

ESCUELA POLITÉCNICA NACIONAL

ESCUELA DE FORMACIÓN DE TECNÓLOGOS

IMPLEMENTACIÓN DE UN PROTOTIPO DE SISTEMA DE ALERTA PARA DETECCIÓN EN TIEMPO REAL DE DESLIZAMIENTOS DE TIERRA

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE
TECNÓLOGO SUPERIOR EN REDES Y TELECOMUNICACIONES**

JOEL ALEXIS QUINTANILLA ARMIJO

joel.quintanilla@epn.edu.ec

DIRECTOR: Ing. Fanny Paulina Flores Estévez, MSc.

fanny.flores@epn.edu.ec

CODIRECTOR: Ing. Fabio Matías González González, MSc.

fabio.gonzalez@epn.edu.ec


Quito, octubre 2021

CERTIFICACIÓN

Certificamos que el presente trabajo fue desarrollado por el Sr. Quintanilla Armijo Joel Alexis como requerimiento parcial a la obtención del título de TECNÓLOGO SUPERIOR EN REDES Y TELECOMUNICACIONES, bajo nuestra supervisión:



Ing. Fanny Flores Estévez MSc.
DIRECTORA DEL PROYECTO



Ing. Fabio González González MSc.
CODIRECTOR DEL PROYECTO

DECLARACIÓN

Yo, Quintanilla Armijo Joel Alexis con CI: 1727145961 declaro bajo juramento que el trabajo 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.

Sin perjuicio de los derechos reconocidos en el primer párrafo del artículo 144 del Código Orgánico de la Economía Social de los Conocimientos, Creatividad e Innovación – COESC-, soy titulares de la obra en mención y otorgo una licencia gratuita, intransferible y no exclusiva de uso con fines académicos a la Escuela Politécnica Nacional.

Entrego toda la información técnica pertinente, en caso de que hubiese una explotación comercial de la obra por parte de la EPN, se negociará los porcentajes de los beneficios conforme lo establece la normativa nacional vigente.



Joel Alexis Quintanilla Armijo

DEDICATORIA

Este proyecto de titulación va dedicado a mis padres Rosario y Heraldo, a mis hermanos que me apoyaron en todo momento, y han estado conmigo dándome ánimo para seguir adelante, pero sobre todo se lo dedico a Dios ya que Él me dio la sabiduría y fortaleza para no darme por vencido.

Joel

AGRADECIMIENTO

Agradezco a Dios por su respaldo en todo momento, por siempre estar conmigo y no dejarme solo en los momentos más difíciles.

A mis padres Rosario y Heraldo, que siempre me impulsaron a seguir, a mis hermanos Nataly y Ricardo que me han brindado su apoyo y me han aconsejado como unos segundos padres.

A mi amiga Josselyn que me ha acompañado en este largo camino, con su apoyo, cariño y consejos que me han levantado el ánimo.

A mi tutora de tesis Ing. Fanny Flores por estar continuamente respaldando el trabajo desde el inicio hasta el final.

A la Escuela Politécnica Nacional, a la ESFOT y sus docentes que con su exigencia me enseñaron a esforzarme y a ser mejor estudiante y persona.

Joel

ÍNDICE DE CONTENIDOS

1	INTRODUCCIÓN	1
1.1	Objetivo general.....	2
1.2	Objetivos específicos	2
1.3	Fundamentos	2
2	METODOLOGÍA.....	8
2.1	Descripción de la metodología usada	8
3	RESULTADOS Y DISCUSIÓN	10
3.1	Análisis de requerimientos técnicos del sistema	10
3.2	Diseño del sistema.....	13
3.3	Implementación del sistema	25
3.4	Pruebas y Análisis de Resultados	33
	Comprobación de los motores	33
	Funcionamiento de LCD I2C.....	34
	Envío de información a base de datos	35
3.5	Manual de Usuario.....	37
4	CONCLUSIONES Y RECOMENDACIONES	37
4.1	Conclusiones.....	37
4.2	Recomendaciones	39
5	REFERENCIAS BIBLIOGRÁFICAS	39
	ANEXOS.....	43
	Anexo 1: Certificado de Funcionamiento	i

ÍNDICE DE FIGURAS

Figura 1.1 Celda de carga 1 (kg).....	3
Figura 1.2 Galga extensiométrica resistiva	3
Figura 1.3 Circuito eléctrico o puente de Wheatstone.....	4
Figura 1.4 Amplificador HX711	4
Figura 1.5 Tensiones por el puente de Wheatstone.....	5
Figura 1.6 Módulo NodeMCU V3	5
Figura 1.7. Placa Arduino Uno	6
Figura 1.8. Pantalla LCD I2C.	7
Figura 1.9 Servidor WAMP en un sistema de Windows.....	7
Figura 1.10 Servidor WAMP instalando aplicaciones web en red.....	8
Figura 3.1 Manejo del Riesgo Sísmico de Quito.....	11
Figura 3.2 Elementos utilizados en el sistema.....	12
Figura 3.3 Flujograma del código de calibración	14
Figura 3.4 Código PHP.....	15
Figura 3.5 Ubicación de los motores de vibración en la maqueta.....	16
Figura 3.6 Distribución de pines de la placa Arduino UNO	17
Figura 3.7 Flujograma de código de placa Arduino	18
Figura 3.8 Distribución de pines del módulo WiFi	19
Figura 3.9 Flujograma del código del módulo WiFi	20
Figura 3.10 Circuito de alimentación para motores de vibración	21
Figura 3.11 Diseño de la maqueta en 3D	22
Figura 3.12 Vista frontal de la maqueta	22
Figura 3.13 Vista lateral para sensor de peso, motores de vibración y bases	22
Figura 3.14 Vista lateral para placa PCB y pantalla LCD I2C	23
Figura 3.15 Diseño en Proteus de la placa PCB	24
Figura 3.16 Parte superior de la placa PCB.....	24
Figura 3.17 Parte inferior de la placa PCB	25
Figura 3.18 Base de madera para la maqueta	25
Figura 3.19 Bases para la construcción de la maqueta.....	26
Figura 3.20 Relleno de la maqueta	26
Figura 3.21 Avance de la construcción de la maqueta.....	26

Figura 3.22 Maqueta estructurada	27
Figura 3.23 Proceso de blanqueamiento de la maqueta	27
Figura 3.24 Motores de vibración en las bases de apoyo	28
Figura 3.25 Circuito de alimentación de motores de vibración.....	28
Figura 3.26 Vista de la ubicación del sensor de peso	29
Figura 3.27 Ubicación de la placa PCB en la maqueta	29
Figura 3.28 Aspecto final de la maqueta.....	29
Figura 3.29 Limpieza de la baquelita	30
Figura 3.30 Estampadora a 155 (°C)	30
Figura 3.31 Estampado de 1 a 3 (min) en la estampadora	30
Figura 3.32 Placa retirada de la plancha	31
Figura 3.33 Placa sumergida en ácido férrico	31
Figura 3.34 Placa pulida	31
Figura 3.35 Perforación de la placa	32
Figura 3.36 Ubicación de Arduino en placa PCB.....	32
Figura 3.37 Ubicación del módulo WiFi en la placa PCB	32
Figura 3.38 Placa PCB con componentes conectados	33
Figura 3.39 Ubicación inicial de la tierra	33
Figura 3.40 Switches de encendido de motores.....	34
Figura 3.41 Caída progresiva de la tierra.....	34
Figura 3.42 Peso detectado por el sensor de peso	35
Figura 3.43 Peso reflejado en el LCD	35
Figura 3.44 Cambio de peso en el sensor	35
Figura 3.45 Cambio de peso reflejado en el LCD	35
Figura 3.46 Envío exitoso de información por el módulo WiFi	36
Figura 3.47 Verificación del envío de información en la base de datos	36
Figura 3.48. Verificación de funcionamiento del sistema	37
Figura 3.49. Manual de Usuario	37

ÍNDICE DE TABLAS

Tabla 1.1 Especificación de uso de fuentes de voltaje	21
--------------------------------------------------------------------	----

RESUMEN

El presente proyecto de titulación, Implementación de Un Prototipo de Sistema de Alerta Para Detección En Tiempo Real de Deslizamiento de Tierra, pretende servir como un medio de prevención para las viviendas que se encuentren aledañas a zonas propensas a ocurrir un deslizamiento de tierra. Además, permitirá mantener un monitoreo constante del estado de las zonas vulnerables mediante un registro constante de la información en una base de datos, por medio de comunicación inalámbrica. En función del estudio realizado, el presente documento está conformado por cinco secciones, las mismas que se detallan a continuación.

En la primera sección se incluye la introducción del proyecto, la misma que contiene el planteamiento del problema, que es el eje principal para poder determinar los diferentes componentes que van a permitir la detección, monitoreo y alerta de un deslizamiento de tierra; además, se detallan el objetivo general, los objetivos específicos orientados a cada una de las tareas que se desarrollan en el proyecto y los fundamentos.

En la segunda sección se trata sobre la metodología aplicada en el desarrollo del proyecto, utilizando los lineamientos y procesos que permiten la ejecución de cada uno de los objetivos.

En la tercera sección se desarrolla los resultados y discusión, en donde se describen los requerimientos técnicos, el diseño del sistema, la implementación, las pruebas de funcionamiento, se incorpora un video de verificación del funcionamiento y un manual de usuario.

La cuarta sección contiene las conclusiones y recomendaciones obtenidas después del diseño y pruebas de funcionamiento del prototipo. Finalmente, en la quinta sección se presentan las referencias bibliográficas que respaldan el desarrollo del documento.

PALABRAS CLAVE: Deslizamiento, comunicación inalámbrica, base de datos, servidor, alerta.

ABSTRACT

This degree project, Implementation of a Prototype of Warning System for Real Time Detection of Landslide, aims to serve as a means of prevention for homes that are close to areas prone to landslides. In addition, it will allow constant monitoring of the status of vulnerable areas through a constant record of information in a database via wireless communication. Based on the study carried out, this document is made up of five sections, which are detailed below.

In the first section, the introduction of the project is included, it presents the problem statement, which is the main axis to determine the different components that will allow the detection, monitoring and warning of a landslide; in addition, the general objective, the specific objectives oriented to each of the tasks to be developed in the project and the fundamentals are detailed.

The second section deals with the methodology applied in the development of the project, using the guidelines and processes that allow the execution of each objective.

The third section develops the results and discussion where the technical requirements, the system design, the implementation, the functional tests and a user manual are described.

The fourth section contains the conclusions and recommendations obtained after the design and testing of the prototype. Finally, the fifth section presents the bibliographical references that support the development of the document.

KEYWORDS: *Sliding, wireless communication, database, server, alerting.*

1 INTRODUCCIÓN

De acuerdo con el Servicio Nacional de Gestión de Riesgos y Emergencias de Ecuador, un deslave constituye la caída de tierra o rocas desde una ladera, a consecuencia de un sismo o por lluvia. Dichos eventos geológicos son más frecuentes en regiones montañosas durante épocas lluviosas; de modo que la tierra tiende a ablandarse y a desprenderse. Sin embargo, la acción del ser humano también propicia a que se produzcan deslaves; por ejemplo, ciertas personas construyen sobre terrenos no aptos y aledaños a las laderas, ocasionando deslizamientos de tierra. Adicionalmente, la deforestación tiende a desgastar el suelo, ablandando la tierra y generando taludes [1].

Resulta alarmante que, hasta enero del 2020, 77 zonas de Quito fueron reportadas en riesgo por deslizamientos; dichos eventos generan el cierre de carreteras, trabajos que ponen en riesgo a obreros de la Empresa Pública Metropolitana de Movilidad y Obras Públicas (EPMMOP) e incluso generan pérdidas materiales y humanas. Tan solo entre septiembre del 2019 y enero de 2020 se reportaron 126 emergencias, de las cuales 59 se debieron a deslizamientos de tierra. Los sectores más susceptibles a deslizamientos en la ciudad se ubican en vías como la Simón Bolívar, Interoceánica, además en barrios como Osorio, Pinar Alto, Comuna y Chillogallo [2].

