Easy-Weigh-Out: Design and Implementation of Internet of Things-Based Smart Luggage System

https://doi.org/10.3991/ijoe.v17i03.17165

Mohammad Haasin Farooq, Muhammad Zain, Muhammad Bilal Khalid,
Saima Zafar (✉)
National University of Computer and Emerging Sciences, Lahore, Pakistan
saima.zafar@nu.edu.pk

Abstract—The Internet of Things (IoT) has dominated the digital world by enabling thousands of physical devices to communicate with each other and share useful data over the Internet. With use cases spanning every sphere of modern life, the IoT is all set to revolutionize the traveling sector by extending services associated with safety, security, surveillance, tracking, comfort, and a lot more. This paper presents the design and implementation of a prototype Easy-Weigh-Out, based on the principles of the IoT which enables a user to run Android-based mobile application and view the weight of luggage and track its location without any contact with it. The hardware of the system is built around Arduino UNO microcontroller which interfaces with a load cell (20 kg) via A/D converter and a NEO-6M Global Positioning System (GPS) module for tracking of the bag. The system uploads data to a Cloud server using ESP8266 Wi-Fi module. Our results show the real-time weight and tracking of the bag at a click of a button as well as long-term data logging.

Keywords—Cloud service, Internet of Things, mobile application, smart traveling bag, tracking

1 Introduction

The Internet of Things (IoT), also referred to as the Internet of Everything (IoE), has brought a new era of computing where an ecosystem of discrete computing devices with sensors are connected through the infrastructure of the Internet thereby allowing collection and communication of useful data [1-2]. The IoT allows connected devices to gather information and analyze it in order to perform a specific task. These devices, however, need to be able to speak to each other to transfer data and share what they are recording. This is where Cloud Computing acts as a catalyst for the development and deployment of IoT applications [3]. Cloud services allow the collected data to be stored and remotely accessed in an efficient manner. This data can then be analyzed by a software, which processes it and performs an action, depending on the application.

The IoT offers benefits for several industries by monitoring data and improving customer experience by saving time [4]. One of IoT's applications is the automation of travel and tourism accessories, such as travelling bags. Traveling and Tourism is one of the fastest growing industries in the world, where customers expect innovation at affordable prices, thereby attracting an increasing competition amongst various companies [5]. Due to a proliferation of technology-based innovations and services, the customers expect technology-based services in traveling and tourism such that both pre-travel and post-travel issues and glitches are minimized. Through investment in the right technology and innovative ideas for use cases comprising of smart features, the IoT has revolutionized the overall travel and tourism industry [6]. These features include smarting-locking system, GPS tracking facility, smart-scaling etc.

In this paper, we present the design and implementation of Easy-Weigh-Out; an IoT-based system for remotely weighing the luggage and tracking it using a mobile application. The system as depicted in Fig. 1 allows the user to log in to a mobile application and be able to find out the weight of a luggage by clicking a button on screen and to track that luggage in real-time from the start of the trip till its end. Our system targets the two most frustrating travel related issues; facing vexation or extra charges due to overweighing luggage and misplaced or lost luggage due to mishandling or negligence. Our system employs Arduino UNO microcontroller which is interfaced with a Load Cell used to measure the weight of luggage in various ranges and the GPS for tracking information of the luggage. Our Load Cell has a capacity to measure a weight up to 20 kg and we use NEO-6M GPS for longitude and latitude information [8]. This real-time data of weight and location is uploaded to an open-source Cloud service ThingSpeak over the Wi-Fi module ESP8266 from where this information is accessed via Android-based mobile application by the end user. The Cloud service also uses MATLAB analytics to show long term data trends and graphs [9]. The device is fixed to the luggage and moves along with it and the mobile application can be used from anywhere in the world to track the luggage.

The rest of the paper is organized as follows: Section II discusses some of the notable related work, Section III elaborates the hardware design of the system, Section IV provides an insight to the software design, Section V presents the graphical visualization of the results that attest to the usefulness of the system, and finally, Section VI concludes the paper.

