```
#include <LiquidCrystal.h>
#include <Keypad.h>
#include <SoftwareSerial.h>
#include <EEPROM.h>
SoftwareSerial mySerial(9, 10);


const byte ROWS = 4;
const byte COLS = 4;

char hexaKeys[ROWS][COLS] = {
  {'1', '2', '3', 'A'},
  {'4', '5', '6', 'B'},
  {'7', '8', '9', 'C'},
  {'*', '0', '#', 'D'}
};
byte rowPins[ROWS] = {46, 47, 48, 49};
byte colPins[COLS] = {50, 51, 52, 53};


Keypad customKeypad = Keypad( makeKeymap(hexaKeys), rowPins, colPins,
ROWS, COLS);
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

int reed1 = 22;
int reed2 = 23;
int reed3 = 24;
int buzzer = 34 ;
int reed1out = 28;
int reed2out = 29;
int reed3out = 30;
int pir = 31;
int lock_on = 0;
int al_on = 0;
int pass;
String mobile;
int inco = 0 ;

void setup() {
  Serial.begin(9600);
  mySerial.begin(9600);
  pinMode ( reed1out , OUTPUT );
  pinMode ( reed2out , OUTPUT );
  pinMode ( reed3out , OUTPUT );
  pinMode ( buzzer , OUTPUT );
  pinMode ( reed1 , INPUT_PULLUP );
  pinMode ( reed2 , INPUT_PULLUP );
  pinMode ( reed3 , INPUT_PULLUP );
  pinMode ( pir , INPUT );
  digitalWrite(reed1out, LOW);
  digitalWrite(reed2out, LOW);
  digitalWrite(reed3out, LOW);
  lcd.begin(20, 4);
  lcd.setCursor(4, 1);
```

```
  lcd.print("Burglar Alarm");
  lcd.setCursor(6, 2);
  lcd.print(" System");
  phone_read();
  pin_read();
  delay(5000);
  lcd.clear();
  Serial.print(pass);
  Serial.print("\n");
  Serial.print(mobile);
  display_home();
}

void loop() {
  char choice = customKeypad.getKey();
  if ( lock_on == 1) {
    if ((digitalRead(reed1) != LOW || digitalRead(reed2) != LOW ||
digitalRead(reed3) != LOW || digitalRead(pir) == HIGH) && al_on == 0 ) {
      al_on == 1;
      digitalWrite(buzzer, HIGH);
      lcd.clear();
      lcd.setCursor(0, 0);
      lcd.print("Waiting...");
      sms();
      alarm();
    }
  }
  if (al_on == 0)
    if (choice == '#' )
    {
      int check = pin_entry();
      if (check == 1)
        unlock();
      else
      {
       inco ++;
       if(inco > 2 )
       {
         lcd.clear();
         lcd.print("waiting...");
         smsco();
         inco = 0;
       }
       lcd.clear();
       lcd.print("Press # to Unlock:");
       loop();
      }
    }
}

void sms() {

  mySerial.println("AT+CMGF=1");
  delay(1000);
```

```
  mySerial.println("AT+CMGS=\"+91" + mobile + "\"");
  delay(1000);
  mySerial.println("Intrusion Detected. Dial 100. ");
  delay(1000);
  mySerial.println((char)26);
  delay(3000);
}

void smsco() {

  mySerial.println("AT+CMGF=1");
  delay(1000);
  mySerial.println("AT+CMGS=\"+91" + mobile + "\"");
  delay(1000);
  mySerial.println("Three invalid attempts detected.");
  delay(1000);
  mySerial.println((char)26);
  delay(3000);
}

void alarm() {
  lcd.clear();
  int flag = pin_entry();
  if (flag == 1)
    unlock();
  else
  {
    lcd.clear();
    lcd.setCursor(4, 1);
    lcd.print("INVALID PIN !!");
    lcd.setCursor(5, 2);
    lcd.print("Try again !!");
    delay(1500);
    alarm();
  }
}

void display_home () {
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("1.Lock");
  lcd.setCursor(0, 1);
  lcd.print("2.Change Pin");
  lcd.setCursor(0, 2);
  lcd.print("3.Change Mob Number ");
  lcd.setCursor(0, 3);
  lcd.print("Enter Choice : ");
  lcd.setCursor(16, 3);
  char choice = customKeypad.getKey();
  while (choice == NO_KEY) {
    choice = customKeypad.getKey();
    if ( choice != NO_KEY) {
      lcd.print(choice);
      delay(2000);
```

```
      switch (choice)
      {
        case '1' : lock();
          break;
        case '2' : change_pin();
          break;
        case '3' : change_number();
          break;
        default : error_msg();
          break;
      }
    }
  }
}

void lock() {

  int check = pin_entry();
  if ( check == 1) {
    digitalWrite(reed1out, LOW);
    digitalWrite(reed2out, LOW);
    digitalWrite(reed3out, LOW);
    lock_on = 1;
    lcd.clear();
    lcd.print("Press # to Unlock:");
  }
  else
    error_msg();
}


int pin_entry() {
  lcd.clear();
  int flag = 0 , ai  , temp = 0;
  lcd.print("Enter Pin : ");
