**Industrial Mini Project Report**

**On**

**ALCOHOL DETECTION SYSTEM**

Submitted in partial fulfillment of the

Requirements for the award of the degree of Bachelor of Technology

In

Computer Science and Engineering

By

**S Siva Nagi Reddy 15241A0546**

**K Abhishek 15241A0526**

**B Kurmaiah 16245A0508**

**K Niteesh 16245A0510**

Under the Guidance of

**Dr.Ch.Mallikarjuna Rao** Professor



**Department of Computer Science and Engineering**

**GOKARAJU RANGARAJU**

**INSTITUTE OF ENGINEERING AND TECHNOLOGY**

**(Autonomous)**

**Bachupally, Kukatpally, Hyderabad-500090.**

**Department of Computer Science and Engineering**

**GOKARAJU RANGARAJU**

**INSTITUTE OF ENGINEERING AND TECHNOLOGY**

**(Autonomous)**

**Bachupally, Kukatpally, Hyderabad-500090.**



**CERTIFICATE**

This is to certify that the Industrial mini project entitled **"Alcohol Detection System"** is submitted by **S.Siva Nagi Reddy(15241A0546), K Abhishek(15241A0526), B.Kurmaiah(16245A0508),D.Niteesh(16245A0510)** in partial fulfillment of the requirement for the award of the degree in **BACHELOR OF TECHNOLOGY** in Computer Science and Engineering during the academic year 2017-2018.

**Internal Guide Head of the Department**

**Dr. Ch.Mallikarjuna Rao Dr. Ch.Mallikarjuna Rao Professor**

**External Examiner**

**DECLARATION**

I hereby declare that the industrial mini project entitled **ALCOHOL DETECTION SYSTEM"** is the work done during the period from 18-10-2017 to 14-04-2018 and is submitted in the partial fulfillment of the requirements for the award of degree of Bachelor of Technology in Computer Science and Engineering from Gokaraju Rangaraju Institute of Engineering and Technology (Autonomous).

The results embodied in this project have not been submitted to any other University or Institution for the award of any degree or diploma.

**Sivanagireddy 15241A0546**

**K.Abhishek 15241A0526**

**B.kurmaiah 16245A0508**

**K.Niteesh 16245A0510**

**ACKNOWLEDGEMENT**

There are many people who helped us directly and indirectly to complete our project successfully. We would like to take this opportunity to thank one and all.

First of all we would like to express our deep gratitude towards our internal guide **Dr.Ch.Mallikarjuna Rao, Professor,** Department of CSE for his support in the completion of my dissertation. We wish to express our sincere thanks to **Dr.Ch.Mallikarjuna Rao, HOD, Department of CSE** and also to our director **Dr.Jandhyala N Murthy** and principal **Dr.J Praveen** and our mini project co-ordinator **Dr.G.S.Bapi Raju** for providing the facilities to complete the dissertation.

We would like to thank all our faculty and friends for their help and constructive criticism during the project period. Finally we are very much indebted to our parents for their moral support and encouragement to achieve our goals.

**Sivanagireddy 15241A0546**

**K.Abhishek 15241A0526**

**B.Kurmaiah 16245A0508**

**K.Niteesh 16245A0510**

**Abstract**

Alcohol Consumption has become one of the major problem these days, mainly among

students. Alcohol is very injurious to health. Its short term effects are like Nausea, Sleep

disruption and others like blurring of speech and its long term effects are Sexual Sterility,

liver Damage, Nerve damage and many social problems like broken relationships. Sadly,

the alcohol consumption is mainly a problem among the college students.

Many students enter the college getting drunk. This spoils the college reputation and

even other students. Another problem is with the incidents with the drunk students.

They lose their sense of balance and engage in fights etc.During normal days it is easy to

identify them but during College Fests it becomes very hard to identify the individuals.

The main aim of this project is to detect the alcohol percentage and if it is more than the

limited value then record their names. The alcohol sensor MQ3 is used to detect the alcohol

percentage in the breath when the person exhales air and it is sent to the arduino. Through

Arduino the value of alcohol percentage is sent to the mobile application and the Percentage

is displayed in it. If the alcohol content is more than the permissible level then their details

and vehicle registration numbers are recorded inorder to take action on them.

**Contents Page.No**

Certificate 2

Acknowledgement 3

Declaration 4

Abstract 5

Contents

Chapter 1: **Introduction** 8

1.1.Description 8

1.2.Project Overview 8

Chapter 2: **System Analysis** 9

2.1.Existing System 9

2.2.Proposed System 9

2.3.Software/HardWare Components required 10

2.4.1System Architecture 11

2.4.2.Data Flow Diagram 11

Chapter 3: **Implementation** 12

3.1.Arduino 12

3.2.Alcohol Sensor MQ-3 16

3.3.Bluetooth Module HC-05 18

3.4.1.Arduino IDE 21

3.4.2.MIT AppInventor 21

3.5.Working 22

3.6.Code Snippets 23

3.6.1.Arduino Source code 23

3.6.2.Mobile Application Source code 26

Chapter 4: **Screenshots** 32

Chapter 5: **Conclusion** 33

Chapter 6: **Benefits** 34

Chapter 7: **Bibliography** 35

1. **INTRODUCTION**

The project is about to notice the students who comes to college consuming alcohol.

This project is used during the college fest ,here in this project we use alcohol sensor

(MQ3) to detect the alcohol consumed by the student. We also use arduino board and

this alcohol sensor is connected to arduino which consists of four pins.When student

exhales through this sensor then it sends values to arduino and displays alcohol per-

centage ,inorder to display this alcohol percentage we are using an app called MITapp

where alcohol percentage can be displayed and student details are noted.

**1.1.DESCRIPTION**

When student exhales through this sensor this sensor gives values to the arduino the alc-

ohol sensor consists of four pins two pins are for analog input and digital input and the

other two are for vcc and ground these pins are connected to the pins of arduino ,here we

use only analog input.As already described that we use MIT app for displaying alcohol

percentage we use blutooth module HC05 to connect with an app .The HC05 consists of

pins like TX and RX pins which is connected to the arduino board .

