Smart Mirror Project Report

Sponsor: MakerKid

Student Members:

Quang Trung Trinh

Tuan Minh Nguyen

Haider Ibrahim

Status

/1 Hardware present?

/1 Title Page

/1 Declaration of Joint Authorship

/1 Proposal (500 words)

/1 Executive Summary

# Declaration of Joint Authorship

We are: Tuan Minh Nguyen, Haider Ibrahim, and Quang Trung Trinh, confirm that this project is contributed work between the group and expressed by our own words. Any materials used in any form (table, research, etc...) in our report will be properly referenced and acknowledged at the end of the report. The work breakdown is as follows: Each of us provided functioning, documented hardware for a sensor or effector. Tuan Minh Nguyen provided functioning and documented hardware for the Ranging Sensor Circuit. Haider Ibrahim provided functioning and documented hardware for the Microphone Circuit. Quang Trung Trinh provided documented hardware for the Speaker Circuit. Specification of our works contribution including: Tuan Minh Nguyen is the lead for peripherals connectivity to microcontroller system; Haider Ibrahim is the lead for establishing the mirror interface and internal software; and Quang Trung Trinh is the product manager, lead for further development of our mobile application, connecting the system to database, budget managing and report generating.

# Proposal

We have developed a mobile application (android), interactable with databases, completed a software engineering course and produced three small embedded system with a custom PCB as well as an enclosure laser cut. Each and everyone of us finished our Internet of Things (IoT) capstone project and had the following materials ready for contribution: a working computing model of a smart phone application, an internet accessible database, an enterprise wireless (capable of storing certificates) connected embedded system prototype with a custom PCB and an enclosure laser cut, and are documented via this technical report targeting OACETT certification guidelines.

Intended project key component descriptions and part numbers

Development platform:

Sensor/Effector 1: Ranging Sensor Circuit - Arduino IDE and Raspberry Pi 3B+

Sensor/Effector 2: Microphone Circuit - Raspberry Pi 3B+

Sensor/Effector 3: Speaker Circuit - Raspberry Pi 3B+

We will continue to develop skills to configure operating systems, networks, and embedded systems using these key components to create a finished fully functioning smart mirror that interactable, accessible via the internet and modificable via mobile application

Our project description/specifications will be reviewed by, Jenifer, ideally an employer in a position to potentially hire once we graduate. They will also ideally attend the ICT Capstone Expo to see the outcome and be eligible to apply for NSERC funded extension projects. This typically means that they are from a Canadian company that has been revenue generating for a minimum of two years and have a minimum of two full time employees.

The small physical prototypes that we build are to be small and safe enough to be brought to class every week as well as be worked on at home. In alignment with the space below the tray in the Humber North Campus Electronics Parts kit the overall project maximum dimensions are 12 13/16" x 6" x 2 7/8" = 32.5cm x 15.25cm x 7.25cm.

Keeping safety and Z462 in mind, the highest AC voltage that will be used is 16Vrms from a wall adapter from which +/- 15V or as high as 45 VDC can be obtained. Maximum power consumption will not exceed 20 Watts. We are working with prototypes and that prototypes are not to be left powered unattended despite the connectivity that we develop.

# Executive Summary

In this Smart Mirror project, we have made a mirror not only able to reflect physical object but also able to be hooked up with a computer system and display information. This opens the wide horizon of potential uses of a furniture that take up a lot of space but highly functional once it is interactable via the internet. Our Smart Mirror right now can display the weather, time, news header, calendar. It is also interactable with the user via our microphone and speaker system which is backed up by a google internal software. The mirror display, sound and LED system can be controlled and customized via our android application. With this application, you can control the volume of the speaker, customize the mirror display with simple drag and drop procedure, change the LED color, set a time for a voicemail to be played and connect your phone to the mirror speaker to play any audio file that you want.

