**DESIGN AND IMPLEMENTATION OF DATABASE COMMUNICATION INTERFACE BETWEEN EMBEDDED SYSTEMS AND COMPUTER SYSTEMS**

**BY**

**OBODOMECHINE JAKE E.**

**F/HD/18/3410003**

Abstract

This project is developed by using a couple of programing languages (python, php, Arduino cc) and a local server (Xampp)to ease control the function and data exchange of an embedded system, the project uses an rfid access control system and a BMI system as test cases, previously data CRUD(create, read, update, delete) operations had to be done form within the source code, this opens avenues for code errors to occur from deleting the wrong data, to damaging code syntax or change general function of the machine as a whole, also this does not make it usable to people with little or no idea about the work about of a code base, This project helps reduce, if not eradicated the need to go into the source code, all CRUD operations are done from the provided control interface. The system runs by the process of comparing data supplied to data available in the database of the Xampp local server, data present can be managed from the GRAPHICAL USER INTERFACE GUI, which is built using PHP, HTML, CSS, JQUERY AND BOOTSTRAP. First necessary data components are saved to the database by admins through the provided interface, this part is important because it is the power house if the system if data isn’t saved, it can’t be retrieved (BMI) or compared (Access control system), data exchange between the embedded system and user interface/database is done with the python code. With these set of languages and database, any registered or authorized person is able to control and regulate the function of the embedded system at a button click and without the need of in-depth technological experience, reduces work load on a technical personnel and since little technical experience is needed, this leads to an increase the general situations where embedded system can be utilized

CHAPTER ONE

**INTRODUCTION**

* 1. **BACKGROUND OF STUDY**

Digital systems can be classified into two categories: general-purpose systems and application-specific systems under which we have the embedded systems. General-purpose systems can be programmed to run a variety of applications, i.e., they are not designed for any special application and can perform most common computing task, an example is a Desktop/laptop system this is a system capable of performing various tasks and is customizable in software, it can perform lots of task with little or no constraints on power, performance or cost also the are contained in themselves and no part of a larger system, as opposed to application-specific systems. Embedded systems are performance specific systems which can be part of a larger host system and perform specific functions within the host system, an embedded system is implemented partly on software and partly on hardware. When standard microprocessors, microcontrollers or DSP processors are used, specialization of an embedded system for a particular application consists primarily on the specialization of software,

The synergy created from the combination of these two systems can be called the next step in digital systems, being able to make them communicate and share data would push the boundary of science one step further, P. J. Plauger (1989) has argued that what makes embedded and real-time systems development special is a preoccupation with time. Timing, speed, sequencing, and synchronization.

This project looks at the connection of these two digital systems, the embedded system which is required to perform a specific task (measure height or weight and manage student access within the school environment) to the desktop/laptop system which holds the database that guides the decision of the embedded system through an “interface”.

Speaking on the focus for this project which involves the designing of an “interface”, this offers the services of a bridge/link between our embedded system and host computer it could come as a wired or wireless medium, for instance connection between your mobile phone and your PC, your mobile device is an embedded system and your USB cord is an interface.

**1.1.1 INTERFACE**

Computers or other device connected to a computer network are called **network gateway** (which is communication with the communication module of embedded device presented) ,they may offer information ,resources ,services, and applications to users or embedded systems on the same nodes ,commuters that participate in networks and use internet protocols can be called IP hosts , Specifically, computers participating in the Internet are called Internet hosts, sometimes Internet nodes.

Network hosts that participate in applications that use the client-server model of computing are classified as server or client systems. Network hosts may also function as nodes in peer-to-peer applications, in which all nodes share and consume resources in an equipotent manner. The network host serves as the control center of the embedded system. It can be used to:

• control the data acquisition unit (DAQ), in this case the embedded system.

• provide a graphical user interface (GUI) for the user to operate the system (with use of web GUI based on http protocol)

• store the acquired information on mass storage devices

• provide the tools to display and/or analyze the acquired data in real or non-real time

As in this case where we interface an embedded system to a PC to improve storage capability, utilize tools provided on a web GUI to analyze and exchange data.

**1.1.2 DATABASE**

These are basically containers for the data, they are computer structures that save, organize, update, manage, protect, and deliver data and a system that contains databases is a database management system DBM, the relational database is the most utilized kind of database, where digital data/information is stored in rows, columns, and tables which are indexed to ease search for relevant information through SQL queries.

Databases have played significant parts in recent evolution of computers, computer programs produced in the 1950s were used mostly for calculators and data like names, phone, numbers were counted as leftovers so focus was placed on languages and algorithms until they became commercial entities and business people utilized them, then the leftovers became Important to them creating the need for an ordered means of storage, granting a significant number of individual or connected devices capability to capture, store and analyze data for the purpose of making decisions like the embedded system needs the pc database. On each device, information is collected and filtered before it is shared with other systems. Ultimately, the information originating from these devices will be essential to gain insight and inform the decisions of the embedded system.

In the pursuit for better and evolved gadgets, improvements need to be made towards realizing this device, which arises the need for a means to interact with large amount of data that a user has to filter through or manage e.g. database of embedded systems used for access control must keep track of every entrance and exit from every access point. With the increase in information to be managed by embedded systems, it becomes important to store and manage data more efficiently and in a uniform manner, Current Methods used for storing and manipulating data objects in embedded and real-time systems are ad hoc, in this system data objects are manipulated as internal data structures. That is, data management of an embedded system is built as part of the overall system, this is not cost-effective with respect to design, implementation and verification e.g. it would be far expensive to place a full database system or interface panel in a school organization with over 20 entry points, in addition, this technique would make it quite difficult to link the devices to other embedded systems or larger database with central access.

Connecting an embedded system to a computer gives it the positive edge of the above-mentioned set of focus granting the system power to provide support for storage and manipulation of data and gives users some significant edges:

1. Reduction of development costs due to the reuse of database systems.
2. Improvement of quality in the design of embedded systems since the database provides support for consistent and safe manipulation of data and at the same time reducing data footprint in the embedded system memory,
3. Increased ease of operation as operations come down to the click of a button, the slide of an icon or simple character input, consequently, this improves the overall user-friendliness and utilization of the system in any field. Making it possible to tailor it to any profession or place of utilization.

This project presents the utilization of a database communication interface to govern the manipulation of data in the embedded systems of a BMI (Body Mass Index) and Access control system using a valid RFID chip as a means of identification and record insertion, thereby overcoming most of the shortcomings of a compact embedded system.

**1.1.3 EMBEDDED SYSTEM**

Embedded systems as stated earlier are systems designed for a specific purpose, they combination of hardware, software and mechanical components, many embedded systems consist of small, computerized parts within a larger device that serves a more general purpose. Some applications of embedded system are in the automotive ﬁled as control system or A washing machine, from an embedded systems point of view has:

1. Hardware: Buttons, Display & buzzer, electronic circuitry.
2. Software: It has a chip on the circuit that holds the software which drives controls & monitors the various operations possible.
3. Mechanical Components: the internals of a washing machine which actually wash the clothes control the input and output of water, the chassis itself.

They are mainly utilized for:

1. Data Collection/Storage/Representation
2. Data communication
3. Data signal processing
4. Monitoring
5. Control

With each new day new ways of implementing them in our daily lives are being discovered and researched.

**1.2 STATEMENT OF PROBLEM**

Machine and computers are utilized in almost all walks of life and not all personnel are tech-savvy, not everybody knows how to strip a system down to its embedded systems or work with a complicated interface and make corrections or improve on its contents.

Based on this it is necessary to formulate a problem to be solved or resolved in the digital design. This project concerns itself on creating a system which gives users an easy way to interact with the hardware of the system/embedded system no matter their professions, also to proffer solutions to the rigidity of an embedded system, rigidity includes the following

1. Inability to improve or upgrade on its hard-wired data.

2. Difficulty to take a backup of embedded files

3. In the event of a fault, troubleshooting is quite difficult.

4. They have limited hardware options.

**1.3 AIM OF THE PROJECT**

Design and implement a means of communication through an interface between embedded systems and computer systems

**1.4 OBJECTIVE OF THE PROJECT**

The objective of this project is to:

1. Design a web based control panel that offers the capability to guide the operation of an embedded system.
2. Create a database for efficient create, read, update and delete operations (CRUD) for both systems
3. In-corporate feature from I & II using an access control system and BMI system as a test case.

**1.5 SCOPE AND LIMITATION**

This project focus on the implementation of a system that eases data manipulation by connecting the hardware system to a computer system database which can be edited or updated without the need for hardware disassembling through a user interface. The principle system can control the movement of an Access programmable cantilever control system and also collect or update user records from a BMI system both using an RFID chip. Thus, the serial communication interfaces on the personal computer (PC) as a liaison between both microcontrollers in the embedded system and database (MySQL) on my computer.

We could have also made use of a wireless system (The Arduino Ethernet Shield with some other materials connects your Arduino to the internet), made use of other forms of embedded system applications or different and better means for database manipulation but due to resource constraints and Time. We have decided to limit ourselves to the use of a serial port USB system to connect the embedded system to the PC, using embedded systems in Access controls and BMI equipment, also utilizing Xampp Apache servers for database collection and storage.

**1.6 PROJECT ORGANIZATION**

This project work will be contained in five (5) chapters.

Chapter one carries the introduction, problem statement, aim, and objective, the scope of the study, limitation of the work and the project report organization. Chapter two carries the literature review of the work along with other reviews of the project and the proposed method used to achieve the system. Chapter three carries the project design methodology and steps which lead to the construction of the project, the schematic representation of the system and diagram used, the program code to run the system. Chapter four involved testing the project design, observation and Bill of the project. This chapter covers the description of the project, the step by step details of the setup of the system, the result, the structure chart of the system, and the database model of the system. Chapter five is the conclusion and recommendation of the project.

**CHAPTER TWO.**

**LITERATURE REVIEW**

**2.1 HISTORICAL BACKGROUND**

The abundance of embedded system utilization in our day to day life can be over looked but not taken for granted, they are a combination of hardware and software code design, and it is impressively being utilized for hardware and software analysis and effective operation. From hardware work to data analysis, embedded products are rapidly peaking interest due to its reliability and time bound perfection. First developed by MIT lab and called “Apollo Guidance Computer (AGC)” designed with 256 words of RAM and 4K words of ROM., having a microchip with clock frequency of 1.024 MHz, and a computing unit of 11 instructions and 16-bit word logic. Today, Embedded systems have evolved and can be found alarm systems, automated teller machines, transmission control, cruise control, fuel injection, anti-lock brakes, active suspension , microwave ovens, answering machines, home security systems, washing machines, lighting systems, fax machines, cell phones, digital cameras, camcorders, portable video games, calculators, and personal digital assistants, copiers, printers and scanners, cash registers, and uncountable more .In simplicity ,all computing systems asides general-purpose computers are embedded systems.

A computer system usually contains valuable information and is a vital part of our proposed system, particularly if the system is a file server in a network, such that it offers data exchange between itself and our embedded systems.

The connection of embedded systems to computer systems through “Interfacing”, gives circuit and hardware functionality a whole new power, Sutherland (1965), p.506 offers a vivid description of the power of interfacing systems together ,it says “A display connected to a digital computer gives us a chance to gain familiarity with concepts not realizable in the physical world ,it is a looking glass into a mathematical wonderland”, most embedded systems are made up of sensory devices which pick up real life information in measurable manners ,while some after picking up said signals are required to give alerts(e.g. basic smoke detector) others are built to pick such signals process it and send it or wait for a confirmation/input from another system(PC or another embedded system) to perform some other function(e.g. access control systems, scanners, biometric readers etc. ),Interfacing gives us the ability to read natures unseen occurrence(humidity, temperature etc.) an also electrical circuitry functions in digital format and also the power to control this occurrences or improve on the functions .In the proceeding sections of this chapter, We look at work done by past researchers and engineers, tools utilized and interfacing methods implemented along with various types of interfacing protocols .

**2.2 REVIEW OF RELATED WORKS**

The review uses the systematic literature review (SLR) method for embedded systems interface project, Ngadiman et al. (2015). in their work “A systematic literature review on attractiveness and learnability factors in web applications” talks about the attractiveness and learnability factors in Web applications. Proposed work, strengths, weaknesses of the related work that are limited to the attractiveness and learnability factors in UI are some of the factors considered in using the SLR. Besides, The SLR method is utilized by Islam. (2013).in his work “A systematic literature review of semiotics perception in interfaces”. Where he used it to identify working operation, tools and challenges of the semiotics perception in user interfaces. Thus, the SLR in this project aims to gather information about existing works on user interface and in relation to Web applications.

Works stated below have been reviewed using the SLR system, detailed above:

A very early implementation of embedded system computer interfacing can be read in **Alan E. Brown and David S. Lin** (1996), monitoring and control of power supply functions using a microcontroller. They introduce an embedded system dedicated to the overseeing or monitoring a power supply machine, the embedded system is made up of a power supply monitoring and control circuit using a microcontroller to remotely monitor and control the functions and conditions of a power supply machine. The embedded system is coupled to the primary and secondary sides of a power supply circuit to monitor important voltage and current signals of the power supply, such as the output voltages and currents, and to control the various parameters of the power supply such as the output voltage and current limits. Analog to digital interface circuitry is provided to convert the power supply voltage and current signals to digital signals which are retrieved by a microcontroller which converts the digital signals to numbers representing the values of the power supply signals, and then stores the numbers. The microcontroller is also interfaced to reference and feedback signals of the power supply to control the power supply's operation. The microcontroller further keeps track of the total elapsed time of operation and the total number of times the power supply has been powered up. The power supply monitoring and control circuit operates as a slave to a host computer system via any of the options in computers input output bus so that a system operator can retrieve all of the monitored information and can control the operation of power supply. The host computer system communicates with the power supply monitoring and control circuit through a serial link, so that the host computer can be remotely located. The present invention also provides self-calibration to assure accurate

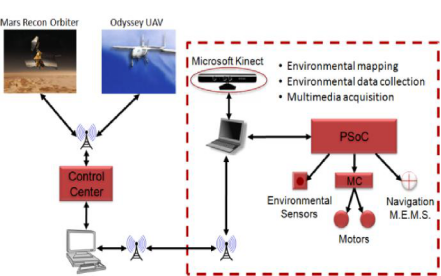
**Gendlal M. Vaidya and D. V. Padole, Ph.D.(**2013), Monitor Interfacing through Soft Core Processor, utilizes an ALTERA’s SOPC (system on programmable chip) development tool with a NIOS II processor to create a soft core processor, the system is interfaced to a computer via the ALTERA’s DE1 board which has provision for an RS232 serial port . The system also offers Field Programmable Gate Array (FPGA) which is loaded first with the Nios II soft core processor and then Real Time Operating System (RTOS) installed on this FPGA based architecture contained with the soft-core processor.

