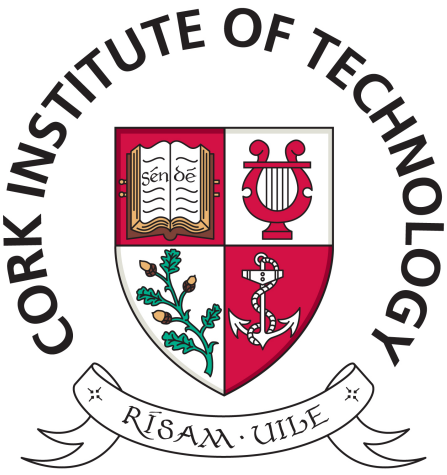


Can IoT Technologies be used to enhance neonatal monitoring systems?

by

Robert Crowley

This thesis has been submitted in partial fulfillment for the degree of Bachelor of Science in Software Development

in the
Faculty of Engineering and Science
Department of Computer Science

January 2018

Declaration of Authorship

I, Robert Crowley, declare that this thesis titled, 'Can IoT Technologies be used to enhance neonatal monitoring systems?' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for an undergraduate degree at Cork Institute of Technology.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at Cork Institute of Technology or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this project report is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: _____

Date: _____

CORK INSTITUTE OF TECHNOLOGY

Abstract

Faculty of Engineering and Science
Department of Computer Science

Bachelor of Science

by Robert Crowley

Parents number one priority is the safety of their children. Parents often struggle to juggle all their responsibilities when they have a young baby in the house as they are always kept busy due to the unpredictability of having an infant in the house. This project would hopefully aid parents manage their day to day living by giving the parents peace of mind when their child is sleeping leaving them to do other things. This project would essentially consist of IoT Devices which all connect back to a base device. The IoT devices would be all positioned around the room to monitor the sleeping baby. The Devices would monitor sound (to monitor if the baby is crying), heat/smoke (to monitor if there is a fire, or if the room is simply too hot or cold), movement (to monitor if an animal such as a dog has entered the room) and possibly an additional feature given I have time would be facial recognition (to detect if the baby has covered their face with a blanket, etc). These devices would transmit information back to the base device if anything is detected which will send an alert to the parents mobile device or the online portal. As you can see in figure 1 which shows the system layout I hope to achieve at the end.From the parents device, there will be options to disable specific alerts e.g. sound alerts. This would make the device more diverse meaning the device could still be used for young toddlers. Users will also be able to view a live feed from the IoT Device. This will enable the parent to check in on their child without having to physically go into the room to check. Each base device will have to be registered with the mobile devices to ensure security and safety for the protection of the baby. Multiple Monitor devices could also be configured to one mobile device. Meaning parents can monitor two children in two different rooms on the same device if needed. This would require a database to manage devices, mobile devices registered to base devices and the preferences for each device. Additional security would need to be put in place for the database as well. In Summary, the project will consist of devices that will monitor for stimulus, and when stimulated will alert the parents via application and/or via web application which will prompt the parent to check on their child. The aim is to aid busy parents to maintain the running of their household and to maintain their peace of mind.

Acknowledgements

I would like to acknowledge my supervisor Dr Sean McSweeney for his continuous guidance and support over the past few months and without whom this project would not have come to fruition ...

Contents

List of Figures

List of Tables

Abbreviations

IoT	**I**nternet **o**f **T**hings
iOS	**i**Phone **O**perating **S**ystem
SIDS	**S**udden **I**nfant **D**eath **S**yndrome
HTML	**H**yper**T**ext **M**arkup **L**anguage
CSS	**C**ascading **S**tyle **S**heet
JSP	**J**ava**S**erver **P**ages
RDBMS	**R**elational **D**ata**B**ase **M**anagement **S**ystem
GERD	**G**astro **E**sophageal **R**eflux **D**isease
TNE	**T**hermo**n**eutral **E**nviornment
IEEE	**I**nstitute of **E**lectrical and **E**lectronic **E**ngineers
AWS	**A**mazon **W**eb **S**ervices
SQL	**S**tructured **Q**uery **L**anguage
NoSQL	**N**ot **O**nly **SQL**
API	**A**pplication **P**rogramming **I**nterface

Dedicated to the two people who inspire me most.
My nephews Conor & Eoin

Chapter 1

Introduction

1.1 Motivation

The motivation behind this project is due to my experiences whilst babysitting my two young nephews (4 months & 24 months). It made me aware, and gave me a better understanding of the difficulties and stress that parents endure when caring for young children. Although both children may have developed a good sleeping pattern, the awareness of potential danger is always prevalent. When children are awake, it is much easier to monitor them. But when asleep, you find yourself monitoring their breathing to make sure everything is okay. Although most times the family pet is completely harmless, the family pet could pose a danger to the infant due to the unpredictable behaviour of our pets often caused by jealousy. I always feel I have to sit next to the kids beds to ensure their safety. This is why I feel that a monitor with some extended features such as video feed, temperature feed, etc. would be most effective and personally would give me more peace of mind when babysitting (and I assume others would be the same). Some of the systems already available are quite expensive, are not very portable, and if portable have a poor battery capacity.

1.2 Executive Summary

The aim of this project is to develop a fully functional baby monitor that can help improve the quality of care an infant can receive. It can also improve the quality of life for the parent/childminder as they will have a deeper understanding of the infants needs. The intention is to reach this functionality by implementing the following features into the monitor:

- Wi-fi abilities
- Accompanying Smart-phone app (both iOS and Android)
- Screen Casting capabilities
- Live video feed with Night Vision and Movement Sensor
- Sound & Talk back features
- Room Temperature and Humidity
- Night Light

Using some of the above features would make the experience with children less stressful and would keep the child's comfortable and safe. The child can be monitored from either a mobile device via the Smart-phone app, laptop, or from a smart television. This will require verification of device ownership, and logging in to ensure safety of information. The room temperature and humidity will be monitored to ensure the babies comfort and safety. The parent/childminder will get a notification to let them know the room is either too hot, too cold or the room needs ventilation, etc. The parent/childminder will also get a notification if there is noise in the room (e.g Babies Crying, etc.) and there will also be the ability to enable a night light, if the child has difficulty sleeping in the dark.

1.3 Contribution

The target of this project is to contribute to increasing the awareness and confidence of parents/childminders who may be responsible for infants. By giving accurate information on an infants environment and well-being there will be heightened sense of awareness.

Objectives include investigating baby monitors currently on the market, to see what sensors are included and what features are built in. The next objective would be to investigate how all those features can be amalgamated to create the system. Once the research has been completed, and the information on the relevant technologies has been retrieved, implementation can being to create the finished product. As mentioned above, the system is designed to make the lives of parents/childminders easier by providing them with relevant information on the infant(s) to ensure their safety.

1.4 Structure of This Document

There will be 5 chapters in this paper, it will be laid out in the following way:

- **Chapter 1: Introduction**
  This chapter gives a brief introduction to the project details and what the project will entail.
- **Chapter 2: Background**
  This chapter will give a in-depth scope of the project which will outline the requirements of the project which will be developed in later chapters
- **Chapter 3: Problem**
  This chapter will mostly focus on the high-level problem definition which will focus on my objectives, which will include my Functional and Non-Functional Requirements.
- **Chapter 4: Implementation Approach**
  This chapter will focus on how the project will work, and how I hope to reach my final product. This will be demonstrated diagrammatically and by creating plans of action.
- **Chapter 5: Conclusions & Future Work**
  This chapter will focus mainly on the plans I made in Chapter 4 and my reflection on these plans, and what adaptions I hope to make in the future.

Chapter 2

Background

2.1 Thematic Area within Computer Science

This project is quite large, due to the broad array of topics it is covering (Web Development, App Development, Advanced Programming, Databases, Internet of Things, Embedded Systems). Internet of Things is the core of this project. Internet of Things (IoT) is a topic that is rapidly growing in the computing industry. More and more research is being conducted into IoT as the years go on. Developing systems such as the Amazon echo or Driver-less cars. The research being conducted is phenomenal and it's changing our daily lives constantly. This project is an IoT device, which again, is designed to change daily life of parents/carers.

This project is a neonatal monitoring system with built-in sensors that will notify the parents/carer that the monitored child requires attention. It will also allow them to check the child from their mobile device or computer. The overall aim is to make neonatal monitoring more efficient and more effective for the parents/carer. The monitor will have features (via the built-in sensors) to detect oxygen levels, CO_2 level, humidity, temperature, movement, sound and will provide a live video feed and ability to play lullabies.

If we take a step back, and look at this quite large project. We can see it is actually mainly made up of a few core areas.

- **Internet of Things**: IoT is defined as being the interconnection via the internet of computing devices embedded in everyday objects, enabling them to send and receive data. This project uses IoT technology to send and receive data between the IoT device and the parents/carers mobile device, to provide real-time updated information.

- **Web Development Platforms**: The technologies being used in Web Development will be languages such as HTML, CSS, JQuery, JavaScript & JSP. These languages will be used in cohesion. Primarily it will be used as the web portal of the project where the user can make changes, register devices and other features.
- **Embedded Systems**: This is how the whole project is powered. The sensors will be relaying information to the embedded device, which will then be processed and relayed to the database which will then be relayed out to the website and application, for the user.
- **Database Management**: This is where an array of information (which has been gathered from the embedded system or the user) will be stored. The data in the database has to be correctly ordered e.g. using Primary Keys, Foreign Keys etc. The type of the database will need to be decided aswell.
- **Cloud Computing**: Cloud computing is a ever developing technology that enables users to store information on the "cloud". The information gathered from the sensor would be stored on the cloud. Due to the sensitive content the sensors gather, not all information would be stored on the cloud for security reasons. Some of the information would be stored locally on the base device or on another device on the network.

In each of these core areas, there has been a lot of research carried out. Investigating the research by other writers will help give an insight into what is already out there, and give a better idea of what we can try and improve on for this project. These are the main areas of interest for this project:

- **Software Engineering**: For the system to be functional, and efficient; a good knowledge of software engineering would be beneficial. The software is what will give the system it's functionality "behind the scenes".
- **Computer Hardware**: For the system to actually materialize, hardware is required. This part mainly consist of manipulating embedded systems into doing what we want them to do, to achieve our ulterior motive.

2.2 Project Scope

As described in Section 1, the project is a baby monitor which will have WiFi capabilities to enable features such as logging in and streaming from your mobile device, or laptop.