Although the smart mirror has been implemented advanced functionality to stand out any regular mirror in the market, we still see a lot of space for development and improvement. Smart mirror can be customized to be a communicational device for video streaming, video call, it can be customized as a perfect fitness device with its features of both reflecting and displaying decently, or it can be a multifunctional smart device like a smart TV but with reflection functionality added on. The horizon is wide and potential of this device is highly scalable. In this project, we have demonstrated a wide range of technical skills such as: electrical system designs and production, mechanical production, programming in multiple platforms and languages, computer system troubleshooting, etc… We hope we can further advance and able to use our knowledge in developing the product with a position of IoT developer.

Contents

[Declaration of Joint Authorship 3](#_Toc27658517)

[Proposal 5](#_Toc27658518)

[Executive Summary 7](#_Toc27658519)

[List of Figures 11](#_Toc27658520)

[1.0 Introduction 13](#_Toc27658521)

[1.1 Scope and Requirements 13](#_Toc27658522)

[2.0 Background 15](#_Toc27658523)

[3.0 Methodology 17](#_Toc27658524)

[3.1 Required Resources 17](#_Toc27658525)

[3.1.1 Parts, Components, Materials 17](#_Toc27658526)

[3.1.2 Manufacturing 17](#_Toc27658527)

[3.1.3 Tools and Facilities 17](#_Toc27658528)

[3.1.4 Shipping, duty, taxes 17](#_Toc27658529)

[3.1.5 Time expenditure 17](#_Toc27658530)

[3.2 Development Platform 17](#_Toc27658531)

[3.2.1 Mobile Application 17](#_Toc27658532)

[3.2.2 Image/firmware 18](#_Toc27658533)

[3.2.3 Breadboard/Independent PCBs 18](#_Toc27658534)

[3.2.4 Printed Circuit Board 21](#_Toc27658535)

[3.2.5 Enclosure 21](#_Toc27658536)

[3.3 Integration 22](#_Toc27658537)

[3.3.1 Wireless Connectivity 22](#_Toc27658538)

[3.3.2 Database Configuration 22](#_Toc27658539)

[3.3.3 Security and Testing 22](#_Toc27658540)

[4.0 Results and Discussions 23](#_Toc27658541)

[5.0 Conclusions 25](#_Toc27658542)

[6.0 References 27](#_Toc27658543)

[7.0 Appendix 29](#_Toc27658544)

[7.1 Firmware code 29](#_Toc27658545)

[7.2 Application code 29](#_Toc27658546)

# List of Figures

[Figure 7. By Android Studio - https://developer.android.com/studio/, CC BY-SA 4.0, https://commons.wikimedia.org/w/index.php?curid=74094999 18](#_Toc27658547)

[Figure 1. Initial schematic. This work is a derivative of "http://fritzing.org/parts/" by Fritzing, used under CC:BY-SA 3.0. 19](#_Toc27658548)

[Figure 2. This work is a derivative of "http://fritzing.org/parts/" by Fritzing, used under CC:BY-SA 3.0. 20](#_Toc27658549)

[Figure 3. Breadboard prototype. 20](#_Toc27658550)

[Figure 4. PCB design This work is a derivative of "http://fritzing.org/parts/" by Fritzing, used under CC:BY-SA 3.0. 21](#_Toc27658551)

[Figure 5. Humber Sense Hat Prototype PCB. 21](#_Toc27658552)

[Figure 6. Example enclosure. 22](#_Toc27658553)

# 1.0 Introduction

# Smart Mirror is a device that not only able to reflect physical object but also able to be hooked up with a computer system and display information. This opens the wide horizon of potential uses of a furniture that take up a lot of space but highly functional once it is interconnected to the internet. It allows users to acquire current weather, time, news header, date and also helps children to get familiar and to learn more about technology as the device is visually cool and the application is easy to use. It also interactive with the user via our microphone and speaker system which is driven by a google service.

## 1.1 Scope and Requirements

Our Smart Mirror project is an Internet of Things (IoT) capstone project that uses a distributed computing model of a smartphone application (android application), a database accessible via internet, an enterprise wireless (capable of storing certificates) connected embedded system prototype with a custom PCB as well as an enclosure laser cut, and is documented via an OACETT certification acceptable technical report.

