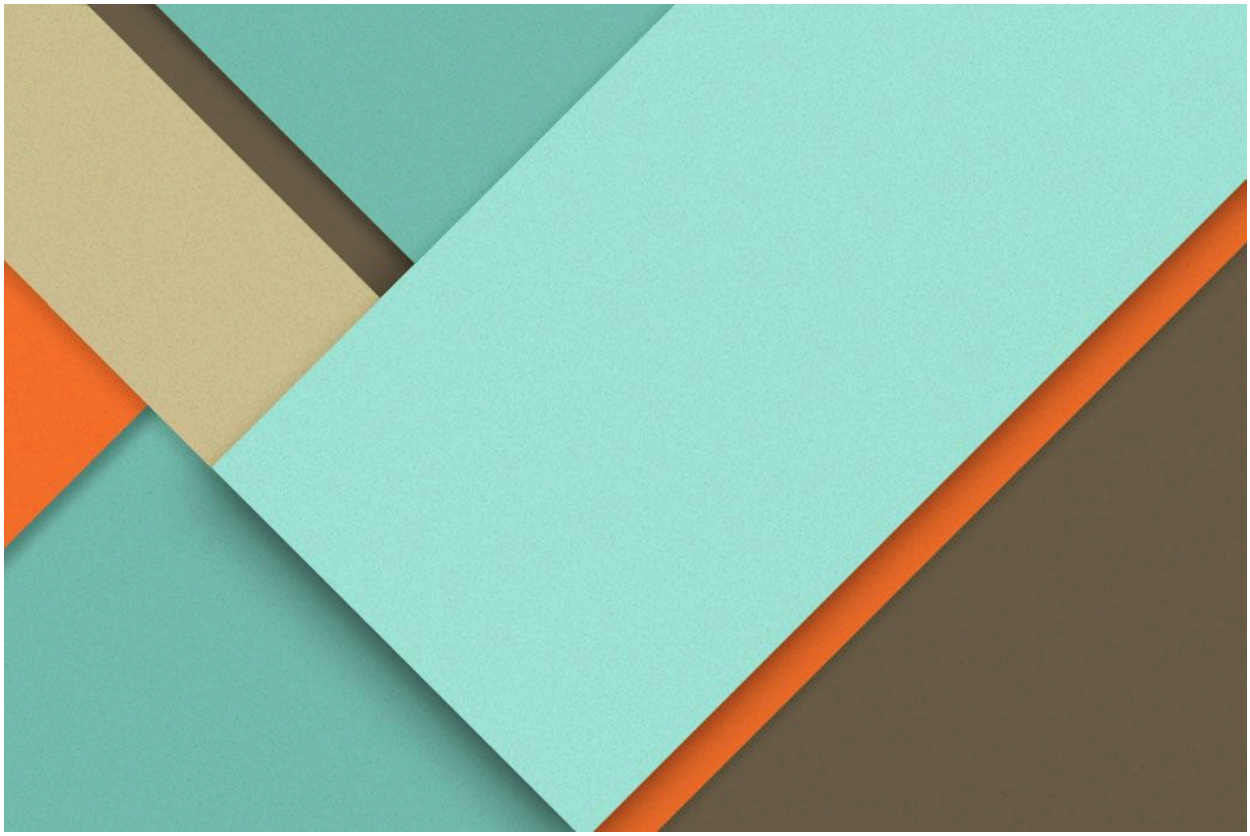


Smart Home System

14.09.2024

—

Team 8

Names

Ziad Ismail Hassan

Seif-Eldeen Hany

Ahmed Samy Ezzat

Ahmed Walid Abdelnaby

Overview

This project focuses on developing a smart home system with enhanced security and automation. It includes a door sector equipped with advanced face recognition, password verification, and alert mechanisms, and an interior sector featuring sensors for temperature, motion detection, and more. Designed to integrate cutting-edge technology, this system aims to provide a comprehensive and user-friendly solution for modern home automation, ensuring both convenience and safety.