The core areas of this project are: Software engineering and computer hardware. Software engineering is broken down into subsections for this project. In this project it is being broken down into Web development and database management. Computer hardware is broken down into Internet of Things, Embedded Systems and cloud computing.

Software engineering entails programming or developing software to be used by the user to achieve a goal, in this case the mobile app, website, and software on the baby monitor would all be included in the software engineering category.

Computer hardware entails configuring the hardware components such as the embedded system (which is a miniature computer, used for small projects like this). In this situation the embedded device will be wired up to sensors which will wait for stimulus and on stimulus will communicate with mobile devices to notify the parents that their child is in distress or at risk.

2.3 A Review of the Thematic Area

Before research for this project was started, a few questions were formulated to help focus the research. The research conducted is based around the following questions:

- What can happen to an infant that monitoring is required?
- What are the important factors that need to be monitored with an infant?
- What modern technology is best suited for the development of a cost-effective IoT baby monitor?
- How is data retrieved from the user or the IoT device to be stored, maintained and managed?
- What environment is most suited for hosting and maintaining an IoT device with a companion application and website?
- What are the security threats associated with IoT devices?
- What similar devices are currently on the market?

With infants, or even small children there are many risks involved, even with something as simple as leaving them to sleep unsupervised. Anything as little as their sleeping position can be fatal in the wrong situations. [1] To reduce risk of Sudden Infant Death Syndrome (SIDS), an infant should be placed in the supine position (Figure 2.3(a)) for each sleep until the infant is 12 months or older. In contradiction with some myths, the supine position does not increase the risk of suffocation, asphyxiation or choking in infants, the same is true for infants suffering from Gastroesophageal Reflux Disease (GERD) due to infants airway anatomy. The infant airway anatomy differs from the adults airway anatomy.[2] (See Figure 2.1)

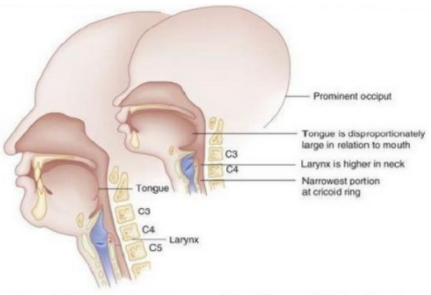


FIGURE 2.1: Airway anatomy of Infant versus that of an adult

Premature infants should be placed in the supine position as soon as possible, this is due to premature infants being at a higher risk of SIDS. Frighteningly, young infants and

more so premature infants are susceptible to SIDS. Another common and avoidable danger to infants is hypothermia. [3]. Hypothermia increases that risk of death and in some cases can lead to premature death. Infants are susceptible to environmental hypothermia. Illnesses such as intracranial hemorrhage and sepsis further increase the risk.[4] Table 2.1 describes the stage of hypothermia in relation to infant body temperature.

Health conditions of neonates Temperature	Range (°C)
Normal Auxiliary	36.6 - 37.5°C
Cold Stress or mild hypothermia	36 - 36.4°C
Moderate hypothermia	32 - 35.9°C
Severe hypothermia	less than 32°C
hypothermia	less than 37.5°C

TABLE 2.1: Health Condition of Neonate in relation to the body temperature

Table 2.1 shows that when the infants body temperature drops below 36.6°C, Hypothermia is setting in. Which means the child is in danger. Table 2.2 indicating the recommended temperature a room should be kept at according to the weight of the infant, and the length of time the child should be kept at that temperature. It is essential that if an infants body temperature dips below the Normal Auxiliary point, urgent medical attention is required and an attempt to reheat the infant should be made whilst awaiting medical intervention.

Weight of neonate	35°C	34°C	33°C	32°C
<1.5 kg	1-10 days	11 days - 3 weeks	3 - 5 weeks	>5 weeks
1.5 - 1.99 kg	N/A	1 - 10 days	11 days - 4 weeks	>4 weeks
2 - 2.49 kg	N/A	1 - 2 days	3 days - 3 weeks	>3 weeks
>2.5 kg	N/A	N/A	1 - 2 days	>3 days

TABLE 2.2: Recommended Room Temperature for neonate according to weight

Asphyxia or suffocation is also an issue that would need to be addressed due to how easily it could happen. It can occur as simple as an infant pulling a blanket over there head, Gastric Reflux, a soft toy falling over on their face or during a common cold illness.

To be in the position to prevent this happening, we need to be aware of the actions and environmental factors which can possibly contribute to these events or giving warning signs of such events.

In 1997 there was an journal entry written by Dorothy Bonn which outlines the effect different sleeping positions had on babies. Although the article was published over 20 years ago, it's still very relevant today. The study was part of the Avon Longitudinal Study of Pregnancy and Childhood (ALSPAC). Research carried out at the Institute of Child Health in Bristol which surveyed first-time mothers of infants born between June 1991 and June 1992. The survey focused mainly on the sleeping position they used for

their newly born infant and any illnesses or signs that appeared at the ages of 4-6 weeks and 6-8 months. When the survey concluded there were 43 health outcomes, which included:

1. Respiratory Symptoms
2. Gastrointestinal Disorders
3. Crying
4. Colic

Two common ailments, Irritant dermatitis (commonly know as 'Nappy Rash') at 4 weeks and seborrheic dermatitis (more commonly known as 'Cradle Cap') at 6 months were significantly more common in infants who were put to sleep in the supine position, than that of infants who were placed in other positions for sleep. From the survey there was a clear indication of which position was favoured by first-time mothers. 70 percent of the surveyors chose to position their sleeping infant in the lateral position (Figure 2.3(b)). Meaning that correlations in the data were statistically strong as seen in Figure 2.2.

The results of the survey shows that infants who were placed in the supine position for sleep were at a lower risk of developing respiratory or gastrointestinal disorders. Infants put to sleep on their side were more likely to be visited by a medical professional in the home, or required to see a medical professional due to health complications. Infants put to sleep on in the front-facing position were linked with an array of medical ailments, mainly respiratory. [5]

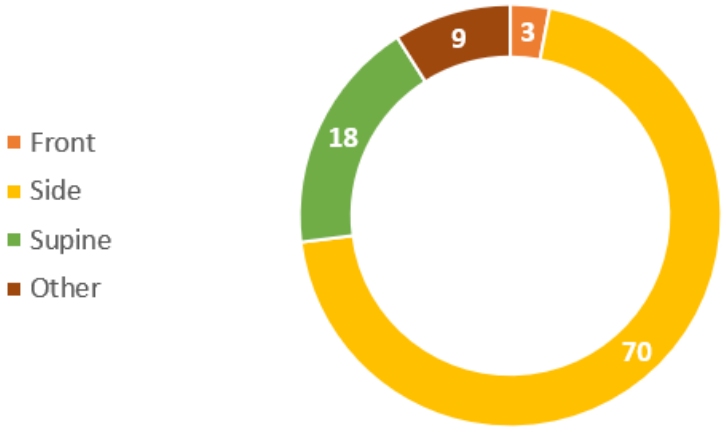


FIGURE 2.2: Results of Survey Carried out by Institute of Child Health, Bristol

The sleep environment of an infant is not solely limited to just Temperature, Humidity, etc. Factor such as Position, Bed Quality, etc. can as-lo be a contributing factor. Young

infants vulnerable to SIDS are infants who change position during there sleep cycle and stay in this incorrect and possibly dangerous sleeping position. Infants put to sleep in the lateral position tend to be twice as likely to suffer from SIDS than infant positioned in the supine position. Unhealthy polystyrene packed pillows or sleeping mats can increase the risk of SIDS three-fold. The modern-day trends tend parents tend to give their sleeping infants large comforters, which may be highly dangerous and another potential contributor of SIDS. When a young infants body temperature increase due to room temperature, fever, sweating or excessive clothing, SIDS can become more prevalent.

The U.S. Department of Health and Human Services released a "Question and Answer" booklet for health care providers in 2005. This booklet outlines the following guidelines to reduce the risk of SIDS:

1. Place infant in supine position for sleep or naps.
2. Always use a safety approved covered mattress.
3. Remove all loose fitting item from infant sleep area to avoid suffocation.
4. Do not permit smoking around the infant
5. Keeps the infants sleeping area in close proximity to your own sleeping area.
6. It is preferred not to put infant to sleep with a soother unless essential
7. Do not cover overheat the baby with blankets or via room heating systems.
8. Avoid products which advertise to reduce SIDS risk without first seeking medical advise.

(a) Supine Position (b) Lateral Position (c) Front Position

FIGURE 2.3: Infant Sleeping Positions

Now that the factors which require monitoring, and why they require monitoring is know investigation into embedded systems is now required. The embedded system needs to meet the following criteria:

- Cost-Effective
- WiFi Capabilities
- Presence of GPIO Pins

When considering potentially devices for use in this project, consideration will be given to technical specification. Some background information on each device will also be provided, which will give a brief overview of the system as a whole. The devices being to be investigated will be:

- **Raspberry Pi**
  - Raspberry Pi Zero
  - Raspberry Pi 3 - Model B
- **ODROID**
  - ODROID-XU4
  - ODROID-C2
- **Arduino**
  - Arduino UNO
  - Arduino Genuino

Raspberry Pi is a small single-board computer developed in the UK by the Raspberry Pi Foundation with the intention of promoting basic computer science in education and in developing countries. The latest release from the Raspberry Pi Foundation is the *Raspberry Pi 3 - Model B*, Released in February 2016. There is also a "minimalist" system, the *Raspberry Pi Zero* which was released by the Raspberry Pi Foundation originally in November 2015, but updated in February 2017 with wireless capabilities.

