



*“Responsabilidad con pensamiento positivo”*

**UNIVERSIDAD TECNOLÓGICA ISRAEL**

**TRABAJO DE TITULACIÓN EN OPCIÓN AL GRADO DE:  
INGENIERO EN ELECTRÓNICA DIGITAL Y  
TELECOMUNICACIONES**

**TEMA:**

IMPLEMENTACIÓN DE UN SISTEMA DE TX Y RX QUE PERMITA COMUNICAR LAS  
ÁREAS DE HERRAMENTALES Y PRODUCCIÓN MEDIANTE EL USO DE LA RED GPRS  
PARA EL CONTROL DE LOS MÓDULOS ANDON DE LA EMPRESA ROAD TRACK S.A.

**AUTOR:**

TITO PEREZ LUIS ALFREDO

**TUTOR:**

PhD. ALFONSO JOSE ZOZAYA

**QUITO, ECUADOR**

**2019**

## DECLARACIÓN

Yo, Luis Alfredo Tito Pérez, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; que he consultado las referencias bibliográficas que se incluyen en este documento, y que no he plagiado dicha información.

---

Luis Alfredo Tito Pérez

**UNIVERSIDAD TECNOLÓGICA ISRAEL****CERTIFICACIÓN DEL TUTOR**

En mi calidad de tutor del trabajo de titulación certifico:

Que el trabajo de titulación **“IMPLEMENTACIÓN DE UN SISTEMA DE TX Y RX QUE PERMITA COMUNICAR LAS ÁREAS DE HERRAMENTALES Y PRODUCCIÓN MEDIANTE EL USO DE LA RED GPRS PARA EL CONTROL DE LOS MÓDULOS ANDON DE LA EMPRESA ROAD TRACK S.A.”**, presentado por el Sr. Luis Alfredo Tito Pérez, estudiante de la carrera de Electrónica Digital y Telecomunicaciones, reúne los requisitos y méritos suficientes para ser sometido a la evaluación del Tribunal de Grado, que se designe, para su correspondiente estudio y calificación.

Quito D.M. Agosto del 2018

TUTOR

.....

Dr. Alfonso José Zozaya

## AGRADECIMIENTO

*A los docentes de la Universidad Israel por impartir sus conocimientos que forman parte de mi preparación y que seguramente serán un pilar fundamental en mi formación con profesional pero por sobre todo como una gran persona y a los compañeros de aulas y de trabajo cuyas experiencias estudiantiles y laborales han sido de gran aporte para mi aprendizaje.*

## DEDICATORIA

*Este trabajo va dedicado a mis padres quien con su infinito amor han sido un gran apoyo en cada momento ya que gracias a sus consejos me han formado con buenos valores y a ser perseverante en todos los aspectos de la vida.*

## TABLA DE CONTENIDO

Declaración.....	ii
Certificación Del Tutor.....	iii
Agradecimiento .....	iv
Dedicatoria.....	v
Tabla De Contenido.....	vi
Lista De Figuras .....	ix
Lista De Tablas.....	xii
Resumen .....	xiii
Summary.....	xiv
Introducción.....	1
Antecedentes.....	1
Planteamiento .....	2
Formulación De La Propuesta .....	2
Justificación .....	3
Objetivos.....	3
Objetivo General:.....	3
Objetivos Específicos: .....	3

## CAPÍTULO I

### 1. FUNDAMENTACIÓN TEÓRICA

1.1 Sistema Andon.....	4
1.1.1 Elementos Del Sistema Andon.....	5
1.2 Comunicación Inalámbrica .....	9
1.2.1 GPRS .....	9
1.3 Comunicación Móvil En Ecuador.....	12
1.4 Base De Datos.....	14
1.4.1 Sistema De Gestión De Base De Datos .....	14
1.4.2 Tipos De Base De Datos.....	14
1.5 Servidor.....	15

## CAPÍTULO II

### 2. METODOLOGÍA

2.1	Métodos De Investigación .....	16
2.1.1	Método Inductivo .....	16
2.1.2	Método Deductivo .....	17
2.1.3	Método Analítico Sintético.....	17
2.2	Tipos De Investigación .....	17
2.2.1	Investigación Bibliográfica .....	17
2.2.2	Investigación De Campo .....	18
2.2.3	Investigación Explicativa .....	18

## CAPÍTULO III

### 3. PROPUESTA

3.1	Descripción Breve De La Propuesta.....	19
3.2	Elementos De La Propuesta.....	20
3.2.1	Transmisor .....	21
3.2.2	Servidor .....	25
3.2.3	Base De Datos .....	26
3.2.4	Receptor.....	26
3.2.5	Visual Studio .....	27
3.3	Aspectos Técnicos .....	27
3.4	Costos.....	29
3.5	Ventajas Del Proyecto .....	31

## CAPÍTULO IV

### 4 IMPLEMENTACIÓN

4.1	Diseño de los Dispositivos .....	33
4.2	Desarrollo del Programa.....	42
4.2.1	Diseño de la Base de Datos .....	44
4.2.2	Desarrollo de la Base de Datos .....	44
4.2.3	Servicios Web.....	49
4.2.4	Programación Arduino .....	53
4.3	Pruebas de Funcionamiento .....	57