**1.2.PROJECT OVERVIEW**

The main aim of this project is to build an alcohol detection system to find drunk

students efficiently.

The strategy used is to build alcohol detector with the help of arduino and integrate it

with MIT App Inventor.

The budget of the project is very less when compared to cost of alcohol detectors

available in the market.

**2.SYSTEM ANALYSIS**

**2.1Existing System:**

Presently there is no alcohol detection system implemented in most of the colleges. Even if an alcohol detector is implemented, it is hard to manually record the details of students.

**2.2.Proposed System:**

The proposed system is to implement this alcohol detection system at the college entrance. Person has to breathe out in front of Alcohol sensor .The alcohol sensor sends the readings to microcontroller. Through Bluetooth module the readings are sent to mobile application. If the alcohol percentage is more than the threshold value then a new page is displayed in the application.Details like vehicle registration numbers are recorded and vehicles are captured.

Vehicles are given back only if students provide their id card. It can be used to take action on them later. If the drunken student does not have a vehicle, his ID card can be taken . This can be used to identify the student to take certain action against them.For recording the details the examiner should install the created mobile application "ALCOHOL DETECTOR" on his android smart phone.

**2.3 Hardware & Software Requirements**

## 2.3.1. Hardware Requirements:

Alcohol Sensor MQ-3

Bluetooth Module HC-05

Arduino UNO

Connecting Wires

High Voltage battery

## 2.3.2. Software Requirements:

Android Mobile

Alcohol detector Mobile Applictation

MIT App Inventor

Arduino IDE

2.4.1.**SYSTEM ARCHITECTURE**

Diagram 2-1

2.4.2.**DATA FLOW DIAGRAM**

Diagram 2-2

**3. IMPLEMENTATION**

Alcohol detection system is implemented as a combination of mobile application with the physical components Arduino, Alcohol sensor Mq3 and Bluetooth Module.

**3.1.Arduino:**

**Arduino Uno** is a microcontroller board based on the ATmega328P . It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.. You can tinker with your UNO without worrying too much about doing something wrong, worst case scenario you can replace the chip for a few dollars and start over again.

Diagram 3-1

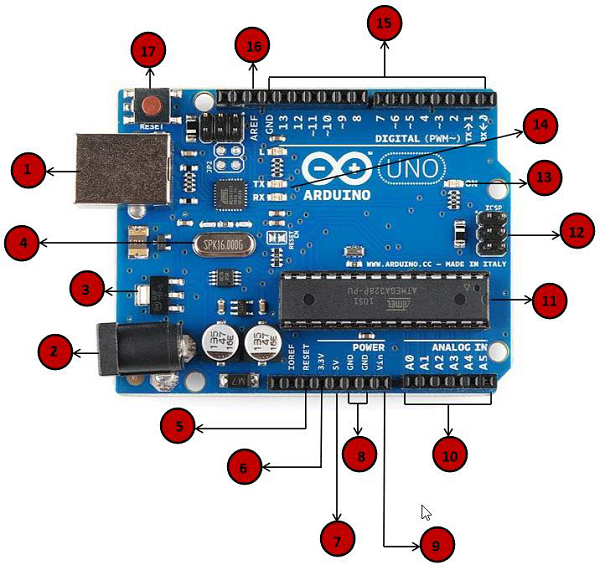
"Uno" means one in Italian and was chosen to mark the release of Arduino Software (IDE) 1.0. The Uno board and version 1.0 of Arduino Software (IDE) were the reference versions of Arduino, now evolved to newer releases. The Uno board is the first in a series of USB Arduino boards, and the reference model for the Arduino platform; for an extensive list of current, past or outdated boards see the Arduino index of boards.

Diagram 3-2:

|  |  |
| --- | --- |
| 1 | **Power USB**  Arduino board can be powered by using the USB cable from your computer. All you need to do is connect the USB cable to the USB connection (1). |
| 2 | **Power (Barrel Jack)**  Arduino boards can be powered directly from the AC mains power supply by connecting it to the Barrel Jack (2). |
| 3 | **Voltage Regulator**  The function of the voltage regulator is to control the voltage given to the Arduino board and stabilize the DC voltages used by the processor and other elements. |
| 4 | **Crystal Oscillator**  The crystal oscillator helps Arduino in dealing with time issues. How does Arduino calculate time? The answer is, by using the crystal oscillator. The number printed on top of the Arduino crystal is 16.000H9H. It tells us that the frequency is 16,000,000 Hertz or 16 MHz. |
| 5 | **Arduino Reset**  You can reset your Arduino board, i.e., start your program from the beginning. You can reset the UNO board in two ways. First, by using the reset button (17) on the board. Second, you can connect an external reset button to the Arduino pin labelled RESET (5). |
| 6,7,8,9 | **Pins (3.3, 5, GND, Vin)**   * 3.3V (6) − Supply 3.3 output volt * 5V (7) − Supply 5 output volt * Most of the components used with Arduino board works fine with 3.3 volt and 5 volt. * GND (8)(Ground) − There are several GND pins on the Arduino, any of which can be used to ground your circuit. * Vin (9) − This pin also can be used to power the Arduino board from an external power source, like AC mains power supply. |
| 10 | **Analog pins**  The Arduino UNO board has five analog input pins A0 through A5. These pins can read the signal from an analog sensor like the humidity sensor or temperature sensor and convert it into a digital value that can be read by the microprocessor. |
| 11 | **Main microcontroller**  Each Arduino board has its own microcontroller (11). You can assume it as the brain of your board. The main IC (integrated circuit) on the Arduino is slightly different from board to board. The microcontrollers are usually of the ATMEL Company. You must know what IC your board has before loading up a new program from the Arduino IDE. This information is available on the top of the IC. For more details about the IC construction and functions, you can refer to the data sheet. |
| 12 | **ICSP pin**  Mostly, ICSP (12) is an AVR, a tiny programming header for the Arduino consisting of MOSI, MISO, SCK, RESET, VCC, and GND. It is often referred to as an SPI (Serial Peripheral Interface), which could be considered as an "expansion" of the output. Actually, you are slaving the output device to the master of the SPI bus. |
| 13 | **Power LED indicator**  This LED should light up when you plug your Arduino into a power source to indicate that your board is powered up correctly. If this light does not turn on, then there is something wrong with the connection. |
| 14 | **TX and RX LEDs**  On your board, you will find two labels: TX (transmit) and RX (receive). They appear in two places on the Arduino UNO board. First, at the digital pins 0 and 1, to indicate the pins responsible for serial communication. Second, the TX and RX led (13). The TX led flashes with different speed while sending the serial data. The speed of flashing depends on the baud rate used by the board. RX flashes during the receiving process. |
| 15 | **Digital I/O**  The Arduino UNO board has 14 digital I/O pins (15) (of which 6 provide PWM (Pulse Width Modulation) output. These pins can be configured to work as input digital pins to read logic values (0 or 1) or as digital output pins to drive different modules like LEDs, relays, etc. The pins labeled “~” can be used to generate PWM. |
| 16 | **AREF**  AREF stands for Analog Reference. It is sometimes, used to set an external reference voltage (between 0 and 5 Volts) as the upper limit for the analog input pins. |