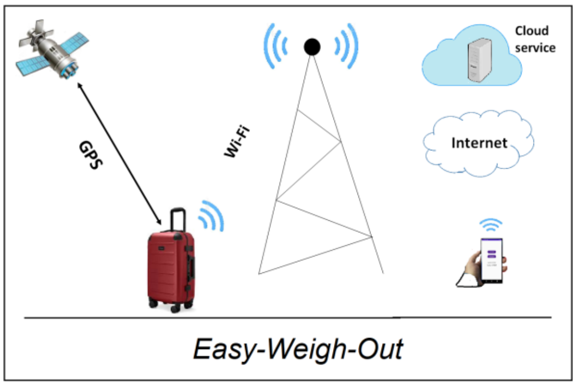


Fig. 1. Easy-Weigh-Out: IoT-based smart luggage system

2 Related Work

A closely related work to ours is a one-touch luggage security system by Abhang *et al.* which uses the GPS and the Short Message Service (SMS) of GSM to provide the exact location of luggage to the user [10]. An interesting feature of their system is geo-fencing which is a virtual geographical radius used to trigger the system whenever the bag crosses the fence. In case that happens, images are captured via a USB camera attached to the bag in order to identify the thief. These images along with the GPS coordinates are sent to the user and displayed on an Android application. The development board used in this system is Raspberry Pi. Another smart luggage system based on Raspberry Pi and Arduino was developed by Sandi *et al.* who make use of USB microphone to send audio commands like “Hey Google” or “Ok Google” for triggering the system to start processing on google Cloud via Raspberry Pi [11]. The processed data is sent back to Raspberry Pi which is executed as voice through speakers. The different functionalities of the system include luggage a locking/unlocking system using Face Recognition Technology (FRT) where if an unknown attempt is observed, the real-time location of the luggage is sent to the user via GSM/GPRS module, a real-time luggage tracking system using GPS sensor with Arduino board, and a smart control system for the movement of the luggage. This system is quite complex as compared to our system which is based on Arduino UNO microcontroller. A multi-purpose smart bag was developed by Shweta *et al.* which is a low cost, solar power based smart bag that uses RF-ID technology to provide features like remembering stuff to be put inside the bag and powering of electronic equipment [12]. The items to be placed in the bag have RF-ID tags attached to them, and these are read by an RF-ID reader whenever an item is placed in the bag. A solar panel is attached at the front of the bag that powers the various electronic equipment as well as batteries. The microcontroller used in this system is a low power Atmel 8-bit microcontroller called ATmega16. A Bluetooth module acts as mediator between the embedded system and the mobile device. This system is relatively expensive as compared to ours, mainly due to its use of solar power and RF-ID technology.

Another smart luggage project was reported at 1000projects which is a system for automatic weighing, location tracking and autonomous movement of luggage employing the GPS and Wi-Fi technologies [13]. A mobile application enables the user to operate these tasks using buttons. Upon selection of buttons manual/autonomous movements are carried out. If the user selects autonomous movement option, the distance of the bag to the user is measured by ultrasonic sensors and the bag starts moving closer to the user with the help of motor drivers. An alert message is sent if an obstacle comes in-between. If manual movement is selected, the bag is moved according to the commands sent by the user’s mobile over Wi-Fi. Similarly weight and location of bag are displayed if required. The development board used for this project is Arduino Mega. Senthikumar *et al.* developed IoT-based passenger luggage tracking system [14]. Arduino was used as development board with a GPS module connected to an alarm. If the bag is outside a certain range, gets stolen, or is lost, the alarm notifies the user. This system is synchronized with a map created on Google Geolocation API to help the user in tracking down the exact location of the luggage. Markers and flags on the map identify the bag when it moves out of range, for example, 20 or 30 meters. This system only tracks the location of the bag, while our system, not only tracks the location of the bag, but also measures its weight it and employs cloud computing in order to provide tracking of the bag irrespective of user location.

