

ELEN4012 PROJECT PLAN - PROTOTYPING OF AN IOT BASED PARKING SYSTEM FOR A SMART CAMPUS

J Ping (704447) & L Timm (704157)
Group 18G11

School of Electrical & Information Engineering, University of the Witwatersrand, Private Bag 3, 2050, Johannesburg, South Africa

Abstract: This report outlines a project plan to implement an Internet of Things (IoT) based smart parking system to alert drivers of available parking bays. The project plan uses Arduino nano and ESP-8266 development boards, ultrasonic sensors, Radio Frequency (RF) communication, the ThingSpeak cloud IoT platform and the Android framework in an attempt to give the project the best chance of success. Testing methodologies have been outlined to quantify the successful operation of the system. Research into IoT hardware, communication technologies, deployment methodologies, data management and data presentation is conducted and analysed. By considering all possible design configurations and flaws that could be encountered, an appropriate and efficient project design is proposed. The project, from design to demonstration, should take 6 weeks.

Key words: Android, Arduino, ESP-8266, Internet of Things, RESTful API, ThingSpeak, Wireless Sensor Network

1 INTRODUCTION

The digital age has seen a momentous increase in the interest of digitizing all aspects of life. The Internet of Things (IoT) is a large range of systems, described by the International Telecommunication Union (ITU) as "a global infrastructure for the Information Society", which makes use of information and communication technologies to interconnect things [1]. An IoT-based parking monitoring system aims to allow parking availability tracking in real time via a mobile application, building towards transforming the University of the Witwatersrand into a Smart Campus. Contained within the remainder of this report is a project plan to design, construct and test a Smart Parking system.

2 BACKGROUND AND CONTEXTUALISATION

The purpose of this section is to demonstrate techniques to identify and communicate parking bay availability. This process includes data acquisition and processing using a sensor module, the deployment of the sensor module, wireless data communication, data management and data presentation. For a minimum viable product, each sensor module must consist of a sensor, microcontroller (MCU), communication module and power supply.

2.1 Hardware Choices

The available and preferred hardware options for the sensor module are considered in the sections that follow.

2.1.1 Microcontroller

When choosing an MCU for an IoT system it is important to consider a number of factors. These include the MCU's power consumption, number and availability of I/O ports, how the MCU will interface with the sensors and how the sensor modules will communicate with one another. Additional considerations are the ease of use of the MCU as well as its size and expense.

A popular, versatile and relatively inexpensive MCU platform is the Arduino Platform. Owing to their small size and reasonable price, the *Arduino Nano V3* and *Arduino Pro Mini* development boards are considered. Arduino de-

vices are well documented with a wide array of libraries, provided through extensive community support, which support a range of inexpensive sensor and communication modules. The *Nano* is advantageous over the *Pro Mini* due to its ease of programming, greater availability and lower current draw during normal operation. Greater power consumption in the *Pro Mini* occurs due to an inefficient on-board power regulator integrated circuit (IC).

An alternative to the Arduino platform is the ESP8266-based development board range. These boards are designed to cater for low consumption requirements, such as the bare-bones ESP8266 MCU, through to intensive processing requirements, such as the dual core ESP32 MCU. The development boards come pre-built with a Wi-Fi module, among other communication modules, which negates the need for additional communication hardware [2].

2.1.2 Sensor

The sensor chosen to detect vehicle presence in an open air parking lot should be tolerant of the environmental conditions in which it must operate. The sensor module will be exposed to large amounts of ambient light, varying light conditions, fluctuating temperatures and rain. As such, the chosen sensor should not rely on light to detect vehicles, should not be significantly affected by changing temperatures and should be weatherproof. Considering also that the cost per sensor module should be as low as possible, connecting multiple sensors to a single MCU is a reasonable design choice.

This specification suggests an accurate distance/proximity measurement sensor with a reasonable range (0.5 - 2 m). Examples of such sensors include microwave radar, RFID, passive infrared and ultrasonic sensors [3]. An ultrasonic distance measurement sensor is preferred for this application due to its high accuracy and ability to set a user-defined range.

2.1.3 Communication

The method of communication between the sensor nodes is a key design consideration as many options are available,

each with their own benefits and trade-offs. A range of communication modules exist which are compatible with most common MCUs. These include Wi-Fi, RF, Bluetooth and NFC communication standards. The chosen mode of communication should be robust and reliable as to prevent data loss during transmission. Furthermore, the communication modules should be able to endure a range of outdoor weather conditions as well as obstructions which may exist between two modules.

The solution should be designed to cater for a system that can be easily and efficiently scaled. This requires the abundant availability of channels within the chosen communication scheme to prevent bandwidth throttling within the sensor network. It is pivotal that a power efficient communication scheme is utilised. Power consumption is proportional to transmission range, and a communication module with variable power modes would be ideal for a scalable and power efficient solution. Enabling line of sight between nodes would require less power as transmission does not need to occur through any obstacles.

2.1.4 Power Supply

The sensor network should be portable and self-sufficient, requiring a power scheme that is independent. The sensor network requires supply modules which are well regulated and offer large capacity. Additionally, the modules should be renewable and affordable. Rechargeable batteries cater for this exact purpose.