The key component for the detection of Alchol Percentage is the MQ-3 Sensor.The description of Alcohol sensor is given below.

**3.2.MQ-3 Alcohol Sensor**

Alcohol sensor  named MQ-3, which detects ethanol in the air. It is one of the straightforward gas sensors so it works almost the same way with other gas sensors. It costs $6.90.Typically, it is used as part of the breathalyzers or breath testers for the detection of ethanol in the human breath.

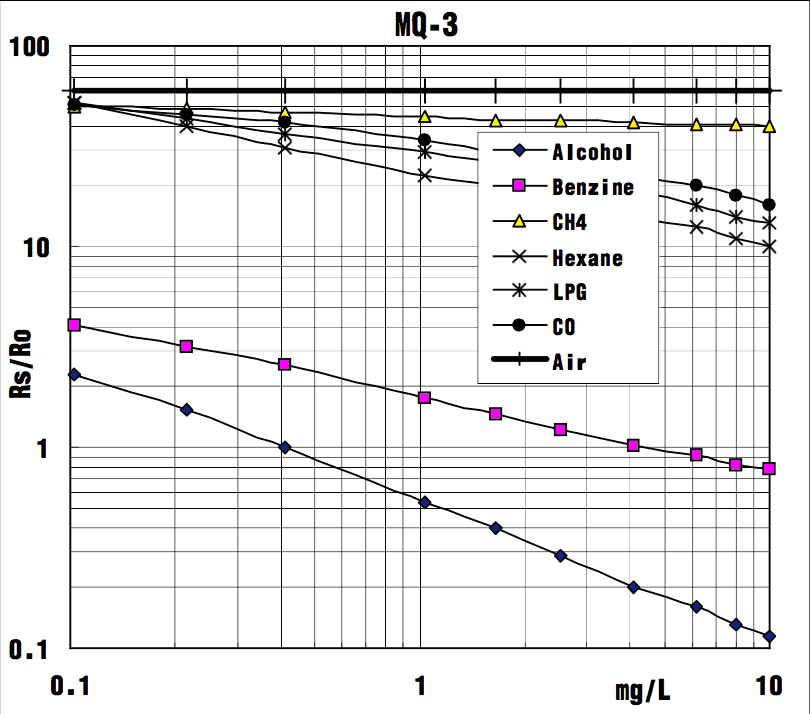
This is an Analog output sensor. This needs to be connected to any one Analog socket in . The sensor makes use of the analog pins of the arduino.

Diagram 3-3

**Working Principle:**

Sensitive material of MQ-3 gas sensor is SnO2, which with lower conductivity in clean air. When the target alcohol gas exist, The sensor’s conductivity is more higher along with the gas concentration rising.

The different ranges of sensitivities to various compounds like Alcohol , Benzene, CH4 , Hexane, LPG, Air are given below.

Diagram:3-4

**Connecting MQ-3 to Arduino**

It is possible to connect the Grove module to Arduino directly by using jumper wires by using the connection as shown in the table below:

The output voltage from the Gas sensor increases when the concentration of gas increases.

|  |  |
| --- | --- |
| **MQ-3 Pin** | **Arduino Pin** |
| 5v | Vin |
| Gnd | Gnd |
| Analog o/p | Any Analog i/p |

**3.3.BLUETOOTH MODULE**

Bluetooth Module is also an important component which acts as a bridge between the arduino and the mobile Application. The name of bluetooth module is HC-05.

[](https://wiki.eprolabs.com/index.php?title=File:HC-05.jpg)

Diagram 3-5

**HC‐05 module** is an easy to use **Bluetooth SPP (Serial Port Protocol) module**,designed for transparent wireless serial connection setup.The HC-05 Bluetooth Module can be used in a Master or Slave configuration, making it a great solution for wireless communication.This serial port bluetooth module is fully qualified **Bluetooth V2.0+EDR (Enhanced Data Rate)** 3Mbps Modulation with complete 2.4GHz radio transceiver and baseband. It uses **CSR Bluecore 04**‐External single chip Rluetooth system with CMOS technology and with AFH (Adaptive Frequency Hopping Feature).

**PIN DESCRPTION**

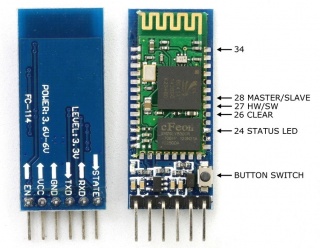
[](https://wiki.eprolabs.com/index.php?title=File:FC-114.jpg)

Diagram 3-6

The HC-05 Bluetooth Module has 6 pins. They are as follows:

**ENABLE:**

When enable is pulled **LOW**, the module is disabled which means the module will **not turn on** and it **fails to communicate**. When enable is **left open or connected to 3.3V**, the module is enabled i.e the module **remains on** and **communication also takes place**.