Ante esta inminente amenaza, el Servicio Nacional de Gestión de Riesgos y Emergencias, recuerda constantemente a los ciudadanos que las pendientes pronunciadas no son lugares seguros para construir; y en caso de habitar en dichos lugares se debe evitar filtraciones de agua mediante el uso de canales limpios que permitan que el agua lluvia corra libremente. Así también se recomienda conservar la vegetación, cuyas raíces aseguran la sujeción de la tierra y permiten regular la cantidad de agua en el suelo [1]. Pese a estas recomendaciones, muchas personas hacen caso omiso, y se repiten constantemente tragedias a consecuencia de los taludes. Ante esta situación, se propone la implementación de un prototipo de sistema de alerta para detección en tiempo real de deslizamientos de tierra. De este modo, mediante el monitoreo del peso de la tierra, se podrá determinar la variación de acumulación en un determinado punto y así alertar que se está produciendo un deslave, permitiendo una evacuación emergente y precautelando la seguridad de los habitantes de la zona aledaña.

1.1 Objetivo general

Implementar un prototipo de sistema de alerta para detección en tiempo real de deslizamientos de tierra.

1.2 Objetivos específicos

- Analizar los requerimientos técnicos para la detección en tiempo real de deslizamientos de tierra.
- Diseñar el prototipo para la detección en tiempo real de deslizamientos de tierra.
- Implementar el prototipo para simular la detección en tiempo real de deslizamientos de tierra.
- Realizar pruebas de funcionamiento del prototipo.

1.3 Fundamentos

Cuando se habla de un deslizamiento de tierra, se habla de un concepto muy amplio; el mismo se genera cuesta abajo y hacia afuera, finalizando con la caída de tierra sobre un plano despejado. La erosión del suelo se produce en el desprendimiento, transporte y depósito de partículas o masas de suelo o rocas, provocadas por fuerzas que generan este tipo de movimiento debido a la acción del agua. Sumado a esto, la lluvia contribuye al desprendimiento de las partículas o granos. La erosión es un proceso recurrente en los suelos que no cuentan con un cimientado fijo o en los suelos aluviales [3].

Un sistema de alerta temprana que detecta deslizamientos de tierra en tiempo real es un mecanismo que acepta el monitoreo de eventos en el transcurso de un año, las 24 horas del día; de esta manera, se informa al personal encargado de monitorear, en caso de haber ocurrido un desastre natural por medio de comunicación inalámbrica [4]. El método que se encargará de identificar los cambios o irregularidades que se pueden presentar en el nivel del suelo son los sensores, que a su vez al identificar este cambio, activarán una alarma programada mediante *hardware* como respuesta al deslizamiento.

Hardware

- Celda de carga

Una celda de carga es un tipo de transductor de fuerza que convierte una fuerza mecánica ya sea peso, tensión o compresión, en una señal eléctrica que se pueda evaluar, cambiar y estandarizar. A medida que aumenta la fuerza que se aplica en el

sensor, la señal eléctrica aumenta proporcionalmente. Este es el elemento que permite conocer el peso aplicado en el sistema. Existen diferentes tipos de celdas de carga, con diferentes métodos de funcionamiento; la celda de carga que se utiliza en el proyecto es la que se observa en la Figura 1.1, que es una celda que tiene un valor máximo de medición de 1 (Kg) [5].

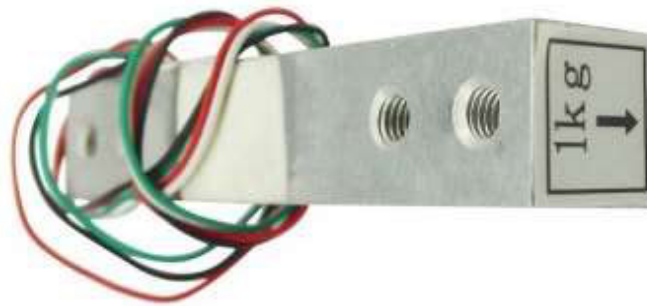


Figura 1.1 Celda de carga 1 (kg) [5]

Para poder variar las magnitudes de la fuerza, utiliza una galga extensiométrica que es un sensor que cambia su resistencia en función de la presión que se ejerce sobre él debido a un efecto piezoresistivo, como se evidencia en la Figura 1.2 [6].

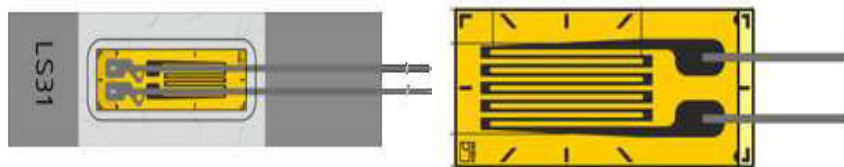


Figura 1.2 Galga extensiométrica resistiva [6]

La galga extensiométrica más común está formada por un alambre o lámina muy fina configurada de tal manera que hay un cambio lineal en la resistencia eléctrica cuando se aplica una deformación en una dirección específica; las resistencias más comunes son las de 120 (Ω), 350 (Ω) y 1000 (Ω) [6].

El inconveniente que se presenta en las galgas extensiométricas es la variación en su resistencia, ya que sus cambios son limitadamente pequeños. En otras palabras, si la resistencia es de 120 (Ω), los cambios que se pueden producir con un peso considerable es desde 0.12 (Ω) aproximadamente. Es aquí donde entra en juego el puente de *Wheatstone*, que consiste en un circuito simple con la capacidad de detectar las pequeñas variaciones de resistencia. El puente de *Wheatstone* se utiliza para medir resistencias desconocidas mediante la técnica determinada como el equilibrio de brazos

de puente. Conociendo la tensión de la fuente y V_G , se puede obtener el valor de la resistencia R_x . Si la resistencia R_x , es una galga extensiométrica, será proporcional al peso que esta determina en la galga mediante el circuito que se muestra en la Figura 1.3 [6].

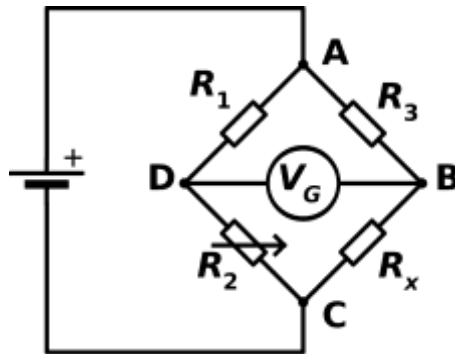


Figura 1.3 Circuito eléctrico o puente de *Wheatstone* [6]

- Amplificador HX711

El amplificador de celda de carga HX711, que se observa en la Figura 1.4, tiene dos hileras de pines; una hilera le permite conectarse a la celda de carga y la otra le da la capacidad de conectarse a un microcontrolador (Arduino). Este módulo es una interfaz entre la celda de carga y el microcontrolador, que permite leer el valor del peso de manera sencilla. Este amplificador es el encargado de percibir los datos del puente de *Wheatstone* formado por la celda de carga, convirtiendo la lectura analógica a digital con su conversor A/D interno de 24 bits [6].

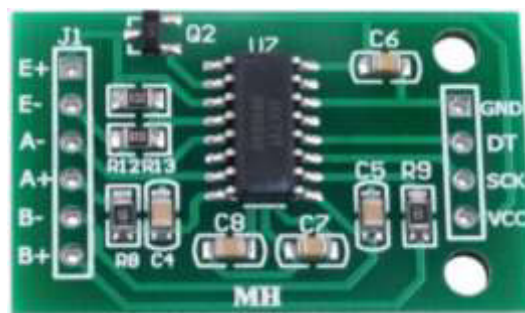


Figura 1.4 Amplificador HX711 [6]

Por lo general, la celda de carga posee 4 puertos, los mismos que ayudarán a medir las 2 tensiones que se obtienen a través del puente de *Wheatstone*. Por un lado, se encuentra la tensión de entrada o de excitación y por otro se encuentra la tensión de salida, como se aprecia en la Figura 1.5.

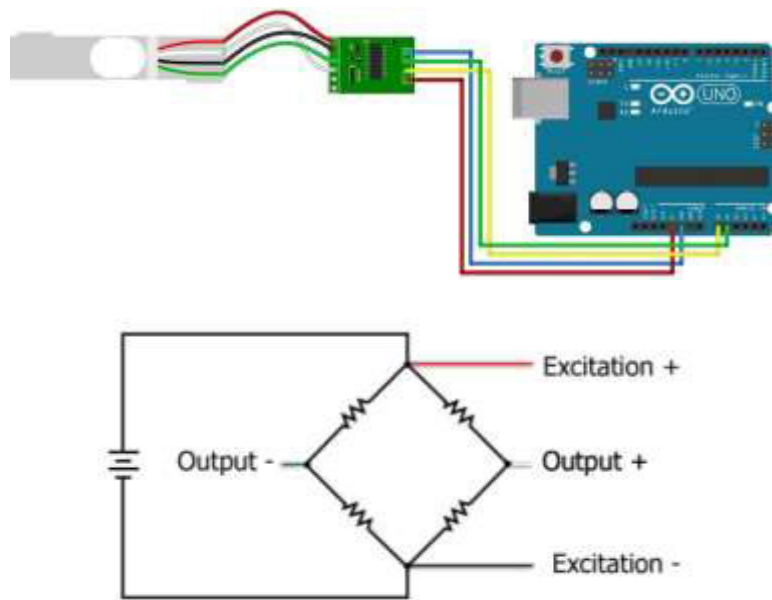


Figura 1.5 Tensiones por el puente de *Wheatstone* [6]

- **ESP8255MOD**

El módulo *NodeMCU* se usa ampliamente en proyectos de Internet de las Cosas (IoT), debido a que es un microcontrolador con un *chip* ESP8266 integrado, el cual provee conectividad *WiFi*, dando apertura a ser utilizado como un puente entre Internet y un microcontrolador (Arduino) [7]. Para el presente proyecto se utiliza el módulo *NodeMCU V3* que se observa en la Figura 1.6, como método de conexión inalámbrica para enviar la información que reflejará el resultado del monitoreo.



Figura 1.6 Módulo *NodeMCU V3* [7]

- **Arduino Uno**

Arduino es una plataforma de creación de electrónica de código abierto, basada en *hardware* y *software* libre, sencillo de utilizar para creadores y desarrolladores; esta plataforma permite crear una serie de microordenadores en una misma placa a la que se le puede dar distintos tipos de uso. Las placas Arduino cuentan con microcontroladores programables Atmel, los que proporcionan una interfaz de entrada que permite al microcontrolador interactuar con una amplia gama de módulos y sensores.

El microcontrolador Arduino Uno es una placa electrónica fundada en el *chip* de Atmel ATmega328; cuenta con un total de 14 pines digitales de entrada/salida, de los cuales 6 pueden ser usados como salida PWM, 6 entradas analógicas, un oscilador de cristal de 16 (MHz), conexión USB, conector de alimentación y un botón de *reset*. Esta placa cuenta con un *software* que permite controlar la conexión USB simulando a un *mouse*, un teclado y un puerto serie [8]; en la Figura 1.7 se puede apreciar la placa Arduino Uno.



Figura 1.7. Placa Arduino Uno [8]

- LCD I2C

I2C es un protocolo de comunicación, conocido generalmente como “bus I2C”; este protocolo fue diseñado para establecer una comunicación entre componentes de una sola placa [8].

La pantalla LCD de 16x2, que se puede apreciar en la Figura 1.8, se basa en el controlador HD44780 que es un módulo ampliamente usado; el inconveniente con este módulo es que necesita muchos pines del microcontrolador para poder controlarlo, por

este motivo se utiliza un adaptador PCF8574 que permite conectar la pantalla usando únicamente dos pines digitales a través del bus I2C [9].



Figura 1.8. Pantalla LCD I2C [9]

Software

- *Uniform Server*

El *Uniform Server* es un servidor que ofrece una solución de desarrollo *web*, que cuenta con un pre empaquetado, un servidor *web Apache*, base de datos *MySQL* y lenguaje de programación *PHP*, como se muestra en la Figura 1.9; es decir, es un paquete *Windows* + *Apache* + *MySQL* + *PHP* (*WAMP*). Este servidor cuenta con una instalación sencilla, también permite configurar de forma automática las distintas aplicaciones ya mencionadas; aparte de estas funciones, proporciona accesos de arranque y parada de servicios [10].

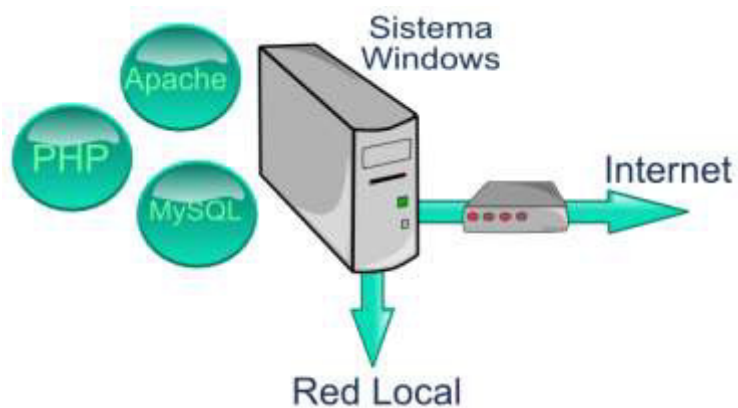


Figura 1.9 Servidor WAMP en un sistema de *Windows* [10]

El servidor WAMP permite instalar aplicaciones *web* que sean accesibles para la red local, tal como se muestra en la Figura 1.10, seleccionando el puerto 80 del *router*; es otra manera de acceder desde Internet. La gran parte de aplicaciones *web* que existen, necesitan de Apache + MySQL + PHP para ejecutarse de manera correcta [10].

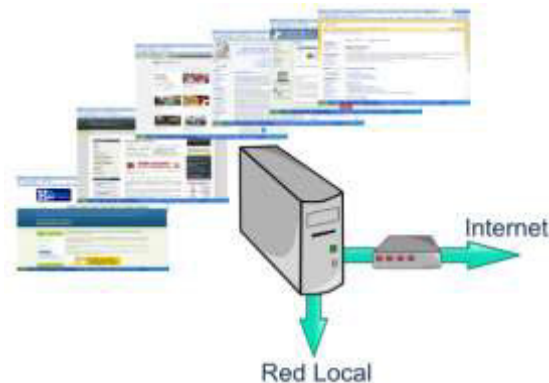


Figura 1.10 Servidor WAMP instalando aplicaciones *web* en red [10]

- **Software Serial**

La placa Arduino cuenta con diferentes métodos de comunicación, lo que le permite interactuar con diferentes dispositivos electrónicos; uno de estos métodos es la comunicación serial, la misma que utiliza *software* para replicar el funcionamiento de los pines seriales, de ahí el nombre "*Software Serial*". Otro método es la conexión inalámbrica por medio de *WiFi*, ya que brinda la opción de controlar aplicaciones móviles o de computadoras. Esta capacidad de conectarse con un computador, posibilita el envío de datos mediante Internet, que esté emparejado con alguna página *web* o que realice las tareas necesarias en el proyecto [11].

2 METODOLOGÍA

2.1 Descripción de la metodología usada

Según el diseño de la investigación, se usó una metodología experimental, debido a que se partió de la implementación de un prototipo para la detección en tiempo real del deslizamiento de tierra [12]. Una de las metas que se aspiró llegar con el proyecto y que la metodología experimental permitió alcanzar es, establecer la causa y el efecto del deslizamiento de tierra, mediante el monitoreo y control en el campo de la aplicación.

Una vez identificada tanto la causa como la consecuencia de un deslizamiento de tierra, se puede llevar a cabo el análisis del comportamiento de los dispositivos del prototipo dentro de la zona de aplicación, debido a que, en algunas zonas del Distrito Metropolitano de Quito (DMQ), existen viviendas construidas en lugares inadecuados, ya sean pendientes, laderas o suelos inestables, lo que las coloca en un alto riesgo al no ser un terreno apto para construir.

Con el propósito de identificar el problema, se realizó una investigación previa de las zonas en las que se tiene un registro de ocurrencia de deslizamientos de tierra dentro del DMQ, con el propósito de identificar el problema, y brindar una posible solución aplicable sin representar una afectación tanto para el suelo como para los habitantes aledaños de la zona.

La información obtenida contribuyó a comprender el comportamiento del suelo e identificar los distintos factores que contribuyen en un deslizamiento de tierra, para lograr prevenir a la población. Adicionalmente, se logró identificar los dispositivos necesarios a ser implementados, así como sus especificaciones de funcionamiento.

Para la construcción del sistema inteligente de alerta, se realizó una investigación previa a la selección de los posibles métodos de comunicación inalámbrica, que permitan tanto el monitoreo, como el envío de información del cambio del estado del terreno donde se produce el deslizamiento de tierra dentro del área del prototipo [12]. El sistema seleccionado se encarga de detectar en tiempo real la presencia del cambio de peso de la tierra en el suelo que sea propenso al deslizamiento de tierra, donde el método inalámbrico seleccionado envía la información que refleja el resultado del monitoreo.

El uso de sistemas de comunicación inalámbrica permite la comunicación entre distintos dispositivos móviles o fijos sin la necesidad de contar con un cableado. En este caso, la importancia de seleccionar los diferentes sensores que cuenten con la capacidad de interconectarse con otros elementos del sistema de manera inalámbrica pudieron ser implementados para la detección de deslizamientos de tierra.

Estos sensores además deben admitir el monitoreo en tiempo real de la variación de peso de la tierra, provocado por diversos factores climáticos que influyeron en el mismo, es decir: lluvia, humedad del suelo, temperatura, dirección y velocidad del viento [13].

Las características que se tomaron en cuenta para la tecnología que se implementó fueron:

- Red tolerante a fallas.

- Los datos recabados en los sensores, se deben guardar y ser consultados.
- Fácil expansión en diferentes lugares.
- Integración con otras tecnologías.
- Tener un bajo consumo energético.

Como parte del diseño, se consideró el desarrollo de los diagramas de flujo que permitieron anticipar las exigencias de los códigos de programación a ser desarrollados. Además, se realizó por medio de *software* un boceto que permita tener una visión más clara y amplia de la maqueta donde el prototipo será implementado. Por otra parte, se diseñó la placa PCB que se construyó.

La implementación se realizó en una maqueta previamente diseñada, la misma que permitió comprobar el funcionamiento del sistema. Se consideraron los parámetros necesarios de energización y comunicación inalámbrica. Se utilizó el módulo Arduino, así como el sensor previamente seleccionado, resultado del análisis correspondiente. Se implementó el código desarrollado en base a los diagramas de flujo diseñados, finalmente se fabricó la placa PCB que, en conjunto con la programación, detectó el deslizamiento de tierra y notificó lo ocurrido [14].

Una vez implementado el prototipo, se realizaron las pruebas necesarias de cada uno de los componentes que conforman el mismo, buscando de esta manera comprobar su funcionamiento; después, se procedió a evaluar el funcionamiento del sistema para identificar y corregir las posibles fallas que pudieran presentarse en la ejecución del mismo.

3 RESULTADOS Y DISCUSIÓN

3.1 Análisis de requerimientos técnicos del sistema

Para iniciar este proyecto, se realizó una investigación de los distintos registros de deslizamientos de tierra ocurridos dentro del área geográfica del DMQ; a partir de dicha investigación, se establecieron los dispositivos a utilizar. Dentro de los principales factores que provocan deslizamiento de tierra, se destacan las precipitaciones altas que se registran en la ciudad de Quito en la zona Este (Iñaquito e Izobamba); un segundo factor constituye el tipo de suelo de Quito, el cual está establecido sobre una serie de depósitos de materiales inestables, tales como cenizas volcánicas, que cuando se colma de agua o llega a secarse demasiado es propenso a perder su dureza [15] [16].

Dentro de las zonas de mayor inestabilidad del DMQ, se encuentran las Lomas de Ilumbisí, Puengasí, Guangüiltagua y San Isidro, cada una de ellas se representan como las zonas Q en el mapa de la Figura 3.1; estas zonas son susceptibles a la erosión, factores como el viento y el agua son la causa de fisuras, lo que da como resultado la pérdida de cohesión del suelo [17] [18].



Figura 3.1 Manejo del Riesgo Sísmico de Quito [18]

Otro factor que influye en los deslizamientos en la zona Este, es el tener una morfología irregular con un cono aluvial, debido a que en Quito se cuenta con una cantidad considerable de pendientes altas por efecto de su ubicación en la Cordillera de los Andes. La unión de todos estos factores, da como resultado los deslizamientos de tierra [19].

Entre el 24 de septiembre del 2019 hasta el 20 de enero del 2020, el total de reportes de emergencias fue de 126, de las cuales 59 fueron categorizadas como deslizamientos de tierra. El director metropolitano de Gestión de Riesgos, Francisco Ruiz, determinó que 77 zonas del área geográfica de la ciudad de Quito se encuentran en riesgo de deslizamiento; debido a esto, los habitantes que se encuentran aledaños a estas zonas conviven con un riesgo constante a ser víctimas de un deslizamiento de tierra [16].

Una vez comprendidos los diferentes factores y características geológicas que provocan los taludes, se realizó un análisis que dio como resultado la definición del tipo de sensor necesario que permita determinar el cambio del nivel de tierra, así como el sistema inalámbrico que cumpla con los requisitos establecidos para el envío de los datos que posteriormente van a ser registrados en la base de datos, y la sirena. La implementación

de todos los dispositivos que formarán parte del sistema de alerta temprana tienen como su principal objetivo alertar a la población que se encuentre dentro del área geográfica del evento.

Considerando que el tipo de suelo de la zonificación realizada dentro del DMQ se encuentra en la zona llamada Cangagua, da como resultado un tipo de deslizamiento lento; por este motivo, se selecciona un sensor de peso en la parte superior de la montaña, de tal manera que las personas puedan ser advertidas y cuenten con un tiempo que les permita resguardar su vida, reduciendo en cierta manera los daños producidos por el mismo.

Así también se determinó que, para establecer la comunicación inalámbrica, se usará un módulo *WiFi*, debido a que presenta una serie de ventajas comparado con otras tecnologías inalámbricas como *Bluetooth* y *Zigbee*, siendo el alcance el factor primordial para el uso de este módulo. Lo que permite la transmisión de datos sin la necesidad de utilizar una red cableada que realice el envío a la base de datos, donde se mantendrá el registro continuo de los eventos que ocurran en el deslizamiento de tierra.

Se determinaron los dispositivos principales como el sensor de peso (celda de carga, galga extensiométrica y amplificador HX711) y los elementos complementarios, tales como el módulo Arduino Uno, la sirena, el LCD i2c y la alimentación necesaria para el sistema. En la Figura 3.2 se muestran los elementos que conforman el sistema.

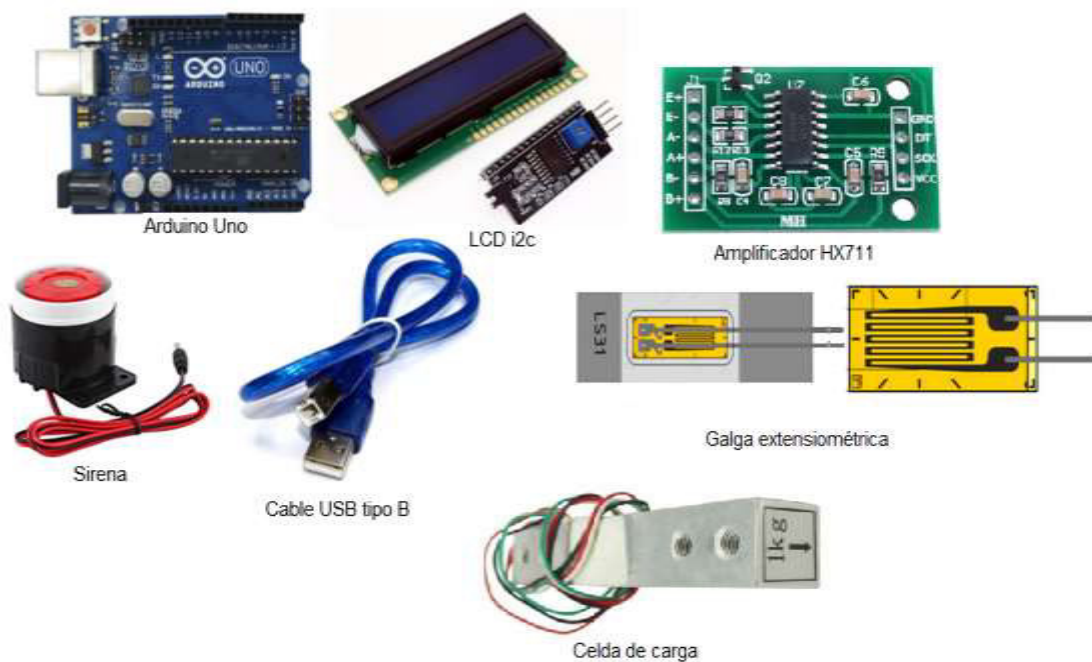


Figura 3.2 Elementos utilizados en el sistema [5] [6] [7] [8] [9] [20]

La adquisición de los dispositivos que forman parte del sistema de alerta temprana se efectuó después de determinar el tipo de suelo y la forma como se genera el movimiento de masa, además del sistema inalámbrico que ayuda con el envío de los datos recabados para ser consultados y monitoreados por la persona que se encuentre encargada. De esta manera se garantiza una respuesta temprana a los organismos encargados de atender estas emergencias.

3.2 Diseño del sistema

Tomando en cuenta cada uno de los dispositivos seleccionados, se procedió con el diseño del sistema para la detección de deslizamiento de tierra; para la detección de la pérdida de peso en el sensor, se realizó la calibración para que el sistema detecte el cambio dentro del rango establecido en el código de Arduino. La calibración se realizó en base al flujograma que se muestra en la Figura 3.3. De acuerdo con el flujograma, el código inicia con la declaración de las variables que posteriormente facilitarían al procesamiento de la señal de reloj y dé la información enviada desde el sensor de peso. Luego, se inicia la comunicación serial y se llama a las variables declaradas anteriormente. A continuación, se presenta en el monitor serial de Arduino IDE el mensaje de lectura del valor de la balanza y otro que indica que no se coloque ningún objeto sobre la balanza; este procedimiento es necesario para la calibración del sensor. Con el proceso de calibración, se estimó el peso inicial de 150 (g), que aplicado a la celda de carga lo interpreta como un valor de -582 180. Después, se establece un valor de tara de 20, que corresponde a un indicador necesario para la interpretación en el amplificador. A continuación, en el monitor serial se muestra un mensaje que está lista para pesar, procediendo a imprimir el peso.

Siguiendo este sistema, el sensor trabaja en la detección desde un rango de 150 (g), tomando en cuenta la cantidad de peso que soporta la maqueta; se establecieron tres posibles estados, dependiendo de la cantidad de tierra que se pierde:

Mayor a 150 Sin peligro

149 – 50 Deslizamiento Moderado

49 – 0 Evacuar

El sensor de peso está conectado a una sirena que emite una alerta de sonido cuando detecte un cambio del peso dentro de los rangos anteriormente mencionados; una vez

registrada la clase de riesgo, se transmite el dato al módulo *WiFi* ESP versión 3; el módulo se conecta a un servidor *WAM*, que enlaza el dato receptado con una base de datos de *MySQL* mediante un código *PHP* que envía el mismo dato por el método *POST*.

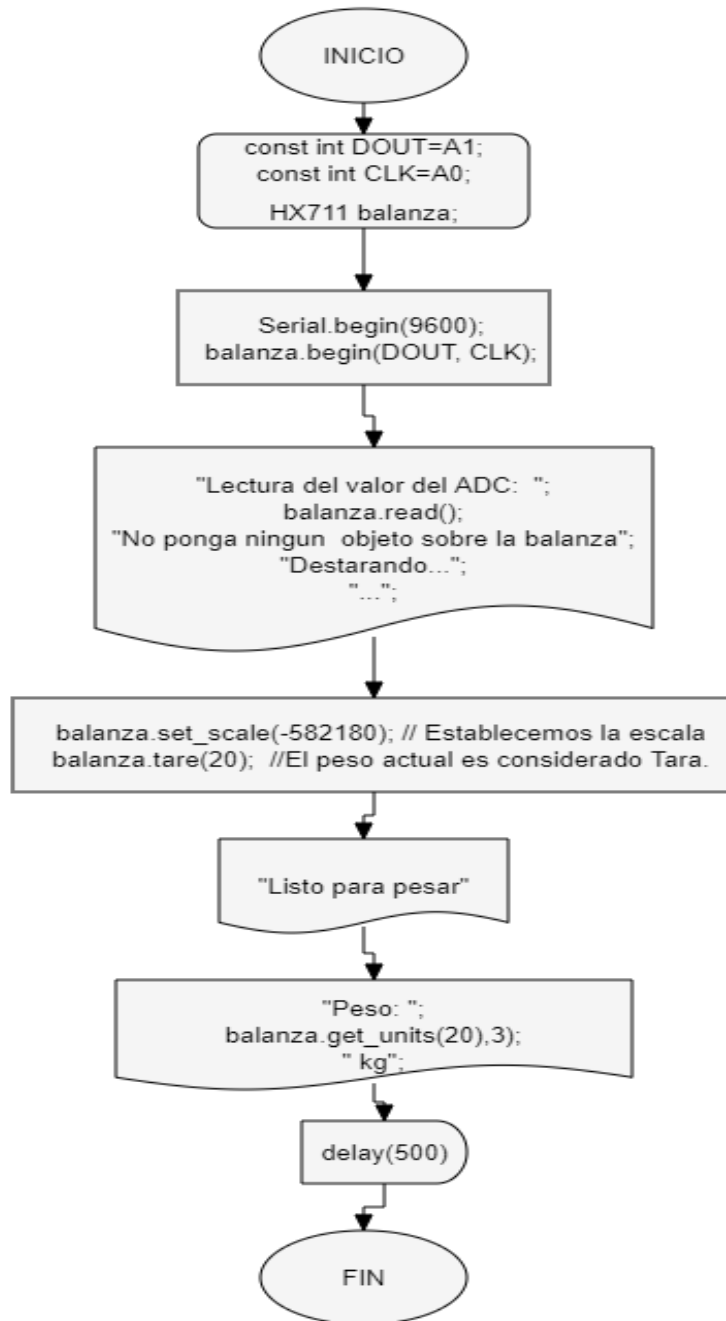


Figura 3.3 Flujograma del código de calibración

Inicialmente se optó por utilizar un sensor de movimiento HC-SR501, el cual se lo podía calibrar directamente desde sus perillas exteriores, este sensor conto con una gran desventaja, ya que, generaba una cantidad muy elevada de falsas alarmas cada vez que se cortaba su haz, ya sea por la intervención humana o factores externos (animales). Por este motivo se optó por utilizar un sensor de peso, cuyo parámetro permite identificar una variación en la zona en la que se lo ubica, siendo dicha variación la perdida peso en la cumbre, lo que se interpreta como un deslizamiento de tierra.

Método de petición *HTTP POST*

El método *POST* establece las medidas para las peticiones *HTTP* al servidor; es por este motivo que los datos no son visibles para el usuario, por lo cual es el adecuado para transmitir información sensible. Este método no tiene límite de cantidad para enviar información; además, permite el envío de fotos y video [21].

Utilizando este método, se realizó el código *PHP* empleando la aplicación *Sublime Text*; la estructura del código se puede observar en la Figura 3.4, en donde se inicia con una petición *mysql_connect* para conectarse al servidor local (*localhost*), incluyendo su contraseña y nombre de usuario (*root*), posteriormente se realiza la conexión a la base de datos "peso" llamando a la petición conexión, se especifica el tipo de datos a enviar (*utf8*) para empezar con la declaración de las variables, se define el método de envío de información (*POST*). Seguidamente, se realiza una petición (*mysql_query*) para poder insertar dentro de las columnas *id*, *fecha*, *riesgo*, *peso*; de la base de datos "peso" y en la tabla con el mismo nombre, se cierra la petición (*mysql_close*) y se manda a imprimir un mensaje que diga "Datos ingresados correctamente", esta estructura es la que permite la conexión entre el dato recibido por el sensor de peso, transferido al módulo *WiFi* y enviado a la base de datos.

```
1 <?php
2 $conexion = mysql_connect("localhost", "root", "root");
3 mysql_select_db("peso", $conexion);
4 mysql_query("SET NAMES 'utf8'");
5
6 $chipid = $_POST ['chipid'];
7 $riesgo = $_POST ['riesgo'];
8 $peso = $_POST ['peso'];
9
10 mysql_query("INSERT INTO 'peso`.`tabla` ('id', 'chipId', 'fecha', 'riesgo', '
    peso`) VALUES (NULL, '$chipid', CURRENT_TIMESTAMP, '$riesgo', '$peso');");
11
12 mysql_close();
13
14 echo "Datos ingresados correctamente.";
15 ?>
```

Figura 3.4 Código *PHP*

Diseño de método de deslizamiento de tierra

Para simular el deslizamiento de tierra, se utilizaron dos motores de vibración, ubicados estratégicamente dentro de la maqueta, como se muestra en la Figura 3.5. Una vez que se enciende el sistema, la vibración de los motores genera el deslizamiento de tierra, la cual es esparcida sobre la maqueta.

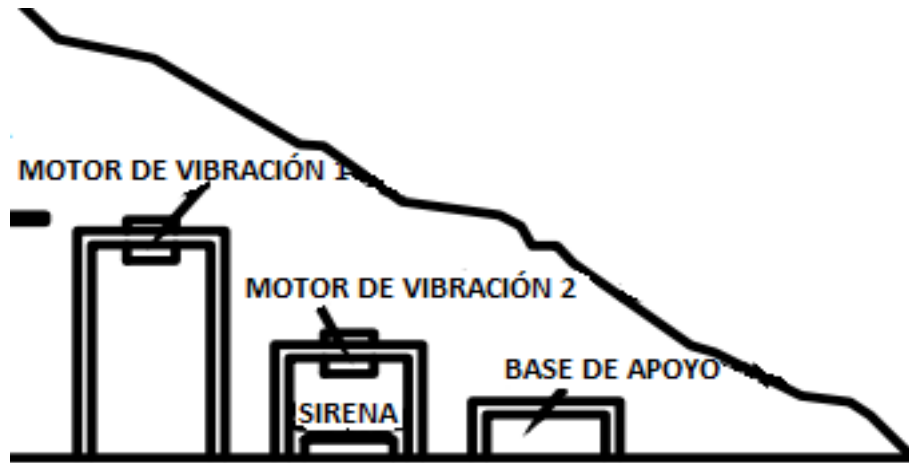


Figura 3.5 Ubicación de los motores de vibración en la maqueta

Distribución de pines de la placa Arduino

La distribución de pines que se usó en la placa Arduino Uno, se presenta en la Figura 3.6. Como se observa, se utilizaron dos pines analógicos del Arduino para conectar al amplificador, desde el cual se implementa el sensor de peso. Además, se conectó la galga a la alimentación de 5 (V) y a tierra. Para la conexión del LCD se utilizaron dos pines analógicos, además de su respectiva alimentación y tierra. La sirena y el LED, elementos que permiten identificar auditiva y visualmente la alerta, fueron conectados a los puentes.

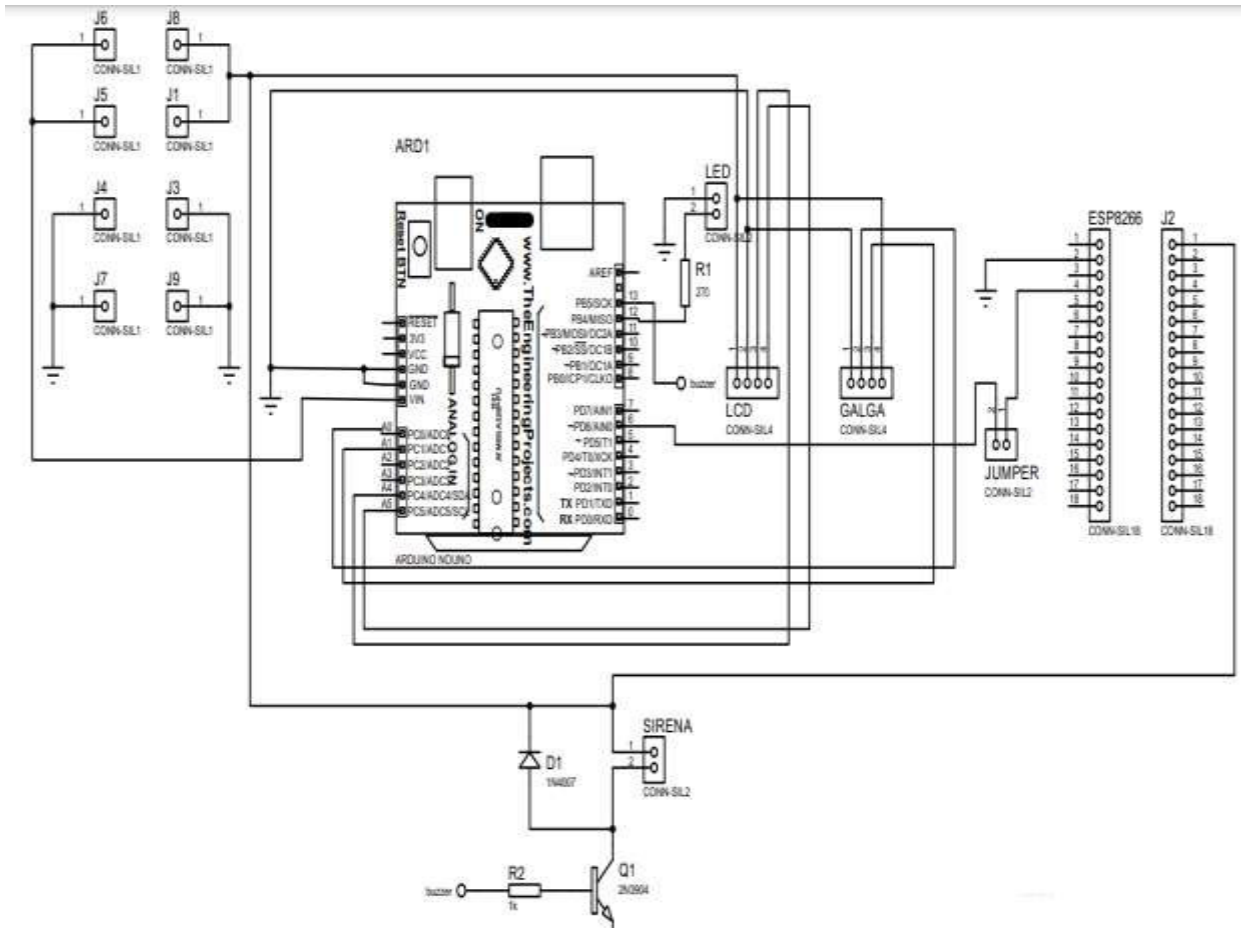


Figura 3.6 Distribución de pines de la placa Arduino Uno

Flujograma del código de placa Arduino

El código realizado para el funcionamiento de la placa Arduino se presenta en la Figura 3.7. como se puede observar, el código inicia con la definición de las variables; posteriormente, se establece el modo de los pines del LED, sirena y balanza. Luego, se muestra un mensaje en el monitor serial de no colocar ningún objeto sobre la balanza. A continuación, se utilizan la escala y tara definidas en el proceso de la calibración para mostrar en el LCD y el monitor serial el mensaje de “Listo”, indicando que se puede iniciar con el proceso de detección de peso. Posteriormente, se establecen tres condiciones: si la lectura del peso es mayor o igual a 150 (g) se imprime en el LCD el mensaje de “Sin peligro”; si se cumple la segunda condición en que la lectura del peso es menor a 150 (g) y mayor o igual a 50 (g), se imprime el mensaje “Deslizamiento moderado”. Finalmente, si el peso es menor a 50 (g), se imprime el mensaje “Evacuar”. En los tres casos, se envía la información correspondiente a la base de datos, la cual es actualizada automáticamente en tiempo real.

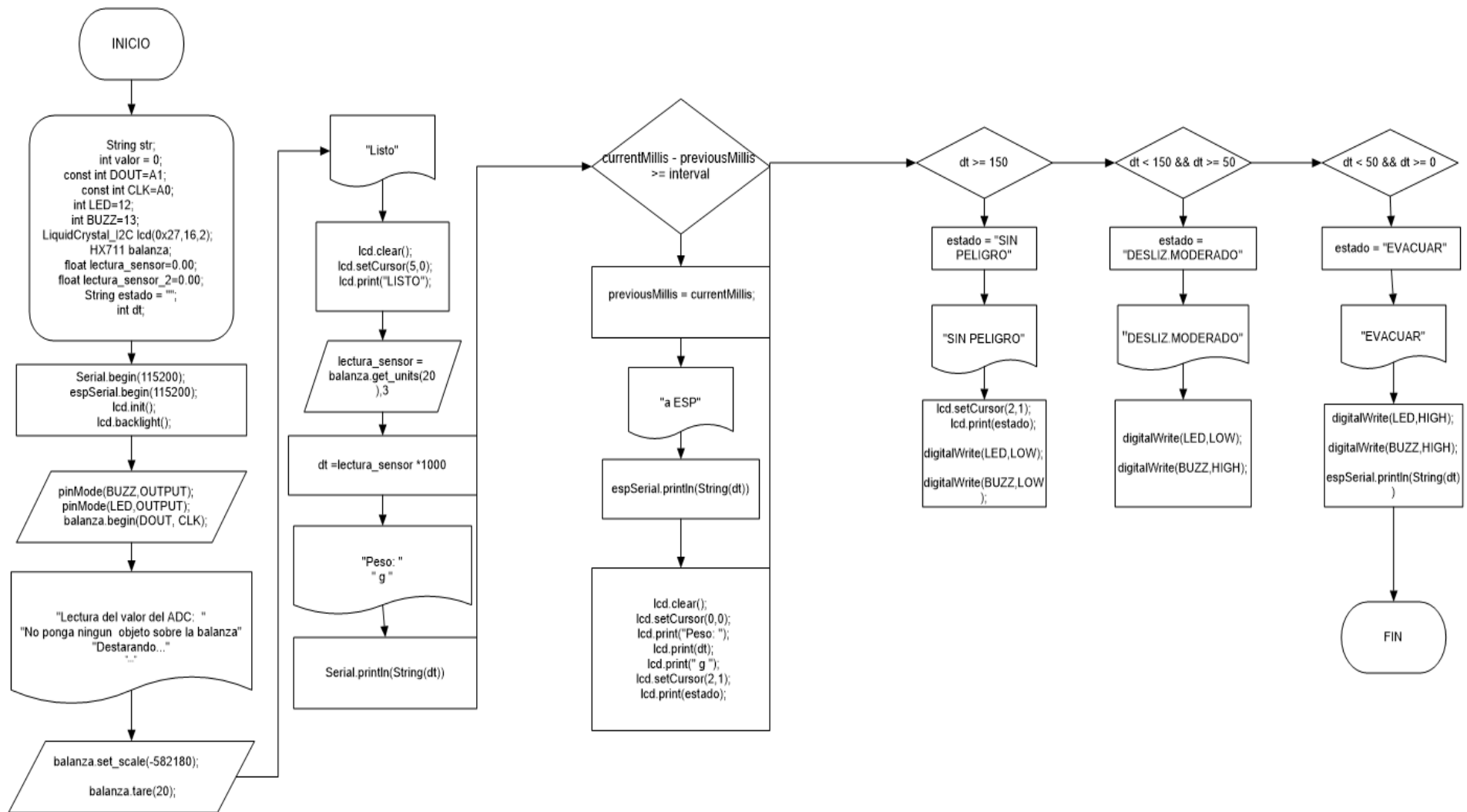


Figura 3.7 Flujograma de código de placa Arduino

Distribución de pines módulo *WiFi*

El módulo ESP v.3 cuenta con un total de 10 pines digitales, de los cuales se tomaron en cuenta el pin digital 10 conectado a un *jumper* y el pin 16 para la conexión serial con el Arduino Uno; a su vez, se utilizaron los pines de alimentación y el pin de tierra, que se muestran en la Figura 3.8.

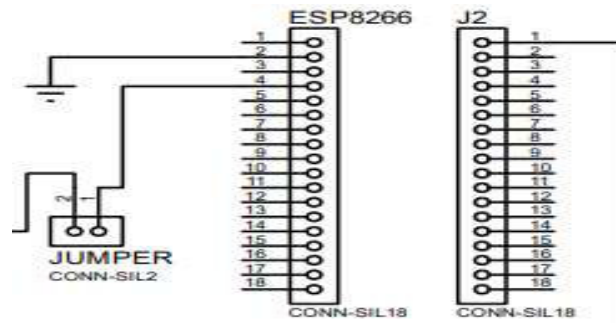


Figura 3.8 Distribución de pines del módulo *WiFi*

Flujograma del código del módulo *WiFi*

El módulo *WiFi* es el que proporciona la característica de la comunicación inalámbrica al sistema; el código inicia con la definición de las variables, entre ellas el *SSID*, y de la red, contraseña y dirección IP. Luego, se establece como puerto de conexión el puerto 80; en caso de no llegar a conectarse, se va a imprimir un mensaje en el monitor serial que indica un “fallo de conexión”. Posteriormente, se inicia la conexión a la red *WiFi* y envía los datos por el método *POST*; una vez establecida la conexión, se imprime un mensaje en el monitor serial que dice “Enviando datos a SQL”, si el tiempo de conexión sobrepasa los 50 (mseg) aparecerá un nuevo mensaje que dice “cliente fuera de tiempo”, caso contrario continuará con un mensaje que dice “línea”. Después, se inicia con la conexión serial, se imprime el *chip ID* del módulo ESP, se inicia la conexión *WiFi*; una vez conectado, aparecerá un mensaje en monitor serial que dice “*WiFi* conectado”. Posteriormente, se abre una sentencia *if* en donde se va a recibir el dato enviado por el Arduino Uno, se transforma en entero y pasa a ser comparado con tres condiciones: la primera si el valor es mayor o igual a 150 (g) “Sin peligro”, la segunda si el valor es menor a 150 (g) y mayor o igual a 50 (g) “Deslizamiento moderado” y la tercera cuando el valor es menor a 50 (g) “Evacuar”; en cada una de las condiciones, se llama a la función del archivo PHP y se envía la información a la base de datos. El flujograma del código se puede observar en la Figura 3.9.

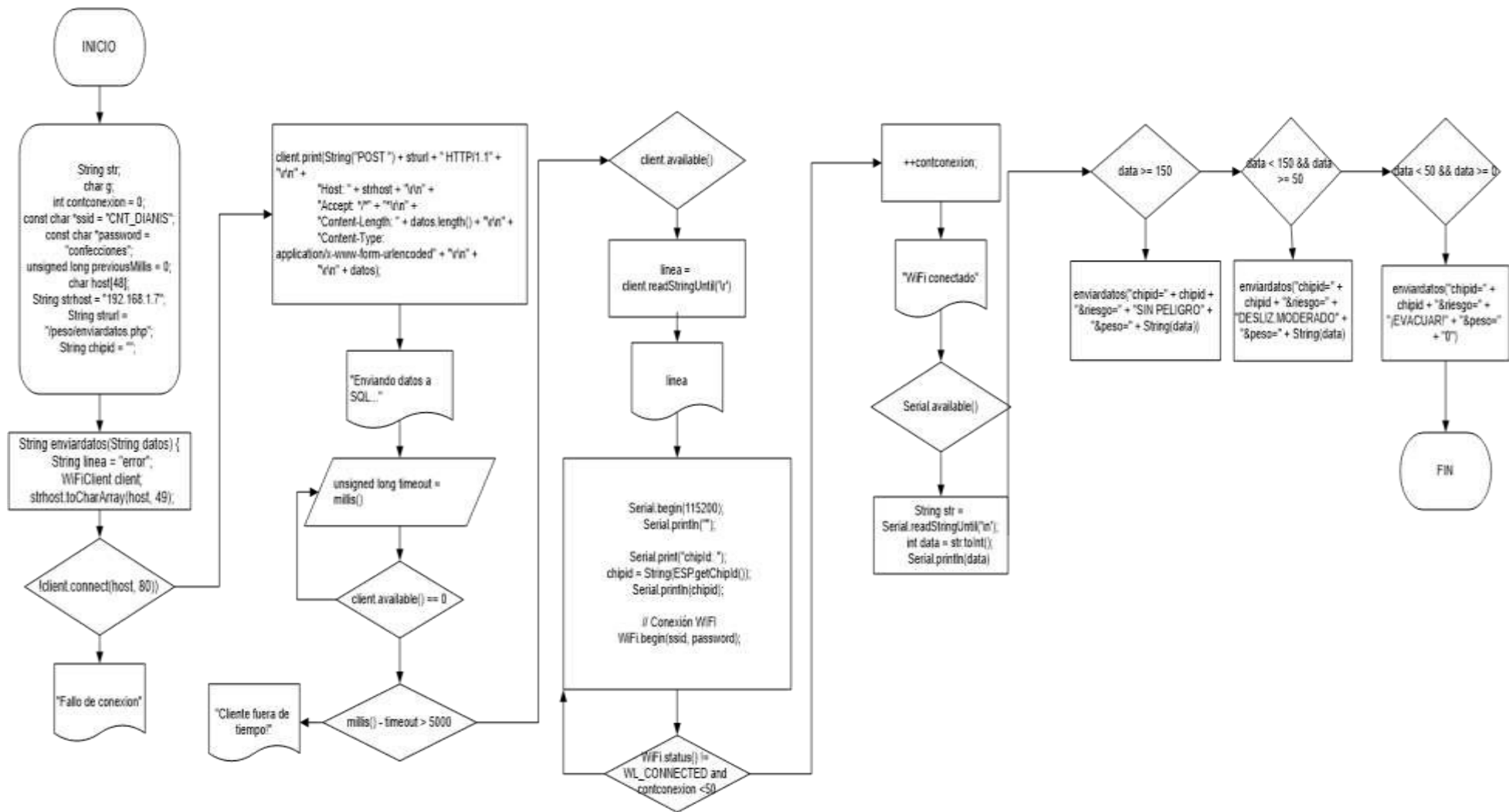


Figura 3.9 Flujograma del código del módulo *WiFi*

Fuente de alimentación

La fuente de alimentación es uno de los componentes más importantes de un circuito electrónico; en el caso del sistema para detección de deslizamiento de tierra, se tomó en cuenta un total de 3 fuentes de alimentación, cuya función se muestra en la Tabla 1.1.

Tabla 1.1 Especificación de uso de fuentes de voltaje

Fuente	Voltaje	Uso
Fuente 1	9 (V _{DC})	Alimentación de motor de vibración 1
Fuente 2	12 (V _{DC})	Alimentación de motor de vibración 2
Fuente 3	9 (V _{DC})	Alimentación de placa PCB

Se realizó el diseño de los circuitos de alimentación para los dos motores de vibración en el *software* Autocad, como se puede observar en la Figura 3.10.

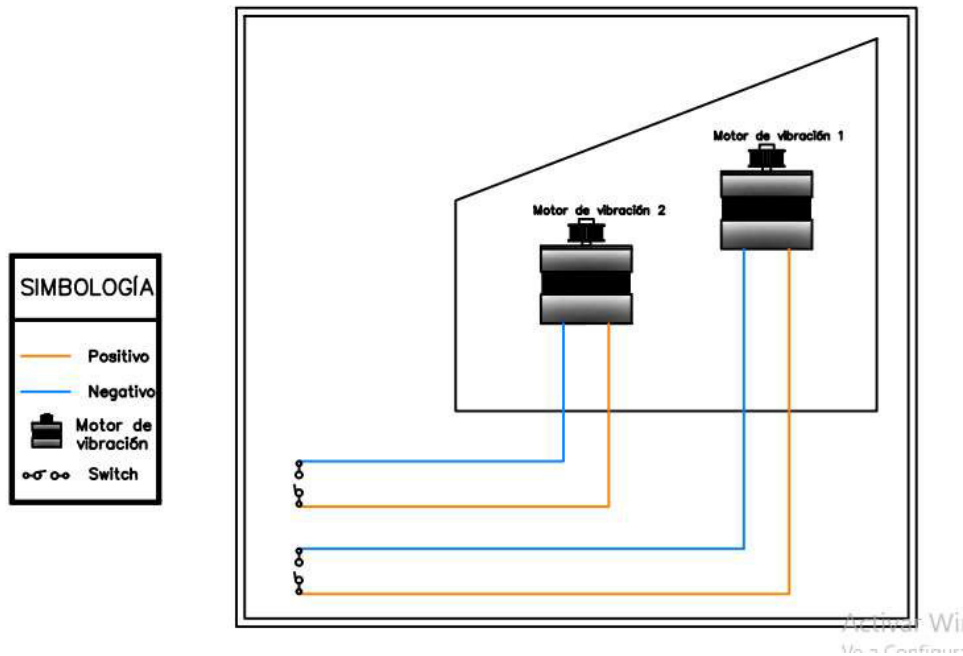


Figura 3.10 Circuito de alimentación para motores de vibración

Diseño de la maqueta

La maqueta se construyó tomando en cuenta un tipo de terreno que sea propenso a producirse un deslizamiento de tierra, es por este motivo que se utilizó la herramienta *Paint 3D*, como se muestra en la Figura 3.11.



Figura 3.11 Diseño de la maqueta en 3D

Una vez realizado el diseño de la maqueta, se determinó la ubicación de los componentes de la placa PCB y los motores de vibración; usando el *software* Autocad se muestran las distintas vistas de la maqueta en la Figura 3.12, así como la ubicación de los componentes dentro de la maqueta como se puede observar en la Figura 3.13 y Figura 3.14.

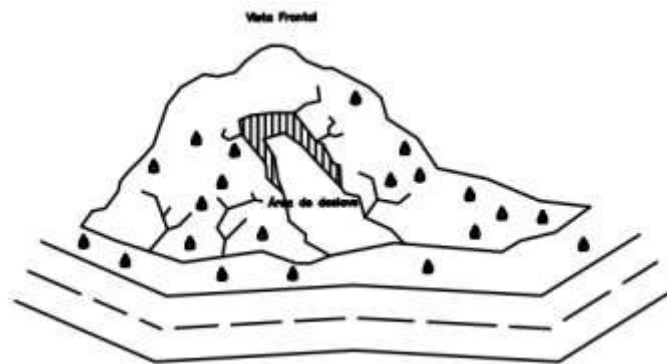


Figura 3.12 Vista frontal de la maqueta



Figura 3.13 Vista lateral para sensor de peso, motores de vibración y bases

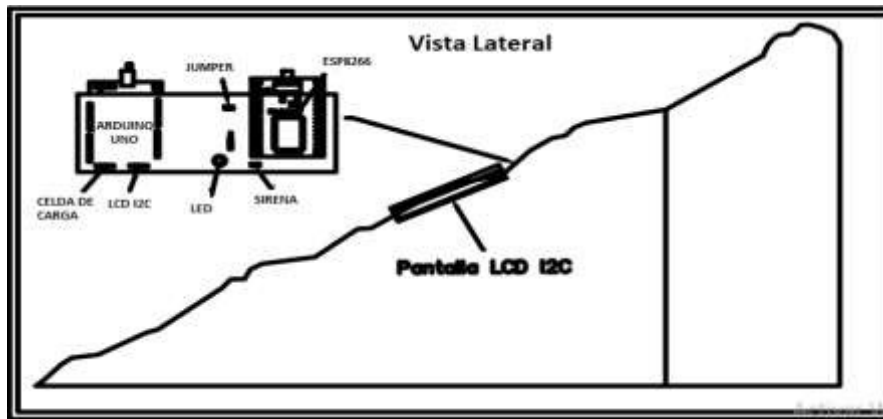


Figura 3.14 Vista lateral para placa PCB y pantalla LCD I2C

Cálculo de valores de los dispositivos

Para evitar caídas de voltaje y, como mecanismo de protección del LED y sirena, se realizó el cálculo respectivo de resistencias; cabe recalcar que para el módulo ESP8266 no se requirió de un cálculo adicional debido a que este cuenta con un regulador interno. A continuación, se presenta el cálculo de la resistencia para el LED:

$$R = \frac{V}{I}$$

Ecuación 3.1 Ley de Ohm [22]

Donde:

- V : 5 (v) Voltaje
- I : 20 (mA) Corriente
- R : (Ω) Resistencia

Usando la **Ecuación 3.1**, se obtiene:

$$R = 250 (\Omega)$$

Refiriéndonos a valores comerciales de resistencias [23], se utilizó una resistencia:

$$R = 270 (\Omega)$$

A continuación, se presenta el cálculo de la resistencia para la sirena. Empleando la ecuación 3.1:

Donde:

- V : 17 (v) Voltaje
- I : 20 (mA) Corriente
- R : (Ω) resistencia

Usando la **Ecuación 3.1** se obtiene:

$$R = 850 (\Omega)$$

Considerando valores comerciales de resistencias [23], se utilizó una resistencia:

$$R = 1000 (\Omega)$$

Diseño de la placa PCB

Una vez comprobado el funcionamiento de los elementos que conforman el sistema en una *protoboard*, se continuó con el diseño en el *software Proteus*, como se muestra en las Figura 3.15, Figura 3.16 y Figura 3.17.

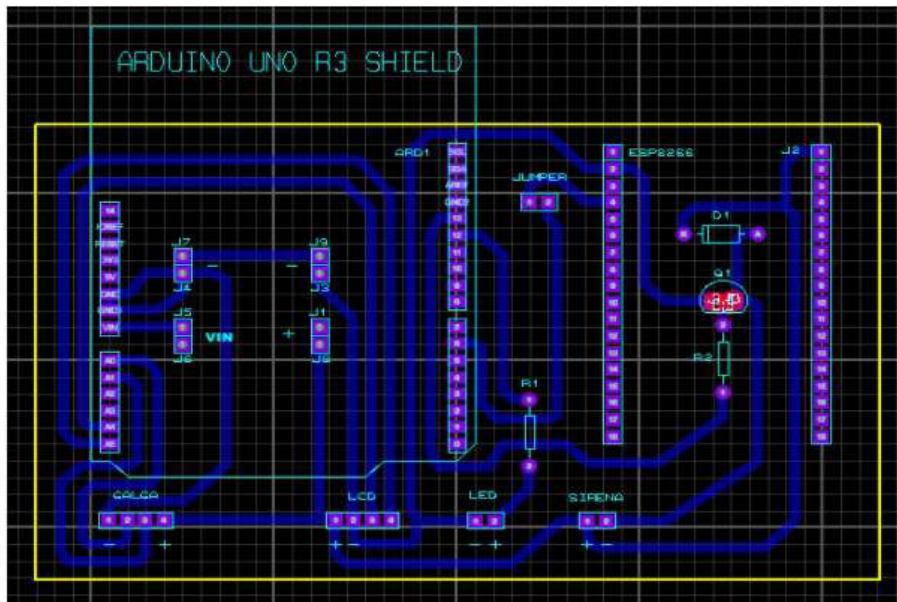


Figura 3.15 Diseño en *Proteus* de la placa PCB

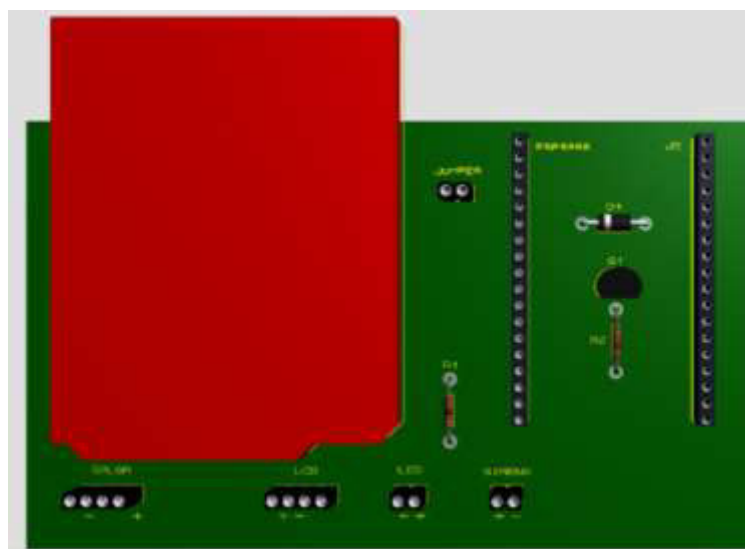


Figura 3.16 Parte superior de la placa PCB

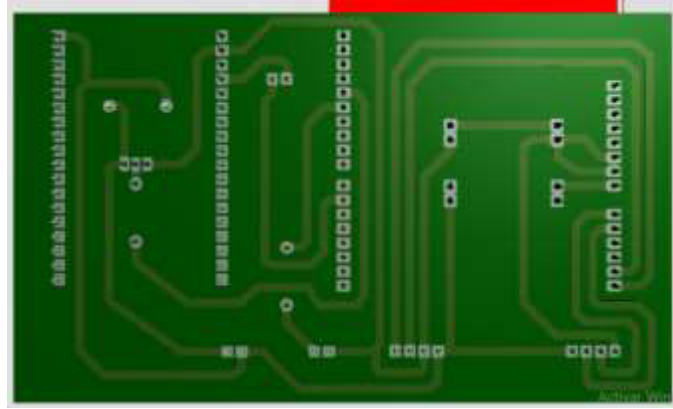


Figura 3.17 Parte inferior de la placa PCB

3.3 Implementación del sistema

Después de realizar el diseño de cada una de las partes del sistema para la detección de deslizamiento, se procedió a la realización de la maqueta y la placa PCB.

Maqueta

Se construyó una base de madera de 1,13 (m) de alto por 0,95 (m) de ancho, como se muestra en la Figura 3.18.



Figura 3.18 Base de madera para la maqueta

Utilizando materiales reciclados, se establecieron las bases para las elevaciones de la maqueta, como se puede observar en la Figura 3.19.



Figura 3.19 Bases para la construcción de la maqueta

Posteriormente, se procedió a rellenar los espacios vacíos de la maqueta, como se puede observar en la Figura 3.20.



Figura 3.20 Relleno de la maqueta

Después, se completó la elaboración con las casas fabricadas con madera de balsa, y se estableció el espacio en donde se ubicaron los componentes del sistema, como se muestra en la Figura 3.21.



Figura 3.21 Avance de la construcción de la maqueta

Posteriormente, se completó el relleno y se colocó una lata que facilite el deslizamiento de tierra; se procedió a blanquear la maqueta para garantizar un proceso adecuado de pintado, como se puede observar en la Figura 3.22 y Figura 3.23.



Figura 3.22 Maqueta estructurada.



Figura 3.23 Proceso de blanqueamiento de la maqueta.

Se construyeron las bases de apoyo en donde se ubicaron los motores de vibración, y se colocaron en la maqueta como se puede observar en la Figura 3.24; además, se realizaron las conexiones de los circuitos de alimentación para los dos motores por debajo de la maqueta, como se muestra en la Figura 3.25.



Figura 3.24 Motores de vibración en las bases de apoyo



Figura 3.25 Circuito de alimentación de motores de vibración

Después, se colocó la placa PCB en la maqueta, como se puede observar en las Figura 3.26 y Figura 3.27.



Figura 3.26 Vista de la ubicación del sensor de peso



Figura 3.27 Ubicación de la placa PCB en la maqueta

Finalmente, con todos los elementos ubicados en la maqueta, se procedió a pintar la misma, dando como resultado el aspecto final que se puede observar en la Figura 3.28.



Figura 3.28 Aspecto final de la maqueta

Placa PCB

Se exportó el circuito para posteriormente imprimirlo en papel transferible; la impresión debe realizarse a láser debido a que permitirá que la tinta no se borre en el instante de tener contacto con el calor de la plancha. Con el circuito ya impreso, se procede a limpiar la placa de cobre con una malla fina de acero, quitando todas las posibles impurezas que puede tener al estar en contacto con las manos o adhesivos, como se muestra en la Figura 3.29.



Figura 3.29 Limpieza de la baquelita

Se coloca el circuito impreso sobre la baquelita para luego ser llevado a una plancha – estampadora, con una temperatura de 155 (°C), dejándola de 1 a 3 (min), como se muestra en la Figura 3.30 y Figura 3.31.



Figura 3.30 Estampadora a 155 (°C)



Figura 3.31 Estampado de 1 a 3 (min) en la estampadora

Después de haber estado en contacto con el calor de la plancha, se deja enfriar la placa para retirar los excedentes de papel que hayan quedado en la placa, como se muestra en la Figura 3.32; posteriormente, se sumerge a la placa en ácido férrico, como se observa en la Figura 3.33, para retirar todo el cobre que no pertenezca a las pistas impresas.



Figura 3.32 Placa retirada de la plancha



Figura 3.33 Placa sumergida en ácido férrico

Después de sacar la placa del ácido férrico, se limpia la tinta que está presente en las pistas, frotando hasta que todas las pistas queden visibles sin excedentes de tinta, dando como resultado la placa que se puede observar en la Figura 3.34.



Figura 3.34 Placa pulida

Una vez quitado todo el exceso de tinta de las pistas, se procede a taladrar cada uno de los espacios donde se ubicarán los espadines y los elementos de la placa, como se puede ver en la Figura 3.35.



Figura 3.35 Perforación de la placa

Una vez taladrada la placa, se procede a soldar todos los espadines para ubicar la placa Arduino Uno, y el módulo *WiFi* en de la placa PCB, como se pueden observar en las Figura 3.36 y Figura 3.37.



Figura 3.36 Ubicación de Arduino en placa PCB



Figura 3.37 Ubicación del módulo *WiFi* en la placa PCB

Finalmente, se conectan todos los componentes a la placa PCB, dando como resultado lo que se puede observar en la Figura 3.38.

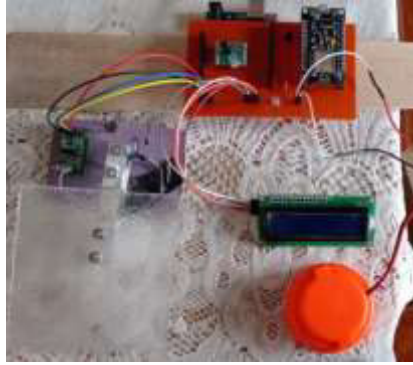


Figura 3.38 Placa PCB con componentes conectados

3.4 Pruebas y Análisis de Resultados

Para la realización de las pruebas, se dividió el funcionamiento del sistema de detección de deslizamiento de tierra en tres partes: comprobación de motores, funcionamiento de LCD I2C y envío de información a base de datos.

Comprobación de los motores

1. **Razón de la prueba:** la prueba se realizará para verificar que el deslizamiento de tierra se efectúe de manera automática, por medio del funcionamiento de los motores de vibración.
2. **Descripción:** se colocaron los motores dentro de la maqueta, uno en la parte superior y otro en la inferior, para que provoque el deslizamiento de tierra solamente con encenderlos a través de un *switch* lo que se visualiza en la Figura 3.40; la tierra se ubicó en la parte superior de la maqueta como se ve en la Figura 3.39 y los motores facilitan su deslizamiento hacia abajo tal y como se indica en la Figura 3.41.



Figura 3.39 Ubicación inicial de la tierra



Figura 3.40 Switches de encendido de motores



Figura 3.41 Caída progresiva de la tierra

3. **Comentarios:** al ser motores de vibración, provocan la caída de tierra de forma continua al tener un contacto directo con la lata donde se colocó la tierra.
4. **Correcciones:** se corrigió la inclinación que tenía inicialmente la lata, debido a que el deslizamiento se producía por acción de la gravedad y no exclusivamente de los motores.

Funcionamiento de LCD I2C

1. **Razón de la prueba:** esta prueba permitió comprobar que el dato del peso receptado por el sensor, se refleje en el LCD, así como el mensaje que se asignó en el código de Arduino.
2. **Descripción:** mientras se produce el deslizamiento de tierra, la información del cambio del peso receptado por el sensor aparecerá en el LCD, así como el tipo de deslizamiento como se muestran en las Figura 3.42, Figura 3.43, Figura 3.44 y Figura 3.45.

```
Peso: 176 g
176
NORMAL
Peso: 175 g
175
NORMAL
Peso: 175 g
175
NORMAL
Peso: 177 g
177
NORMAL
Peso: 176 g
```

Figura 3.42 Peso detectado por el sensor de peso



Figura 3.43 Peso reflejado en el LCD

```
DESLIZ.MODERADO
Peso: 59 g
59
DESLIZ.MODERADO
Peso: 59 g
59
DESLIZ.MODERADO
Peso: 59 g
59
a ESP
DESLIZ.MODERADO
Peso: 60 g
60
DESLIZ.MODERADO
Peso: 59 g
--
```

Figura 3.44 Cambio de peso en el sensor

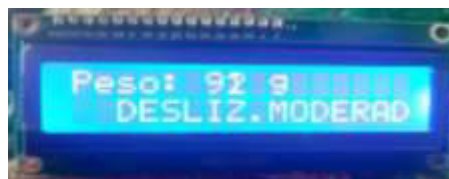


Figura 3.45 Cambio de peso reflejado en el LCD

3. **Comentarios:** se puede observar que el LCD refleja el cambio en el peso y el mensaje que se envía, a la par con el cambio que detecta el sensor de peso.

Envío de información a base de datos

1. **Razón de la prueba:** esta prueba se realizó con el fin de comprobar el correcto envío de información obtenida por el módulo *WiFi*.

2. **Descripción:** mientras se realiza el deslizamiento de tierra y se presenta el cambio en el peso que detecta el sensor mostrado en la Figura 3.46, cuando el cambio en el peso entra en los rangos especificados en el código, envía a la base de datos el tipo de deslizamiento como se muestra en la Figura 3.47, la hora y fecha del deslizamiento, así como la cantidad en (kg) de la tierra perdida.

```
Datos ingresados correctamente.
175
176
Enviando datos a SQL...
Datos ingresados correctamente.
176
176
Enviando datos a SQL...
Datos ingresados correctamente.
```

Figura 3.46 Envío exitoso de información por el módulo *WiFi*

	id	chipid	fecha	riesgo	peso
<input type="checkbox"/> Edit Copy Delete	7090	2342306	2021-09-16 09:25:17	SIN PELIGRO	175
<input type="checkbox"/> Edit Copy Delete	7089	2342306	2021-09-16 09:25:11	SIN PELIGRO	176
<input type="checkbox"/> Edit Copy Delete	7088	2342306	2021-09-16 09:25:04	SIN PELIGRO	176
<input type="checkbox"/> Edit Copy Delete	7087	2342306	2021-09-16 09:24:58	SIN PELIGRO	176
<input type="checkbox"/> Edit Copy Delete	7086	2342306	2021-09-16 09:24:52	SIN PELIGRO	175
<input type="checkbox"/> Edit Copy Delete	7085	2342306	2021-09-16 09:24:46	SIN PELIGRO	174
<input type="checkbox"/> Edit Copy Delete	7084	2342306	2021-09-16 09:24:40	SIN PELIGRO	176
<input type="checkbox"/> Edit Copy Delete	7083	2342306	2021-09-16 09:24:34	SIN PELIGRO	176
<input type="checkbox"/> Edit Copy Delete	7082	2342306	2021-09-16 09:24:28	SIN PELIGRO	177
<input type="checkbox"/> Edit Copy Delete	7081	2342306	2021-09-16 09:24:22	SIN PELIGRO	176

Figura 3.47 Verificación del envío de información en la base de datos

3. **Comentarios:** como se puede observar, el registro de información obtenida por el sensor de peso, es correctamente enviada por medio de *WiFi* a la base de datos.

La verificación del funcionamiento del sistema se puede evidenciar a través del video asociado al código QR presentado en la Figura 3.48.



https://youtu.be/2rbhxR_37Us

Figura 3.48. Verificación de funcionamiento del sistema

3.5 Manual de Usuario

El manual de usuario se puede observar a través del acceso al código QR que se puede apreciar en la Figura 3.49.



<https://youtu.be/u7ypmOnsJYE>

Figura 3.49. Manual de Usuario

4 CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones

- Se identificaron las causas que provocan un deslizamiento de tierra, tales como la erosión, suelo aluvial, lluvia, sedimentos; así como las consecuencias que tiene el mismo dentro de un área geográfica en la que debido a su estructura es propenso a que ocurra este tipo de catástrofe natural. Por este motivo, surge la necesidad de buscar un método para mitigar en lo posible los daños provocados;

tomando en cuenta este principio, se implementó un prototipo de un sistema que esté en la capacidad de no solo detectar el momento en que se produce el deslizamiento, sino también de poder alertar a la población que se encuentre cercana al área del percance, buscando de esta manera en el mejor caso reducir el impacto del mismo.

- Se procedió a realizar un estudio de los dispositivos necesarios para cumplir con las necesidades que exige el prototipo, es así como se determinó el uso de un sensor de peso, así como de los elementos complementarios del sistema. Dentro de estos dispositivos se añadió el uso del método de comunicación inalámbrica que envíe los eventos a un sistema de registro, por el cual se va a poder mantener un monitoreo constante del área propensa a que ocurran este tipo de desastres naturales.
- Los códigos diseñados dan las instrucciones a cada uno de los componentes del sistema de cómo deben interactuar entre sí, tomando en cuenta las funciones necesarias para enlazar el código del microcontrolador con el módulo ESP8266. Además, es importante la forma de conexión necesaria para su comunicación, la cual se logra a través de una conexión serial entre el módulo *WiFi* y el microcontrolador Arduino Uno.
- El diseño de la placa PCB fue realizado tomando en cuenta los voltajes que necesitan los elementos del prototipo para su adecuado funcionamiento, ya que, si el voltaje que reciben no está dentro de su umbral, el sistema no cumplirá con su objetivo, así como el diseño de la maqueta en donde será implementado el sistema determinando la ubicación estratégica de cada componente del sistema. Todo ello, permitió la detección del cambio producido durante el deslizamiento de tierra desde su inicio en la parte superior hasta su fin en la parte inferior de la ladera.
- Finalmente, las pruebas de funcionamiento realizadas a cada uno de los componentes del sistema, permitieron corregir y mejorar cada componente. Tal como la prueba de inclinación de la ladera, generando la caída de tierra con una velocidad adecuada desde su estado inicial de sin peligro, hasta su cambio a un deslizamiento moderado activando la alerta audible. Dando como resultado final la verificación del correcto comportamiento del sistema de detección de deslizamiento de tierra, la comunicación inalámbrica a través de *WiFi*, el constante monitoreo y registro en la base de datos levantada en el servidor *Uniform Server*.

4.2 Recomendaciones

- Se debe tener en cuenta que, debido al tipo de sensor utilizado en el prototipo, la calibración del mismo se debe diseñar en base de la cantidad mínima de peso que se va a testear.
- Los diferentes programas de diseño tanto de circuitos y modelo, proporcionan la oportunidad de una simulación, lo que permite tener una visión general del funcionamiento del mismo; a pesar de esta funcionalidad, se recomienda realizar la implementación física de cada diseño, para poder comprobar su funcionamiento real.
- Se recomienda realizar un análisis previo del área propensa a ocurrir el deslizamiento de tierra, así como las condiciones que producen el mismo, dando la oportunidad a futuros desarrolladores a utilizar otro tipo de sensores, tales como sensores de vibración, humedad, presión, etc.
- Es recomendable desarrollar a futuro un sistema de *backup* de la información para que se guarden los eventos en la base de datos.

5 REFERENCIAS BIBLIOGRÁFICAS

- [1 E. Acosta, E. Gavilanez, G. García, M. Meza, M. Naranjo, R. Muñoz, C. Ocampo, A. Paucar y A. Cárdenas, «ESTUDIO DE FACTORES MEDIOAMBIENTALES QUE INTERVIENEN EN LA OCURRENCIA DE DESLIZAMIENTOS N EL CERRO SUSANGA (CHIMBO, ECUADOR),» *Investigación Talentos*, vol. I, nº 1, pp. 53-53, 2014.
- [2 T. I. D. Database, «EM-DAT,» CRED/UCLouvain, [En línea]. Available:] <https://public.emdat.be/data>. [Último acceso: 3 Agosto 2021].
- [3 J. Suarez, «DESLIZAMIENTOS Y ESTABILIDAD DE TALUDES EN ZONAS TROPICALES,» Ingeniería de Suelos Ltda. , Bucaramanga , 1998.
- [4 «Servicio Integrado de Seguridad ECU 911,» Junio 2019. [En línea]. Available:] <https://www.ecu911.gob.ec/sistema-de-alerta-temprana-sat-beneficia-a-las-localidades-del-perfil-costanero-del-ecuador/>. [Último acceso: 8 Agosto 2021].

[5 R. Díaz, «Celda de Carga_bascula Electronica_resumen,» INSTRUMENTACION] ELECTRONICA, 2015.

[6 L. d. Valle, «Programarfacil.com,» [En línea]. Available:] <https://programarfacil.com/blog/arduino-blog/hx711-arduino-bascula-digital/>. [Último acceso: 8 Agosto 2021].

[7 M. Fezari y A. Zaytoonath, «NodeMCU V3 For Fast IoT Application Development,»] FAcultuy of IT, 2018.

[8 «descubrearduino,» 3 Julio 2021. [En línea]. Available:] <https://descubrearduino.com/arduino-uno/>. [Último acceso: 28 Agosto 2021].

[9 O. Fernadez, «codigoelectronica,» 27 Febrero 2021. [En línea]. Available:] <http://codigoelectronica.com/blog/arduino-lcd-i2c>. [Último acceso: 10 Septiembre 2021].

[1 A. Ruiz, «Observatorio Tecnológico,» Creative Commons, 1 Marzo 2010. [En línea]. 0] Available:

<http://recursostic.educacion.es/observatorio/web/fr/software/servidores/800-monografico-servidores-wamp>. [Último acceso: 8 Agosto 2021].

[1 «Creatividad Codificada,» 6 Noviembre 2019. [En línea]. Available: 1] <https://creatividadcodificada.com/arduino/comunicacion-serial-con-arduino/>. [Último acceso: 12 Agosto 2021].

[1 E. Marquina, «Sistema inteligente de alerta en la detección de fuego y gas para 2] mejorar la seguridad de restaurantes,» *SISTEMAS, CIBERNÉTICA E INFORMÁTICA*, vol. 17, nº 2, pp. 37-38, 2020.

[1 B. Christian, «SISTEMA DE COMUNICACION MEDIANTE WSN PARA 3] DETECCION Y ALERTA TEMPRANA DE DESLIZAMIENTOS DE TIERRA,» Enero 2018. [En línea]. Available: https://repositorio.uta.edu.ec/jspui/bitstream/123456789/27123/1/Tesis_1358ec.pdf

[1 SEPREDENAC, «Guía - La amenaza por deslizamiento en ámbito municipal,» de 4] *Programa de fortalecimiento de capacidades para el manejo del riesgo por deslaves*, El Salvador, Municipalidad de San José, pp. 10-14.

- [1 El Comercio, «Seis factores aumentan riesgo de derrumbes en la av. Simón Bolívar,» Grupo EL COMERCIO, 7 Enero 2020. [En línea]. Available: <https://www.elcomercio.com/actualidad/factores-riesgo-derrumbes-simon-bolivar/>. [Último acceso: 19 Agosto 2021].
- [1 El Comercio, «77 zonas de Quito están en riesgo por deslizamientos,» Grupo EL COMERCIO, 21 Enero 2020. [En línea]. Available: <https://www.elcomercio.com/actualidad/quito/quito-riesgo-deslizamientos-lluvias-clima.html>. [Último acceso: 19 Agosto 2021].
- [1 «Geología de Quito: fallas y sismicidad,» GEOTECNIA FÁCIL, 2020. [En línea]. Available: https://geotecniafacil.com/geologia-fallas-quito/#Cangahua_Quito. [Último acceso: 20 Agosto 2021].
- [1 J. Valverde, J. Fernández, E. Jiménez, T. Vaca y F. Alarcón, 8] «MICROZONIFICACIÓN SÍSMICA DE LOS SUELOS DEL DISTRITO METROPOLITANO DE LA CIUDAD DE QUITO,» Quito.
- [1 P. Cruz, «ESTUDIO DE IMPACTO AMBIENTAL DE LA LOTIZACIÓN SAN CARLOS 9] DEL SUR,» ESCUELA DE POSGRADO EN CIENCIAS ADMINISTRATIVAS Y ECONÓMICAS , Quito, 2006.
- [2 «12v24products,» 17 Junio 2021. [En línea]. Available: https://www.12v24vproducts.org/es/sirena-externo-alarma-12v#Kerui_mini_sirena_de_alarma_cc_12v__120db_cuerno_de_alarma_con_cable_para_sistema_de_alarma_de_casa_gsm. [Último acceso: 21 Septiembre 2021].
- [2 D. Web, «Ionos,» 11 Agosto 2020. [En línea]. Available: <https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/get-vs-post/>. [Último acceso: 14 Agosto 2021].
- [2 J. Fernandez, «fisicalab,» [En línea]. Available: <https://www.fisicalab.com/apartado/ley-de-ohm>. [Último acceso: 20 Septiembre 2021].
- [2 C. Veloso, «electrontools,» 14 Abril 2016. [En línea]. Available: <https://www.electrontools.com/Home/WP/valores-comerciales-de-resistencias/>. [Último acceso: 20 Septiembre 2021].

[2 A. Cordero, *Sistema de alerta temprana para monitoreo de deslizamientos*,
4] Infraestructura Vial, 2012.

ANEXOS

ANEXO 1: CERTIFICADO DE FUNCIONAMIENTO



ESCUELA POLITECNICA NACIONAL

Campus Politécnico "J. Rubén Orellana R

Quito, 27 de septiembre de 2021

CERTIFICADO DE FUNCIONAMIENTO DE PROYECTO DE TITULACIÓN

Yo, *Fanny Paulina Flores Estévez*, docente a tiempo completo de la Escuela Politécnica Nacional y como director de este trabajo de titulación, certifico que he constatado el correcto funcionamiento del prototipo de un sistema de alerta ante deslizamientos de tierra, el cual fue implementado por el estudiante Joel Alexis Quintanilla Armijo.

El proyecto cumple con los requerimientos de diseño y parámetros necesarios para que dicho prototipo sea considerado para una implementación real funcional que permita alertar a los habitantes de zonas aledañas a deslizamientos de tierra.

A handwritten signature in blue ink, appearing to read 'Fanny Paulina Flores Estévez', positioned above a horizontal line.

DIRECTOR

Ing. Fanny Paulina Flores Estévez, MSc.

Ladrón de Guevara E11-253, Escuela de Formación de Tecnólogos, Oficina 14. EXT: 2712

email: fanny.flores@epn.edu.ec

Quito-Ecuador

ANEXO 2: CÓDIGO DE PROGRAMACIÓN EN ARDUINO IDE

```
//Programa para detectar deslizamientos de tierra con sensor de peso
```

```
//Librerías
```

```
#include <HX711.h>
```

```
#include <Wire.h>
```

```
#include <LiquidCrystal_I2C.h>
```

```
#include <SoftwareSerial.h>
```

```
SoftwareSerial espSerial(5, 6);
```

```
String str;
```

```
int valor = 0;
```

```
//Variables
```

```
const int DOUT=A1;    //Variable sensor
```

```
const int CLK=A0;
```

```
int LED=12;          //Variable del led
```

```
int BUZZ=13;         //Variable del BUZZER
```

```
//Programa
```

```
LiquidCrystal_I2C lcd(0x27,16,2);
```

```
HX711 balanza;
```

```

float lectura_sensor=0.00;

String estado = "";

int dt;

unsigned long previousMillis = 0;

const long interval = 5000;

void setup() {

  Serial.begin(115200);

  espSerial.begin(115200);

  lcd.init();

  lcd.backlight();

  pinMode(BUZZ,OUTPUT);      //Configuro el pin del LED como salida
  pinMode(LED,OUTPUT);      //Configuro el pin del BUZZ como salida

  balanza.begin(SEÑAL, RELOJ);

  Serial.print("Valor de conversión: ");

  Serial.println(balanza.read());

  Serial.println("Mantenga libre la balanza");

  Serial.println("Descartano----");

  Serial.println("-----");

  balanza.set_scale(-582180);    // Se puntualiza la escala

  balanza.tare(50);             //Tara del peso

  Serial.println("Preparado");

```

```

lcd.clear();

lcd.setCursor(5,0);

lcd.print("LISTO");

delay(500);

}

void loop() {

  lectura_sensor = balanza.get_units(20),3;

  dt =lectura_sensor *1000; //Conversión de kilogramos a gramos

  Serial.print("Peso: ");

  Serial.print(dt);

  Serial.println(" g ");

  Serial.println(String(dt));

  unsigned long currentMillis = millis();

  if (currentMillis - previousMillis >= interval) {

    previousMillis = currentMillis;

    Serial.println("a ESP");

    espSerial.println(String(dt));

  }

  lcd.clear();

  lcd.setCursor(0,0);

  lcd.print("Peso: ");

  lcd.print(dt);

  lcd.print(" g ");

  lcd.setCursor(2,1);

```

```

lcd.print(estado);

//condicion de alarmas

if(dt >= 150){          //Condición para cuando el rango es mayor a 150(g)

    estado = "SIN PELIGRO";

    Serial.println("SIN PELIGRO");

    lcd.setCursor(2,1);

    lcd.print(estado);

    digitalWrite(LED,LOW);

    digitalWrite(BUZZ,LOW);

}

else if (dt < 150 && dt >= 50){ //Condición para cuando el rango es menor a 150(g) y
mayor a 50(g)

    estado = "DESLIZ.MODERADO";

    Serial.println("DESLIZ.MODERADO");

    digitalWrite(LED,LOW);

    digitalWrite(BUZZ,HIGH);

}

else if (dt < 50 && dt >= 0){ //Condición para cuando el rango es menor a 50(g) y
mayor a 0(g)

    estado = "EVACUAR";

    Serial.println("EVACUAR");

    digitalWrite(LED,HIGH);

    digitalWrite(BUZZ,HIGH);

    espSerial.println(String(dt));

}

}

```