Secured Smart House Prototyping

Serik Zhilibayev
Department of Computer Science
Nazarbayev University
Astana, Kazakhstan
serik.zhilibayev@nu.edu.kz

Aibek Ziyashev
Department of Computer Science
Nazarbayev University
Astana, Kazakhstan
aibek.ziyashev@nu.edu.kz

Altyn Zhelambayeva
Department of Computer Science
Nazarbayev University
Astana, Kazakhstan
altyn.zhelambayeva@nu.edu.kz

Advisor: Professor Adnan Yazici

***Abstract*—This project provides an implementation of smart house with home automation, security, person identification and automatic notifications. Home automation is implemented using light sensor and actuators in the house such as LEDs and servo motors (for window control). Security system is implemented using gas, humidity, human infrared sensors that collect the data from the environment. The user can control actuators remotely, view environmental sensor readings and real-time video streaming using user-friendly web and mobile applications. The readings of environmental sensors are analyzed on the server for gas and water leak and potential intrusion using gas, humidity and human infrared sensors. The camera recognizes the face and identifies whether the person is a resident of the house or an unknown person. Automatic notification system through the Telegram Bot notifies the user if an unknown person entered the house or if the system detected gas or water leak. Telegram Bot also provides options for the user for further actions.**

***Index Terms*—IoT, smart house, automation, security, person identification, automatic notification, sensors and actuators**

I. Introduction

Internet of Things (IoT) is defined as the connection of multiple devices to the Internet to collect sensing data, perform smart monitoring, recognition and administration [1]. IoT principles were effectively utilized in smart house systems implementations to enable real time domestic monitoring and sensing. Smart house is an intelligent system equipped with the variety of sensing devices that provide homeowners comfort, security and convenience [2]. A number of smart house implementations addressed home automation and security systems to provide homeowners with comfort and prevent unexpected intrusions, gas or water leak. The aim of this project is to implement a smart house system by integrating hardware and software components and using available algorithms and tools to implement automatic notification and person identification systems. In the first part of the course Senior Project I, we implemented home automation, movement detection, live-streaming and automatic notification system. In Senior Project II, we extend the security component of the system by implementing face recognition and person identification, improve the architecture of the system by establishing a wireless connection between hardware components of the system and develop mobile application.

The first component of our smart house project is home automation system which is implemented using environmental sensors such as light, humidity, gas sensors and actuators such as LEDs and servo motors. Each of the three rooms in smart house system has Arduino microcontroller with connected environmental sensors that collect the data about the house environment. Three Arduino microcontrollers communicate through wireless connection. Then the data is sent to the server through the sync (Raspberry Pi). The readings of the sensors are then displayed on the user-friendly web and mobile interfaces that enable the user to monitor environmental conditions of the house remotely. Through the web and mobile applications, the user is also able to control system's actuators: servo motors on windows and LEDs. The user can open or close the window in any room and turn on and off LED in any room.

The security component of our system detects gas or water leaks and potential intrusion. The readings of gas, water and human infrared sensors are processed on the server and analyzed for the gas, water leak and the level of activity in the house. Automatic notification system enables the user connected to the Internet to receive notifications from Telegram Bot in case any potential intrusion, gas or water leak is detected. There are several options for further actions provided to the user on the Telegram Bot interface. The user can command Telegram Bot to contact police, firefighters or other emergency contact. These options are available to the user independent of the Telegram notification, so that the user can anytime send commands to Telegram Bot to call any emergency contact. We extended the security component of the system by implementing a face recognition and person identification systems for intruder detection. In case the identified person is one of the house residents, the system identifies the person and no notification is sent to the homeowner. In contrast, when non-resident person enters the house, the person identification system recognizes the person as unknown and notifies home owner about the potential intrusion.

The second section of the paper provides an overview of previous works that addressed smart house system implementation focusing on home automation and security. Section III discusses the overall architecture of the project. Hardware and software architectures are described in sections IV and V, respectively.

II. Related works

Simple implementation and design of a home automation system using Arduino is performed in [3]. The main idea of the research is to give a user an individual ability to remotely or automatically control things around the home. They use Arduino Mega for the controller, relay switches, sensors (Temperature, PIR, Light, Gas, Power Board sensors), LCD, Arduino Wifi Shield, Bluetooth Shield, Android OS mobile phone, and Internet (web browsers). Arduino communicates with the internet by running in itself micro web server through HTTP requests. Android application online provides ON/OFF commands. The mobile application can send commands through Bluetooth also. In the web application, we can see the readings of sensors in each room, if some alerts occurred you can see it from the web page. The server is based on PHP, and communication is transferred through JSON.

The same set of sensors and devices are used to implement both smart security and home automation systems in the research [4]. The system reacts on any unexpected intrusions and notifies the user with the voice call. The system provides the user with an opportunity to analyze the situation to identify whether there is a burglar or users unexpected guest.. After notifying the user, the system provides options for the user either to trigger an alarm system, call police or to open the door and accommodate a guest. The system is implemented using TI-CC3200 Launchpad board with an embedded microcontroller and Wi-Fi. The advantage of the system is that it does not require user to be connected to the internet since only the board needs to have internet connection. It only requires the user to use key-pad when receiving a voice call. The system is also platform independent and can be used from phones with different operating systems. Passive Infrared Sensors (PIR) were used to detect the infrared wavelength emitted by the human body or animals in the proximity. The sensors might also be calibrated by setting higher sensitivity threshold in order to ignore the thermal energy emitted by domestic pets.