**Vcc:**

Supply Voltage 3.3V to 5V

**GND:**

Ground pin

**TXD & RXD:**

These two pins acts as an UART interface for communication.

**STATE:**

It acts as a status indicator. When the module is **not connected to / paired** with any other bluetooth device, signal goes **Low**. At this **low state**, the **led flashes continuously** which denotes that the module is **not paired** with other device. When this module is **connected to/paired** with any other bluetooth device, the signal goes **High**. At this **high state**, the **led blinks with a constant delay** say for example 2s delay which indicates that the module is **paired**.

**BUTTON SWITCH:**

This is used to switch the module into AT command mode.To enable AT command mode,press the button switch for a second.With the help of AT commands,the user can change the parameters of this module but only when the module is not paired with any other BT device.If the module is connected to any other bluetooth device, it starts to communicate with that device and fails to work in AT command mode.

**CONNECTING HC-05 to ARDUINO**

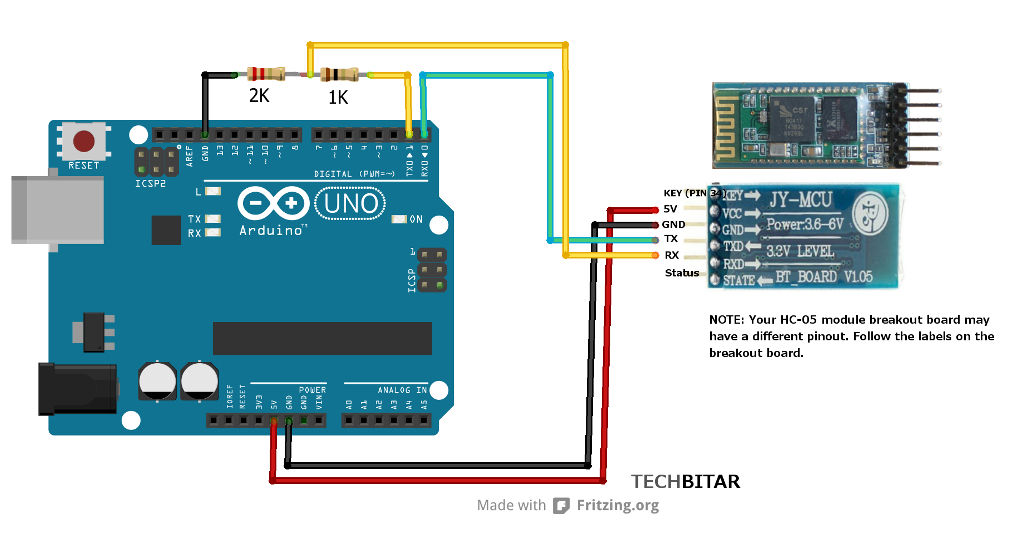


Diagram 3-7

* Vcc is connected to the 5v.
* Gnd is connected to the Gnd
* TX of HC-05 is connected to the RX of the Arduino.
* RX of HC-05 is connected to the TX of the Arduino.

**3.4.1.ARDUINO IDE**

We use the Arduino IDE for writing the source code for Arduino UNO. The description of the software is given below.

The Arduino Integrated Development Environment - or Arduino Software (IDE) - contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino and Genuino hardware to upload programs and communicate with them.

**3.4.2.MIT APP INVENTOR**

We use the Mit app inventor Environment for creating the mobile application. The description of the Mit app Inventor is given below.

**App Inventor for Android** is an open-source web application originally provided by [Google](https://en.wikipedia.org/wiki/Google), and now maintained by the Massachusetts Institute of Technology (MIT).

It allows newcomers to computer programming to create software applications for the  Android operating system (OS).

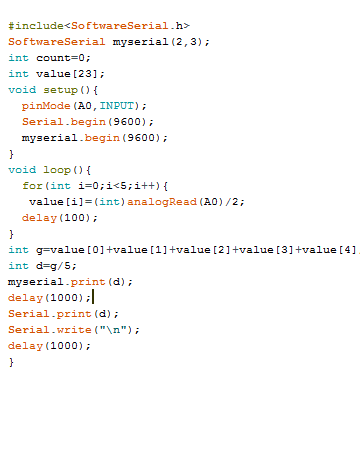
It uses a graphical interface, very similar to Scratch and  user interface, which allows users to drag-and-drop visual objects to create an application that can run on Android devices.

**3.5.WORKING:**

* The entire setup is kept in a conceivable box and can be carried any where.
* The mobile application created in the Mit appinventor should be installed in the user's android mobile.
* During the test the testee should blow air or exhale onto the Alcohol Sensor MQ-3.
* Alcohol Sensor being sensitive to alcohol produces value accordingly.
* The value produced by Alcohol Sensor is an analog value.
* Through Jumper wires the analog value is sent to the Arduino UNO.
* From the Arduino UNO the value is sent to Bluetooth Module HC-05.
* Through the Bluetooth Module the value is sent to the Mobile Application.
* In the mobile application the value is displayed.
* If the alcohol level is greater than 100 then the person should be stopped from entering the college and his details are recorded.
* Else he is allowed into the college.

**3.6.CODE SNIPPETS:**

**3.6.1Arduino Source Code:**



In the above program

**Software Serial:**

The Arduino hardware has built-in support for serial communication on pins 0 and 1 (which also goes to the computer via the USB connection). The native serial support happens via a piece of hardware (built into the chip) called a [UART](http://en.wikipedia.org/wiki/UART). This hardware allows the Atmega chip to receive serial communication even while working on other tasks, as long as there room in the 64 byte serial buffer.