A range of rechargeable batteries exist which include lithium ion (Li-ion), lithium polymer (Li-Po) and nickel metal hydride (Ni-Mh) options. Li-ion batteries generally offer high capacity in 3.7 V multiples but are known to be quite dangerous when exposed to high temperature ranges [4]. Li-Po batteries offer a large range of capacities in 3.7 V multiples but the price increases exponentially as capacity increases [5]. Ni-Mh batteries are versatile in offering a large range of capacities as well as a range of voltages [6].

The chosen supply will need to be regulated to ensure stable operation of the sensor nodes, as any fluctuation in supply may damage or prevent the sensor nodes from functioning correctly. To ensure the supplied voltage is at an expected level, the use of a boost converter could be utilised to step up the default supply voltage of the batteries.

2.2 Communication Technologies

The Industrial, Scientific and Medical (ISM) band refers to a group of radio bands that are unlicensed and internationally reserved for the use of RF transmissions. Any communication operating within the ISM band should be tolerant of interference from other ISM equipment. There has been a surge in low-power, short-range communication platforms utilising the ISM band such as Bluetooth, Wi-Fi, NFC and mobile networks [7]. Thus a large variety of communication methods are available offering a range of power consumptions, transmission ranges and data rates.

A variety of application-appropriate communication technologies are available to communicate between sensor nodes. These are discussed in the sections that follow.

2.2.1 Wi-Fi

This communication scheme is well known and easily accessible for use within the IoT market. It's range will vary depending on a number of factors, including the specific IEEE 802.11 protocol which the module is utilising, the strength of the module's transmitter, and radio interference from the surrounding environment [8]. Low energy consumption modules require between 14 dBm (25 mW) to 19 dBm (79 mW) for typical transmission with a data rate of 72 Mbps [2]. At maximum transmission power in an open environment, a maximum transmission distance of approximately 100 m is achieved [2]. The ESP8266 Wi-Fi module allows four simultaneous client connections with unidirectional communication [2].

2.2.2 Bluetooth

Bluetooth is another well known communication scheme which is available within the IoT market. Bluetooth utilises the IEEE 802.15.1 standard. Multiple variants exist which offer differing combinations of low energy consumption, high data rates and long transmission distances [9]. Low energy consumption modules require a maximum of 3 dBm (2 mW) for a typical transmission, having a maximum range of approximately 150 m and a data rate of 0.23 Mbps [10]. The long range variant offers a maximum transmission distance of approximately 450 m at a data rate of 0.23 Mbps, and requires 8 dBm (6 mW) to transmit [11]. The number of simultaneous connections allowed varies per device, with the maximum amount being seven, with unidirectional communication [12].

2.2.3 Radio Frequency

A range of RF-based transceivers exist which utilise different ISM bands. The NRF24 refers to a protocol used by Nordic Semiconductors which utilises the 2.4 GHz ISM band for RF communication [13]. Nordic has produced a range of NRF24 based transceivers which have become prominent within the IoT market due to their low cost and adjustable power consumption. The nRF24L01+ transceiver, for example, offers a maximum transmission distance of approximately 100 m with a data rate of 0.25 Mbps and a transmission power requirement of 0 dBm (1 mW) [13]. Furthermore, the data rate can be configured up to 2 Mbps and transmission power modes can be selected ranging down to -18 dBm (16 μ W) [13]. These transceivers facilitate 125 simultaneous connections with bidirectional communication [13].

An alternative is the RFM69 transceiver, which offers similar operating characteristics to the NRF24-based transceivers but operates in the 433/868/915 Mhz ISM bands and supports up to 255 simultaneous connections [14]. Naturally, the RFM69 transceiver exhibits marginally higher power requirements to sustain these ca-

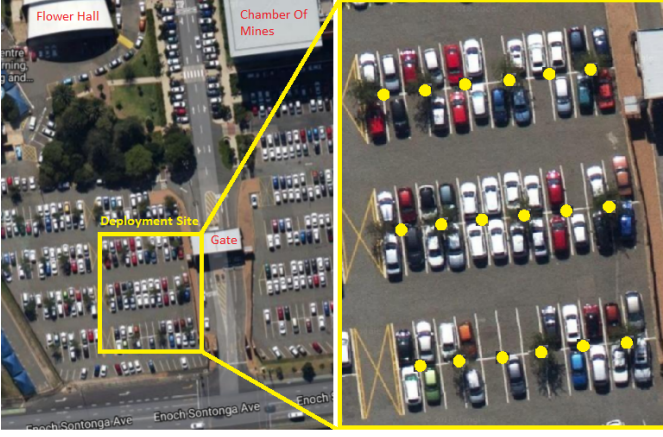


Figure 1: Smart Parking deployment site (Google Maps satellite view) and module placement map

pabilities and is more expensive than the NRF24-based transceivers [14].

2.3 Deployment Methodologies

The parking system is to be deployed in the *3rd, 4th and Postgraduate* parking lot near the Flower Hall building, located on West Campus of Wits University. The designed system covers 72 parking spaces, with four sensors being monitored by each MCU. As such, 18 hardware modules need to be packaged and deployed. *Figure 1* shows the planned deployment layout. Considerations have been made for the module rigidity, resistance to environmental conditions, security and positioning.

