

**T.C.
ISTANBUL AYDIN UNIVERSITY
INSTITUTE OF GRADUATE STUDIES**



**IMAGE PROCESSING AND ARDUINO
INTERFACE APPLICATIONS OF
AN AUTOMATED DESK**

MASTER'S THESIS

Roucoz EL MORR

**Department of Electrical & Electronic Engineering
Electrical and Electronics Engineering Program**

June, 2021

T.C.
ISTANBUL AYDIN UNIVERSITY
INSTITUTE OF GRADUATE STUDIES



IMAGE PROCESSING AND ARDUINO
INTERFACE APPLICATIONS OF
AN AUTOMATED DESK

MASTER'S THESIS

Roucoz EL MORR

(Y1913.300014)

Department of Electrical & Electronic Engineering
Electrical and Electronics Engineering Program

Thesis Advisor: Prof. Dr. Eylem Gülce ÇOKER

June, 2021

ONAY FORMU

DECLARATION

I hereby declare with the respect that the study “ Image Processing and Arduino Interface Applications of an Automated Desk ”, which I submitted as a Master thesis, is written without any assistance in violation of scientific ethics and traditions in all the processes from the project phase to the conclusion of the thesis and that the works I have benefited are from those shown in the Bibliography.
(1/6/2021)

Roucoz EL MORR

FOREWORD

First, I would like to express my endless gratitude to God for being who I am right now and helping me to find patience, strength within myself to complete this thesis.

Second, I would like to thank my family, especially my Parents for supporting me throughout my life. My mother constantly pushed me to be the best person that I can be, and I would not be the person I am today without you in my life. I would like to thank my sisters for all the support and encouragement.

I would like to express my special gratitude and thanks to my advisor Prof. Dr. EYLEM GÜLCE ÇOKER at Istanbul Aydin University for her continuous support, patience, motivation, and immense knowledge that helped me successfully carried out this thesis and all my teachers for their kind guidance and encouragement.

Prof. Dr. EYLEM GÜLCE ÇOKER is not only an academic professor, but a friend with a great heart that keeps encouraging me.

Finally, I would like to thank all my friends, colleagues, and my university for the support I have received during two years..

June, 2021

Roucoz EL MORR

IMAGE PROCESSING AND ARDUINO INTERFACE APPLICATIONS OF AN AUTOMATED DESK

ABSTRACT

The twenty first century is the digital age where books are replaced by technological gadgets, and where employees, office workers, students, and managers spend a lot of hours sitting in the same position, focusing on small screens, writing reports, and achieving their own daily stuff without noticing that their health is negatively affected, hence suffering from health problems that limit their creativity and productivity.

Smart Desk is an innovative solution that registers a person's daily routine, gently notifies him/her of the best times to sit and stand. It is a combination of a desk with sensors and an application that consists of a schedule program where all orders are received by the Arduino controller, via Bluetooth, which orders the motor driver to achieve the corresponding output.

The system is designed with a dual mode: "Student Mode" and "Worker Mode" where the user specifies which mode to be displayed with its relative features. In both modes, there is a reminder system that allows the user to have a private desk that is customized to order food, call a taxi/partner, opens a music, control the light projection and the temperature with respect to the environment, force the worker to drink and do stretching activities after a sufficient time of sitting by adjusting the desk with respect to the height so it can move up or down, manually and automatically. Also, the desk is qualified by a voice recognition system to ensure security, and a camera based on image processing system to recognize any person and give information about him.

Keywords: Smart Desk, Image Processing, Arduino, Windows Application, Mechanical Design.

OTOMATİK MASANIN GÖRÜNTÜ İŞLEME VE ARDUINO ARAYÜZ UYGULAMALARI

ÖZET

Yirmi birinci yüzyıl, kitapların yerini teknolojik araçların aldığı ve çalışanların, ofis çalışanlarının, öğrencilerin ve yöneticilerin aynı pozisyonda oturarak, küçük ekranlara odaklanarak, rapor yazarak ve kendi günlüklerini başararak çok fazla saat geçirdiği dijital çağdır. Sağlıklarının olumsuz etkilendiğini fark etmeden, bu nedenle yaratıcılıklarını ve üretkenliklerini sınırlayan sağlık sorunları yaşıyorlar.

Mekanik Masa, bir kişinin günlük rutinini kaydeden, oturması ve ayakta durması için en iyi zamanları nazikçe bildiren yenilikçi bir çözümdür. Sensörlü bir masanın ve tüm siparişlerin Bluetooth aracılığıyla Arduino denetleyicisi tarafından alındığı ve motor sürücüsüne karşılık gelen çıktıyı elde etmesini emreden bir program programından oluşan bir uygulamanın birleşimidir.

Sistem ikili mod ile tasarlanmıştır: Kullanıcının ilgili özellikleriyle hangi modun görüntüleneceğini belirlediği "Öğrenci Modu" ve "Çalışma Modu". Her iki modda da kullanıcıya yemek siparişi vermek, taksi / partner çağırmak, müzik açmak, çevreye göre ışık projeksiyonunu ve sıcaklığı kontrol etmek, zorlamak için özelleştirilmiş özel bir masaya sahip olmasını sağlayan hatırlatma sistemi bulunmaktadır İşçinin yeterli bir oturuş süresinden sonra masayı yüksekliğine göre ayarlayarak içmesi ve germe aktiviteleri yapması, böylece manuel ve otomatik olarak yukarı veya aşağı hareket edebilmesi. Ayrıca masa, güvenliği sağlamak için ses tanıma sistemi ve herhangi bir kişiyi tanımak ve hakkında bilgi vermek için görüntü işleme sistemine dayalı bir kamera ile yetkilendirilmiştir.

Anahtar Kelimeler: Akıllı Masa, Görüntü İşleme, Arduino, Windows Uygulaması, Mekanik Tasarım.

TABLE OF CONTENTS

DECLARATION	i
FORWORD	ii
ABSTRACT	iii
ÖZET	iv
ABBREVIATIONS	vii
LIST OF TABLES	viii
LIST OF FIGURES	ix
I. INTRODUCTION	1
A. Statement of the Problem	1
B. Hypothesis or Key Questions	2
C. Proposed Solution.....	2
D. Specific Objectives.....	3
E. Relevance of this Work	4
F. State of the Art.....	5
II. LITERATURE REVIEW	6
III. MATERIALS AND METHODS	10
A. Alternative Design Approaches.....	10
a. Hardware	10
b. Peripherals	12
c. Programming Language.....	13
d. Technical Experience	13
B. Selected Design Approach	14
a. Technical Capabilities	14
b. Cost and Experience	15
c. Developmrent Time.....	15
C. Design Specifications	15
D. Multiple Realistic Design Constraints.....	16

1. Economic	16
2. Environmental	16
3. Social	16
4. Political.....	15
5. Ethical.....	17
6. Health and Safety	17
7. Manufacturability	17
8. Sustainability	17
9. Legality.....	18
10. Inspect Ability	18
E. Materials	18
1. Arduino.....	18
2. Servo Motor.....	19
3. Pressure Sensor.....	19
4. Pistons	19
5. GSM Module	20
6. Voice Recognition Module	21
F. Methods	21
G. Cost Analysis.....	24
IV. RESULTS AND DISCUSSION.....	26
A. Testing and Validation.....	26
B. Results	27
C. Discussion	28
V. CONCLUSIONS AND FUTURE WORK	31
A. Conclusions.....	31
B. Future work	30
VI. REFERENCES	32
APPENDICES	34
RESUME.....	62

ABBREVIATIONS

GSM	: Global System for Mobile Communications
SMS	: Short Message Service
USB	: Universal Serial Bus
AC	: Alternating Current
DC	: Direct Current
GPRS	: General Packet Radio Service
mA	: Millimetre Amper
V	: Volt
LED	: Light Emiting Diode

LIST OF TABLES

Table 1 Objectives and Timeline.	3
Table 2 Cost Description.	24
Table 3 Worker Mode Desk Features.	27
Table 4 Student Mode Desk Features.	27

LIST OF FIGURES

Figure 1 Sit-to-Stand Adapters	6
Figure 2 Fixed Standing Desk.....	7
Figure 3 Hand-Cranked Sit-to-Stand Desk.....	8
Figure 4 Powered Sit-to-Stand Desk.....	8
Figure 5 Exercise Desk	9
Figure 6 Desk Design Using Auto Desk Inventor	10
Figure 7 Desk Design Using Auto Desk Inventor	11
Figure 8 Desk Design Using Auto Desk Inventor	11
Figure 9 Top and Side View of the Desk.....	12
Figure 10 Top and Side View of the Desk Moving Legs	12
Figure 11 Image Processing Diagram	13
Figure 12 System Design	14
Figure 13 Arduino Mega.....	18
Figure 14 Servo Motor	19
Figure 15 Pressure Sensor.....	19
Figure 16 Piston	20
Figure 17 GSM Module	20
Figure 18 Voice Recognition Module.....	21
Figure 19 Simulation of Arduino Mega with GSM Module.....	22
Figure 20 Simulation of Arduino UNO with Bluetooth Module.....	23
Figure 21 Circuit Schematic.....	23
Figure 22 Windows Application Flowchart.....	24
Figure 23 Worker Mode Visual Basic Design Interface	28
Figure 24 Student Mode Visual Basic Design Interface.....	28

I. INTRODUCTION

Following up the technological development requires sacrifice of many human needs specifically health needs. Working for long period of times finishing reports, doing research, designing new products... these entire tasks tires human's body. Almost every problem faced by office workers is related to physical disorders caused by sitting for too long gazing at their desktop's screens which leads to a decrease in worker's productivity and creativity.

Providing workers with a product that encourages them to work and takes care of their health is not an easy job hence the project in hands aims to eliminate the difficulties faced at work and improve work conditions.

Smart Desk is an innovative solution that registers a person's daily routine, gently notifies him/her of the best times to sit and stand. It is a combination of a desk with sensors and an application that consists of a schedule program where all orders are received by the Arduino controller, via Bluetooth, which orders the motor driver to activate the corresponding output.

