IoT Based Smart Parking System

Omnia Mohamed, Ahmed Abdelkarim, Ahmed Youssef, Hussein Ashraf, Al-Shimaa Essam and Sabry Atef
October 6 University, Egypt, Omn.Mohamed.201703677@o6u.edu.eg, ahmedabdelkareems19@gmail.com,

ahmedapostat@gmail.com, husseinashraf903@gmail.com, shimaaesamkk45@gmail.com, sabryatef148@gmail.com

Supervisor: Mahmoud M. Salim, Lecturer
October 6 University, Egypt, msalim.nw@gmail.com

***Abstract*– Smart city investments are playing an increasingly important role in enhancing cities' regional and global competitiveness to attract new residents and businesses. Providing an open data platform with access to city information, allow businesses to make informed decisions through data analytics from integrated smart city technologies. As an important component of traffic system, parking management system is playing an important role and affecting people's daily life. By detecting and processing the information from parking slots, smart parking system allows drivers to obtain real-time parking information and alleviates parking contentions. The use of intelligent systems has become the most prevalent among the world contribute to the implementation of daily business in a more efficient and flexible reveal. Due to the increase of the number of cars (clients), more parking slots are required with an efficient management system. Our smart parking system is supposed to be designed to solve the parking problem including the following features (User management and control, online payment for the users, automatic control for the parking gate and help users to find the nearest free space to park.**

***Keywords*-- Smart parking App, QR scanner, Raspberry pi, AEP.**

I. Introduction and Previous Work

In the past decades, we have seen the development of smart city technology, where the main goal of this technology is to provide a higher quality of life and create more effective and efficient cities. This became an essential requirement considering the rapid growth in urban populations. Advancements in the "big data" and in the Internet of Things (IOT) fields have allowed cities to access any information from any device anywhere therefore a well-designed data analytics strategy is required. These strategies need to access and analyze a massive amount of information and provide a clean meaningful, actionable output. If a city can monitor the desired metrics in real-time, the service levels will rise quickly. According to statistical the smart city transportation investments are expected to rise over 25% annually over the next five years. Creating a full vision, connected transportation systems around a city is expected to drastically enhance the efficiency and effectiveness throughout that city, which represents the main goal of smart city technology.

A. Previous Work

Caroline J. Rodier and et al in [1][2] introduced a report presents an evaluation of the first transit-based smart parking project in the U.S. at the San Francisco Bay Area Rapid Transit (BART) District station in Oakland, California.

Susan A. Shaheen and et al in [3] presented early findings from an application of advanced parking technologies to increase parking capacity at a transit station during the first half of 2004 in the San Francisco Bay Area, California (USA). Project partners include California Partners for Advanced Transit and Highways (PATH), the California Department of Transportation (Caltrans), the BART District, ParkingCarmaTM, and the Quixote Corporation.

Merriman in [4] and Ferguson in [5] mentioned that a review of the literature suggests that parking shortages at suburban rail stations may significantly constrain transit ridership. In addition, motorists may respond to pre-trip and en-route information on parking availability at transit stations by increasing their transit use.

Rodier et al in [6], mentioned that regular commuters appear to be more responsive to parking information in conjunction with transit than more basic PGI systems because this type of real-time information has greater relevance to their commute trip (e.g., transit station parking availability, next train information, and/or roadway accident downstream).

Glohr in [7], stated that in addition to providing realtime information about space availability and transit schedules, smart parking systems can take advantage of new technologies to improve the ease and convenience of parking payment.

Halleman in [8], Hodel and et al in [9], mentioned that combining the concepts of its forerunners, e-parking is an innovative business platform that allows drivers to inquire about parking availability, reserve a space, and even pay for parking upon departure—all from inside an individual's car. Drivers access the central system via cellular phone, personal digital assistant (PDA), and/or Internet.

Griffith in [10], introduced Smart parking management technologies that can provide a cost-effective way to address the parking requirements and constraints at parking lots. It can be defined as the use of advanced technologies to help motorists locate, reserve, and pay for parking.

B. Paper Contributions

In this paper, we present a number of technologies that constitute a complete IoT- based smart parking system, which represent our project contributions as follows*:

- A mobile application that is used to manage the parking procedures including user registration and login, monitoring free and busy slots, charging user account …etc.
- Application Enablement Platform (AEP) is used that provides all the foundational services (communication, data storage, management, application building and enablement, user

* This work was supported by the ITIDA-ITAC graduation projects funding program with grant number GP2021.R16.104. Also, this project was installed realistically at the main parking area of October 6 University.

interface security, and analytics) required to build a solution from the ground up. We use Master of Things as our AEP.

