

Research Article

An Intelligent, Secure, and Smart Home Automation System

Rizwan Majeed,¹ Nurul Azma Abdullah,¹ Imran Ashraf,² Yousaf Bin Zikria,² Muhammad Faheem Mushtaq^{1,3} ,^{1,3} and Muhammad Umer⁴

¹Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia (UTHM), Batu Pahat, Parit Raja 86400, Johor, Malaysia

²Department of Information and Communication Engineering, Yeungnam University, Gyeongsan 38541, Republic of Korea

³Department of Information Technology, Khwaja Fareed University of Engineering and Information Technology, Rahim Yar Khan 64200, Pakistan

⁴Department of Computer Science, Khwaja Fareed University of Engineering and Information Technology, Rahim Yar Khan 64200, Pakistan

Correspondence should be addressed to Muhammad Faheem Mushtaq; faheem.mushtaq@kfueit.edu.pk

Received 18 February 2020; Revised 26 July 2020; Accepted 7 October 2020; Published 29 October 2020

Academic Editor: Chenxi Huang

Copyright © 2020 Rizwan Majeed et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The idea of a smart home is getting attention for the last few years. The key challenges in a smart home are intelligent decision making, secure identification, and authentication of the IoT devices, continuous connectivity, data security, and privacy issues. The existing systems are targeting one or two of these issues whereas a smart home automation system that is not only secure but also has intelligent decision making and analytical abilities is the need of time. In this paper, we present a novel idea of a smart home that uses a machine learning algorithm (Support Vector Machine) for intelligent decision making and also uses blockchain technology to ensure identification and authentication of the IoT devices. Emerging blockchain technology plays a vital role by providing a reliable, secure, and decentralized mechanism for identification and authentication of the IoT devices used in the proposed home automation system. Moreover, the SVM classifier is applied to classify the status of devices used in the proposed smart home automation system into one of the two categories, i.e., “ON” and “OFF.” This system is based on Raspberry Pi, 5 V relay circuit, and some sensors. A mobile application is developed using the Android platform. Raspberry Pi acting as the server maintains the database of each appliance. The HTTP web interface and apache server are used for communication among the Android app and Raspberry Pi. The proposed idea is tested in the lab and real life to validate its effectiveness and usefulness. It is also ensured that the hardware and technology used in the proposed idea are cheap, easily available, and replicable. The experimental results highlight its significance and validate the proof of the concept.

1. Introduction

In this age of technology, the Internet of Things refers to the idiosyncratically identifiable objects and their virtual representation in cyberspace [1–4]. IoT is a pristine information processing and accretion method encompassing RFID [5] sensor machinery, smart technology, nanotechnology, and other technological advancements. IoT is not alone noble technology; instead, it circumvents considerable complementary technical development and provides capabilities that are appropriated together and aid to bridge the gap between the virtual and physical world [6]. Everyone is living a very busy life, and they

want to get ease in every aspect of life. The Internet of Things (IoT) is a very vast field, and this research cannot cover up the whole thing to automate. But, the ease of people Home Automation is the first domain that comes in mind. Home Automation [7–10] is a term used to describe the working of all home appliances together, and we are controlling things by using an Android smartphone [11, 12], tablet, or computer having internet connectivity. The popularity of home automation is increased rapidly during the recent years. People like the comfort of maintaining and changing the status of appliances from any part of the world using remote access. It is eventually becoming the need of every person.

IoT provides more flexible and low-cost solutions for daily life problems which ultimately improve the user's life [13]. Although many previous researchers proposed many home automation systems by using different sensor combinations [14–18], as shown in Table 1, with the detailed description of previous studies provided, we may conclude that there are some reasons for the motivation to design an efficient home automation system.

- (i) Previously proposed home automation systems are expensive and complex
- (ii) Previously proposed Bluetooth home automation systems required intrusive installation
- (iii) Previously proposed home automation systems use the connectivity of the Internet which may be not useable in those areas where the internet is not available
- (iv) Previously proposed home automation systems have not focused on security and safety problems that may arise from their implementation
- (v) The previously proposed home automation system has no intelligent decision-making ability

To assure home automation continues in a normal fashion, it is necessary to check all important parameters that may complicate the system, one of which is no GUI environment for the users. Users cannot understand the working of the system due to the lack of a GUI environment. Secondly, there is no device restoration which is harmful to home appliances. Thirdly, there is no prediction of the electricity bill for the users, and all the previous home automation solutions are very expensive. So, we provide an effective solution to overcome all these problems by undertaking the following objectives.

- (i) The proposed solution is low cost because it did not use IP-based devices such as bulb and light.
- (ii) The proposed solution provides electricity bill prediction to the users. The proposed solution uses GSM technology which means a Global System for Mobile Communication (GSM) modem to control home appliances such as light, conditional system, and security system via Short Message Service (SMS) text messages.
- (iii) The proposed solution provides device restoration which means when you restore a computer or other electronic device, you return it to a previous state.
- (iv) The proposed solution is a cloud-based solution which means it controls home appliances through the Internet.
- (v) The proposed solution provides Arduino and Raspberry implementation. For those who love to tinker with electronics, the Raspberry Pi and Arduino have become invaluable tools. They are both extremely popular options but are also very budget-friendly. Arduino is best used for real-time applications of hardware and software, and Raspberry Pi easily connects to the Internet.

- (vi) The proposed solution provides a prototype home design for the ease of the users.
- (vii) The proposed solution has an intelligent decision-making ability to classify the status of IoT devices used in the smart home automation system.
- (viii) The proposed solution ensures secure identification and authentication of users using blockchain technology.

The rest of the paper is structured as follows: Section 2 highlights the related work and key contributions in the area of smart home and smart buildings. Section 3 discusses the used approach, and Section 4 presents the implementation details of the used approach along with used hardware and software. Section 5 presents the results of the experiments along with the discussion. The paper is concluded in Section 6.

2. Related Work

The research gap identified in previous studies in the domain of smart homes and smart buildings is reported in Table 1.

Table 1 clearly mentions the research gap in the existing studies in the domain of smart homes, smart buildings, and smart environments. A few of the open challenges in smart environments are unsecured authentication of users and the device registration system that requires keeping track of signals between multiple transmitters. A few other systems are facing issues such as more time delay in turn on/off of an appliance, high cost, only showing temperature, lack of secured communication among connected devices, and lack of security and user authentication. Similarly, a few other issues are limited control of light and device on/off and limited GUI support for Symbian OS cell phones; limited range (50–100 m).

3. Used Approach

In this paper, a smart home automation system is presented that allows the user to check and change the status of electronic home appliances and the working state of sensors. Appliances include common lighting, heating, ventilation, and air-conditioning electronic devices. This product not only saves time but also provides energy-efficient solutions which let you know which device of your home consumes more power. This cost-efficient and the energy-saving system can also be installed in hotels and industrial/domestic places. Easy GUI icon-based interface and notifications of application let you be in touch with your home even if you are on a trip far away from your home. The concept of automating ordinary available installed devices of the home costs much less than special IP-based (RJ-45) [27] devices giving uniqueness and providing a solution that is affordable to people of middle-class families. The working of the proposed home automation system is shown in Figure 1.

In the admin side of the app, the user will be able to draw the complete structure of his/her home using easy drag and drop interface. Firstly, the user adds floors of the house and, then, selects any added floor to add rooms inside it. After

TABLE 1: Previous studies and the research gap.

Work year	Sensor/technology used	Purpose	Limitation
[19] 2020	ArduinoTmega2560, IoT	Assist handicapped people to control home appliances	Unsecured authentication of users
[20] 2019	Elegoo Mega 2560, Web Server	Intimate owner about the door opening event in the home or office	A registration system requires keeping track of signals between multiple transmitters.
[21] 2018	Arduino UNO, ESP-8266, local Wi-Fi	Remotely control electrical and electronic devices in the home	More time delay in turn on/off of an appliance
[22] 2017	Wireless sensor network	Provide novel secured and energy efficient home automation	Expensive; only shows temperature
[23] 2016	Electrical switch, Internet	Regular monitoring of home appliances	Lack of secured communication among connected devices
[24] 2015	GPRS, speech recognition	Remote operating of electrical devices in home	Lack of security and user authentication
[25] 2013	Android ADK	Support handicapped and old-age people to control home appliances	Limited control of light and device on/off
[26] 2011	Bluetooth	Cell phone-based control of home appliances	Limited GUI support for Symbian OS cell phones; limited range (50–100 m)
[14] 2009	ZigBee microcontroller	Connectivity of devices within the home	Short range; low data speed
[15] 2002	Bluetooth	Control home appliances	Intrusiveness of the installation
[16] 2004	Personal computer (PC) based web server (Java-based technology)	Remotely connectivity of devices	Intrusive and expensive wired installation
[17] 1998	Phone-based controller	Remotely control home appliances	No graphical user interface
[18] 1993	Novel control network (hand gesture technology)	Remote control of objects using free-hand gestures	Inaccuracy of hand gestures; user fatigue if repetitive hand gestures are required