Also in 2013 **Anupa.K and Channabasappa Baligar** , Real-Time Communication between Aero Gas Turbine Engine Controller and Pilot Online Monitoring System , in their development of a pilot online monitoring system thy point out that, In embedded system it’s paramount to observe that the behavior of the application is the same on as the PCs development environment as well as deployed environment (target). The aero gas turbine system is found in defense aircrafts it, it utilizes and embedded device to monitor and gather periodic data on engine parameters of the aircraft ,it consists of actuators, sensor and DECU(Digital electronic control unit),The EECU accts like a black box collecting data every 30milliseconds from the aero gas turbine engine. The online monitoring system functionality is to monitor, store and analyze engine parameters. This is accomplished via serial communication, in the absence of the online monitoring system, the simulator this thesis talks about could be utilized, Aero engine comprises of ADC, DAC, MIL-STD-1553B interface, timer, Pentium III processor, VME backplane, power cards etc which is used in embedded controller display. There is a requisite for real time communication between aero gas turbine engine controller and pilot online monitoring system. This is accomplished by using RS232 serial communication. The application software which is designed and developed will be real time and it takes care of serial communication protocol.

**Paul Ashaolu**. (2012), Development of an Interactive 3-D Virtual Environment Test Bed for Motorbikes, interfaces an series of embedded system to a PC which renders 3-D virtual scenes via USB or R232, the thesis speaks about a project in three parts, each with their independent task the sensor module, the embedded systems and the visualization module. This architecture allows each modules of the system to be developed, tested, and individually improved on or diagnosed thereby making the development faster. It utilizes embedded system sensors (Lambda sensor measures the oxygen and lambda values of the motorbike’s engine) placed at strategic points on the live bike to detect the dynamics and other needed quantities of the live motorbike and all connected to an Arduino micro controller via an interfaced card ,The Arduino collects signals fed to it, does the necessary calculations, prints the sensor values as Serial Port variables i.e. streams of string arrays and parses it to our PC which host the capability to parse these streams as floats and sets the speed and rotation of the motorbike object and also shows it on the speedometer in the visualization software.

In a more advanced field, the concept of interfacing is utilized by **Garth Herman**. Et al. (2014), Implementation of Kalman Filter on PSoC-5 Microcontroller for Mobile Robot Localization. Computer plays an important role in the system ,acts as an initiator of the process and also as an intermediary ,All forms of interfacing in the proposed system is done via WAN network and serial communication depending on host pcs location, the work introduces a mobile robot with an embedded hard ware which receives global task from a host computing station(HCS),the HCS receives Ariel imagery from a satellite ,An operator uses this imagery to guide the robot path ,so as to avoid obstacles and provide destination.

The robot can be controlled manually of automatically, the robots embedded hardware is made up of 3 layers: The Algorithm Layer, Platform Layer, and Driver Layer. The Algorithm Layer is responsible for robot’s navigation, it computes navigational search algorithms, obstacle detection, and obstacle avoidance. The Platform Layer is responsible for the steering, sensor fusion, image processing, and kinematics of the mobile robot. The Driver Layer contains the sensor interfaces used for Robotic unit. Sonar and Infrared sensors interface with the PSoC-5. The Inter-Integrated Circuit (I2C), PWM (pulse-width modulation) control, ADC (analog-to-digital converter) and UART (universal asynchronous receiver/transmitter) standards are used for the above sensor interfaces. An accelerometer indicates the direction of gravity when at rest. A gyroscope indicates radial velocity. A magnetometer measures the strength of surrounding magnetic fields, a Kalman Filter is implemented on a PSoC-5 microcontroller to fuse data from the IMU sensors. This reduces standard deviation between measurements and improves reported heading accuracy, data collected is then communicated to the host PC via WAN integrated on the PSoC-5 or serial communication.



**Fig 2.1 Garth Herman System control and communication.**

*Source: the internet*

One very common system that has continuously improved on its means of embedded system, compute interfacing if the academic attendance management systems. **Kamal** (2015) in a project titled development of academic attendance management system using Bluetooth technology, designed and developed a system that manages attendance of students. The system consists of an embedded finger print scanning system made up of Arduino UNO, Adafruit Fingerprint Sensor, interfacing the embedded attendance system to the laptop computer is done using HC-05 Bluetooth Module (Master/Slave). The aforementioned components are connected together to obtain attendance from the student which is transmitted through the Bluetooth to the laptop for collation. The drawbacks of this system are that the operation -time for the hand-held device is not stated, the attendance size was not stated and the Bluetooth device has a communication distance of 10 meters thus exceeding 10 meters there will be no communication. Also, **Somasundaram et.al** (2016) in a project titled Mobile based Attendance Management System, deployed an attendance management system where students take attendance by first login on to a webpage to register and then after registration attendance taking process begins. The student on the other hand uses a mobile phone with an android mobile application which enable communication between the lecturer and students via sms. A significant drawback is if the server is down attendance for the day using the system is not realizable, **Neha et.al (**2013) in a project titled an efficient automatic attendance system using fingerprint reconstruction technique widened the scope of interfacing within an environment using a LAN connection, he developed an attendance management system using finger print recognition system. The attendance system consists of Fingerprint Scanner, LCD/Display Module, Computer LAN connection. This attendance system consists of 100 fingerprint scanners, 100 desktop computers and a LAN infrastructure. Fingerprint scanner will be used to input fingerprint of teachers/students into the computer software.

**Dimitri Diakopoulos and Ajay Kapur**. (2011), HIDUINO: A firmware for building driverless USB-MIDI devices using the Arduino microcontroller, their work talks about overcoming and creating solution to a general problem in existing microcontroller platforms: protocols for communication. Arduino, and other similar platforms like Wiring2 and Gainer3, implement Virtual COM ports, interfacing via USB for basic serial I/O between microcontroller and host, They needed a better way to connect musical instruments to an embedded system which reads musical analog signal and coverts it to digital signal, without the need for a new Arduino sketch and accompanying ‘decoder’ software which would interpret the raw serial data and convert to music-friendly MIDI() or OSC. and even with the conversion to MIDI it required the use of virtual MIDI loopback drivers (or proprietary loopback software, in the case of Windows)., the solution as the birth of HIDUINO, The HIDUINO is a firmware programmed to the Arduino which receives MIDI messages over a USART interface, repackages it as USB-MIDI or USB-HID and send to the host system via USB.

Interfacing is utilized in **Song Suns et.al** project Experiments in Attacking FPGA-Based Embedded Systems using Differential Power Analysis, for control and information monitoring, they present an automated data acquisition design for an FPGA based implementation of Data Encryption Standard (DES), and discuss obstacles they are encountered when performing Differential Power Analysis (DPA) attack on a commercial FPGA development board, Their goal was to show the security community the challenges inherent in performing real-world DPA attacks on both software (e.g. smartcard , DSP) and hardware (e.g. ASIC , FPGA) processing elements.in doing this they interfaced

The goal of their experiment is to automate the data collection and analysis stages of the DPA attack targeting a commercial FPGA board. Their setup consists of essentially three parts:

1. the digital oscilloscope which collects differential power traces from the FGPA board,
2. The FGPA board which is connected the oscilloscope via probes,
3. The host pc ,this serves as a control center for the whole system having a MATLAB software and Xilinx ,it coordinates the activities of our board and oscilloscope ,the host pc is interfaced with the oscilloscope via USB ,and to the FPGA , board via USB(for FPGA configuration download) and via serial connection (for data communication between PC software and FGPA software).

**Dr. Rajendra R. Dube and Sushama S. Pawar**. (2013), Design and Implementation of Traffic Monitoring System Based on Embedded Web Technology. They combine embedded system and host computer via the internet to monitor and control traffic, using web technology as its core the embedded Web traffic monitoring system combines with traffic information acquisition, traffic surveillance, traffic control, information publication and other traffic control functions. All data gathered by these systems are collected, stored, managed, transmitted, analyzed, and displayed on a Host computer. The complete project was divided into two parts divided into two subsystems. First part supervises traffic signal to monitor the violation of red signal. In this, system the Raspberry pi board, camera and one ultrasonic sensor are implemented. As the traffic light turns red, ultrasonic sensor at signal unit will start detecting vehicle breaking traffic rule. If any vehicle breaks the red signal, it will be detected by ultrasonic sensor GH-311. At the same time a camera captures the photo of that vehicle and sends it to the monitoring centers host computer using web technology.

**Yuhua Hu. (**2015), Wirelessly Connected Sensor Acquisition System for Remote Nursing Applications. Implements two embedded system with two different micro controllers connected to a desktop for monitoring and notifications to aid Medical personnel, Medical personnel can monitor patients from a comfortable distance, reducing staff stress as they don’t have to keep checking on patient. And for cases where personals cannot be close to or available in person patient monitoring is an alternative ,He implemented a hospital monitory system using two different micro controllers a Raspberry pi micro controller and an Arduino micro controller, considering The Arduino microcontroller it has a temperature and humidity censor(DHT-11)(embedded system 1), After acquiring the analog data, the Arduino’s central processor can transfer the analog signal to digital signal, and then send to a Bluetooth shield. The Bluetooth module can send digital signal to computer. While the Raspberry pi system interfaces with our PC via Wi-Fi , the PC is set as a hot-spot which the Raspberry pi can be connected to it and an Ip address I assigned to the Raspberry pi ,The main purpose of the Raspberry Pi is to stream the video using the OV5674 5MP camera module ,video is captured and the signal is sent to the processor of the Raspberry Pi. And the Raspberry Pi can send the camera signal to the Wi-Fi module. The Wi-Fi module can encode the signal into the digital signal and send the digital to the Computer from which the ward, area or incubator is monitored.

**Eetu-Pekka Kouhia**. (development of an Arduino-based embedded system.). utilizes an Arduino to create a microcontroller-based embedded system for monitoring greenhouse environmental variables. The system over comes the limitation its predecessor by reducing the power consumption, the user can control the greenhouse environment through a website. The website interface offers a 24-hour line chart which hold monitoring data.

The system utilizes Arduino uno to monitor environmental variables, In addition to that environmental variables can also be changed through relays ,The web server monitoring system was designed with C programming language ,environmental variables read by the sensor is processed by the Arduino and displayed on the web interface , The web interface was accessed via a hard coded (practice of embedding data directly into the source code of a program or other executable object, as opposed to obtaining the data from external sources or generating it at run-time. Wikipedia) ip address, the user interface is accessed via a host pc which is connected to our embedded system via LAN. Realtime clock module time and date is displayed under the main heading. Time is updated every minute together with sensor data and line chart.

* LIMITATIONS OF Eetu-Pekka Kouhia work.

If webserver does not respond to client data request line chart is shown empty. Power fluctuations may affect SD card storage sequence, which may also lead to chart showing empty

The concept of interfacing the Arduino with a monitor system **or computer** system to show meaningful and readable data, is implemented in **Daing Noor Farhanah Mohamad Ishak. Et al**. (2017), Arduino Based Infant Monitoring System, where he proposes a system for monitoring infant in an incubator and records data and displays information recorded on a computer system. The data recorded by the system can be further referred by the neonatal intensive care unit (NICU) personnel for diagnostic or research purposes. He equips an incubator with humidity sensor SN-HMD-MOD for humidity sensory, it has a fast response time and low power consumption which is suitable for medical applications. When the sensor is placed inside the incubator, it will detect the moisture around it. The data is transferred to the Arduino and then processed to be sent to a PC for further analysis. An LCD displays humidity value for all to see, the system utilizes a. SN-PULSE pulse rate sensor which is a pulse sensor is attached to an infant to monitor heart pulse. It is a plug-and-play heart sensor that can detect the pulse per minute for premature babies continuously., these values are converted and sent to a PC which is under supervision or owned by a medical personal via Arduino, though he didn’t state exact methods utilized to interface the PC, he showed that a LAN or internet method could be utilized.

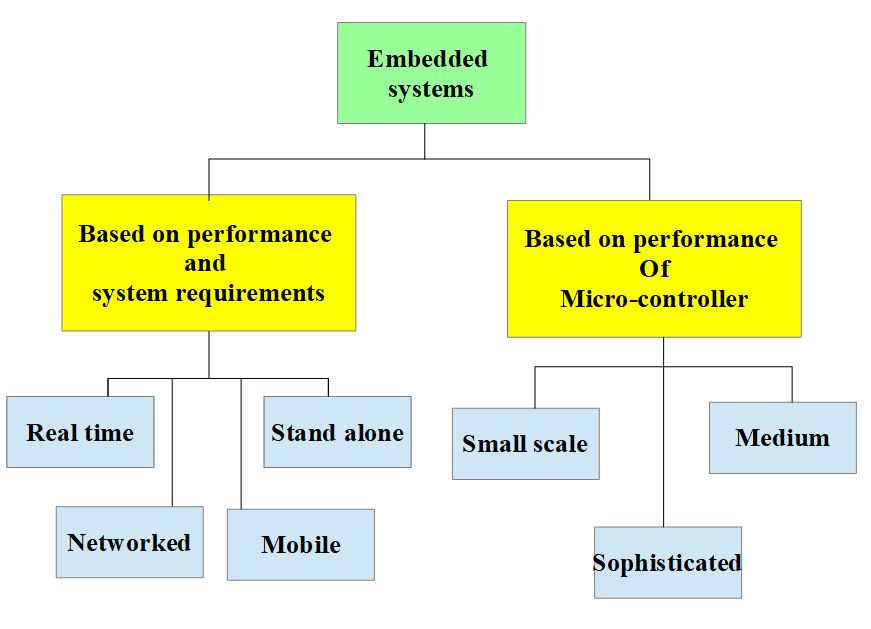
The areas of implementation are numerous and more fields are being discovered, the system is continually being utilized in its birth field engineering, it can be used in medicine, architecture, financial sector, aviation, agriculture and so many more

* 1. **REVIEW OF EMBEDDED SYSTEMS.**

Embedded system has become an integral part of our lives, as technology advances so does, the implementation of embedded systems, on their own they can function with little or no human interference, these systems can be found in equipment instrumentation and home appliances and is likely to continue in the recent future.