The SoftwareSerial library has been developed to allow serial communication on other digital pins of the Arduino, using software to replicate the functionality (hence the name "SoftwareSerial"). It is possible to have multiple software serial ports with speeds up to 115200 bps.

**myserial(Rx,Tx):**

It is the object of the SoftwareSerial class.

**setup():**

Use it to initialize variables, pin modes, start using libraries, etc. The setup() function will only run once, after each powerup or reset of the Arduino board.

**pinMode(A0, INPUT):**

This is used to set A0 to receive the input into the arduino board.

**myserial.begin(9600):**

It is used to initialize the serial communication in bluetooth.

**loop():**

After creating a setup() function, which initializes and sets the initial values, the loop() function does precisely what its name suggests, and loops consecutively, allowing your program to change and respond. Use it to actively control the Arduino board.

**analogRead(A0):**

analogRead() returns a number between 0 and 1023 that is proportional to the amount of voltage being applied to the pin.

Here we are using A0 as the input pin.

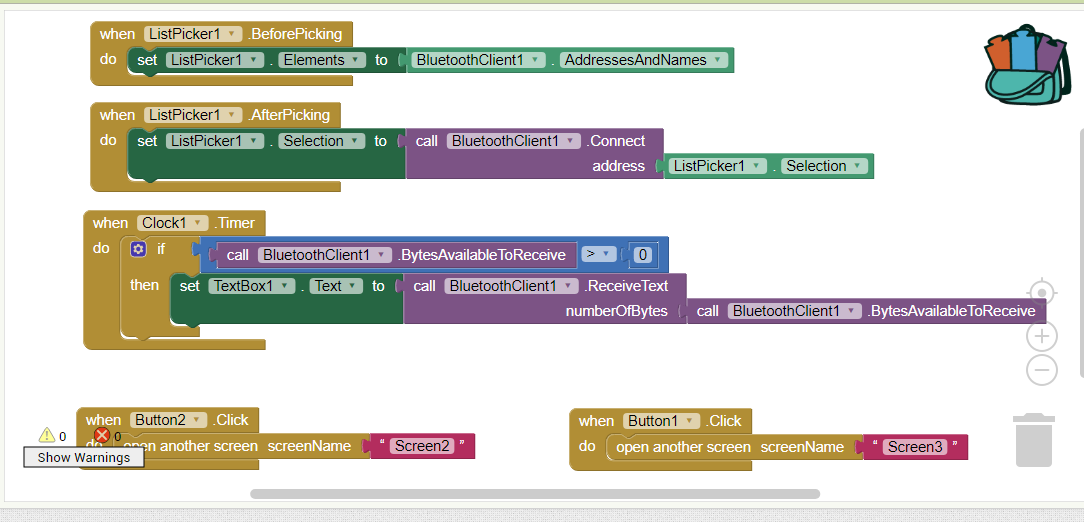
**mySerial.print():**

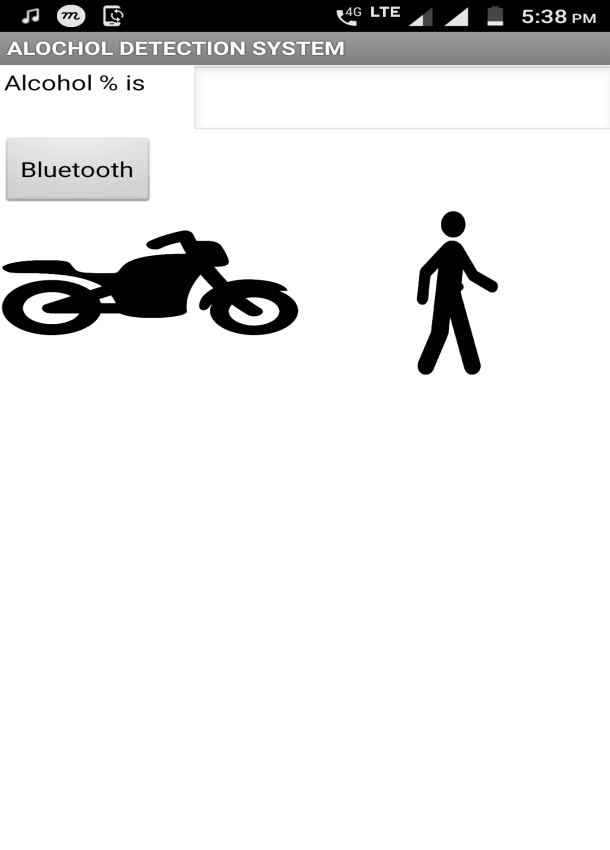
mySerial.print() prints data to the transmit pin of the software serial port.

**3.6.2.MOBILE APPLICATION CODE:**

Mobile Application has been developed using the MIT AppInventor software. It is an open source software.It is a GUI and code is written as blocks.

Screen 1:



**Screen 1:**

The output for the above code looks like this. This page is very crucial to the mobile application.This is because we display the alcohol content in this page.

**List Picker:** It is used to select one choice from available multiple choices.

We set this list picker to the Bluetooth Client addresses.

**Bluetooth Client:** It is a nonvisible component embedded in the program which helps us to connect to near by bluetooth devices.

After connecting to bluetooth client we start the **Clock** to continuously sense if there is any data is incoming.If there is any data available it accepts it and prints in the text box.

Here we have two choices.If the alcohol level is less than 100 then the person is determined as not drunk.

Else if the alcohol level is greater than 100 then we need to record the details.