A. Statement of the Problem

The 21st century is the digital age where books are replaced by technological gadgets, and where employees, office workers, students, and managers spend a lot of hours sitting in the same position, focusing on small screens, writing reports, and achieving their own daily stuff without noticing that their health is negatively affected, hence suffering from health problems that limit their creativity and productivity. Hence, according to a research conducted by MAYO CLINIC, it had been shown that sitting for long periods of time can result in an increase number of health concerns including obesity and metabolic syndrome, high blood sugar, excess body fat around the waist and abnormal cholesterol levels (R & Laskowski)

B. Hypothesis or Key Questions

There are a lot of needs that the office worker asks for them in order to have a comfortable, reliable and an automated technological environment, that are the best ways for a productive work. Some of the problems are listed below:

An adjustable desk that allows the user to feel better while he's working. Also the size of the office, and the important question: why the user doesn't have its transportable desk?

A working schedule that is organized from the morning to the evening, and some health problems such as neglecting the three initial dishes during the day, sitting for a while without moving, enhancing the mood with respect to the environment, and forgetting many issues such as calling the partners or identifying an employee in a huge company.

C. Proposed Solution

Being a worker or even a student you must follow up the technology to adapt to global standards in both areas and to do so you must integrate technology in every single detail of your daily routine. What if there is a product that allows you to do your work easily and reminds you to take care of your health. Smart Desk is a working desk that helps you to feel comfortable during working hour. It is not just a desk it is a full package of hardware and software that together forms the core of this desk.

Desks are more than just furniture. They are where problems are solved, ideas created, and businesses are built; that's why Smart Desk was created. No desk is more advanced, or more effective than Smart Desk and here it comes the proof. Smart desk is not just a desk, this desk will rise to your height by reminding you to stand up after being sitting for too long to provide you with the perfect position while working. Not to forget, Smart Desk uses a reminder system that reminds you to eat on lunch time, drink water to keep your body hydrated, call a partner in case you forget, call a taxi when you want to leave, adjust light projection on your desk and the room temperature according to the environmental conditions. What makes this project unique is that it contains two main features: image processing and voice

recognition. These main specifications allow the user to have privacy and confidentiality while working. The voice recognition feature allows the user to lock and unlock his laptop and to interact with the desk through the built-in application used for that purpose. The image processing feature allows the user to recognize and determine the identity of workers or staff members. To allow more flexibility, Smart Desk is designed to be portable and be taken anywhere to serve user's needs (cruise, 2001).

D. Specific Objectives

As mentioned earlier, every worker or student face problems while doing their work or studying. These problems can affect their health and performance in the first place which leads to a drop in the rate of productivity. How these people can manage their responsibilities in a manner where their health and performance are both kept up to the required standards? Smart Desk came as a solution with many features (listed in the previous sections) that allow workers and students to stay comfortable and focus while doing their job. And to produce such desk, a timeline table was made to make sure that the product will be done correctly and successfully.

Table 1 Objectives and Timeline.

Objectives	Timeline
List of all the components with the overall cost	28-9-2020
Hardware design with drawing sheets	1-10-2020 to 10-10-2020
Features of the windows application and starting by the software	13-10-2020 to 27-10-2020
Arduino interface between the application and all the components with the required software	1-11-2020 to 15-11-2020

Table 1 (Cont.) Objectives and Timeline.

Image processing software based on face recognition process	20-12-2020 to 1-1-2021
Implementing all the components on the desk with all the required connections	4-1-2021 to 10-1-2021
Testing the desk and its efficiency	11-1-2021

E. Relevance of this Work

This project has a great use and importance. It is a step forward in the world of technology that motivates workers to be more attentive, productive, and creative. Also it is highly recommended to be used by companies and educational organizations because of these distinctive features in one design. Concerning work organization and security manners, the laptop placed on the smart desk will open and shut down on user's voice. The user will also be able to control light projection and according to the environment. Also the user has the ability to call anyone by only saying "call ...", and if the user wants to change his mood he can listen to music by just saying "turn on music". All these facilities are done using voice recognition to assure privacy and security to the user. When it comes to health, the desk is specialized with a reminder system used to remind the user to stand up so it can moves up or down (manually or automatically) and do some stretching activities after a long time of sitting, or reminds the user to drink water to hydrate the body. Also, it reminds the user to order his/her favorite food by sending a notification at lunch time (specified by the user). And at leaving time, the smart desk sends a notification before five minutes (also determined by the user) to call a taxi to go home.

F. State of the Art

Each company, association, university cares about its profits and seeks to reduce losses by providing a suitable work environment that satisfies workers. To do this, technology must be integrated in every plan that aims to increase productivity rates. An innovative product, Smart Desk, is made to meet all conditions that help workers, students, employees, and managers ... to come up with original ideas.

Building such a product, provides a safer and more efficient system by means of managing time with the presence of a reminder system that schedules all appointments and plans according to user's desires. Also, the reminder system will remind the user to drink, eat, practice sport, call a partner/taxi, listen to music and the most important specification is that the application will notify the user to stand up if he/she was sitting for too long or to sit down if he/she was standing for too long. All these specifications are made to meet user's desires that make him/her more comfortable, creative and productive in its filed.

This desk is unique and differs from all current desks since it is based on new features which are voice recognition and image processing. These two features provide the user with high level of security where the desk will only take and receive commands from the intended user. In addition, this desk allows the user to identify any other person (for example other employees in a huge company) by using face recognition camera module.

With the extra high tech features that are added to this control system, it is expected to form a significant mark in Lebanon where this concept is new and eco-friendly.

II. LITERATURE REVIEW

To make the project more valuable than other designs, some researches must be done to make sure that the project is innovative and unique enough to compete in the market where there might be similar products. A quick look to what have been done before will reveal the strength of the product.

To begin with, Smart Desk is not a unique product or a new idea, the concept is already applied in many countries but what was done here is to enhance the existing ones and provide more tools and features to facilitate work and increase the rate of creativity and productivity that surely will increase the profits in each company or business.

Different types of desks were implemented each one with different design and specifications. Let's take a look at some of these designs:

- **Sit-to-Stand Adapters**

This is basically a mini-desk (as seen in the figure below). It stands on the top of the actual traditional desk and uses a manual control system or a gas-operated lifting mechanism as that used in office chairs. This mini-desk is characterized by its small shape where it can be placed on any desk or table (Michael).



Figure 1 Sit-to-Stand Adapters.

- **Fixed Standing Desks**

This type of desk is fixed on certain height and cannot be adjusted to different positions. It is known that these desks are, sometimes, equipped with sport machines to profit of the time while the user is standing. But one of its disadvantages that it cannot be adjustable to sit so the user will need a secondary desk equipped with a chair to sit in case the user felt tired of been standing for too long (Michael).



Figure 2 Fixed Standing Desk.

- **Hand-Cranked Sit-to-Stand Desks**

This is a perfect type of desks for users who want a flexible desk to fit in one work area that can accommodate both sitting and standing. It depends on manual motion in order to control the position and the stability of the desk (Michael).



Figure 3 Hand-Cranked Sit-to-Stand Desk.

- **Powered Sit-to-Stand Desks**

It's a new type of standing desks where it uses electric motor, placed on the legs of the desk, to control its motion accurately and correctly. This desk is characterized but its ability to move up or down in few seconds without consuming human energy as in the case of desk with manual control (Michael).



Figure 4. Powered Sit-to-Stand Desk.

- **Exercise Desks**

This type of desks is an alternative to powered desks where a sport machine added to it. It is a combination of a desk and a sport machine to make sure that the user takes care of its health while working. It is a good combination that makes beneficial time consumption (Michael).



Figure 5 Exercise Desk.

III. MATERIALS AND METHODS

A. Alternative Design Approaches

a. Hardware

There are many independent companies that are specialized in manufacturing office desks and automation systems, these companies design and create their own kits starting from data acquisition to end up with many designs for the desk, some of them use Auto Cad and others Auto Desk Inventor.

The Smart Desk will be designed on the Auto Desk Inventor professional program that allows you to create the main parts of the desk body, assemble all the components, and show measures and sizes of your design. This program not only offers you a creative environment, but also allows you to draw the sheet of materials that includes all the measures, and to record your work with all the features as a final state. The figures below show the design of the desk and its drawing sheets (Auto Desk, 2000).

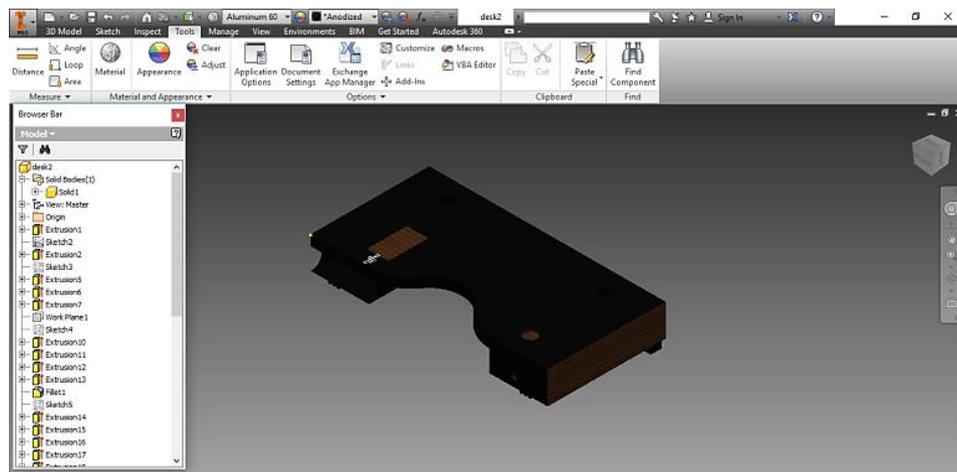


Figure 6 Desk Design Using Auto Desk Inventor.