Embedded systems can be classified into two main categories:

* + - 1. Performance and Functional requirements
      2. Performance of the Microcontrollerthese categories have their sub-classes ,diagram below gives a detailed classification ;



**Fig.2.2 Types of embedded systems**

Explaining the systems in fig 2.2 above:

1. **REAL TIME:** These are systems which give required operations at specific times. they follow given deadlines to complete a task, they can also be classified as soft or hard real time systems.
2. **STAND ALONE**: These do not work with host systems(computer), they work by themselves, all I/O operations weather digital or analog occur within them, it processes and converts data and gives resulting output to the connected device to utilize e.g. mp3 players, digital cameras, video game consoles. Etc.
3. **NETWORKED EMBEDDED SYSTEMS:** These systems utilize LAN, WAN or the internet to access resources, and this utilization could be wired or wireless, this type of systems are the fastest rising types of embedded system applications.
4. **MOBILE EMBEDDED SYSTEMS**: As the name implies, these are portable embedded systems like our cell phones, their limitations are in the resources they need and memory they consume.
5. **MEDIUM SCALE EMBEDDED SYSTEMS:** just like the rest it has a software and hardware complexities, for the hardware it utilizes either a 16 or 32bit microcontroller, RISC (reduced instruction set computer) or DSPs (digital signal processing). the software can be developed using C, C++, java, visual C++, RTOS etc.
6. **SMALL SCALE EMBEDDED SYSTEMS:** This can be developed with a single 8- or 16-bit micro controller which may or may not utilize a battery.
7. **SOPHISTICATED EMBEDDED SYSTEMS:** Thesesystems have large software hardware complexities and use ASIPs, IPs, PLAs, scalable or configurable processors, utilized in advanced applications that need both software, hardware co-design and components which have to assemble in the final system.

Based on Performance and Functional requirements tis project functions I real time or close to real time, while based o on Performance of the Microcontroller it a medium scale system, utilizing the Arduino uno micro controller which uses a C or C++ programming language**.**

* 1. **REVIEW OF EMBEDDED SYSTEM COMMUNICATION PROTOCOLS.**
* **Protocol**: A set of rules and regulations is called a protocol.
* **Communication:**Exchange of information from one system to another system using a medium or interface.
* **Communication Protocol:**A set of rules and regulations that allow two electronic devices to connect to exchange the data with one and another.

This means of connections and data sharing can be called communication protocols, they are split into two distinct categories:

* + 1. **Wireless communication protocols**

Below are common wireless protocols:

**1. BLUETOOTH**

These are one of the most basic form of data exchange protocols over short distance, it utilizes short wavelength UHF radio waves in the ISM band from 2.4 to 2.485GHz, it was originally made as an alternative to RS-232 data cables, invented by telecom vender Ericsson in 1994.

It could connect to more devices than any other protocol available then(up to 7 devices),operating at frequencies between 2402 and 2480 MHz, or 2400 and 2483.5 MHz including guard bands 2 MHz wide at the bottom end and 3.5 MHz wide at the top, Bluetooth uses a radio technology called frequency-hopping spread spectrum, dividing transmitted data into packets and transmits each one on one off the 79 designated channels ,each channel having a 1MHz,with 800hops/s .

1. **WI-FI**

Utilized for wireless LAN, with a radio frequency of 802.11 based on IEEE 802.11 standards, it is used n devices like PC. smart phone. tablet, gaming consoles, printers E.T.C, these devices can connect use a WLAN network and wireless access point(hotspot) to connect to the internet. These access points have a great range outdoors but about 20 meters in walled or demarcated areas. For devices to send information to themselves wirelessly they must both be on the local network, Wi-Fi most commonly uses the 2.4 gigahertz (12 cm) UHF and 5 gigahertz (6 cm) SHF ISM radio bands. Having no physical connections, it is more vulnerable to attack than wired connections, such as Ethernet.

1. **ZIGBEE**

This is an of high level-level communication protocol, IEEE 802.5.4 based according to IEEE standard, utilized to establish personal area networks (PAN) with small, low-power digital radios, for home automation systems or medical device data collection, generally for mini projects in need of wireless connection. Created to offer simpler, less expensive and less power consuming wireless PAN, in contrast to WI-FI or Bluetooth.

Due to its low power consumption rate, it can transmit data through a distance of 10–100 meters line-of-sight, depending on power output and environmental characteristics. ZigBee devices can transmit data over long distances by passing data through a mesh network of intermediate devices to reach more distant ones, often used in low data rate applications that require long battery life and secure networking (ZigBee networks are secured by 128-bit symmetric encryption keys.) with a rate of 250 kbit/s, best suited for intermittent data transmissions from a sensor or input device.

* + 1. **Wired communication protocols**

Wired communication protocols could be intra or inter:

1. **Inter System Protocol:**

These are protocol that offer communication between two devices e.g., our embedded system and host PC. The communication is done through a inter bus system, they include:

* + - 1. **MODBUS**

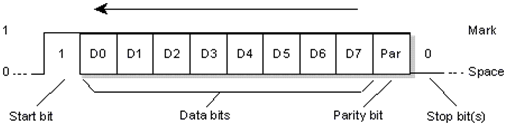
This interface Is built around messages, its format has no dependence of the type of physical interface used ,whether it is an RS232 or Modbus/TCP over ethernet same messages are used .The same protocol can be used irrespective of the connection ,this gives the Modbus interface a long life cycle, Due to independence on interface it makes it easy to upgrade system hardware of networks with large changes in software.

Devices can communicate with several Modbus nodes at once, independent of their interfaces. Serial Modbus connections utilizes two basic transmission modes, either ASCII or RTU (remote terminal unit). The transmission mode in serial communications defines the way the Modbus messages are coded. With Modbus/ASCII, the messages are in a readable ASCII format. The Modbus/RTU format uses binary coding which makes the message unreadable when monitoring, but at reduced size of each message giving allowance for more data exchange in the same time span. All nodes configured in either of the transmission’s modes must use the same mode for receiving messages A device configured to use Modbus/ASCII cannot understand messages in Modbus/RTU and vice versa.

* + - 1. **UART Protocol:**

UART stands for universal asynchronous transmitter and receiver .it is a two wired serial communication protocol, each wire also known as data cables are labelled as Rx and Tx. It is a half duplex (data reception and sending occurs but not at the same time) communication which transfers and receives data in serial bits without pulses. Most controllers have hardware UART on board but it uses a single data line for transmitting and receiving.

Ex: Emails, SMS, Walkie-talkie.

[](https://www.elprocus.com/wp-content/uploads/2013/09/communica.png)

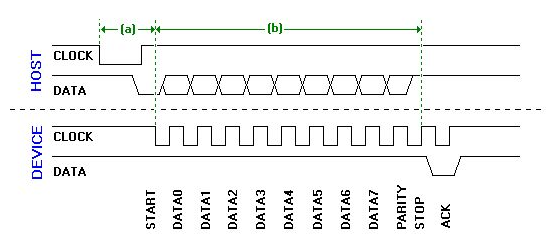
**Fig 2.3 UART Protocol Data Flow**

*Source: the internet*

**3. USART Protocol**

USART stands for universal synchronous and asynchronous transmitter and receiver,it is very similar to the UART in always except., It is a full-duplex protocol meaning it transmits and receives data simultaneously to different board rates. Different devices communicate with microcontroller to this protocol.

Ex:-Telecommunications.

[](https://www.elprocus.com/wp-content/uploads/2013/09/communication.png)

**Fig 2.4 USART Protocol Data Flow**

*Source: the internet*

1. **UNIVERSAL SERIAL BUS (USB)**

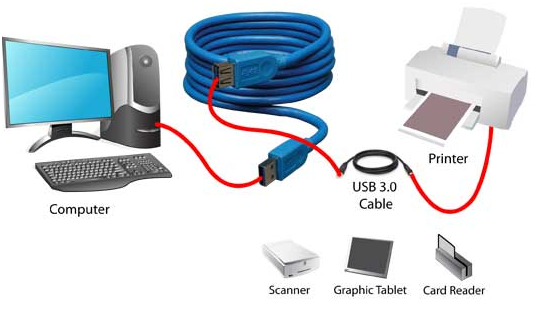
This one of the most common wired interfaces, it is easily applicable to provide data exchange between devices and a host controller such as a personal computer (PC). Due to its ease of access and multiple utilizations, the USB has replaced a wide range of interfaces like the parallel and serial port. Its plug and play feature allow spontaneous configuration of devices by operating systems, it also has a hot swapping feature which allows removal and replacement of new peripherals without having to reboot.

There have been three major USB standards, 3.1 being the newest:

• USB 3.1: Called Superspeed+, USB 3.1 compliant devices are able to transfer data at 10 Gbps (10,240 Mpbs),

• USB 3.0: Called SuperSpeed USB, USB 3.0 compliant hardware can reach a maximum transmission rate of 5 Gbps (5,120 Mbps),

• USB 2.0: Called High-Speed USB, USB 2.0 compliant devices can reach a maximum transmission rate of 480 Mbps, • USB 1.1: Called Full Speed USB, USB 1.1 devices can reach a maximum transmission rate of 12 Mbps.



**Fig 2.5 USB Protocol Communication**

*Source: the internet*

Table 2.1: Showing the Differences Between Inter System Protocols:

|  |  |  |  |
| --- | --- | --- | --- |
| Inter-system protocols  Characteristics | UART | USART | USB |
| * + - 1. **Meaning** | Universal Asynchronous receiver and transmitter | Universal Synchronous and asynchronous receiver and transmitter | Universal serial bus |
| * + - 1. **Wire protocol** | Has a two-wire protocol Rx and Tx | Has a two-wire protocol Rx and Tx | Has a two-wire protocol D+ and D- |
| * + - 1. **Transmission** | It transmits and receives data byte by byte without classes pulse | It transmits and receives data byte by byte with classes pulse | It sends and receives data along with clock pulse |
| * + - 1. **Form of communication** | It is a half-duplex communication | It is a full-duplex communication | It is a full-duplex communication |
| 1. **Speed** | It is the slowest | Faster than the UART but slower than the USB | fastest |

1. **Intra System Protocol:** used to transfer data between two devices on the same circuit board
   * + 1. **A CONTROLLER AREA NETWORK – CAN BUS**

The CAN bus offers micro-controllers and devices the power to communicate with each other without the need for a host, it is a message-based protocol, initially designed to minimize copper usage by means of multiplex electrical wiring within automobiles. It requires two wires CAN High (H+) and CAN low (H-).

CAN is a serial bus standard for connecting Electronic Control Units [ECUs] also known as nodes which can have multiple masters. the can network requires Two or more nodes to communicate. The node arrangement or hierarchy can be as simple a simple I/O device up or as complex as an embedded computer with a CAN interface and sophisticated software. The node may also be a gateway (PC) which could utilize a USB or ethernet to exchange data between a computers port to devices on the CAN network.

Each node requires a:

1. **Central processing unit, microprocessor, or host processor**

* the host processor decides what the received messages mean and what messages it wants to transmit.
* Sensors, actuators and control devices can be connected to the host processor.

1. **CAN controller; often an integral part of the microcontroller**

* Receiving: the CAN controller stores the received serial bits from the bus until an entire message is available, which can then be fetched by the host processor (usually by the CAN controller triggering an interrupt).
* Sending: the host processor sends the transmit message(s) to a CAN controller, which transmits the bits serially onto the bus when the bus is free.

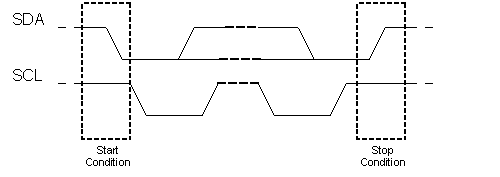
1. **Transceiver**

* Receiving: it converts the data stream from CAN bus levels to levels that the CAN controller uses. It usually has protective circuitry to protect the CAN controller.
* Transmitting: it converts the data stream from the CAN controller to CAN bus levels.

The CAN specifications use the terms "dominant" bits and "recessive" bits where dominant is a logical 0 (actively driven to a voltage by the transmitter) and recessive is a logical 1 (passively returned to a voltage by a resistor). The idle state is represented by the recessive level (Logical 1). If one node transmits a dominant bit and another node transmits a recessive bit then there is a collision and the dominant bit "wins". This means there is no delay to the higher-priority message, and the node transmitting the lower priority message automatically attempts to re-transmit six-bit clocks after the end of the dominant message. This makes CAN very suitable as a real time prioritized communications system.

* + - 1. **I2C Protocol**

I2C stands for inter integrated circuit. I2C it utilizes two wires connecting peripherals to carry information between devices to the micro-controller. the wires are called SDA (serial data line) and SCL (serial clock line). It is a master to slave communication protocol. Each slave with unique address. Master device sends the address of the target slave device along with a read/write flag. If address sent matches any slave device that device is ON, remaining slave devices are placed in disable mode. On the ON device communication commences between master and that slave device immediately address is confirmed and data can be transmitted or received. For every 8-bit data transmitted by the transmitter, the receiver replies 1-bit of acknowledgement. When the communication is completed master issues the stop condition.

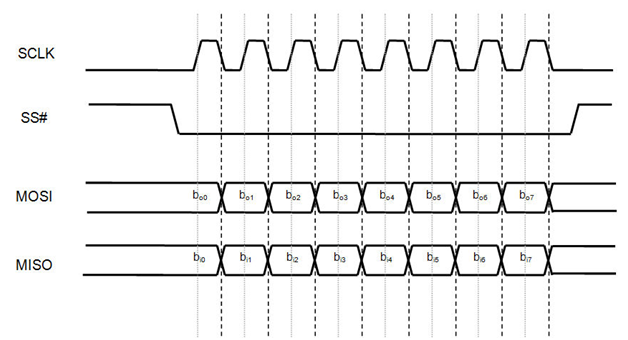


**Fig 2.6 I2C protocol data flow**

*Source: the internet*

* + - 1. **SPI Protocol**

SPI stands for serial peripheral interface also called a 4-wire protocol due to the fact that the system utilizes four wires MOSI, MISO, SS, and SCLK. It is one of the serial communication protocols developed by Motorola. SPI protocol also implements a master slave system.it is a full-duplex communication protocol and not limited to 8-bit in the case of bit transmission.



**Fig 2.7 SPI Protocol Data Flow**

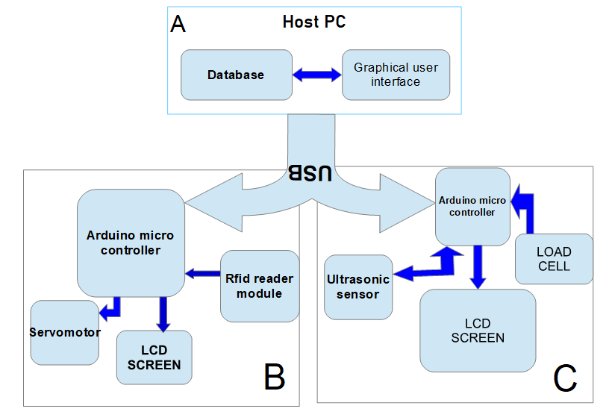
*Source: the internet*

From the above mentioned this project has been designed using one or more of the above-mentioned types of embedded systems and protocols, In the coming section we look at the general placement of the embedded systems to be implemented in this project and its communication protocol.