Here we have two buttons. 1)Vehicle

2)Pedestrian

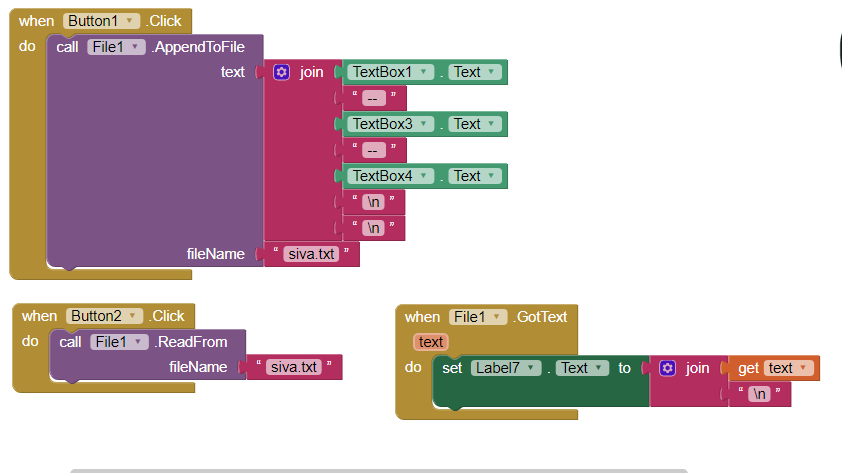
If the person came on vehicle then we need to click onto the Vehicle icon. Screen 2 will open

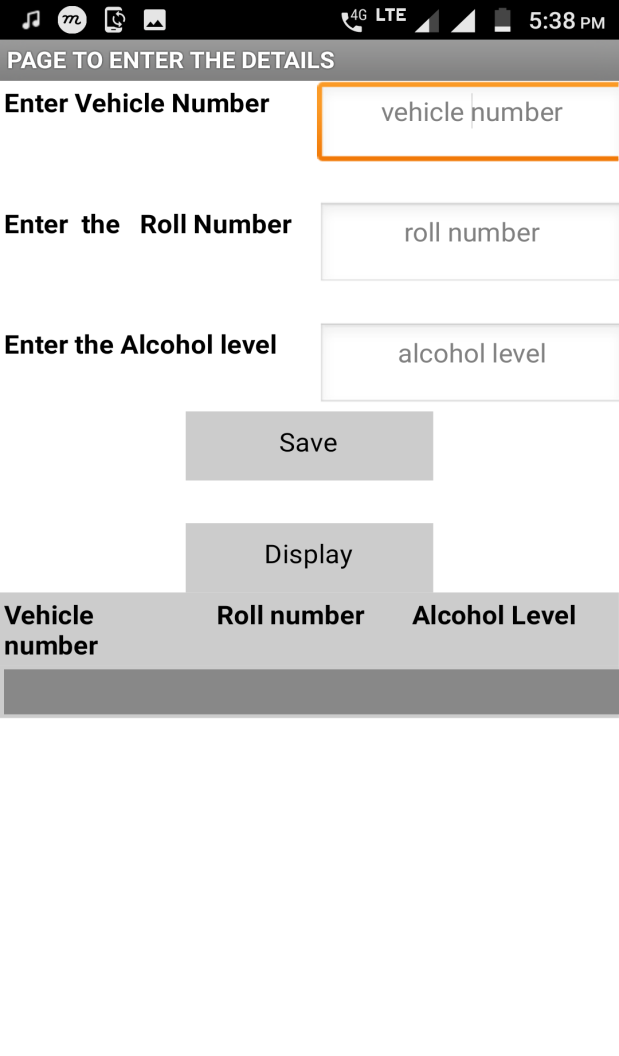
If the person is a pedestrian we need to click on the Pedestrian icon. Screen 3 will open.

Screen 2:

Screen 2 will be opened when we click on the vehicle icon in the Screen 1.

Screen2 deals with the saving of the details of the student.



**Screen2**

The output of the above program looks like this. As mentioned before it details with the storing and display of student details.

In this we have two buttons

1)Save

2)Display

**Save**: On Clicking save the details are saved

**Display:**Displays the stored values.

Here we use the **FILE** as the storage unit.

In the given three text boxes we need to enter the details of the students accordingly.

Then we need to press the Save button.

On Clicking the Save button the contents of the 3 text boxes are joined as a single record and appended to the file.Here the file name is "siva.txt".

We can create our own files and the text can be stored into that file.

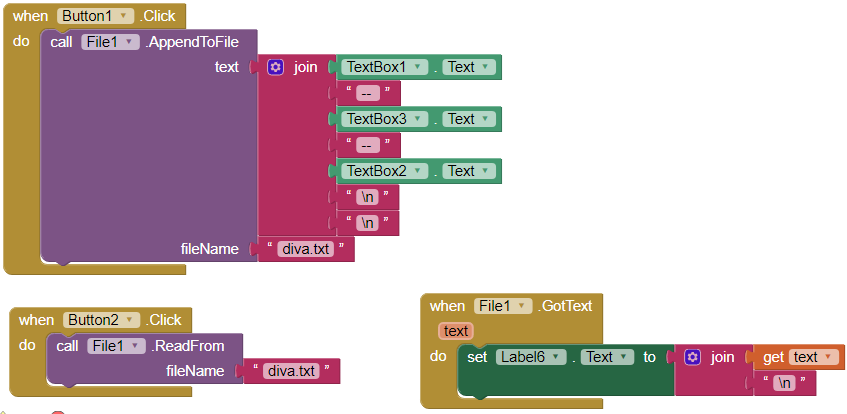
If we want to display the details of the students we need to press the **Display** button.

On clicking the display button, the file calls the read() method and the records are retrieved from the file and are displayed in the below table.

Screen 3:

Screen 3 will be opened when we click on the pedestrian icon in the Screen 1.

Screen3 deals with the saving of the details of the student.



**Screen 3:**

The output of the above program looks like this. As mentioned before it details with the storing and display of student details. It is similar to Screen 2.

In this we have two buttons

1)Save

2)Display

**Save**: On Clicking save the details are saved

**Display:** Displays the stored values.

Here we use the **FILE** as the storage unit.