Our hardware included a Raspberry Pi 3B+ as a central microprocessor that control the mirror’s parts and connected to the online database. We use Raspbian OS as our operating system and development platform for the internal software of the mirror. The mirror peripherals including the following parts: Monitor (Output Device), Speaker (Output Device), Microphone (Input Device), LED (Output Device), and Ranging Sensor (Input and Output Device). The internal software of the mirror has the functions of:

+, Read and send signals from and to the mirror’s peripherals.

+, Read and send data from a to our online database.

+, Processing input signal from the mirror’s peripherals.

+, Able to run media player.

We will also use a 3D printed frames for our smart mirror to hold all the electrical parts together as a finished product.

For database, we use Firebase real-time database to store mirror’s configurations and user identifications in JSON node model. A user-end software was also developed using Android Studio development platform. The program was written in Java and Xml. The name of the software is Smart Mirror and is downloadable via Google Play Store. The application is capable of:

+, Signing In / Signing Up users

+, Connecting to a Mirror’s database with serial number

+, Controlling the Speakers output and play time

+, Changing the LED color, turn on/off LED and Ranging Sensor

+, Monitoring the display of the mirror

+, Read and Send data to Firebase database

The Mirror system and the software will be connected via Firebase real-time database and not directly connected to each other.

Report

/1 Hardware present?

/1 Introduction (500 words)

/1 Scope and Requirements

/1 Background (500 words)

/1 References

# 2.0 Background

Firstly, we would like to thank our mentor Jennifer Turliuk from MakerKids for supporting our project. “MakerKids runs summer camps, after-school programs and parties for kids from the age of 5 to 13 to learn about Robotics, Coding and Minecraft” (MakerKids). Their main purpose is to help the kids to develop their skills as well as their confidence. The iDTech website wrote:” Kids have big imaginations; too big to be contained. Where in the past they only had art supplies like crayons and colored markers at their disposal to get those ideas out and into a conveyable form, they now have computers, tablets, and so much more to help them turn such thoughts into reality.” (Ryan, 2018). As a result, the kids will have qualified level of knowledge to become innovators, inventors and entrepreneurs in the future by making their imaginations become visible.

Secondly, we would like to thank our professor Kristian Medri for helping us a lot throughout this project. He gave us good advices on how to meet the requirements of MakerKids.

Our task is to create an object that can help the kids to learn about technology by having characteristics such as: visually cool so the kids will be interested in; hackable by the kids so the code must not be too hard; easy and simple interaction. This is not only to attract the kids but also to develop their skills interacting with Technology to solve visible problems or change visible peripherals. Our project and MakerKids want to help children understand that Technology is not just about writing some boring code, it is about solving life problems and make life easier.

# 3.0 Methodology

## 3.1 Required Resources

Report

/1 Parts/components/materials (500 words)

/1 PCB, case (500 words)

/1 Tools, facilities (500 words)

/1 Shipping, duty, taxes (250 words)

/1 Working time versus lead time (250 words)

### 3.1.1 Parts, Components, Materials

### 3.1.2 Manufacturing

### 3.1.3 Tools and Facilities

### 3.1.4 Shipping, duty, taxes

### 3.1.5 Time expenditure

Working time versus lead time.

## 3.2 Development Platform

### 3.2.1 Mobile Application

Status

/1 Hardware present?

/1 Memo by student A + How did you make your Mobile Application? (500 words)

/1 Login activity

/1 Data visualization activity

/1 Action control activity

Include screenshots such as Figure 1. Testing. Progress.



Figure 1. By Android Studio - https://developer.android.com/studio/, CC BY-SA 4.0, https://commons.wikimedia.org/w/index.php?curid=74094999

### 3.2.2 Image/firmware

Status

/1 Hardware present?