- At each parking slot, a NodeMCU with its IR sensor is installed to detect the slot status either idle or busy and report this information the user though the mobile application.
- Raspberry pi zero W is installed at the main gate of the parking area to scan the user's QR Code using its camera module to verify user's data with the aid of the AEP. An OpenCV software is used to decode the QR code to extract its information. After receiving the user's data, AEP checks user's credit and access to the parking area. If the user is authenticated, the gate automatically opens through the aid of a connected relay and NodeMCU then gets its instructions from the AEP.
- To get out from parking user must scan QR code again, then AEP calculates the time of entry, exit and calculating the expected cost and automatically opening gate.

C. Paper Organization

The rest of this paper is organized as follows. Section II explains the Need for the IoT-Cloud Integration showing introduction to IoT, what is IoT, why IoT matters and how system actually works. In Section III, System Architecture is illustrated by introducing our project outlines,, determining devices used from microcontrollers and sensors and learning how to programming them together to achieve our purpose. Besides, the way to connect devices to the used platform is described so we present the best way to connect them by the MQTT protocol. In addition, we explain how to set our broker, MoT interface, user interface, and developer interface. In addition, we present how to create database for sensor readings and the used application. In section IV, the project implementations and working are depicted. Finally, the conclusions of this paper are presented in Section V.

II. Need for IoT-Cloud Integration

The applications for IoT are extended across a broad variety of use cases and verticals. However, all complete IoT systems are the same in that they represent the integration of four distinct components: sensors/devices, connectivity, data processing, and a user interface. We outline the meaning of each one below and how they integrate together to form a complete IoT system.

1) Sensors/Devices: First, sensors or devices collect data from their environment. This data could be as simple as a temperature reading or as complex as a full video feed. We use "sensors/devices," because multiple sensors can be bundled together or sensors can be part of a device that does more than just sense things. For example, your phone is a device that has multiple sensors (camera, accelerometer, GPS, etc.), but your phone is not just a sensor since it can perform many actions. However, whether it's a standalone sensor or a full device, in this first step data is being collected from the environment by something.

2) Connectivity: Next, that data is sent to the cloud, but it needs a way to get there! The sensors/devices can be connected to the cloud through a variety of methods including: cellular, satellite, Wi-Fi, Bluetooth, low-power wide-area networks (LPWAN), connecting via a gateway/router or connecting directly to the internet via ethernet. Each option has tradeoffs between power consumption, range, and bandwidth. Choosing which connectivity option is best comes down to the specific IoT application, but they all accomplish the same task: getting data to the cloud.

3) Data Processing: Once the data gets to the cloud (we'll cover what the cloud means in a later section,) software performs some kind of processing on it. This could be very simple, such as checking that the temperature reading is within an acceptable range. Or it could also be very complex, such as using computer vision on video to identify objects (such as intruders on a property). But what happens when the temperature is too high or if there is an intruder on property? That's where the user comes in.

4) User Interface: Next, the information is made useful to the end-user in some way. This could be via an alert to the user (email, text, notification, etc.). For example, a text alert when the temperature is too high in the company's cold storage. A user might have an interface that allows them to proactively check in on the system. For example, a user might want to check the video feeds on various properties via a phone app or a web browser. However, it's not always a one-way street. Depending on the IoT application, the user may also be able to perform an action and affect the system. For example, the user might remotely adjust the temperature in the cold storage via an app on their phone. And some actions are performed automatically. Rather than waiting for you to adjust the temperature, the system could do it automatically via predefined rules. Rather than just call you to alert you of an intruder, the IoT system could also automatically notify security teams or relevant authorities [11].

III. System Architecture

Fig. 1 describes how does our system actually work as follows. When a user wishes to park, it shows its QR code to the scanner, which gets the user's information that is published to the AEP. The AEP will inform the user with the free slot

* This work was supported by the ITIDA-ITAC graduation projects funding program with grant number GP2021.R16.104. Also, this project was installed realistically at the main parking area of October 6 University.

number and the gate is automatically opened by a controlled relay. Access online payment, calculating the time of entry and exit and calculating the expected cost. To get out from parking scan your QR code again, the gate is automatically opened by a controlled relay.