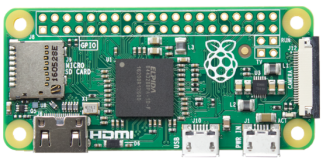


FIGURE 2.4: Raspberry Pi Zero

Raspberry Pi Zero as seen in Figure 2.4, released in November 2015 and re-released in February 2017. The Raspberry Pi Zero is half the size of the *Raspberry Pi 3 - Model B* being only 65mm long by 30mm wide, with twice the capability. A small, cost-effective single-board computer perfect for small projects. The price of the Raspberry Pi Comes

in at €11.04 when purchased from Approved re-sellers.[6] The following are the hardware specifications of the *Raspberry Pi Zero*:

- 1GHz single-core CPU
- 512MB RAM
- Mini HDMI port
- Micro USB Power
- HAT-compatible 40-pin header
- Composite video and reset headers
- CSI camera connection (v1.3 only)

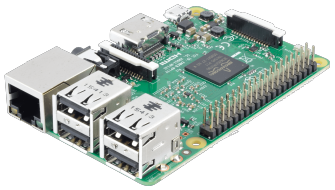


FIGURE 2.5: Raspberry Pi 3 - Model B

Raspberry Pi 3 - Model B as seen in Figure 2.5, released in February 2016. The *Raspberry Pi 3 - Model B* supports powerful external USB devices via the 4 USB 2.0 ports. The *Raspberry Pi 3 - Model B* also comes with built-in wireless capabilities and Bluetooth connectivity. The *Raspberry Pi 3 - Model B* is small in size being only 85.60 mm 56.5 mm. The price of the Raspberry Pi 3 - Model B comes in at €36.69 when purchased from Approved re-sellers. [6] The following are the hardware specifications of the *Raspberry Pi 3 - Model B*:

- 1.2GHz quad-core CPU
- 1GB RAM
- WiFi on-board
- Bluetooth Low Energy (BLE) on-board
- 40-pin extended GPIO
- USB 2.0 Ports x4
- 4 pole Stereo outputs and Composite video port
- Full size HDMI
- CSI camera port
- DSI display port
- MicroSD port
- Upgraded switched Micro USB power source

ODROID developed by Hardkernel Co. Ltd., develop single-board computers and tablets. They are an open source company located in South Korea, Although "ODROID" is a combination of the words "open" and "Android", the hardware is not as "open" as the name would suggest due to the fact Hardkernel retains some parts of the design. Many of the systems developed by Hardkernel are not only capable of running Android, but they can also run Linux distributions. The two most recent releases by Hardkernel in the line of single-board computers are their ODROID-XU4 released in 2015, and the ODROID-C2 released in 2016.

FIGURE 2.6: ODROID-UX4

[7]*ODROID-XU4* as seen in Figure 2.6 is a single-board computer with more powerful, energy-efficient hardware with a smaller size. Hardkernel offer open source support for this board. ODROID-XU4 can run various distributions of Linux including 16.04. Android 4.4 (KitKat) up to Android 7.1 (Nougat) can be run on the ODROID-XU4. The ODROID-XU4 has excellent data transfer speeds, a feature required in modern day computing and to support advanced processing power on ARM devices. ODROID-XU4 has small size of 82mm x 58mm. The ODROID-XU4 comes in at $59.00. [8] The following are the hardware specifications of the *ODROID-UX4*:

- Samsung Exynos5422 Cortex-A15 2GHz and Cortex-A7 Octa core CPUs
- Mali-T628 MP6
- 2GB LPDDR3 RAM PoP stacked
- eMMC5.0 HS400 Flash Storage
- Gigabit Ethernet Port
- HDMI 1.4a for display
- Linux Kernell 4.14 LTS

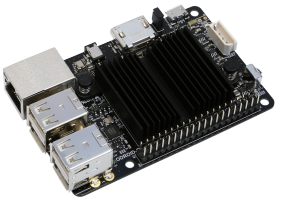


FIGURE 2.7: ODROID-C2

[9]*ODROID-C2* as seen in Figure 2.7 is a single-board computer which aims to be a cheap, small and flexible solution to everyday computing. Built with an ARM 64-bit Quad-Core CPU and open source software, the ODROID-C2 can serve as a platform to make lots of application and projects for an array of different purposes. ODROID-C2 has quite a similar size to the ODROID-UX4 at 85mm x 56mm. The *ODROID-C2* comes in at $46.00. [8] The following are the hardware specifications of the *ODROID-C2*:

- Amlogic ARM Cortex-A53 1.5GHz quad core CPUs
- Mali-450 GPU
- 2GB DDR3 SDRAM
- Gigabit Ethernet
- HDMI 2.0 4K/60Hz Display
- H.265 5K/60FPS and H.264 4K30FPS capable VPU
- 40-Pin GPIOs + 7-Pin I2S
- eMMC5.0 HS400 Flash Storage Slot / UHS-1 SDR50 MicroSD Card slot
- USB 2.0 Host x4, USB OTG x1
- Infrared Receiver
- Ubuntu 16.04 or Adroid 6.0 (Marshmallow) based on Kernel 3.14LTS

Arduino is an open-source platform which is easy-to-use. Arduino board have the ability to read data received, power sensors, light up an LED, retrieve information from a twitter message, etc. The system can turn the input into an output, by sending a set of instructions e.g. Activating a motor or lighting an LED. Arduino boards are cost-effective, cross-platform compatible, simple and most importantly open source. The most recent Arduino board to be released is the Arduino Genuino which was released in April 2016, but the most common Arduino board is the Arduino UNO which was first released in September 2010. [10]

FIGURE 2.8: Arduino UNO

Arduino UNO Rev3 as seen in Figure 2.8, is a micro-controller board based on the ATmeag328P micro-controller. The UNO has 14 digital Input/output pins. The UNO also caters to analog inputs via the 6 analog inputs on-board. The UNO caters the need

of a micro-controller for project development. The UNO is small only being 68.6 mm 53.3 mm. The UNO is quite cost-effective coming in at $22.00 from a trusted re-seller. [11] The following are the hardware specifications of the *Arduino UNO Rev3*:

- ATmega328P Micro controller
- 5V Operating Voltage
- 7-12V Input Voltage
- 14 Digital I/O Pins
- 6 PWM Digital I/O Pins
- 6 Analog Input Pins
- 32 KB Flash Memory
- 2KB SRAM
- 1KB EEPROM
- 16MHz Clock Speed
- 13 Built-in LED

FIGURE 2.9: Genuino 101

Genuino 101 is a board primary aimed to aid learning and development. Genuino 101 delivers high performance, yet low power consumption provided by the Intel Curie Module with the simplicity of Arduino. It has the same robust form factor and peripheral list of Arduino UNO with the addition of a Bluetooth module and a 6-axis accelerometer/gyro. Again the Genuino 101 is small in size at only 61.5 mm x 25 mm. The Genuino 101 can be purchased from a trusted re-seller for $30.00. [12] The following are the hardware specifications of the *Genuino 101*:

- Intel Curie Micro-controller
- 3.3V Operating Voltage
- 7-12V Input Voltage
- 14 Digital I/0 Pins
- 4 PWN Digital I/O Pins
- 6 Analog Input Pins
- 24kB SRAM
- 196kB Flash memory

- 32MHz Clock Speed
- 12 Built-in LED

One of the most important features to analyze of these devices, is the price. As stated before, the aim of this project is to create an Affordable system. This involves finding the device which caters to our needs, and is economically friendly too. Taking all the factors into consideration, the Raspberry Pi 3 - B Model is the device that is most suited to this project. It is easily expandable to lots of projects, especially sensor projects. The pi is not the most economical device investigated, but it is a good system that can be easily built upon.

Now that we know what device is most suited to this projects, we can start investigating the information being stored and how it should be stored. In this project, an array of data will be stored. Information ranging from User Accounts (name, e-mail address, password, etc.), Infant information (infant name, age, gender, etc.) and data gathered from the sensors on the device will also need to be stored. The data gathered can take different forms, such as audio, video, or raw data. From this it is clear that complex data will need to be stored. SQL does not deal with complex data very well in comparison to NoSQL. NoSQL works with complex data and objects much better.

Recently there has been a rise in the interest of non relational data storage, meaning data does not need to follow any particular pattern or "schema". Examples of this move to NoSQL can be seen with Goggle's Big-Table and Facebook's Cassandra.[13] A NoSQL database would provide the a mechanism that employs a less constrained consistency model than a relational database to collect and retrieve large masses of data. The purpose of NoSQL is the simple appending, and querying of large data masses. NoSQL supports features such as a simplistic design, horizontal scaling, and finer availability control. Rational Databases horizontally distribute the load, this is where NoSQL differs from relational database. NoSQL databases do not distribute a logical entity across multiple tables, it is always stored in one place and they do not enforce referential integrity between these logical entities. The two main type of NoSQL databases we will be investigating are:

- Document Based
  - Apache CouchDB
  - MongoDB
- Graphically Based
  - Neo4j

A document based database is one which is designed for storing, retrieving and managing document-oriented information. This would be used in situation where significant amount of information needs to be processed quickly. For example, in an organization such as Facebook, where users expect fast query speed, and there are millions of entries. It would be near impossible for a relational database to handle data of that volume efficiently. Examples of Document Based databases are Apache CouchDB, MongoDB, Terrastore and OrientDB. For research purposes we are only going to be investigating Apache CouchDB and MongoDB. These are two popular platforms for Document Based NoSQL.

Apache CouchDB [14] is an open-source software which focuses on having a scalable, easy-to-use architecture. CouchDB has a document-oriented NoSQL database architecture and is implement in Erlang, a concurrency-oriented language; JSON is used to store data, JavaScript is used to query the data using MapReduce, and HTTP for an API. Features of CouchDB include:

- ACID Semantics
- Offline Functionality
- Distributed Architecture with replication
- Document Storage
- Eventual Consistency
- Map/Reduce views and indexes
- HTTP API

MongoDB [15] stores data in flexible documents which resemble JSON. This means the fields in the documents can differ and the data structure can be changed over time without previous data becoming unusable. The model maps the objects in your applications code, making data easy to work with. Ad Hoc queries, indexing, and real-time aggregation provide efficient ways to access and analyze your data. MongoDB is a distributed database at its core, so high availability, horizontal scaling, and distribution are built in and easy to use. MongoDB is free, open-source and published under the GNU Affero General Public License.

In comparison to other databases which compute relationships at query time (which is resource expensive). A graphical database stores connections as "first-class citizens", readily available for any operations (comparable to joins in relational Databases). Accessing the already persistent connection is an efficient, constant time operation that allows quick traversal through large masses of data. From the research carried out on graphical databases, only one medium was found which is Neo4j.

Neo4j is an open source NoSQL graphical database implemented in Java and Scala. Development starting in 2003, to be released in 2007 publicly. Neo4j is widely used by organization in almost all industries. Neo4j implements the Property Graph efficiently down to the storage level. As opposed to graph processing or in-memory libraries, Neo4j provides full database characteristics including ACID transaction compliance, cluster support and run-time fail-over. [16]

In recent years, there has been a big increase of services which offer online hosting for websites, software projects, etc. One of the biggest providers of these services is Amazon Web Services(AWS). AWS would be known as an "Online Hosting" service which provides cloud computing services. They provide services under the following areas:

- Compute
- Storage
- Database
- Migration
- Developer Tools
- Media Services
- Security, Identity & Compliance
- Analytics
- Machine Learning
- Mobile Services
- AR & VR
- Desktop & App Streaming
- Internet of Things
- Game Development

and more ...