3 Hardware Design of Easy-Weigh-Out

Major hardware components of Easy-Weigh-Out system comprise of Arduino UNO microcontroller which is the main micro-processing controlling the flow of data and processing, Load Cell and GPS module for input of weight and location of the luggage to the microcontroller, a Wi-Fi module which sends this data wirelessly to the Cloud service from where it is retrieved and displayed on a mobile application. The hardware block diagram of the complete system is shown in Fig. 2. The figure shows two sensors interfaced with Arduino UNO, the Load Cell is interfaced through an A/D converter which converts analog signal to digital signal and the GPS module. Arduino UNO is powered by a battery and Wi-Fi module is interfaced at the output to send data wirelessly to the Cloud service. In this section, we elaborate important attributes of these hardware components.

3.1 Microcontroller

Arduino UNO is an easy-to-use microcontroller with cross-platform compatibility and is relatively inexpensive compared to other microcontrollers. It has a simple and transparent programming environment, and provides open source tools for expansion [15]. Thus Arduino is the most suitable microcontroller for our smart luggage system. It acts as a central hub for taking inputs, processing them and sending outputs. Our hardware design employs Arduino board ATmega328P which is powered by a 7-12V power supply with an operating voltage of 5V. The small-sized battery is mounted within the system package.

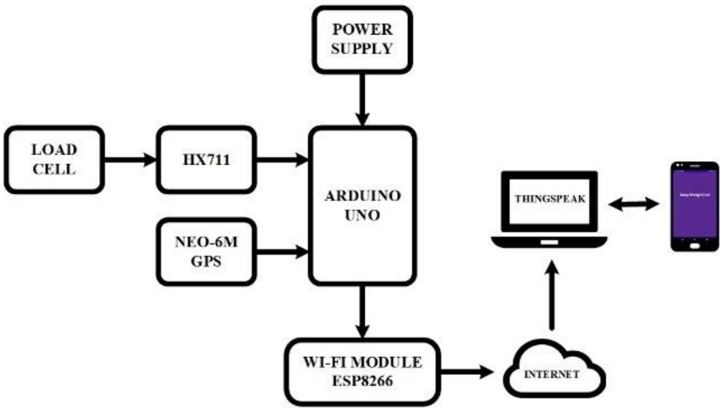


Fig. 2. Hardware block diagram of Easy-Weigh-Out system

It has 3.3V and 5V on-board power supply pins, along with a ground pin. The on-chip peripherals include; a USB port, 14 digital input/output pins (out of which 6 provide PWM output), 6 analog input pins, 2 external interrupts and 2 serial Tx and Rx pins used to transmit and receive serial data. It also provides the functionality of Serial Peripheral Interface (SPI) communication and a frequency of 16 MHz [16]. The microcontroller is programmable with the Arduino IDE (Integrated Development Environment) via a B type USB cable. It can be reset by pressing the on-chip reset button.

3.2 Sensors

To measure the weight of the luggage, we used a Load Cell of 20 kg. Its signal is amplified using an analog-to-digital (ADC) amplifier, HX-711. The Load Cell is a transducer which transforms force or pressure into electrical output. This electrical signal is directly proportional to the force being applied. The HX-711 weighing sensor module is a 24-bit high precision ADC converter. Therefore, after amplifying the electrical signal, it converts it into a digital signal and then feeds it to the Arduino microcontroller, which derives the weight from this input [17]. In order to achieve greater accuracy while measuring the weight, a solid base is required. Therefore, we fixed a metal sheet with the Load Cell, upon which the weight is placed.

In order to track the location of the luggage, we use a NEO-6M GPS module. This module communicates continuously with the satellite in order to get the coordinates. These coordinates are then sent to the Arduino using Universal Asynchronous Receiver/Transmitter (UART), which extracts the required data to estimate the location [18]. This includes the longitude and latitude of the exact position of the luggage.