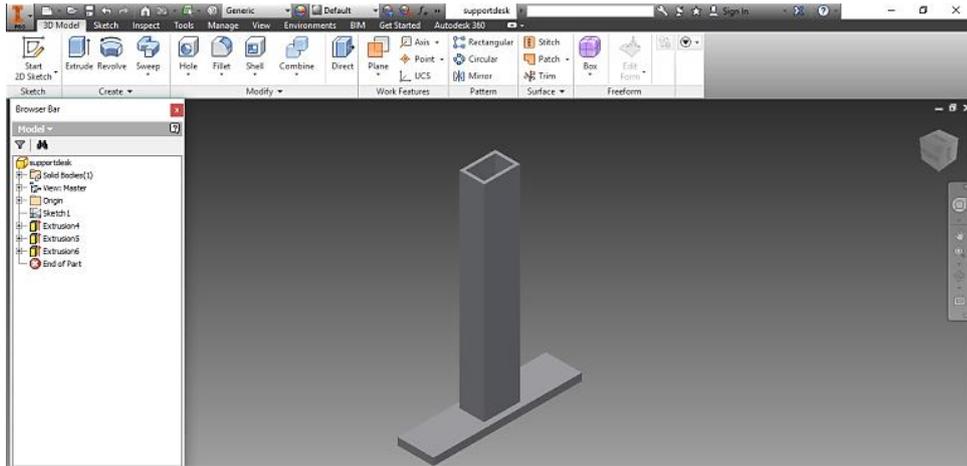


Figure 7 Desk Design Using Auto Desk Inventor.

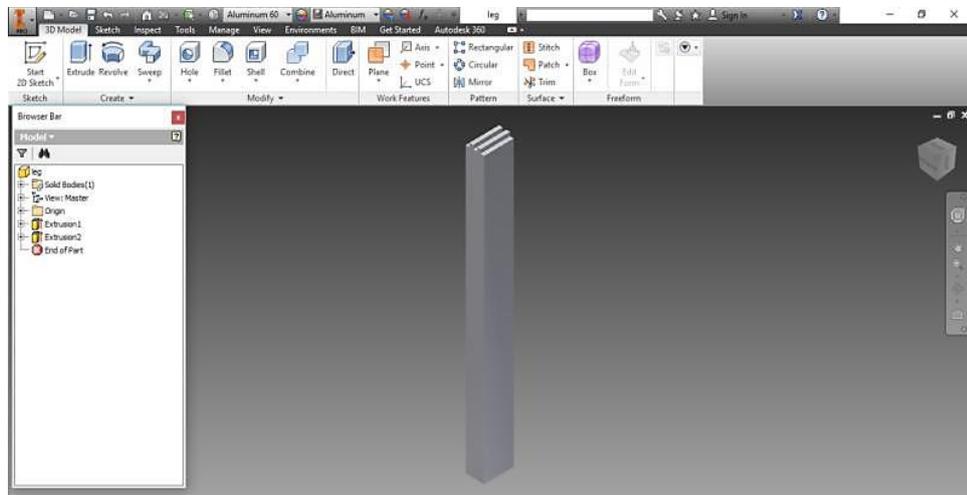


Figure 8 Desk Design Using Auto Desk Inventor.

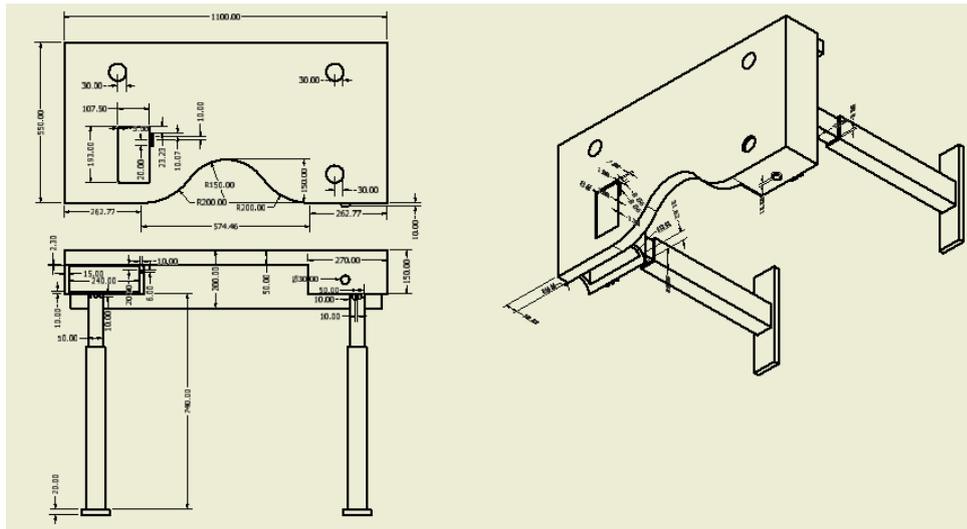


Figure 9 Top and Side View of the Desk.

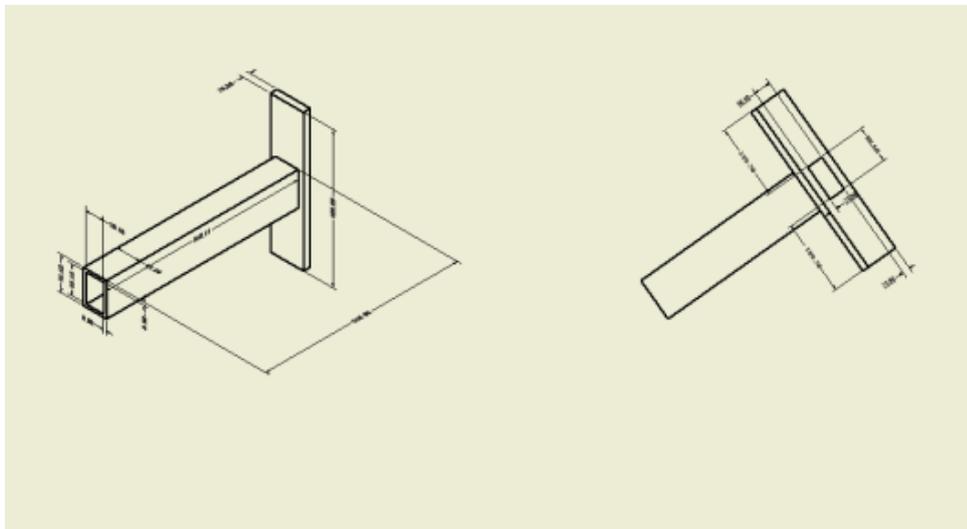


Figure 10 Top and Side View of the Desk Moving Legs.

b. Peripherals

The minor importance of the system is to organize the day of an office worker from the time he enters his office, until he leaves. In addition, this desk is also designed for the student to get rid of the school bag, books, and to organize his schedule, exams and tutorials. The office worker environment is characterized by an interaction between the user and the customized desk through verbal orders and voice notification responses about food, drink, light, posture, and the working rate.

Concerning security and privacy, image processing is used and it is based on the user's face, it also identifies the user's guests. This desk is designed to work

anywhere; indoor or outdoor, to ensure the ability of the user to work or to study at any time.

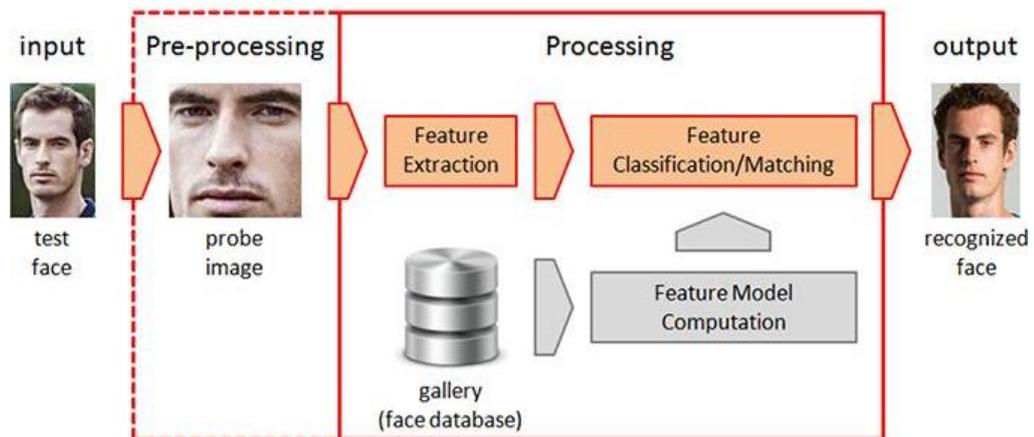


Figure 11 Image Processing Diagram.

c. Programming language

Many programming languages are used to communicate between the hardware and the user; hence software and hardware are combined in a way to ensure a well-organized and an acceptable system. The main two programming languages used are C# and XAML where the programs are coded using Visual Studio program.

The Arduino IDE software is designed to deal with C programming language in order to write the GSM, voice recognition, Bluetooth, and 8 channel relay programs.

Visual basic is used to design the windows application that consists of a reminder system which contains many notifications and output controllers. Windows OPEN CV is used for the image processing feature, and it depends on the compatibility of the tablet to install the libraries of the image processing system (programming world, 2005).

d. Technical experience

If the system was done and implemented in another approach, the technical experience must be the same in all the ways by which the project should be done, some of the essential experiences are software development including, but not limited to web development and windows development. Adding to this, there are some hardware experiences must be taken into consideration which includes the knowledge about microcontrollers and Arduino.

Bearing into minds the capacity organizations to contribute a lot of cash and time to supply a complete product. However, these organizations likewise employ distinctive architects with several of a few encounters, by this way the organization fortifies its specialized understanding and rules the business sectors.

B. Selected Design Approach

a. Technical Capabilities

The framework proposed requires some specialized capacities summed by the learning and experience, including Arduino information and image processing background. Keeping in mind the end goal to have some information in registers frameworks and processors, likewise programming improvement including web and windows advancement. However, the designer should have a general outline on inventor program and how it works including assembly and the drawing sheets.