Another home automation and security system was designed and implemented in [5]. The system design consists of inputs, outputs and the Arduino-Mega microcontroller 2560 microcontroller. The PIR sensor, LM35 temperature sensor, MQ2 gas and smoke sensor as well as keypad provide input to the system. After the information is processed in the microcontroller, it produces audio-visual outputs which include an alarm, LCD screen, LED. The user can activate and deactivate the system by pressing a password on the keypad. When the system is active and PIR sensors detect the motion, Raspberry Pi sends signal to the web-cam which captures the images that are then sent to the user. When the temperature reading exceeds 65 C or the smoke level exceeds 200 ppm, the user is notified with fire and smoke alerts respectively. Similar to [9], SIM 900a GSM technology was used to connect with the user through AT commands. However, it is suggested by the authors that the limitation of this technology is that SMS alerts can be sent only when there are no network problems.

Using Zigbee to monitor sensing devices was proposed in [9], [10]. Smart security system with face recognition based on wireless network, Zigbee and Principal Component Analysis was implemented in [10]. The user can monitor visitors and control the electromagnetic door lock on the web page that streams in the real time. The monitoring and control system consists of two parts: Wireless Information Unit (WIU) that is implemented with Raspberry Pi and Wireless Control Unit (WCU). The unexpected motion is detected, Raspberry Pi sends SMS and email to web-server, the system notifies the user about the motion with the photo and names of detected people. The recognition system is based on the decomposition of the image into principal components which are the set of eigenvectors of the covariance matrix of the set of faces. Each face is then compared with the linear combination of eigenvectors existing in the database. Like in [5] and [9], SIM900A GSM module was used to communicate with the user. RS232 level converter circuitry of the module enables it to connect to the serial interface of the microcontroller. The SMS module, therefore, connects the embedded processor with the GSM network.

A number of approaches were developed for face recognition and person identification systems. FaceNet system based on the convolutional neural network measures the similarity of the face by mapping the images of the faces to the Euclidean space [6]. Face recognition is then implemented using the created space with FaceNet embeddings as feature vectors. The system uses 128 bytes per face and performs with 99.63% accuracy on the Labeled Faces in the Wild (LFW) dataset. In another study, the researchers integrated Histograms of Oriented Gradients (HoG) with cascade-rejectors approach for person identification [7]. HoGs are the features of variable-size blocks. The reseaercher used AdaBoost to select the features and the appropriate set of blocks. The system maintains similar accuracy level to other available approaches and processes 5 to 30 frames per second. The review of the existing edge and gradient based descriptors was performed in [8]. They found that fine-scale gradients, fine orientation binning, relatively coarse spatial binning and high-quality local contrast normalization were crucial for the performance of the algorithm. Their experiemental results suggest that HoG descriptors perform more accurately in comparison to other existing feature sets for human detection.

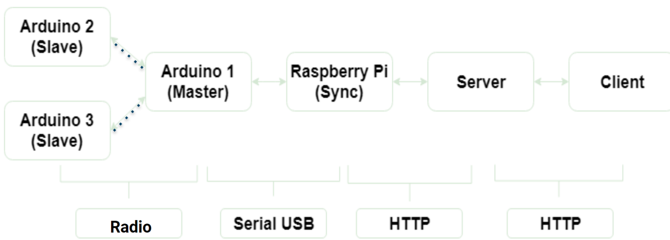

Fig. 1. Overview of the overall architecture

III. Overall System Design

Hardware components of the system implemented using three Arduinos (one in each room) connected through I2C communication which is the protocol that enables the communication between multiple master and slave arduinos. In our system, there is one master arduino that collects sensor readings from slave arduinos' and its own environmental sensors. The master arduino also sends the commands to actuators of slave arduinos. Master arduino is connected to Raspberry Pi microcomputer (Sync) through Serial and USB connections, and Raspberry PI communicates with the server through HTTP Post requests. The server also sends the information to user-friendly web interface (Client) through HTTP requests. The connection of all neighboring components of the system is bidierictional which enables the information flow two and from the individual components of the system.

IV. Hardware Architecture

	Hardwares	Quantity
	Hardwares	Quantity
1	Arduino Uno	3
2	Gas Sensor	1
3	Humidity Sensor	1
4	Light Sensor	3
5	PIR Sensor	3
6	LED Stripe	2 meters
7	Servo Motor	4
8	Raspberry Pi3	1
9	BeagleBone Black	1

Fig. 2. The list of system's hardware components and their quantities

The aim of this hardware system is to collect data from rooms' sensors such as light, gas, humidity and PIR (motion) sensors and send it to the sync, execute actuators' commands to rotate the camera, stream the environment, open and close windows and turn on and off LEDs.

