

sentmessage[i]='\0';

}

smsg=0;

return;

}

else{

ServoClose();

for(i=0;i<10;i++){

Data[i]='\0';

}

data\_count=0;

//continue;

}

}

//Sendmsg();

flag=0;

}

}

void dictionary(){

for(i=0;i<10;i++){

bdictdata[i]='\0';

}

ServoOpen1();

delay(1000);

ServoClose();

while(1){

key=customKeypad.waitForKey();

if(key!='\*'){

bdictdata[ddc]=key;

ddc++;

}

if(key=='#'){

for(i=0;i<26;i++){

if(!strcmp(bdictdata,brailey[i])){

//lcd.clear();

data\_count=strlen(bdictdata);

for(k=0;k<data\_count-1;k++){

if(bdictdata[k]=='2'){

ServoOpen1();

}

if(bdictdata[k]=='3'){

ServoOpen2();

}

if(bdictdata[k]=='5'){

ServoOpen3();

}

if(bdictdata[k]=='6'){

ServoOpen4();

}

if(bdictdata[k]=='8'){

ServoOpen5();

}

if(bdictdata[k]=='9'){

ServoOpen6();

}

}

//delay(1000);

cha[0]=alpha[i];

strcpy(text,cha);

dictdata[wc]=alpha[i];

wc++;

door = 0;

//clearData();

flag=1;

for(i=0;i<10;i++){

Data[i]='\0';

bdictdata[i]='\0';

}

ddc=0;

delay(1000);

ServoClose();

break;

}

else{

ServoOpen1();

ServoOpen2();

ServoOpen3();

ServoClose();

}

}

}

if(key=='\*'){

//ServoOpen1();

//delay(1000);

ServoClose();

for(i=0;i<2;i++){

if(!strcmp(dictdata,dword[i])){

ServoOpen2();

//lcd.clear();

dictdatacount=strlen(dmeaning[i]);

for(m=0;m<dictdatacount;m++){

for(al=0;al<26;al++){

//dictmean[m]=dmeaning[i][m];

if(dmeaning[i][m]==alpha[al]){

pwcount=strlen(brailey[al]);

for(pc=0;pc<pwcount;pc++){

pword[pc]=brailey[al][pc];

}

for(k=0;k<pwcount-1;k++){

if(pword[k]=='2'){

ServoOpen1();

}

if(pword[k]=='3'){

ServoOpen2();

}

if(pword[k]=='5'){

ServoOpen3();

}

if(pword[k]=='6'){

ServoOpen4();

}

if(pword[k]=='8'){

ServoOpen5();

}

if(pword[k]=='9'){

ServoOpen6();

}

}

delay(1000);

ServoClose();

}

//delay(1000);

//ServoClose();

}

}

delay(2000);

}

}

wc=0;

return;

}

}

# }