Sections

1. Computer Vision
2. Hardware Project Deliverables
3. Micro-controllers

Details

I. Computer Vision:

Overview

In the first week we tired to plan how are we going to work on the face recognition section,data management and The GUI,The study of tkinter and sqlite3 libraries was the beginning, we used each library in separate codes, then we tried to combine the three sections (database part, data collection part, face recognition part) together In one code in addition to some work on the cv2 code to enhance the quality of the face recognition system.

1. Libraries

We used :

A. sklearn

We used from this library the module sklearn.metrics and from the pairwise metrics we used the cosine similarity function that computes the similarity

between pair of feature vectors, we used it to compare the live image we get from the camera with the images in the homeowners' database and we adjusted the similarity percentage so it works perfectly.

B. Sqlite3

We used this library to create the database that we will save the homeowners data in so it is easy to reach for it when we need it, and it really helps with error handling.

C. Os

We used this library to create the folders we save the images of homeowners in and to get the images when we do the face recognition part.

D. cv2

It is the main library of our face detection part, we used it to make the cam capture frames and then we can operate on them and we used from this library a pre-trained Haar cascade classifier (Haar cascade is a machine learning object detection method) for detecting faces from the XML file haarcascade_frontalface_default and it has a high speed of detection which is really good and we needed some delay sometimes in the system so instead of importing time library we did use the waitkey() function to delay when we need, and we used it to change the color of the frames to gray for face detection because the Haar cascade classifier works only with gray images and we used the class HOGDescriptor of the cv2 library to compute the feature vector for the images before comparing them so we reduce the lightning effect, and because the function compute() in class HOGDescriptor deals with relevantly small number of pixels we needed to use the resize() function to resize the images but and also the HOGDescriptor returns multi-dimensional arrays but the cosine similarity needs flat vectors so we use the function flatten().

E. Tkinter

We used this library for the gui implementation .

F. Serial

We used this library for communication with arduino.

2. How the code works

First, we open the cam trying to detect a face when we do we start capturing images of the face for 3 seconds and every frame we capture we compare it with the database to see if it is found and if any image matches the database images we

notify the arduino to open the door and start the system and the gui window for system management pops up for that homeowner entering the house and if the face is not recognized as one of the homeowners, then the stranger gui pops up giving the homeowner the options to open the door only or open the door and add the person or he can choose to not act so the door will stay closed, if he chose to open the door the code only notify the arduino to open the door and start the system and if he chose to add the user too then the system management gui pops up to let the homeowner type in the user id and name then press add user to add him, on pressing the button, the camera opens again capturing multiple images of the person to add him to the database, there also an option in the system management gui to delet user from database and another option to update password when the update password button is pressed it only notify the arduino to take the input password on the keypad and and replace it with the current password.

II. Hardware Project Deliverables

PCB design

We used LCSC as a supplier and took all parts parameters from him and we downloaded 3D models from SnapMagic, we also applied the rules from JLCPCB

We used an A4 size for the schematic design, there is a schematic library containing all parts designs and PCB library containing all parts PCB design using Top Overlay and Mechanical29 layer for designators, Top Layer for smd pads, Mechanical 1 for 3D models and multilayer for holes.

III. Micro-controllers

Automated Door System with Password Authentication, Temperature-Based Fan Control, and Lighting System

Overview:

This system is designed to control access to a door using a password input via a keypad. Upon correct password entry or receiving a signal from face recognition, the door unlocks via a servo motor, and a fan's speed is adjusted based on the ambient temperature. Additionally, the system includes a lighting control mechanism that adjusts brightness according to the surrounding light levels and a buzzer for incorrect password alerts.