When deployed, the hardware should be raised from ground level, not only to prevent the circuitry from being damaged by excessive water runoff, but also to provide the ultrasonic sensors with the best viewing angle; ensuring that all shapes and sizes of vehicles can be detected. The use of an ultrasonic sensor provides challenges regarding waterproofing, as a result of ultrasonic waves being impermeable to almost all materials. As such, there cannot be any barrier between the sensors and the cars being detected, requiring a runoff system that will keep this circuitry dry.

Raising the hardware requires that the sensor nodes are resilient to strong winds, and should not break if they are bumped by vehicles driving into or between parking spaces. In an attempt to keep costs to a minimum, a PVC stand is proposed that will be rigid, stable and provide the right topology to detect vehicles at average bumper height. A top and isometric view of the proposed stand is provided in *Figure 2*. Considerations must be made for the security of the modules once they are deployed.

2.4 Data Management

Optimised collection and processing of sensor data is required to provide the relevant parking bay information to users. These components are described as in the sections that follow.

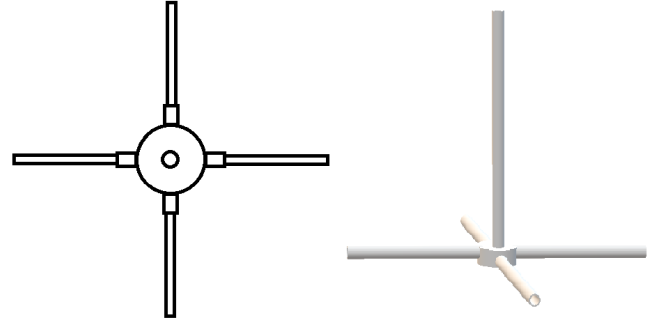


Figure 2: Top (left) and isometric (right) view of the proposed PVC stand

2.4.1 Mesh Network

The concept of a mesh network relates to a network topology in which infrastructure nodes, being the sensor nodes, are connected in a range of configurations to allow for the efficient routing of data between the nodes within the network [15]. In this specific scenario, the goal relates to implementing a network topology whereby all sensor data is successfully transmitted to the gateway node within the network. Furthermore, the mesh network should be designed in a way that facilitates the ability to enable self-configuration of the nodes within the topology. This allows for dynamic distribution of workloads, which is important in the scenario where a node may happen to fail [15]. This results in improved fault-tolerance and minimised system maintenance. The major considerations of the topology design in the context of this project are node transmission distance and node power consumption. Two topologies offer varying combinations of these factors.

Figure 3 illustrates a topology whereby a node will always transmit to its adjacent node until the gateway is reached. This results in a topology which yields a low transmission distance requirement but a high power requirement due to the amount of time which nodes must be powered as well as a large total number of transmissions per round of data gathering.

Figure 4 illustrates a topology whereby nodes are bound to segments and within each segment all nodes in a row will transmit to a single node. The receiving nodes will then transmit to the adjacent segment and the receiving node will transmit to the gateway node via the closest row node. This results in a topology which yields a longer transmission distance requirement but a lower power requirement due to the reduced time which nodes must be powered as well as a lower total number of transmissions for per round of data gathering.

2.4.2 Data Upload

The following communication methods are considered in uploading sensor data to the utilised data platform.

GSM/GPRS: Global System for Mobile (GSM) communication, also commonly referred to as 2G, due to be-

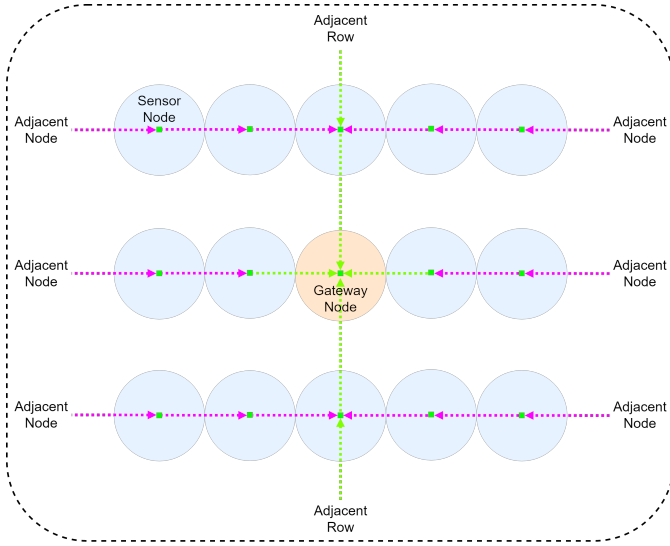


Figure 3: Node mesh network which utilises single node hop communication for short range transmission

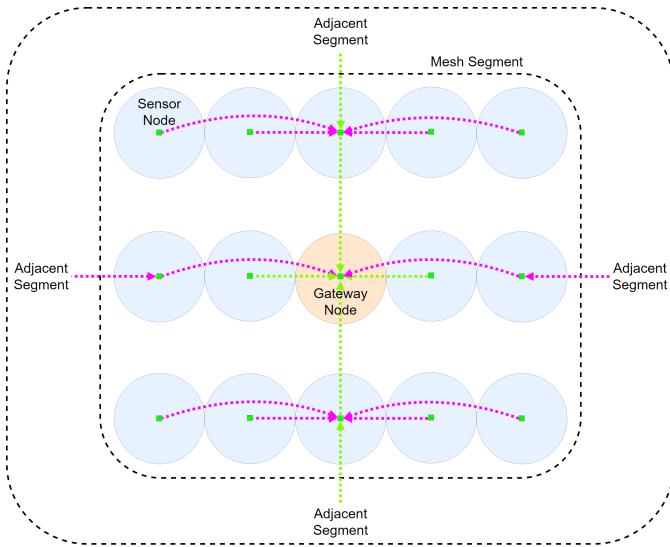


Figure 4: Node mesh network which utilises node-skip communication in a segmented mesh configuration for longer range transmission with fewer hops

ing a second-generation cellular network provides support for voice calls, Simple Message System (SMS) and data communication via General Packet Radio Service (GPRS). GPRS is a packet switching technology which can provide idealised data rates between 0.056 Mbps to 0.114 Mbps [16]. To utilise GPRS for data communication, an Access Point Name (APN) and username/password pair is required from a network operator [16].