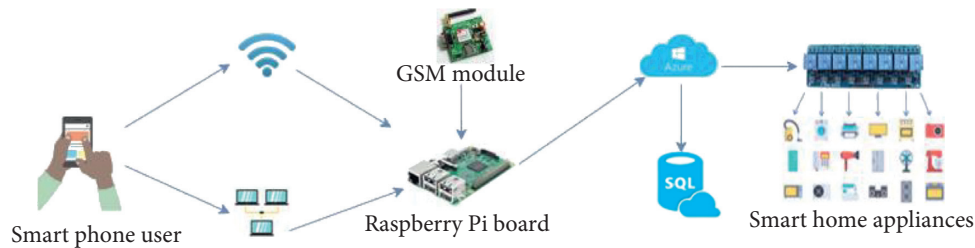


FIGURE 1: Complete working structure of home automation.

selecting the room, the user will be able to add appliances and can set the position of each device according to the real structure of a home. There is also an option available to add custom floors, rooms, and devices inside the home structure. After completing the setup of home from the admin application user sync database in the form of JSON [28] to our server, when a user logs in using credentials, he will be able to see the complete structure of home the same as he made in the admin side as complete data are fetched from the server and get updated using VOLLEY [29, 30] service after 30 sec. At the main screen of the application, the user can see the number of total devices installed at home, and the floor details are shown. Three more tabs lie on the lower bottom of the application. The 2nd tab contains the complete information of sensor status. The third tab contains the history of each device status along with the name of the user who changes status and on which time. The fourth tab is logout from the application. Shared preferences are managed for credentials, and the information is saved.

Moreover, the SVM classifier is applied to classify the status of home appliances used in the smart home automation system. It classifies the status of home appliances into two categories, i.e., “ON” and “OFF” according to the use of these appliances. We have also compared the used prediction model with other models and choose the best one. Furthermore, blockchain technology is used to secure the whole communication and authentication between IoT devices and users that request to make “ON” or “OFF” any home appliances. Blockchain technology also ensures the security of data transfer between IoT devices, servers, applications, and users of the smart home automation system.

3.1. SVM-Based Classification. The proposed smart home automation system has ability to make an intelligent decision regarding the status of home appliances. To accomplish it, we used a linear SVM classifier. SVM was initially proposed by Vapnik [31]. SVM is a supervised machine learning

algorithm that can be used for both classification and regression challenges [32]. In the proposed system, we have employed it for classification purposes. Moreover, many techniques can be used for the classification of data such as decision tree, neural network, and K-nearest neighbor. We have preferred to use SVM for intelligent decision making in the proposed solution due to the following reasons.

- (i) SVM is a suitable choice where binary output classes are required as in our case. Our proposed system needs to classify the status of devices into one of two classes, i.e., “ON” and “OFF.”
- (ii) It is required to find the best hyperplane using a kernel function to classify the data point of one class from another class in classification problems. SVM finds the optimal hyperplane by separating the data points of both classes.
- (iii) When classification was performed using the SVM classifier, the resultant hyperplane has maximum margin between data points of target classes. However, the hyperplane generated by other classifiers does not achieve maximum margin [33].

To mathematically describe the SVM hyperplane and max hyperplane, we used equations (1) and (2) given by [34].

$$\left(\begin{matrix} \leftarrow \\ x_1 \end{matrix}, y_1 \right) \cdots \left(\begin{matrix} \leftarrow \\ x_n \end{matrix}, y_n \right), \quad (1)$$

where y has the value of either -1 or $+1$ and to represent whether the data points \leftarrow belong to which class, the value of y is used. Equation (2) is used to obtain the maximum margin hyperplane used by SVM.

$$\text{Max Hyperplane} = \frac{\sum_{i=1}^n x_i \text{ where } y_i = 1}{\sum_{j=1}^n x_j \text{ where } y_j = -1}. \quad (2)$$

In our problem, the input dataset contains the readings of sensors that include temperature, smoke, and light sensor. The output of the problem is status classes, i.e., “ON” or “OFF.” The value of $y = 0$ represents that the device is “OFF,” and $y = 1$ represents that the device is “ON.”

SVM is a classic two-type classifier for segregating the two classes by determining an optimal hyperplane in a two-dimensional plane, as shown in Figure 2. The green color dots represent class 1 and the red color dots represent class 2, the optimal separating line denotes by H , the H_1 line is parallel to H and linking samples in class 1, and the H_2 line is also parallel to H and linking samples in class 2, as shown in Figure 2. w denotes that the distance between the H_1 and H_2 is called the margin, i.e., equal to $2/\|w\|$.

It is needed to separate the two classes in such a way that misclassification error becomes zero and maximize the margin. This can be achieved by using optimal separating line H that becomes the optimal separating hyperplane through spreading two-dimensional space into higher dimensional space. The equation of a separating hyperplane to separate the two classes using the optimal separating hyperplane is given as follows:

$$wx + b = 0. \quad (3)$$

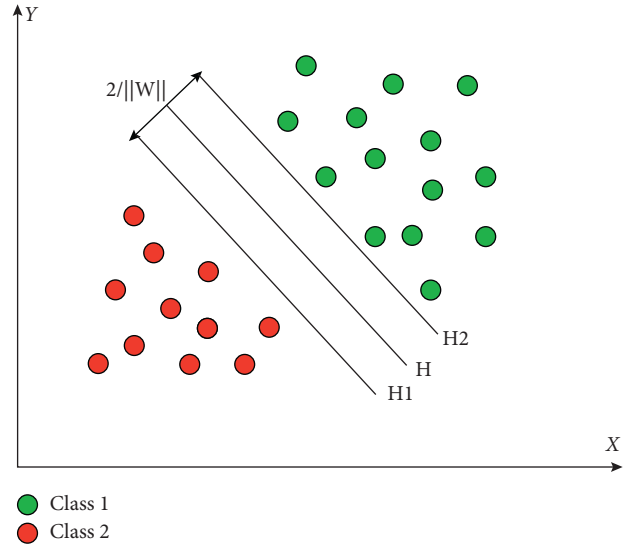


FIGURE 2: Schematic diagram of the hyperplane.

In equation (3), w is the weight vector and the offset of the separating hyperplane is b . A linearly separable hyperplane between class 1 and class 2 is shown in Figure 3.

3.2. Blockchain for Security of Smart Home. A key objective of the smart home automation system is to provide a trusted, secure identification and authentication of IoT devices. To ensure these objectives, we have used blockchain technology. Blockchain technology was introduced by Nakamoto in 2008 [35]. The main features of blockchain technology are security, anonymity, and decentralization [36]. These features can be fruitful to IoT by providing more security and reduced dependency on the central server. Moreover, blockchain technology also ensures tempered resistance data structure by utilizing data encryption and timestamp. In the proposed approach, the implementation of the blockchain module is accomplished in Java by defining the block's contents in a hash that is a unique identifier. Using each block, a block hash is computed, and a hash SHA-256 is, then, computed from it. On achieving a threshold, requests for connectivity are granted through managing blockchain, and then, a block is created. To validate the complete blockchain, a block of the chain is looped-over that verify a current block's hash to its corresponding previous block's hash.

Whenever a connectivity request is generated by a user, it will be authenticated using a sequence of steps described in the flowchart given in Figure 3.

We have also designed an algorithm to explain the working of blockchain technology in the proposed approach given as follows. The whole process of blockchain implementation is also depicted in a flowchart shown in Figure 3 and Algorithm 1.

First of all, a block is created using a block class implemented in Java calculating a hash using a previous hash, timestamp, and data string. After creating a block, a hash will be generated using the SHA256 algorithm

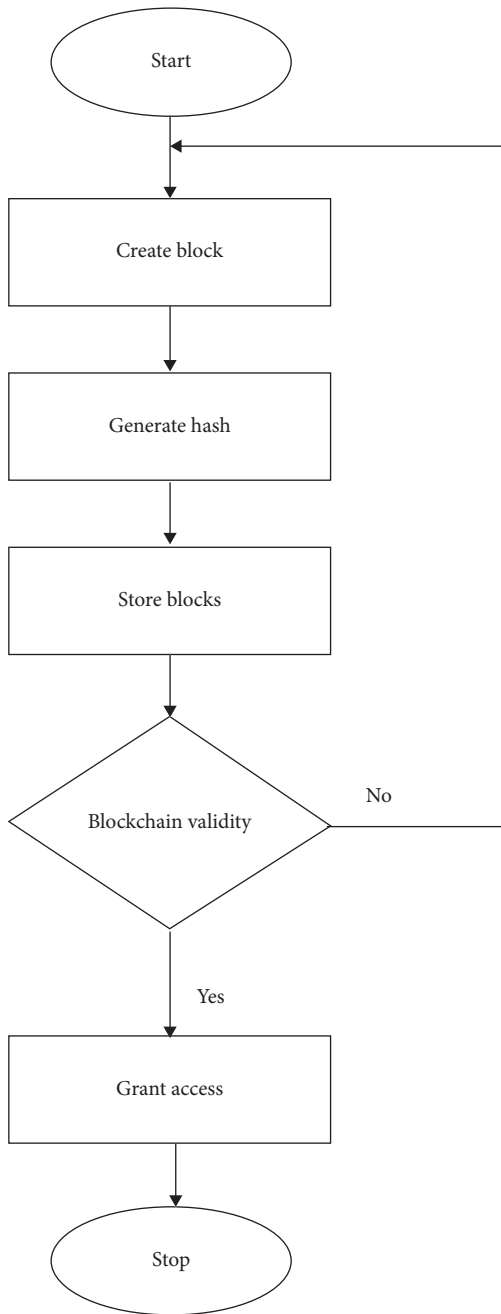


FIGURE 3: Flowchart of blockchain implementation in the proposed approach.

implemented in Java. When a hash is created, the next step is to store the generated blocks. At the final step, validation of the blockchain will be performed to check whether the hash is equal to the calculated hash. If it is true, then the block is valid and access is granted; otherwise, the whole process is repeated.