3.3 Wi-Fi module

In order to send data from sensors to the open-source Cloud service ThingSpeak, we use ESP8266 Wi-Fi module. It is a low-cost Wi-Fi microchip with an integrated TCP/IP protocol stack. The module can work both as a client to host an application, or as server that receives information from another application process. The ESP8266 module is pre-programmed with an AT command set firmware and, hence, it can be directly connected to Arduino to get Wi-Fi connectivity [19]. Upon connection, the microcontroller gets an IP address using which it can communicate over the Internet. After configuring the module with Arduino, we establish a TCP connection to send data to ThingSpeak. The system Wi-Fi module is dependent on Wi-Fi connectivity at the airports and would be set up in advance so that the system automatically configures and connects to Wi-Fi as it moves across airports.

4 Software Design of Easy-Weigh-Out

The software implementation comprises of two parts: programming of Arduino microcontroller for interfacing of hardware components, and software implementation for the design and development of Android-based mobile application. In this section, we explain these software implementations in succession.

4.1 Programming arduino microcontroller

The Arduino Integrated Development Environment (IDE) is an open-source software by Arduino.cc for writing and compiling the code into Arduino module. It was used to program Arduino UNO microcontroller [20]. For testing purpose, the sensors were integrated with the microcontroller, in order to retrieve the sensed data and then send it to the cloud. The Wi-Fi module ESP8266 was first initialized by sending AT commands in order to provide the microcontroller an access to our Wi-Fi network, and then configured as TCP / IP client. After that, we wrote a code to read data from two different sensors; a Load Cell (20 Kg) to measure the weight of luggage and a NEO-6M GPS module to map real-time location of the luggage and simultaneously upload to the cloud. An open source IoT application, ThingSpeak, was then used to visualize and analyze the uploaded data using MATLAB [21]. An IP was assigned to the Wi-Fi module to send data to the cloud. A channel was created on ThingSpeak with three fields; two for the location of luggage (latitude and longitude), and one for its weight. The API key generated from the channel was written to Wi-Fi module first, and then, real time data was visualized on ThingSpeak.

4.2 Android application development

The Android Application was developed using Android Studio, which is the official IDE for Google’s Android operating system [22]. It is based on IntelliJ IDEA, which is a high cost IDE that supports Android [23]. The back-end of the application is programmed in Java programming language, while the front-end is designed using eXtensible Markup Language (XML). The purpose of our application is to communicate with the microcontroller through ThingSpeak cloud service. It does this by fetching Representational State Transfer (REST) Application Programming Interface (API) data through GET request method, by first setting up a secure Hyper Text Transfer Protocol (HTTP) connection with the server. For this, ThingSpeak website Uniform Resource Locator (URL), which consists of channel URL, channel ID, and API key, is provided. Upon getting an HTTP_OK response, the JavaScript Object Notation (JSON) data is received and stored. After parsing the JSON data, real time luggage weight, as well as the luggage location using latitude and longitude, are displayed on the application. The selection can be made by pressing the appropriate button as shown in the results section. The Graphical User Interface (GUI) of Android application which we developed for our system is shown in the results section.

5 Results

The complete circuit set up of our system is shown in Fig. 3. It clearly shows all the connections between the sensors and the Arduino board. This section presents the experimental results obtained after implementation of the system. A graphical representation of the sensed data on ThingSpeak cloud is shown along with its discussion. The data fetched by the user on the mobile application is also presented here. Table 1 gives the details of various components used in our system.