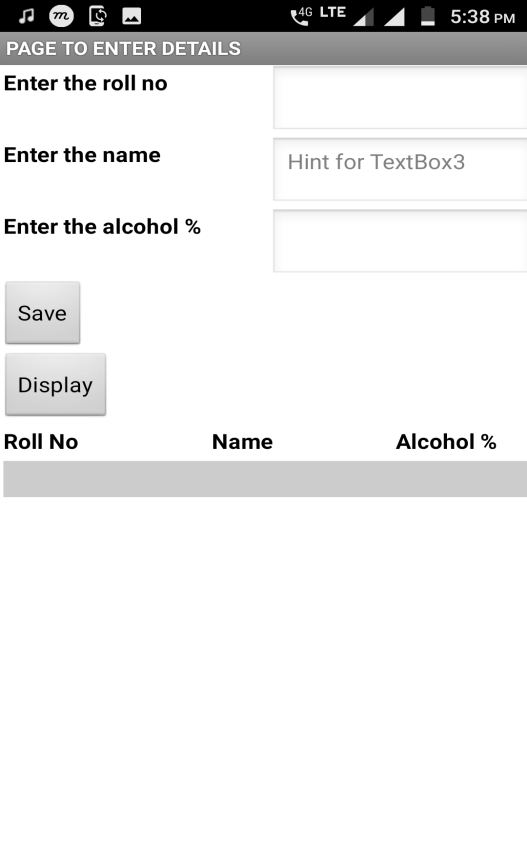
In the given three text boxes we need to enter the details of the students accordingly.

Then we need to press the Save button.

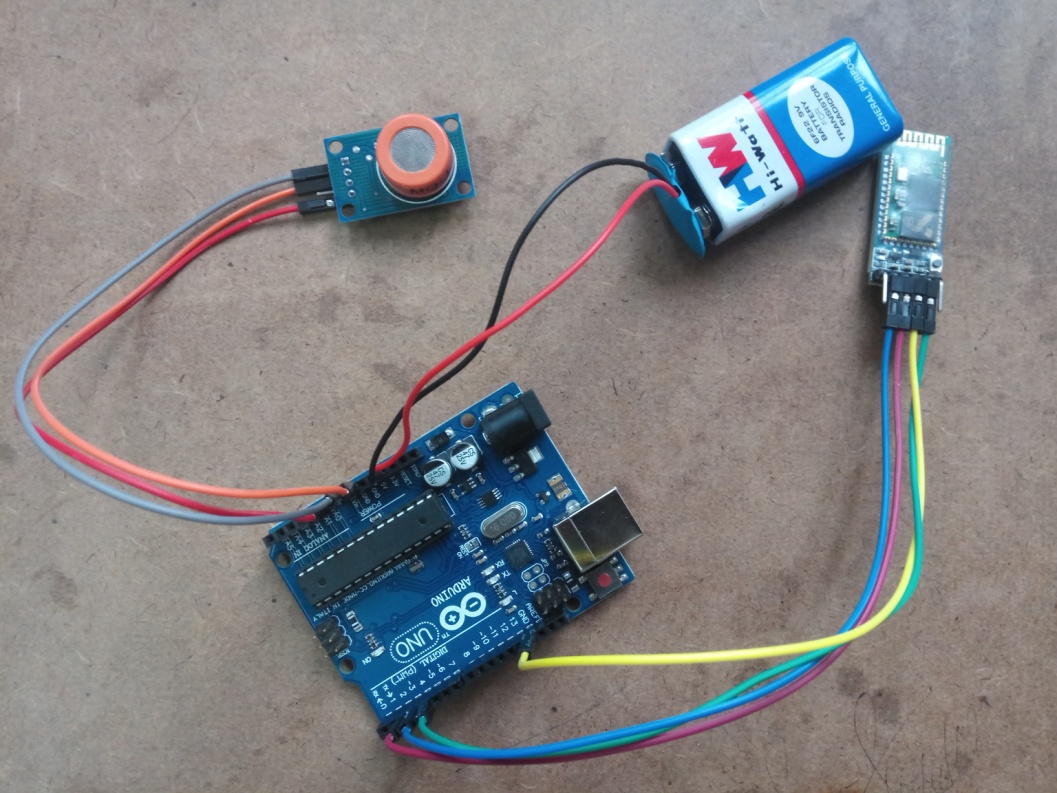
On Clicking the Save button the contents of the 3 text boxes are joined as a single record and appended to the file.Here the file name is "diva.txt".

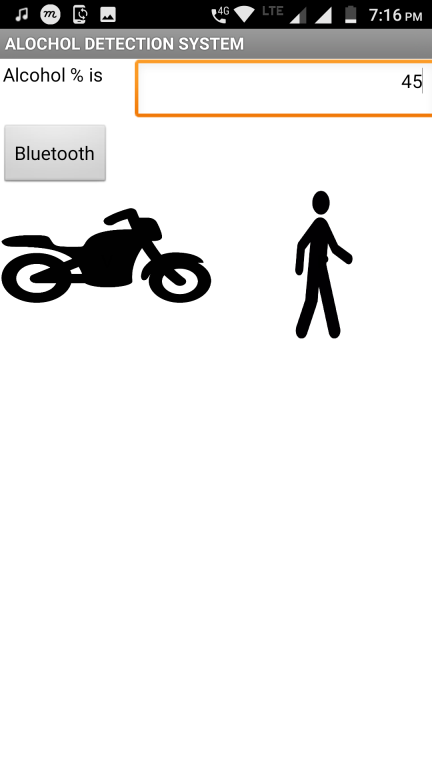
We can create our own files and the text can be stored into that file.