A. Arduino Uno

Arduino Uno is a microcontroller that depends on the ATmega328P microcontroller. Arduino contains 6 analog inputs, an USB, 14 digital pins where 6 of them support PWM outputs, and supports 3 type of communications such as SPI, I2C, UART [13]. In this project we utilize 2 digital pins to control lights and servo motors, 2 analog pins for I2C communication (A4 for SDA, A5 for SCL), 4 digital pins from sensors like light, humidity, motion, gas sensor. And also we have a network of 3 arduinos, where they communicate via master-slaves I2C protocol and Serial (UART/USB) communication where master arduino send the collected data to intermediary. In our project every Arduino collects readings from sensors and convert it to JSON format, and send it to master arduino or to Raspberry Pi intermediary.

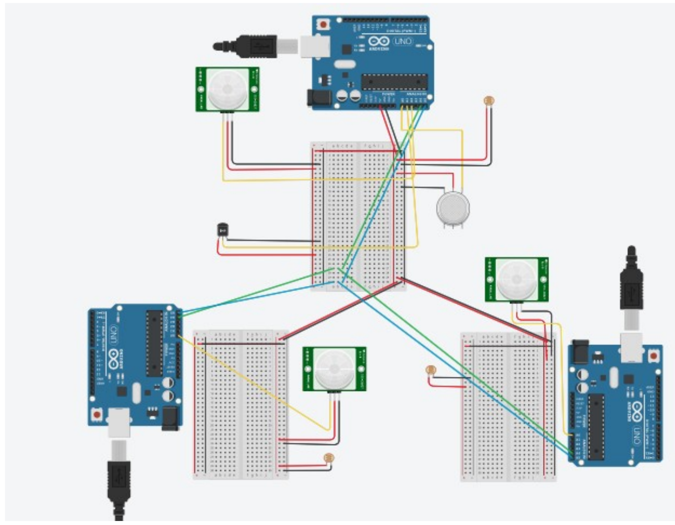

Fig. 3. Arduino Unos connected through I2C communication

B. Raspberry Pi 3

Raspberry Pi 3 is a single board computer. It has Quad Core 1.2GHz Broadcom BCM2837 64bit CPU, 1GB RAM, BCM43438 wireless LAN and Bluetooth Low Energy (BLE) on board, 100 Base Ethernet, 40-pin extended GPIO, 4 USB 2 ports, CSI camera port for connecting a Raspberry Pi camera and upgraded switched Micro USB power source up to 2.5A. In our project Raspberry Pi is used as intermediary to connect with server, and communicate with master Arduino by Serial Port [14]. Furthermore, it responsible for capturing rooms where movements were occurred.

C. Servo Motors

The aim of Servo motors are to control camera, window and also the door [15]. It performs angular operations. Its maximum turn is 188 degree. It is connected to digital pins. And consumes voltage between 4.8V - 6.0V. Also it plays crucial role in controlling camera. For example, in Figure 1 we can see camera, where its scenery is controlled by servo motor. It should turn to the room where the motion was detected.

D. Photoresistor (Light Sensor)

Light sensor is actually a resistor that varies its resistance with an amount of luminosity outside. It connects to analog pin [16]. So, In our project we use photoresistor as an photoswitch for window and lights, when the sunlight outside will decrease it should close windows and turn on lights in the room.

E. MQ-2 Gas Sensor

Gas Sensor (MQ2) is used for gas leakage and smoke detection in areas like house or industry. It detects H2, LPG, CH4, CO, Alcohol, Smoke or Propane. The sensor is very sensitive and in dangerous situations it responds very quick.

F. PIR motion sensor

The PIR motion sensor is an input device which is manufactured by Sunfounder, for the prototype to be used as a motion detector in the house [17]. In order to track any movement at home, the motion sensor was installed in the house, while the sensor wires were connected directly to Arduino Uno. The output of the sensor is a digital output, which is considered as high or low. As motion will be detected, the light turns on in the room and camera faces the room where movement is detected.

G. House prototype

In the projects house prototype the base and walls are made of hard carton. It has three rooms such as living room, kitchen and bedroom. Every room has a windows and doors. As we can see in the Figure 1 kitchen has has 1 motion sensor, 1 gas sensor, 1 humidity sensor and 1 light sensor, in bedroom and living room prototype contains 1 pir sensor and 1 light sensor. 3 windows and camera are controlled by servo motors.