  lcd.setCursor(0, 1);
  char choice;
  while ( flag < 4) {
    choice = customKeypad.getKey();
    if (choice != NO_KEY ) {
      ai = choice - '0';
      temp = (temp * 10) + ai;
      flag += 1;
      lcd.print("*");
    }
  }
  if ( temp == pass)
    return 1;
  else
    return 0;
}
void unlock() {
  lock_on = 0;
  al_on = 0;
```

```
  digitalWrite(buzzer , LOW);
  display_home();
}

void error_msg() {
  lcd.clear();
  lcd.setCursor(2, 0);
  lcd.print("Invalid choice !!");
  lcd.setCursor(5, 1);
  lcd.print("Try Again !!");
  delay(3000);
  lcd.clear();
  display_home();
}

void change_pin() {
  lcd.clear();
  int temp = 0 , temp1 = 0 , flag = 0 , ai;
  lcd.print("Enter Pin : ");
  lcd.setCursor(0, 1);
  int choice;
  while ( flag < 4) {
    choice = customKeypad.getKey();
    if (choice != NO_KEY ) {
      ai = choice - '0';
      temp = (temp * 10) + ai;
      flag += 1;
      lcd.print("*");
    }
  }
  if ( temp == pass) {
    lcd.clear();
    lcd.print("Enter new pin : ");
    lcd.setCursor(0, 1);
    temp = 0;
    flag = 0;
    while ( flag < 4) {
      choice = customKeypad.getKey();
      if (choice != NO_KEY ) {
        ai = choice - '0';
        temp = (temp * 10) + ai;
        flag += 1;
        lcd.print("*");
      }
    }
    lcd.setCursor(0, 2);
    lcd.print("Confirm pin : ");
    lcd.setCursor(0, 3);
    flag = 0;
    while ( flag < 4) {
      choice = customKeypad.getKey();
      if (choice != NO_KEY ) {
        ai = choice - '0';
        temp1 = (temp1 * 10) + ai;
```

```
        flag += 1;
        lcd.print("*");
      }
    }

    if ( temp == temp1) {
      pass = temp;
      pin_write(pass);
      lcd.clear();
      lcd.setCursor(3, 1);
      lcd.print("Pin Changed !!");
      delay(5000);
      lcd.clear();
    }
    else
      error_msg();
  }
  else
    error_msg();

  display_home();
}

void change_number() {
  lcd.clear();
  int flag = 0 , ai , temp = 0, i;
  lcd.print("Enter Pin : ");
  lcd.setCursor(0, 1);
  char choice;
  char ch;
  char num[12] , num1[12] ;
  while ( flag < 4) {
    choice = customKeypad.getKey();
    if (choice != NO_KEY ) {
      ai = choice - '0';
      temp = (temp * 10) + ai;
      flag += 1;
      lcd.print("*");
    }
  }
  if ( temp == pass) {
    lcd.clear();
    lcd.print("Enter new number : ");
    lcd.setCursor(0, 1);
    temp = 0;
    flag = 0;
    i = 0;
    while ( flag < 10) {
      ch = customKeypad.getKey();
      if (ch != NO_KEY ) {
        num[i] = ch; ;
        flag += 1;
        i += 1;
        lcd.print(ch);
```

```
      }
    }
    lcd.clear();
    lcd.print("Confirm new number: ");
    lcd.setCursor(0, 1);
    flag = 0;
    i = 0;
    while ( flag < 10) {
      ch = customKeypad.getKey();
      if (ch != NO_KEY ) {
        num1[i] = ch;
        flag += 1;
        i += 1;
        lcd.print(ch);
      }
    }
    Serial.print("\n");
    Serial.print(num);
    Serial.print("\n");
    Serial.print(num1);
    Serial.print("\n");
    String n1(num);
    String n2(num1);
    if ( n1.equals(n2) ) {
      phone_write(num1);
      mobile = n1;
      lcd.clear();
      lcd.setCursor(2, 1);
      lcd.print("Number Changed !!");
      delay(5000);
      lcd.clear();
    }
    else
      error_msg();
  }
  else
    error_msg();
  display_home();
}

void phone_read()
{
  int addr = 14;
  int s;
  char num[10];
  int i = 0;
  while (addr > 4 )
  {
    num[i] = EEPROM.read(addr) + '0' ;
    i++;
    addr--;
  }
  String st(num);
  mobile = st;
```

```
}

void phone_write( char num[11] )
{
  int s;
  int i = 0;
  int addr = 14;
  while (addr > 4)
  {
    s = num[i] - '0';
    EEPROM.write(addr, s);
    i++;
    addr--;
  }
  String st1(num);
  mobile = st1;
}

void pin_write( int pas)
{
  int s;
  int addr = 3;
  while (pas > 0)
  {
    s = pas % 10;
    EEPROM.write(addr, s);
    pas /= 10;
    addr--;
  }
  pass = pas;
}

void pin_read()
{
  int s;
  int pas = 0;
  int addr = 0;
  while (addr < 4)
  {
    s = EEPROM.read(addr);
    pas = pas * 10 + s;
    addr++;
  }
  pass = pas;
}
```