/1 Memo by student B + How did you make your Image/firmware? (500 words)

/1 Code can be run via serial or remote desktop

/1 Wireless connectivity

/1 Sensor/effector code on repository

### 3.2.3 Breadboard/Independent PCBs

Status

/1 Hardware present?

/1 Memo by student C + How did you make your hardware? (500 words)

/1 Sensor/effector 1 functional

/1 Sensor/effector 2 functional

/1 Sensor/effector 3 functional

The initial schematic design, Figure 2, based on datasheets (Bosch Sensortec, 2019) led to a breadboard layout Figure 3 that was realized Figure 4.

How did you build your Prototype: Breadboard?

Then a PCB was designed, Figure 5, and populated (Figure 6). Bill of Materials, Case, Time commitment. Testing. Progress.



Figure 2. Initial schematic. This work is a derivative of "http://fritzing.org/parts/" by Fritzing, used under CC:BY-SA 3.0.



Figure 3. This work is a derivative of "http://fritzing.org/parts/" by Fritzing, used under CC:BY-SA 3.0.



Figure 4. Breadboard prototype.

### 3.2.4 Printed Circuit Board

Demo

/1 Hardware present?

/1 PCB Complete and correct

/1 PCB Soldered wire visible but trim, no holes or vacancies

/1 PCB Tested with multimeter

/1 PCB Powered up

How did you build your Prototype: PCB?



Figure 5. PCB design This work is a derivative of "http://fritzing.org/parts/" by Fritzing, used under CC:BY-SA 3.0.



Figure 6. Humber Sense Hat Prototype PCB.

### 3.2.5 Enclosure

Demo

/1 Hardware present?

/1 Case encloses development platform and custom PCB.

/1 Appropriate parts securely attached.

/1 Appropriate parts accessible.

/1 Design file in repository, photo in report.

How did you build your Prototype: Case?



Figure 7. Example enclosure.

## 3.3 Integration

Demo

/1 Hardware present?

/1 Data sent by hardware

/1 Data retrieved by mobile application

/1 Action initiated by mobile application

/1 Action recieved by hardware

Report

/1 Enterprise wireless connectivity (250)

/1 Database configuration (250 words)

/1 Security considerations (500 words)

/1 Unit testing (900 words)

/1 Production testing (100 words)

### 3.3.1 Enterprise Wireless Connectivity

How did you make a Database accessible by both your Prototype and Mobile Application?

### 3.3.2 Database Configuration

### 3.3.3 Security

### 3.3.4 Testing

Unit testing and Production testing.

# 4.0 Results and Discussions

Is your prototype perfect? What did you learn?

# 5.0 Conclusions

If you were making 1000 of these.

Report

/1 Hardware present?

/1 Checklist truthful

/1 Valid Comments

/1 Results and Discussion (500 words)

/1 Conclusion

# 6.0 References

OACETT. (2017, March). *I need to Complete a Technology Report*. Retrieved from The Ontario Association of Certified Engineering Technicians and Technologists: https://www.oacett.org/Membership/Technology-Report-and-Seminar

MakerKids. (2020, January 17). Empowering Kids to be Creators, Not Just Consumers. Retrieved January 20, 2020, from <https://makerkids.com/>

Ryan. (2018, May 10). Positive Effects & Benefits of Technology for Children: Kids' Development. Retrieved January 20, 2020, from https://www.idtech.com/blog/benefits-of-technology-for-children

# 7.0 Appendix

## 7.1 Firmware code

Demo

/1 Hardware present?

/3 Code runs concurrently for all sensors/effectors

/1 Project repository contains integrated code

Status

/1 Memo including updates

/1 Financial update

/1 Progress update

/1 Modified Code Files in Appendix

/1 Link to Complete Code in Repository

## 7.2 Application code

Demo

/1 Hardware present?

/1 Memo by student A

/1 Login activity

/1 Data visualization activity

/1 Action control activity

Report

/1 Login activity

/1 Data visualization activity

/1 Action control activity

/1 Modified Code Files in Appendix

/1 Link to Complete Code in Repository