* 1. **DESIGN AND FEATURES.**

The project as mentioned earlier has a primary aim of interfacing two embedded systems, an RFID access control system and a body mass index (BMI) system to a host pc.

Interfacing would be done with embedded systems designed by my predecessors, the access Control system flow and casing was designed and implemented by Adekunle Abiodun Y. and Oladipupo Ismail O. they called it Programmable Access Control System, it utilizes a cantilever gate, which raises up to grant access when RFID card scanned has a positive match in their systems record. The system is designed for security and simplicity, to prevent unwanted access into an organization or to a place. It is automatic and card controlled, and offers design ﬂexibility making easy to integrate into this project, while the body mass index system utilizes an ultrasonic sensor, a load cell and a well-structured frame to collect student’s height and weight and save and display it on an LCD, the BMI system as designed by “Mr. Olawale”, combination of these two embedded systems can be visualized in the block diagram below



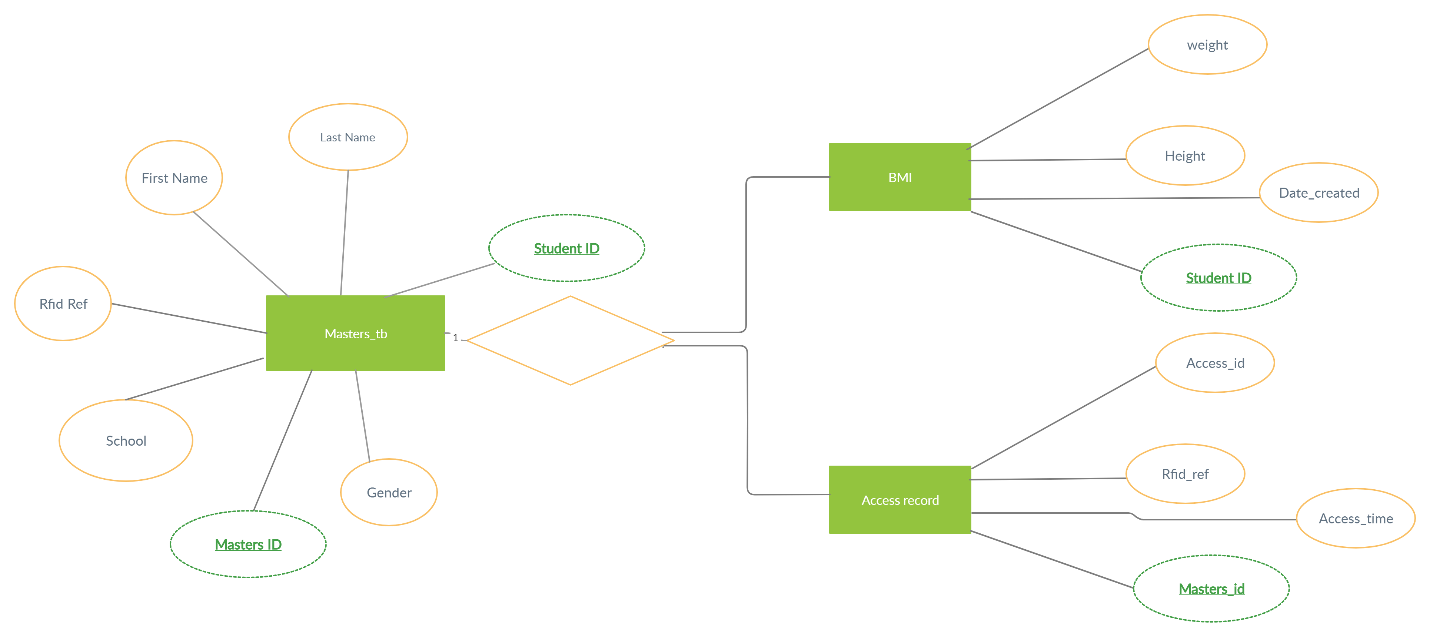
**Fig 2.8 Block Diagram of Proposed System**

The above block diagram in fig 2.8 shows the placement of the proposed system, the system encompasses three blocks (A, B, C), two of the blocks (B, C) are the embedded system block interfaced to the host pc (block A) through the use of a USB connector for serial communication.

Block B makes up the access control embedded system, with the Arduino system serving as an intermediary between the host PC and components of block B. The Rfid module reads students Rfid card and passes the cards details to the Arduino microcontroller, the Arduino sketch/script is written to only accept and read details ,initially it reads and serially transmits Rfid card details to the host pc using the digital pins 0 (RX) and 1 (TX) meant for serial communication on the Arduino  and inter system USB protocol, discussed earlier, then it also reads host pc response for decision ,the host PC compares card details to details already stored in the systems database ,if details match a response in form of a variable (1 or yes) is sent back to the embedded system which permits entrance else a response in the form of (Null,0 or no ) is sent keeping door closed ,also depending on response sent back to our Arduino the LCD gives or displays specific information .

Based on the same information exchange protocol of block B, Block C is used to collect student’s height and weight, Students height is collected using an appropriately placed ultrasonic sensor while weight will be collected using a load cell (), which utilizes a HX711 sensor, which digitally converts and amplifies Load cell output. Amplified value will be sent to the Arduino, Arduino calculates output of HX711, converts to weight and sends to our host pc database along with students’ weight. Block C differs from block B in the sense that, it performs no form of decisions making. its only meant to accept and deduce information.

Block A serves as the information hub(database) and user interaction space(G.U.I), Block A is the host pc, its holds database facility for storage of information collected by either of the embedded systems through the USB interface, The system administrator can control or manage information collected by either of the embedded systems, giving administrator ability to influence embedded system function from host PC G.U.I., administrator can view, add, delete and update students record, saved in the webserver database, below is an entity diagram of the proposed systems database:



**Fig 2.9 Entity Diagram of Systems Database**

The entity diagram shows the database schema for our proposed system. The system is meant to utilize a relational digital database, first proposed in (1970) by E. F. Codd, it uses a structure that allows us to identify and access data in relation to another piece of data in the database. Often, data in a relational database is organized into tables, it comes in various forms but for this work we use an SQL relational database system. The database schema has a masters table referred to as “masters\_tb” it holds all necessary student record and their respective rfid card index/number “rfid ref”, it relates or tags student or personnel to a specific rfid card. The masters table is connected to two child tables which uses a database feature known as “foreign key” to access and reference data in the masters’ table.

The BMI table holds record of students’ height and weight of each student along with date record was created, the “student\_id” entity is used as a reference key to the masters table and aid record generation while the Access record table holds record which serve as an access log, showing who and when a card was used, just like the BMI table the Access record table uses “Masters\_id” to as a reference key to access the masters table.

The system as a whole is meant to control and monitor students access (Block A and B) and also be able to collect students body mass index (Block A and C), through a networked embedded system which interfaces an Arduino based access control system and BMI data acquisition to a host pc via a USB wired communication protocol interface for the purpose of embedded system control, provision of a GUI and data storage utility. The coming chapter will offer explanations and specifications on system components both hardware and software.

**CHAPTER THREE**

**DESIGN AND IMPLEMENTATION**

This chapter covers the detailed explanation of methodology and project design that is used to make this project complete and working in great condition. As said earlier the system uses an Arduino micro controller which is interfaced to a computer system via a USB cord, the computer holds the control interface which offers a visible means of controlling the embedded systems function, the web interface was developed using PHP, HTML and PYTHON languages and the hardware is built using the micro controller as a connection point between a couple of components, from power supply units to final actions.

The project utilizes a descriptive research type which involves describing an existing or current phenomenon, this can be broken down in to three major steps data gathering/planning, implementing and analyzing.

On a broader view the steps above can be listed as

* Data collection
* Hardware and Software requirements
* Software code design and implementation
* Hardware prototype
* Analyze performance
* Identify the conclusion
  1. **DATA COLLECTION**

This involves all the steps taken to gather knowledge towards meeting the requirements of to meeting the project needs, this involves research towards familiarizing myself with the required code in this case python and c++ (Arduino c) documentation, a secure server to hold data into and how best to create our embedded system, along with the fittings of each component using the Arduino micro controller, all of these were done using the web(google and YouTube) to bring myself up to speed on these tools.

* 1. **HARDWARE AND SOFTWARE REQUIREMENT**

Using the schematics diagram above to represent our connections, the systems components are explained along with how they function

* **Arduino Uno**

The Arduino Uno as shown in fig. 3.1 is an open source microcontroller board based on the Microchip Atmegs328p microcontroller and developed by arduino.cc., it is a prototyping platform based on flexible, easy-to-use hardware and software, The board is equipped with sets of digital and analog input/output pins that may be interfaced to various expansion boards and other circuits used to receive inputs from the environment, the micro controller on the board is programmed to using the Arduino programming language. In the system this is used to receive inputs from the RFID reader which is compared to records in our database, in other for access to be granted, also used to as an means to collect and manage user height and weight collected by our BMI system, in both cases it serves as an interface between our environment readers and our computer system (An interface)

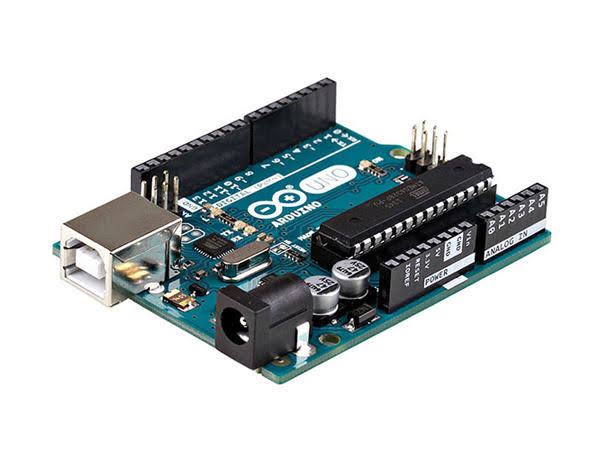


Fig. 3.1 Arduino Uno

* **RFID READER**

RFID reader works as a central place for the RFID system. It reads tags data through the RFID antennas at a certain frequency. Basically, the reader is an electronic apparatus which produce and accept a radio signals. The antennas contains an attached reader, the reader translates the tags’ radio signals through antenna which acts as a Transceiver depending on the tags capacity. The reader consist a build-in anti-collision schemes and a single reader can operate on multiple frequencies. As a result, these readers are expected to collect or write data onto tag and pass it to computer systems using USB cables to serial ports through an Arduino microcontroller, used in this project mainly as a means or data accessing and recording especially in the BMI setup which students statistics are saved to a particular tag which has been recognized by the system.



**Fig 3.2 RFID READER RC522**

* **Stepper motor driver**

A two phase motor driver is used to support speed control and direction of the stepper motor, it has 7 kind of micro steps (1,2/A, 2/B, 4,8,16,32) and 8 kinds of current control (0.5A,1A,1.5A,2A,2.5A,2.8A,3.0A,3.5A) In all which can be applied to our rotor or stepper motor. And all signal terminals adopt high-speed opt coupler isolation, enhancing its anti-high frequency interference daily, in our case we use a TB6600. Just like the image below, since we cannot use our Arduino to supply the voltage needed by the stepper motor and to avoid damage to the Arduino ,the motor driver comes in ,receiving input signal from the Arduino and in turn move the motor which has the barricade attached ,in whatever direction we please



Fig. 3.3: stepper motor driver

**Table 3. 0: Electrical Specifications**

|  |  |
| --- | --- |
| Input current | 0-5.0a |
| Output current | 0.5-4.0a |
| Power (max) | 160w |
| Micro step | 1.2/a, 2/b, 4,8, 16,32 |
| Temperature | -10-450C |
| Humidity | No condensation |
| Weight | 0.2kg |
| Dimension | 96x56x33mm |

**Wiring instructions**

There are three input signals in all. Step pulse signal PUL +, PUL. Direction signal DIR +, DIR. Off-line signal EN +, EN. The driver supports common-cathode and common anode circuit

* **The Stepper Motor**

**The Nema 23 stepper motor is utilized for implementation, it has** a 2.3 X 2.3 inch faceplate. With a max torque is 3Nm. This torques can be variated depending on the speed. It moves at 1.80 step angle (200 steps/revolution). Each phase draws 2.8A at 3.2V, allowing for a holding torque of 19kg-cm.The essence of using stepper motor Nema 23 is the required torque to carry the cantilever rod and to make easy flexibility and movement of the rod.



Fig. 3.4: Stepper Motor

* **LCD screen 20 by 4**

This is a controller circuit integrated device display terminal that could be connected to an Arduino, it is used to display the user process in and process out. It is powered via the Arduino +5v pin. it is meant to display user steps and serve as a means of notification in this project

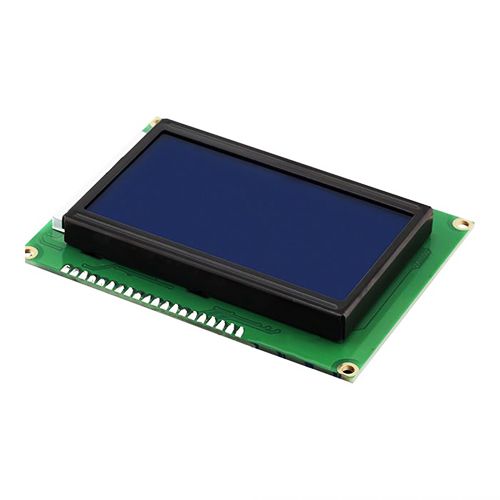


Fig. 3.5: LCD screen 20 by 4

The pin description of a 16\*2 LCD display is seen in the table below;

|  |  |  |  |
| --- | --- | --- | --- |
| Pin | Symbol | Input/output (I/O) | Description |
|  |  |  |  |
| 1 | VSS | / | Ground |
|  |  |  |  |
| 2 | VDD | / | +5V Power Supply |
|  |  |  |  |
| 3 | VEE | / | Power supply to control contrast |
|  |  |  |  |
|  |  |  | RS=0 To select command register, |
| 4 | RS | I | RS=1 To select data register |
|  |  |  |  |
|  |  |  | R/W=0 For write, |
| 5 | R/W | I | R/W=1 For read |
|  |  |  |  |
|  | E | I/O | Enable (Used by the LCD to latch information |
| 6 |  |  | present to the data bus) |
|  |  |  |  |
| 7 | DB0 | I/O | The 8-bits data bus |
|  |  |  |  |
| 8 | DB1 | I/O | The 8-bits data bus |
|  |  |  |  |
| 9 | DB2 | I/O | The 8-bits data bus |
|  |  |  |  |
| 10 | DB3 | I/O | The 8-bits data bus |
|  |  |  |  |
| 11 | DB4 | I/O | The 8-bits data bus |
|  |  |  |  |
| 12 | DB5 | I/O | The 8-bits data bus |
|  |  |  |  |
| 13 | DB6 | I/O | The 8-bits data bus |
|  |  |  |  |
| 14 | DB7 | I/O | The 8-bits data bus |
|  |  |  |  |

* **Aluminum rod**

This is the barrier rod, tighten to the motor shaft which serve as a security obstruction or door being access controlled from intruders as shown in figure 3.6.