GSM modules are commercially available and are compatible with most major development boards. These modules require a Subscriber Identification Module (SIM) which is an integrated circuit with the purpose of securely storing the International Mobile Subscriber Identity (IMSI) number with its corresponding authentication key [17]. These are used to identify and authenticate subscribers on the

mobile network. Such an example of a GSM module is the common Quectel M10 module which is a quad-band GSM/GPRS modem requiring a supply of just 3.4 V [18].

Wits Wi-Fi: The University of the Witwatersrand (Wits) offers a high speed network, offering local and international data rates of 10 Gbps and 589 Mbps respectively [19]. Due to the constant need for connection on the main campus, the network is extremely reliable with minimal down-time.

The use of a development board such as the ESP12-E, which features a built in Wi-Fi module, could easily be used as a gateway node for the sensor network. This module could periodically collect all respective sensor data and thereafter connect to the Wi-Fi network provided by Wits to upload the sensor data to the chosen data platform [2].

4G/LTE: This is the fourth generation of broadband cellular network technology, succeeding 3G. This generation places emphasis on data rates. This is demonstrated in modules such as the Quectel EC25 Long-Term Evolution (LTE) category 4 module which offers download and upload data rates of up to 150 Mbps and 50 Mbps respectively, requiring a supply of 5 V [20]. This caters for solutions where large amounts of data needs to be transferred continuously.

2.4.3 Data Platform

AWS Greengrass: Amazon Web Services (AWS) offers Greengrass which is software that facilitates the running of local compute, messaging, data caching, synchronisation, and machine learning capabilities for connected IoT devices in a secure manner. It provides useful off-line capabilities by still allowing devices to run AWS Lambda functions, synchronise device data, and communicate with other devices [21]. Greengrass is network optimised, ensuring the cost of transmitting data to the cloud is kept to a minimum [21].

Greengrass extends AWS to IoT devices, allowing them to act on generated data, whilst providing the cloud platform for management, analytics, and storage of the device data. A number of programming languages are available for use. Furthermore, a programming model is available which facilitates the creation and testing of device software in the cloud, followed by deployment to connected devices [21]. Data filtering can be effected so that only key information is extracted and transmitted to the cloud. Greengrass continuously authenticates and encrypts all device data throughout the connection process by utilising the Message Queuing Telemetry Transport (MQTT) protocol [21]. End-to-end security is effected using asymmetric key authentication over Transport Layer Security (TLS) 1.2 [21]. A maximum of 200 devices are supported per group, and unlimited per an account, with a maximum message size of 128 KB per device transmission to the cloud, and 30 messages being allowed per second [22].

Cloud IoT: Google Cloud Platform (GCP) offers Cloud IoT which is a fully managed suite of integrated services

which facilitate convenient and secure connection, device management and global data collection at a large scale. The collected data can be processed, analysed and visualised in real time and effect any required events as a result. Data is collected by the Cloud IoT Core service which is published to the Cloud Pub/Sub service to provide downstream analytics [23]. Ad-hoc analysis can be performed by utilising the Google BigQuery service [23]. Advanced analytics and machine learning can be performed with the Cloud Machine Learning Engine [23]. Data visualisation can be performed with reports and dashboards in Google Data Studio [23].

The Cloud IoT suite provides insights into device operational efficiency, device management and firmware updates. Out-of-box support is provided for leading manufacturers such as Microchip and Arm. Global device management is efficiently managed using Cloud IoT Core's protocol bridge with automatic load balancing and horizontal scaling [23]. Both MQTT and Hypertext Protocol (HTTP) protocols are offered to cater for the configuration of device connections to the cloud [24]. End-to-end security is effected using asymmetric key authentication over TLS 1.2 [24]. There is no limit on the maximum number of connected devices, the maximum message size is limited to 256 KB per device transmission and 100 messages transmissions are allowed per second [25]. Maximum download and upload data rates are limited to 5 Mbps per device [25].

ThingSpeak: MathWorks offers ThingSpeak which is an IoT analytics platform service which facilitates the collection, aggregation, analysis and visualisation of live data streams in the cloud [26]. MATLAB code can be executed in ThingSpeak to perform online processing and analysis of data from devices [26]. Both MQTT and HTTP protocols are supported by the platform [26]. Analytics can be scheduled to be run periodically or in real-time [26]. Third-party services such as Twitter and Twilio can be utilised to communicate device events [26]. Native support is provided for common embedded hardware prototyping platforms like Arduino, ESP-8266, Particle and Raspberry Pi [26].