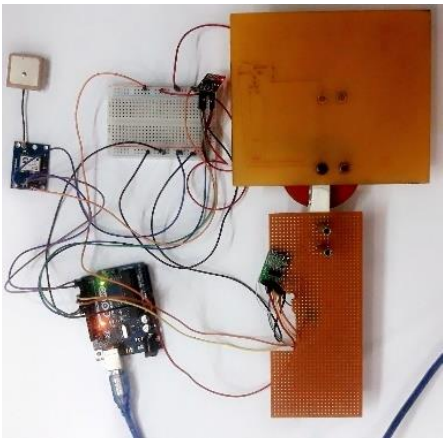


Fig. 3. Circuit implementation of complete system

Table 1. Details of components

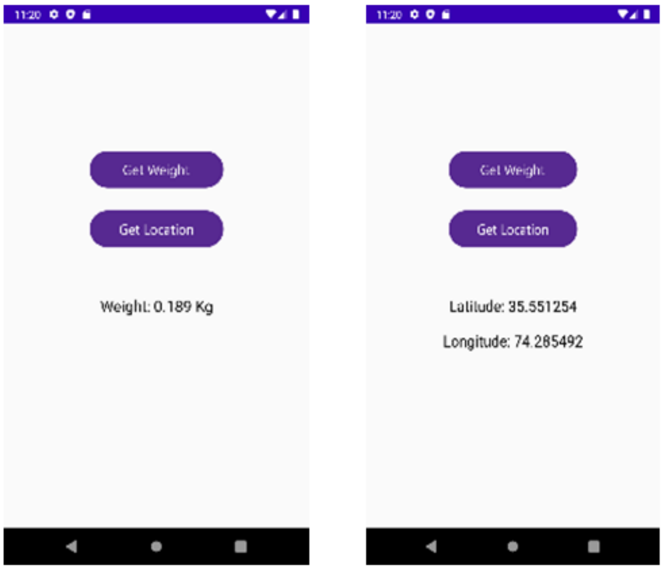
Components	Details
Sensor	Hx-711 with Load Cell, NEO-6M GPS Module
Connectivity	ESP8266 Wi-Fi Module
Microcontroller	Arduino UNO
Programming platform	Arduino IDE
Cloud service	ThingSpeak
User interface	Android Application

5.1 Application graphical user interface

The user has a choice of displaying the weight of the bag and checking its location. The GUI along with fetched results is shown in Fig. 4. If the user presses the “Get Weight” button, the weight of the luggage is fetched from the cloud and displayed on the screen in kilograms. If the user presses the “Get Location” button, the current latitude and longitude of luggage is fetched from the cloud and displayed on the screen.

5.2 Graphical analysis of data

A record of luggage weight measured over a period of time is shown in Fig. 5. The graph shows the different intervals of time in a day on the x-axis and the weight of luggage (in kilograms) on the y-axis and. The readings are updated after every 15 seconds on ThingSpeak.



Weight: 0.189 Kg

Latitude: 35.551254

Longitude: 74.285492

Fig. 4. Weight and Location readings on system GUI

To demonstrate accuracy, we placed different weights on the scale, as seen in the graph. It should be noted that the latest value shown in the graph, is fetched and displayed on the mobile application, as shown in Fig. 4. A record of the latitude and longitude that were monitored over a period of time are shown in Fig. 6 and 7 respectively. The graphs show the different intervals of time in a day on the x-axis and latitude and longitude on the y-axis correspondingly. The readings are updated after every 15 seconds on ThingSpeak. In order to show variations in location, we moved our system around and that is reflected in the graphs. It should also be noted that the latest values shown in the graphs are fetched and displayed on the mobile application, as shown in Fig. 4. We tested our system by fixing it with a wheeled luggage and using the mobile application to track it from anywhere. Presently the system was developed only for one luggage therefore we did not include any luggage ID or color. In future if the system is extended for tracking more luggage items then unique IDs can be assigned to various luggage items and they can be tracked and monitored individually by clicking a button on mobile application displaying the ID.

Also in the presence of a large number of luggage at an airport, each fitted with a device, the mobile application will be able to monitor and track only the luggage fitted with its device due to connectivity of both device microcontroller and mobile application with the same Cloud service. That would not be a problem at all.