Fig 3.6: Aluminum rod

* **Ultrasonic Sensor**

The ultrasonic sensor uses SONAR to determine the distance of an object just like the bats do.it is a non-contact range detector which sends out sound signals within 2cm to 400cm and uses the time taken for the sent out signals to bounce back or return.

It is used here to determine the height of an individual in the BMI placed at the top, the distance measured by the Ultrasonic sensor is subtracted from the measurable height or the BMI.

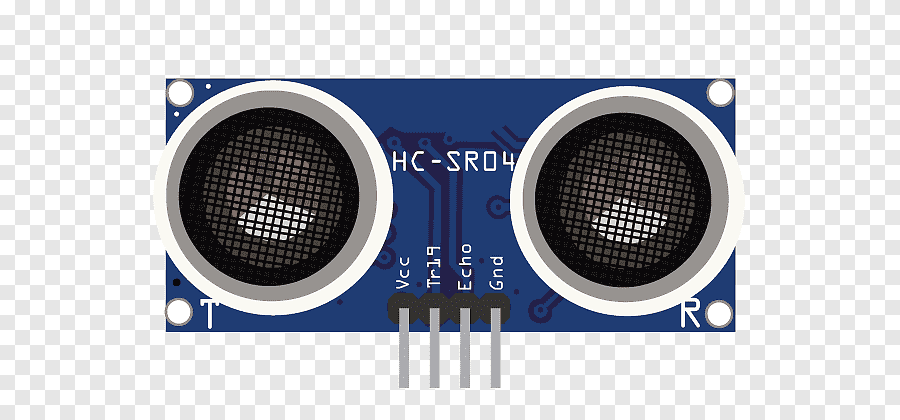


Fig 3.7: Ultrasonic sensor

* **Load Cell**

These come in a couple of shapes and sizes, it measure strain and relate it to weight of object on it. A load is applied to a strain gauge, which causes the gauge to strain a certain amount and output a voltage proportional to the applied load, it is simply meant to measure weight of humans standing on it and this weight is persisted to the database



Fig 3.8: Load cell

* **Centralized Computer Server (PC)**

In order for the system to become operational, a centralized computer server (CCS) is needed. The computer server will serve as the front-end of the system that provides an interface between the RFID hardware and application based system, which is the “main brain” of the system. It is used to network multiple RFID interrogators and Arduino Uno microcontrollers to centrally process input and output information using USB wires and Serial Ports. The data that flows to computer enters through computer serial ports. Serial ports are detected in the computer as ComPort or Communication Ports. Personal Computer (PC) consist of a running front end and database communicating system, for the project it hold compiler for the python program written to compare RFID cards scanned on the reader to RFID tags on the database, it could also display collection of data tags to the admin so it can be managed adequately (from deleting or deregistering tags to editing and adding new tags).

* **Database Management System**

DBMS for short is used to store, define, manipulate and retrieve data for a particular system, A DBMS generally manipulates the data itself, the data format, field names, record structure and file structure. It also defines rules to validate and manipulate this data. Database is a part of the centralized computer server or personal computer (PC) that handles the storage, retrieval, and updating of data in a computer system. The Database system contains an SQL server in which all the data will be stored. All the information loaded to the interface will come from the system’s Database server. This will basically hold the information needed for decision in accessing access controlled areas. Pronounced either "Microsoft S-Q-L" or "MS Sequel," It is an open source relational database management system. It is based on the structure query language (SQL), which is used for adding, removing, and modifying information in the database. The project utilizes a Xampp database, it is an open source server used to test projects on local servers before making it available to the web, and it also offers a database management capability to various languages making it the most viable for the project.



Fig 3.9: Xampp Database Interface

* **Integrated Development Environment(IDE)**

IDE’s are software’s for building applications that combines common developer tools into a single graphical user interface, for this project, intellij products PYCHARM and PHPSTORM are used, while PYCHARM is used for processing and compiling python codes PHPSTORM is used for php codes which is used to develop interfaces which do not need connection to our micro controller

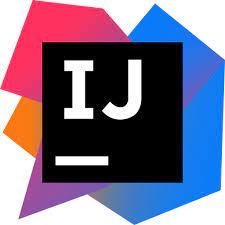


Fig 3.10: Intellij IDE

* 1. **SOFTWARE DESIGN AND IMPLEMENTATION**

The primary software is the code placed in our micro controller for RFID card reading and on correct card read, grant user access, along with code written for height measurement, weight measurement for the BMI system. The code as mentioned earlier is written with the Arduino cc programming language.

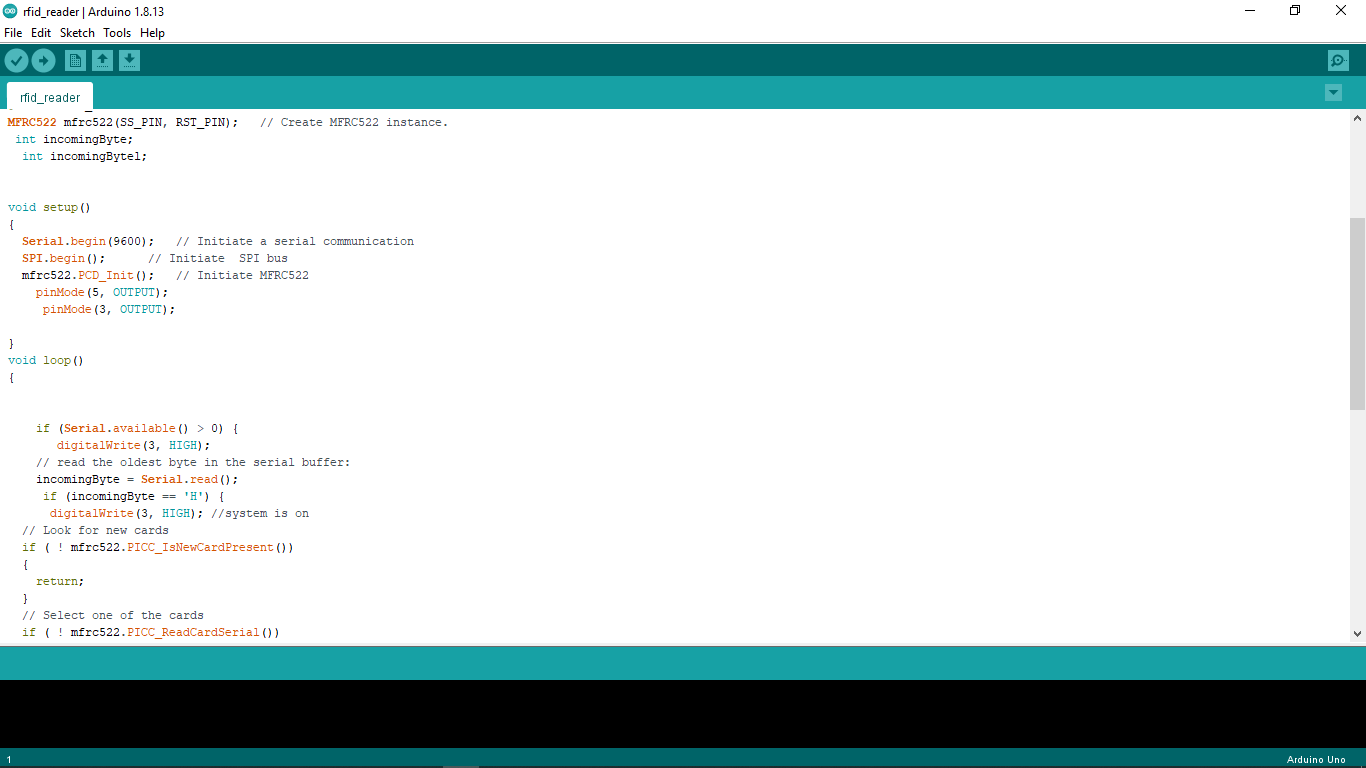


Fig 3.11: RFID CODE SNIPPET

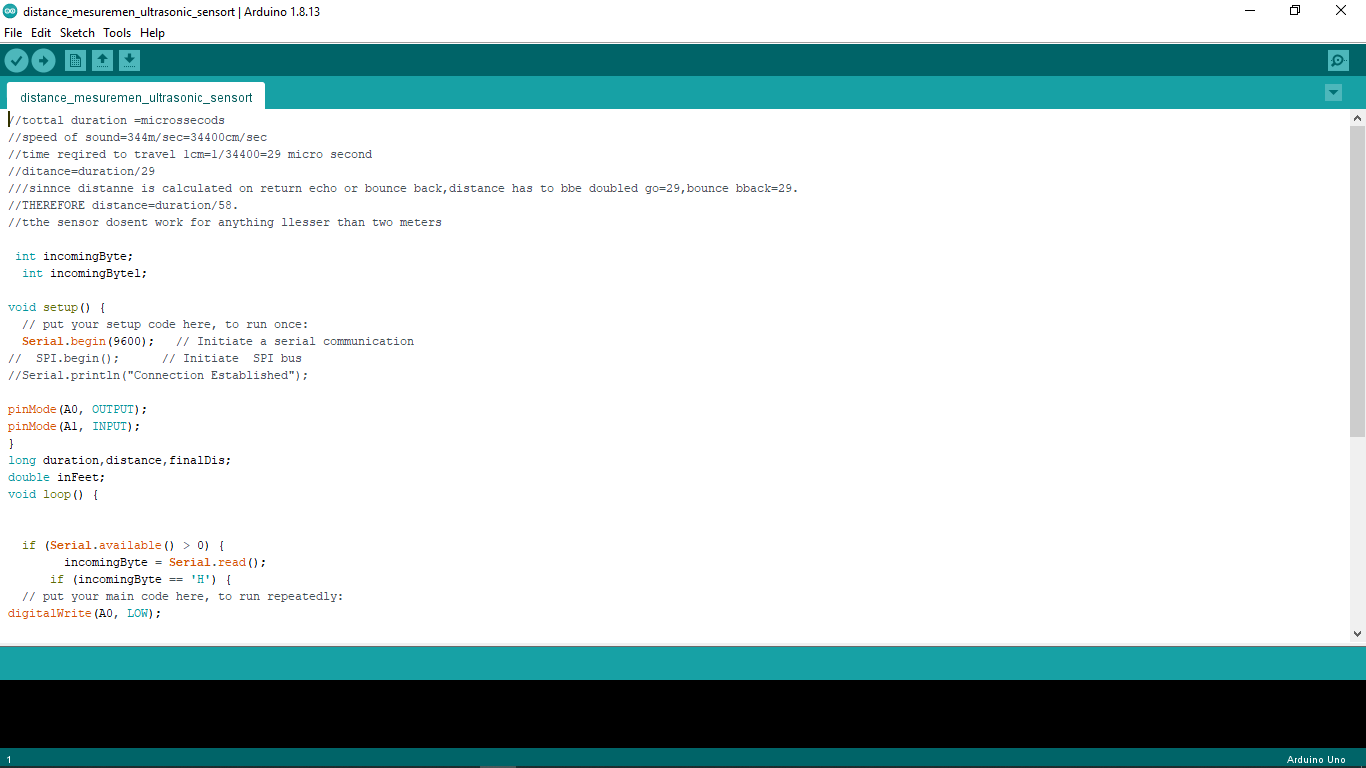


Fig 3.12: BMI CODE SNIPPET

Depending on the primary codes above and operations our admin needs to performs our user interface written in python, php, css and html displays data and information, the python code is able to send data back to the micro controller giving it data to further process and perform actions (e.g. if user data exist in database, a letter is printed to the micro controller which opens the lever)

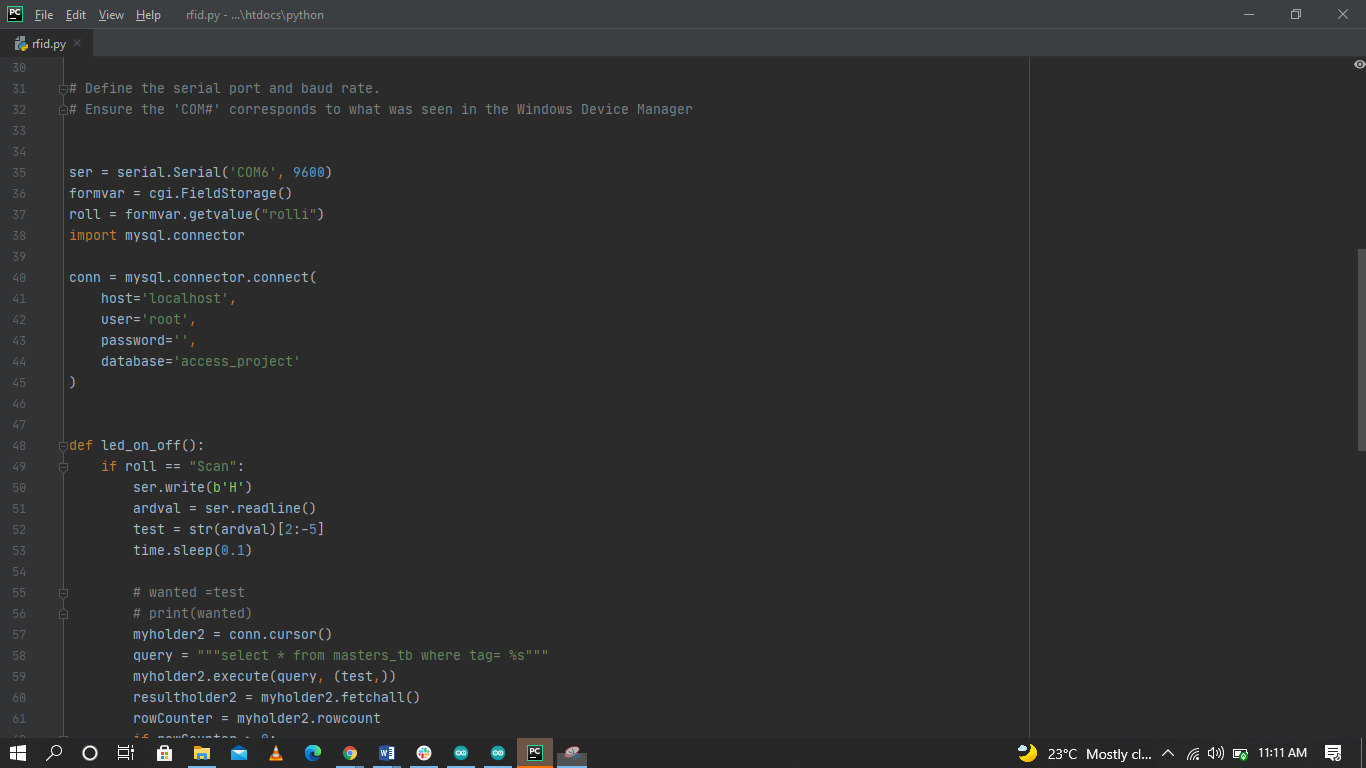


Fig 3.13: Pycharm python snippet for access control system

* **PYTHON.**

This is a high level programming language, emphasizing on code readability, used to build websites and software, automate tasks, and conduct data analysis, here it is used to communicate with the Arduino code and pass signals between the interface and the embedded system ,once a card is scanned it passes signals to the Arduino and back.