Online Hosting entails using online services which will be the "backbone" of your system. This requires setting up different services, depending on the requirements. The online hosting service that will be investigated is AWS (Amazon Web Services). For this project, we require the following Services:

1. Information Storage
2. Website Hosting
3. Mobile Application Hosting
4. Internet of Things

These are some of the services (provided by Amazon Web Services) that will be used in the project, to give it advanced functionality:

- Amazon DynamoDB - DynamoDB is a fast and flexible NoSQL database service for applications that require consistency and speed at any scale. It is a fully managed cloud database which supports both document and key-value storage models. Its flexible data model, reliability and scaling of capacity makes it a great fit for Mobile apps, web apps and IoT.

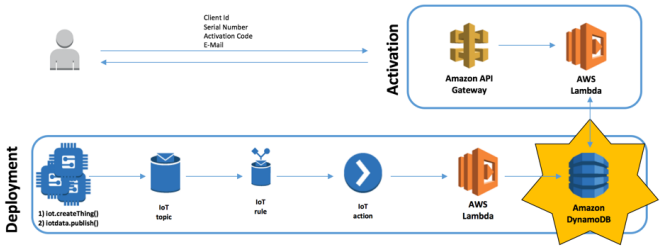


FIGURE 2.10: Amazon DynamoDB

- Amazon S3 - S3 is object storage built to store and retrieve any amount of data from your applications/projects. S3 delivers 99.9% durability, and stores data for millions of applications used by market leaders in almost every industry. S3 provides intensive security and compliance capabilities that meet even the strictest of requirements. It's a flexible way to store, manage and retrieve data of any kind securely.

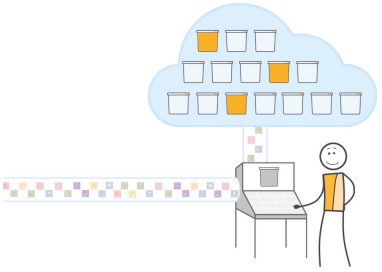


FIGURE 2.11: Amazon S3

- Amazon EC2 - EC2 is a web service which provides secure, scalable computing capacity in the cloud. Designed to make web-scale cloud computing easier for developers. It provides the user with full control of your resources. EC2 reduces time required to obtain and boot a new server instance in minutes. E2 provides developers to build failure resilient application.

- AWS IoT Core - IoT Core is a managed cloud platform that lets connected devices easily and securely interact with cloud application and other devices. With IoT

Core, your applications can keep track of and communicate with all your devices, all the time, even when they are not connected. Benefits of IoT Core include: Management of devices, data security and data processing.

Compared to conventional computing systems, Internet of Things systems are at higher security risk for several reasons:

- IoT systems do not have a defined boundary and are subject to constant change due to device and user mobility.
- IoT systems are highly diverse in terms of their communication medium, protocols, platforms, and devices.
- IoT devices could be automatic systems which control another. Much alike the phrase "Blind leading the blind".
- IoT devices may not be designed to be connected to the internet.
- IoT systems, or portions of them, could be physically unprotected and/or controlled by different parties.
- Unlike smart phone applications, which require permission for installation and many other interactions, permission requests may not be feasible in large scale IoT.

Vulnerability	Examples
Insecure web/mobile/cloud interface	Inability to change default usernames and passwords; weak passwords; lack of robust password recovery mechanisms; exposed credentials; lack of account lockout; susceptibility to cross-site scripting, cross-site request forgery, and/or SQL injection
Insufficient authentication/ authorization	Privilege escalation; lack of granular access control
Insecure network services	Vulnerability to denial-of-service, buffer overflow, and fuzzing attacks; network ports or services unnecessarily exposed to the Internet
Lack of transport encryption/integrity verification	Transmission of unencrypted data and credentials
Privacy concerns	Collection of unnecessary user data; exposed personal data; insufficient controls on who has access to user data; sensitive data not de-identified or anonymized; lack of data retention limits
Insufficient security configurability	Lack of granular permissions model; inability to separate administrators from users; weak password policies; no security logging; lack of data encryption options; no user notification of security events
Insecure software/firmware	Lack of secure update mechanism; update files not encrypted; update files not verified before upload; insecure update server; hardcoded credentials
Poor physical security	Device easy to disassemble; access to software via USB ports; removable storage media

TABLE 2.3: Common IoT vulnerabilities identified by OWASP

Consequently, many IoT systems lack even the most basic security features. Table 2.3 lists the most common IoT vulnerabilities, which were identified by the Open Web Application Security Project (OWASP). A report written in July 2014 on the topic of IoT security by *HP* found that, on overage 25 vulnerabilities existed per device. For example, 80 percent of devices failed to require password of sufficient complexity and length, 70 percent did not encrypt local and remote traffic communications, and 60

percent contained vulnerable user interface and/or vulnerable firmware. There are some well-know (and simple) security practices that address the most common vulnerabilities. The US Computer - Emergency Readiness Team (US-CERT) released an alert in October 2016 about the *Mirai Bot-net* which provided a comprehensive list of such practices, which include:

- Ensure that all default passwords are changed to strong passwords;
- Updating IoT devices with security patches;
- Disabling Universal Plug and Play (UPnP) on routers unless absolutely necessary;
- Monitoring IP ports 2323/TCP and 23/TCP for attempts to gain unauthorized control over IoT devices using the network terminal (Telnet);
- Monitoring for anomalous traffic on port 48101, as infected devices often attempt to spread malaware by using this port to send results to the threat actor.

The US-CERT alert also recommended specific actions by the end user such as acquiring IoT devices from good and trustworthy companies, and understanding the devices communication capabilities, as they're at higher risk of malaware infection.These are reasonable security practices and would provide a first-line of defence, but their application is clearly limited by the scalability of human interaction with IoT devices. Approaches are also required to automatically manage the security of such devices. Another challenge of IoT devices is that even if a software vulnerability is detected, the device may not receive a patch for the vulnerability for a long time. Under these conditions, instruction-detection techniques become even more important. Additionally, as many of the devices may not have powerful processors or adequate memory, the instruction-detection analysis would likely take place at the gateway device. [17]

Due to this project being a product which would be used by parents, grandparents, child minders, etc. A 24 hour survey was carried out on the 30th of October, where a link to a survey on *SurveyMonkey* was posted on Facebook, LinkedIn, etc. The link to the survey can be seen in the references section [18]. Over the 24-hour period 152 people took the survey. Of the 152, there was an analysis of only 100. These are the results of the survey carried out.

Question 1 - Have you regular interactions with infant children (0-18 month)?

- Option 1: Yes (63%)
- Option 2: No (37%)

Question 2 - Do you use a baby monitor? (currently or previously)

- Option 1: Yes (69%)
- Option 2: No (31%)

Question 3 - What features would you like to see on a baby monitor?

- Option 1: Wi-fi enabled (42%)
- Option 2: Live video feed (60%)
- Option 3: Talk back (34%)
- Option 4: Lullabies (35%)
- Option 5: Night Vision (63%)
- Option 6: Room Temperature (67%)
- Option 7: Movement Sensor (59%)
- Option 8: Night Light (46%)
- Option 9: Smart phone App (58%)

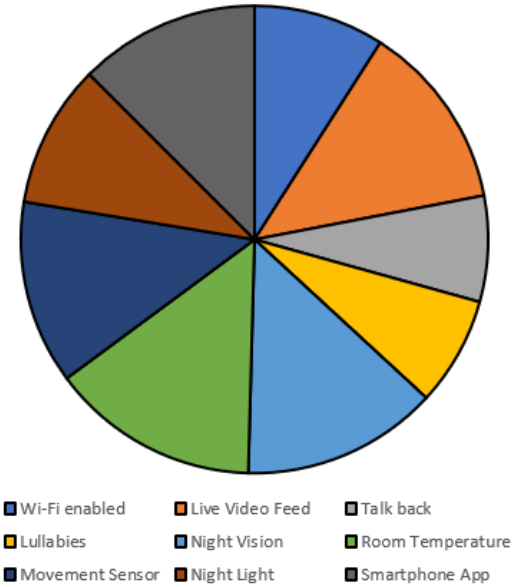


FIGURE 2.12: Pie Chart Representing most in-demand features

Comments:

- Screen casting to smart TV

- Long lifespan (3 to 5 years)
- A zoom-in functionality
- Ability to add extra cameras (for twins/triplets) & alternate between them.

Question 4 - Which method of power supply would you prefer?

- Option 1: Battery Operated (28%)
- Option 2: Electrical (via plug) (72%)

Numerous people suggested having both capabilities, which would make it more portable

Question 5 - At what age would you think children no longer require monitoring? Average Age: 20 months

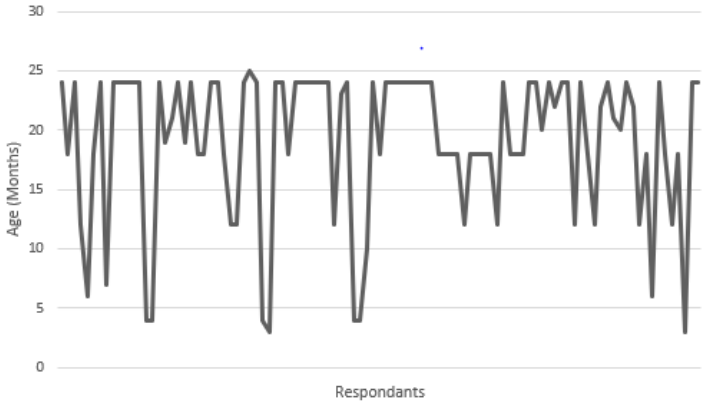


FIGURE 2.13: Line Graph representing the max age for monitoring children

Question 6 - How much would you pay for a baby monitor with all the features you selected above? Average Price: €77