4. Implementation Details

To understand the working of the project, Figure 1 explains the complete working structure and integration of different devices with each other. The arrows show the flow of projects

starting from user smartphones to changing the state of electronic devices. There are two network modes in which the user can interact with our Raspberry [37–41]. If the user is sitting inside the home (home local network), then the user will be able to use all of the IoT services at the local network without connecting to the internet cloud. This will also result in faster communication of devices with the user as everything is happening locally.

The second network mode is used if the user is residing outside the home anywhere in the world. Then, the user first connects to the internet. The processed request is sent to the Microsoft Azure Cloud [42, 43]. Based on credentials provided by the user matched with the Azure cloud database, the user request is sent to the respective Raspberry for processing. The account of each user is maintained individually at the Microsoft Azure Cloud Database. Services of each user are handled based on credentials from which the request is being generated. APIs are called from the cloud if the user is outside the home network. Then, the same APIs are also residing inside the Raspberry Pi server if the user is in the home network.

Data sharing between application and server database is performed in the form of JSON. APIs are secured using multiple hashing techniques. The change in the state of any device is performed using Raspberry Pi GPIO [44] pins. Raspberry Pi receives the request from the server. According to the request from the user, Raspberry Pi responds to devices. The database of each request generated by the individual user is maintained at cloud servers. The user can check the complete history of processed requests on his smartphone by setting the duration. Sensors installed inside the home update its state continuously after 30 sec and respond to the change to the Raspberry Pi server. In response, the Raspberry Pi server syncs all data to the cloud database, and the values of the mobile application are updated.

4.1. Hardware Components. Many electronic components, as well as sensors, are used in this project. Our working structure diagram Figure 1 shows a few of the components used in this project. A complete description of components along with description is shown in Table 2.

4.1.1. Raspberry Pi 2B. The Raspberry Pi is a series of small single-board computers to promote the teaching of basic computer science in schools and developing countries. It contains a 900 MHz quad-core ARM Cortex-A7 CPU processor along with 1 GB of RAM. It supports 100 MBPS Ethernet. It also contains 4 USB ports along with 40 GPIO pins, full HDMI support with camera, and card interface and supports a micro SD card. It contains a 3.5 mm audio jack and composite video support too.

4.1.2. 5 V Relay Circuit. A relay is an electrically operated device. It has a control system and (also called input circuit or input contactor) and a controlled system (also called output circuit or output contactor). It is frequently used in

- (1) At the first step, a block is created with the help of the block class
- (2) A hash is created using the SHA-256 algorithm on the successful creation of an already created block otherwise repeat step 1
- (3) After the generation of hash, the next step is to store the blocks
- (4) At the last step, the validity of the blockchain will be accomplished. If the block is valid, then request for connectivity is granted
- (5) If the block is not valid, steps 1 to 4 are repeated

ALGORITHM 1: Working algorithm of blockchain technology.

TABLE 2: Electronic components with specification details used in the project.

Components	Specifications
Raspberry Pi 2B	40 GPIO pins, 1 GB RAM A 900 MHz quad-core ARM Cortex-A7 CPU, operational voltage 7–12 V
Relay circuit pack	The 5 V operational 8-relays circuit pack
L293D motor control shield [45]	Supply-voltage range: 4.5–36 V; output current: 600 mA/channel
Smartphone mobile	Android supported
DS18B20 temperature sensor [46]	Temperature range: –55 to 125°C (–67°F to +257°F)
LM393 LDR sensor [47]	Digital switching outputs (0 and 1), external 3.3 V–5 V vcc
MQ2 smoke sensor [48]	Combustible gas, smoke

the automatic control circuit. To put it simply, it is an automatic switch to controlling a high-current circuit with a low-current signal.

4.1.3. L293D Motor Shield. L293D [49] is a monolithic integrated, high-voltage, high-current, 4-channel driver. This means that using this chip, you can use DC motors and power supplies of up to 16 Volts, that is, some pretty big motors, and the chip can supply a maximum current of 600 mA per channel; the L293D chip is also what is known as a type of H-Bridge. The H-Bridge is typically an electrical circuit that enables a voltage to be applied across a load in either direction to an output, e.g., motor.

4.1.4. DS18B20 Temperature Sensor. The DS18B20 [50] temperature sensor is a 1-wire digital temperature sensor. This means that you can read the temperature with a very simple circuit setup. It communicates on a common bus, which means that you can connect several devices and read their values using just one GPIO pin of the Raspberry Pi.

4.1.5. LM393 LDR Sensor. The LM393 [51] is a simple photoresistor light sensor that has both analog and digital outputs. The digital output has a trim potentiometer that can be used to set a trigger light level. In the following example, we will use the analog output to measure the light level. If you are interested in reading the digital output, please refer to the pulse sensor example sketch.

4.1.6. MQ2 Smoke Sensor. MQ2 [52] is one of the commonly used gas sensors in the MQ2 sensor series. It is a Metal Oxide Semiconductor- (MOS-) type gas sensor also known as chemiresistor as the detection is based upon the change of resistance of the sensing material when the Gas comes in contact with the material. Using a simple voltage divider network, concentrations of the gas can be detected. An MQ2

Gas sensor works on 5 V DC and draws around 800 mW. It can detect LPG, smoke, alcohol, propane, hydrogen, methane, and carbon monoxide concentrations anywhere from 200 to 10000 ppm.

4.1.7. GSM Module. This module can be mostly used for developing IoT (Internet of Things) and embedded applications. It is a dual-mode module that works on frequencies 900 MHz and 1800 MHz. It has a feature of multislot class, i.e., class 10/class 8. The pins RXD and TXD are used to receive and transmit data. It works on a voltage of 3.4 V to 4.5 V, and any higher voltage may damage the module. The GSM module also requires low power consumption.

4.2. Software Components. For the development of the mobile application, there are many platforms available such as Symbian, Android, IOS, and Windows Mobile.

In our research paper, the Android platform is used for the development of the whole project. The main reason for the use of the Android platform is that it is the most widely used platform all over the world. The Android operating system is supported by almost every brand device. Java language with an Android software development kit (SDK) has been used for the development and implementation of home automation application.

4.2.1. Android Studio. Android Studio [53, 54] is used for the development of Android apk as it supports all development tools such as debuggers, libraries, and a handset emulator. The Volley library is used for all sensor services. Material design library is used to make the application more interactive.

4.2.2. Server-Side Scripting. For the development of the server-side on cloud and inside Raspberry Pi, LAMP (Linux, Apache, MySQL, PHP) is used to provide all backend functionality.