* **PHP.**

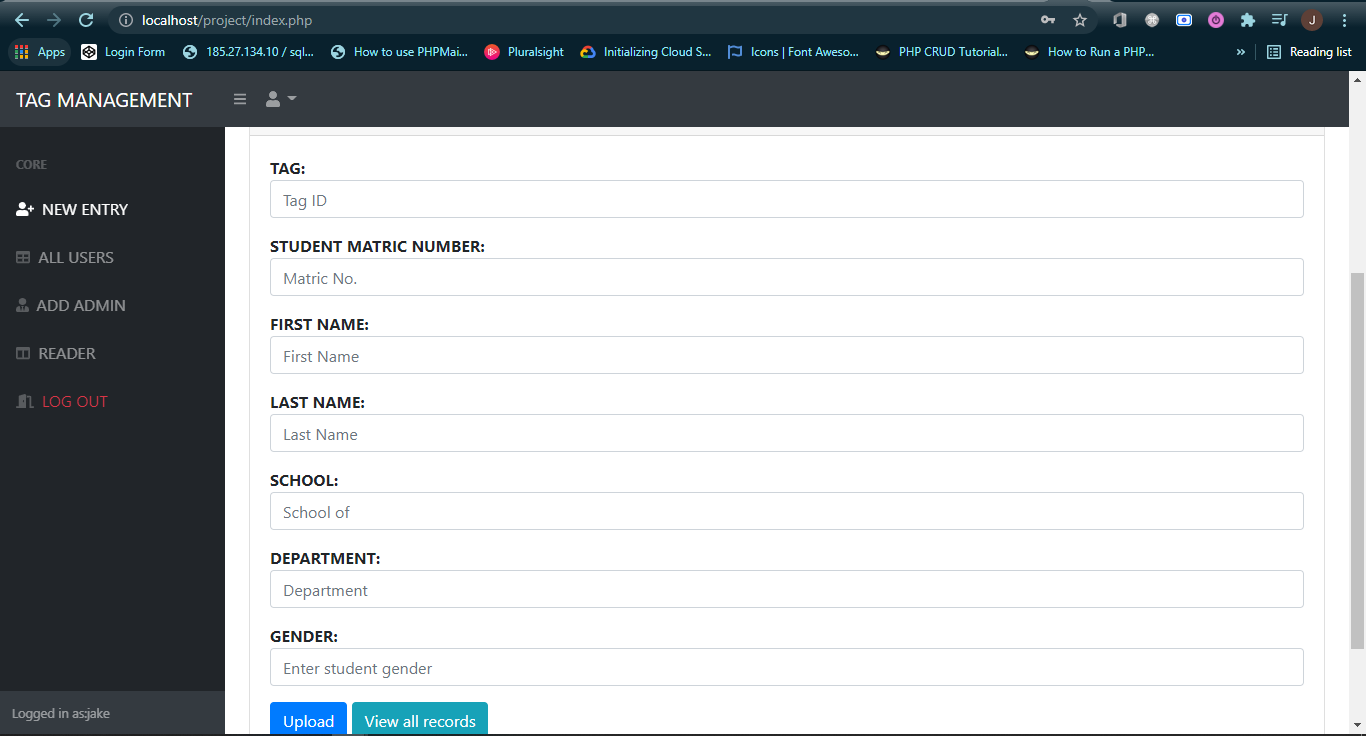
PHP is an acronym for "PHP: Hypertext Preprocessor" · PHP is a widely-used, open source scripting language, used to communicate with a database they are called server side script ,use to perform CRUD operations to the database, here it is used for the creation, editing, de-registering a tag.

* **HTML.**

HTML is the standard markup language for Web pages. With HTML you can create your own Website infusing a couple of libraries from cascading stylesheet (css) to JavaScript to bootstrap and we can create a user friendly page.

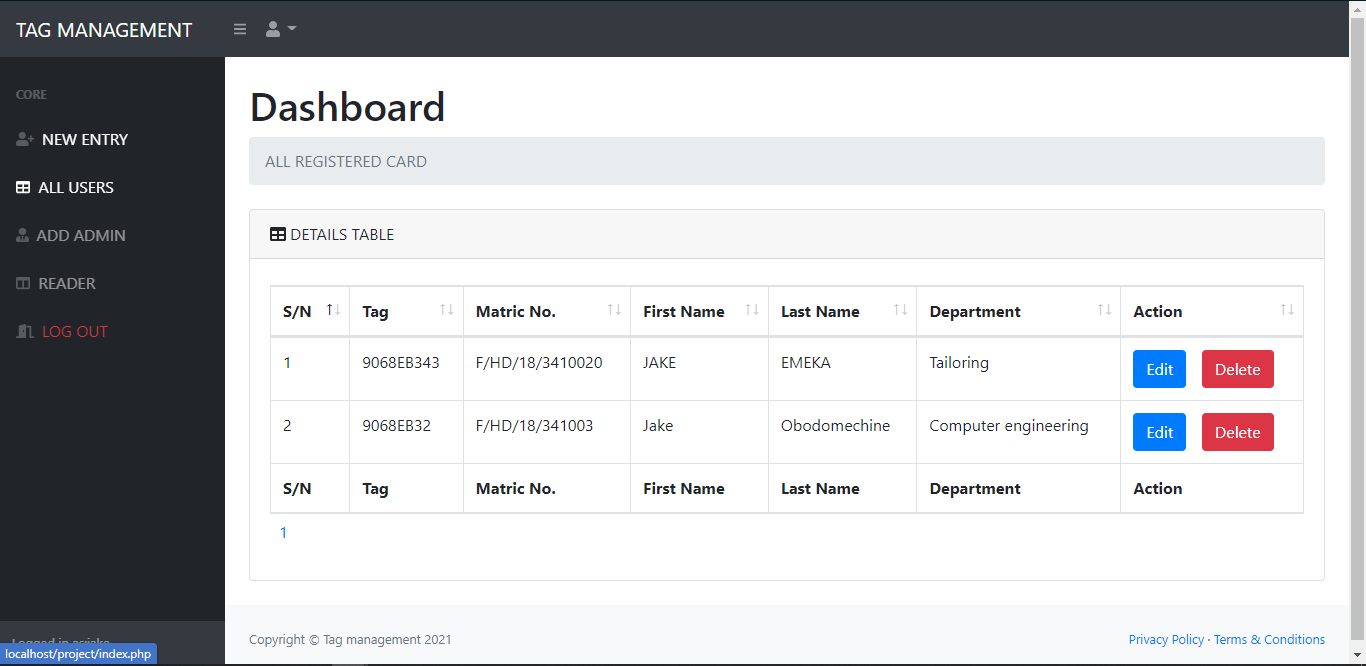
* **Interface results.**

The primary goal of the project is to make embedded systems easier to manipulate, access and updated without the need to enter the hardcode through an interface, using the tools previously stated above (Pycharm, PHPstorm, and Arduino CC), a user interface have been provided for the access control embedded system, screen shots below show a web view of various operations that can be used to support and aid embedded system operations.



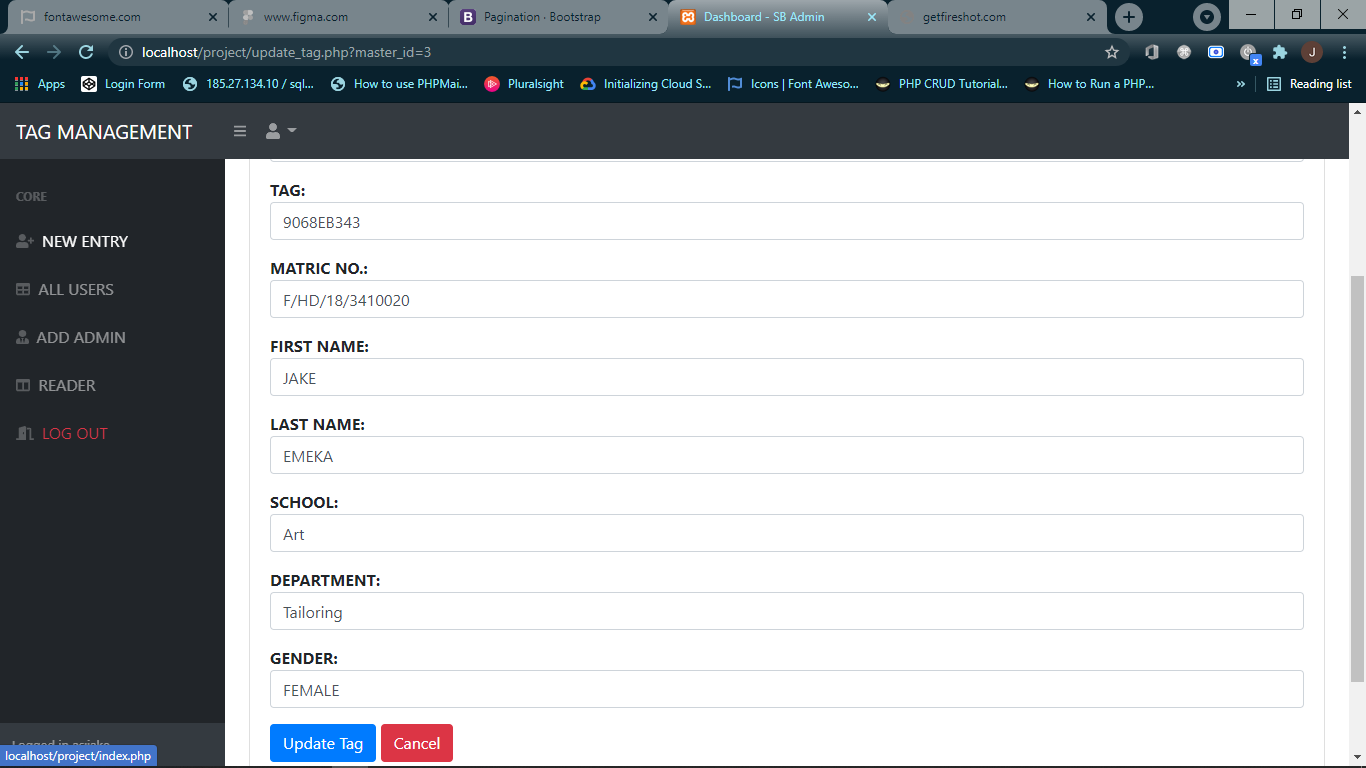
**Fig 3.14 Interface to register new Tags along with student details**

**Fig 3.14**  shows the interface where a new tag can be registered and their user details attached to the database ,once saved successfully ,user with these cards are able to access anywhere and have their records improved all that is needed is the identity on the RFID card.



**Fig 3.15 View All registered details and delete redundant tag details**

For tracking, security and detail purposes the 3.15 interface offers means to get fully detailed information of each user and how they use the access card along with information on updates done to the access card, also once a student graduates or card user is no longer active, card details can be taken of the database at a click of a button, without the need to go into the Arduino cc code to remove or alter details, once tag id can’t be found, no transaction can be initiated.

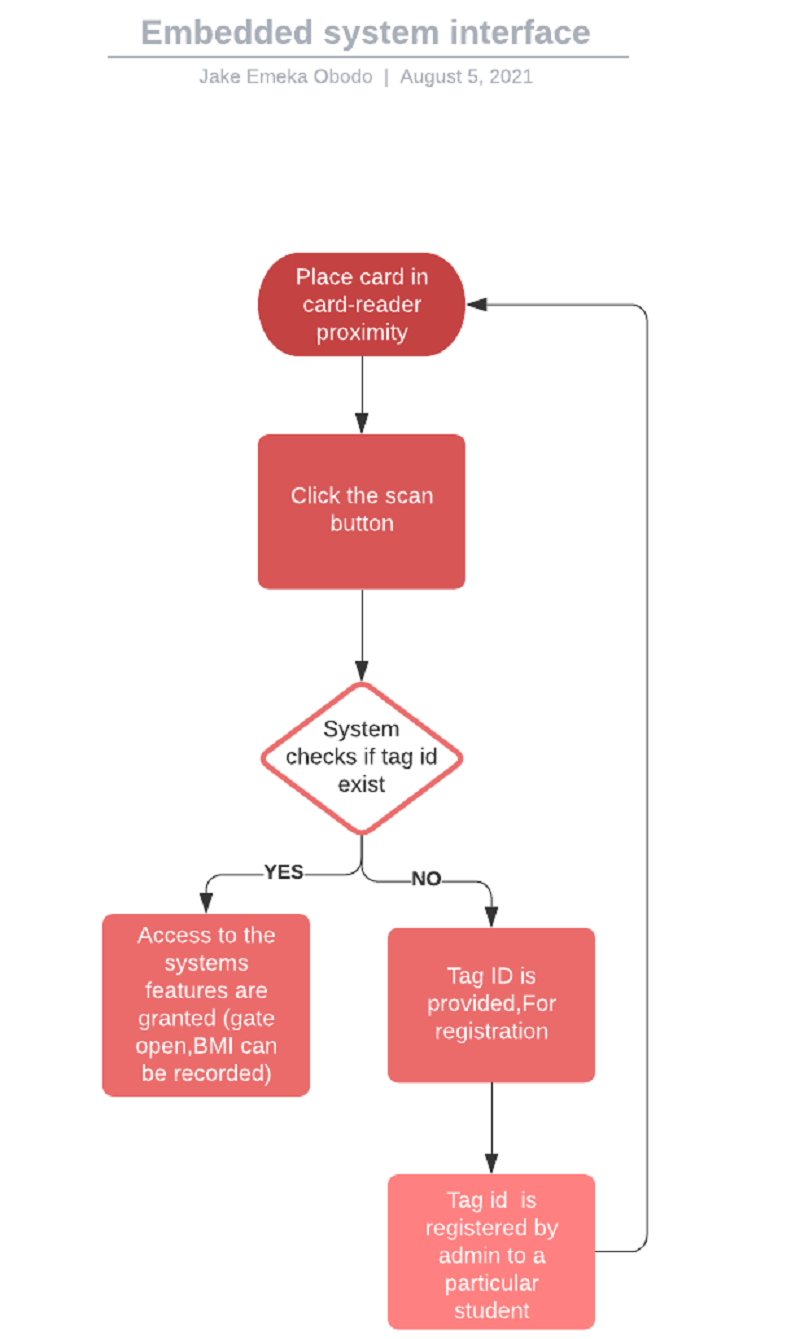


**Fig 3.16 Modify details attached to a tag or the tag**

For the purpose of human errors and inconvenience fig 3.16 offers an interface with a dual purpose ,first the ability to correct entry error in user details or card details and again removes the constant need to go into the hard code for corrections, second to get full details on a particular user.