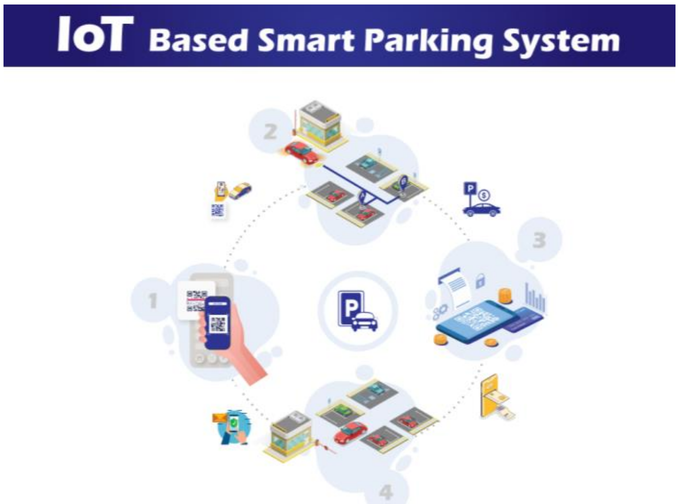

Fig. 1 Smart Parking System Architecture.

Our Smart Parking system is made up of interchangeable components and fully integrates parking, guidance, payment and analytics as well as a host of other complementary services and options. This makes the complete Smart Parking solution truly a sum of its parts, and an industry leader in the technology and parking sector.

- *Parking Sensors:* for our parking system to scan QR Code we use Raspberry pi zero W with its camera module for detecting and decoding QR Code then send its data to Application Enablement platform [12][13], then for automatic opening gate we use relay with NodeMCU that has ESP-12E Wi-Fi module to connect platform, finally for detecting free slots we use LASER sensor with NodeMCU.
- *Processing Units:* we use Raspberry pi zero W that known as a single-board computer, which means exactly what it sounds like: it's a computer, just like a desktop, laptop, or smartphone, but built on a single printed circuit board. Like most single-board computers, Raspberry Pi is small but that doesn't mean it's not powerful: a Raspberry Pi can do anything a bigger and more power-hungry computer can do, though not necessarily as quickly. Raspberry pi specifications as CPU (1-GHZ, Broadcom BCM2835), RAM (512 MB), Wireless (802.11n / Bluetooth 4.1/ LE), Ports (Micro USB, mini-HDMI), I/O (40 GPIO Pins, CSI camera connector). These allow a massive range of sensors, motors, LEDs and accessories to be connected to the Pi [14]. We use also NodeMCU The development board equips the ESP-12E module containing ESP8266 chip having 32-bit LX106 RISC microprocessor. operates at 80 to 160 MHz adjustable clock frequency and supports RTOS. There's 128 KB RAM and 4MB of Flash memory (for program and data storage) just enough to cope with the large strings that make up web pages, JSON/XML data, and everything we throw at IoT devices nowadays. The ESP8266 Integrates 802.11b/g/n HT40 Wi-Fi transceiver, so it can not only connect to a Wi-Fi network and interact with the Internet, but it can also set up a network of its own, allowing other devices to connect directly to it. This makes the ESP8266 NodeMCU even more versatile [15].

- *The cloud:* we use MOT MQTT (Message Queuing Telemetry Transport) protocol. It is a lightweight messaging protocol that uses the publish/subscribe method and translates messages between multiple devices. Using MQTT protocol, we can also send/receive data and control various output devices, like read sensor data. It's developed on top of TCP, which is why it's faster than similar protocols like HTTP. Other than that, it has many other advantages over other protocols like its very lightweight, so it doesn't consume excess memory, it can work with very less network bandwidth, on top of that, it has a robust security protocol inbuilt. These features make it suitable for many applications. In order to understand the working of the MQTT protocol, we just need to understand three basic things:
  1) MQTT Client: An MQTT client is any device (it can be a microcontroller or a server) that runs MQTT functions and communicates with a central server, which is known as the "broker." The broker handles the data communication between the connected clients.
  2) MQTT Publisher: When a client wants to send any information, the client is known as a "Publisher". The publisher will publish the information on a particular topic. "Topic" is a path where we can publish/subscribe messages. The broker then sends the information published by the user to the clients (also known as Subscriber) that have subscribed to that specific topic.
  3) MQTT Subscriber: The MQTT Subscriber subscribes to topics on an MQTT broker to read the messages sent by the broker [16].
- Application Enablement Platform: An IoT Application Enablement platform AEP is not an app or simply means of managing assets (device

* This work was supported by the ITIDA-ITAC graduation projects funding program with grant number GP2021.R16.104. Also, this project was installed realistically at the main parking area of October 6 University.

clouds), it is meant to provide all *the* foundational services (communication; data storage; management; application building and enablement; user interface security; and analytics) required to build a solution from the ground up. It is worth noting that Master of Things (MOT) platform is the used AEP.