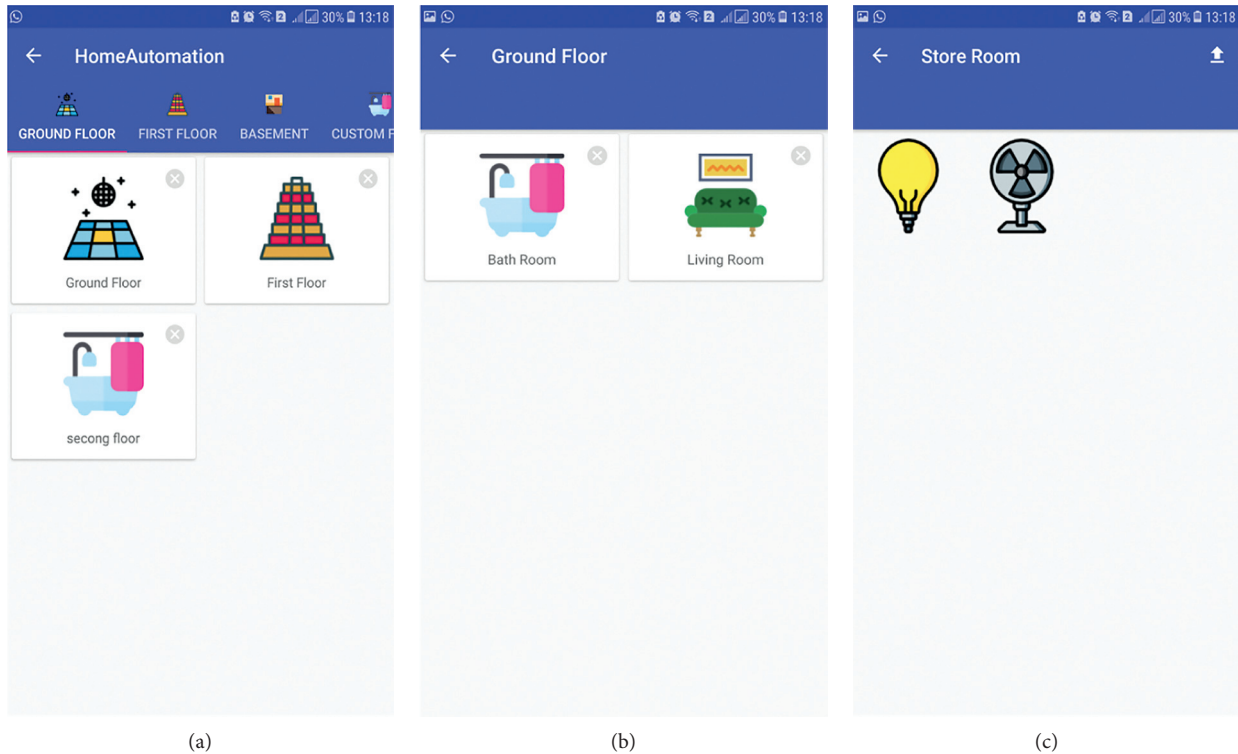


FIGURE 4: Working flow of each appliance.

4.3. Mobile Applications for Smart Home. Mobile Application consists of two modes of operation. In the admin mode, the user able will be able to draw his/her complete home prototype using an easy drag and drop interface. Each device is assigned with a Raspberry Pi pin number which controls the actual working of electronic devices on the backend.

If we look at the user side of the mobile application from which the user will not only be able to see the complete prototype of the home which is designed but also can control the working of each appliance on basis of pin configuration that he did on the admin side, as shown in Figure 4. The main screen contains the number of devices installed in the house along with floor names and the number of rooms inside it. Tab layout is used for switching between app services.

A nice and interactive GUI-based interface with attractive icons lets the user know about its working easily, as shown in Figure 5. Icons changing are used to let the user know about the current state of an electronic device with a touch active button to change state. There is also an intensity bar to let the user maintain its fan speed and light brightness according to its choice. The status of devices is shown with the active state of working and active ago with the passive state.

In the second tab of the main screen, the state of each sensor is shown along with current values. The values of sensors get updated using backend services which refresh themselves after every 30 seconds, as shown in Figure 6. Currently, we are dealing with the live status of two sensors such as temperature and light, as shown in Figure 7. The light sensor value 0.0 shows that currently, daylight is active and the sensor is in off state.

The third tab on the main screen gives us complete details about the history of the device. The complete log with the details of the user who changes the status of appliances is maintained with time-stamped details. The user can check the on and off duration of appliances from history as well. Another interesting feature of this application is that the user will receive a notification on his smartphone if any device is working for more than two hours. It is similar to an alarm or reminder to the user to check and balance the power consumption of each appliance whether a person is sitting in a particular room or there is electricity wastage, as shown in Figure 8.

There is also a billing calculation system based on device electricity consumption and the time duration in which the device remains in the working state.

5. Results and Discussion

The functional behavior of this project can be further understood using Figure 9. The complete workflow of the project comprises two ways as we discussed earlier. In the first scenario, the functional behavior of the project assumes that the user is outside the home and accessing each device remotely. This scenario assumes the involvement of using the Microsoft Azure cloud database. Each request from the user is firstly received on the cloud, and according to the credentials provided by the user, APIs are called.

In the second scenario, we assume that the user is inside the home and connected to the same network to which Raspberry Pi is connected. All the requests generated by the user do not go to the internet or cloud rather than operating

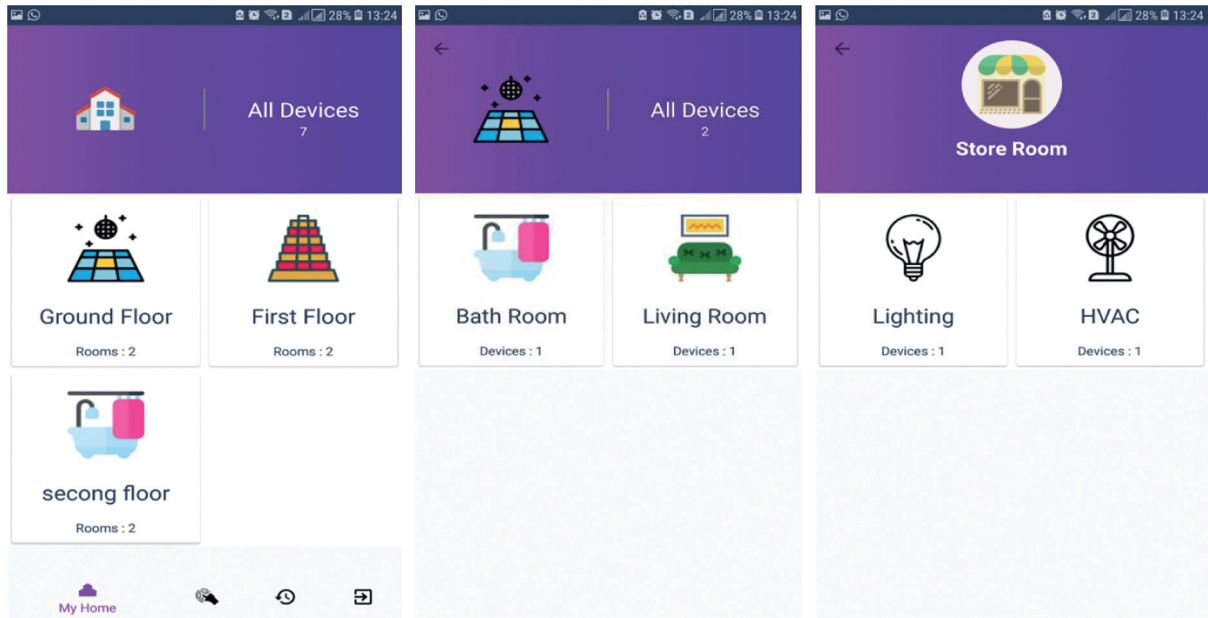


FIGURE 5: GUI of the user main screen.

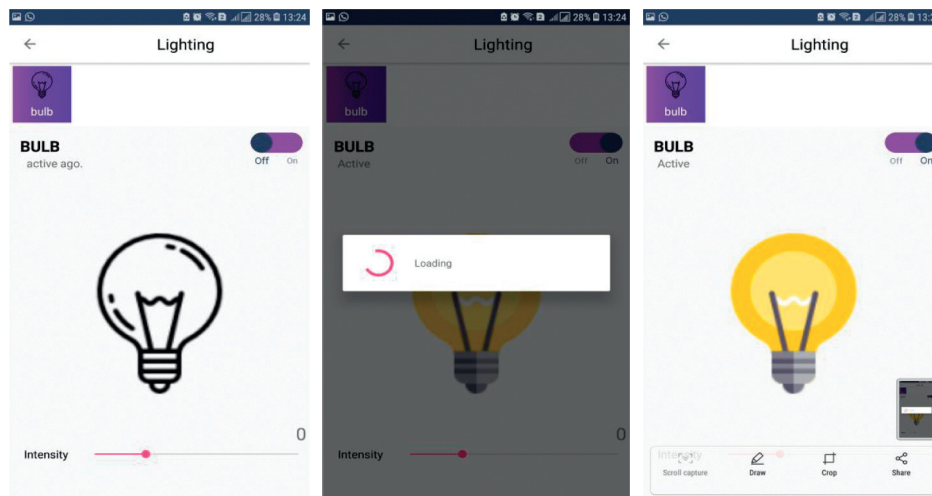


FIGURE 6: Changing the state of the appliance.

on Raspberry Pi server internally, as shown in Figure 10. This will lead to the processing of requests more quickly than the involvement of the cloud.

We have also stored data taken from the history tab of the developed App and stored it in a Microsoft Excel sheet (see Figure 11) and further used it to show the relationship between data that are discussed in this section. The excel sheet contains the values of temperature, smoke, and light. There is a status column where 0 represents the “OFF” status of smoke and light while 1 represents the “ONN” status of smoke and light. We have also drawn a joint plot between temperature and smoke, as shown in Figure 12 where each observation of temperature is shown along the x -axis and smoke values are shown along the y -axis. The histogram along the x -axis represents the different readings of temperature, and the histogram along the y -axis represents the different readings of smoke.