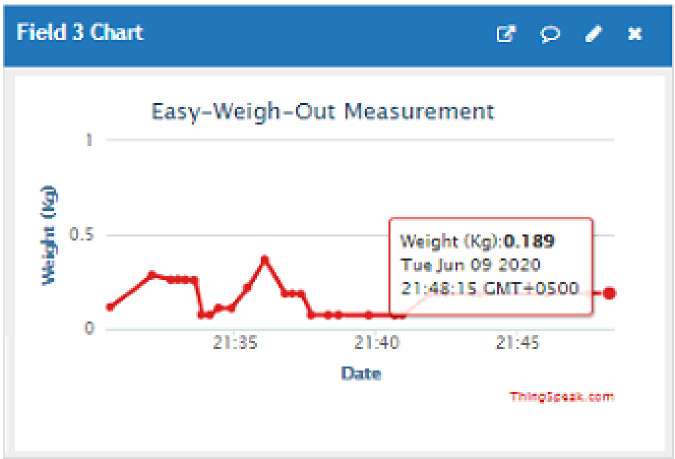


Fig. 5. A record of luggage weight on Cloud

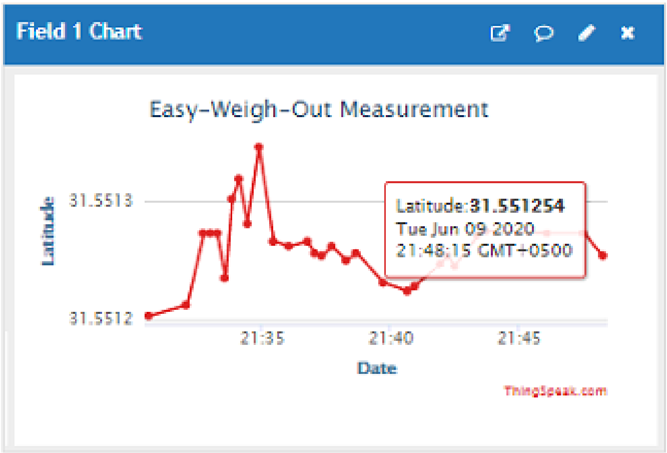


Fig. 6. A record of luggage location (Latitude) on Cloud

6 Conclusion

In a time when the world is being revolutionized by new technologies, IoT promises a huge prospect, especially in the domain of travel and tourism, since traveling has become a necessity. This paper presented an IoT-based system Easy-Weigh-Out for a smart traveling luggage that allows the user to weigh the luggage and track its location in real-time. This system is a stepping stone for many IoT applications, especially in the travel accessories market. It serves as a guideline for design and implementation of a variety of IoT-based applications or use-cases in other domains as well such

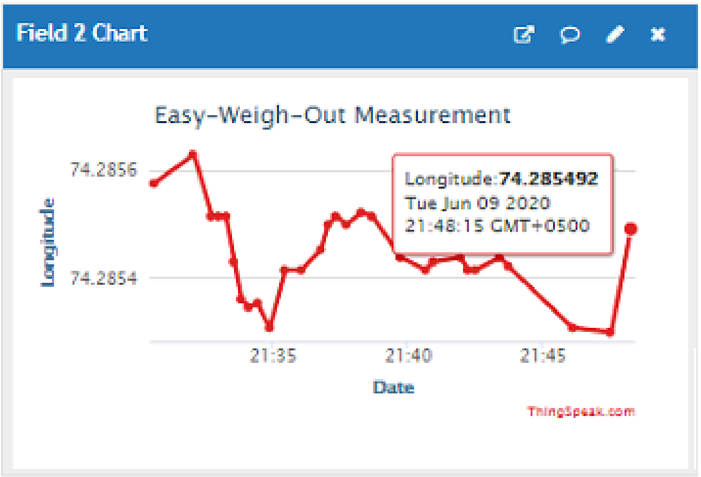


Fig. 7. A record of luggage location (Longitude) on Cloud

as healthcare, farming, water monitoring, security and surveillance etc [24-26]. This work can be extended to include more features such as alarms generation on emergency situation such as undue pressure on the luggage and inclusion of camera to report real-time footage of luggage carriers or handlers.

7 References

[1] Caron X., Bosua R., Maynard S.B., and Ahmad A. (2015). The Internet of Things (IoT) and its impact on individual privacy: An Australian perspective. Computer Law & Security Review, 32(1): 4-15. https://doi.org/10.1016/j.clsr.2015.12.001

[2] Ismail N. (2017). The Impact of the Internet of Things. [Online]. Available: https://www.information-age.com/impact-internet-things-iot-123467503/. [Accessed 9 May 2020].