Key Components:

- **Servo Motor**: Controls the door, opening and closing it based on authentication.
- **Keypad**: Allows the user to enter a 4-digit password to unlock the door.
- **Temperature Sensor**: Reads the temperature and adjusts the fan speed.
- **Fan**: Controlled by a DC motor, its speed is adjusted based on the temperature.
- **Light Sensor (LDR)**: Monitors ambient light and controls the brightness of the lighting system.
- **PIR Sensor**: Detects movement inside the room and can trigger warnings if unauthorized movement is detected.
- **Buzzer**: Sounds an alert when an incorrect password is entered.
- **LEDs**: Provide visual feedback, indicating fan speed or errors.

System Workflow:

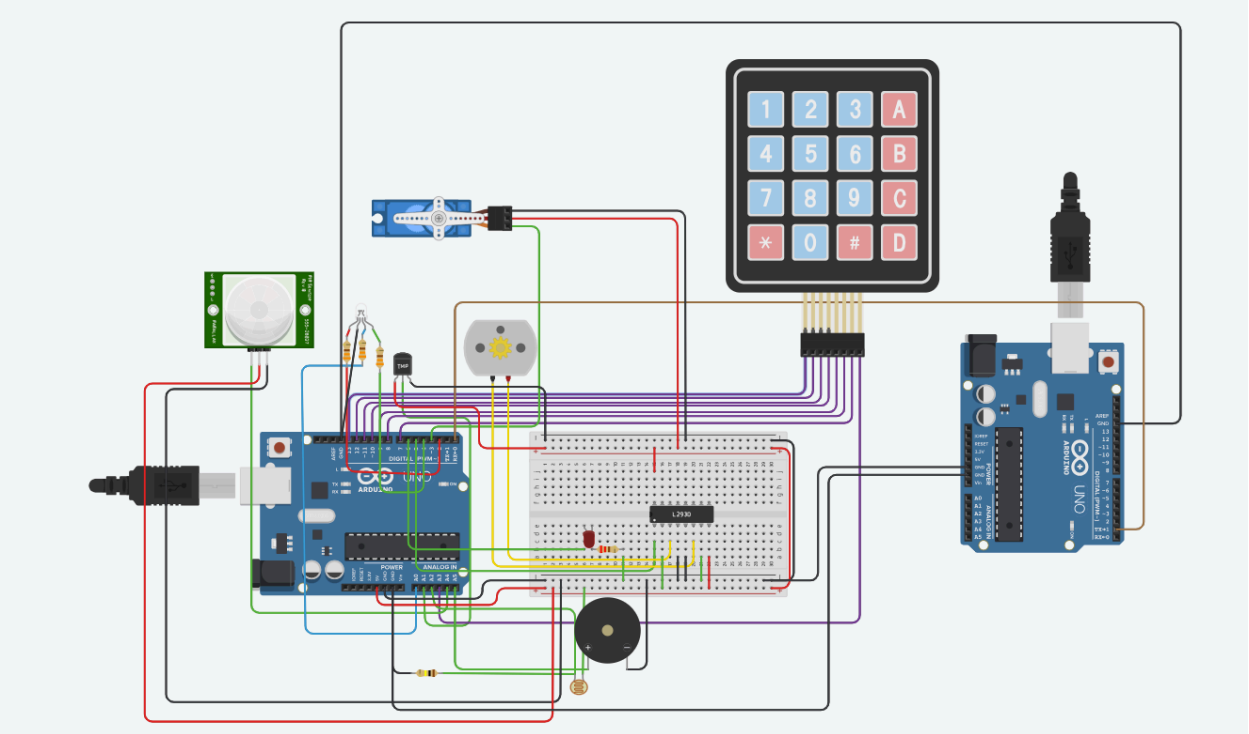
1. **Password Entry and Authentication:**
    - The system begins by waiting for the user to input a 4-digit password via the keypad.
    - A password is considered valid if the user presses `*` before entering the 4 digits and `#` afterward.
    - If the correct password is entered, the door opens by rotating the servo motor, the fan speed is set based on the temperature, and the lighting system adjusts its brightness based on the light sensor reading.
    - If an incorrect password is entered, the buzzer sounds, and the user is prompted to try again.
2. **Face Recognition Input:**
    - The system can also interact with a face recognition module. When a signal `'o'` is received via serial communication, it signifies that the recognized face matches the authorized user, and the door unlocks without password input.
    - If a signal `'u'` is received, the system prompts the user to update the password, allowing new password entry via the keypad.
3. **Temperature-Based Fan Control:**
    - The fan's speed is adjusted based on the ambient temperature read from the temperature sensor. The speed is mapped across three ranges:
        - Low speed (below 20°C)
        - Medium speed (20-30°C)
        - High speed (above 30°C)
    - Visual indicators (LEDs) are used to display the fan's current speed.