In Figure 11, the joint plot is drawn that shows the relationship between light and temperature. The values of light are shown along the x -axis, and temperature is shown along the y -axis. The histogram of light and temperature is also shown in Figure 13 along the x -axis and y -axis, respectively.

A scatter plot is also drawn to show the relationship between smoke and light (see Figure 14). The smoke values are shown along the x -axis, and light values are shown along the y -axis. The histogram of smoke is also represented along the x -axis, and the histogram of light is represented along the y -axis as shown in Figure 14.

We have shown the status of light from taken readings using the two-dimensional kernel density plot in Figure 15. It shows the status of light when it is “ON” and “OFF.” The values of light fall between zero and six hundred, and the

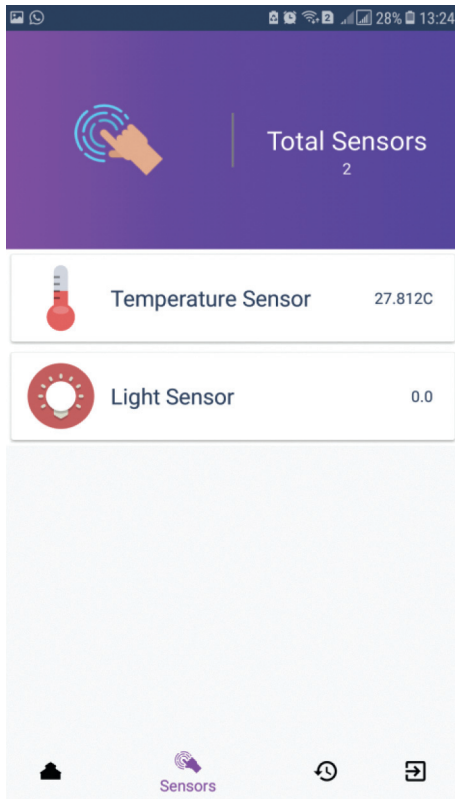


FIGURE 7: Sensor service data.

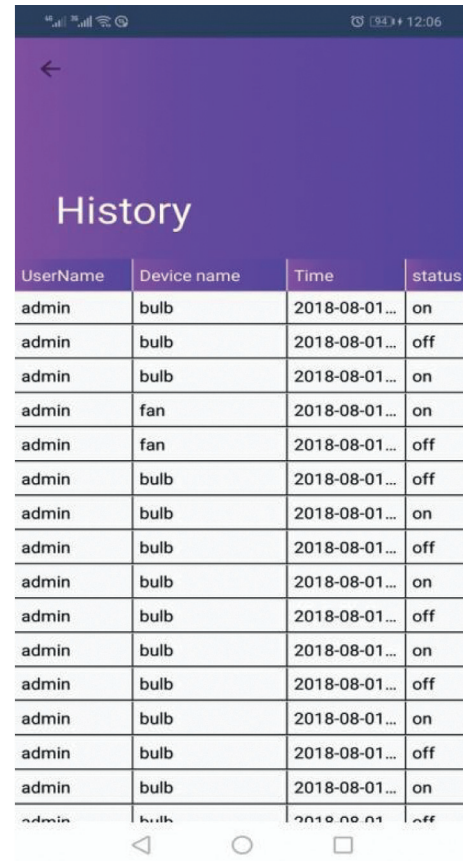


FIGURE 8: A history log of the device.

status of values lie between 0 and 1 that shows “OFF” and “ONN,” respectively.

The kernel density plot for smoke is shown in Figure 15 that shows the status of smoke along the x -axis and its values along the y -axis. The presence of smoke is represented as zero and absence as one in Figure 16.

One of the most common types of graphics that are used to show the numeric values of variables is the box plot. It divides the dataset into three quartiles. The line that divides the box into two parts represents the median of the data, and the end of the corner shows the upper and lower quartiles. The extreme lines represent the highest and lowest values excluding outliers. It also hides the number of the values that lie behind the variable. We have also drawn a box plot of light, temperature, and smoke, as shown in Figures 17–19.

To make intelligent decisions about the status of home appliances, we have used a linear SVM classifier. First of all, we divided the recorded data into training and test data sets using the built-in function `train_test_split` (\cdot) of Scikit learn in Python. Figure 20 shows the confusion matrix of trained SVM classifiers that shows% of truly predicted and wrongly predicted classes.

In the confusion matrix, the instances of the true class are represented horizontally while instances of the predicted class are represented vertically. There were a total 49 predicted “OFF” classes out of which only one class was wrongly predicted as “ON” that was actually “OFF” while the total “ON” classes were 17 out of which only 2 classes were wrongly predicted as “OFF” that was actually “ON.” We have

also plotted the ROC curve of the proposed linear SVM classifier, as shown in Figure 21. In the ROC curve, the true positive rate of a linear SVM classifier was shown on the y -axis and the false positive rate was represented along the x -axis. The ROC of the linear SVM classifier showed an $AUC = 0.957$, as shown in Figure 21.

We have also compared linear SVM classifiers with other available classifiers such as KNN, Decision Tree, and Random Forest. Among all these classifiers, SVM outperforms, as shown in Figure 22. Therefore, the presented approach uses a linear SVM classifier for decision making about the status of home appliances.

In Figure 22, the accuracy of the SVM classifier is 95.73% that is greater than that of all other machine learning classifiers, i.e., K-Nearest Neighbour (KNN), Naïve Base (NB), Random Forest (RF), and Decision Tree (DT) having accuracy 94.32%, 93.80%, 93.24%, and 94.73%, respectively.

The implementation is performed in a way that if the Raspberry Pi restarts or any mishap happens in electricity, then all the devices are set again to the state before the restart. This involves the use of a database server that maintains the state of each device. The last state record of each device is being fetched and set again to each device accordingly. The application will get its details from the server too.

Sensors are installed to get the updates at regular intervals. There is a condition set out for user comforts that if

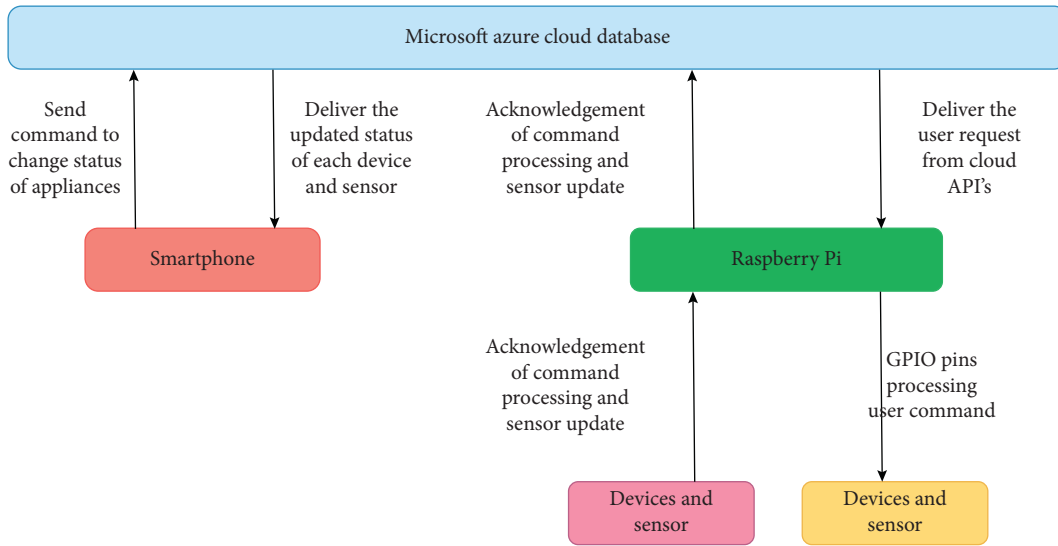


FIGURE 9: Microsoft Azure cloud database.