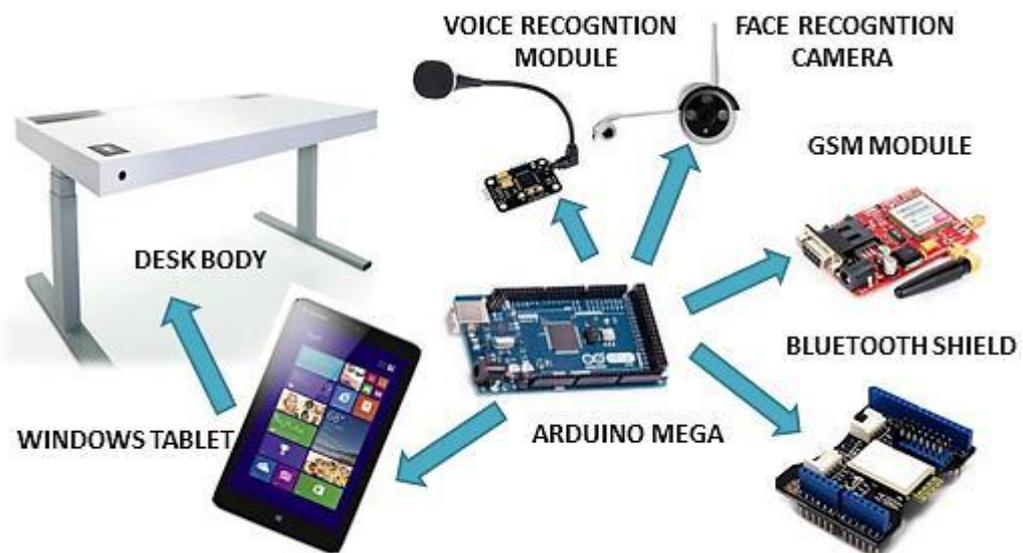


Figure 12 System Design.

b. Cost and Experience

The task incorporates an easy to use unit that comprises of an arrangement of various compartments. It is made in a way that all users are able to understand the product functionality without any previous studies about it. In addition, the utilization of open source advances. For example, the Arduino and the windows improvement condition, concede the engineers an extraordinary favorable position over different outlines. These outlines have a relative highly cost since the undertaking joins between the computerization and the plan.

c. Development Time

Since the assigned project done by the help of open source licensed programs like Arduino IDE 1.8, Visual Studio, Visual Basic and OPEN CV, the development time would be reduced due to the dynamic and flexible environment. As a result, this system should be ready and fully functioning in less than five months.

C. Design Specifications

The desk is tightly designed according to the figure below. The main specifications of the system are wrapped around the Internet of Things, and it is considered an important tool or feature that can be added to the connected world in order to provide the user with a comfortable environment of communication. The desk can be adjusted with respect to your position and posture, this task is accomplished by the Arduino that gives the order for the eight channel relay module to be normally close so that the pistons are activated and can extend the desk for twenty five centimeters, once you accept the notification which insists on the sitting time.

The application designed on Visual Basic, can organize your day and the schedule of meetings. Many voice commands can be given to the voice recognition module and many responds will be issued. The libraries of the image processing system will be installed on the tablet with windows OPEN CV to assure the compatibility with the camera in order to recognize the users and their guests. Besides that, the Arduino can monitor and control the desk; it can be connected to the camera and the tablet by using a Bluetooth shield to manipulate data received from the sensors, the camera and the user. Not to forget to mention that the desk can

receive calls and SMS by the GSM shield connected to the Arduino which interfaces with the tablet.

D. Multiple Realistic Design Constraints

1. Economic

A similar Smart Desk available in the markets with features limited by the tablet without image processing and it costs \$800. The cost of our project is around \$1000 without any profit.

This price is logical and not exceeds the available amount. This desk can simplify the technological environment and put all the communications tools such as automation and image processing in a nutshell. Hence, the product has a huge impact on the economy. The maintenance cost of the product is very low since the desk is a long term product.

2. Environmental

The pistons that adjust the position of the desk can produce a squeaky noise when they are activated. The speakers may produce an audio noise while interacting with the office worker. This desk is friendly to the environment since there is no direct pollution or any negative impact on the environment.

The desk uses electric power to supply all the controllers and components. Hence, there is an indirect air pollution caused by electric stations. Because of the power consumption through many components and this power comes from electric stations that produce air pollution.

3. Social

This product is designed to facilitate the work of the employee in any company, or an instructor in a school or university, or any office worker by letting the user to be in control of everything. People who are not familiar with technology and still working on a classic and old desk may not accept this desk to be a point of transformation of their offices. The design allows the user to identify his guests through face recognition feature.

4. Political

The design is characterized by its mobility, so it can be taken anywhere. This feature can cause political problems whether outside the office in a local place, or while travelling to another country where there are some obstacles that banned the transportation of such desk via airplanes.

5. Ethical

The product consists of a schedule, private data, messages, and secret information about the user, so it is integrated in user's private life. The user is exposed to radiations generated by electronic devices integrated into the desk. The desk agrees with the moral principles and the ethical theories such as utilitarianism since it is designed for the benefit of the most.

6. Health and Safety

The noise generated by the pistons and the speakers can disturb the user. The user isn't exposed to any kind of danger while working on the desk. Precautions and instructions are added on the desk table to inform the children about the product and the method of dealing with all its features, in order to avoid any damage. The Smart Desk will be designed for users to maintain a better health by its schedule and notifications.

7. Manufacturability

The hardware design will be implemented with respect to the measurements with a small manufacturing tolerance. The adjustment of desk's position will be done without a friction between any adjacent surfaces. The use of pistons to control desk's position is important to avoid any kind of vibration.

The product can have different shapes and sizes and its surface can be paintable. Aluminum can be welded in order to assemble the desk's parts.

8. Sustainability

The product will be the first engineering design at our company Smart specified in designing intelligent furniture. Also, it will combine all the technological and security features in one system. Severe radiation and corrosive fluid cannot alter the

design. The product is ductile, durable, and can sustain your long working life since there are good quality components with packaging services.

9. Legality

The software of the design meets the required codes. A privacy system should be added to protect the data of the user. There are no illegal or toxic materials used in the design.

10. Inspect ability

Maintenance services are available if any problem occurs in the inspected desk. The goal of the design is to maintain a better health by organizing the working day, the time for meals and breaks. Also, it supplies the user with all other needs.

E. Materials

The materials needed to complete this system are composed of the following elements:

1. Arduino Mega

The Arduino Mega is a microcontroller board based on the ATmega1280. It has fifty four digital input/output pins. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with an AC-to-DC adapter or battery to get started. In this project, the orders are received by the Arduino controller via Bluetooth which gives an order for the motor driver that activates the corresponding output (arduinomega, 2007).



Figure 13 Arduino Mega.

2. Servo Motor

Servo motors are self-contained electric devices that rotate or push parts of a machine with great precision. Servo motors are used to make Arduino functional and make the desk moves up or down, forward or backward.



Figure 14 Servo Motor.

3. Pressure Sensor

A pressure sensor is a device that senses and measures pressure. The pressure sensor in electronic circuits is in the form of an integrated circuit. In this project, pressure sensors are linked to the chair to sense the presence of the user and control the action of the desk; if it should move forward or backward according to the position of the user on the chair. (Pressure sensor , 2010).



Figure 15 Pressure Sensor.

4. Pistons

A piston is an electrical motor that slides up and down inside tubes and cause various parts of the engine to move. It simulates the actuator in order to move the desk up, down, forward or backward.



Figure 16 Piston.

5. GSM module

With the Arduino GSM Shield the library empowers an Arduino board to do the greater part of the operations you can do with a GSM telephone: put and get voice calls, send and get SMS, and associate with the web over a GPRS arrange. The GSM shield has a modem that exchanges information from a serial port to the GSM network.

The modem executes operations by means of a progression of AT orders. The library abstracts low level interchanges between the modem and SIM card. It depends on the Software Serial library for correspondence between the modem and Arduino.

Regularly, every individual order is a piece of a bigger arrangement important to execute a specific capacity. The library can likewise get data and return it to you when essential (GSM Module).



Figure 17 GSM Module.

6. Voice recognition module

The voice recognition module works at an information voltage scope of 4.5 - 5 volts and will draw a current under 40 mA. This module can work with 99% acknowledgment exactness in the event that it is utilized under perfect conditions.

The decision of receiver and the clamor in nature assumes an essential part in influencing the execution of the module. It's smarter to pick a mouthpiece with great affectability and attempt to decrease the clamor in your experience while offering orders to get the most extreme execution out of the module (Zhon)



Figure 18 Voice Recognition Module.

F. Methods

Many steps should be followed in order to get a senior project that covers all the tasks listed in the previous paragraphs:

- Drawing all the parts with the appropriate measurements, then assembling them to have the desk body, using inventor professional program.
- Printing the drawing sheets designed on the inventor from the simplest draw to the most complicated.
- Writing down all the tasks and the features which will be accomplished during our progress.
- Checking for the availability of all the components such as the tablet, sensors, shields, camera and others.
- Designing a flowchart for the application and listing all the options which are involved in the design.

- Identifying the Arduino software and being familiar with all the required programs for the interface by using Arduino IDE 1.8.
- Working on the image processing program and its interface with the Arduino and the windows application.
- Starting by the implementation of the programs for the application using visual studio.
- Testing the interface between the Arduino and the servo motors that allows the adjustment of the desk body.
- Linking the Arduino, tablet, camera, shields, sensors, and motors all together.
- Recording many voice orders using the voice recognition module. Such as, light on/off, music on/off, call home, open book stair and others.
- Entering pictures for image processing system so the user and the visitors are easily identified.
- Implementing the desk body.
- Connecting all the components together.
- Testing the system in order to avoid problems concerning connections and interfaces.