[3] Stonefly. (2018). The Role of Cloud Computing in the IOT (Internet of Things). [Online]. Available: https://stonefly.com/blog/role-cloud-computing-internet-things. [Accessed 9 May 2020].

[4] Rouse M. (2020). Internet of Things (IoT). [Online]. Available: https://internetofthingsagenda.techtarget.com/definition/Internet-of-Things-IoT. [Accessed 9 May 2020].

[5] Dave N. (2018). 8 Ways in Which IoT is Shaping the Future of Travel Industry. Digital Doughnut [Online] Available https://www.digitaldoughnut.com/articles/2018/january/ways-in-which-iot-is-shaping-the-future-of-travel. [Accessed 9 May 2020]

[6] Mallon S. (2018). IoT Is the Most Important Development of the 21st Century. SmartDataCollective. [Online]. Available: https://www.smartdatacollective.com/iot-most-important-development-of-21st-century/. [Accessed 9 May 2020].

[7] Saddam. (2017). Arduino Weight Measurement using Load Cell and HX711 Module. CircuitDigest. [Online]. Available: https://circuitdigest.com/microcontroller-projects/arduino-weight-measurement-using-load-cell. [Accessed 9 May 2020].

[8] Sharma R. (2017). How to Interface GPS Module (NEO-6m) with Arduino. Arduino Project Hub. [Online]. Available: https://create.arduino.cc/projecthub/ruchir1674/how-to-interface-gps-module-neo-6m-with-arduino-8f90ad. [Accessed 9 May 2020]. https://doi.org/10.1007/978-1-4302-3886-7_5

[9] MathWorks. ThingSpeak. [Online]. Available: https://www.mathworks.com/help/thingspeak/. [Accessed 9 May 2020].

[10] Abhang A.L., Mahale C.L., Desai V.R., and Biswas P. (2018). Smart Bag. International Journal of Scientific Research in Science and Technology. 4(7):227-231.

[11] Sandi A.M., Chalwa S., Patil S., and Chidri S. (2019). Smart, secured and solace luggage bag using internet of things and computer vision. International Research Journal of Engineering and Technology. 6(6):344-347.

[12] Shweta M., Tanvi P., and Nilashree M. (2016). Multipurpose Smart Bag. Procedia Computer Science. 79: 77-84. https://doi.org/10.1016/j.procs.2016.03.011

[13] 1000 Projects. (2018). Smart Luggage System IoT Project. [Online]. Available: https://1000projects.org/smart-luggage-system-iot-project.html. [Accessed 9 May 2020].

[14] Senthikumar S., Brindha K., Rathi R., Charanya R., and Jain M. (2017). Luggage tracking system using IOT. International Journal of Pure and Applied Mathematics. 117(1): 49-55.

[15] Arduino. (2020). Introduction. [Online]. Available: https://www.arduino.cc/en/guide/introduction#:~:text=Arduino%20is%20an%20open%2Dsource,an%20LED%2C%20publishing%20something%20online

[16] Component101. (2018). Arduino-uno. [Online]. Available: https://components101.com/microcontrollers/arduino-uno

[17] Saddam. (2017). Arduino Weight Measurement using Load Cell and HX711 Module. [Online]. Available: https://circuitdigest.com/microcontroller-projects/arduino-weight-measurement-using-load-cell

[18] Saddam (2016). Track a vehicle on Google Maps using Arduino, ESP8266 & GPS. [Online]. Available: https://circuitdigest.com/microcontroller-projects/arduino-vehicle-tracker-on-google-maps-using-esp8266

[19] Sparkfun. (2020). WiFi Module - ESP8266. [Online]. Available: https://www.sparkfun.com/products/13678 [Accessed 10 June 2020].

[20] Aqeel A. (2018). Introduction to Arduino IDE. [Online]. Available: https://www.theengineeringprojects.com/2018/10/introduction-to-arduino-ide.html.