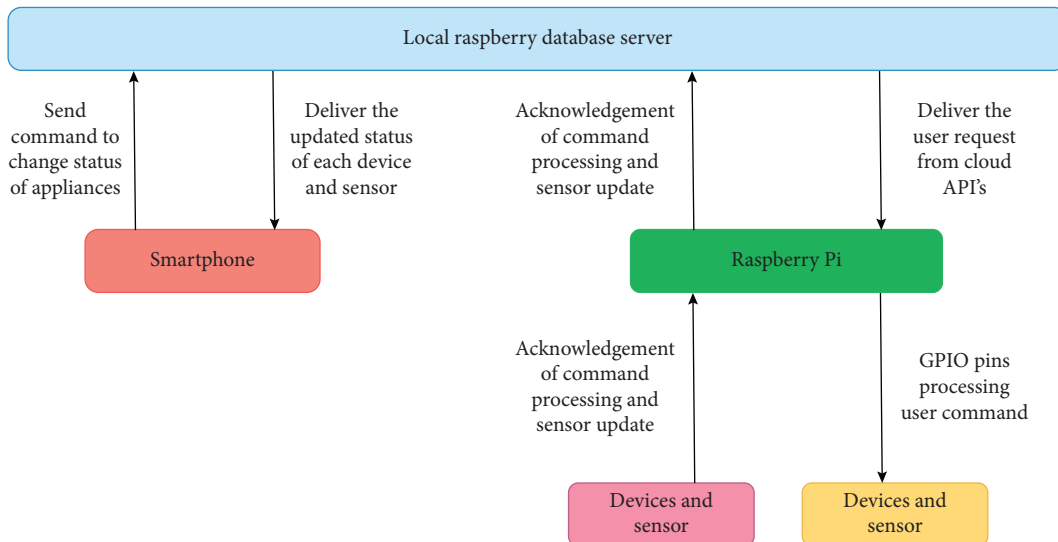


FIGURE 10: Local database server.

Temperature	Smoke	Light	Status
28.54	0	0	0
28.68	0	0	0
28.78	0.5	51	1
28.9	0	0	0
29.35	0	0	0
29.65	0.8	154	1
29.78	0.6	182	1
29.88	0.7	203	1
29.05	0.51	204	1
29.67	0.56	256	1
29.99	0.58	258	1
30.05	1.2	287	1
30.1	0.32	290	1
31.21	0	0	0
31.56	0.26	298	1
31.88	0	0	0
32.56	0	0	0

FIGURE 11: Recorded data.

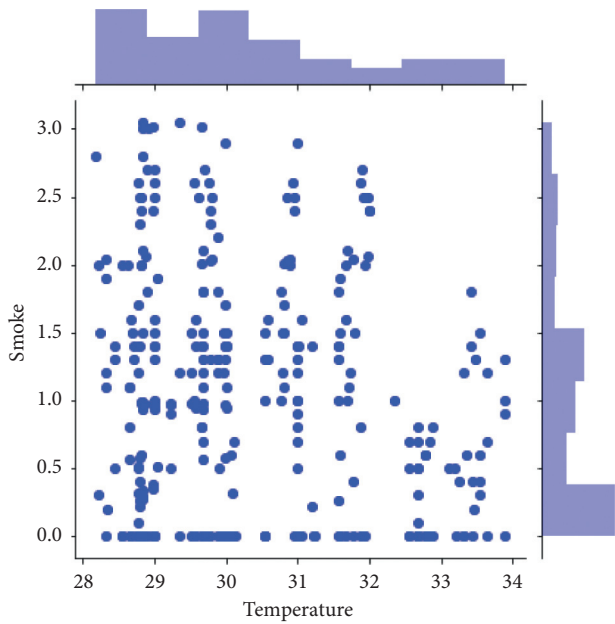


FIGURE 12: Scatter plot between temperature and smoke.

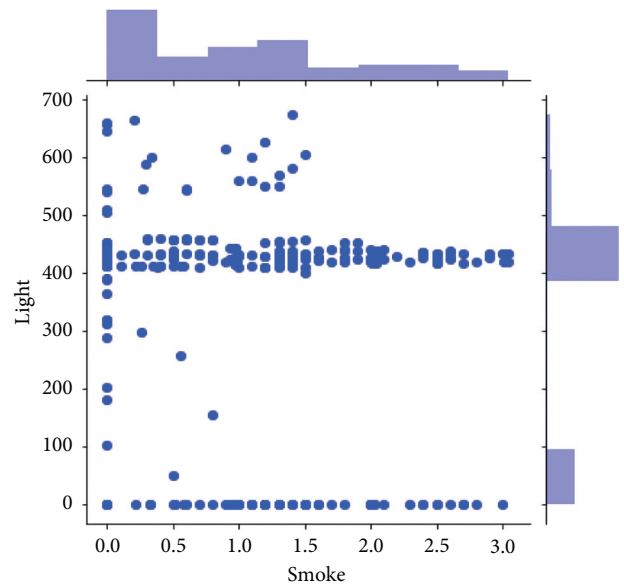


FIGURE 14: Scatter plot between smoke and light.

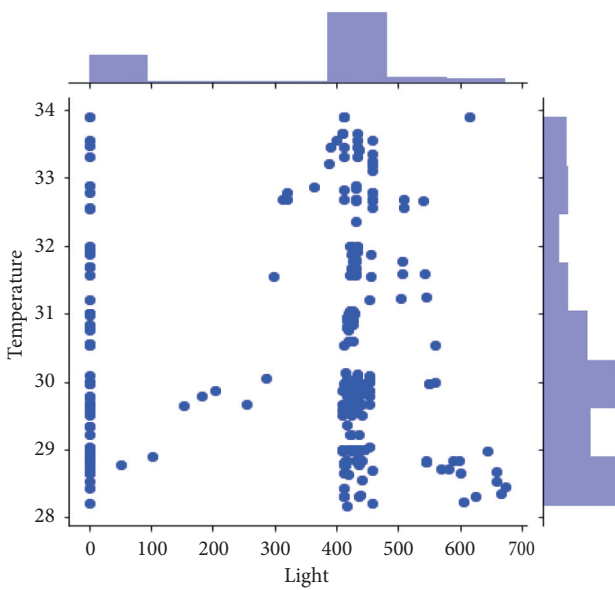


FIGURE 13: Scatter plot between light and temperature.

the temperature of the home rises up to the threshold set by the user, then all ventilation fans will start working automatically and vice-versa. Light sensors controlled the state of lights installed outside the home automatically in daylight and on the night. A comparison of the old and the proposed system is shown in Table 3.

All the features and functionality comparisons described in the table above will make this project unique from the rest. Designing the prototype of your own house and setting each device according to the room design will make it easier for a user to operate the electronic device with more ease.

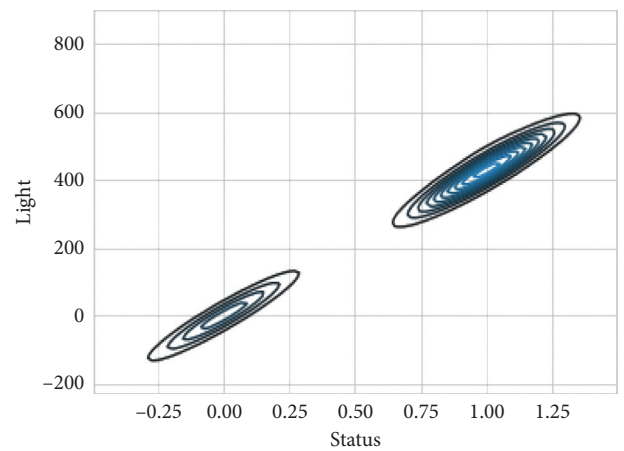


FIGURE 15: Kernel density plot of light.

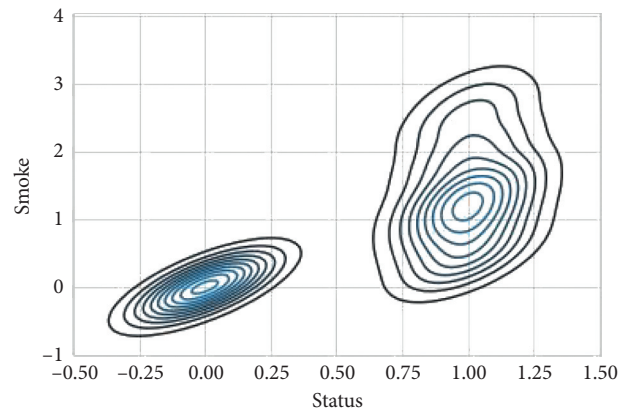


FIGURE 16: Kernel density plot of light.

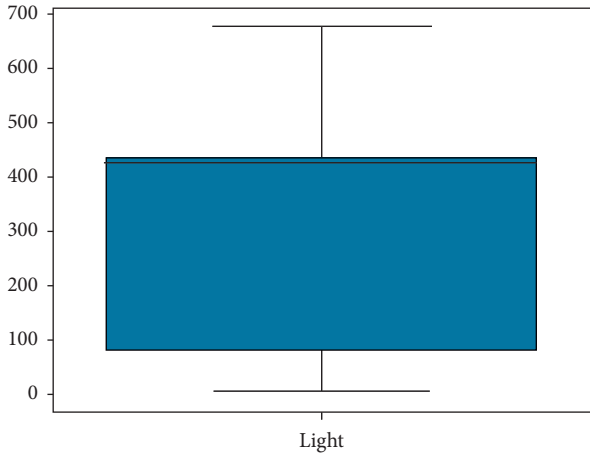


FIGURE 17: Box plot of light.