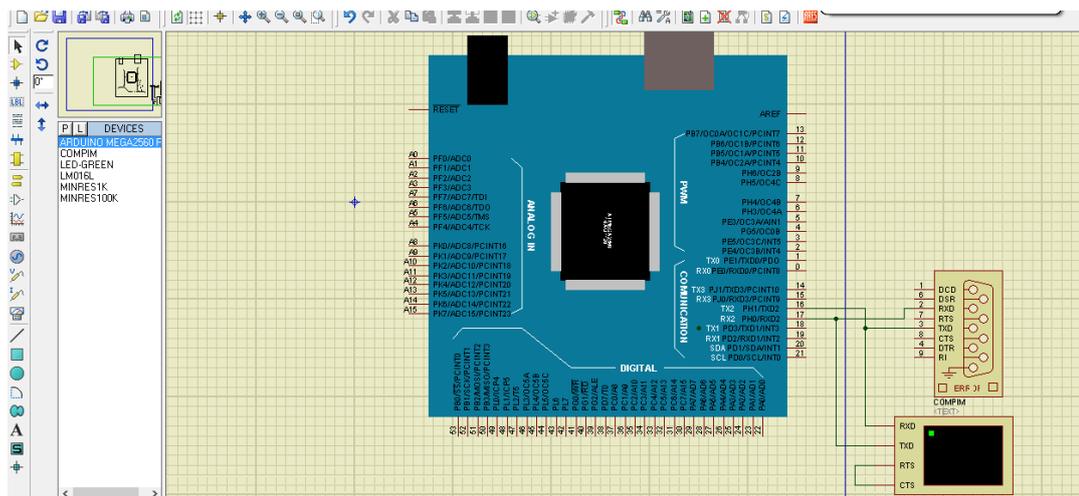


Figure 19 Simulation of Arduino Mega with GSM Module.

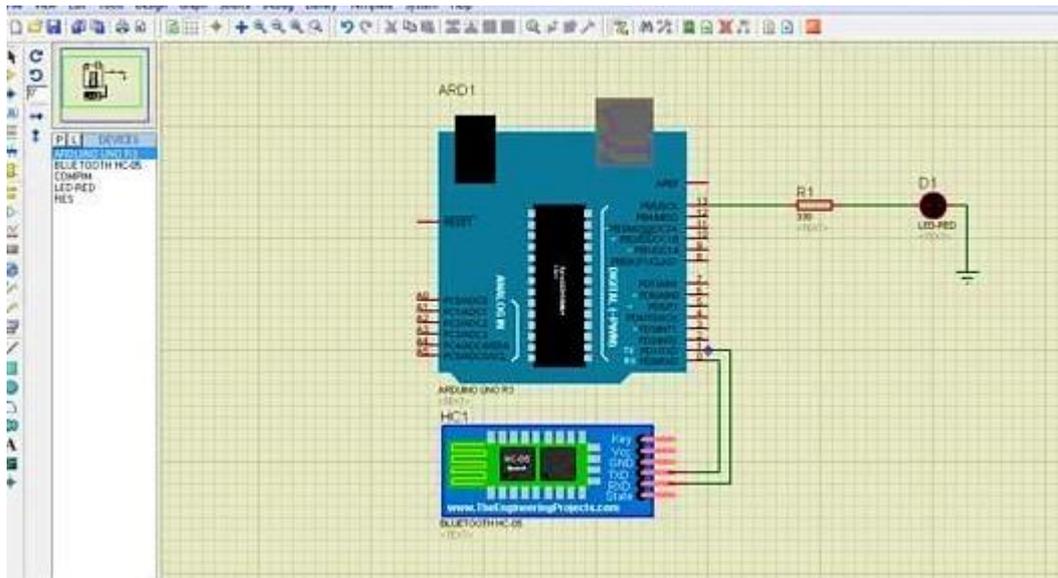


Figure 20 Simulation of Arduino UNO with Bluetooth Module.

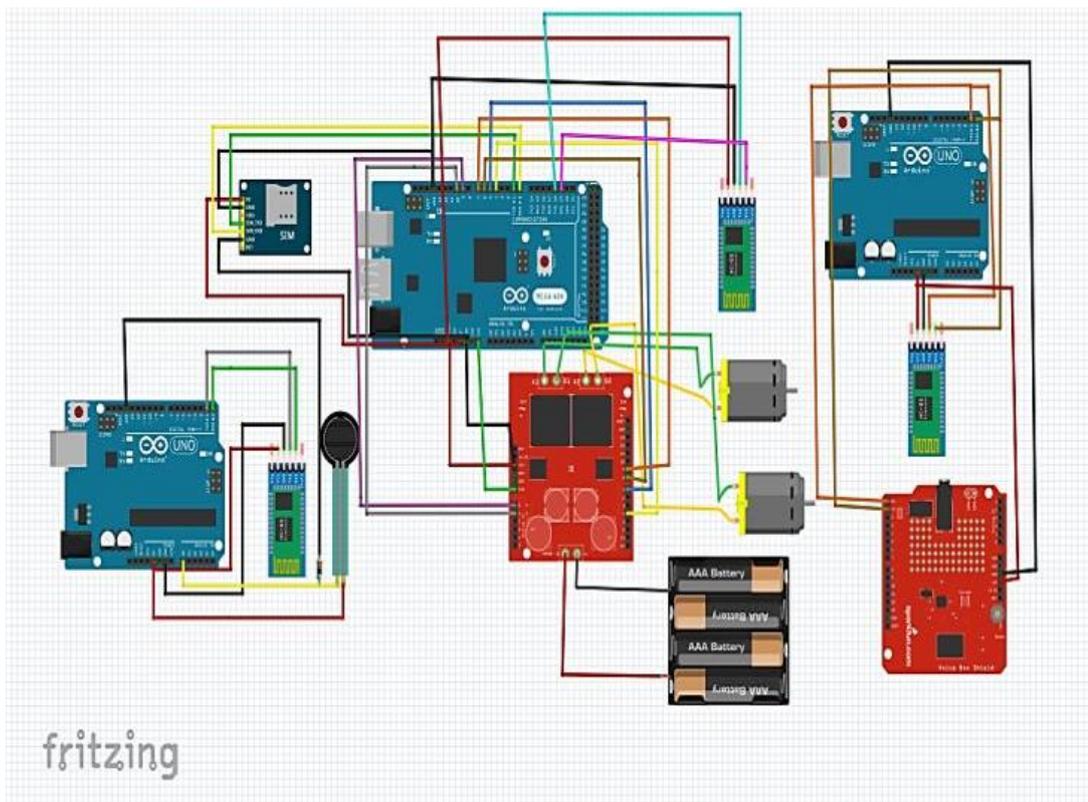


Figure 21 Circuit Schematic.

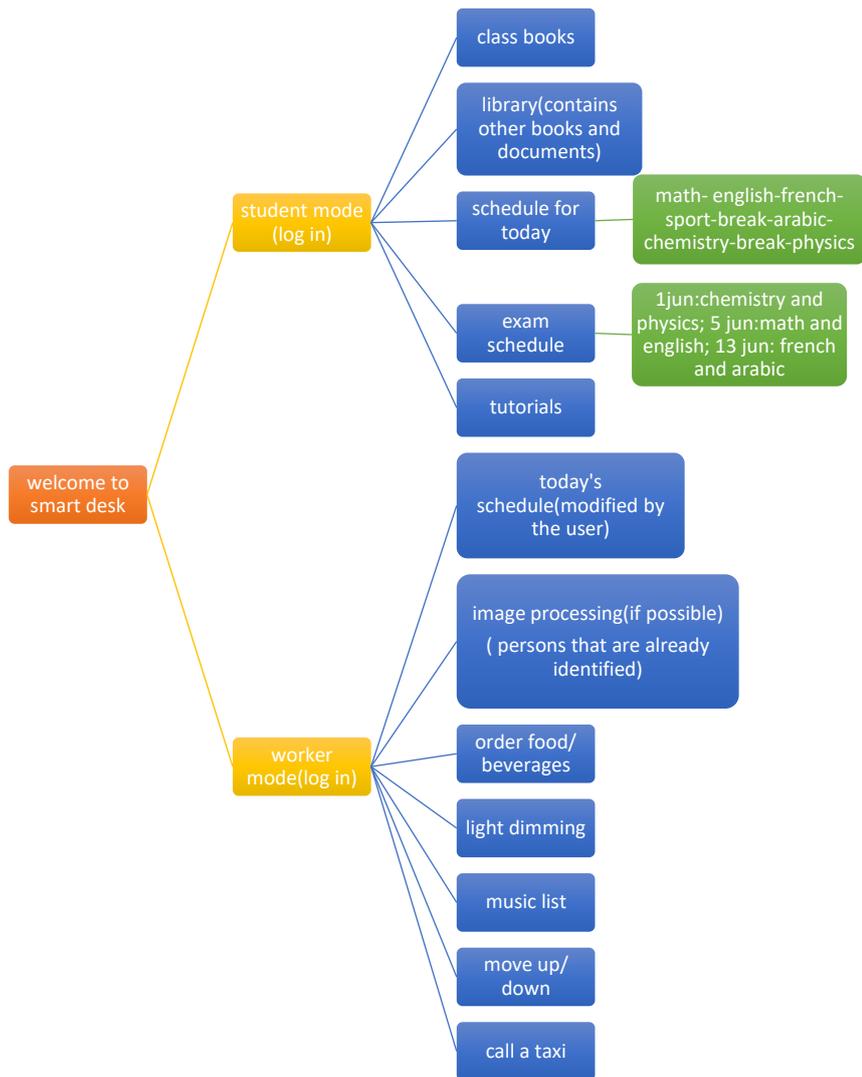


Figure 22 Windows Application Flowchart.

G. Cost Analysis

Table 2 Cost Description.

Item	Description	Quantity	Price (USD)	Total
Arduino	Mega	1	\$14.00	\$14.00
5Mp Camera	Arduino	1	\$60.00	\$60.00
GSM Module	Arduino	1	\$30.00	\$30.00

Table 2 (Cont.) Cost Description.