A limit exists on the number of simultaneously connected devices depending on the selected pricing tier. There exists maximum message size limit of 3 KB per device transmission [27]. The free pricing tier offers a single channel transmission every 15 seconds with a maximum of three gateway devices simultaneously connected [27]. The paid tier offers single channel transmission every second with an unlimited number of simultaneously connected devices [27].

SigFox: An alternative IoT network is offered by SigFox whereby it facilitates the broadcasting of device data without the requirement of establishing or maintaining the network connection to do so [28]. This is designed to reduce signal overhead and offer a compact and optimised connection protocol [29]. This is implemented through the use of Ultra Narrow Band (UNB) modulation [28]. SigFox operates within the 200 KHz public band [28]. A RESTful API is used to provide the uploaded data to a platform or

application of choice [29].

A SigFox Software Defined Radio (SDR) module is required to interface with the network and can be obtained as development board shields which are compatible with platforms such as Arduino [28]. The maximum upload and download data rate ranges between 100 bps and 600 bps and incurs a maximum message size limit of 12 B which can be uploaded a maximum of six times per hour [28].

2.5 Data Presentation

Parking bay availability is communicated to the user through data presentation. The components of this process are described as in the sections that follow.

2.5.1 API

Parking bay data needs to be accessible from the data platform which has been configured to receive and process the sensor data from the network. This can be securely and easily implemented through the use of an Application Programming Interface (API). Each data platform offers a preconfigured API to access the processed cloud data through HTTP requests. These requests can be sent from any internet application, which is capable of executing the requests, to obtain the processed data and parse it as required to present information on parking bay availability to the user.

2.5.2 User Application

Users require an easily accessible network-based application to access the processed data provided by the data platform. This can be provided through a range of web or mobile applications. The adoption of a web application would prove convenient for a developer due to the vast range of web application frameworks that provide out-of-box API integration and a large range of capabilities for creating a well designed user interface/user experience (UI/UX). The adoption of a mobile application provides fewer design components to be customised for the user however, it provides portability which would allow for a user to easily check their mobile device for an empty parking bay as they are about to enter the parking lot.

3 PROJECT DESIGN

3.1 Hardware Choices

For the sensor nodes a combination of hardware choices have been made. The chosen MCU, an Arduino Nano V3, is powered by two rechargeable AA 1.2 V 3800 mAh Ni-Mh batteries in series, boosted to 5 V using a small boost (step-up) converter circuit module. The 5 V supply rail is also responsible for powering the four HCSR04 Ultrasonic Sensors attached to each module.

The MCU will be responsible not only for deciding when and for how long the sensors should be powered (using a MOSFET transistor as a switch due to its low voltage drop) but also to trigger their ultrasonic pulse for distance

detection via the return echo signal. This will be achieved through the software configuration in an attempt to reduce power consumption and in turn extend the life of each module before a recharge is required.

The final component of each sensor node is the communication module, the NRF24L01+ transceiver. This module, powered by the 3.3 V rail of the Arduino, transmits the sensor data over the mesh network to the access point, where all sensor data is combined and transmitted to the cloud by the gateway node. The central node/access point consists of an ESP-12E module, powered similarly to that of the Nano modules mentioned above. The proposed circuit design for a typical sensor node is illustrated in *Figure 5*.

3.2 Communication Technologies

The NRF24L01+ transceiver makes use of NRF24 RF technology to transmit data in the free 2.4GHz ISM band. Each transceiver is capable of using 125 different channels in this band [30]. The module can operate at a range of baud rates, where lower transfer rates increase its range. The achievable range is high but can be decreased with a decrease in transmission amplification power. The modules operate best in open air. To achieve this the devices should be configured such that data is transferred through as few obstacles as possible.

3.3 Deployment Methodologies

The sensor nodes will be deployed in a rigid plastic encasing raised off the ground to average bumper height. This container will have the required windows for the ultrasonic sensors which will need to be recessed from the window. In turn all circuitry will be raised off the bottom of the encasing and drainage holes will be drilled into the underside to enable drainage if water is to enter via the windows.

To raise the casing off the ground, a PVC stand will be assembled from electrical PVC conduits. The base will be held together by a 4 way PVC round box with four balance legs for stability. A vertical pole will be erected from the center of the stand on which the encased sensor module will be attached. The stand will be placed at the junction of four parking spaces and will be weighed down to prevent it from being knocked over.

3.4 Data Management

The design choices made in relation to optimised collection and processing of sensor data are described as in the sections that follow.

3.4.1 Mesh Network

The chosen mesh network implements the node-skip topology to reduce the number of node hops required when transmitting sensor data, thus reducing the power consumption of the system. Additional transmitting power will be required for longer distance transmission but is accounted for by optimising the maximum number of nodes

which can be skipped.

3.4.2 Data Upload

The Wits Wi-Fi network has been chosen as the preferred data upload communication scheme. This is due to the infrastructure being readily available and being highly stable with regards to up-time. The network can be connected to by the gateway node which will be an ESP-12E module with an integrated Wi-Fi module.

3.4.3 Data Platform

ThingSpeak will be utilised as the data platform service. This platform is favourable due to its product offerings under a free tier range. Sensor data uploads can be accommodated every 15 seconds which is more than sufficient for the system requirements. A single data upload channel is available which is sufficient due to the system making use of a single gateway node. The maximum message size of 3 KB is sufficient to store all parking bay information. The estimated message size will be 72 B under the assumption that each parking bay is represented by a Boolean value. Furthermore, the platform provides native support for the ESP-8266 platform which is to be used as the gateway node.