III. Implementation & Working

In the previous section we discussed the architecture and technical stack related to smart parking system. In this section, we talk about the implementation and working in real world scenario, which was realistically implemented at the main parking area of October 6 University, Giza, Egypt. The complete process, which consists of entry and exit subprocesses, is explained with the following flow chart:

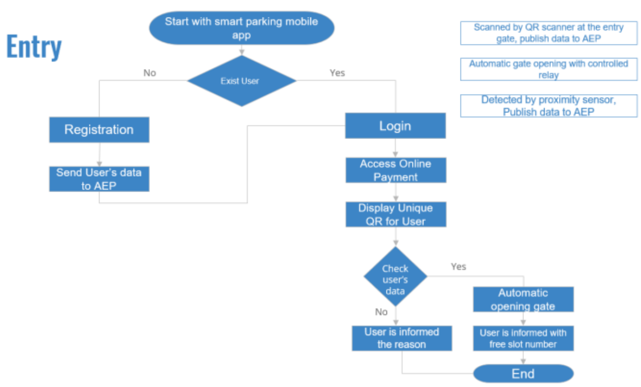

Fig. 2 System Flow Chart for Entry.

- **Step 1**: Registration into smart parking system mobile application as shown in Fig 3.

Fig 3. Registration Page

- **Step 2**: Logging to get the home page as shown in Fig 4.

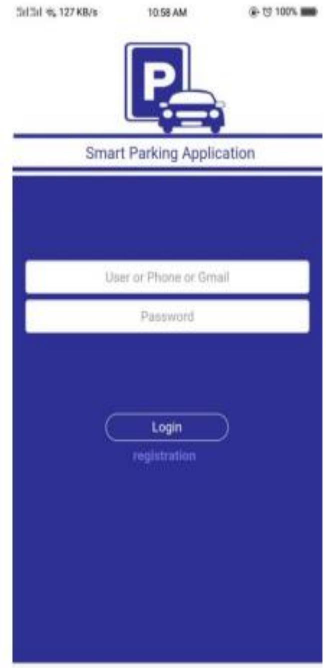

Fig 4. Login page

Now you can get all application features as shown in Fig 5.

Fig 5. Home page

- **Step 3**: Paying the parking charges using user's wallet as shown in Fig 6.

* This work was supported by the ITIDA-ITAC graduation projects funding program with grant number GP2021.R16.104. Also, this project was installed realistically at the main parking area of October 6 University.

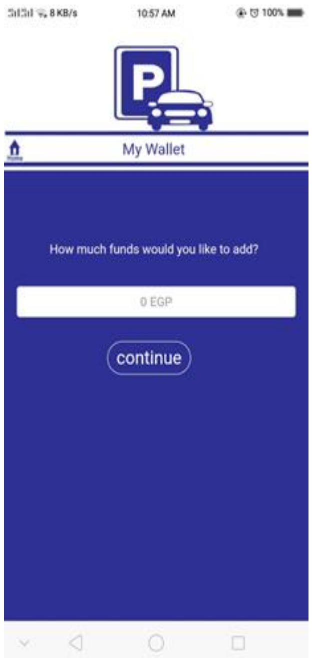

Fig 6. User Wallet

- **Step 4**: Displaying user’s unique QR Code as shown in Fig 7.

Fig 7. QR Code for User

- **Step 5**: Scanning user’s QR Code at the entry gate for automatic opening gate.
- **Step 6**: Displaying the nearest free slot as shown in Fig 8.

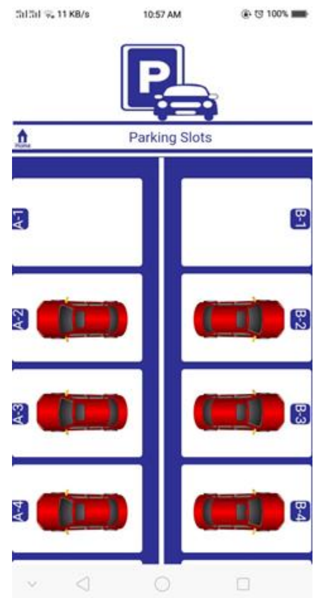

Fig 8. Parking Slot Page

To get out from parking you have to follow these steps:

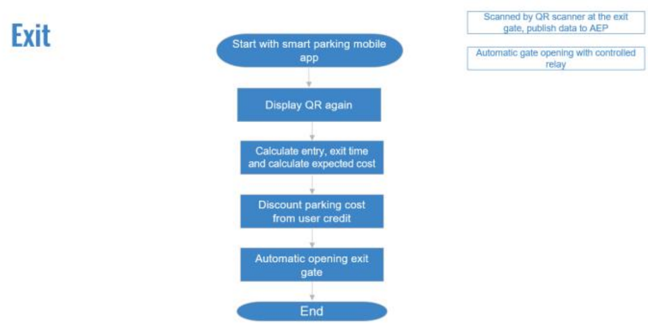

Fig 9. System Flow Chart for Exit.