Voice Recognition Module	Arduino	1	\$48.00	\$48.00
Servo Motor	9G	1	\$4.00	\$4.00
8-channel Relay Module	Arduino	1	\$6.00	\$6.00
9V Adapter	Arduino Power Supply	1	\$6.00	\$6.00
5V Adapter	Relay Power Supply	1	\$5.00	\$5.00
Arduino Jumpers	M/F – 40/strip	2	\$2.00	\$4.00
Arduino Jumpers	M/M – 40/strip	1	\$1.67	\$1.67
Fluorescent LED	12V	1	\$7.00	\$7.00
Consumables	Wires, Heat Sinks, Electrical	1	\$10.00	\$10.00
Desk Body	Wood/Aluminum	1	\$200.00	\$200.00
Lenovo MIIX 310	Tablet	1	\$230.00	\$230.00
Raspberry Pi	Raspberry Pi 3	1	\$60.00	\$60.00
Chair	Desk chair	1	\$50.00	\$50.00
Pistons	Electric	4	\$140.00	\$560.00
Labor Cost				\$1620.00
Final Total Cost				\$3015.67

IV. RESULTS AND DISCUSSION

A. Testing and Validation

After we finish the software and hardware part, we start to implement all the components, modules, processors, and circuits on the desk. These parts are connected to the tablet via a serial and wireless communication. Then we test each feature as follows:

The motion of the desk with respect to the user; where it moves up to attend a maximum height of 110cm when the user had been sitting for a certain period of time allowing the Arduino, which is connected to the chair, to give the order to the speakers to say the following: “you’ve been sitting for one hour and a half, would you like to stand up”. This is done via a Bluetooth communication between the tablet and the Arduino.

Voice recognition is active and can detect many orders whether as a respond to the questions of the desk such as: “it’s time for lunch what do you want to eat”, “your first meeting is at 10’o clock with Mr. James”, or as a regular order to the desk to do some actions as: “call MacDonald’s”, “ good morning desk”, “ desk close”, “move up”, “move down”.

The windows application is designed with a dual mode where the worker mode contains working features such as: call a taxi, play/stop and pause music, order food, light slider, notes, and up/ down movements. Besides that, the student mode contains learning features such as: class books, tutorials, exam’s schedule, and library. This application is used to set new events and notifications that the desk can state them verbally to remind the user.

The image processing feature, where the camera placed at the middle of the desk, is able to capture user’s and guest’s pictures in real time with information about their

names and backgrounds. Once the desk is used for a second time by them, the camera is able to square the face and tell the name of the recognized person.

B. Results

As results, the desk is characterized by the following:

Table 3 Worker Mode Desk Features.

Worker Mode
Moving up/ down
Voice reminder system (Eating, drinking, meeting, leaving, calling...)
Verbal orders (Food, music, light control, move up, move down...)
Guest's and user's identification
Mobility (indoor, outdoor)

Table 4 Student Mode Desk Features.

Student Mode
Moving up/ down
School's schedule (course materials)
Exam's schedule
Class books and tutorials



Figure 23 Worker Mode Visual Basic Design Interface.

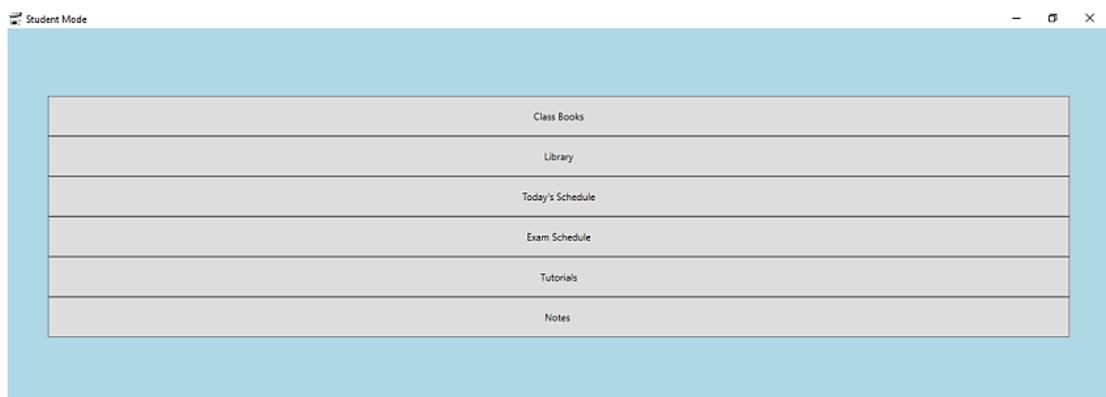


Figure 24 Student Mode Visual Basic Design Interface.

C. Discussion

Representatives think the most about their chance, well-being and profitability. Imagine a scenario in which there is an item that can guarantee every one of these requests in an extremely straightforward outline.

All organizations around the globe have a similar focus: to build specialist's profitability at the least conceivable cost and to expand efficiency premium work conditions must be fulfilled. Our item gives the client broad abilities by utilizing the application that enables the client to take control of all parameters.

As specified in the previous chapters of the report, Smart work area has unique however related objectives that make work all the more fascinating for specialists. It weighs on the capacity to control and oversee time by presenting two basic highlights that are voice recognition and image processing.

Furthermore, it ensures that the client deals with its well-being by utilizing the update framework that tells the client to drink, eat, to stand up and to do some stretching exercises since sitting for delayed periods can make harm to our bodies.

V. CONCLUSIONS AND FUTURE WORK

A. Conclusions

As mentioned in the previous chapters, Smart desk has different but related goals that make work more interesting for workers. It stresses on the ability to control and manage time by introducing two essential features that are voice recognition and image processing. In addition, it makes sure that the user takes care of its health by using the reminder system that notifies the user to drink, order food, to stand up and to do some stretching exercise since sitting for prolonged periods can cause damage to our bodies.

The desk is unique because of its ability to respond to user's commands and interact with him. Also, it is able to replace traditional learning techniques that is adopted by the Lebanese schooling system, by a technological method that allows the student to get away of all the heavy books and follow the easiest way of learning by just a small screen that contains tutorials, class books, exams, and materials schedule. Thus say hello to a desk that replaces your assistant, school bag, books, and mobile phone.

B. Future Work

The mix between the equipment and the product makes the item an artful culmination. It is a stage forward in the realm of innovation that leads work to be more mindful, profitable, and imaginative.

Additionally, it is much prescribed for some organizations and understudies in view of its unmistakable highlights in a single plan. The innovation is presently incorporated in a work area body used to control every one of the gadgets in a room with the end goal that the machines can improve the specialist's wellbeing much in view of its particular knowledge framework.

This framework will be stretched out to a wide territory, with the goal that the client will be fulfilled and can have a decent efficiency in the work. Likewise, image processing framework can be utilized locally, as well as at the outside to give the client a fundamental foundation about the general population that meets with them.

The progression will be extended to attain other furniture and office supplies such as sofas, beds, kitchen, pillows, and closets to integrate user's life in the technological world.

VI. REFERENCES

BOOKS

CRUISE, R. The height-adjustable, standing Stir Kinetic Desk, **Automation System**,4th Edition, 2001.

MICHEAL, C. (n.d.). The Different Kinds of Standing Desks. **How to Geek**, 5th Edition, 2020.

ELECTRONIC SOURCES

AUTO DESK. (2000). Retrieved November 10, 2020, from Auto Desk Inventor:
<https://www.autodeskinventor.com/>

R, E., & Laskowski, M. (n.d.). “What are the risks of sitting too much?”. Retrieved septembre 4, 2020, from mayoclinic:
<https://www.mayoclinic.org/healthylifestyle/adult-health/expert-answers/sitting>

ZHON, O. (n.d.). Voice Recognition module. Retrieved january 2, 2021, from instructables:
<https://www.instructables.com/id/definition/voicerecognitionmodule/>

URL-1 “Programing world”. (2005). Retrieved november 15, 2020, from Instructables: <https://www.instructables.com/id/programmingworld>

URL-2 Retrieved december 4, 2020, from arduino:
<https://www.arduino.cc/definition/reference/arduinomega>

URL-3 Retrieved december 15, 2020, from electrotech:
<https://www.electrotech.com/products/pressuresensor>

URL-4 Retrieved december 27, 2020, from instructables:
<https://www.instructables.com/id/interfacing-SIM900A-GSM-Modem-with-Arduino/>

APPENDICES

Image Processing Program

```
using System;
using System.Diagnostics;
using Emgu.CV.Structure;

namespace Emgu.CV
{
    /// <summary>
    /// An object recognizer using PCA (Principle Components Analysis)
    /// </summary>
    [Serializable]
    public class EigenObjectRecognizer
    {
        private Image<Gray, Single>[] _eigenImages;
        private Image<Gray, Single> _avgImage;
        private Matrix<float>[] _eigenValues;
        private string[] _labels;
        private double _eigenDistanceThreshold;

        /// <summary>
        /// Get the eigen vectors that form the eigen space
```

```
/// </summary>
```

```
/// <remarks>The set method is primary used for deserialization, do not attempts  
to set it unless you know what you are doing</remarks>
```

```
public Image<Gray, Single>[] EigenImages
```

```
{  
    get { return _eigenImages; }  
    set { _eigenImages = value; }  
}
```

```
/// <summary>
```

```
/// Get or set the labels for the corresponding training image
```

```
/// </summary>
```

```
public String[] Labels
```

```
{  
    get { return _labels; }  
    set { _labels = value; }  
}
```

```
/// <summary>
```

```
/// Get or set the eigen distance threshold.
```

```
/// The smaller the number, the more likely an examined image will be treated as  
unrecognized object.
```

```
/// Set it to a huge number (e.g. 5000) and the recognizer will always treated the  
examined image as one of the known object.
```

```
/// </summary>
```

```
public double EigenDistanceThreshold
```

```

    {
        get { return _eigenDistanceThreshold; }
        set { _eigenDistanceThreshold = value; }
    }

    /// <summary>
    /// Get the average Image.
    /// </summary>

    /// <remarks>The set method is primary used for deserialization, do not attempts
to set it unless you know what you are doing</remarks>

    public Image<Gray, Single> AverageImage
    {
        get { return _avgImage; }
        set { _avgImage = value; }
    }

    /// <summary>
    /// Get the eigen values of each of the training image
    /// </summary>

    /// <remarks>The set method is primary used for deserialization, do not attempts
to set it unless you know what you are doing</remarks>

    public Matrix<float>[] EigenValues
    {
        get { return _eigenValues; }
        set { _eigenValues = value; }
    }