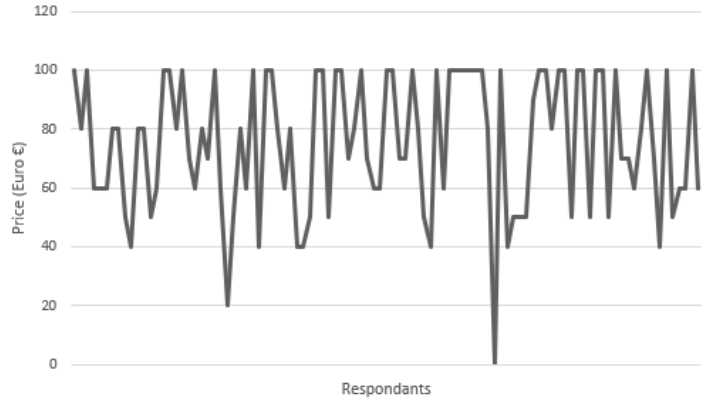


FIGURE 2.14: Line Graph representing the "Reasonable" price range

Question 7 - Any other feedback would be greatly appreciated Comments:

- Baby monitors are very important,helps save babies lives.
- Needs to be very portable for use in hotels, granny's House, etc
- sensor pad as when baby begins to move around the alarm can be activated if they are to far away from the mat , its quite small for the cot beds.
- Sound and night vision very important.
- Having a baby monitor with just basic room temp, vision, and sound is always enough.
- Would have loved a plug in one with battery back up.
- I have a 3 and 5 year old and still use a monitor. Had a picture monitor for my youngest until she was 2
- I would like the option to monitor child until at least age 5yrs as they can get into difficulty when mobile in their bedroom
- A WiFi enabled monitor would be really useful (especially if mobile app enabled) but it would also need to be secure so that the WiFi signal or video feed could not be hacked.
- I'd be worried about a WiFi connection in case it gets hacked, which seems silly, but if it had talk back or a live video feed, those features could be exploited.
- The more portable the receiver is, the better.

2.4 Current State of the Art

There is a wide variety of baby monitors on the market at the moment. All of these monitors offer different features, for different prices. Some of the prices of the baby monitors are wildly overpriced for the facilities they provide to parents. Some of the most expensive baby monitors do not provide very many features other than sound detection or in some cases a video feed. Some of the monitors on the market have flaws which could be fixed by implementing the system using a different method. One of the more expensive products found during research featured the following:

- Video Sensor
- Under-mattress SensorPad for infant movement detection
- Talk-back
- Temperature display
- Parent Unit
- Sound Monitoring
- LCD Full Colour Screen

The monitor which had the features mentioned above cost €229.99 from a reputable retailer. Although the features included are plentiful and it has a nice design, the monitor is still totally overpriced. If you purchase a cheaper baby monitor the lowest found from a reputable retailer which only offered the most basic feature of detecting sound came in at €26.99.

Using Internet of Things for baby monitors is just the tip of the iceberg in relation to the potential applications that could be and are being created. IoT devices are growing at a rapid pace, devices like the Amazon Echo (Figure 2.15).

FIGURE 2.15: Amazon Echo

Chapter 3

Can IoT Technologies be used to enhance neonatal monitoring systems?

3.1 Problem Definition

The aim of this project is to produce a low-cost sensory device that can be used to provide parents/caregivers a higher sense of security when taking care of young infants. The device should be set up with sensors which will detect different stimulus and send a message to the parents/caregivers mobile device. The parent/caregiver should also be able to monitor the infant from their mobile device or Desktop showing a live video feed, statistics, etc.

The overall aim, is to increasing the safety of young infants (while sleeping) by monitoring for distress or environmental issues (E.g. Room Temperature, etc.) and to increase the confidence/peace of mind of parents/caregivers when caring for small infants.

3.2 Objectives

My objectives for this project are as follows:

- I want to create a diverse, low-cost device that can monitor an infant for signs of distress
- I want to implement the device in such a way that it will observe its surroundings and report any issues (e.g. Too hot, cold, etc.)
- I hope to be able create a separate data-set, which once enough data is gathered will try and predict the reason the child is crying.
- I want to develop a web portal which can be accessed which can change device functionality. E.g. Disable temperature monitoring.
- I want to develop an iOS and Android application allowing the monitor to be accessed remotely by the parents/caregivers.

3.3 Functional Requirements

The following are the functional requirements of this project, which are broken down into the sub-sections:

- User shall be able to add another device
- Multiple users shall be able to use the system
- User shall be able to add different profiles for infants
- Users shall be able to log in to the system
- Users shall be able to create an account
- users shall be able to register a device
- A live feed from the monitor shall be available on the Application/Website
- The device shall detect Temperature
- The device shall detect Humidity
- The device shall detect excessive Movement
- The device shall detect smoke
- The device shall play lullabies for the infant
- The device shall send a notification to the device of the parent/carer on stimulus
- The user shall be able to send a summary of previous days to the users email address.
- The user shall be able to give a summary of the stimulus once the issue has been resolved.
- The device shall record snippets of the feed, on user request (via app/website).
- User shall be able to cast the live feed to a smart device. Chromecast, Smart TV, etc.
- The device shall display lights to indicate;
  - **Power** - Green Light
  - **Temperature** - Blue, Yellow, Red Light
  - **Battery** Status - Red Light
- User shall be able to reset their password, if forgotten.

3.4 Non-Functional Requirements

1. At maximum only 2 users can be logged in at any given time
2. At maximum only 2 users can view the live feed at any given time
3. The system should be responsive 99.9% of the time
4. A rooms temperature is deemed "Too Hot" when a temperature reaches 22°C or above.
5. A rooms temperature is deemed "Too Cold" when a temperature drops to 18°C or below.
6. A users password must follow the following guidelines:
   (a) Consist of non-dictionary words
   (b) Have a length of 12 or more characters
   (c) Contain one uppercase character
   (d) Contain one numeric character
   (e) Contain one non-alphanumeric character

Chapter 4

Implementation Approach

4.1 Architecture

At this stage, majority of research has been completed. So this helps with making an educated decision for the technologies and hardware to use for the project. Below in Figure 4.1 you will see the basic flow of the project.

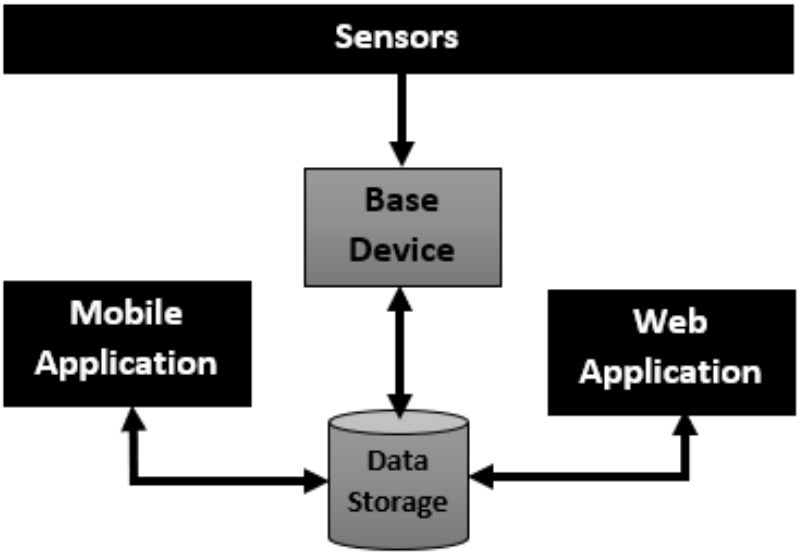


FIGURE 4.1: Basic System Flow

The sensors send information to the "base device" in this case the Raspberry Pi 3 - Model B, the information gets processed in the "base device" and gets sent to "data storage" which will be the NoSQL database. The data is then retrieved by the Mobile and Web Application. Information can be sent from the Applications, to the "Data Storage" to the "base device" in an event when an update to information needs to be made e.g. Disabling a feature. For this project to become a reality an array of different

technologies are going to have to work in unison. Technologies in the following areas are going to be required:

- Software Development
- Web Development
- App Development
- Database Systems

Although, some of these areas have lots of common features, they are all different and unless properly configured they will not be able to work together.

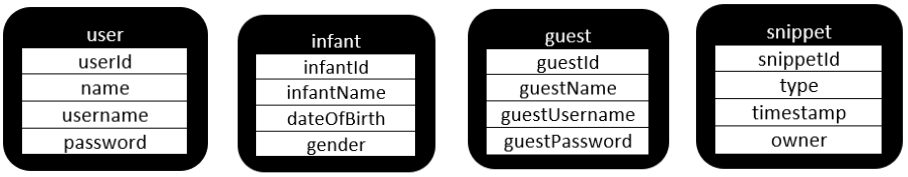
For the web application numerous technologies are required to work together to create a fully functional application. Generally web applications would be made using HTML, CSS, JavaScript, PHP, or other. But due to the growing popularity of the Spring framework, this will be the technology used to develop the web application. Spring Framework is an application framework and an inversion of control container for the Java Platform. While Spring has Java at its core, major languages such as HTML, CSS, etc. are used in conjunction with Java. The intention for the web applications is to create an application that is clear and easy to use. Due to the possibility of elderly people using the application, who may not be as computer literate as users who are frequent users. For developing the Spring Web Application, Spring Tool Suite (STS) will be used as the IDE (Integrated Development Environment). The version of Spring being used will be the most recent version (5.0.1). To ensure the application is fresh and current HTML5 will be used with CSS3. Both which are being constantly updated. The application will consist of 3 sections.

- Landing - This will offer the user the ability to either create an account or login. With information about the system included.
- Dashboard - This will show the user the devices they own, give them an option to add/remove a device and log out.
- Stats - This will display information such as device status (on or off), the stats being detected by the device, and give the option to disable/enable features, return to dashboard and option to see a live video feed.

Developing the mobile application will not be as difficult due to the contents being very concise. The application will be developed on Android (and potentially iOS). Android applications are generally developed using Java. Kotlin is also another possibility to

use, as it is 100% inter-operable with java. Kotlin is a statically-typed programming language which runs on the JVM (Java Virtual Machine) and can also be complied to JavaScript source or use LLVM compiler infrastructure. Both these languages can be used in Android studio the IDE for developing android applications. iOS applications are developed using Swift. Swift is general-usage, multi-paradigm, compiled programming language developed by Apple for iOS, macOS, etc.

Once the Web Application and Mobile Application have been developed the database now need to be configured. For the applications to have full functionality, database access is required for login, account creation, etc. The database type being used for this project is NoSQL due to requirements to store objects, and the schema-less design. The language that will be used is MongoDB. MongoDB can be embedded into Spring and Amazon Web Services supports MongoDB. Below are the collections that we need to be set up, with the intended entries.