- **Step 1:** Displaying user’s QR Code again.
- **Step 2:** Scanning the QR Code at exit gate to discount parking cost then automatic opening gate.

V. CONCLUSIONS

The services provided by smart parking have become the essence of building smart cities. This Report focuses on implementing an integrated solution for smart parking. The proposed system has several advantages including automatically opening gate, accessing online payment, detecting parking spaces using the IoT technology, calculating the time of entry and exit and calculating the expected cost. Besides, an attractive and effective application was designed for iOS and Android mobile phones to manage the system. The main system benefits are the reducing of time wasted for conventional parking, pollution, and fuel consumption.

ACKNOWLEDGMENT

* This work was supported by the ITIDA-ITAC graduation projects funding program with grant number GP2021.R16.104. Also, this project was installed realistically at the main parking area of October 6 University.

First of all, praise is to Allah who has guided us to this success. Special thanks to our supervisor Dr. Mahmoud Salim who hasn't hesitate to help us someday. We will not commend his science and knowledge as they are in needless to this, but we will commend the scientific ethics, academic skills, accuracy in work, and respect for time we have learnt from them. we are proud to work under his supervision, so we say thank you very much to him. Many thanks to SpimeSenseLab company for their continued support and for providing us the free access to their Master of Things platform. We are grateful for the funding given by the Egyptian government represented by ITAC – Information Technology Academia Collaboration graduation project funding program.

REFERENCES

[1] C. J. Rodier, S. A. Shaheen, and C. Kemmerer, "Smart Parking Management Field Test: A Bay Area Rapid Transit (BART) District Parking Demonstration", UCB-ITS-PRR-2008.

[2] S. A. Shaheen, C. J. Rodier and A. M. Eaken, "Applying integrated its technologies to parking management systems: A transit-based case study in the San Francisco Bay area", UCD-ITS-RR-04-18, July 2004.

[3] Chinmayee Lingam. Graduate project report, Faculty of the Department of Computing Sciences Texas A&M University-Corpus Christi Corpus Christi, Texas, 2018.

[4] D. Merriman, "How many parking spaces does it take to create one additional transit passenger?", Regional Science and Urban Economics vol. 28, no. 5, pp. 565–584, 1998.

[5] E. Ferguson, "Parking management and commuter rail: the case of Northeastern Illinois", Journal of Public Transportation, vol. 3, no. 2, pp. 99–12, 2000.

[6] C. Rodier, S. Shaheen and A. Eaken, "Transit-Based Smart Parking in the San Francisco Bay Area: An Assessment of User Demand and Behavioral Effects", Journal of Transportation Research Record, vol. 1927, no. 1, pp. 167—173, 2005.

[7] E. Glohr, "Lansing community college parking smart card trailblazer". The Parking Professional Journal, June, 2002.

[8] B. Halleman, "Europe's space program (parking space, naturally)", Traffic Technology International, February/March, pp. 46–49., 2003.

[9] T.B. Hodel and S. Cong, "Parking space optimization services, a uniformed web application architecture", proceedings of Intelligent Transport Systems and Services (ITS), Madrid, Spain, 2003.

[10] E. Griffith, "Pointing the Way", ITS International, March/April, 2000, pp. 72.

[11] https://www.scribd.com/document/415553860/Leverege-Intro-to-IoT-eBook.

[12] https://www.hackster.io/gatoninja236/scan-qr-codes-in-real-time-with-raspberry-pi-a5268b.

[13] https://peppe8o.com/using-raspberry-pi-with-dht11-temperature-and-humidity-sensor-and-python/.

[14] https://projects.raspberrypi.org/ar-SA/projects/raspberry-pi-setting-up/4.

[15] https://lastminuteengineers.com/esp8266-nodemcu-arduino-tutorial/.

[16] https://circuitdigest.com/microcontroller-projects/performingmqtt-communication-with-esp8266-nodemcu-using-arduino-ide.

* This work was supported by the ITIDA-ITAC graduation projects funding program with grant number GP2021.R16.104. Also, this project was installed realistically at the main parking area of October 6 University.