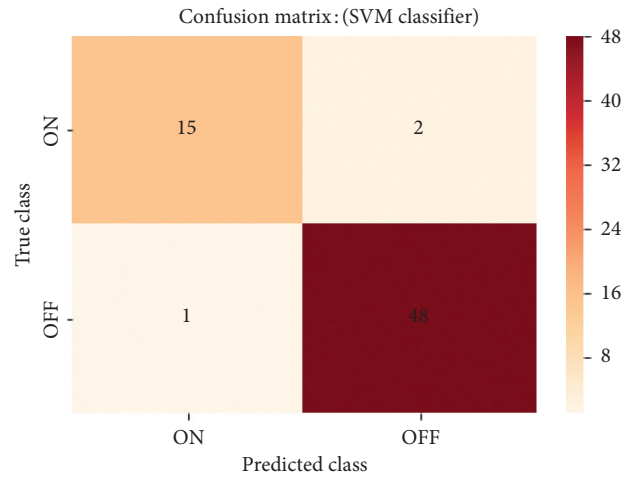


FIGURE 20: Confusion matrix.

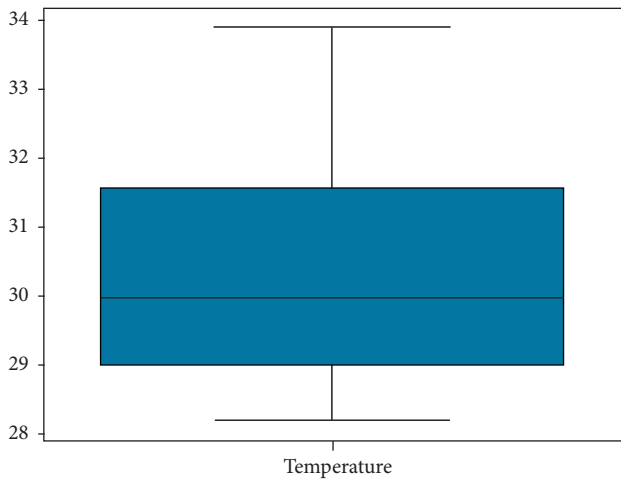


FIGURE 18: Box plot of temperature.

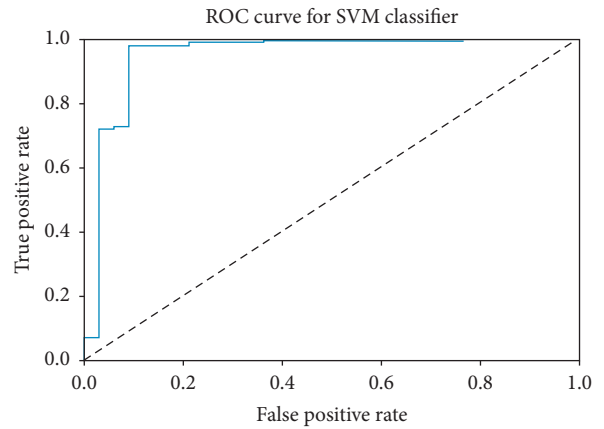


FIGURE 21: ROC curve of the SVM classifier.

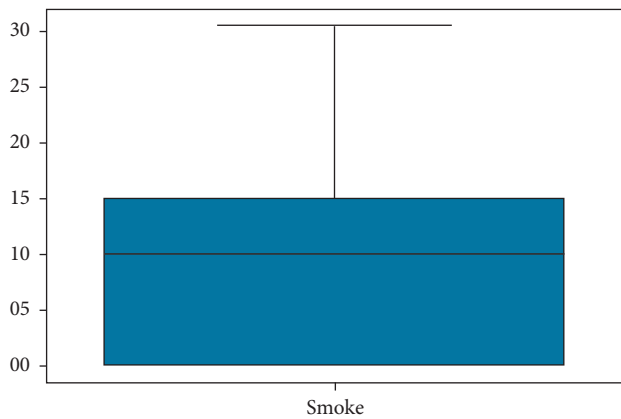


FIGURE 19: Box plot of smoke.

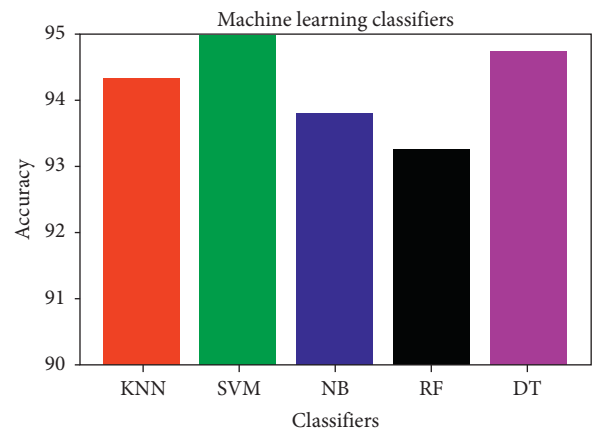


FIGURE 22: Comparison between machine learning classifiers.

TABLE 3: Comparison between the old and proposed system.

Functionality	Old system	Proposed system
Raspberry support	No	Yes
Data logging	No	Yes
Home prototype design	No	Yes
Sensor update	No	Yes
Intensity control	No	Yes
Billing system	No	Yes
Cloud database support	No	Yes
Machine learning classifier	No	Yes
Blockchain technology	No	Yes

6. Conclusions

In this research paper, the complete design scheme and working methodology of a home automation project are explained. The reason behind the paper is to get enhancement ideas to make it better and user friendly. This proposed system comprises two modes, the admin side in which the user able to design the complete prototype of the home and the user side in which the user able to control each device of the home with an easy GUI-based interface. The proposed system also has decision-making ability about the status of each device of the home. A machine learning algorithm SVM with a linear kernel is applied for decision making about the status of home appliances either ON or OFF. The presented approach also ensures secure identification and authentication of IoT devices using blockchain technology, while the existing systems which have not copped with intelligent decision making and analytical abilities are the need of time. The existing system also does not provide secure communication between IoT devices. Moreover, this project is user acceptance tested verified and operational in many homes. It provides simplicity, flexibility, reliability, and a low-cost system that is affordable to middle-class families too.

Data Availability

The data used to support the findings of the study are available from the corresponding author upon request.

Disclosure

Rizwan Majeed, Imran Ashraf, and Yousaf Bin Zikria are primary co-authors.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

- [1] E. S. A. Ahmed, "Internet of things applications, challenges and related future technologies," *Internet of Things (IoT) Applications*, vol. 67, no. 2, pp. 126–148, 2017, https://www.researchgate.net/profile/Zeinab_Kamal2.
- [2] R. Piyare, "Internet of things: ubiquitous home control and monitoring system using android based smart phone,"

- International Journal of Internet of Things*, vol. 2, no. 1, pp. 5–11, 2013.
- [3] K. Mandula, R. Parupalli, C. A. Murty, E. Magesh, and R. Lunagariya, "Mobile based home automation using internet of things (IoT)," in *Proceedings of the 2015 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT)*, pp. 340–343, IEEE, Kumaracoil, India, December 2015.
- [4] B. Kang, S. Park, T. Lee, and S. Park, "IoT-based monitoring system using tri-level context making model for smart home services," in *Proceedings of the 2015 IEEE International Conference on Consumer Electronics (ICCE)*, pp. 198–199, IEEE, Las Vegas, NV, USA, January 2015.
- [5] M. Darianian and M. P. Michael, "Smart home mobile RFID-based internet-of-things systems and services," in *Proceedings of the 2008 International Conference on Advanced Computer Theory and Engineering*, pp. 116–120, IEEE, Phuket, Thailand, December 2008.
- [6] A. D. Plessis and B. Theron, "Virtual world—physical world: what is the real world?" *International Journal of Management Science and Business Administration*, vol. 2, no. 6, pp. 43–57, 2016.
- [7] G. Lobaccaro, S. Carlucci, and E. Löfström, "A review of systems and technologies for smart homes and smart grids," *Energies*, vol. 9, no. 5, p. 348, 2016.
- [8] O. Bingol, K. Tasdelen, Z. Keskin, and Y. E. Kocaturk, "Web-based smart home automation: PLC-controlled implementation," *Acta Polytechnica Hungarica*, vol. 11, no. 3, pp. 51–63, 2014.
- [9] A. ElShafee and K. A. Hamed, "Design and implementation of a wifi based home automation system," *World Academy of Science, Engineering and Technology*, vol. 68, pp. 2177–2180, 2012.
- [10] T. A. Abdulrahman, O. H. Isiwekpeni, N. T. Surajudeen-Bakinde, and A. O. Otuoze, "Design, specification and implementation of a distributed home automation system," *Procedia Computer Science*, vol. 94, pp. 473–478, 2016.
- [11] M. Sarwar and T. Soomro, "Impact of smartphone's on society," *European Journal of Scientific Research*, vol. 98, 2013.
- [12] M. Fahim, I. Fatima, S. Lee, and Y. K. Lee, "Daily life activity tracking application for smart homes using android smartphone," in *Proceedings of the 2012 14th International Conference on Advanced Communication Technology (ICACT)*, IEEE, PyeongChang, South Korea, pp. 241–245, February 2012.
- [13] S. Otoum, B. Kantarci, and H. T. Mouftah, "On the feasibility of deep learning in sensor network intrusion detection," *IEEE Networking Letters*, vol. 1, no. 2, pp. 68–71, 2019.
- [14] K. Gill, S.-H. Yang, F. Yao, and X. Lu, "A zigbee-based home automation system," *IEEE Transactions on Consumer Electronics*, vol. 55, no. 2, pp. 422–430, 2009.
- [15] N. Sriskanthan, F. Tan, and A. Karande, "Bluetooth based home automation system," *Microprocessors and Microsystems*, vol. 26, no. 6, pp. 281–289, 2002.
- [16] A. R. Al-Ali and M. Al-Rousan, "Java-based home automation system," *IEEE Transactions on Consumer Electronics*, vol. 50, no. 2, pp. 498–504, 2004.
- [17] H. Ardam and I. Coskun, "A remote controller for home and office appliances by telephone," *IEEE Transactions on Consumer Electronics*, vol. 44, no. 4, pp. 1291–1297, 1998.
- [18] T. Baudel and M. Beaudouin-Lafon, "Charade," *Communications of the ACM*, vol. 36, no. 7, pp. 28–35, 1993.
- [19] A. S. Abdulraheem, A. A. Salih, A. I. Abdulla et al., "Home automation system based on IoT," 2020.