* 1. **DATA PROCESSING FLOW CHART.**

Here we take a look in to the steps involved when using the system.



**Fig 3.17 Flow chart showing system operation steps**

* 1. **CONSTRUCTION PROCEDURE**

In building this project, the following procedures were properly considered,

1. Purposing of the entire materials / Components needed
2. Checked components rating of each component, to avoid damages and extra expenses
3. Used drafted out schematic diagram as reference to assembling components
4. Testing the completed system to see if the design works and finally, implementation of design of the project.
5. The next step is to mount the RFID component which is soldered to the appropriate place. It has 6 wires: two represents the antenna wires and the other four wires is for the Tx, Rx, 5v, and the gnd. The 5v wire is connected to the 5v pin on the Arduino and the GND wire is connected to the GND pin on the Arduino. The Tx pin is connected to the Arduino pin 3 and the Rx is connected to the Arduino pin 2.
6. Tested the ability of the card reader, to read a tag and give its information from the database or give only the tag number if the tag can’t be found in our database.
7. The next step is to connect the wires of the stepper motor to the stepper motor drivers. The stepper motor has 4 wires in which after testing for continuity two each must run continuity. The four wires represent the bipolar two phase of the motor: to move forward and to move backward.
8. After connecting the stepper motor wires, the pull and dir wires are connected on the stepper motor driver to the Arduino which symbolizes the signal for the movement and direction of the motor.
9. Implemented the prefer signal to be sent in other to trigger the movement of the stepper motor when a registered card is flashed across the RFID reader,
10. The next step is to connect the LCD screen pin to the pin on Arduino. It has 16pins, in the course of this project 12 pins is being used, which is connected to the appropriate pin on the board.
11. Then make the LCD display appropriate message when a card is flashed across the RFID.
12. The final step is soldering the wires on the PCB.
13. Updates any components left and do mounting at PCB.
    1. **SOLDERING PROCESS**

Some components were soldered due to their shaky components, like the legs of the LCD screen to the board and RFID reader to its legs and to the inter connecting wires. Then after that, the case, where the entire circuit was mounted follow by other external components such as, LCD, smart card slot/reader and indicating LED. Other components (Arduino, power pack) were attached to the steel box while the BMI had wires passes within it metal enclosure from the Arduino to the ultrasonic sensor and load cell to the Arduino.

* 1. **ASSEMBLING OF SECTIONS**

With the available casing, provision had to be made to interface the embedded system Arduino to the pc .The sections were properly laid out and assembled into the casing where the general coupling and linkages into the peripheral devices took place. The RFID reader is attached to the surface of the constructions so it could easily read smart cards

* 1. **PACKAGING**

This is a very important aspect of the design work. It is the appearance given to the final work. To give and presentable and safe work our embedded systems are enclosed

After soldering on the PCB, we do not leave the work like that; it has to be cased. Packaging could serve two major functions.

1. Serve to protect the components used for the design.
2. Serve to make the finished work look attractive.

An iron box is used in the packaging of the access control. The dimension and design of the box was arrived at after considering various factors such as the width and length of the components

The dimension for the casing is:

Length – 1ft i.e 12 inches

Height – 3ft i.e 36 inches

The BMI utilizes a horizontal rod connected to flat surfaces the one above holding then ultrasonic sensor for height and the other at the base with the load cell system.

* 1. **QUALITY ASSURANCE**

This is part of the work done on the embedded systems to check and cross check that all embedded system function run as smoothly as it did before it was enclosed and soldered down and track that there are no short circuits, cutoffs or breakages in connection, A multi meter is used for this, a failed continuity is checked and fixed with a solder or re-insert connection points, while on the interface it involves cross checking and refactoring our build up code, seeing if we could make it better more efficient and in a lesser amount of processing time.

**CHAPTER FOUR**

**TESTING, RESULT AND DISCUSSION**

With the realization and layout of all the components that make up our system, we put them together and test their inter relationships showing the outcome of what these components put together is what this chapter aims to present.

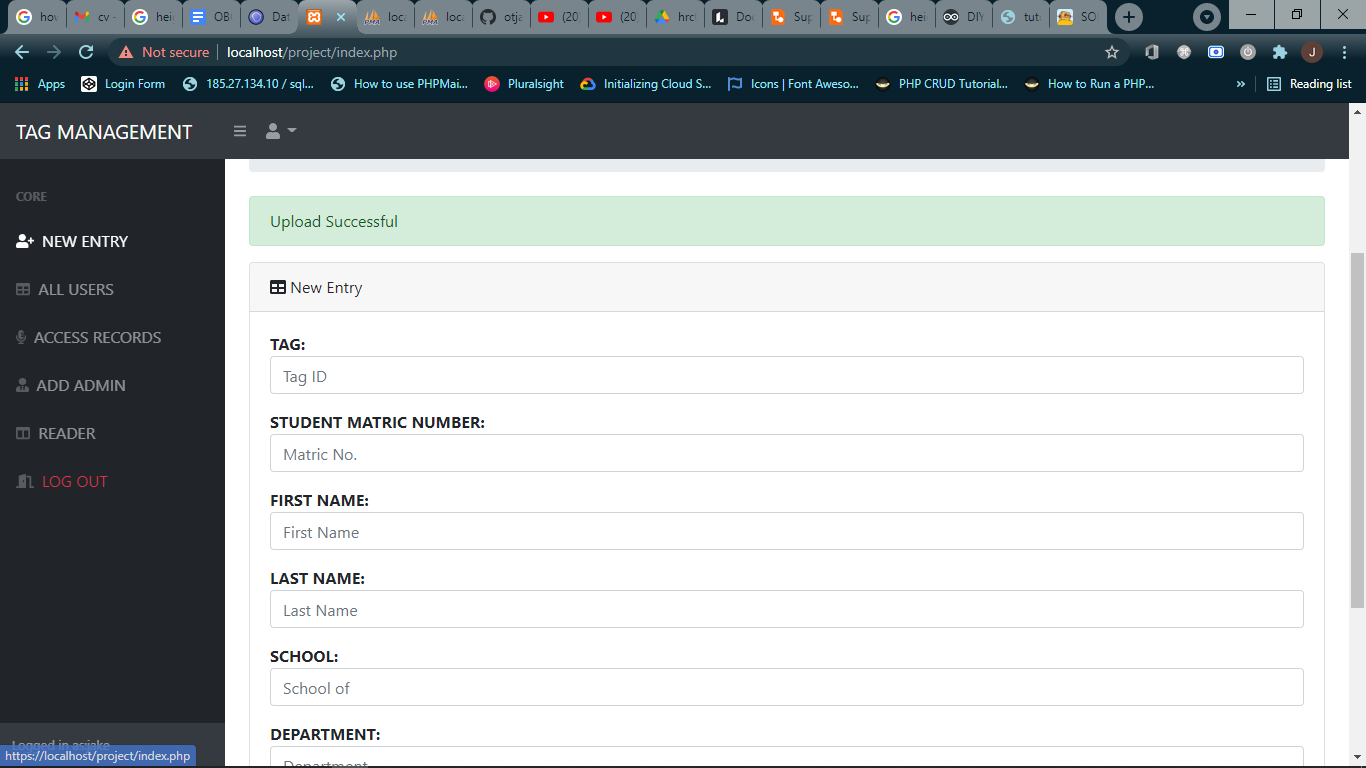
**4.1 REALIZATION OF THE SYSTEM**

The figures below show the realization test of the hardware part of the system

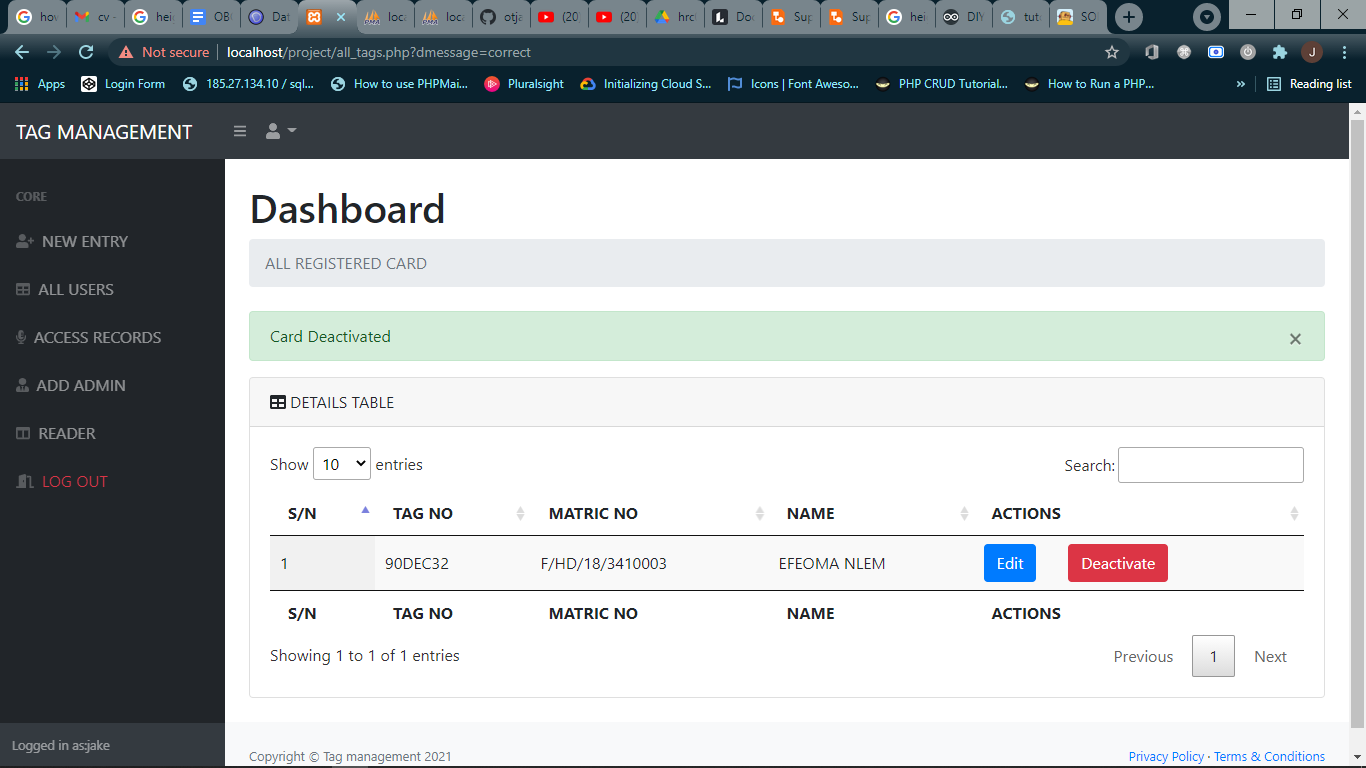
\*\*hardware pictures

**4.2. RESULT**

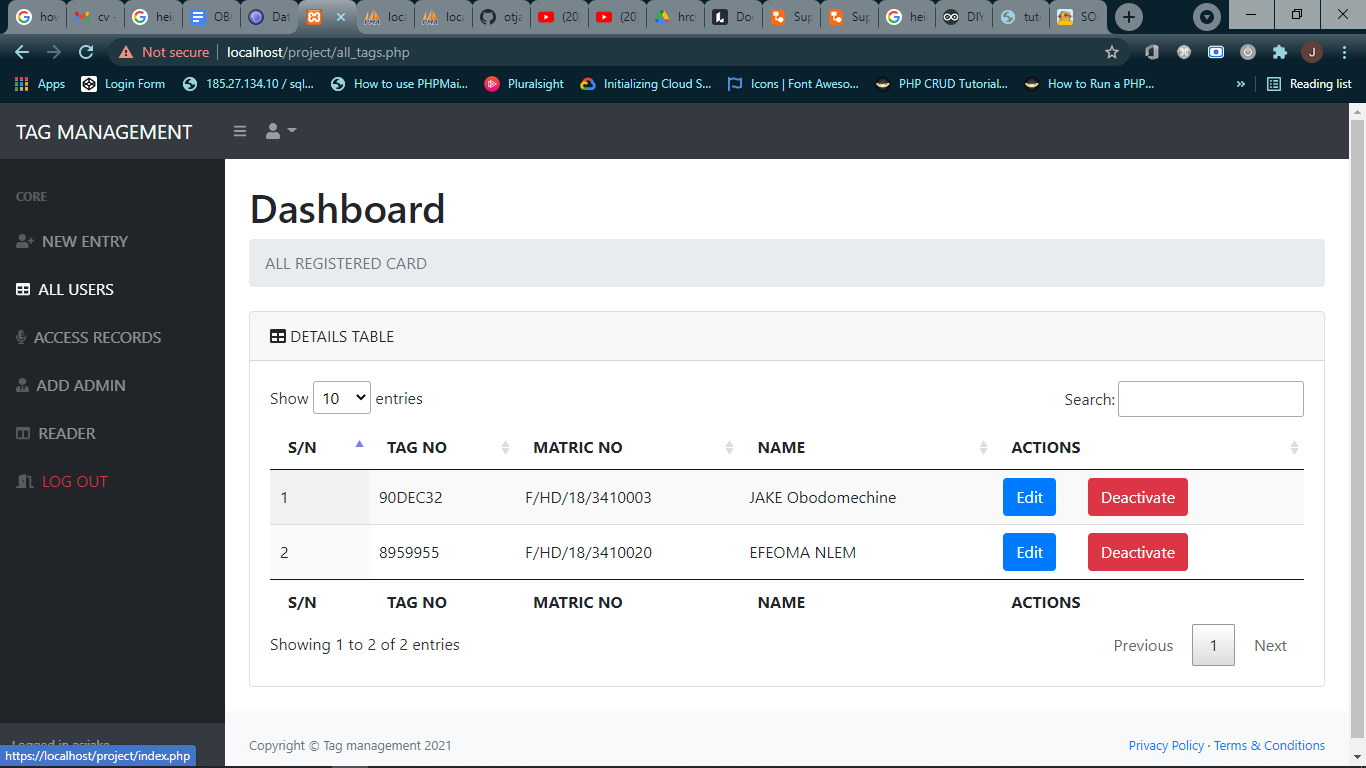
The couple of screen shots below show the results of the interface, it’s functioning and inter relation to the embedded system