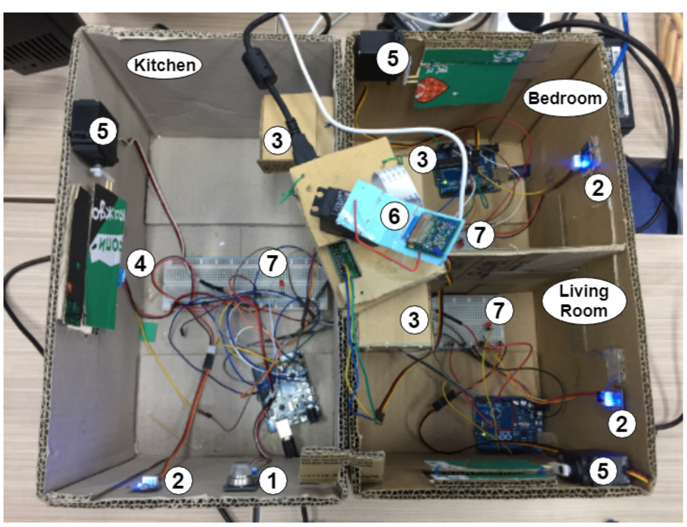

Fig. 4. House prototype

1. Gas sensor
2. Light Sensors
3. PIR (Motion) Sensors
4. Humidity Sensor
5. Servo motors
6. Camera
7. LEDs

H. I2C Protocol (Communication Between Arduinos)

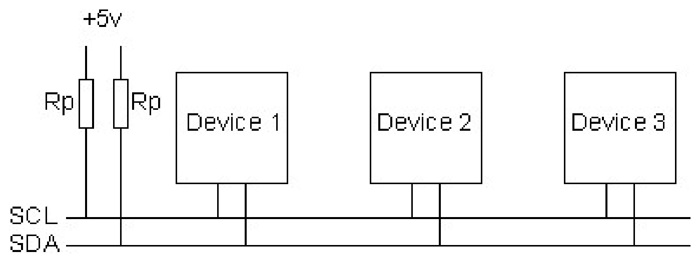

Fig. 5. Inter-Integrated Circuit (I2C) Communication [18]

In the scope of the Senior Project I course, the Communication between multiple arduinos is established by wired Inter-Integrated Circuit (I2C) protocol [18]. I2C uses only two bidirectional open collector or open drain lines, Serial Data Line (SDA) and Serial Clock Line (SCL), pulled up with resistors. Typical voltages used are +5 V or +3.3 V, although systems with other voltages are permitted [19]. There are both 7 and 8-bit versions of I2C addresses. 7 bits identify the device, and the eighth bit determines if it's being written to or read from. This communication is limited in sending or receiving data size. It only allows to send 32 byte data.

Every Arduino has its own role. In this project we followed master-slave pattern of communication. The master arduino collects data from two other slaves. As it is only allowed to send 32 byte of message, the arduino program divides JSON message into multiple chunks less than or equal to 32 bytes. Every complete message has its terminal char * (asterix). Master always make requests to slave nodes. Master counts chunk and terminates after receiving chunk that ends with asterix char. Arduino program uses special library called Wire. The functions OnReceive() and OnRequest() from Wire library enable executing tasks that should only be executed when slave arduino receives the request from master arduino and other tasks that should be executed when the slave sends its sensor readings to master arduino.

I. Arduino and Sync communication

Serial Communication is used for conversation between master and sync. It is implemented by USB connection. The communication is implemented in Python and can be executed on Raspberry Pi. All Arduino boards have at least one serial port (also known as a UART or USART): Serial. It communicates on digital pins 0 (RX) and 1 (TX) as well as with the computer via USB. Thus, pins 0 and 1 could not be used for digital input or output if the arduinos are connected.

J. Arduino Wireless communication

In the scope of the Senior Project II, to improve the design of the architecture from Senior Project I, we established wireless communication between Arduino microcontrollers using nRF24L01 radio modules (Figure 6). Arduino microcontrollers communicate using the common channel. Master Arduino receives the commands by Serial connection from the Sync and sends these signals to the wireless channel. The slave Arduinos retrieve the commands that are pertinent to their identification numbers and execute these commands. Each slave Arduino sends sensor readings to the common channel. The Master Arduino collects these readings, wraps it into the JSON object and sends Master's and Slaves' sensor readings to the Sync using Serial connection.

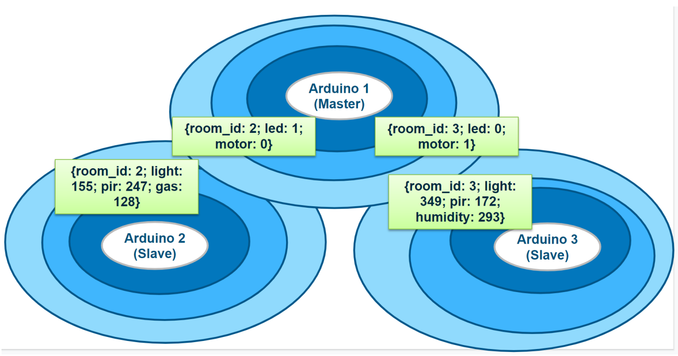

Fig. 6. Arduino wireless connection

V. System Software Design

The software part of the project plays major role in given project about smart house due to projects aim of taking care of critical situations and notifying users. Basically, sensors placed inside house prototype will read data and send it to server for further use. Server tracks the data received from the Sync and communicates with front-end to demonstrate sensor readings simultaneously.

A. Server and Data Storage

For our project we use Java-based framework named Spring boot due to its accessibility, flexibility in easy way of integrating database into system without constructing and particular class and nicely explained documentation with guides that allows to easily implement given framework as the basis for back-end. In the initial stages of the project we implemented basic RESTful server that allowed us to access information about users and make post requests for saving new ones. Basic entities consisted of user, user role, and personal information about users relatives. Since given project emphasized notification system, telegram bot api included as background framework for communicating with user, sending notification and receiving commands from him. This led to the creation of new entity that allowed to save id of the conversation between user and bot into database. In the later stages of the project schedule we began to test different types of sensors and send all the data to server by using Python via post request; thus, new entities related to sensors constructed. In the final phase of the project we added cross-origin request to establish communication between front-end and back-end.
Database is important part of every project with multiple data and for this smart house project Postgresql named database was used. During each registration user related information will be adjusted for simple usage like creating inactive telegram entity and home with no rooms, which will be added later during each sensor reading for particular room.