```

```

private EigenObjectRecognizer()
{
}

/// <summary>
/// Create an object recognizer using the specific training data and parameters, it
will always return the most similar object
/// </summary>
/// <param name="images">The images used for training, each of them should be
the same size. It's recommended the images are histogram normalized</param>
/// <param name="termCrit">The criteria for recognizer training</param>
public EigenObjectRecognizer(Image<Gray, Byte>[] images, ref
MCvTermCriteria termCrit)
    : this(images, GenerateLabels(images.Length), ref termCrit)
{
}

private static String[] GenerateLabels(int size)
{
    String[] labels = new string[size];
    for (int i = 0; i < size; i++)
        labels[i] = i.ToString();
    return labels;
}

```

```

    /// <summary>

    /// Create an object recognizer using the specific training data and parameters, it
    will always return the most similar object

    /// </summary>

    /// <param name="images">The images used for training, each of them should be
    the same size. It's recommended the images are histogram normalized</param>

    /// <param name="labels">The labels corresponding to the images</param>

    /// <param name="termCrit">The criteria for recognizer training</param>

    public EigenObjectRecognizer(Image<Gray, Byte>[] images, String[] labels, ref
    MCvTermCriteria termCrit)

        : this(images, labels, 0, ref termCrit)

    {

    }

    /// <summary>

    /// Create an object recognizer using the specific training data and parameters

    /// </summary>

    /// <param name="images">The images used for training, each of them should be
    the same size. It's recommended the images are histogram normalized</param>

    /// <param name="labels">The labels corresponding to the images</param>

    /// <param name="eigenDistanceThreshold">

    /// The eigen distance threshold, (0, ~1000].

    /// The smaller the number, the more likely an examined image will be treated as
    unrecognized object.

    /// If the threshold is &lt; 0, the recognizer will always treated the examined
    image as one of the known object.

```

```

/// </param>

/// <param name="termCrit">The criteria for recognizer training</param>

public EigenObjectRecognizer(Image<Gray, Byte>[] images, String[] labels,
double eigenDistanceThreshold, ref MCvTermCriteria termCrit)
{
    Debug.Assert(images.Length == labels.Length, "The number of images should
equals the number of labels");

    Debug.Assert(eigenDistanceThreshold >= 0.0, "Eigen-distance threshold
should always >= 0.0");

    CalcEigenObjects(images, ref termCrit, out _eigenImages, out _avgImage);
    /*
    _avgImage.SerializationCompressionRatio = 9;
    foreach (Image<Gray, Single> img in _eigenImages)
        //Set the compression ration to best compression. The serialized object can
therefore save spaces
        img.SerializationCompressionRatio = 9;
    */
    _eigenValues = Array.ConvertAll<Image<Gray, Byte>,
Matrix<float>>(images,
        delegate(Image<Gray, Byte> img)
        {
            return new Matrix<float>(EigenDecomposite(img, _eigenImages,
_avgImage));
        });
    _labels = labels;
    _eigenDistanceThreshold = eigenDistanceThreshold;
}

```

```

#region static methods

/// <summary>
/// Caculate the eigen images for the specific training image
/// </summary>

/// <param name="trainingImages">The images used for training </param>
/// <param name="termCrit">The criteria for training</param>
/// <param name="eigenImages">The resulting eigen images</param>
/// <param name="avg">The resulting average image</param>

public static void CalcEigenObjects(Image<Gray, Byte>[] trainingImages, ref
MCvTermCriteria termCrit, out Image<Gray, Single>[] eigenImages, out
Image<Gray, Single> avg)
{
    int width = trainingImages[0].Width;
    int height = trainingImages[0].Height;

    IntPtr[] inObjs = Array.ConvertAll<Image<Gray, Byte>,
IntPtr>(trainingImages, delegate(Image<Gray, Byte> img) { return img.Ptr; });

    if (termCrit.max_iter <= 0 || termCrit.max_iter > trainingImages.Length)
        termCrit.max_iter = trainingImages.Length;

    int maxEigenObjs = termCrit.max_iter;

#region initialize eigen images
    eigenImages = new Image<Gray, float>[maxEigenObjs];
    for (int i = 0; i < eigenImages.Length; i++)

```

```

        eigenImages[i] = new Image<Gray, float>(width, height);

        IntPtr[] eigObjs = Array.ConvertAll<Image<Gray, Single>,
IntPtr>(eigenImages, delegate(Image<Gray, Single> img) { return img.Ptr; });

        #endregion

        avg = new Image<Gray, Single>(width, height);

        CvInvoke.cvCalcEigenObjects(
            inObjs,
            ref termCrit,
            eigObjs,
            null,
            avg.Ptr);
    }

    /// <summary>
    /// Decompose the image as eigen values, using the Serial call response:
    int firstSensor = 0; // first analog sensor
    int secondSensor = 0; // second analog sensor
    int thirdSensor = 0; // digital sensor
    int inByte = 0; // incoming serial byte

    void setup() {
        // start serial port at 9600 bps:
        Serial.begin(9600);
        while (!Serial) {
            ; // wait for serial port to connect. Needed for native USB port only

```

```

}

pinMode(2, INPUT); // digital sensor is on digital pin 2

establishContact(); // send a byte to establish contact until receiver responds
}

void loop() {

// if we get a valid byte, read analog ins:
if (Serial.available() > 0) {

// get incoming byte:
inByte = Serial.read();

// read first analog input, divide by 4 to make the range 0-255:
firstSensor = analogRead(A0) / 4;

// delay 10ms to let the ADC recover:
delay(10);

// read second analog input, divide by 4 to make the range 0-255:
secondSensor = analogRead(1) / 4;

// read switch, map it to 0 or 255L
thirdSensor = map(digitalRead(2), 0, 1, 0, 255);

// send sensor values:
Serial.write(firstSensor);

Serial.write(secondSensor);

Serial.write(thirdSensor);

}

}

```

```

void establishContact() {
    while (Serial.available() <= 0) {
        Serial.print('A'); // send a capital A
        delay(300);
    }
}

int bgcolor;           // Background color
int fgcolor;          // Fill color
Serial myPort;        // The serial port
int[] serialInArray = new int[3]; // Where we'll put what we receive
int serialCount = 0;  // A count of how many bytes we receive
int xpos, ypos;       // Starting position of the ball
boolean firstContact = false; // Whether we've heard from the microcontroller

void setup() {
    size(256, 256); // Stage size
    noStroke();    // No border on the next thing drawn

    // Set the starting position of the ball (middle of the stage)
    xpos = width/2;
    ypos = height/2;
}

```

```

// Print a list of the serial ports for debugging purposes
// if using Processing 2.1 or later, use Serial.printArray()
println(Serial.list());

// I know that the first port in the serial list on my mac
// is always my FTDI adaptor, so I open Serial.list()[0].
// On Windows machines, this generally opens COM1.
// Open whatever port is the one you're using.
String portName = Serial.list()[0];
myPort = new Serial(this, portName, 9600);
}

void draw() {
  background(bgcolor);
  fill(fgcolor);
  // Draw the shape
  ellipse(xpos, ypos, 20, 20);
}

void serialEvent(Serial myPort) {
  // read a byte from the serial port:
  int inByte = myPort.read();
  // if this is the first byte received, and it's an A,
  // clear the serial buffer and note that you've

```

```

// had first contact from the microcontroller.

// Otherwise, add the incoming byte to the array:
if (firstContact == false) {
    if (inByte == 'A') {
        myPort.clear();    // clear the serial port buffer
        firstContact = true; // you've had first contact from the microcontroller
        myPort.write('A'); // ask for more
    }
}
else {
    // Add the latest byte from the serial port to array:
    serialInArray[serialCount] = inByte;
    serialCount++;

    // If we have 3 bytes:
    if (serialCount > 2 ) {
        xpos = serialInArray[0];
        ypos = serialInArray[1];
        fgcolor = serialInArray[2];

        // print the values (for debugging purposes only):
        println(xpos + "\t" + ypos + "\t" + fgcolor);

        // Send a capital A to request new sensor readings:
        myPort.write('A');
    }
}

```

```

        // Reset serialCount:
        serialCount = 0;
    }
}

Application namespace
using System;
using System.Collections.Generic;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Threading.Tasks;
using System.Windows;

namespace SmartDeskApp
{
    /// <summary>
    /// Interaction logic for App.xaml
    /// </summary>
    public partial class App : Application
    {
    }
} using System;
using System.Windows;
using System.Collections.Generic;

```

```

namespace SmartDeskApp

List box

{
    /// <summary>
    /// Interaction logic for listBox1.xaml
    /// </summary>

    public partial class listBox1 : Window
    {
        public listBox1()
        {
            InitializeComponent();

            List<TodoItem> items = new List<TodoItem>();

            items.Add(new TodoItem() { Title = "Complete this WPF tutorial",
Completion = 45 });

            items.Add(new TodoItem() { Title = "Learn C#", Completion = 80 });

            items.Add(new TodoItem() { Title = "Wash the car", Completion = 0 });

            lbTodoList.ItemsSource = items;
        }

        private void lbTodoList_SelectionChanged(object sender,
System.Windows.Controls.SelectionChangedEventArgs e)
        {
            if (lbTodoList.SelectedItem != null)

                this.Title = (lbTodoList.SelectedItem as TodoItem).Title;
        }
    }
}

```

```
private void btnShowSelectedItem_Click(object sender, RoutedEventArgs e)
{
    foreach (object o in lbTodoList.SelectedItems)
        MessageBox.Show((o as TodoItem).Title);
}
```

```
private void btnSelectLast_Click(object sender, RoutedEventArgs e)
{
    lbTodoList.SelectedIndex = lbTodoList.Items.Count - 1;
}
```

```
private void btnSelectNext_Click(object sender, RoutedEventArgs e)
{
    foreach (object o in lbTodoList.Items)
    {
        if ((o is TodoItem) && ((o as TodoItem).Title.Contains("C#")))
        {
            lbTodoList.SelectedItem = o;
            break;
        }
    }
}
```

```
private void btnSelectAll_Click(object sender, RoutedEventArgs e)
{
```

```

        foreach (object o in lbTodoList.Items)
            lbTodoList.SelectedItems.Add(o);
    }
}

public class TodoItem
{
    public string Title { get; set; }
    public int Completion { get; set; }
}
}