3.5 Data Presentation

The process of translating raw sensor data into useful information occurs within the data presentation element of the designed solution. This will be served through a mobile application which is to be developed on the Android platform. The application will provide users, being students who wish to search for available parking spots, with a graphical user interface (GUI) in which an aerial view of the surveyed parking bays will be presented. An available parking bay will be highlighted in green, whilst an occupied parking bay will be highlighted in red.

The required sensor data will be obtained from the utilised data platform in the form of a RESTful API which will allow the application to obtain the sensor data through HTTP requests. The updated sensor data will be periodically requested by the application to provide regular GUI updates of available parking bays.

4 PROJECT TESTING

A number of system tests are required to validate the design and successful operation of the IoT parking system. These tests can be categorised and are described in the sections that follow.

4.1 Hardware

- **System power supply:** The system is sufficiently powered by a suitable power supply that can match the voltage and current requirements of the nodes.
- **Minimal power consumption:** The physical system should be evaluated to ensure that power consumption

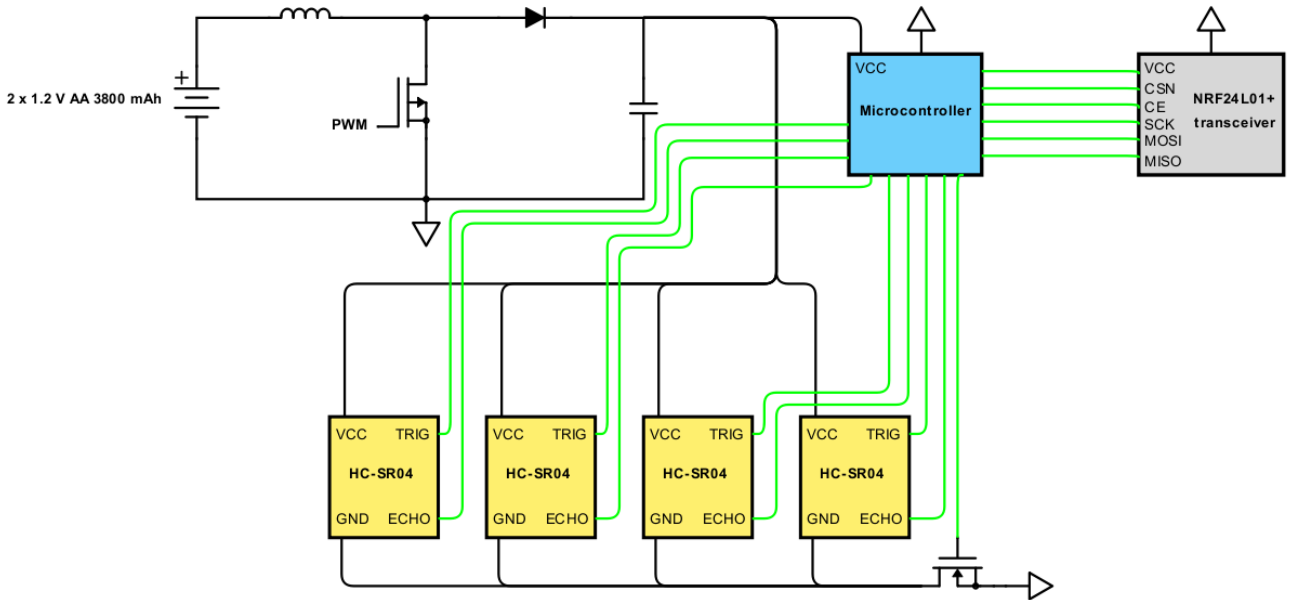


Figure 5: Circuit design of sensor node unit containing boost converter, MCU, ultrasonic sensors, and NRF24 transceiver

is kept to a minimum. This can be achieved through the use of power efficient hardware in conjunction with an optimised software configuration.

- **System accuracy:** Sensors in the network should provide reliable and accurate readings. The MCUs should process this data to ensure false-positives are mitigated.
- **Robustness:** The deployed system is able to successfully withstand exposure to a range of environmental conditions. Additionally, the deployed nodes should be resilient to human interference in the form of car bumping or physical tampering.

4.2 Communication

- **Reliable data transfer:** Sensor data should be successfully transmitted between nodes. Additionally, the received data should be error free or errors should be detected and corrected.
- **Concurrent connections:** Nodes should be able to successfully receive data from multiple transmitting nodes.

4.3 Data Management

- **Mesh network operation:** Nodes should transmit data towards the gateway node. Transmission will be made to a non-gateway node if the gateway node is more than the maximum allowed number of hops away. Received data should be aggregated and transmitted towards the gateway node until the destination is reached.
- **Stable data upload:** The gateway node is able to connect to the Wits Wi-Fi network and periodically transmit the aggregated data of the sensor network to

data platform.

- **Data processing:** Data received from the sensor network is processed in a timely manner and made available through a RESTful API.

4.4 Data Presentation

- **Obtainable data:** Processed data can be requested and retrieved from the data platform via the offered API.
- **Intuitive GUI:** Application users are able to easily navigate the GUI and understand the presented data to locate available parking bays.
- **Application updates:** Data updates are periodically requested by the application. Additionally, users can opt to manually request a data update.