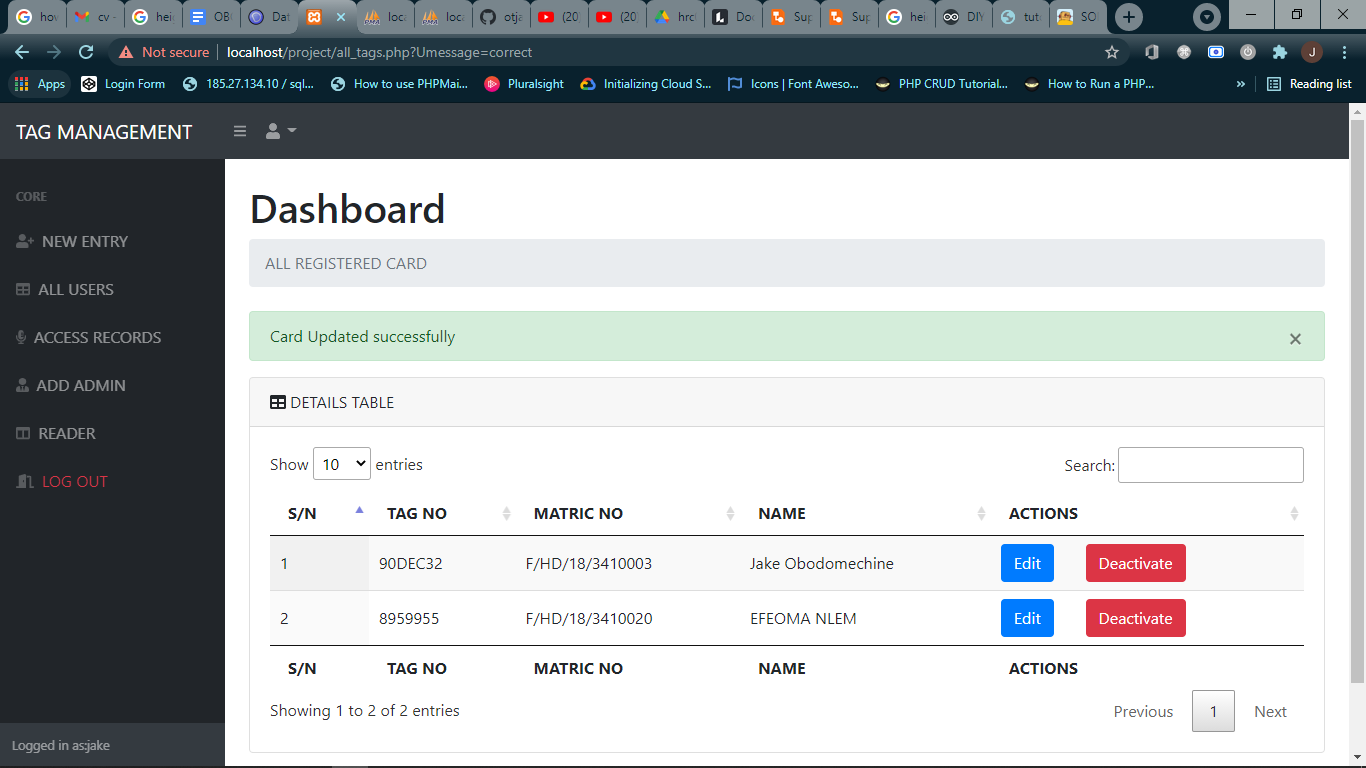
**Fig 4.1 Successful addition of a new tag**



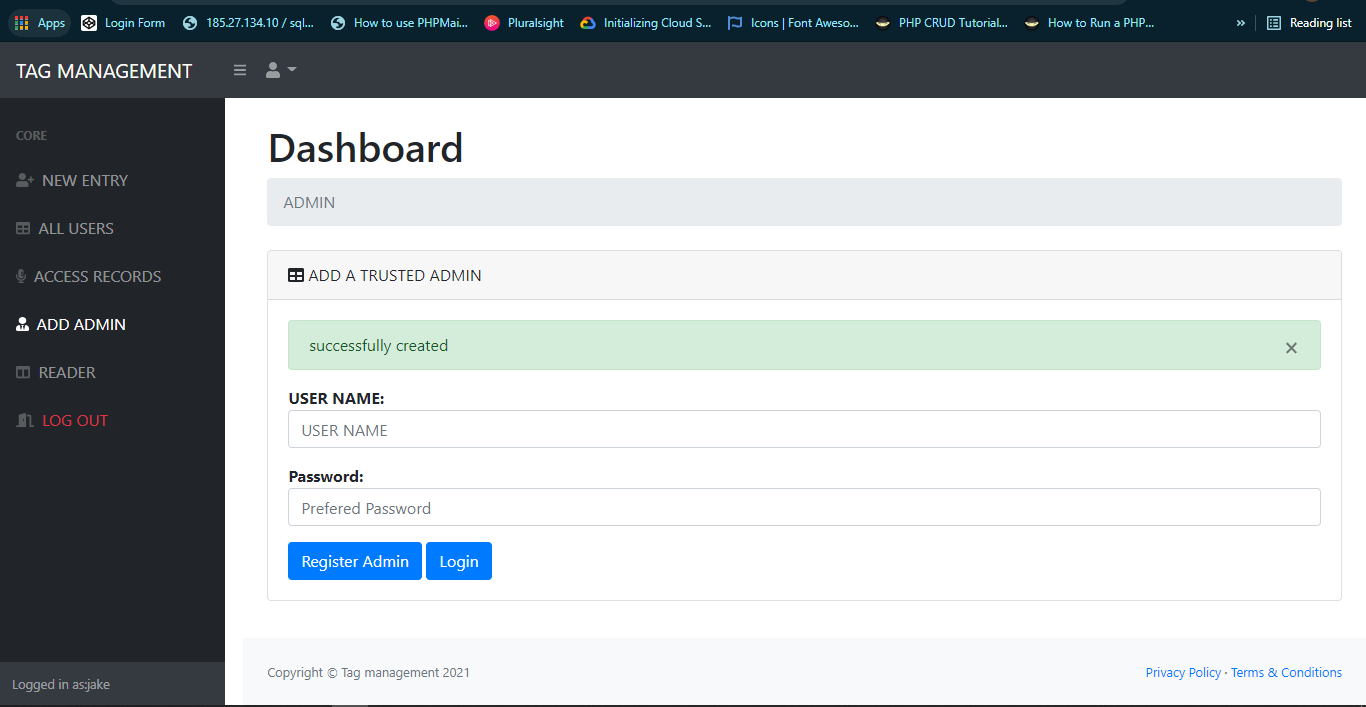
**Fig 4.2 Successful Deactivation of a tag**



**Fig 4.3 Record of All Tags and users**



**Fig 4.4 Successful card detail update**



**Fig 4.5 Successful creation of an admin**

**4.3 COST EXTIMATE OF THE PROJECT**

|  |  |  |
| --- | --- | --- |
| **NAME** | **QUANTITY** | **PRICE(#)** |
| **Arduino UNO** | **2** | **3500** |
| **LCD SCREENS** | **2** | **1200** |
| **RFID READERS** | **2** | **800** |
| **JUMPER CABLES** | **A set (60)** | **650** |
| **BREAD BOARD** | **1** | **500** |
| **STEPPER MOTOR** | **1** |  |
| **STEPPER MOTOR DRIVER** | **1** |  |
| **LOAD CELL** | **1** | **1100** |
| **ULTRASONIC SENSOR** | **2** | **700** |
|  | **TOTAL** | **8,450** |

* 1. **MAINTENANCE SYSTEM**For The purpose of maintenance and security a login interface is also designed to restrict unwanted access and keep our records confidential.  
     Other steps include

1. In the event of upgrades to libraries used ,these should be applied immediately to boost security and keep system up to date
2. Records no longer used, or students no longer within the college should be deleted to avoid using up too much storage space
3. Care should be taken when editing a student’s record so as to avoid ,mistakes in access or medical details(BMI)
   1. **AREAS OF APPLICATION**

This project is applicable in areas where access to the embedded system is limited and unwanted , areas where damage that occur from wear and tear of components parts due to constant opening and reopening of the embedded system. Also area where accommodations to non-technical officials are being used or employed.

**CHAPTER FIVE**

**SUMMARY OF MAJOR FINDINGS, RECOMMENDATIONS AND CONCLUSION**

* 1. **DISCUSSION**

Stating the steps to have come to the realization of our final system, From chapter 1, we take a general introduction of which we state the content of the study and carry out the initial problem statement which is our driving force but along the course of implementation, to mention a few from getting the frequency which our card reader can read to, then writing the code that communicates signal both in Arduino cc and python to the movement of stepper motor in the right direction were some implementation problems faced . In chapter II we had the review of the related literature where we talked about the different type’s interface being implemented or used with embedded systems. In chapter III we had the research methodology which was the Occasion for us to show our reflection line to carry out this project through the study of the system, the hardware development for the system and finally the software development for the system parts (such as the circuit code to control the motor of the door unit, the circuit of the UNO Card for storing hardware program, the LCD wiring for displaying messages of the attendance system, development of the PYTHON code interfacing the ARDUINO CC to the UI interface at a single click using PHP,HTML and CSS And the storing of data of the system in the database found in XAMPP Control Panel) of the material led piece that we have already realize and the method we used. In chapter IV we had the results which presents the different control interface along with the hardware interaction each step presents.

**5.2 CONCLUSION**

In conclusion, the objective to build a control interface to aid connection between the functional components of embedded systems and its database, and this has been achieved through the use of an access control system and a body mass index measurement system. In terms of performance and efficiency, this project has provided a convenient method of controlling embedded systems from a control interface masking it easier and more convenient for Admins (tech literate and non tech literate) to be able to use the system without the need to go into the source code at any point , the data is more organized. This

System is also a user-friendly system as data manipulation and retrieval can be done via the Interface, making it a universal control system for most control systems. Thus, it can be implemented in either an academic institution or in organizations.

**5.3 LIMITATIONS AND RECOMMENDATION**

Every good engineering design innovation has limitations and the limitations of this design

Would be improved upon in future, however some improvements that can be made on this project to increase its reliability and effectiveness are listed below;

1. An ultrasonic sensor or mode of identifying card proximity so as to auto launch process might be needed removing the need to the page with only a scan button

2. Roles can be added to the states to further tighten interface access or security.

3. Interface can be made a single page application (SPA), which does away with the need for pages to reload or submissions to be redirected.

4. Internet features can be added and media queries can be improved to support a wide range of mobile and web interface.

5. Notifications and long range communication features can also be added to increate versatility

**REFERRENCE LIST**

Mateus Krepsky Ludwich and Antônio Augusto Fröhlich. (2011) “Abstracting hardware devices to embedded java applications”.

Abhijeet Rai (2019) RFID Based Access Control Using Arduino

Retrieved from <https://www.electronicsforu.com/electronics-projects/rfid-based-access-control-using-arduino> Date accessed: 05/05/2020.

Anjana Bose (2015) “Embedded System – Characteristics, Types, Advantages & Disadvantages” retrieved from <https://electricalfundablog.com/embedded-system-characteristics-types-advantages-disadvantages>.

Arduino Weight Measurement using Load Cell and HX711 Module (2020) retrieved from

<https://circuitdigest.com/microcontroller-projects/arduino-weight-measurement-using-load-cell>

Rfid Based Security Access Control System with Arduino. By Kayode. <https://hub360.com.ng/rfid-based-security-access-control-system-with-arduino>

Anupa.K, and Channabasappa Baligar(2013)“Real-Time Communication between Aero Gas Turbine Engine Controller and Pilot Online Monitoring System “

Eetu-Pekka Kouhia (2016) “Development of an arduino-based embedded system”.

Dimitri Dimakopoulos and Ajay Kapur (2011) HIDUINO: A firmware for building driverless USB-MIDI devices using the Arduino microcontroller.

Dr. Rajendra R. Dube1, Sushama S. Pawar (2013) “Design and Implementation of Traffic Monitoring System Based on Embedded Web Technology”.

Gendlal M. Vaidya D. V. and Padole, Ph.D(2013) “Monitor Interfacing through Soft Core Processor”.

**Garth Herman**. Et al. (2014), “Implementation of Kalman Filter on PSoC-5 Microcontroller for Mobile Robot Localization”.

Anurag Ran and Preeti Chib “Computer: - A Communication Device” p1 -- p3

Arduino Basics retrieved from [https://www.tutorialspoint.com/arduino/arduino\_ultrasonic\_sensor.htm#:~:text=Advertisements,or%201%E2%80%9D%20to%2013%20feet.](https://www.tutorialspoint.com/arduino/arduino_ultrasonic_sensor.htm%23:~:text=Advertisements,or%201%E2%80%9D%20to%2013%20feet.)

Hello world! LCD implementation retrieved from <https://www.arduino.cc/en/Tutorial/LibraryExamples/HelloWorld>

Koange Longa Tamo Silas, and Feugang Joseph Ledoux(2018)“DESIGN AND REALISATION OF AN ELECTRONIC ATTENDANCE SYSTEM BASED ON RFID WITH AN AUTHOMATIC DOOR UNIT “

Agnes Voisard (2017) “Designing and integrating user interfaces of geographic database applications”

Ononiwu G., Chiagozie, Okorafor G. Nwaji. “Radio Frequency Identification (RFID) Based Attendance System with Automatic Door Unit”. in Academic Research, International, ISSNL-L: