

ESCUELA POLITÉCNICA NACIONAL

ESCUELA DE FORMACIÓN DE TECNÓLOGOS

**IMPLEMENTACIÓN DE PROTOTIPOS DE SISTEMAS DE ALERTA
BASADOS EN COMUNICACIÓN INALÁMBRICA**

**IMPLEMENTACIÓN DE UN PROTOTIPO DE SISTEMA DE ALERTA
POR DETECCIÓN DE ALCOHOL**

**TRABAJO DE INTEGRACIÓN CURRICULAR PRESENTADO COMO
REQUISITO PARA LA OBTENCIÓN DEL TÍTULO DE TECNÓLOGO SUPERIOR
EN REDES Y TELECOMUNICACIONES**

EDISSON ROLANDO ORTIZ SALAZAR

DIRECTOR: FANNY PAULINA FLORES ESTÉVEZ

DMQ, agosto 2022

CERTIFICACIONES

Yo, EDISSON ROLANDO ORTIZ SALAZAR declaro que el trabajo de integración curricular aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

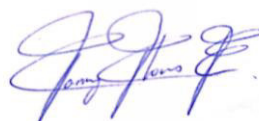


EDISSON ROLANDO ORTIZ SALAZAR

edisson.ortiz@epn.edu.ec

edissonortiz1996@yahoo.com

Certifico que el presente trabajo de integración curricular fue desarrollado por EDISSON ROLANDO ORTIZ SALAZAR, bajo mi supervisión.



FANNY PAULINA FLORES ESTÉVEZ

DIRECTOR

fanny.flores@epn.edu.ec

DECLARACIÓN DE AUTORÍA

A través de la presente declaración, afirmamos que el trabajo de integración curricular aquí descrito, así como el (los) producto(s) resultante(s) del mismo, son públicos y estarán a disposición de la comunidad a través del repositorio institucional de la Escuela Politécnica Nacional; sin embargo, la titularidad de los derechos patrimoniales nos corresponde a los autores que hemos contribuido en el desarrollo del presente trabajo; observando para el efecto las disposiciones establecidas por el órgano competente en propiedad intelectual, la normativa interna y demás normas.



EDISSON ROLANDO ORTIZ SALAZAR

DEDICATORIA

Dedico este proyecto a mis padres que sin lugar a duda han sabido acompañarme en toda situación de manera incondicional y han dado su mejor esfuerzo para que pueda salir adelante, a mi hermana que es un fuerte soporte emocional y a mi familia en general que de una u otra manera han brindado su apoyo en circunstancias adversas. Todas las metas que he cumplido y llegaré cumplirlas será gracias a ellos.

Edisson.

AGRADECIMIENTO

En primera instancia agradezco a la comunidad politécnica, gracias a sus docentes y autoridades he sabido llevar a cabo una trayectoria digna dentro de la carrera, para de este modo poder formarme profesionalmente incluso fuera del aspecto académico. A la Ing. Fanny Flores que con paciencia y mucha dedicación ha sabido guiar el desarrollo del proyecto de titulación, aportando con su magnífico conocimiento, experiencia y pasión por la docencia.

A mis amigos, que han llegado a convertirse en mi segunda familia y han estado conmigo en los buenos y malos momentos, a mi novia que ha permanecido a mi lado a pesar de que la universidad carcoma mi tiempo de manera inimaginable y a todos quienes han podido ser parte de mi formación profesional.

Edisson.

ÍNDICE DE CONTENIDOS

CERTIFICACIONES.....	I
DECLARACIÓN DE AUTORÍA	II
DEDICATORIA.....	III
AGRADECIMIENTO.....	IV
ÍNDICE DE CONTENIDOS	V
RESUMEN.....	VII
ABSTRACT.....	VIII
1 DESCRIPCIÓN DEL COMPONENTE DESARROLLADO.....	1
1.1 Objetivo general.....	2
1.2 Objetivos específicos.....	2
1.3 Alcance	2
1.4 Marco Teórico	3
Introducción.....	3
<i>Bluetooth</i>	3
Placa Arduino Nano	4
Sensor de gases MQ-3	4
Módulo <i>bluetooth</i> HC-05	5
2 METODOLOGÍA.....	6
3 RESULTADOS	8
3.1 Requerimientos del prototipo.....	8
3.2 Definición de componentes de hardware y software	10
Hardware	10
Software	12
App Inventor.....	12
Arduino IDE	13
3.3 Diseño del prototipo	13
Circuito electrónico.....	13

Consumo energético	14
Rangos de acción.....	16
Diagrama de flujo	17
Modelo 3D	19
3.4 Implementación del prototipo	19
Elaboración de la placa de circuito impreso	19
Programación para el Arduino Nano	23
Montaje del prototipo en el <i>case</i>	28
Creación de la aplicación móvil	28
Costos de materiales	30
3.5 Pruebas de funcionamiento	31
4 CONCLUSIONES.....	35
5 RECOMENDACIONES	37
6 REFERENCIAS BIBLIOGRÁFICAS	37
7 ANEXOS.....	40
ANEXO I: Certificado de Originalidad	i
ANEXO II: Enlaces	ii
ANEXO III: Códigos Fuente	iii

RESUMEN

Este proyecto de titulación está enfocado en el desarrollo de un prototipo de un sistema de medición de alcohol que cuenta con conectividad *bluetooth* y es controlado por la placa de experimentación Arduino. Se envía notificaciones a un dispositivo móvil conectado a través de una aplicación celular, dependiendo los rangos de lectura. El proceso aborda puntos estipulados del siguiente modo:

La sección 1 se enfoca en el desarrollo general del proyecto y la propuesta de objetivos para este, tanto el general como los específicos.

La sección 2 describe la metodología aplicada en conformidad con los objetivos planteados.

En la sección 3 se presenta la selección de materiales para el desarrollo del proyecto, tanto software como hardware. Además, aplica la realización del circuito electrónico aplicado en el prototipo y la implementación de todos los elementos para conformar el sistema de medición de alcohol.

En las secciones 4 y 5 se presenta el desarrollo de las conclusiones a relucir del prototipo creado y se presentan las recomendaciones debidas en caso de tomar el proyecto como referencia.

Finalmente, en las secciones 6 y 7 se añaden las referencias bibliográficas y los concernientes anexos que sirven como base y sustento para el desarrollo del prototipo.

PALABRAS CLAVE: medición de alcohol, *bluetooth*, Arduino, aplicación celular, rangos de lectura.

ABSTRACT

This degree project is focused on the development of a prototype of an alcohol measurement system that has bluetooth connectivity and is controlled by the Arduino experimentation board. Notifications are sent to a connected mobile device via a mobile app, depending on reading ranges. The process addresses stipulated points as follows:

Section 1 focuses on the overall development of the project and the proposal of objectives for it, both general and specific.

Section 2 describes the methodology applied in accordance with the stated objectives.

Section 3 presents the selection of materials for the development of the project, both software and hardware. In addition, it applies the realization of the electronic circuit applied in the prototype and the implementation of all the elements to make up the alcohol measurement system.

Sections 4 and 5 present the development of the conclusions of the prototype created and they also present the appropriate recommendations in case of taking the project as a reference.

Finally, in sections 6 and 7, the bibliographic references and the related annexes are added, which serve as the basis and support for the development of the prototype.

KEYWORDS: *alcohol measurement, bluetooth, Arduino, mobile app, reading ranges.*

1 DESCRIPCIÓN DEL COMPONENTE DESARROLLADO

El siguiente prototipo fue diseñado con la finalidad de presentar un dispositivo móvil capaz de realizar mediciones de alcohol y enviar alertas conforme la situación lo amerite. El dispositivo de medición habilita la medición de alcohol por medio del aliento de las personas, esto tiene un enfoque para los conductores profesionales y no profesionales. En el caso de que las mediciones registradas por el sistema detecten cierto valor límite, el dispositivo emitirá una lectura de la medición a un dispositivo móvil y junto a esta se presenta de manera visual y sonora, la sanción respectiva; las sanciones y valores límites están establecidos dentro del Código Integral Penal (COIP), con vigencia en el Ecuador.

Para el funcionamiento del sistema, el sensor MQ-3 recolecta información sobre la presencia de alcohol en el ambiente; para esto, la persona sujeta a la medición deberá exhalar directamente sobre el sensor. Después de esta acción, el sensor envía los datos mediante el pin A0 a la placa Arduino Nano, la que previamente está configurada mediante un código desarrollado en Arduino IDE. La información se procesa en el Arduino y se envía de manera inmediata al módulo de comunicación inalámbrica (HC-05). Seguido a esto, el módulo *bluetooth* que se encuentra conectado a los puertos seriales del Arduino, transmite la información a un dispositivo móvil por medio de una aplicación; aquí se puede visualizar la información de la medición realizada y para aplicar un proceso transparente ante el sujeto de medición, se genera una alerta sonora que dicta el contenido mostrado en pantalla.

La aplicación móvil es aplicada en sistemas Android y consta de una interfaz sencilla e intuitiva; únicamente el usuario debe emparejar los dispositivos por medio de *bluetooth*, eso se realiza por medio de la selección de dispositivos propia de la aplicación. De ahí en adelante el dispositivo de medición envía información de manera inalámbrica. Para proporcionar autonomía al sistema, está alimentado por medio de una batería LiPo de 7.4 (V), dotando al sistema de una funcionalidad continua de dos horas y media; este factor es el que aporta la cualidad de portabilidad al prototipo. Todos los componentes están encapsulados en una *case* cuya función es la de proteger a los elementos que conforman el sistema de medición; además, en este *case* se incorporan medios de accesibilidad a los modos del prototipo (profesional y no profesional) y al control de suministro de energía, por medio de interruptores montados en la sección frontal, al igual que el sensor de alcohol.

1.1 Objetivo general

Implementar un prototipo de sistema de alerta por detección de alcohol.

1.2 Objetivos específicos

- Establecer los requerimientos del prototipo.
- Definir los componentes de hardware y software.
- Diseñar el prototipo.
- Implementar el prototipo.
- Realizar pruebas de funcionamiento.

1.3 Alcance

En lo que respecta al estudio de los requerimientos, se abarca un análisis dentro de un sector en específico que es la seguridad vial, esto para el desarrollo enfocado al prototipo. El sistema propuesto radica desde la fuente de poder del dispositivo (alimentación eléctrica), la protección por parte de un *case* adaptado para el dispositivo y el método de comunicación inalámbrica aplicado. Los principales elementos que forman parte del prototipo son el sensor de captación de datos, el procesador de información y el módulo de comunicación. El estudio realizado es tomado en base para la selección de los elementos mencionados disponibles en el mercado.

La creación del diseño abarca el modelado 3D del dispositivo final, realizado mediante un programa enfocado a la modelación de objetos en tercera dimensión; esto permite generar un concepto visual del prototipo con la finalidad de mantener un esquema adecuado para su uso. Con respecto a la autonomía que sugiere el dispositivo, está justificada en base al cálculo y al consumo de cada uno de los elementos en función de la batería utilizada para la alimentación del circuito. El diagrama de flujo muestra el algoritmo a seguir que tendrá el código del programa, tanto el diagrama de flujo como el programa fueron desarrollados en aplicaciones especializadas.

La creación del prototipo considera la implementación de un circuito electrónico, la realización de un modelo en un programa adecuado, la aplicación del modelo de circuito en una placa de cobre y su proceso hasta dar con la placa electrónica sobre la que están montados los elementos del sistema. También, en base al dimensionamiento de los elementos, un *case* fue fabricado; de esta manera se provee de protección al sistema de medición y a su alimentación. El *case* está adecuado para la fácil manipulación del

usuario. Se aplicó un sensor de alcohol para captar los datos en el ambiente y transmitirlos a la placa de desarrollo para que sean procesados de modo adecuado. Las pruebas efectuadas son aplicadas dentro de un ambiente controlado en el cual no existen factores externos que pudieran afectar a la medición. Con esto, la información enviada a un dispositivo móvil (en el que se pueden apreciar las lecturas y alertas por medio de una aplicación), es eficaz y el usuario está en la capacidad de tomar las acciones correspondientes en caso de existir alguna infracción a los rangos establecidos según la ley vigente en el Ecuador. Finalmente, para la mejor comprensión del prototipo, se grabaron dos videos correspondientes al funcionamiento y al modo de uso del dispositivo a manera de manual.

1.4 Marco Teórico

Introducción

El consumo de alcohol es uno de los principales factores en los siniestros viales, esto se debe a que el consumo de bebidas alcohólicas deteriora paulatinamente las capacidades de conducción de las personas, provocando así que los conductores vean afectadas sus condiciones de visión, movilidad y discernimiento [1].

Un conductor al presentar estado étílico con una cantidad de 0.02 a 0.05 (g/dl) tiene una susceptibilidad tres veces mayor a la normal de sufrir un atentado en vías; a medida que el consumo aumenta, el riesgo también lo hace, como se observa en la Figura 1.1 [1].

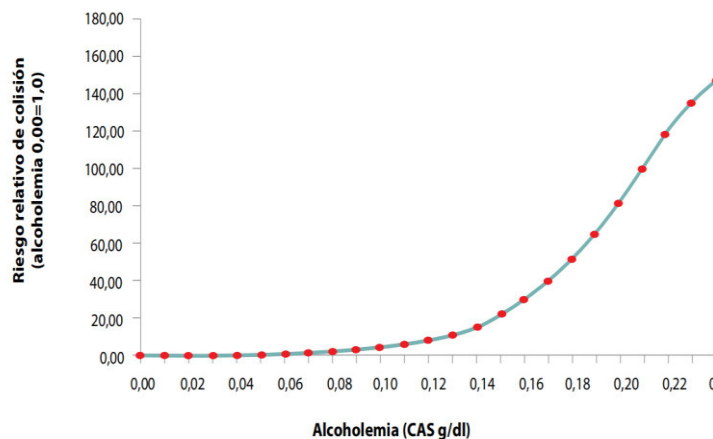


Figura 1.1 Relación de riesgo relativo de colisión referente a los gramos de alcohol por decilitro [1]

Bluetooth

La comunicación *bluetooth* está definida como un estándar que recurre a la comunicación y transmisión entre dispositivos con la principal intención de suprimir la

conexión cableada entre dispositivos móviles y fijos ubicados dentro de una red pequeña. La comunicación entre equipos *bluetooth* se radica en un margen de 10 (m) y su velocidad varía según la tecnología que se use; la banda de frecuencia que usa la tecnología *bluetooth* está en el rango de la banda abierta de 2.4 (GHz) y en la actualidad son dispositivos aplicados para equipos móviles debido al bajo costo de energía [2].

Placa Arduino Nano

Para realizar el control y procesamiento de datos captados por parte de los sensores del alcoholímetro, hay que proporcionar un sistema adecuado para dicho análisis, aquí es donde entra Arduino, al ser una placa electrónica que ofrece propiedades de hardware y software libre. Por contar con un desarrollo con un largo periodo de existencia, propone confiabilidad al usuario, además de bajos costos de implementación, debido a su larga estancia en el mercado y a la actual competencia; sin embargo, no deja de ser la alternativa más fiable al momento de realizar operaciones de control y sinterización de datos [3].

Dentro de la gama de Arduino, es posible hallar modelos que se diferencian a primera vista por su tamaño, siendo el más pequeño el Arduino Nano; a pesar de ocupar espacio físico reducido, este dispositivo adecúa una capacidad satisfactoria por parte de su microcontrolador (Figura 1.2), el ATmega328P. Además, propone un menor consumo en lo que a energía se refiere; por lo tanto, es un medio preciso para la realización de sistemas con fines portables [3].

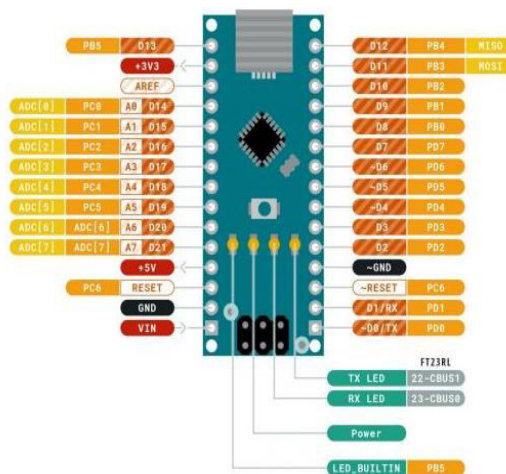


Figura 1.2 Distribución de pines en la placa Arduino Nano [4]

Sensor de gases MQ-3

Dentro del mundo de Arduino, existen varios complementos para la placa; uno de ellos es el sensor de gas MQ-3, como se muestra en la Figura 1.3. Este administra su

funcionamiento mediante la aplicación de transductores capaces de determinar la relación existente entre una resistencia de óxido dotado de características semiconductoras y una reacción química producida por el gas detectado [5].

La reacción producida provoca que una capa de la resistencia mencionada tenga una acción de liberar o absorber electrones, provocando así un cambio en la resistividad del elemento y por lo mismo el registro dinámico de la interrelación entre el gas recibido y el propio sensor [5].

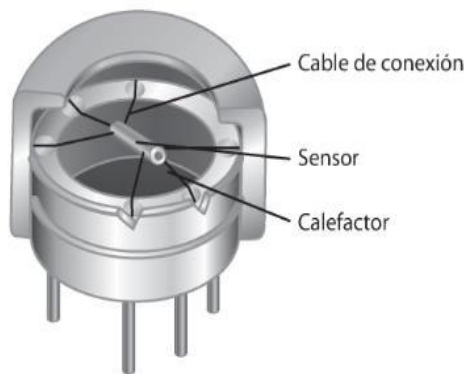


Figura 1.3 Elementos del sensor Arduino MQ-3 [4]

Módulo *bluetooth* HC-05

La implementación de dispositivos adicionales dentro de la placa Arduino es fundamental para iniciar el registro de datos externos; sin embargo, también existen complementos capaces de vincular la información analizada con otros artefactos electrónicos como teléfonos celulares, computadores, o dispositivos sin interfaz gráfica con la finalidad de almacenamiento de datos o simplemente visualización. Uno de los aditamentos ya mencionados es el módulo *bluetooth*, este se caracteriza por brindar generalmente una conexión personal; es decir, es capaz de crear una Red de Área Personal (PAN) [6].

Mediante *bluetooth*, el usuario puede implementar sistemas de control y monitoreo de los datos registrados y procesados por la placa en conjunto del sensor adecuado. De modo común, se usa este tipo de sistemas debido al bajo costo que presentan en relación con otros. Además, a pesar de ser un dispositivo que tiene un alcance de pocos metros, puede ser adecuado de modo escalable, permitiendo vincular cierto número de dispositivos sin tener que realizar un cambio radical en el núcleo de operación [6].

El módulo *bluetooth* HC-05, aparte de poseer dimensiones compactas, como se aprecia en la Figura 1.4, al ser compatible con Arduino es una de las mejores opciones de aplicación cuando se trate de transferencia de datos dentro de un área cercana. Dicho

elemento cuenta con la factibilidad de operar con un Protocolo de Puerto Serial (SPP), creando un intercambio de datos de forma transparente [6].

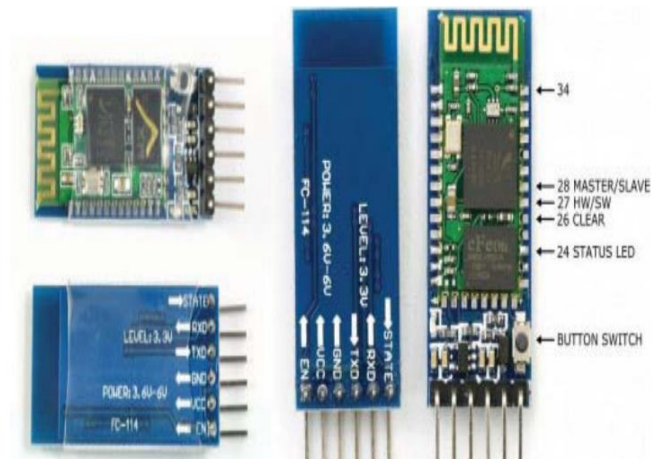


Figura 1.4 Módulo Bluetooth HC-05 y su distribución de pines [7]

2 METODOLOGÍA

El desarrollo del proyecto contempla el cumplimiento de cinco objetivos esenciales, que permiten la implementación de un prototipo de detección de alcohol en base a comunicación inalámbrica. Dichos objetivos permitieron manejar de manera adecuada tanto los procedimientos de medición como la aplicación de los dispositivos propuestos y se encuentran relacionados como se aprecia en la Figura 2.1.

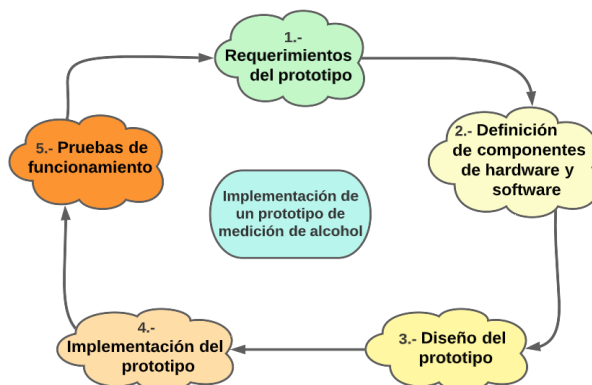


Figura 2.1 Metodología implementada

El proyecto tomó forma en base a la portabilidad, ya que se precauteló el ambiente en el cual se pondrá en funcionamiento; al tratarse de un dispositivo que tendrá uso en lugares exteriores, fue necesario gestionar sus parámetros para darle la identidad de

dispositivo portable y no dependiente de la red eléctrica. La aplicación de la conectividad *bluetooth* se dio debido a que las mediciones deben mostrarse de manera inmediata en un dispositivo de control. La conexión por medio de *bluetooth* entre dos dispositivos era la adecuada ya que esta proporcionaba una mejor respuesta en situaciones donde el usuario del medidor debe tener un elemento de visualización a la mano.

En lo que respecta a los componentes, fueron elegidos según la necesidad. Para la medición de los gases en el ambiente, el sensor MQ-3 fue más que adecuado dada su condición de sensor de alcohol. Para la selección del microcontrolador, se tomó en consideración el tamaño y la funcionalidad de un par de modelos de Arduino, el Uno y el Nano; haciendo una comparación entre estos y escogiendo el que menor tamaño presenta ya que ambos compartían similitud en funcionamiento. Hablando del módulo *bluetooth*, se entabló la recopilación de información entre el módulo HC-05 y HC-06 para escoger así al que convenía usar en el proyecto.

La creación del sistema de conexión en la placa electrónica se realizó con el software Proteus. Dado a que la simulación de los componentes usaba medidas reales, el posicionamiento de los objetos en la placa electrónica se realizó sin inconvenientes. Previo al desarrollo del código fuente usado para el Arduino, se propuso un diagrama de flujo, cuya finalidad fue la de guiar el funcionamiento que tiene el sistema. Los mensajes de alerta se generan en base a la condición que propone el sistema legal del Ecuador; la aplicación directa de los rangos fue establecida en base a los artículos plasmados en el Código Integral Penal. Las medidas y alertas tuvieron especial modo de visualización debido a que el sistema se sincronizó con un dispositivo móvil por medio de una aplicación realizada en App Inventor.

Al momento de realizar la placa electrónica, Proteus con su componente Ares, dieron como resultado un circuito electrónico en base a la conexión y posición previamente establecidas entre los componentes. Este diseño fue trasladado a una baquelita, la cual fue procesada y perforada para la posterior soldadura de espadines, sobre los cuales se montaron los componentes del sistema de medición.

Al finalizar con el montaje, se modeló el sistema de manera física aplicando la ayuda de SketchUp, para facultar el resultado final del proyecto de manera visual. Una vez realizado todo el sistema, se ubicó dentro de un *case* fabricado de materiales acrílicos que garantizan la protección del sistema.

Teniendo todo el sistema a punto, se aplicaron evaluaciones de funcionamiento, aquí se aplicó una prueba para cada uno de los modos y rangos propuestos dentro del código

de programación. Las mediciones fueron reveladas a través de la aplicación móvil y de esta manera se evaluó el funcionamiento en conjunto de los elementos montados en el *case* de acrílico; cada una de las condiciones fue mostrada en la pantalla del dispositivo móvil a manera de texto y de audio.

3 RESULTADOS

El prototipo aplicado para el sistema de medición de alcohol, ayuda al usuario a recibir mediciones directamente en un dispositivo móvil que cuente con la aplicación necesaria ya instalada. Las mediciones se darán de manera periódica y mostrarán en pantalla los valores provenientes del sensor de alcohol; en caso de que la medición se encuentre dentro de un rango establecido, el sistema efectuará una alerta tanto visual como sonora. Para la medición del nivel de alcohol, el sensor MQ-3 debe estar a una distancia no mayor a 15 (cm) como recomendación del sujeto a medición. Esta información irá directa al Arduino Nano con un código de programación previamente cargado para poder analizar los datos y enviarlos de manera directa por los puertos seriales en los cuales está conectado el módulo *bluetooth* HC-05, mismo que se encargará de transmitir la información de lectura de manera inalámbrica hasta el dispositivo móvil conectado, en el que se mostrará la información correspondiente.

El sistema de alerta se dará en función del modo en el cual se encuentre activo el prototipo; esta alerta se dará cuando las mediciones enviadas se encuentren dentro de los rangos asignados. Los modos y el encendido del dispositivo se adecúan por medio de interruptores ubicados en la sección frontal del *case*. El proyecto se aplica considerando cinco puntos cruciales: primero, los requerimientos para la implementación del sistema; después, definir los componentes tanto en software como hardware; como tercer y cuarto punto se implementa el diseño y aplicación que conlleva el prototipo; finalmente, luego de haber concluido las fases previstas, el sistema de medición será sujeto a pruebas de funcionamiento.

3.1 Requerimientos del prototipo

La implementación del prototipo de medición de alcohol usando una placa Arduino, ayuda al usuario a realizar dichas mediciones de manera sencilla y portátil; de este modo se garantiza una mayor movilidad en relación con los dispositivos convencionales. Además, al proponer una conectividad dentro de una PAN por medio del módulo *bluetooth*, aplica de mejor manera la vinculación a otros dispositivos cercanos (distancia máxima de 5 (m)) en los cuales se podrá realizar la visualización de los datos analizados.

Al momento que se realice la medición de alcohol a un tercero, los datos serán procesados y enviados al dispositivo vinculado mediante *bluetooth*; de este modo, la medición realizada será transparente tanto para el usuario como para la persona a la que se le haga la medición.

El dispositivo en cuestión será el encargado de notificar al usuario los resultados obtenidos empleando frases y alertas que avisen del estado etílico de la persona a quien se realizó la prueba. La conectividad inalámbrica estará dentro de la banda de 2,4 (GHz), por lo que una gran cantidad de dispositivos móviles como ordenadores, tabletas u otros, serán compatibles con el sistema, estableciendo de esta manera una cualidad de compatibilidad tecnológica muy alta.

El prototipo planteado va dedicado a la medición de alcohol en entornos viales; es decir, será aplicado para detectar de manera efectiva el nivel de alcohol que presenten los conductores, a través de su aliento, para que de este modo las autoridades encargadas tomen las respectivas acciones. Debido a que este dispositivo estará aplicado en ambientes exteriores, tendrá un *case* adecuado para fomentar la durabilidad de este y que así se faculte una vida útil más larga.

El *case* estará aplicado para mantener protegidos a todos los componentes, teniendo principal enfoque en el sensor de gas, ya que sin este no se podrá efectuar la medición. El diseño de la carcasa propone un modelo compacto y ergonómico para que sea cómodo para el usuario, además de tener pequeñas leyendas en la misma para indicar el modo de uso del dispositivo; así cualquier persona estará capacitada para usarlo en casos fortuitos.

La creación de este contenedor estará realizada en una impresora 3D, por lo que empleará filamentos de plástico en base a polibutadieno, así la protección será mayor ya que se aplica resistencia a golpes [8].

Tomando en cuenta que será un aparato dedicado a alerta y movilidad, no será dependiente de una fuente de energía grande o peor aún una conexión cableada a una toma de corriente alterna. La opción adecuada para el funcionamiento es la implementación de una batería recargable y extraíble, con esto el dispositivo asumirá una cualidad de autónomo y será usado sin ninguna complicación en cualquier situación. El uso de una batería de 5 (V) a 12 (V) es suficiente para que el Arduino y el resto de los componentes funcionen apropiadamente.


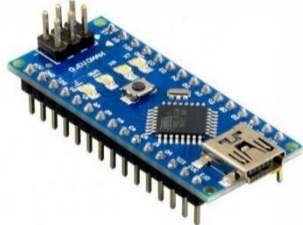
3.2 Definición de componentes de hardware y software

Hardware

La adecuación de la placa de implementación de Arduino se basa fortuitamente en la comparación existente entre algunas de sus versiones, siendo estas: Nano y Uno. El uso de este sistema de micro procesamiento destaca por su sencilla aplicación en el ámbito de la programación (con Arduino IDE) ya que únicamente basta acoplar un código fuente al dispositivo para que entre en funcionamiento [9].

La diferencia entre las placas de desarrollo Arduino es visible ya que su tamaño es una de las principales características a tomar en cuenta; la comparativa radica entre el Arduino Uno y el seleccionado para la implementación, el Nano. Debido a que el Arduino Uno tiene un tamaño medio entre sus tres presentaciones habituales, se ha optado por el Arduino Nano, aportando con el diseño ergonómico y compacto que se definió inicialmente. Las diferencias entre las placas se aprecian en la Tabla 3.1.

Tabla 3.1 Comparativa entre Arduino Uno y Nano [10]

Parámetro	Arduino Uno	Arduino Nano
Voltaje de ingreso	7-12 (V)	7-12 (V)
Microcontrolador	ATmega328P	ATmega328
Puertos Digitales y PWM	20	22
Puertos Analógicos	6	8
Dimensiones	8x5.51 (cm)	1.8x4.5 (cm)
Dispositivo		

Una vez seleccionada la placa de desarrollo, hay que enfocarse en el tipo de sensor aplicado para la medición de alcohol, en este caso el MQ-3, sensor al reaccionar de manera química con ciertos gases como alcoholes, bencina y otros como metano e incluso combustibles como GLP, aunque en poca concentración. La elección de este

complemento se debe al rango de acción que tiene a la detección de alcohol en relación con otros de la familia MQ; GLP y el CO para los módulos MQ-2 y MQ-5, respectivamente [11]. Como se puede observar en la Figura 3.1, la reacción del alcohol con el sensor MQ-3 es la más adecuada en comparación con sus similares.

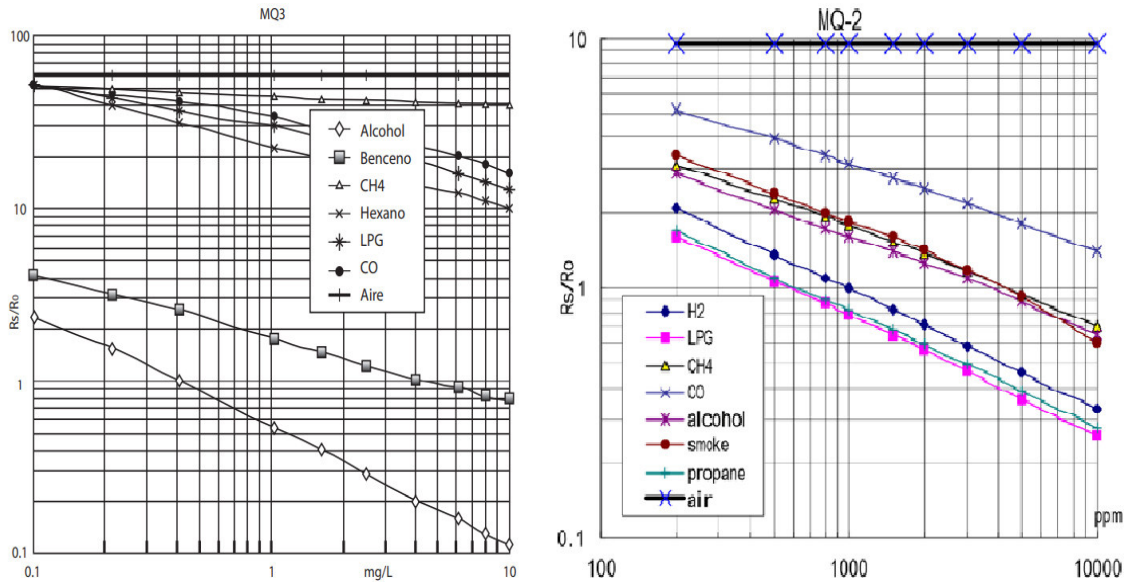


Figura 3.1 Comparación de reacción entre el sensor MQ-2 y MQ-3 [5] [12]

La comunicación inalámbrica está impartida por *bluetooth*; como se trata de un dispositivo enfocado al uso móvil, el dispositivo encargado de la comunicación debe poseer una característica de funcionamiento con bajos recursos energéticos dentro de un espacio de cerca de 5 (m) a la redonda. Por este motivo, la tecnología más adecuada para el presente proyecto en la transmisión de datos es *bluetooth*. Dentro de la familia de módulos HC, se encuentran el HC-05 y el HC-06, a pesar de que físicamente presentan una diferencia entre los pines de conexión, siendo 6 el número mayor de pines por parte del HC-05. También existe una consideración en la aplicación, el módulo HC-06 únicamente cuenta con la capacidad de recibir información; es decir, tiene la funcionalidad de esclavo. El módulo HC-05 adecúa sus funcionalidades de modo que puede gestionar la comunicación como receptor (esclavo) y como emisor (maestro). Por el hecho de que el HC-05 proporciona mayor transferencia de información con otros dispositivos, es el adecuado para enviar mensajes de alerta en función de las órdenes que la placa Arduino aplique. El reducido tamaño, banda de trabajo y mínimo consumo de energía, lo hacen el candidato ideal para la creación de un elemento móvil; las características están detalladas en la Tabla 3.2.

Tabla 3.2 Características del módulo bluetooth HC-05 [6] [13]

Características	HC-05	HC-05
Versión de <i>bluetooth</i>	2.0+EDR	2.0+EDR
Modulación	3 (Mbps)	3 (Mbps)
Banda de trabajo	2.4 (GHz)	2.4 (GHz)
Voltaje de operación	1.8 – 3.6 (V)	3.1 – 4.2 (V)
Modos	Maestro - Esclavo	Esclavo

Software

App Inventor

El uso de la aplicación en línea “App Inventor” se justifica por la fácil gestión de contenido y programación por parte del usuario. Al manejar una interfaz gráfica para la selección de componentes, la aplicación es muy intuitiva y logra una gran conexión para el manejo de la fachada del programa creado; por parte de la programación, se debe tener noción de sistemas básicos y funcionamiento de algunas funciones. App Inventor propone un sistema de programación de bloques que maneja de manera precisa el control de componentes adecuados en la pantalla de fachada. La propuesta del sistema de bloques es encasillar a manera de rompecabezas las funciones que realizará la aplicación en cuestión, sin presentar la complicación que suele poseer el lenguaje de alto o bajo nivel. Otro aporte de esta aplicación en línea es la creación de aplicaciones para dispositivos móviles de modo sencillo, únicamente se necesita aplicar en la sección de importación para que un archivo con extensión “apk” pueda ser generado.

A diferencia de otros programas de creación de aplicaciones como Andromo, App Inventor muestra una interfaz mejor organizada e intuitiva; por lo tanto, tendrá un mejor aprovechamiento ya que el usuario es capaz de crear aplicaciones de manera rápida, sin tener muchas complicaciones o realizar pagos extra. Además, cuenta con el respaldo de una institución de renombre como es el MIT.

Este programa cuenta con la facilidad de adecuar ambientes como el de *bluetooth* para manejar la conexión entre el dispositivo móvil al cual se instaló la aplicación creada. Añadiendo la facultad de cliente *bluetooth* al sistema que ejecute la aplicación, el usuario maneja el comportamiento del dispositivo, que para este caso se empleará a manera de

receptor; los datos recibidos estarán almacenados en variables que puedan ser tratadas por las demás acciones que se proponen.

Arduino IDE

La selección de Arduino IDE se debe específicamente a que este programa facilita la edición de códigos fuente directamente para Arduino; el panel principal incluye las funciones principales que serán ejecutadas por la placa Arduino, “setup” y “void”. Ya que es una aplicación propia de Arduino, el programa ofrece mucha flexibilidad con los modelos de placas, facilitando de esta manera el uso del Arduino Nano.

Los códigos pueden ser verificados antes de crear una compilación de estos; de este modo, se simplifica la localización de errores en el código en caso de que los hubiese. Además, al momento de subir un código a la placa Arduino, lo único que se debe hacer es conectar la placa, seleccionar el modelo en uso y presionar la opción “Subir”. Por la sencillez y compatibilidad con los dispositivos Arduino, Arduino IDE se faculta como la mejor elección al momento de crear un código fuente para estos casos.

3.3 Diseño del prototipo

Circuito electrónico

Para iniciar el diseño del circuito electrónico, se emplea el programa de diseño de circuitos Proteus; este simulador permite realizar la implementación de dispositivos por medio de representaciones de los dispositivos electrónicos. En la interfaz principal se pueden montar los materiales necesarios para verificar de manera virtual el espacio físico de cada uno de los componentes aplicados, como se observa en la Figura 3.2. Además, dicho programa ayuda a facultar el diseño de una placa de circuito impreso; una de sus opciones muestra de manera gráfica el diseño que tendrá el circuito impreso finalizado, aquí se puede manipular de manera eficiente el camino que tomarán las pistas de cobre en el diseño planteado.

Las vías de cobre están representadas por líneas que conectan a los diversos componentes del circuito; para este caso, la placa Arduino, el módulo *bluetooth*, el sensor MQ-3 y la respectiva sección de alimentación representada por conectores o bornes.

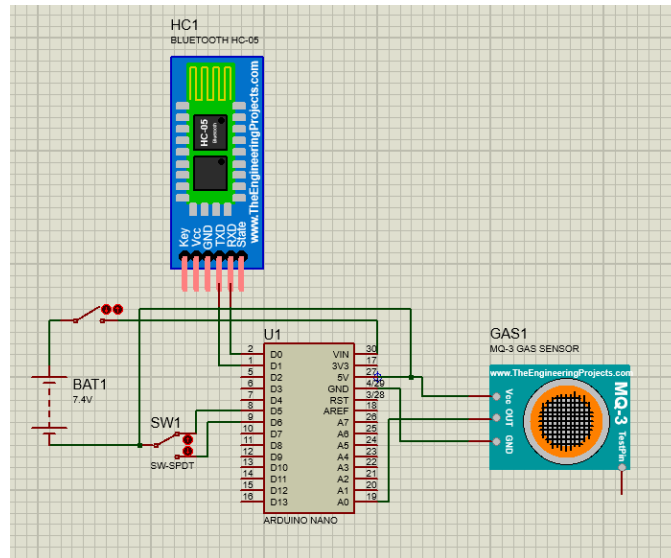


Figura 3.2 Diseño del circuito impreso en el programa Proteus

Consumo energético

Para el funcionamiento adecuado de los equipos, la fuente de voltaje debe entregar 5 (V) como voltaje de ingreso a la placa Arduino, este voltaje debe alimentar al sensor MQ-3 y al módulo *bluetooth*. Los elementos presentarán un consumo entre 3.3 (V) hasta 5 (V), estos valores se muestran individualmente en la Tabla 3.3.

Tabla 3.3 Voltajes y corrientes de operación para los principales elementos del sistema de medición de alcohol [13] [14] [15]

	Arduino Nano	Sensor MQ-3	Módulo HC-05
Voltaje de operación	5 (V)	5 (V)	3.3 (V) – 5 (V)
Voltaje de alimentación	7 (V) – 12 (V)	5 (V)	5 (V)
Corriente de operación	50 (mA)	150 (mA)	50 (mA)

Cada una de las partes del medidor de alcohol estará alimentada en su totalidad por medio de una batería de 7.4 (V) LiPo de 500 (mAh). Tomando en cuenta la capacidad que entrega la fuente de voltaje, se presenta el cálculo de consumo y el tiempo estimado de utilización del dispositivo por cada carga.

Primero, se realiza la sumatoria de las corrientes requeridas por el conjunto de elementos del sistema en general, como se ve en la Ecuación 3.1.

$$\text{Corriente consumida} = I_{\text{Arduino}} + I_{\text{Sensor}} + I_{\text{Bluetooth}}$$

Ecuación 3.1 Suma de corrientes de consumo en cada elemento

Donde:

I_{Arduino} = Corriente de consumo del Arduino

I_{Sensor} = Corriente de consumo del sensor MQ-3

$I_{\text{Bluetooth}}$ = Corriente de consumo del módulo *bluetooth* HC-05

Entonces, al establecer la corriente de consumo de cada elemento, la corriente total queda de la siguiente manera:

$$\text{Corriente consumida} = (50 + 150 + 50)(mA)$$

$$\text{Corriente consumida} = 200 (mA)$$

La corriente necesaria para el funcionamiento del medidor de alcohol representa un total de 200 (mA); la batería usada generará una autonomía de entre 2 horas a 2 horas y 30 minutos sin dificultad, así se explica en la Ecuación 3.2.

$$T_{\text{Autonomía}} = \frac{I \text{ de batería}}{I \text{ consumida}}$$

Ecuación 3.2 Tiempo calculado para la autonomía del dispositivo

Donde:

$T_{\text{Autonomía}}$ = Tiempo de autonomía del dispositivo en horas

$I \text{ de batería}$ = Corriente que suministra la batería

$I \text{ consumida}$ = Corriente de consumo total del sistema

El tiempo de autonomía que proporciona la batería está declarado en horas, con esto se establece un tiempo determinado para la utilización del dispositivo, el cual se muestra a continuación:

$$T_{\text{Autonomía}} = \frac{500 (mAh)}{200 (mA)}$$

$$T_{\text{Autonomía}} = 2.5 (h)$$

Aplicando una regla de tres, se puede determinar la cantidad de minutos, tomando la sección decimal del Tiempo de autonomía representado por $T_{Autonomía}$. Esto se aprecia en la Ecuación 3.3.

$$M_{Autonomía} = \frac{0.5 (h) \times 60 (min)}{1 (h)} = 30 (min)$$

Ecuación 3.3 Transformación de tiempo en horas a minutos

Donde:

$M_{Autonomía}$ = Minutos de autonomía que representa la sección decimal de $T_{Autonomía}$

El tiempo total resultante será la cantidad entera de $T_{Autonomía}$ en representación de las horas y $M_{Autonomía}$ para los minutos; es decir, el sistema garantiza funcionamiento durante 2 horas y 30 minutos de manera ininterrumpida.

Así, se asegura que el consumo de la batería satisface las condiciones de uso regulares, ya que es tiempo más que suficiente para la utilización del artefacto, debido a que de manera general los controles de tránsito no duran más de dos horas.

Rangos de acción

Los límites en los cuales las alertas son enviadas a la aplicación móvil, tienen base en las condiciones aplicadas por el Código Integral Penal (COIP), vigente en Ecuador y con última modificación el 17 de febrero del 2021 [16]. En función del tipo de conductor, los rangos irán variando; en el caso de que sea un conductor profesional, se aplicarán los rangos establecidos en la Tabla 3.4, aquí se aplica una sanción para las personas que superen los 0.1 grados alcohólicos. Esto se debe a que el conducir en estado etílico, aumenta la probabilidad de sufrir accidentes y en el caso de los transportistas profesionales estos accidentes pueden involucrar más de una vida.

Tabla 3.4 Sanciones para transportistas profesionales según el COIP [16]

Sanciones para transportistas profesionales	
Grados / Litro de sangre	Sanción
0.1	Privación de libertad por 90 días, pérdida total de los puntos de licencia.

Para los conductores no profesionales, la situación es menos severa; esta clase de usuarios tendrán sanciones a partir de los 0.3 grados de alcohol. Si el conductor supera con creces el límite inicial; es decir, 0.3 grados alcohólicos, la sanción seguirá aumentando en función del estado etílico del individuo; esto se muestra en la Tabla 3.5.

Tabla 3.5 Sanciones para conductores no profesionales según el COIP [16]

Sanciones para transportistas no profesionales	
Grados / Litro de sangre	Sanción
0.3 - 0.8	Pérdida de 5 puntos de licencia, privación de libertad por 5 días y pago de 1 salario básico unificado.
0.8 - 1.2	Pérdida de 10 puntos de licencia, privación de libertad por 15 días y pago de 2 salarios básicos unificados.
1.2 en adelante	60 días de licencia suspendida, privación de libertad por 30 días y pago de 3 salarios básicos unificados.

Los controles de alcoholemia se realizan por medio del aliento del conductor [17]; por este motivo, el sensor de alcohol registrará la concentración de grados alcohólicos y ejecutará una acción dependiendo del rango en el cual se haya gestado la medida. En cada rango, el microcontrolador será el encargado de enviar el rango en el que se encuentra el conductor y su respectiva sanción. Ya que existen dos clases de conductores, el dispositivo está adecuado con un modo de selección para realizar la medición en cada uno de los escenarios ya mencionados.

Diagrama de flujo

El código fuente diseñado para la validación de datos registrados por el sensor de alcohol MQ-3 está basado en un algoritmo que compara límites propuestos con los valores de ingreso. De esta manera, cada vez que el valor anotado en el Arduino se encuentre en las condiciones determinadas en el código, este se encargará de generar una serie de mensajes que servirán de información para el usuario. Cuando los datos registrados estén entre el valor límite 1 y el valor límite 2, el Arduino gestionará la impresión del mensaje "Está sobre el valor legal de 0.3 grados", seguido por la sanción correspondiente. Al momento de que el valor esté entre los rangos de límites 2 y 3, se enviará el siguiente texto "Está sobre el valor legal de 0.8 grados", igualmente seguido

de la respectiva sanción. Finalmente, cuando se supere el valor establecido como el límite 3, aparecerá lo siguiente: "Está sobre el valor legal de 1.2 grados", incluyendo la respectiva información legal para proceder con la sanción. Estos datos se enviarán al módulo *bluetooth* que se encargará de la comunicación inalámbrica con la aplicación móvil en la que se visualizarán los mensajes de Arduino. Entre cada medición hay que esperar al menos un segundo previo a la repetición del proceso. El algoritmo utilizado está en la Figura 3.3, presentada a continuación.

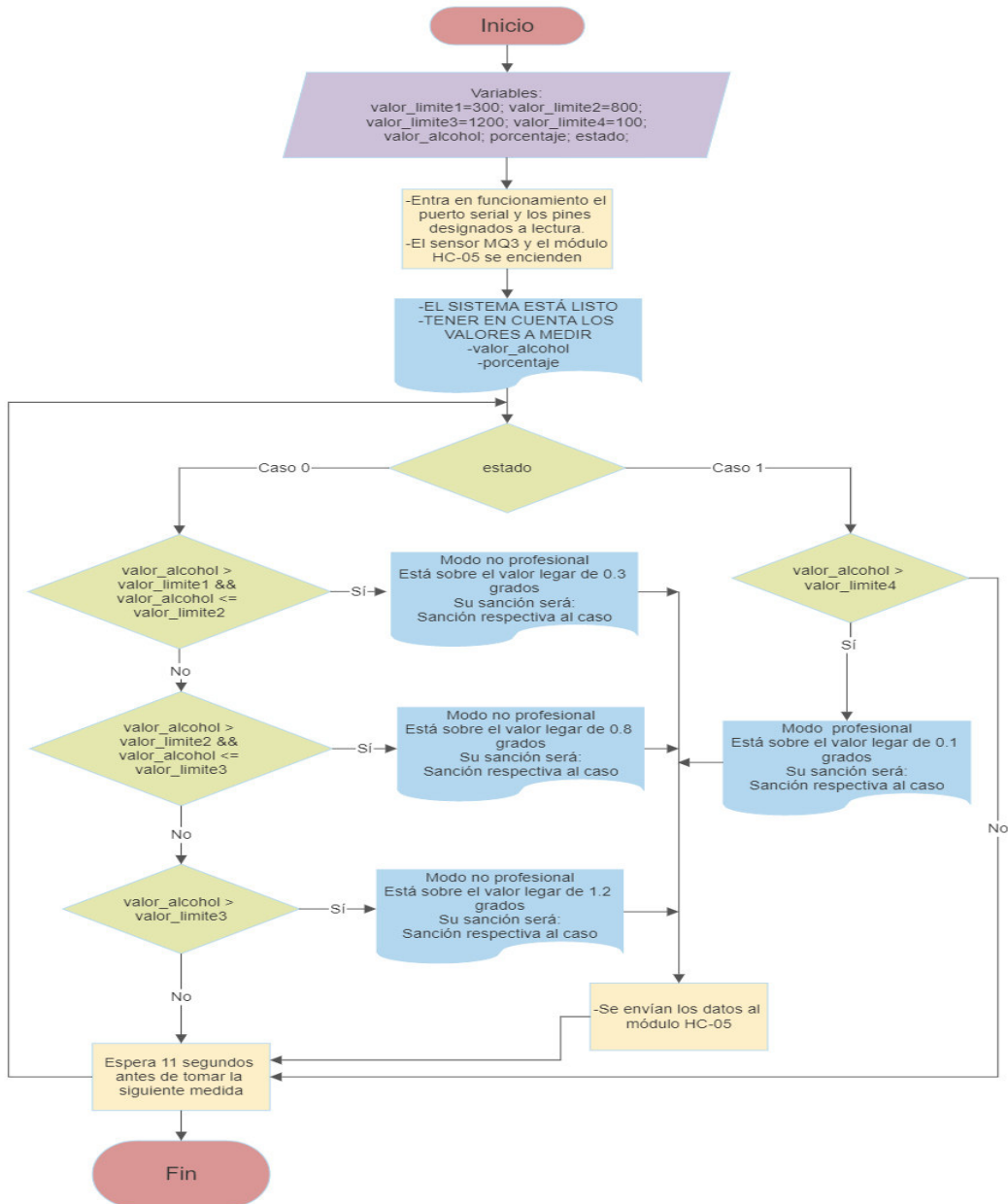


Figura 3.3 Diagrama de flujo del sistema de medición de alcohol

Modelo 3D

Para el modelamiento del sistema de medición se aplicó el programa Sketchup, el cual permite recrear elementos en tercera dimensión para apreciar de manera virtual las dimensiones que estos poseen. A pesar de contar con varios tipos de elementos modelados, se aplicó los correspondientes materiales electrónicos con el propósito de simular el resultado final del proyecto de manera física, esto incluye el montaje de la placa Arduino, el sensor de alcohol MQ-3, el módulo HC-05 y la fuente de alimentación, dentro de su respectivo case. El modelamiento del sistema se presenta en la Figura 3.4.

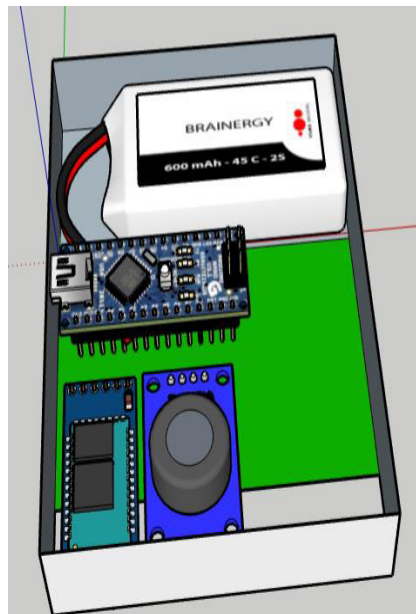


Figura 3.4 Dispositivo de medición de alcohol modelado en Sketchup

3.4 Implementación del prototipo

Elaboración de la placa de circuito impreso

Para la creación de la placa electrónica del sistema se aplica el diseño inicial en Proteus; dicho diseño constará de las conexiones entre los materiales que serán montados en la placa. El diseño del circuito impreso está proporcionado a medidas reales para cada material incorporado y está impreso a manera invertida (en vista de espejo), esto se hace con la finalidad de que, al momento de aplicar el diseño en la placa de cobre, las pistas se conecten de manera adecuada en el circuito. El diseño del circuito impreso está aplicado en la baquelita y se encuentra visible en la Figura 3.5.

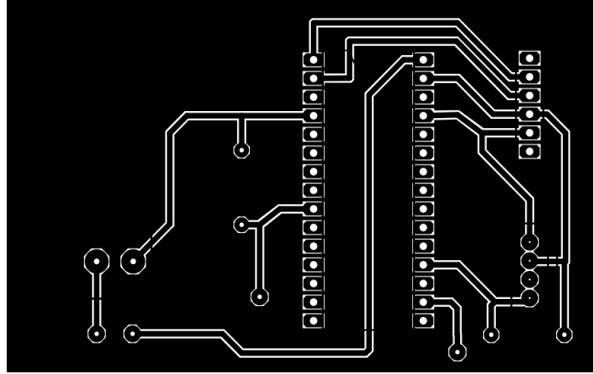


Figura 3.5 Diseño del circuito impreso

Este circuito impreso es transferido por medio de una impresión en papel cuché (Figura 3.6) y el método de planchado (Figura 3.7 y Figura 3.8) con la finalidad de que las zonas que conforman las pistas del circuito sean recubiertas y protegidas al momento de realizar la degradación del cobre por el uso de compuestos químicos.

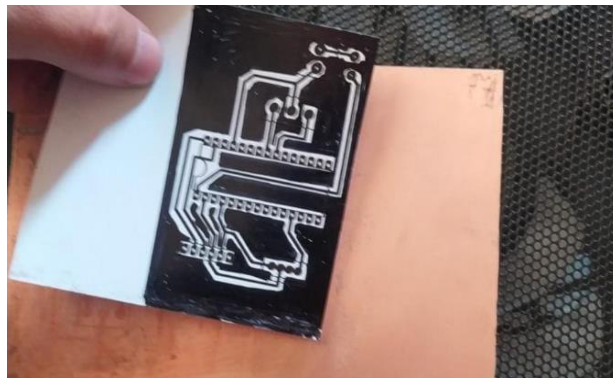


Figura 3.6 Impresión del circuito en papel cuché



Figura 3.7 Centrado de la impresión para la transferencia



Figura 3.8 Proceso de planchado sobre la baquelita

Cuando el circuito impreso haya sido transferido completamente a la baquelita, hay que empezar con la disolución del cobre; este proceso se realiza aplicando la placa recubierta con el circuito en una solución de agua y cloruro férrico (FeCl_3), esto se observa en la Figura 3.9. Una vez el proceso de degradado del cobre en la baquelita culmine, la placa queda únicamente con las secciones recubiertas, como se muestra en la Figura 3.10.



Figura 3.9 Aplicación de cloruro férrico



Figura 3.10 Resultado de la corrosión de cobre

Con el circuito impregnado directo en las pistas de cobre, hay que realizar agujeros para cada uno de los respectivos componentes; en este caso se aplicó espadines, cables y una bornera para que el desmontaje de los elementos de la placa sea viable en caso de averías o daños físicos. El proceso de soldadura está plasmado mediante la Figura 3.11 y en la Figura 3.12.

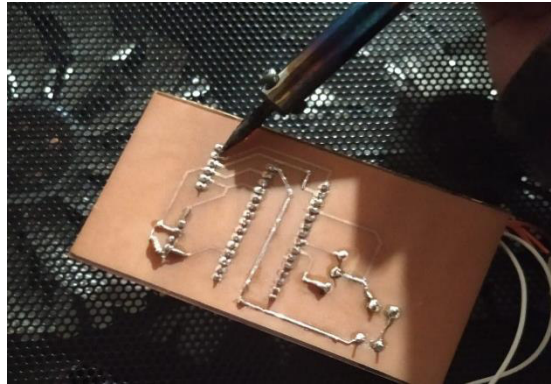


Figura 3.11 Proceso de soldadura

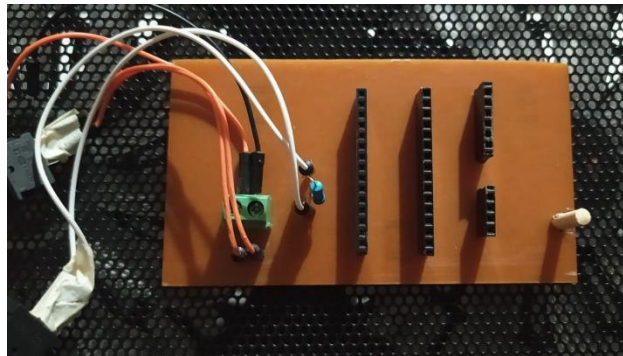


Figura 3.12 Espadines y componentes soldados en la placa

Ya teniendo los materiales soldados en la placa, se procede a ubicar el microcontrolador, el sensor de gases y el módulo *bluetooth*; la Figura 3.13 muestra la disposición de cada uno de los elementos mencionados.

La conexión de la placa Arduino será la central dado que tanto el módulo de conexión inalámbrica como el sensor de gases están dispuestos de manera en que tomen su alimentación del pin que ofrece 5 (V). Además, debido a que la conexión del sensor va al pin analógico A0 y el módulo HC-05 está vinculado a los puertos seriales, la disposición que esos elementos tienen dentro de la placa electrónica es la adecuada debido al fácil enrutamiento de conexión que se presenta.

La aplicación de interruptores tiene la finalidad de poder encender y apagar el sistema de medición a necesidad del usuario para tener una mejor gestión de energía y también para la adecuación de los modos de funcionamiento del prototipo; es decir, para conmutar entre el modo profesional y el no profesional.

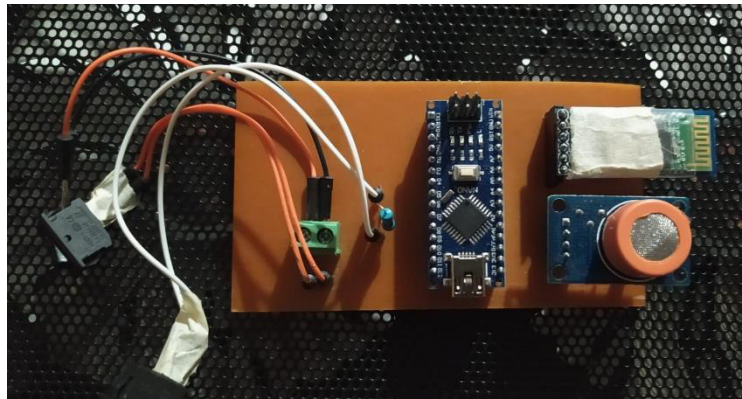


Figura 3.13 Montaje de componentes en la placa electrónica

Ya con los principales actores posicionados de manera correcta, es necesario aplicarlos dentro de su *case*; este *case* constituye en su totalidad de acrílico transparente para que el sistema pueda ser apreciado. Además, el diseño que se propone se adecúa a las dimensiones de un teléfono móvil, exceptuando por el grosor que, dado a la presencia de espadines en la placa, alcanza los 3 (cm); dicho *case* es apreciable en la Figura 3.14.



Figura 3.14 Case de acrílico para el sistema de medición de alcohol

Programación para el Arduino Nano

Para el funcionamiento del sistema, la placa Arduino Nano debe contener un código fuente compilado; para ello se propone el código fuente del ANEXO III. El código será el encargado de determinar el procesamiento de los datos a través de Arduino y su

vinculación con los dispositivos adyacentes; es decir, el módulo *bluetooth* HC-05 y el sensor MQ-3. El proceso de codificación se muestra a continuación.

En primera instancia, se fijan los valores limitados por el COIP, estableciéndolos a manera de variables enteras de tipo “int”, para determinar los valores de manera precisa. El nombre de estas variables será “valor_limite#”, donde “#” expresa el número iterativo a partir de 1 de la cantidad de valores asignados. De manera similar, se aplica la variable “valor_alcohol” que es de tipo “float” ya que representa valores decimales. También se tienen variables complementarias que son “estado” y “c”, de tipo “int” y “char”, respectivamente. La inclusión de estas variables es para gestionar el cambio de modo del sistema en el caso de “estado” y simplemente crear una línea de texto vacía para “c”. La declaración de variables se aprecia en la Figura 3.15.

```
int valor_limite1= 300; // Fija el valor limite en el que se activa la alarma
int valor_limite2= 800;
int valor_limite3= 1200;
int valor_limite4= 100;
float valor_alcohol;
int estado;
char c = ' ';
```

Figura 3.15 Declaración de variables globales

La función “*setup*” (Figura 3.16) se encarga de configurar los puertos y pines del Arduino; aquí únicamente se presenta la velocidad de transferencia estándar, es decir, 9600 (Baudios), los cuales están estipulados por la sentencia `Serial.begin`. También se configuran los pines 5 y 6, como entrada y salida, para controlar los modos de operación del sistema; cuando el pin 6 envía una señal de encendido o “*HIGH*”, el pin 5 recepta la señal de cambio de modo. Únicamente como adicional, se imprime por medio del comando `Serial.println` un mensaje de inicialización del sistema.

```
void setup() {
  //Mensaje inicial
  Serial.begin(9600);
  Serial.println("EL SISTEMA ESTA LISTO");
  Serial.println("TENER EN CUENTA LOS VALORES A MEDIR");
  //Establece el comportamiento de los pines
  pinMode(5, INPUT);
  pinMode(6, OUTPUT);
}
```

Figura 3.16 Función *setup* dentro del código

Dentro de la función “*loop*”, se establecen 3 secciones importantes. La primera (Figura 3.17) es el comportamiento de los datos, los pines designados 5 y 6 (el que siempre estará en estado “*HIGH*”) están encargados del cambio de modo de profesional a no

profesional, dependiendo de la lectura del pin 5 que se encargará de cambiar el valor de la variable “estado”. El pin A0, es el encargado de recopilar la información enviada desde el sensor de gases; esto se logra con la función `analogRead`, que se encarga de recoger lecturas en el puerto analógico asignado.

```
void loop() {
  //Comportamiento ingreso o salida de datos
  digitalWrite(6,HIGH);
  estado=digitalRead(5);
  valor_alcohol=analogRead(A0);
}
```

Figura 3.17 Comportamiento de los pines asignados

La segunda sección (Figura 3.18), está encargada de la impresión y tratamiento de los datos recolectados por el pin analógico. La impresión se da directamente por el puerto serial en el cual esté conectado el módulo HC-05, aquí se mostrarán las variables propuestas como “valor_alcohol” para saber el valor real que capta el sensor y con el cual se hará la comparación con los límites establecidos y el valor de porcentaje de alcohol por litro de sangre. Hay que tomar en cuenta que este valor se estima con la prueba de aliento, ya que es el método usado convencionalmente para realizar pruebas de alcoholemia en conductores. El valor mostrado del porcentaje de alcohol se determina con la Ecuación 3.4.

```
//Impresión de datos al puerto serial
Serial.println(valor_alcohol); // Envía al Serial el valor leído del Sensor MQ3
float porcentaje=(valor_alcohol/10000); //Calcula el porcentaje
Serial.println(porcentaje); // Envía al Serial el porcentaje
Serial.println(' ');
delay (5000);
```

Figura 3.18 Impresión de datos en el puerto serial

$$porcentaje = \frac{valor_alcohol}{10000}$$

Ecuación 3.4 Cálculo del porcentaje de alcohol por litro de sangre

En la tercera sección (Figura 3.19) se realiza la validación de los datos; al aplicar una sentencia “*switch*” en función de la condición de la variable “estado”, se ejecuta el cambio de modo. Para el modo no profesional, cuando el registro de “estado” es 0, considera los 3 primeros valores límite propuestos; es decir 300, 800 y 1200. Utilizando el condicional “*if*” se dan tres casos de acción; cuando el valor está entre 300 y menor

a 800 (Figura 3.19Figura 3.19), se aplica la impresión de texto "Está sobre el valor legal de 0.3 grados" seguido por la sanción adecuada; si el valor está entre 800 y menor a 1200 (Figura 3.20) se imprime "Está sobre el valor legal de 0.8 grados" y la sanción; el caso final es si el valor de lectura supera a 1200 (Figura 3.21), se ejecuta la impresión de "Está sobre el valor legal de 1.2 grados" seguido de la sanción respectiva.

```
switch(estado){
  //modo no profesional
  case 0:

  if(valor_alcohol > valor_limite1 && valor_alcohol <= valor_limite2 )
  { // Si la medida de gas metano es mayor de valor limite1

    Serial.println("Modo no profesional");
    delay(3000);
    Serial.println("Esta sobre el valor legal de 0.3 grados");
    Serial.println(' ');
    Serial.println("Su sancion sera:");
    Serial.println("Perdida de 5 puntos de licencia");
    Serial.println("Privacion de libertad por 5 dias");
    Serial.println("Pago de 1 salario basico unificado");
    Serial.println(' ');|
    delay(3000);
  }
}
```

Figura 3.19 Inicio de la función *switch* y primer caso del modo no profesional

```
if(valor_alcohol > valor_limite2 && valor_alcohol <= valor_limite3)
{ // Si la medida de gas metano es mayor de valor limite2

  Serial.println("Modo no profesional");
  delay(3000);
  Serial.println("Esta sobre el valor legal de 0.8 grados");
  Serial.println(' ');
  Serial.println("Su sancion sera:");
  Serial.println("Perdida de 10 puntos de licencia");
  Serial.println("Privacion de libertad por 15 dias");
  Serial.println("Pago de 2 salario basico unificado");
  Serial.println(' ');
  delay(3000);
}
```

Figura 3.20 Segundo caso del modo no profesional

```

if(valor_alcohol > valor_limite3)
{
    // Si la medida de gas metano es mayor de valor limite3

    Serial.println("Modo no profesional");
    delay(3000);
    Serial.println("Esta sobre el valor legal de 1.2 grados");
    Serial.println(' ');
    Serial.println("Su sancion sera:");
    Serial.println("Suspension de licencia por 60 dias");
    Serial.println("Privacion de libertad por 30 dias");
    Serial.println("Pago de 3 salario basico unificado");
    Serial.println(' ');
    delay(3000);
}

```

Figura 3.21 Tercer caso del modo no profesional

Para el modo profesional (Figura 3.22), es decir, cuando la lectura de la variable “estado” sea 1, se aplica un único método de sanción; esto se debe a que en el COIP solo existirá un valor límite para transportistas profesionales que es de 0.1 grados de alcohol por litro de sangre, este valor se representa con la variable “valor_límite4”.

Aquí únicamente se aplica condición si la lectura supera el valor establecido, haciendo uso de un condicional “//”, seguido se aplica la impresión al puerto serial de “Está sobre el valor legal de 0.1 grados” junto a su correspondiente sanción.

```

//modo profesional
case 1:
if(valor_alcohol > valor_limite4)
{
    // Si la medida de gas metano es mayor de valor limite4

    Serial.println("Modo profesional");
    delay(3000);
    Serial.println("Esta sobre el valor legal de 0.1 grados");
    Serial.println(' ');
    Serial.println("Su sancion sera:");
    Serial.println("Reducción de 30 puntos en la licencia");
    Serial.println("Privacion de libertad por 90 dias");
    Serial.println(' ');
    delay(3000);
}
break;

```

Figura 3.22 Código para el modo profesional

Para finalizar la sección de la función “switch”, se aplica una impresión por defecto que aplica una alerta en caso de que la variable “estado” tenga un valor inadecuado. Además, el código finaliza aplicando un lapso de 8 (s) hasta realizar la próxima medición, todo esto se aprecia en la Figura 3.23.

```

default:
Serial.println("Problemas con el selector de modos");
delay(5000);
break;
}
delay (8000); // Espera 8 s para realizar la proxima medida
}

```

Figura 3.23 Acción por defecto de la función *switch* y espera del programa

Adicionalmente, las sanciones mencionadas serán textualmente las mismas que aparecen en la Tabla 3.4 para conductores profesionales y en la Tabla 3.5 para conductores no profesionales.

Montaje del prototipo en el case

Finalmente, para culminar el proceso de montaje, la placa electrónica se coloca dentro del case junto con la batería LiPo correspondiente que será la fuente de alimentación. La batería que usa es de 7,4 (V) conectada de manera directa al Arduino Nano. La placa Arduino al soportar hasta 12 (v) para su funcionamiento, no necesitará de reguladores de voltaje, por este motivo no se realizó una sección reguladora para la alimentación de la batería. Por medio de la Figura 3.24, se aprecia el resultado de todos los componentes conectados y listos para su funcionamiento.

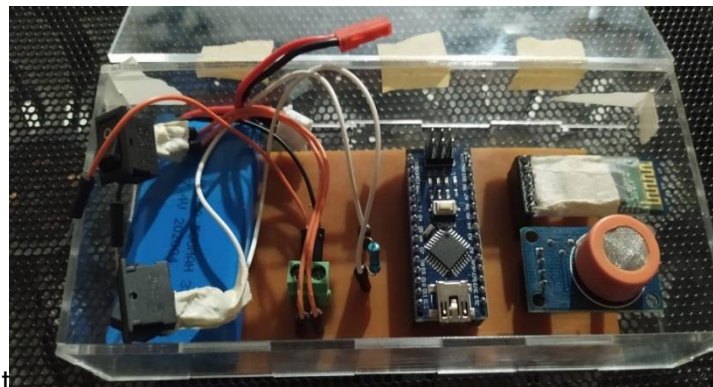


Figura 3.24 Sistema de medición de alcohol dentro del case

Creación de la aplicación móvil

La aplicación móvil fue creada con la idea de proveer una interfaz más directa con el usuario a través de un dispositivo de uso cotidiano; es decir, su teléfono celular. Para este proceso se utilizó la aplicación en línea App Inventor. Dentro de App Inventor se maneja procesos de programación a partir de bloques; al ingresar en un nuevo proyecto dentro del programa, se muestra una pequeña pantalla en la cual los elementos a implementar serán ubicados. Para este caso se utilizó: selectores para escoger los

dispositivos *bluetooth* vinculados al dispositivo móvil, etiquetas con la finalidad que muestren información, un reloj que tiene la función de sincronizador, un sistema de texto a voz y otras características con fines estéticos. Dicho contenido se presenta en la Figura 3.25.

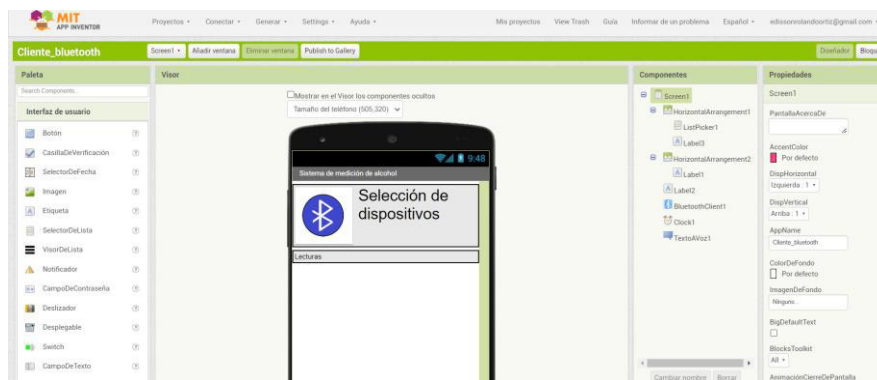


Figura 3.25 Ventana principal del editor de aplicaciones

Para la programación en sí, se desplaza a la sección de “bloques” en la interfaz del programa; aquí se muestran las opciones de programación de cada uno de los elementos incorporados con anterioridad. Se adecúan los bloques para crear sistemas. El primero es la inicialización de una variable global para el almacenamiento de los datos captados por el *bluetooth*; después, se aplica la funcionalidad del selector y el comportamiento de la aplicación al vincularse a un dispositivo; finalmente, se ajusta la salida de datos por medio de las etiquetas aplicadas y el elemento de conversión de texto a voz en función del reloj propuesto para la sincronización de los datos. Todo esto se aprecia de mejor manera en la Figura 3.26.

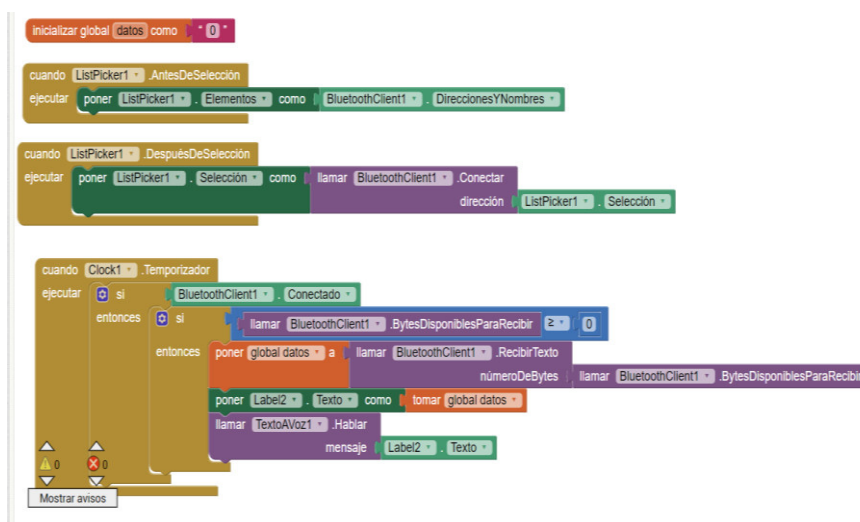


Figura 3.26 Ventana de codificación por bloques dentro del editor de aplicaciones

Al terminar todos los procesos, simplemente se exporta la aplicación a manera de un archivo con extensión *apk*, el cual estará dispuesto a descarga de manera directa o a través de un código QR una vez haya finalizado la compilación de la aplicación realizada. Dicha ventana de descarga se encuentra en la Figura 3.27.

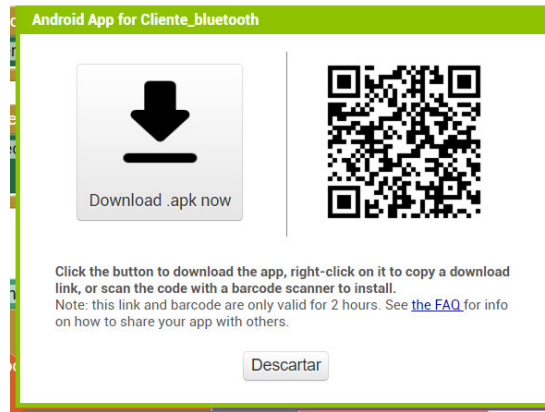


Figura 3.27 Ventana para la descarga de la aplicación móvil

Costos de materiales

Todos los materiales adquiridos están considerados con un costo dentro del mercado ecuatoriano; por este motivo se integra la Tabla 3.6, que da un referente entorno al precio de los materiales aplicados. Cabe indicar que se ha considerado únicamente el costo de los elementos principales; además, no se ha tomado en cuenta el rubro correspondiente a la mano de obra y al desarrollado intelectual.

Tabla 3.6 Costo de elaboración del prototipo de medición de alcohol

Costo de materiales		
Elemento	Cantidad	Costo
Arduino Nano	1	\$9.00
Sensor MQ-3	1	\$8.00
Módulo HC-05	1	\$3.00
Baquelita	1	\$0.50
Batería LiPo 7.4 (V)	1	\$10.00
Case	1	\$3.00
Total		\$33.50

Los materiales principales son la placa Arduino Nano, el sensor de alcohol MQ-3, el módulo de comunicación *bluetooth* HC-05, la baquelita para el circuito electrónico, la fuente de alimentación y el *case*. Materiales como los interruptores y el cableado no se

tomará en cuenta dado su bajo coste o ya que pueden ser sustituidos con elementos reciclados.

3.5 Pruebas de funcionamiento

Una vez finalizó el montaje del sistema sobre la placa electrónica y el *case* y se desarrolló el programa para el funcionamiento del Arduino Nano en conjunto con el módulo de conexión inalámbrica y el sensor de alcohol, se procede a realizar las correspondientes pruebas que justifiquen el adecuado funcionamiento del sistema de medición de alcohol.

Cuando el sistema está encendido (Figura 3.28), el módulo *bluetooth* entrará en un modo promiscuo, el cual permite compartir su dirección física y su nombre a cualquier dispositivo que cuente con conectividad del mismo tipo; para ello, se debe realizar la vinculación entre dispositivos (Figura 3.29).

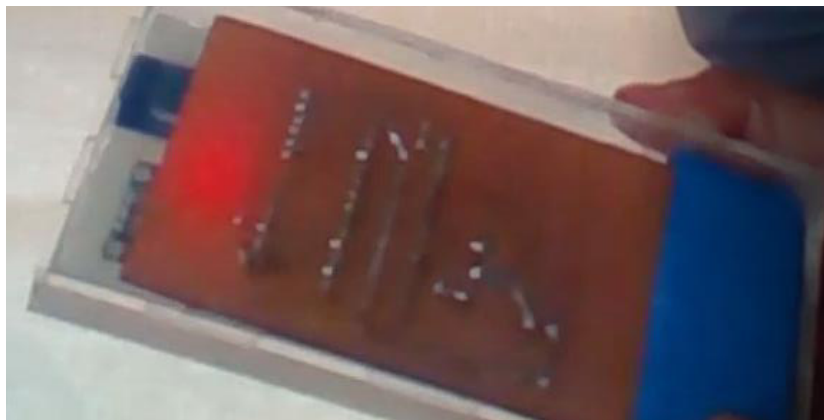


Figura 3.28 Dispositivo de medición encendido

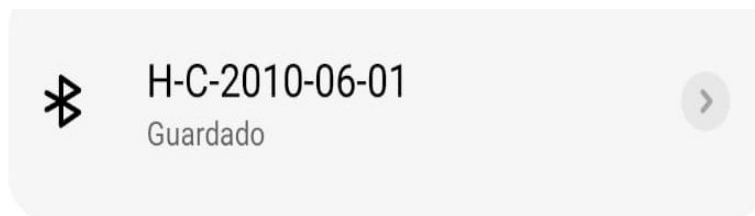


Figura 3.29 Módulo Bluetooth vinculado en el dispositivo móvil

Una vez el dispositivo esté vinculado al módulo, se procede a abrir la aplicación móvil a la cual se denominó “Cliente_bluetooth” (Figura 3.30); aquí se encontrarán elementos como la selección de dispositivos registrados en el dispositivo móvil (un teléfono celular

para este caso) y un recuadro de texto en el que se mostrarán las lecturas que dará el sistema, esto se aprecia en la Figura 3.31.

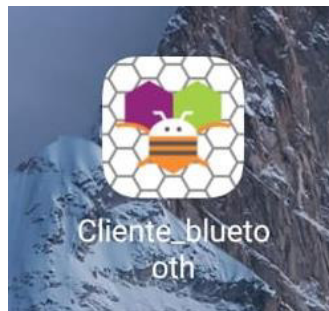


Figura 3.30 Aplicación para dispositivos móviles



Figura 3.31 Interfaz de la aplicación para móviles

Al ingresar a la sección de selección de dispositivos por medio del ícono de *Bluetooth*, aparecerá el listado de elementos registrados, aquí se seleccionará el nombre del dispositivo del sistema que es: H-C-2010-06-01 (Figura 3.32).

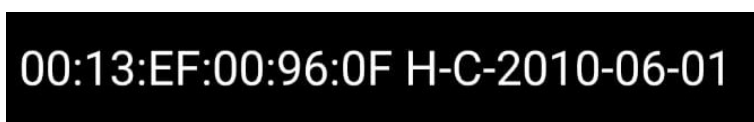


Figura 3.32 Dirección física y nombre del dispositivo HC-05

El sistema estará vinculado al dispositivo, por lo tanto, las lecturas del sensor empezarán a mostrarse en un formato como muestra la interfaz de la aplicación en la Figura 3.33; donde se muestra el valor registrado por el sensor de alcohol, el porcentaje en una escala sobre 1. Para el caso de entrar a uno de los rangos de alerta, se muestra el modo en el cual se encuentra el dispositivo, el rango en el que está y la sanción respectiva (Figura 3.34, Figura 3.35, Figura 3.36, Figura 3.37).



Figura 3.33 Formato de lectura del sistema de medición



Figura 3.34 Alerta al ingresar al rango de 0.3 grados, modo no profesional



Figura 3.35 Alerta al ingresar al rango de 0.8 grados, modo no profesional

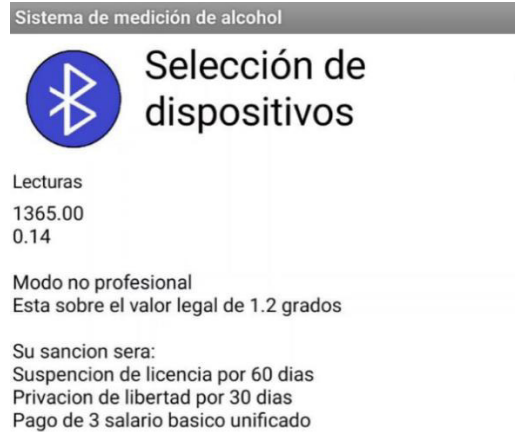


Figura 3.36 Alerta al ingresar al rango de 1.2 grados, modo no profesional

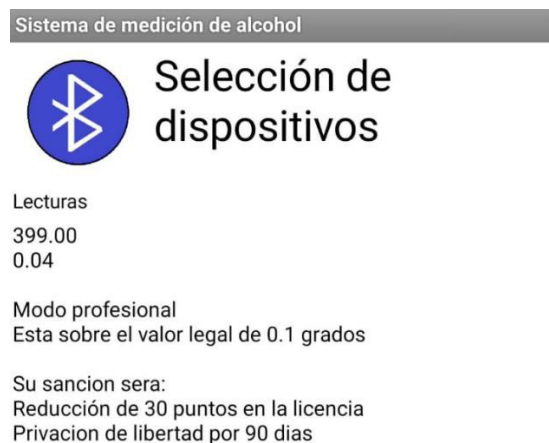


Figura 3.37 Alerta al ingresar al rango de 0.1 grados, modo profesional

El gestor de modalidad estará implementado con un interruptor colocado en la parte frontal del *case*. Servirá para conmutar entre el modo profesional y no profesional; ubicándose justo al lado del interruptor de encendido del sistema, como se muestra en la Figura 3.38.



Figura 3.38 Disposición de interruptores en el *case*

Los resultados obtenidos en base a las pruebas de funcionamiento están representados en la Tabla 3.7, la que establece mediante una “X” en el renglón correspondiente si las pruebas tuvieron un desempeño adecuado respecto a las metas propuestas o si se presentó alguna complicación al momento de efectuar la evaluación en alguno de los parámetros presentes. Dando así la validación del funcionamiento óptimo del sistema de medición de alcohol en función de su comportamiento portátil, medición, sus rangos de análisis y sus modos de uso.

Tabla 3.7 Resultados de funcionamiento

Parámetro de prueba	Funcionamiento	
	Correcto	Incorrecto
Comportamiento portátil	X	
Medición	X	
Rango de 0.3 grados (modo no profesional)	X	
Rango de 0.8 grados (modo no profesional)	X	
Rango de 1.2 grados (modo no profesional)	X	
Rango de 0.1 grados (modo profesional)	X	

Las pruebas aplicadas están registradas de manera digital en el Anexo II.I por medio de un código QR que redirecciona a la ubicación de un archivo de video y el manual de usuario se muestra mediante un código ubicado en el Anexo II.II.

4 CONCLUSIONES

- El prototipo cumple en efecto con los requerimientos planteados. El sensor aplicado (MQ-3), es el adecuado para la detección de alcohol en el ambiente dado su esquema de funcionamiento. La funcionalidad dentro de una red PAN está cubierta por el módulo *bluetooth*, el cual brinda una transmisión de datos directa al dispositivo móvil con el que esté vinculado. La autonomía por parte de la batería LiPo cumple con el intervalo de voltaje establecido entre 5 y 12 (V). El *case* encapsula y protege a los componentes internos, cumpliendo así el requerimiento de protección. Por último, la conexión e interfaz de usuario que

brinda la aplicación móvil es la adecuada para el sistema ya que propone un método de alerta transparente y sencillo de comprender.

- Con la conexión *bluetooth* entre el dispositivo de medida de alcohol y el teléfono móvil, se apreció el funcionamiento correcto de todos los dispositivos aplicados. El sensor MQ-3 envió todos los datos concernientes por el puerto digital del Arduino, el cual procesó esta información para transmitirla de manera inalámbrica al celular en el que se aprecia, por medio de la interfaz gráfica, el sistema de alerta propuesto cuando los límites de alcohol establecidos son superados.
- El reducido tamaño de los materiales y sensores proveen portabilidad adecuada al dispositivo, además de una perfecta autonomía para el tiempo de uso requerido. El Arduino Nano es perfectamente funcional con los módulos aplicados y responde de manera satisfactoria con la aplicación generada por medio de App Inventor dentro de la que se gestó el programa de comunicación y alerta para el dispositivo móvil.
- Debido a la implementación del sistema modular de Arduino, el circuito impreso únicamente está centrado para las vías de alimentación de los elementos. Por el bajo consumo del sistema de medición, la batería proporciona la suficiente energía para el correcto funcionamiento del dispositivo. Además, provee un comportamiento adecuado al momento de realizar las mediciones del aliento alcohólico, el cual está establecido entre 0.1 a 0.8 grados de alcohol, mostrando así la alerta respectiva en pantalla según el caso lo amerite.
- Las pruebas de funcionamiento mostraron que el prototipo se desenvuelve de manera apropiada en cada una de las situaciones. El dispositivo al captar la presencia de alcohol registra los datos de modo adecuado, que son mostrados en la interfaz de la aplicación móvil; en el caso de llegar a uno de los rangos establecidos dentro de la programación, por medio del dispositivo móvil se efectúa el sistema de alerta tanto visual como sonora. En cada alerta se especifica la medición obtenida, el modo de operación del prototipo, el rango en el que se encuentra la medición y la sanción respectiva.
- La medición de alcohol realizada por el prototipo no es ciertamente la verdadera en relación con el nivel de alcohol en la sangre de una persona, esto se da debido a que el sensor MQ-3 únicamente detectará la presencia de alcohol en el aliento del conductor, esto se muestra en las pruebas de funcionamiento. Con unas pocas copas de alcohol, el valor mostrado en pantalla llegó a encontrarse dentro

del rango máximo a pesar de que el sujeto a prueba no se encontraba en un estado etílico avanzado.

5 RECOMENDACIONES

- Para proyectos futuros basados en la implementación del presente prototipo, se recomienda implementar un registro automático en el sistema de la multa registrada; de este modo, se evitaría situaciones de posible coima a los agentes de tránsito.
- Partiendo del prototipo presentado, otros proyectos pueden enfocarse en la implementación de un interruptor, el cual permita realizar mediciones a voluntad; es decir, que, si el usuario acciona un pulsador o algo similar, la medición se realice en ese momento y no esté sujeta al lapso comprendido en la programación.
- Si en algún momento, la ley vigente en el Ecuador llegase a cambiar y afectara de manera directa a los rangos establecidos por el sistema, estos podrán ser cambiados de manera sencilla. Simplemente hay que ingresar al código de programación y rectificar los datos aplicados en los valores límite. Incluso, si se modifica el número de rangos, estos están en la facultad de ser cambiados dentro de las sentencias “if” correspondientes.
- Dado a que el prototipo no muestra los niveles de alcohol verdaderos, ya que emplea el aliento de la persona, se puede hacer uso de otro tipo de sistemas para detectar de manera real el valor por litro de sangre. Uno de estos métodos, y siendo el más preciso, es directamente una prueba de sangre; esto verificará de manera real el porcentaje de alcohol contenido por cada litro de sangre.

6 REFERENCIAS BIBLIOGRÁFICAS

- [1] OPS, *Beber y Conducir*, . Washington, D.C.: PAHO, 2017.

- [2] S. Virginia, «Etitudela,» [En línea]. Available: <http://www.etitudela.com/entrenadorcomunicaciones/downloads/bluetoothguiarapida.pdf>. [Último acceso: 18 Mayo 2022].
- [3] A. Moreno Muñoz y S. Córcoles Córcoles, *Arduino. Curso práctico*, Madrid: RA-MA, 2018.
- [4] Arduino, «Hardware Libre,» Hardware Libre, [En línea]. Available: <https://www.hwlibre.com/arduino-nano/>. [Último acceso: 9 Mayo 2022].
- [5] L. G. Corono Ramírez , G. S. Abarca Jiménez y J. Mares Carreño, *Sensores y actuadores. Aplicaciones con Arduino.*, México: Grupo Editorial Patria S. A. de C. V., 2014.
- [6] V. Madan y S. R. N. Reddy, «GSM-Bluetooth based Remote Monitoring and Control,» *International Journal of Computer Applications*, vol. 46, nº 1, pp. 20-28, 2012.
- [7] V. Kumar Singh, B. Khan, A. Sahu, S. Kumar y A. Beg, «Speed & Direction Control of DC Motor through,» de *International Conference on Advanced Computation and Telecommunication (ICACAT)*, Bhopal, India, 2018.
- [8] 3D Native, «3D Native,» 3D Native, 17 Junio 2020. [En línea]. Available: <https://www.3dnatives.com/es/plasticos-impresion-3d-22072015/>. [Último acceso: 11 Mayo 2022].
- [9] A. E, M. P, D. C y L. G, «An Experimental Comparison of Arduino IDE Compatible Platforms for Digital Control and Data Acquisition Applications,» de *CHILECON*, Valparaíso, 2019.
- [10] Descubre Arduino, «Descubre Arduino,» Descubre Arduino, 3 Julio 2021. [En línea]. Available: <https://descubrearduino.com/arduino-nano-pinout/>. [Último acceso: 18 Mayo 2022].
- [11] Naylamp Mechatronics, «Naylamp Mechatronics,» Naylamp Mechatronics, 13 Julio 2016. [En línea]. Available: https://naylampmechatronics.com/blog/42_tutorial-sensores-de-gas-mq2-mq3-mq7-y-mq135.html. [Último acceso: 18 Mayo 2022].

- [12] L. Luis, «Luis Llamas,» 21 Octubre 2016. [En línea]. Available: <https://www.luisllamas.es/arduino-detector-gas-mq/>. [Último acceso: 18 Mayo 2022].
- [13] Guangzhou, «All data sheet,» 6 Abril 2011. [En línea]. Available: <https://html.alldatasheet.com/html-pdf/1179032/ETC1/HC-06/109/1/HC-06.html>. [Último acceso: 6 Junio 2022].
- [14] Arduino, «Arduino.cl,» Arduino.cl, [En línea]. Available: <https://arduino.cl/arduino-nano/>. [Último acceso: 8 Junio 2022].
- [15] ETC2 , «ETC2,» [En línea]. Available: <https://html.alldatasheet.com/html-pdf/1150676/ETC2/MQ3/107/1/MQ3.html>. [Último acceso: 8 Junio 2022].
- [16] Corte Nacional de Justicia, *CÓDIGO ORGÁNICO INTEGRAL PENAL*, Quito, 2021.
- [17] Agencia Nacional de Tránsito, «Portovial,» 10 Agosto 2021. [En línea]. Available: <https://portovial.gob.ec/sitio/descargas/leyes/ley-organica-transporte-terrestre-transito-y-seguridad-vial.pdf>. [Último acceso: 8 Junio 2022].

7 ANEXOS

La lista de los **Anexos** se muestra a continuación:

ANEXO I. Certificado de originalidad

ANEXO II. Enlaces

ANEXO III. Conjunto de datos extensos

ANEXO I: Certificado de Originalidad

CERTIFICADO DE ORIGINALIDAD

Quito, D.M. 20 de Agosto de 2022

De mi consideración:

Yo, FANNY PAULINA FLORES ESTÉVEZ, en calidad de Director del Trabajo de Integración Curricular titulado IMPLEMENTACIÓN DE UN PROTOTIPO DE SISTEMA DE ALERTA POR DETECCIÓN DE ALCOHOL elaborado por el estudiante EDISSON ROLANDO ORTIZ SALAZAR de la carrera en TECNOLOGÍA SUPERIOR EN REDES Y TELECOMUNICACIONES, certifico que he empleado la herramienta Turnitin para la revisión de originalidad del documento escrito completo, producto del Trabajo de Integración Curricular indicado.

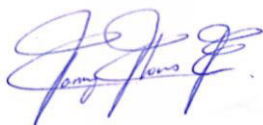
El documento escrito tiene un índice de similitud del 10%.

Es todo cuanto puedo certificar en honor a la verdad, pudiendo el interesado hacer uso del presente documento para los trámites de titulación.

NOTA: Se adjunta el link del informe generado por la herramienta Turnitin.

[Turnitin TIC2022A Edison Ortiz.pdf](#)

Atentamente,



FANNY PAULINA FLORES ESTÉVEZ

Docente

Escuela de Formación de Tecnólogos

ANEXO II: Enlaces

Anexo II.I Código QR de la implementación y pruebas de funcionamiento



Enlace del video: <https://www.youtube.com/watch?v=GEMeabu-aFk&t=281s>

Anexo II.II Código QR del manual de usuario



Enlace del video: <https://www.youtube.com/watch?v=aysJzyBo89s>

ANEXO III: Códigos Fuente

```
int valor_limite1= 300; // Fija el valor limite en el que se activa la alarma
int valor_limite2= 800;
int valor_limite3= 1200;
int valor_limite4= 100;
float valor_alcohol;
float valor_alcoho;
int estado;
char c = ' ';
void setup() {
    //Mensaje inicial
    Serial.begin(9600);
    Serial.println("EL SISTEMA ESTA LISTO");
    Serial.println("TENER EN CUENTA LOS VALORES A MEDIR");
    //Establece el comportamiento de los pines
    pinMode(5,INPUT);
    pinMode(6, OUTPUT);
}
void loop() {
    //Comportamiento ingreso o salida de datos
    digitalWrite(6,HIGH);
    estado=digitalRead(5);
    valor_alcoho=analogRead(A0);
    valor_alcohol=valor_alcoho;
    //Impresión de datos al puerto serial
    Serial.println(valor_alcohol);    // Envia al Serial el valor leído del Sensor MQ-3
    float porcentaje=(valor_alcohol/10000); //Calcula el porcentaje
    Serial.println(porcentaje);      // Envia al Serial el porcentaje
    Serial.println(' ');
    delay (5000);

    //Validacion de límites
    switch(estado){
        //modo no profesional
```

case 0:

```
if(valor_alcohol > valor_limite1 && valor_alcohol <= valor_limite2 )
{ // Si la medida de gas metano es mayor de valor limite 1
  Serial.println("Modo no profesional");
  delay(3000);
  Serial.println("Esta sobre el valor legal de 0.3 grados");
  Serial.println(' ');
  Serial.println("Su sancion sera:");
  Serial.println("Perdida de 5 puntos de licencia");
  Serial.println("Privacion de libertad por 5 dias");
  Serial.println("Pago de 1 salario basico unificado");
  Serial.println(' ');
  delay(3000);
}
```

```
if(valor_alcohol > valor_limite2 && valor_alcohol <= valor_limite3)
{ // Si la medida de gas metano es mayor de valor limite2
  Serial.println("Modo no profesional");
  delay(3000);
  Serial.println("Esta sobre el valor legal de 0.8 grados");
  Serial.println(' ');
  Serial.println("Su sancion sera:");
  Serial.println("Perdida de 10 puntos de licencia");
  Serial.println("Privacion de libertad por 15 dias");
  Serial.println("Pago de 2 salario basico unificado");
  Serial.println(' ');
  delay(3000);
}
```

```
if(valor_alcohol > valor_limite3)
{ // Si la medida de gas metano es mayor de valor limite3
  Serial.println("Modo no profesional");
  delay(3000);
  Serial.println("Esta sobre el valor legal de 1.2 grados");
  Serial.println(' ');
```

```

    Serial.println("Su sancion sera:");
    Serial.println("Suspension de licencia por 60 dias");
    Serial.println("Privacion de libertad por 30 dias");
    Serial.println("Pago de 3 salario basico unificado");
    Serial.println(' ');
    delay(3000);
}
break;
//modo profesional
case 1:
if(valor_alcohol > valor_limite4)
{ // Si la medida de gas metano es mayor de valor limite4

    Serial.println("Modo profesional");
    delay(3000);
    Serial.println("Esta sobre el valor legal de 0.1 grados");
    Serial.println(' ');
    Serial.println("Su sancion sera:");
    Serial.println("Reducción de 30 puntos en la licencia");
    Serial.println("Privacion de libertad por 90 dias");
    Serial.println(' ');
    delay(3000);
}
break;
default:
Serial.println("Problemas con el selector de modos");
delay(3000);
break;
}
delay (8000); // Espera 8 s para realizar la proxima medida
}

```