```

List view

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Data;
using System.Windows.Documents;
using System.Windows.Input;
using System.Windows.Media;
using System.Windows.Media.Imaging;
using System.Windows.Shapes;

```

```

namespace SmartDeskApp
{
    /// <summary>
    /// Interaction logic for listview1.xaml
    /// </summary>
    public partial class listview1 : Window
    {
        public listview1()
        {
            InitializeComponent();

            List<User3> items = new List<User3>();

            items.Add(new User3() { Name = "John Doe", Age = 42, Mail = "john@doe-
family.com" });

            items.Add(new User3() { Name = "Jane Doe", Age = 39, Mail = "jane@doe-
family.com" });

            items.Add(new User3() { Name = "Sammy Doe", Age = 13, Mail =
"sammy.doe@gmail.com" });

            lvDataBinding.ItemsSource = items;
        }
        public class User3
        {
            public string Name { get; set; }

            public int Age { get; set; }

            public string Mail { get; set; }
        }
    }
}

```

```

    }
}
}
Main window
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Data;
using System.Windows.Documents;
using System.Windows.Input;
using System.Windows.Media;
using System.Windows.Media.Imaging;
using System.Windows.Navigation;
using System.Windows.Shapes;

namespace SmartDeskApp
{
    /// <summary>
    /// Interaction logic for MainWindow.xaml
    /// </summary>
    public partial class MainWindow : Window

```

```

{
    public MainWindow()
    {
        InitializeComponent();
    }

    private void Button_Click(object sender, RoutedEventArgs e)
    {
        // Window2 w1 = new Window2();

        // w1.Show();

        // Window3 w11 = new Window3();

        //w11.Show();

        WL w5 = new WL();

        w5.Show();

        //ES w6 = new ES();

        //w6.Show();

    }

    private void Button_Click_1(object sender, RoutedEventArgs e)
    {
        SL w2 = new SL();

        w2.Show();

    }
}

```

```

}

Media player

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Shapes;

using Microsoft.Win32;

namespace SmartDeskApp
{
    /// <summary>
    /// Interaction logic for MediaP.xaml
    /// </summary>
    public partial class MediaP : Window
    {

```



```

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Shapes;

using System.Xml.Serialization;

namespace SmartDeskApp
{
    /// <summary>
    /// Interaction logic for WM.xaml
    /// </summary>
    public partial class WM : Window
    {
        SerialPort serialPort1;

        public WM()
        {
            InitializeComponent();

            serialPort1 = new SerialPort();
        }
    }
}

```

```

serialPort1.PortName = "COM23";

serialPort1.BaudRate = 9600;

serialPort1.Parity = System.IO.Ports.Parity.None;

serialPort1.DataBits = 8;

serialPort1.StopBits = System.IO.Ports.StopBits.One;

serialPort1.Handshake = System.IO.Ports.Handshake.None;

serialPort1.ReadTimeout = 500;

serialPort1.WriteTimeout = 500;

serialPort1.DtrEnable = true;

serialPort1.RtsEnable = true;

}

private MediaPlayer mediaPlayer = new MediaPlayer();

private void Food_Click(object sender, RoutedEventArgs e)
{
    try { serialPort1.Open();

        serialPort1.WriteLine("F");

        serialPort1.Close();

    }

    catch { MessageBox.Show("Serial not open"); }
}

```

```

private void MusicP_Click(object sender, RoutedEventArgs e)
{
    OpenFileDialog openFileDialog = new OpenFileDialog();
    openFileDialog.Filter = "MP3 files (*.mp3)|*.mp3|All files (*.*)|*.*";
    if (openFileDialog.ShowDialog() == true)
    {
        mediaPlayer.Open(new Uri(openFileDialog.FileName));
        // C:\Users\Roukoz murr\Downloads\Dreezy- Up and Down Lyrics.mp3
        mediaPlayer.Play();
    }
}

private void MusicS_Click(object sender, RoutedEventArgs e)
{
    mediaPlayer.Close();
}

private void textE_Click(object sender, RoutedEventArgs e)
{
    richtextbox2 r = new richtextbox2();
    r.Show();
}
}
}

```

```

//private void Button_Click_3(object sender, RoutedEventArgs e)
//{
//    OpenFileDialog openFileDialog = new OpenFileDialog();
//    openFileDialog.Filter = "MP3 files (*.mp3)|*.mp3|All files (*.*)|*.*";
//    if (openFileDialog.ShowDialog() == true)
//    {
//        mediaPlayer.Open(new Uri(openFileDialog.FileName));
//        // C:\Users\Roukoz murr\Downloads\Dreezy- Up and Down Lyrics.mp3
//        mediaPlayer.Play();
//    }
//}

//private void Button_Click_4(object sender, RoutedEventArgs e)
//{
//    mediaPlayer.Close();
//}
//}

```

Student mode

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows;
using System.Windows.Controls;

```

```

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Shapes;

namespace SmartDeskApp
{
    /// <summary>
    /// Interaction logic for SM.xaml
    /// </summary>
    public partial class SM : Window
    {
        public SM()
        {
            InitializeComponent();
        }

        private void Button_Click(object sender, RoutedEventArgs e)
        {
            CB w1 = new CB();

            w1.Show();
        }
    }
}

```

```
private void Button_Click_1(object sender, RoutedEventArgs e)
{
    L w = new L();
    w.Show();

}
```

```
private void Button_Click_2(object sender, RoutedEventArgs e)
{
    T ww1 = new T();
    ww1.Show();
}
```

```
private void Button_Click_3(object sender, RoutedEventArgs e)
{
    // TS www1 = new TS();
    Window2 p = new Window2();
    p.Show();

}
```

```
private void Button_Click_4(object sender, RoutedEventArgs e)
{
    // ES wwww1 = new ES();
    Window2 wwww1 = new Window2();
```

```
        www1.Show();  
    }  
  
    private void textE_Click(object sender, RoutedEventArgs e)  
    {  
        richtextbox2 r = new richtextbox2();  
        r.Show();  
    }  
}  
}
```

RESUME

Objective

- Seeking an engineering position that will incorporate my graduate studies in electronics and programming engineering.

Work Experience

Engineering customer support expert
Zinaty Electronic Center ,Zahle,Lebanon

August 2017-July 2018

Designed a Pic Microcontroller projects: water tank level indicator, 7segment display, garage control, alarm and others.

Designed a Locker by LabView: a locker that can open by entering a password on the LCD touch screen and it is accompanied by an alarm system.

Designed an Automatic Voltage Regulator: an AC voltage regulation system to protect the electronic devices from an over or under load.

Produced an Audio equalizer: equalization mode to control the audio by adjusting the middle, the base and the tribal.

Produced a Three output DC power supply: a supply that transform AC to DC voltage and gives you three outputs two are variables and the third is a 5v fixed output.

Studied an Uninterrupted Power Supply: known as UPS it is an inverter with an adapter system, it transform the AC to DC. The DC voltage remains in the batteries then it is consumed after its conversion to DC voltage.

Engineering Project Leader
EDUCAT, Lille, France

August 2018-December 2018

Designed and Produced a Mechanical sensor system for a Wheeling chair: a mechanical design that contains sensors and electrical boards to prevent the user from any obstacle by estimating the distance and this system is applicable to all wheeling chair

IT Consultant, **September 2016-january 2017**
American University of Science and Technology,Zahle,Lebanon

Leading and Managing university's laboratories

Studied a Smart pillow: a pillow that wakes you up by vibration or music and enhances your mood, it can be controlled by an application from your phone.

Designed a PLC controlled Mixer: a system that is controlled by a plc and have the capacity to mix to liquid together with corresponding to the orders of the plc for the pumps, valves and other outputs.

Produced a Multifunctioning car: a smart robot car that avoids obstacles and it can be controlled from your phone via Bluetooth or using an infrared remote control.

Conferences

Participation IEEE Conference - Information & Communication Systems, N. Halabi, A. Hussein ,the Beirut, Lebanon, April 2016.

Participation Arduino Robotics conference , American university of science and technology, zahle, Lebanon, october 2017.

Participation Denise Marcheznay-Vice President CGI in France, ISEN, Lille, september 2018.

Education

Master in Electrical and Electronics Engineering **September 2020**
GPA 3.44/4.00
Istanbul Aydin University, Florya, Istanbul, Turkey

Bachelor of Science (BS) Degree in Computer **December 2018**
and Communications Engineering with emphasis
in mechatronics Engineering (CCEM) GPA 3.51/4.00
American University of Science and Technology, Beirut, Lebanon

Erasmus student exchange program **August 2018**
computer and automation
Lille Catholic University, Higher Institute of Electrical
and Numerical, Lille, Yncrea, France

Certifications

1st project in innovation and entrepreneurship **June 2019**
INJAZ Innovation and Entrepreneurship

14th exposition for engineering and computer science **May 2018**
American University of Science and Technology, Beirut, Lebanon

Bank of Beirut Innovation and entrepreneurship **April 2017**
American University of Science and Technology, Beirut, Lebanon

1st Industrial control project exposition **December 2017**
American University of Science and Technology, Beirut, Lebanon

Speech Contest Certificate **April 2016**
American University of Science and Technology, Beirut, Lebanon

Competency in English (ECCE) **December 2015**
University of Michigan, Michigan, United States of America

Technical Skills

Programming Languages : C/C++, Assembly, HTML, CSS, PHP, IOS
Circuit Simulation : MATLAB, Lab View, Proteus, PCB express.
PLC : Programming and implementation.
Networking : Design, simulation, and implementation.
Mechanics : Auto Desk Inventor, Auto Cad

Personal Description

Detail oriented and thoughtful professional prepared to excel in the field of communications and mechatronics engineering.

Hard worker and fast learner.

Self-motivated individual and enjoy team work.

Fluent in English, French, Turkish and Arabic spoken and written.