- [20] M. A. Hoque and C. Davidson, "Design and implementation of an IoT-based smart home security system," *International Journal of Networked and Distributed Computing*, vol. 7, no. 2, pp. 85–92, 2019.
- [21] L. M. Satapathy, S. K. Bastia, and N. Mohanty, "Arduino based home automation using internet of things (IoT)," *International Journal of Pure and Applied Mathematics*, vol. 118, no. 17, pp. 769–778, 2018.
- [22] S. Pirbhulal, H. Zhang, M. E. E. Alahi et al., "A novel secure IoT-based smart home automation system using a wireless sensor network," *Sensors*, vol. 17, no. 1, p. 69, 2017.
- [23] P. Kumar and U. C. Pati, "IoT based monitoring and control of appliances for smart home," in *Proceedings of the 2016 IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*, pp. 1145–1150, IEEE, Bangalore, India, May 2016.
- [24] S. B. Sangeetha, "Intelligent interface based speech recognition for home automation using android application," in *Proceedings of the 2015 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)*, pp. 1–11, IEEE, Coimbatore, India, March 2015.
- [25] D. Javale, M. Mohsin, S. Nandanwar, and M. Shingate, "Home automation and security system using android ADK," *International Journal of Electronics Communication and Computer Technology (IJECCCT)*, vol. 3, no. 2, pp. 382–385, 2013.
- [26] R. Piyare, M. Tazil, Bluetooth based home automation system using cell phone," in *Proceedings of the 2011 IEEE 15th International Symposium on Consumer Electronics (ISCE)*, pp. 192–195, IEEE, Singapore, June 2011.
- [27] G. Chong, L. Zhihao, and Y. Yifeng, "The research and implement of smart home system based on internet of things," in *Proceedings of the 2011 International Conference on Electronics, Communications and Control (ICECC)*, pp. 2944–2947, IEEE, Ningbo, China, September 2011.
- [28] Z. U. Haq, G. F. Khan, and T. Hussain, "A comprehensive analysis of XML and JSON web technologies," *New Developments in Circuits, Systems, Signal Processing, Communications and Computers*, pp. 102–109, 2013.
- [29] Y. J. Song, S. B. Ou, and J. W. Lee, *An Analysis of Existing Android Image Loading Libraries: Picasso, Glide, Fresco, AUIL and Volley*, DEStech Transactions on Engineering and Technology Research, Lancaster, PA, USA, 2016.
- [30] L. Hang and D.-H. Kim, "Design and implementation of intelligent fire notification service using IP camera in smart home," *International Journal of Control and Automation*, vol. 11, no. 1, pp. 131–142, 2018.
- [31] V. N. Vapnik, *The Nature of Statistical Learning Theory*, Springer-Verlag, New York, NY, USA, 1995.
- [32] S.-T. NelloCristianini, *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*, Cambridge University Press, Cambridge, UK, 1st edition, 2000.
- [33] V. Jakkula, *Tutorial on Support Vector Machine (SVM)*, p. 37, Washington State University, Pullman, WA, USA, 2006.
- [34] J. Han, J. Pei, and M. Kamber, *Data Mining: Concepts and Techniques*, Morgan Kaufmann, Burlington, MA, USA, 2011.
- [35] S. Nakamoto, "Bitcoin: apeer-to-peer electronic cash system," Triumph Books, Chicago, IL, USA, 2008.
- [36] K. Christidis and M. Devetsikiotis, "Blockchains and smart contracts for the internet of things," *IEEE Access*, vol. 4, pp. 2292–2303, 2016.
- [37] M. Maksimović, V. Vujović, N. Davidović, V. Milošević, and B. Perišić, "Raspberry pi as Internet of things hardware: performances and constraints," in *Proceedings of IcETRAN 2014*, Vrnjačka Banja, Serbia, June 2014.
- [38] V. Patchava, H. B. Kandala, and P. R. Babu, "A smart home automation technique with raspberry pi using iot," in *Proceedings of the 2015 International Conference on Smart Sensors and Systems (IC-SSS)*, pp. 1–4, IEEE, Bangalore, India, December 2015.
- [39] M. Richardson and S. Wallace, *Getting Started with Raspberry Pi*, O'Reilly Media, Inc., Sebastopol, CA, USA, 2012.
- [40] E. Upton and G. Halfacree, *Raspberry Pi User Guide*, John Wiley & Sons, NewYork, NY, USA, 2014.
- [41] F. Leccese, M. Cagnetti, and D. Trinca, "A smart city application: a fully controlled street lighting isle based on raspberry-pi card, a zigbee sensor network and wimax," *Sensors*, vol. 14, no. 12, pp. 24408–24424, 2014.
- [42] L. Qian, Z. Luo, Y. Du, and L. Guo, "Cloud computing: an overview," in *Proceedings of IEEE International Conference on Cloud Computing*, pp. 626–631, Beijing, China, December 2009.
- [43] Z. Wei, S. Qin, D. Jia, and Y. Yang, "Research and design of cloud architecture for smart home," in *Proceedings of the 2010 IEEE International Conference on Software Engineering and Service Sciences*, pp. 86–89, IEEE, Beijing, China, July 2010.
- [44] J. D. Brock, R. F. Bruce, and M. E. Cameron, "Changing the world with a raspberry pi," *Journal of Computing Sciences in Colleges*, vol. 29, no. 2, pp. 151–153, 2013.
- [45] Sunfounder, "L293D Motor driver shield," 2019, http://wiki.sunfounder.cc/index.php?title=L293D_Motor_Driver_Shield.
- [46] Sparkfun, "Temperature sensor-waterproof (DS18B20)," 2019, <https://www.sparkfun.com/products/11050>.
- [47] Lm393 optical photosensitive LDR light sensor module, <https://www.daraz.pk/products/lm393-optical-photosensitive-ldr-light-sensor-module-i100135168-s1118660259.html>, 2019.
- [48] Arduino and MQ2 gas sensor, <https://www.hackster.io/karimmufte/arduino-and-mq2-gas-sensor-57f98c>, 2019.
- [49] S. A. I. Quadri and P. Sathish, "IoT based home automation and surveillance system," in *Proceedings of the 2017 International Conference on Intelligent Computing and Control Systems (ICICCS)*, pp. 861–866, IEEE, Madurai, India, June 2017.
- [50] Y. Upadhyay, A. Borole, and D. Dileepan, "MQTT based secured home automation system," in *Proceedings of the 2016 Symposium on Colossal Data Analysis and Networking (CDAN)*, pp. 1–4, IEEE, Indore, India, March 2016.
- [51] S. S. Harsha, S. C. Reddy, and S. P. Mary, "Enhanced home automation system using internet of things," in *Proceedings of the 2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, pp. 89–93, IEEE, Palladam, India, February 2017.
- [52] S. Chatteraj, "Smart home automation based on different sensors and arduino as the master controller," *International Journal of Scientific and Research Publications*, vol. 5, no. 10, pp. 1–4, 2015.
- [53] H. Esmaeel, "Apply android studio (SDK) tools," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 5, 2015.
- [54] K. K. Rout, S. Mishra, and A. Routray, "Development of an internet of things(iot) based introductory laboratory for under graduate engineering students," in *Proceedings of the 16th International Conference on Information Technology (ICIT)*, Bhubaneswar, India, October 2017.