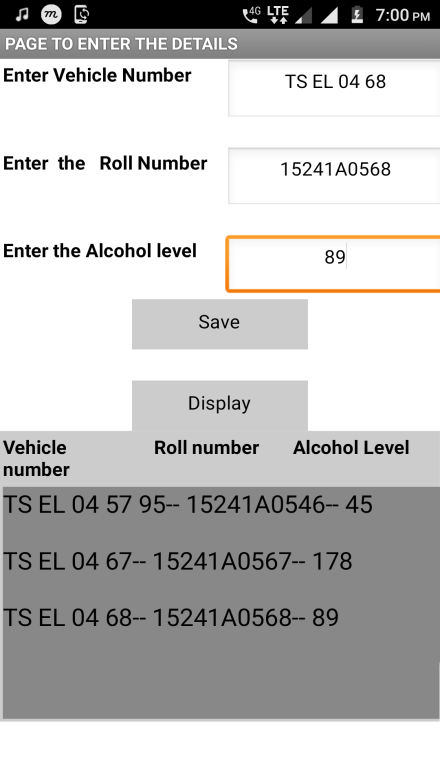
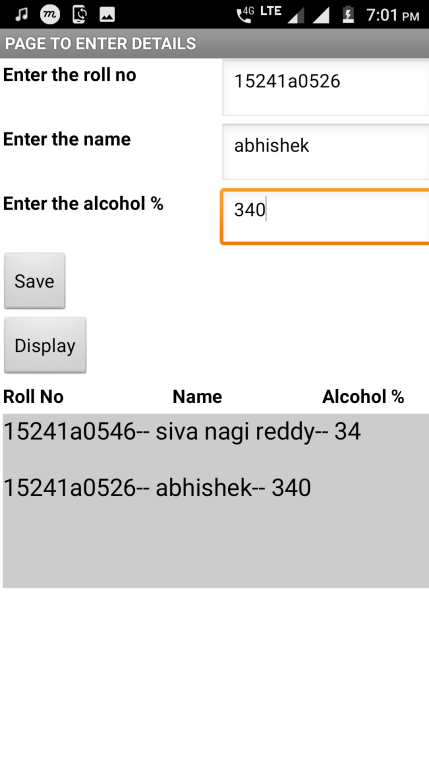
If we want to display the details of the students we need to press the **Display** button.

On clicking the display button, the file calls the read() method and the records are retrieved from the file and are displayed in the below table.

**4.1.Screen Shots:**







**5.CONCLUSION**

Our project Alcohol Detection system has been successfully implemented. The Alcohol detection system project can immensely decrease the rate of the drunken students entering into the college. Thus it can make the college a safe place without any undesired incidents happening. Considering the errors in the measurement , it can be improved.

**6.BENEFITS**

As the prevention of drunk driving is a major concern for society today, the advantages of using alcohol detection system  to detect the level of alcohol in a person’s system are numerous. Below you’ll find an outline of the main advantages of their use:

* It is easy to use.
* It is portable. It can be carried anywhere.
* It can be applied to any public places like schools and colleges.
* It decreases the rate of drunk students entering into the college immensly.
* Finally, it makes the college a better place.

**7.BIBLIOGRAPHY**

* Hök B, Pettersson H, Andersson G. Contactless measurement of breath alcohol. Paper presented at: Micro Structure Workshop; May 9–10, 2006; Västerås, Sweden.
* "Sensor Workshop" Sensor report of mq-3

http://sensorworkshop.blogspot.com/2008/04/sensor-report-mq3-gas-sensor.html

### Serial Port Bluetooth Module (Master/Slave) : HC-05 - ITEAD Wiki

https://www.itead.cc/wiki/Serial\_Port\_Bluetooth\_Module\_(Master/Slave)\_:\_HC-05

### Arduino - Software

https://www.arduino.cc/en/Main/Software

* Building a Mobile App: Design and Program Your Own App! by Sarah Guthals [ ISBN: 978-1-119-37642-2 ]Mar 2017

|  |  |  |  |
| --- | --- | --- | --- |
| **CONTENTS** | | | |
| Certificate | |  | I |
| Declaration | |  | Ii |
| Acknowledgement | |  | Iii |
| Abstract | |  | Iv |
| Hardware and Software requirements | |  | V |
| Chapter 1: Hadoop Overview | |  | 01 |
|  | 1.1 .History of Hadoop |  | 02 |
|  | 1.2. Hadoop Architecture |  |  |
|  | 1.2.1 Hadoop Versions | | 03 |
|  | **1.2.2 High Availability (HA) in Hadoop 2.0** | | 04 |
|  | 1.2.3 YARN – Yet Another Resource Negotiator | | 05 |
|  | 1.2.4 Map Reduce | | 06 |
|  | 1.3 Hadoop Distributed File System |  | 07 |
|  | 1.4 Advantages and Disadvantages of Hadoop |  | 08 |
|  | 1.4.1 Advantages of Hadoop |  | 08 |
|  | 1.4.2 Disadvantages of Hadoop |  | 08 |
|  | 1.5 Hadoop Components and Supporting Tools |  | 09 |
|  | 1.5.1 Pig |  | 09 |
|  | 1.5.2 Hive |  | 09 |
|  | **1.5.3 Sqoop** |  | 09 |
|  | **1.5.4 Flume** |  | 10 |
|  | **1.5.5 HBase** |  | 10 |
|  | **1.5.6 Oozie** |  | 10 |
|  | **1.5.7 Zookeeper** |  | 10 |
|  | **1.5.8** Ambari |  | 11 |
|  | **1.5.9** Mahout |  | 11 |
| Chapter 2: Bigdata Overview | |  |  |
|  | 2.1 What is Big Data |  | 12 |
|  | 2.2 Big Data Challenges |  | 13 |
| Chapter 3: Hadoop Installation | |  |  |
|  | 3.1 Hypervisor |  |  |
|  | 3.1.1 Bare-metal hypervisor | | 14 |
|  | 3.1.2 Hosted hypervisor | | 15 |
|  | 3.2 What is VMWare |  | 15 |
|  | 3.3 Installation of VMWare and Big insights |  | 16 |
| Chapter 4: Analysis of BigData | |  |  |
|  | 4.1 Environmental setup |  | 18 |
|  | 4.2 Data collection |  | 19 |
|  | 4.3 Preprocessing of data |  | 20 |
|  | 4.4 Data Processing |  | 21 |
|  | 4.5 Output Graphs after analysis |  | 22 - 28 |
| Chapter 5: Conclusion | |  | 29 |
| Chapter 6: References | |  | 30 |