B. Web application

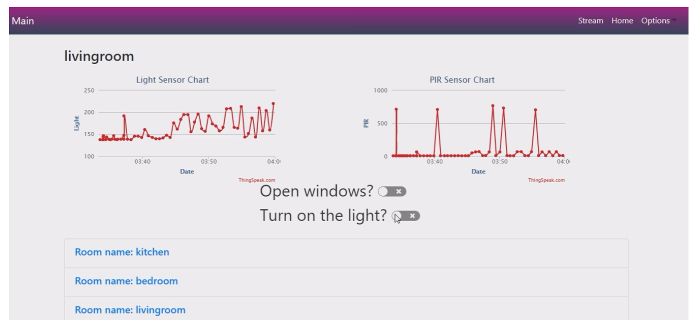

Fig. 7. Web application interface that demonstrates sensor data for particular room

Web application is another major part of the project that allows community to understand more about smart house system. Thus, our team developed user simple friendly web application written on "React JS" that communicates with server. This web interface has basic login, sign up features and is able to demonstrate users home status, reading from sensors in each room, and control hardware parts, servo motor and light tapes (Figure 7).

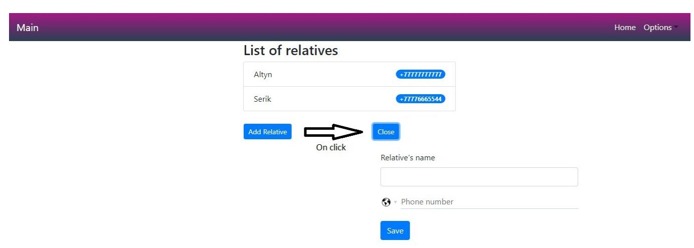

Fig. 8. Web application interface that demonstrates list of relatives that can be notified. By clicking on "Add Relative" button, user will open form for entering relative data and by pressing "Close" button, this form will close.

Another feature of web application is the ability to add relatives into contact list for notification during critical situations (Figure 8). User will be able to open input menu for entering relatives data by pressing "Add Relative" button and if there are no errors with input, data will be stored and then retrieved from database for the purpose of displaying it to user.

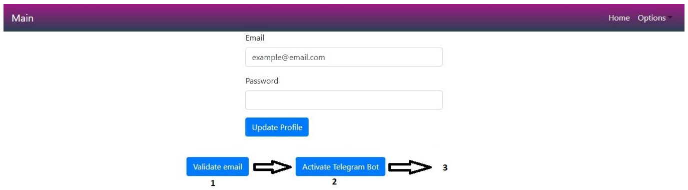

Fig. 9. Web application interface that allows user to update email and password. 1) Button for validating user email. 2) Button for sending message with steps to start communication with system telegram bot. 3) If user validated email and activated his telegram account, there will be no button and space will be empty.

Third feature of our web application is the email and password change. Furthermore, for security measures, which will be described in next parts of this paper, we implemented email validation (Figure 9). By clicking "Validate email" button, user will receive message with validation link. After email validation completed, system provides user with feature of activating his telegram account. By clicking "Activate Telegram Bot" button, user will receive message with the following steps:

1. Find our teams telegram bot named "tripleZ_bot" in "Telegram" application.
2. After finding telegram bot, start conversation with bot.
3. Send message with activation code (each user has unique activation code generated during registration by server).

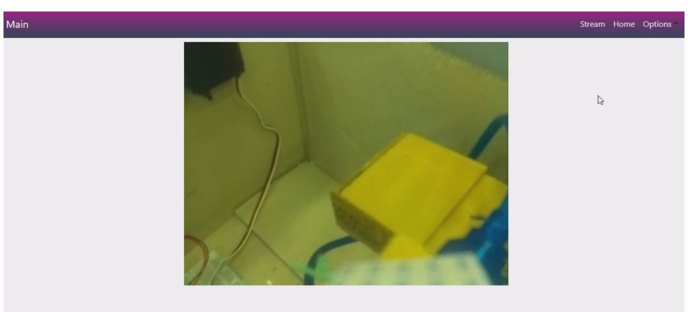

Fig. 10. Web application interface that demonstrates video streaming in room were possible intrusion happened

Final feature of user-friendly web application is live-streaming that simultaneously records and broadcasts to user situation at home in real time. Camera will stream room where PIR sensor detected motion recently.

C. Mobile Application

Another client-based application in this project is mobile application that was developed for remote control purposes only. It means that by downloading this application, user will be able to control lights and windows inside the house using mobile phone. Due to limited functions, mobile application has simple design displayed on Figure 11.

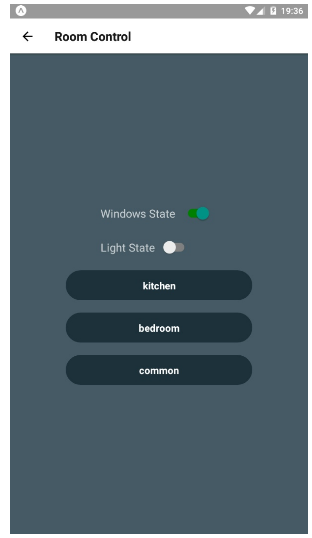

Fig. 11. Remote control page

D. Automatic System Notification

In the beginning of this paper we emphasize automatic notification system as the main core of the smart house project. This system will work as a tool for notifying user during critical situations and works based on the algorithm presented below.

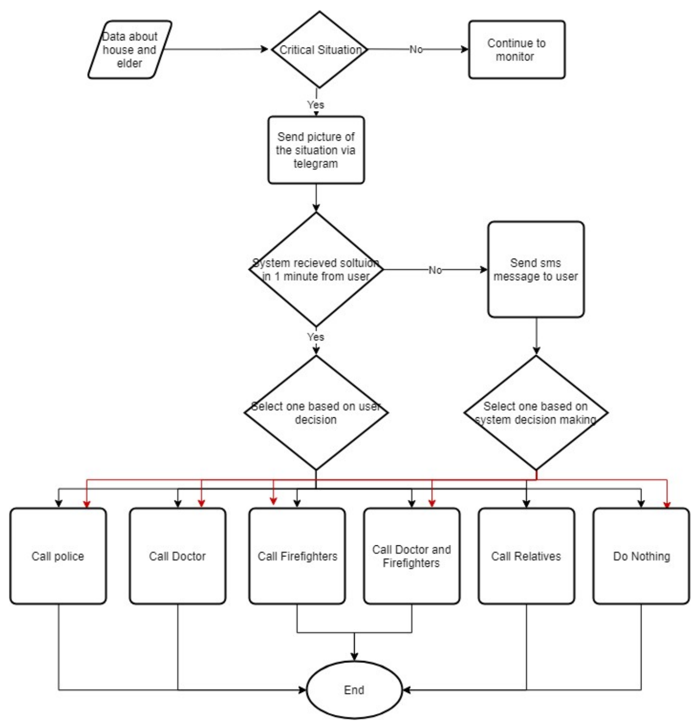

Fig. 12. Flow diagram of the automatic notification system.

Flow diagram in Figure 11 describes basic functionality of automatic notification system. As it was mentioned above, all the data read from sensors will be sent to server, and then server will parse all the received JSON data from sensors into database. These steps continue until the moment sensors detect abnormality where one of the sensors will exceed threshold value unique to every type of sensor. The moment abnormality occurred, server will start automatic user notification with report of situation. After sending notification system act based on the condition specified on Figure 12:

1. If user provides command in one minute after sending notification, system will act accordingly and inform specified association. Finally, system will continue to monitor sensors.
2. Another possible result is that user will be offline or will not notice new message from telegram bot. In this case, system will act based on pre-trained decision making that will be implemented in the future.

E. Security

Major part of every software project is security, especially in cases where some security problem may lead to data loss. First step in implementation of security measures in smart house system is the simple login, sign up. Furthermore, to verify users during authentication, user data must be encrypted, and tokens are perfect tools for user verification. From the start to the end, system will follow steps presented below:

1. If user is registering into the system, after passing simple validation like unique email and username, client-side of the web interface will send given data to server and where user profile data will be saved into database.
2. If user is logging in into the system, both username and password will be sent to server for validation. If provided data passes validation, server will generate token based on username, encrypted unique key known only to server and client. Newly generated token then will be sent to the client as the response. In the end, received token will be stored into browser local storage and will be included in header during request.
3. For every request by authenticated request, server will verify user by retrieving user data from token. If this user exists, server will perform command.
4. Finally, after completing all the actions, and logging out from the web application, client-side will remove user token from local storage. During new login, cycle will start from (b) and continue until user logs out.

Another security measure is applied to telegram. It was mentioned that other than viewing all the data from sensors, user can be notified by system during critical situations. To prevent system from being spammed by random people and reduce pressure from server, every user has to validate his email and activate his telegram account. Method is simple and easy to apply. User will validate his email by following link from the email and then receive activation code for conversation with telegram bot. After sending this code, system will welcome new user and explain how it works.

F. Intrusion detection with face identification

In the second Semester we extended our security system including face identification with PiCamera. Our security system has database where all faces of people living in the house are registered. Furthermore, we implemented Histogram of Oriented Gradients (HOG) based face detection and identification algorithms for finding out whether the person in the house is intruder.

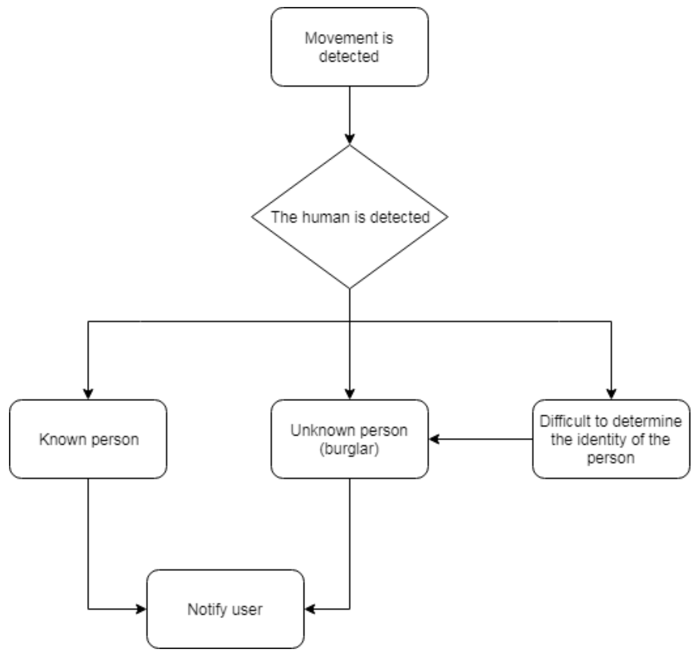

Fig. 13. Flow diagram of the intruder detection system.

In the Figure 13 we can see the flow diagram of the intruder detection. Our system is simple, if the person is not found in database we send notification to the home owner.

VI. Discussion

In this section, we provide an overview of the most important challenges faced when implementing home automation, security, automatic notification and person identification components of the system.

A. Senior Project I

1) Hardware communication: We faced some problems with I2C communication. There are deficiency of sources that explain the communication between multiple Arduinos. Most of them explain only communication between 2 Arduinos, while our project consists of 3 Arduinos placed in each of 3 rooms. There was a problem in plugging in wires, so that they could share same electric environment (have common power source). Hence, by browsing websites dedicated for Arduino, we found that we should plug all VCC wires to one source, and all ground wires to separate source. Another challenge we faced during hardware implementation is incorrect work of Arduino Uno. Initially we thought that our connection was incorrect; however, after failing to run simple LED blinking program on this board, we understood that it was broken. After replacing broken Arduino with the working one, we found another problem related to power sharing and with the help of students from robotics major we understood that each Arduino board must have power source from separate nodes.

2) Web application: Interaction with the system is one of the key features of the web application. However, due to the limited knowledge in the area of web developing, it took more time to create user friendly web application. Furthermore, there were a lot of changes on both client and server side of the project.

One of the key reasons for significant changes on server side of the project is often changeable database tables due to the addition of new rows into the existing tables, or creation of new entities. Furthermore, it took some time to integrate several programming languages into project environment. Major challenge related to the optimization of the system is sending message via email. It takes several seconds to send email to one user and if several people connected to the system at the same time with the same aim of validation email, this process may crash. Since this problem is not fixed, sending message with email has to be optimized.

Client side of the project is another section of software part of the project and has higher number of challenges during period of project implementation than server side. The first reason is that client side was implement with ReactJS and during basic project installation for testing purpose, NodeJS cross-platform downloaded incorrect packages. As a result, for every run, client threw error of incorrect package version. To solve, this problem, we had to download basic React script from open source websites. Another challenge is the ReactJS version update. After implementing basic login, sign up security measures with token validation, we found out that it is better to update ReactJS version from 15.x.x to 16.x.x. However, the main issue is that they are much different from each other. Many bootstrap styles had to be re-implemented and new packages had to be downloaded.

3) Hardware components: Before connecting Raspberry Pi with master-slave-slave cluster of Arduinos, we had to update Raspberry Pi operating system because the last update was 2 years ago and undo previously manually configured network settings such proxy server and network interfaces. After updating the system, there was no available memory in the system and memory on the operating system was not correctly partitioned. We solved this problem by formatting the SD card and installing new operating system Raspbian.

Currently we are still waiting when healthcare components of the system will be delivered to start integrating them into the system to display the readings on the web interface and notify the user about any health problems of elderly residents of the house.

B. Senior Project II

1) Wireless communication: There were some problems with wireless communication throughout the semester. The first challenge was to connect three Arduino microcontrollers wirelessly because the protocols of the available libraries discuss the use of Radiohead libraries for two Arduino microcontrollers with one Master and one Slave Arduinos. After establishing the wireless communication between three microcontrollers, we had to figure out how to reliably extract the actuator commands or environmental sensor readings from the common channel ensuring that the data is not corrupted.

2) Person identification: We tested several algorithms before selecting the most optimal one for the intruder detection system. Initially, we tried to use Convolutional Neural Network (CNN) deep learning algorithm on the Raspberry Pi. However, CNN on the Raspberry Pi was inefficient because of the computational limitations of the Raspberry Pi that took much time to process the video stream. We found out that the HOG algorithm on the server provided much better results and decided to integrate it into our system.

3) Server and Client: In Senior Project I, we built web application for interaction with the user. In Senior Project II, we decided to extend client-side of the project by building the mobile application. It was developed on Android Studio, which was later rewritten on react-native due to the experience with ReactJS. Furthermore, react-native is cross-platform mobile application development framework that can be used on both Android and IOS devices.

Along with the development of mobile application, we deployed our server on https://heroku.com website and have access to it with running it in IDE.

4) Hardware Components: The Radiohead library that we used had clashes with the servo motor library. Both Radiohead and servo motor libraries shared the same function with different timers. We tried to solve this problem ourselves by searching for the alternative functions and manipulating the timer values. However, we found that the Radiohead library provides the solution to this problem in its documentation. We followed the instructions and solved eliminated this clashes in timers.

VII. Conclusion

In this project, we implemented smart house system with home automation and security components. The smart healthcare part of the project will be implemented in Senior Project II course due to the absence of healthcare sensor devices. In this first part of the course we implemented environmental sensing, automatic notification system through Telegram Bot and user-friendly web interface. The user can interact with the system through web-interface that provides the user the readings of environmental sensors. The system also automatically notifies the user about potential intrusions and gas or water leak.

REFERENCES

[1] S. Amendola et al., “RFID Technology for IoT-Based Personal Healthcare in Smart Spaces“, *IEEE Internet of Things Journal*, vol. 1, no. 2, April 2014, pp. 144-152. [Online]. Available: [doi: 10.1109/JIOT.2014.2313981].

[2] R. Harper, “Inside the Smart Home: Ideas, Possibilities and Methods“, *Inside the Smart Home*, pp. 1-13. [Online]. Available: [doi: 10.1007/1-85233-854-7_1].

[3] N. David et al., “Design of a Home Automation System Using Arduino“, *International Journal of Scientific and Engineering Research*, vol. 6, no. 6, June 2015. [Online]. Available: https://www.researchgate.net/publication/279179486_Design_of_a_Home_Automation_System_Using_Arduino.

[4] R. K. Kodali et al., “IoT based smart security and home automation system“ in *Conf. 2016 International Conference on Computing, Communication and Automation (ICCCA)*. Noida: ICCCA, April 2016. [Online]. Available: [doi: 10.1109/CCAA.2016.7813916].

[5] M. Shariq Suhail et al., “Multi-functional secured smart home“ in *Conf. 2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*. Jaipur: IEEE, Nov. 2016. [Online]. Available: [doi: 10.1109/ICACCI.2016.7732455].

[6] F. Schroff, D. Kalenichenko and J. Philbin, ”FaceNet: A unified embedding for face recognition and clustering,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Boston, MA: IEEE, 2015, pp. 815-823. [Online]. Available: [doi: 10.1109/CVPR.2015.7298682].

[7] Z. Qiang, M. Yeh, K. Cheng, and S. Avidan, ”Fast human detection using a cascade of histograms of oriented gradients.” in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’06)*, 2006, vol. 2, pp. 1491-1498. IEEE. [Online].

[8] D. Navneet, B. Triggs, ”Histograms of oriented gradients for human detection.” in *International Conference on computer vision Pattern Recognition (CVPR’05)*, 2005, vol. 1, pp. 886-893. IEEE Computer Society.

[9] M. Asadullah and A. Raza, “An overview of home automation systems“ in *Conf. Robotics and Artificial Intelligence (ICRAI)*. Rawalpindi: ICRAI, Dec. 2016. [Online]. Available: [doi: 10.1109/ICRAI.2016.7791223].

[10] M. Sahani et al., “Web-Based Online Embedded Door Access Control and Home Security System Based on Face Recognition“ in *Conf. 2015 International Conference on Circuit, Power and Computing Technologies [ICCPCT]*. Cebu: ICCPCT, 2015.

[11] S. Baker, W. Xiang and I. M. Atkinson, “Internet of Things for Smart Healthcare: Technologies, Challenges, and Opportunities“, *IEEE Access*, pp. 1-1. [Online]. Available: [doi: 10.1109/ACCESS.2017.2775180].

[12] A. Odunmbaku, “Elderly Monitoring System with Sleep and Fall Detector“ in *Conf. International Conference on IoT Technologies for HealthCare*. Rome: Springer LNICST, Jan. 2015.

[13] Arduino, *ARDUINO UNO REV3*. [Online]. Available: https://store.arduino.cc/usa/arduino-uno-rev3.

[14] Raspberry Pi, Raspberry Pi 3 Model B. [Online]. Available: https://www.raspberrypi.org/products/raspberry-pi-3-model-b/.

[15] Servo Motor, *Servo Motor SG-90*. [Online]. Available: https://components101.com/servo-motor-basics-pinout-datasheet.

[16] SunFounder, *Photoresistor Sensor*. [Online]. Available: https://www.sunfounder.com/learn/sensor_kit_v1_for_arduino/lesson-21-photoresistor-sensor-sensor-kit-v1-for-arduino.html.

[17] SunFounder, *Hardware Description*. [Online]. Available: https://www.sunfounder.com/learn/Smart-Home-Kit-V2-0-for-Arduino/hardware-description-smart-home-kit-v2-0-for-arduino.html.

[18] Arduino, *Wire Library*. [Online]. Available: https://www.arduino.cc/en/Reference/Wire.

[19] SparkFun, *I2C at the Hardware Level*. [Online]. Available: https://learn.sparkfun.com/tutorials/i2c/i2c-at-the-hardware-level.

[20] UCI, *MHEALTH Dataset Data Set* . [Online]. Available: https://archive.ics.uci.edu/ml/datasets/MHEALTH+Dataset.

[21] S. Yao, “DeepSense: A Unified Deep Learning Framework for Time-Series Mobile Sensing Data Processing“ in *Conf. 26th International Conference on World Wide Web*. Perth: WWW, April 2017, pp. 351-360.

[22] M. Novk, F. Jakab, and L. Lain, “Anomaly Detection in User Daily Patterns in Smart-Home Environment“, *Cyber Journals: Multidisciplinary Journals in Science and Technology, Journal of Selected Areas in Health Informatics (JSHI)*. vol. 3, no. 6, June 2013.