(a) User Collection (b) Infant Collection (c) Guest Collection (d) Snippet Collection

FIGURE 4.2: Collections in NoSQL database

At this stage, there are four collections. While there are only four at the moment, there may be more by the end of the project. The four collections at the moment are:

- User collection (seen in Figure 4.2(a)) is used to store data on the users of the system. i.e. parents.
- Infant collection (seen in Figure 4.2(b)) is used to store data on the infant being monitored.
- Guest collection (seen in Figure 4.2(c)) is used to store data on a guest accessing the system. i.e. A Babysitter.
- Snippet collection (seen in Figure 4.2(d)) is used to store data on video or sound snippets taken during monitoring. This data will contain video footage, sound clips or images.

For the base device as mentioned above a Raspberry Pi 3 - Model B will be used to create the sensor device. Majority of the "programming" will be done on the Raspberry Pi. Majority of the programming will comprise of connecting the sensors to the Raspberry

Pi and processing the information collected by the sensors for storage in the NoSQL database. Majority of the programming on the Raspberry Pi will be done by Python, but some C programming might also be used if required. Along with the raspberry pi, sensor will need to be purchased. Sensors and other equipment that need to be purchased include:

- Temperature/Humidity Sensor
- CO_2 Sensor
- Camera/Motion Sensor
- LED Lights
- RGB LED Lights
- Microphone
- Speaker

4.1.1 Use Case Description

A use case is a set of actions or steps of an event which defines the interaction between an "actor" (being a human or an external system) and a system to achieve a goal. The use cases will be split into the appropriate sections. Use case for Web application, Mobile Application and Raspberry Pi.

In the web application, there are two possible users. The parent and a guest. (The guest may be a babysitter). As can be seen in the Use Case Diagram in Figure 4.3 the parent "actor" has full run of the system.

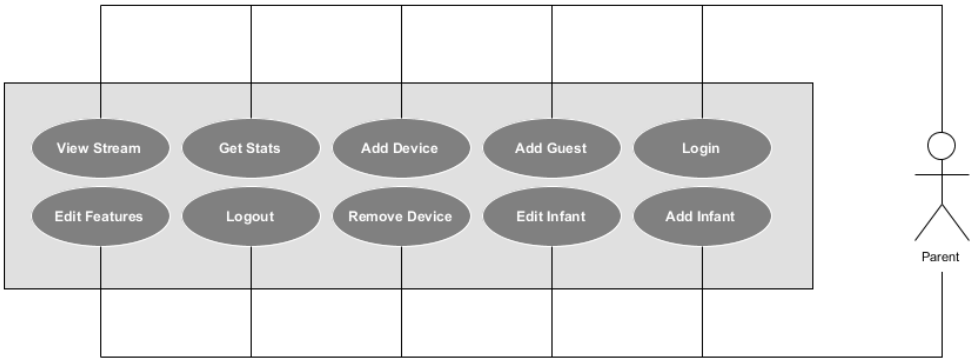


FIGURE 4.3: Web Application Use Case - Parent

The second user of the Web Application is the guest which could be a babysitter, or a family friend, etc. The guest is provided with a random username and password, and is

given limited functionality on the application. As can be seen in the Use Case Diagram in Figure 4.4 the guest "actor" has limited functionality in the system.

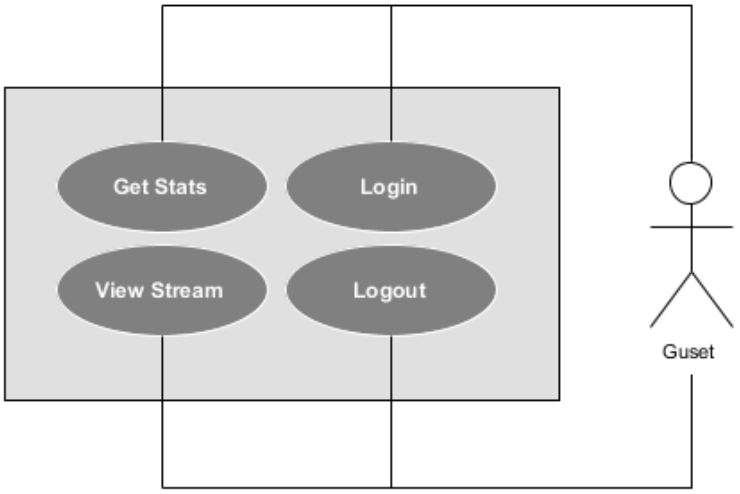


FIGURE 4.4: Web Application Use Case - Guest

In the mobile application, there can also be two possible users. The parent and a guest. As can be seen in the Use Case Diagram the parent "actor" has access to all the possible functionality in the Mobile application as seen in the Use Case Diagram in Figure 4.5

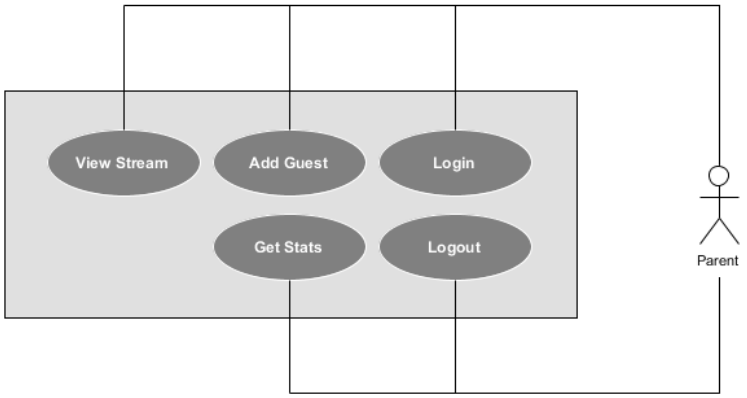


FIGURE 4.5: Mobile Application Use Case - Parent

The second user as mentioned above is the guest. The guest could take the form of a babysitter for example. The guest "actor" does not have full access to the system, rather they can only do what they have to do as seen in the Use Case Diagram in Figure 4.6. The interaction in this project is not primary just human users. Systems will be sending information to other systems during this project.

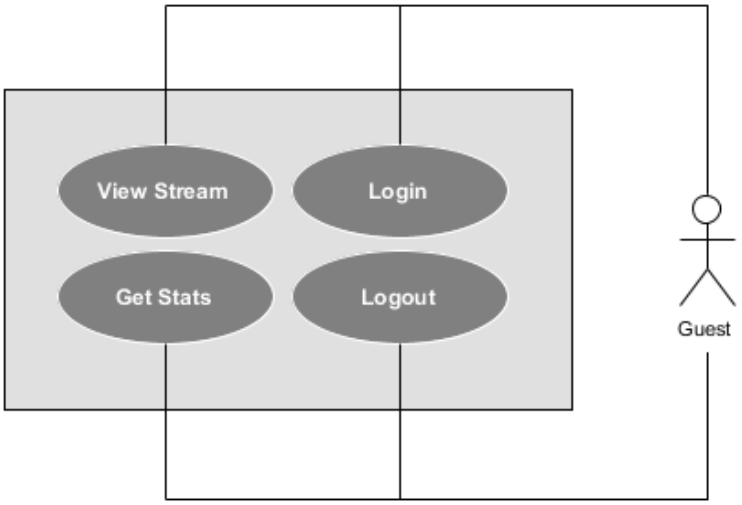


FIGURE 4.6: Mobile Application Use Case - Guest

As discussed in the previous section, the Raspberry Pi will have senors attached to it. These sensors will send information back to the Pi giving it information on the surrounding environment. Each of the sensors carries out a similar function at a high level, but at lower levels the functions are very different. Below in Figure 4.7 you will see, most sensors carry out the same tasks.

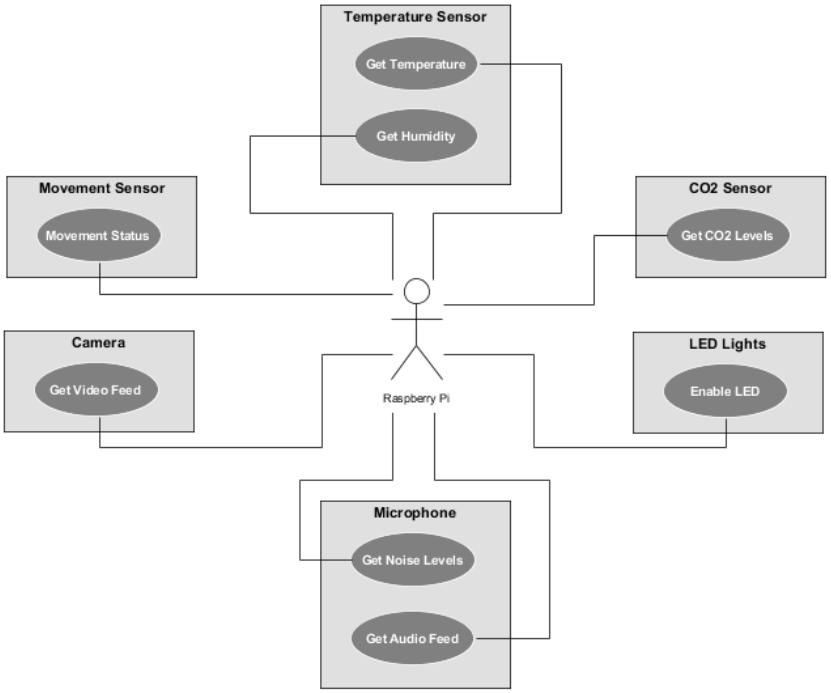


FIGURE 4.7: Sensors Use Case - Raspberry Pi

4.2 Risk Assessment

When beginning the implementation of any software project, it is important to realize that there are things that go wrong, and if you're prepared for things to go wrong the impact will not be as great. To prepare for potential problems, a risk assessment can be quite useful. A plan of action for anticipated problems can be useful, in a situating where an anticipated problem, arises. The biggest risks when conducting this project are:

- **Risk:** Sensors not communicating with Raspberry Pi
  - **Risk Level:** High
  - **Risk Information:** Senors connected to the Raspberry Pi are not transmitting information to the raspberry pi, re-wiring may be required. Some of the sensors may not be suitable for the Raspberry Pi, which will cause issues.
- **Risk:** Failure to complete project by completion date
  - **Risk Level:** Medium
  - **Risk Information:** All sprints will not be completed by deadline. Making "snapshots" at various stages where the system is "working" will reduce the risk of the system not working at the deadline date. If system is not working, there is a backup "working" state.
- **Risk:** Sensors not giving true readings
  - **Risk Level:** Medium
  - **Risk Information:** Although the sensors are not giving accurate readings, the sensors are returning some value. The issue may be with the sensor, or in the software. Check both or try another sensor.
- **Risk:** Information is not being relayed to and from the mobile/web application.
  - **Risk Level:** High
  - **Risk Information:** The information being collected by the device, is not being relayed to the applications, usually would indicate an issue between the database and the applications due to the information being loaded from the data source.
- **Risk:** insufficient Finances
  - **Risk Level:** High
  - **Risk Information:** If an occasion arises, where service from Amazon Web Services exceed the budget, an alternative solution will need to be found.