4. **Lighting System:**
    - The lighting system's brightness is dynamically controlled based on the readings from the LDR sensor. As the ambient light changes, the lighting system adjusts its brightness accordingly.
5. **Movement Detection:**
    - The system monitors movement within the room using the PIR sensor. If movement is detected after the door is unlocked, a warning is issued. If no movement is detected for 30 minutes after the door is unlocked, the system automatically resets, turning off the fan and lights.
6. **System Reset:**
    - After 30 minutes of inactivity (no movement detected), the system resets itself, turning off the motor, LEDs, and lights, ensuring energy efficiency and security.

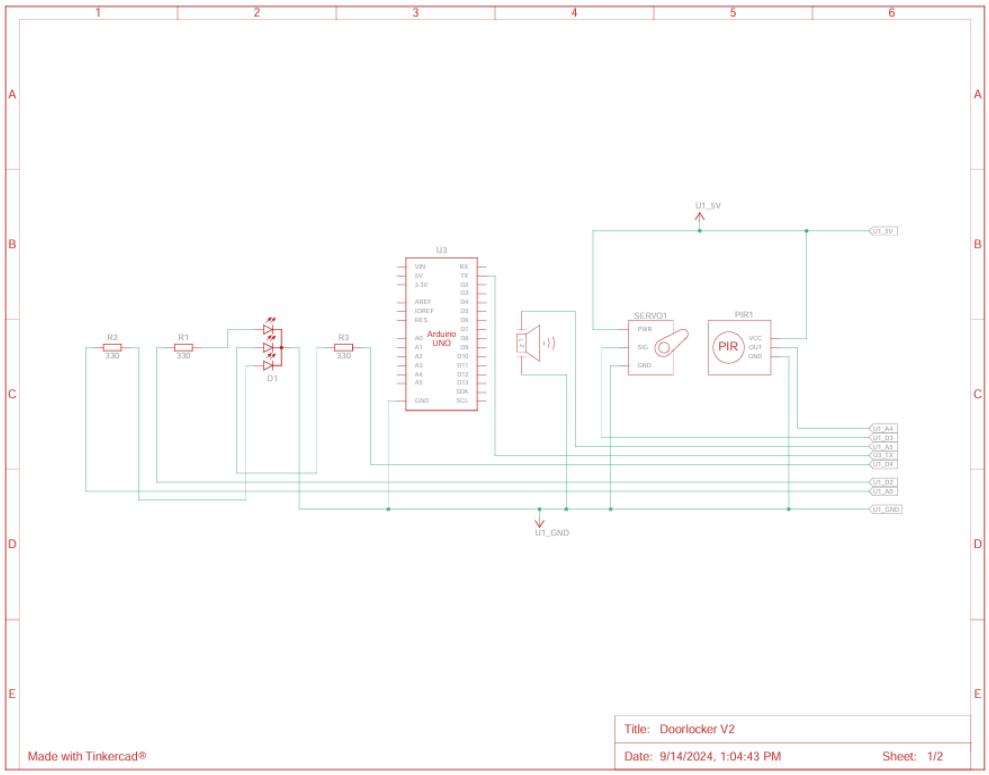
IV. Tinkercad Design

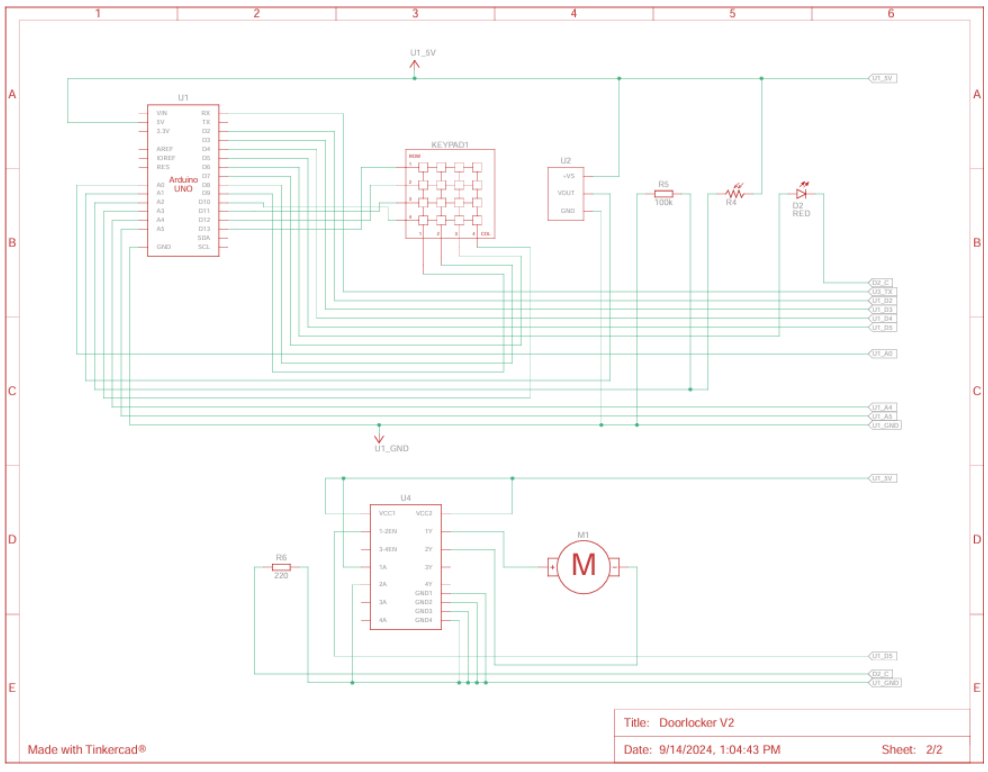
Replacing **Atmega8A** by **Arduino UNO** and **Laptop** by **Arduino UNO** to send “o” to open the door or “u” to update the password. (Link For the design:- Smart Door Locker)

Circuit view:-



Schematic:-


Arduino UNO
SERVO1
PIR1
PIR
U1_5V
U1_GND
R2
R1
R3
D1
Made with Tinkercad®
Title: Doorlocker V2
Date: 9/14/2024, 1:04:43 PM
Sheet: 1/2


Arduino UNO
KEYPAD1
U2
R5
100K
R4
D2
RED
U4
M1
M
R6
220
U1_5V
U1_GND
Made with Tinkercad®
Title: Doorlocker V2
Date: 9/14/2024, 1:04:43 PM
Sheet: 2/2

V. Problems & How We Overcome?

1. A way to use arduino code instead of normal C code

We successfully utilized the Arduino Uno as an In-System Programmer (ISP) to burn the bootloader onto an ATmega8A microcontroller. Additionally, we were able to upload Arduino code to the microcontroller using the Arduino Uno. Thus, we can effectively operate the ATmega8A as if it were part of the Arduino platform.

2. Didn't find FTDI Module

We overcome this by using **Arduino Uno** as **FTDI Module** by making code to make it transmit the signal coming from the laptop from face recognition python code and filter it to be either "o" (Opening the door) or "u" (Updating password).

VI. Important Links

1. **GitHub Link For Project :-**

   Project Link For More Information

2. **Video For Project On Real Life:-**

   Video For the System

Note:- All Codes and Data will be attached in the file.zip that will be delivered with this report