5 PROJECT MANAGEMENT

Management of the project includes the management of work, time-management and scheduling. The project can be divided into distinct phases in which specific workflow should take place. *Figure 6* demonstrates the projected project timeline and the phase progression for the project.

5.1 Design Phase

The design phase consists of a series of iterations of ideation, feasibility analysis, refinement and critical analysis. An important part of the design phase is research, most of which has been presented in the early sections of this report. The final design should ideally be complete before the beginning of the project, yet realistically will continue throughout the implementation and testing phases.

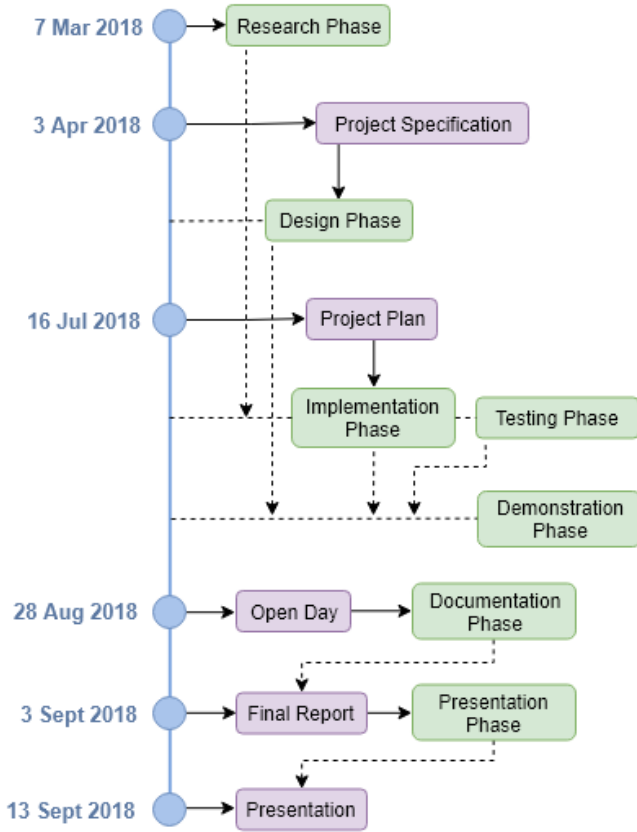


Figure 6: Projected timeline of the project, including workflow (green) and deadlines (purple)

Both the hardware and software components of the design must be considered here, to the point that all possible design implementations are known and the outcomes can be predicted with reasonable accuracy.

5.2 Implementation Phase

The implementation phase will start once all hardware components have been procured and the design is finalised. Implementation will begin with the assembly of the sensor nodes as well as their programming. Following this the communication within the mesh network will be set up and tested. Once sensor data can be successfully retrieved by the access point, the data will be transmitted, via the gateway node, to the ThingSpeak cloud. From here a user application will be developed which can retrieve the parking availability data using an available API and presented to the user via an intuitive user interface.

5.3 Testing Phase

Testing will occur throughout the implementation phase. All individual modules, their collaboration, communications and accuracy will be verified as the system is put together. Once the finalised system is deployed into the parking lot, the system success rate will be tested using a statistical approach. Adjustments to improve the system accuracy will be made and, iteratively, a low variance final product will be produced.

5.4 Demonstration Phase

System demonstration is set to take place at Open Day (28th August 2018), six weeks after the beginning of the implementation phase. Preparation for this phase involves creating posters as well as having the system deployed in the parking lot. On Open Day, the developed system will be demonstrated in real time via the mobile application at the group's stand.

5.5 Documentation Phase

The final report is to be written and submitted during this phase. Although documentation will take place throughout all of the previous stages, writing the report is the sole focus of this phase.

5.6 Presentation Phase

The required final presentation is prepared in this phase. It is to be presented to the board of examiners once the final report has been submitted.

6 CONCLUSION

Research into popular technologies and techniques to develop an IoT based smart parking system has been undertaken. From these, the most plausible options have been chosen, justified and aggregated into a most likely design which will be used to complete the remainder of the project. Essential testing methodologies to validate the system have been discussed. The management of the remainder of the project has been explored, including time and workflow management plans. These have been investigated as number of phases through which the project much traverse to meet its goals and come to completion.

REFERENCES

- [1] F. Wortmann and K. Flchter. "Internet of Things: Technology and Value Added." *Business & Information Systems Engineering*, vol. 57, no. 3, pp. 221–224, Jun. 2015.
- [2] Shenzhen Anxinke Technology Co. "ESP-12E WiFi Module." <https://www.kloppenborg.net/images/blog/esp8266/esp8266-esp12e-specs.pdf>, 2015. Last Accessed: 10/07/2018.
- [3] M.Y.I. Idris, Y.Y. Leng, E.M. Tamil, N.M. Noor, Z. Razak. "Car Park System: A Review of Smart Parking System and its Technology." vol. 8, pp. 101–113.
- [4] EEMB Co. "LIR18650 Lithium Ion Battery 2600mAh." <https://www.ineltro.ch/media/downloads/SAAItem/45/45958/36e3e7f3-2049-4adb-a2a7-79c654d92915.pdf>, 2010. Last Accessed: 10/07/2018.
- [5] HiCharge. "RN-220-XP Lithium Polymer Cell - 1100mAh." <http://ww1.microchip.com/downloads/en/DeviceDoc/Acc-Rn-220XP-Battery-DS.pdf>, 2013. Last Accessed: 10/07/2018.
- [6] Energizer. "Nickel Metal Hydride (NiMH) Handbook and Application Manual." http://data.energizer.com/pdfs/nickelmetalhydride_appman.pdf, 2018. Last Accessed: 10/07/2018.