4.3 Methodology

During this project there are many different elements which need to be addressed. Research and Implementation being the most intensive. Many different techniques were utilized especially in the Background section on this thesis. Websites such as Google Scholar[19] and Cork Institute of Technologies Library Website [20] were used in the research phase. Before writing for the Background section began, a collection of relevant information had to be compiled. This gave a strong platform to build upon. Information gathered comprised of books, journals, wiki entries, reports and websites. The main area to concentrate on first was infant safety/infant mortality. This would give a good insight into what information needs to be monitored and why.

Once the research was completed, implementation was soon the way and plans needed to be put in place. For implementation of the project, structure and planning really helps and at the evaluation stages really stand to the developers. To prepare for the implementation Use Case Diagrams (seen in the previous section), wire-frames (seen in the Appendix A) and plans (seen in upcoming sections) needed to be made. This project involves a lot of different technologies, which need to amalgamate to create a fully functional system. For this to become a reality a wide knowledge base is required. Some of the technology being used is familiar e.g. Spring, Java, HTML, and so on. But some are completely new and so will pose a challenge. Further research into the Raspberry Pi and Raspberry Projects needs to be done. This can be done using book, journals or even YouTube videos. Although proficient in Python, programming in C and dealing with GPIO pins is rather daunting.

The core approach to this project is scrum (as seen in Figure 4.8). For the scrum cycle to be fully effective, a product backlog first needs to be created. A product backlog is a set of tasks that needs to be completed to a high standard before the project is ready for release. These are concise tasks that help direct the work effectively. You then develop a sprint backlog. You take one or more tasks from your product backlog and these tasks will be the subject of the current sprint. Once the sprint backlog is decided upon, the sprint begins. Everyday an analysis or a "scrum" occurs to make sure everything is going to plan, and any weaknesses or issues can be resolved with minimal disruption or time wastage. The sprint is given a general time line, and once the sprint is completed the product should be "marketable". The next task is to conduct a sprint review, which will give feedback on the work done, and can also give information on tasks which may not have been completed to "marketable" standards.

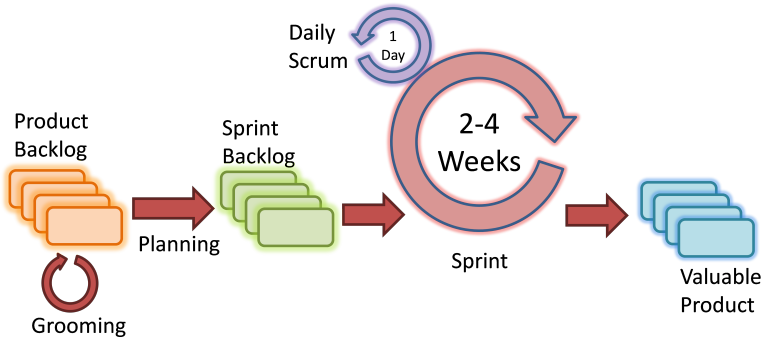


FIGURE 4.8: Scrum Life-Cycle

4.4 Implementation Plan Schedule

For the implementation, the life-cycle being used is the "scrum" life-cycle. As mentioned in the previous section, the "scrum" life-cycle sets main goals at project start called the "product backlog", chooses one or two items at a time from the product backlog to create the "spring backlog", these become the main objects of the sprint. Then a "scrum" occurs frequently to ensure everything is running smoothly and any issues can be "ironed out". Once the sprint is completed a review is carried out to evaluate the results. Below are the sprints which will hopefully be stuck to during this project. After every sprint a review is required to ensure everything has been done to a high standard.

Sprint No.	Completion Date	Sprint Details
Sprint 1	January 20th 2018	Complete Web Application Design
Sprint 2	January 26th 2018	Complete Mobile Application Design
Sprint 3	February 2nd 2018	Setup Database
Sprint 4	February 28th 2018	Configure Raspberry Pi & get information from sensors
Sprint 5	March 7th 2018	Connect Raspberry Pi to database, update information on database from Raspberry Pi
Sprint 6	March 30th 2018	Connect Web/Mobile application to database and view relevant information from sensors
Sprint 7	April 10th 2018	Setup streaming service from Raspberry, to be viewed on Mobile/Web Applications.
Sprint 8	April 15th 2018	Ensure security is implemented, implement where required
Sprint 9	April 20th 2018	Find bugs, Fix Bugs

TABLE 4.1: Sprints

4.5 Evaluation

When coming to the completion of a project, it is good to have a set of preset goals that need to be achieved for the project to be deemed a success. The evaluation section of this thesis will be used for just that. These goals will be similar to the functional requirement, but simply fulfilling the functional requirements does not mean that the system will work. Setting goals against your functional requirements will be a better evaluation. These are the following criteria which need to be achieved in order for this project to be deemed successful.

- The web and mobile application should allow users to login without issue.
- The web and mobile application should allow user to retrieve their password if forgotten.
- The web and mobile application should allow new users to create an account.
- The web and mobile application should display to the user the devices (baby monitors) they own.
- The web application should allow users to add and remove devices
- The web and mobile application should allow the user to view the statistics of their infants sleeping area.
- The web application should allow users to enable or disable features as they see fit.
- The web and mobile application should allow users to video a live feed of the infant
- The web and mobile application should be connected to a database retrieving information on a constant basis.
- The raspberry pi should operate in conjunction with a number of sensors
- The sensors working in conjunction with the raspberry pi should collect information and transmit them back to the raspberry pi.
- The raspberry pi should process the information received by the sensors and log them in the database for retrieval by the applications.
- The raspberry pi should have security protection to prevent malicious attacks.
- The web and mobile application should protect the information being displayed using secure connections to the database.

- The system, web and mobile application should have an up-time of 99.9

Above there are fifteen criteria which hopefully they project will fulfill, it will be deemed a success if the project can fulfill majority or if not all of the criteria.

4.6 Prototype

To aid with implementation, prototyping can be a very effective method of defining what you want the system to achieve or look like but at a very basic level. Wire-framing can be a good technique of prototyping as it gives an insight into the layout of the application. For example in the Web application, There are three sections which would all be represented in wire-frames. This can be achieved using software such as pencil, which make prototyping without coding much easier. The wire-frames are not making the design "set in stone". As you progress through the implementation phase, you may get inspired and the whole design may change. But otherwise the interface would look similar to the wire-frames.

As can be seen in Figure A.1 the first page viewed by the user, is much like a home page which offer the user to login or create an account. To left there will be information on the system and an image. In Figure A.2 you can see the dashboard once you login. This will show the devices you have available to you. It will also tell you the status of each device (on/off). Once you choose on it will take you to the Stats Page as seen in A.3. Here you can see all the statistics on the infants sleeping area. It will also allow "parent" users to manage the features enabled, manage infant accounts, Create a guest user (e.g. Babysitter), view the live feed, or return to the dashboard.

The mobile application has a much more basic design. It only consists of 2 screen, which are the login screen as seen in Figure A.4 which allows the user to login using their username and password. It also offers users to create an account. Once the user is logged in they will be prompted to select a device from the drop-down menu. In the light-green bubbles, the statistics of the infants sleeping area are shown and you are given the option to view the live feed.

All the wire-frames referenced above are in Appendix A.

Chapter 5

Conclusions and Future Work

5.1 Discussion

Doing research of this type for a project is quite foreign to me and certainly writing a thesis on the research conducted is foreign. Although at first it seemed to be a quite daunting and laborious task, it starts to flow much easier as you start writing. There were still challenges during the research that made writing this thesis a little more difficult:

- The adjustment to formally researching and referencing the research proved very challenging. Online services like EverNote helped keep track of the online content I was reading, as I was able to copy and paste snippets of text into EverNote and the source would be with it. This made the referencing of information found much easier.
- At the start of the research phase, I found the research being conducted was very boring and I found little interest in the content I was reading. But once I started reading more and more, the articles and Journals I was reading linked up and the interest in the topic of child safety grew. This interest is what got me through the end of my research as I was able to make my own conclusions from documents I read, opposed to taking the authors conclusions "as gospel".
- One of the bigger issues I found during the research was finding sources that needed payment. This was a big issue with finding sources on Google Scholar. Some of the sources suggested required payment. With the CIT Library Website, all sources were free to use.

5.2 Conclusion

During this project, I have read over an array of journals, books, websites, and even a few pregnancy magazines. I have learned a lot of information in relation to the problem domain. Some of the most important information gained include:

- The safest position to sleep an infant is on their back
- An infants airways have different anatomy than that of an adults
- The ideal body temperature for an infant is approximately 37°C

5.3 Future Work

Had more time been given for the research project, I would have liked to include machine learning in my research as I hope given the time that I can include a machine learning element in the final project. The machine learning element would monitor the babies cry and over time try and predict its needs. This will only be implemented given the time is available.

I also would have liked to also have done some research on facial recognition, in order for the camera to detect if the infants face is being covered by a blanket or the like. Again alike the machine learning element this may be a feature I implement given the time is available once all other features have been implemented correctly.

Bibliography

[1] R. Y. Moon, "Sids and other sleep-related infant deaths: Expansion of recommendations for a safe infant sleeping environment," *PEDIATRICS*, vol. 128, no. 5, November 2011. [Online]. Available: http://pediatrics.aappublications.org/content/early/2016/10/25/peds.2016-2938

[2] L. Adewale, "Anatomy and assessment of the pediatric airway," *Pediatric Anesthesia*, vol. 19, no. s1, pp. 1–8, 2009.