[21] Wikipedia. (2020). Thing Speak. [Online]. Available: https://en.wikipedia.org/wiki/ThingSpeak

[22] Wikipedia. (2020). Android Studio. [Online]. Available: https://en.wikipedia.org/wiki/Android_Studio

[23] Sampaio J. (2016). Introduction to Android Studio IDE. [Online]. Available: http://mrbool.com/introduction-to-android-studio-ide/34022. [Accessed 10 June 2020]. https://doi.org/10.1007/978-1-4842-5937-5_4

[24] Chihana S., and Phiri J. (2018). Application of IoT in Zambia's FRA Grain Traceability Process. Intenational Journal of Recent Contributions from Engineering, Science & IT. 6(4): 87-95. https://doi.org/10.3991/ijes.v6i4.9332

[25] Abdellatif M.M., and Mohamed W. (2020). Telemedicine: An IoT based Remote Healthcare System. International Journal of Online and Biomedical Engineering. 16(6): 72-81. https://doi.org/10.3991/ijoe.v16i06.13651

[26] Rahmadya B., Zaini Z., and Muharam M. (2020). IoT: A Mobile Application and Multi-hop Communication in Wireless Sensor Network for Water Monitoring. International Journal of Interactive Mobile Technologies. 14(11): 288-296. https://doi.org/10.3991/ijim.v14i11.13681

8 Authors

Mohammad Haasin Farooq is a graduate of electrical engineering from the National University of Computer and Emerging Sciences (FAST-NU), Lahore campus, Pakistan. He is an aspiring Android Developer with a keen interest in working with new technologies, particularly the Internet of Things. He has worked on a number of projects in the health department and, most recently, has published a blood donation mobile application on Google Play Store. Haasin hopes to continue his pursuit of knowledge by gaining a Master's degree in the field of Data Science and integrate new cutting-edge technologies, such as artificial intelligence, into his work.

Muhammad Zain is a graduate of electrical engineering from the National University of Computer and Emerging Sciences (FAST-NU), Lahore campus, Pakistan. He is a dedicated person who is skilled in the fields of software development and data science. He has worked on a number of projects related to Software Development, Machine Learning and Internet of Things. Recently he deployed a Loan Prediction System on a web application that helps a bank judge the eligibility of the submitted application using machine learning algorithms. Zain hopes to pursue a career where he can use technologies like artificial intelligence and IoT to make new software or gadgets.

Muhammad Bilal Khalid is a graduate of electrical engineering from the National University of Computer and Emerging Sciences (FAST-NU), Lahore, Pakistan. He is an enthusiastic electronics learner and keeps himself busy in acquiring the knowledge of growing fields in technology, particularly Embedded Systems and the Internet of Things. He has devoted most of his time in applying the modern concepts of embedded systems by working on a number of projects. Designing an embedded system for a Packaging Industry is one of his recent successful projects. Moreover, Bilal aims to serve the industry in the fields of Internet of Things and Embedded Systems with his hardworking nature and experience, and cope with the latest innovations in that domain.

Saima Zafar is an academic, engineering educator, researcher and Professor in the Department of Electrical Engineering at National University of Computer and Emerging Sciences (FAST NU), Lahore campus in Pakistan. She has published her scholarly work in reputed international journals including IEEE Access, Springer EURASIP JWCN, Springer EURASIP JIVP and Energies and is also a reviewer of prestigious international journals such as IEEE Systems journal, IEEE Internet of Things Journal, IEEE Access, JNCA, IJCS, KSII Transactions on Internet and Information Systems and Applied Computing and Informatics. She is a member of the departmental board of studies, graduate research committee and academic council of the university. Her research interests include wireless communications and networks, 5G cellular network, network protocol design, data center networking and Internet of Things. Email: saima.zafar@nu.edu.pk

Article submitted 2020-07-20. Resubmitted 2020-10-16. Final acceptance 2021-01-04. Final version published as submitted by the authors.