- [7] Texas Instruments. "ISM-Band and Short Range Device Regulatory Compliance Overview." URL <http://www.ti.com/lit/an/swra048/swra048.pdf>. Last Accessed: 11/07/2018.
- [8] "802.11-2016 - IEEE Standard for Information technology-Telecommunications and information exchange between systems Local and metropolitan area networks-Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications.", 2016.
- [9] "802.15.3-2003 - IEEE Standard for Information technology- Local and metropolitan area networks-Specific requirements- Part 15.3: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for High Rate Wireless Personal Area Networks (WPAN).", 2003.
- [10] Silicon Labs. "BLE112 DATA SHEET." <https://www.silabs.com/documents/login/data-sheets/BLE112-DataSheet.pdf>, 2015. Last Accessed: 10/07/2018.
- [11] Silicon Labs. "BLE112LR DATA SHEET." <https://www.silabs.com/documents/login/data-sheets/BLE121LR-DataSheet.pdf>, 2014. Last Accessed: 10/07/2018.
- [12] Silicon Labs. "Understanding the Bluetooth Connection Process." URL https://www.silabs.com/community/wireless/bluetooth/knowledge-base.entry.html/2017/05/16/understanding_thebl-sicq. Last Accessed: 10/07/2018.
- [13] Nordic Semiconductors. "nRF24L01+ Single Chip 2.4GHz Transceiver Data Sheet." https://www.nordicsemi.com/eng/nordic/download_resource/8765/2/2284534/2726, 2008. Last Accessed: 10/07/2018.
- [14] HopeRF Microelectronics. "RFM69W ISM TRANSCEIVER MODULE DATA SHEET." <http://www.hoperf.com/upload/rf/RFM69W-V1.3.pdf>, 2006. Last Accessed: 10/07/2018.
- [15] P. Ashwood-Smith. "Shortest Path Bridging IEEE 802.1aq Overview.", 2011.
- [16] B. Walke. "The roots of GPRS: the first system for mobile packet-based global internet access.", 2013.
- [17] 3GPP. "Specification of the Subscriber Identity Module - Mobile Equipment (SIM-ME) interface." URL <https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=2793>. Last Accessed: 11/07/2018.
- [18] Quectel. "M10 Quectel Cellular Modem Data Sheet." URL https://www.quectel.com/uploadfile/product/quectel_m10_gsm_specification_v3.0.pdf. Last Accessed: 11/07/2018.
- [19] MyBroadband. "South African University Internet speed shootout." URL <https://mybroadband.co.za/news/broadband/122916-south-african-university-internet-speed-shootout.html>. Last Accessed: 11/07/2018.
- [20] Quectel. "Quectel EC25 IoT/M2M-optimized LTE Cat 4 Module Data Sheet." URL https://www.quectel.com/UploadFile/Product/Quectel_EC25_LTE_Specification_V1.5.pdf. Last Accessed: 11/07/2018.
- [21] Amazon Web Services. "AWS Greengrass." URL <https://aws.amazon.com/greengrass/>. Last Accessed: 11/07/2018.
- [22] Amazon Web Services. "AWS Service Limits." URL https://docs.aws.amazon.com/general/latest/gr/aws_service_limits.html. Last Accessed: 12/07/2018.
- [23] Google Cloud. "GOOGLE CLOUD IOT." URL <https://cloud.google.com/solutions/iot/>. Last Accessed: 12/07/2018.
- [24] Google Cloud. "Google Cloud Internet of Things Core - Device Security." URL <https://cloud.google.com/iot/docs/concepts/device-security>. Last Accessed: 12/07/2018.
- [25] Google Cloud. "Google Cloud Internet of Things Core - Quotas and Limits." URL <https://cloud.google.com/iot/quotas>. Last Accessed: 12/07/2018.
- [26] MathWorks. "ThingSpeak for IoT." URL https://thingspeak.com/pages/commercial_learn_mores. Last Accessed: 12/07/2018.
- [27] MathWorks. "ThingSpeak Standard License." URL https://thingspeak.com/prices/thingspeak_standard. Last Accessed: 12/07/2018.
- [28] SigFox. "Sigfox Technical Overview." URL https://storage.sbg1.cloud.ovh.net/v1/AUTH_669d7dfced0b44518cb186841d7cbd75/dev_medias/build_technical0verview.pdf. Last Accessed: 12/07/2018.
- [29] SigFox. "SigFox White Paper - Security." URL https://www.sigfox.com/sites/default/files/1701-SIGFOX-White_Paper_Security.pdf. Last Accessed: 12/07/2018.
- [30] D. Nedelkovski. "Arduino Wireless Communication NRF24L01 Tutorial." URL <https://howtomechatronics.com/tutorials/arduino/arduino-wireless-communication-nrf24l01-tutorial/>. Last Accessed: 04/07/2018.