[3] A. S. I. B. Debashis De, Anwesha Mukherjee, "Design a smart neonatal health monitoring system using smcc," *Healthcare Technology Letters*, vol. 4, no. 1, pp. 13–19, August 2016. [Online]. Available: http://digital-library.theiet.org/content/journals/10.1049/htl.2016.0054

[4] [Online]. Available: https://www.merckmanuals.com/professional/pediatrics/perinatal-problems/hypothermia-in-neonates

[5] D. Bonn, "Back stays best position for sleeping babies," 1997.

[6] [Online]. Available: https://thepihut.com/collections/raspberry-pi

[7] [Online]. Available: http://www.hardkernel.com/main/products/prdt_info.php?g_code=G143452239825

[8] [Online]. Available: http://www.hardkernel.com/main/main.php

[9] [Online]. Available: http://www.hardkernel.com/main/products/prdt_info.php?g_code=G145457216438

[10] [Online]. Available: https://www.arduino.cc/en/Guide/Introduction

[11] [Online]. Available: https://store.arduino.cc/arduino-uno-rev3

[12] [Online]. Available: https://store.arduino.cc/genuino-101

[13] Y. L. S. P. Jong Sung Hwang, Sangwon Lee, "A selection method of database system in bigdata environment: A case study from smart education service in korea," *Int. J. Advance Soft Compu. Appl*, vol. 7, no. 1, March 2015. [Online]. Available: http://home.ijasca.com/data/documents/2.Jong-Sung-Hwang-et-al.pdf

[14] N. S. J. L. Chris J, Anderson, *CouchDB: The Definitive Guide*. O'Reilly Media, November 2009. [Online]. Available: http://guide.couchdb.org/editions/1/en/index.html

[15] [Online]. Available: https://www.mongodb.com/what-is-mongodb

[16] [Online]. Available: https://neo4j.com/developer/graph-database/

[17] E. Bertino and N. Islam, "Botnets and internet of things security," *Computer*, vol. 50, no. 2, pp. 76–79, 2017.

[18] [Online]. Available: https://www.surveymonkey.com/r/MKZV9VL

[19] [Online]. Available: https://scholar.google.com/

[20] [Online]. Available: https://library.cit.ie/

Appendix A

Wireframe Models

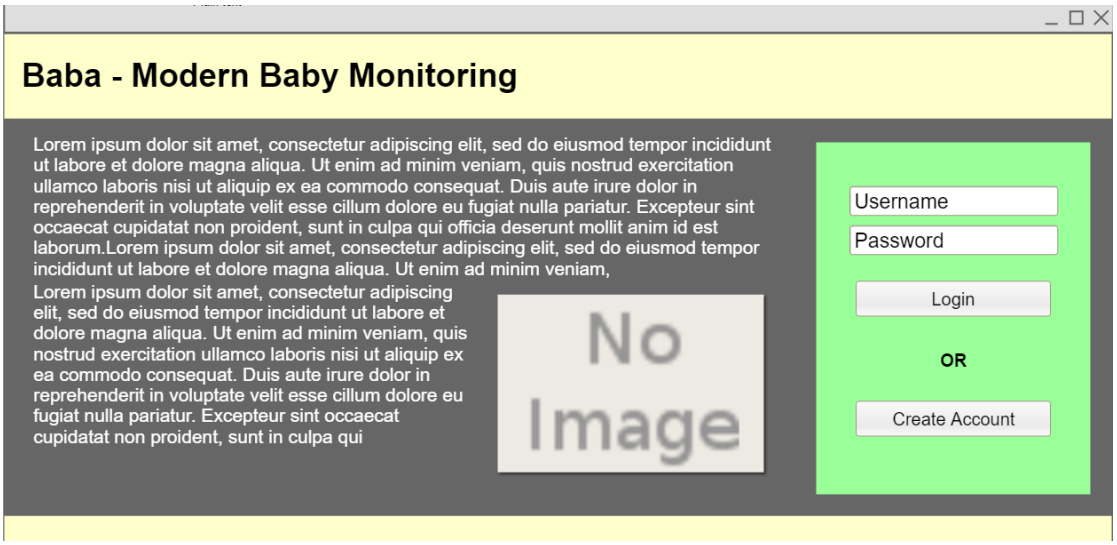


FIGURE A.1: Web Application - Landing Page

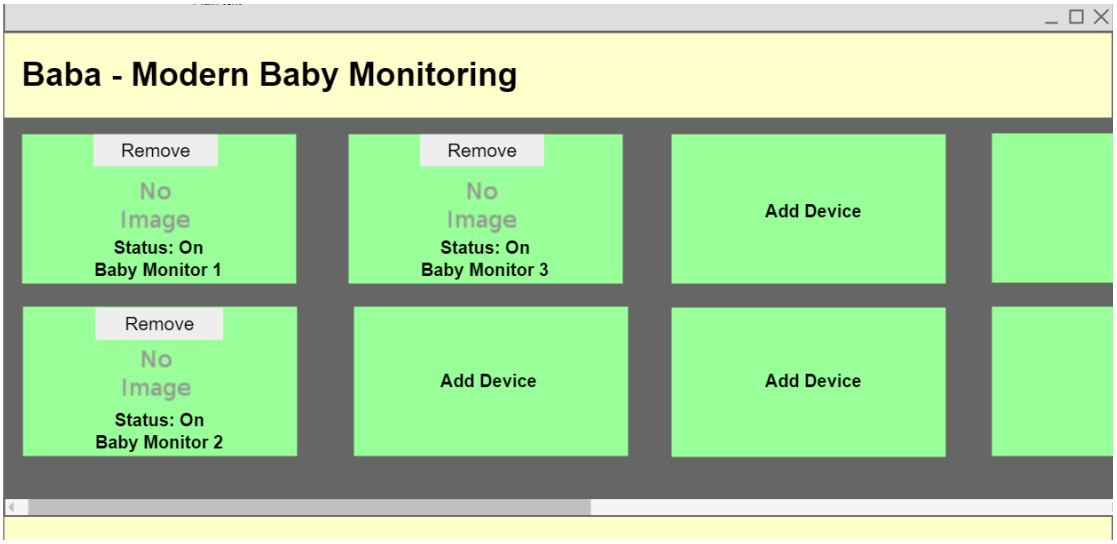


FIGURE A.2: Web Application - Dashboard

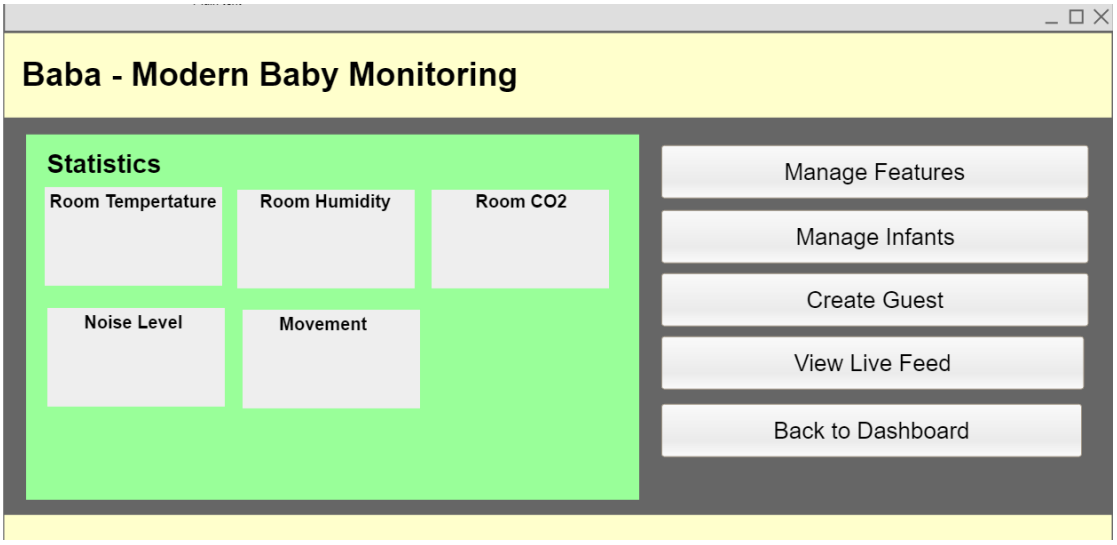


FIGURE A.3: Web Application - Statistics Page

FIGURE A.4: Mobile Application - Login Screen

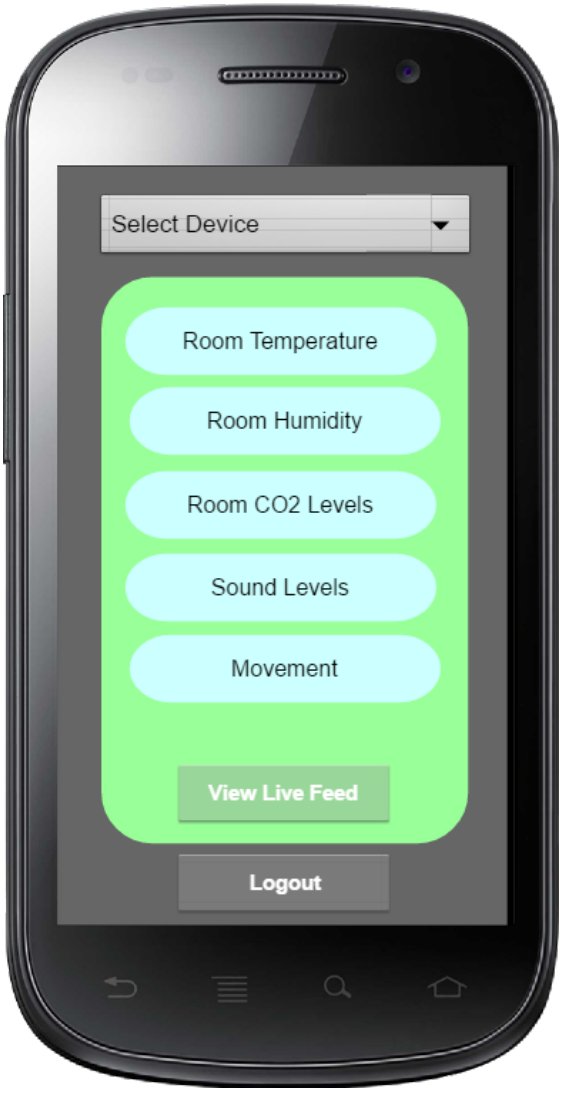


FIGURE A.5: Mobile Application - Dashboard