4.3.1	Puesta a Punto.....	59
4.3.2	Conexión a la Base de Datos .....	60
4.3.3	Funcionamiento del Transmisor .....	61
4.3.4	Funcionamiento del Receptor.....	66
4.4	Mediciones.....	69
4.5	Análisis de Resultados .....	72
<b>CONCLUSIONES .....</b>		<b>74</b>
<b>RECOMENDACIONES .....</b>		<b>76</b>
<b>ANEXOS .....</b>		<b>80</b>
	ANEXO 1 Programación GET .....	81
	ANEXO 2 Programación POST .....	82
	ANEXO 3 Programación Arduino.....	83
	ANEXO 4 Comandos AT .....	84
	ANEXO 5 Librería GET para Andon .....	85
	ANEXO 6 Librería SEND para Andon .....	87
	ANEXO 7 Desarrollo del proyecto.....	89
	ANEXO 8 Hoja de datos Tip 122 .....	93
	ANEXO 9 Manual de usuario.....	94



## LISTA DE FIGURAS

Figura. 1.1. Modulo o slave.....	6
Figura. 1.2. LCD 16*2.....	7
Figura. 1.3. Conversor RS-485 a RS-232.....	8
Figura. 1.4. Arquitectura de la red GPRS.....	10
Figura. 1.5. Espectro de frecuencias.....	11
Figura. 1.6. Esquema de comunicación mediante GPRS .....	12
Figura. 1.7. Líneas activas en el Ecuador .....	13
Figura. 1.8. Esquema de usuarios de un servidor .....	15
Figura. 3.1. Diagrama de bloques de la propuesta.....	20
Figura. 3.2. Módulo SIM-900.....	22
Figura. 3.3. Banda de frecuencias de la operadora Conecel S.A.....	23
Figura. 3.4. Arduino Mega .....	24
Figura. 3.5. Diagrama de Gantt de la propuesta .....	30
Figura. 4.1. Diagrama de conexiones .....	34
Figura. 4.2. Diagrama de conexiones indicador .....	35
Figura. 4.3. Diagrama de conexiones modem .....	36
Figura. 4.4. Layout PCB.....	36
Figura. 4.5. Esquema 3D del circuito terminado .....	37
Figura. 4.6. Ubicación de elementos en la placa .....	38
Figura. 4.7. Soldadura de los elementos sobre la placa.....	38
Figura. 4.8. Ensamble de la placa con el modem .....	39
Figura. 4.9. Ensamble de la placa con el Arduino .....	40
Figura. 4.10. Ensamblaje Arduino Mega con Shield SIM-900 .....	40
Figura. 4.11. Carcasa de protección .....	41
Figura. 4.12. Ensamble final del dispositivo .....	41
Figura. 4.13. Diagrama de flujo de la propuesta .....	42
Figura. 4.14. Panel de control XAMPP .....	43
Figura. 4.15. Página web de Xampp.....	45
Figura. 4.16. Creación de base de datos .....	45
Figura. 4.17. Tabla de datos .....	46
Figura. 4.18. Parámetros de la tabla de datos .....	46

Figura. 4.19. Estructura de la tabla de datos.....	47
Figura. 4.20. Seguridad de la tabla .....	48
Figura. 4.21. Privilegios globales .....	48
Figura. 4.22. Servicio Web Tipo POST .....	49
Figura. 4.23. Verificación de servicios web .....	49
Figura. 4.24. Cadena de recepción tipo JSON.....	50
Figura. 4.25. Verificación con ayuda del Postman.....	50
Figura. 4.26. Evento ingresado correctamente .....	51
Figura. 4.27. Servicio web tipo GET.....	51
Figura. 4.28. Verificación servicio GET .....	52
Figura. 4.29. Respuesta del servicio GET .....	52
Figura. 4.30. Eventos guardados en la base de datos .....	52
Figura. 4.31. Creación de las librerías para control de Modems .....	54
Figura. 4.32. Librerías añadidas al programa .....	55
Figura. 4.33. Diseño de ventana de control .....	56
Figura. 4.34. Ventana de control eliminación de datos .....	57
Figura. 4.35. Alerta generada por el Andon System .....	58
Figura. 4.36. Elementos del proyecto .....	59
Figura. 4.37. Software de verificación de conexión a la base de datos.....	60
Figura. 4.38. Ubicación del transmisor de datos Andon .....	61
Figura. 4.39. Transmisor de datos Andon .....	62
Figura. 4.40. Condiciones iniciales de funcionamiento.....	62
Figura. 4.41. Datos del incidente producido en la estación.....	63
Figura. 4.42. Estado del envío de la información .....	64
Figura. 4.43. Evento ingresado correctamente .....	64
Figura. 4.44. Estado del mensaje de texto .....	65
Figura. 4.45. Desactivación de alarmas .....	65
Figura. 4.46. Ubicación dispositivo Receptor de datos Andon .....	66
Figura. 4.47. Receptor de datos Andon .....	66
Figura. 4.48. Visor de alarmas Andon System .....	67
Figura. 4.49. Receptor de datos Andon .....	67
Figura. 4.50. Estado de recepción del evento .....	68
Figura. 4.51. Información extraída de la base de datos .....	68
Figura. 4.52. Servidor ubicado en el área de Sistemas .....	69

Figura. 4.53. Información extraída de la base de datos ..... 72

Figura. 4.54. Información extraída de la base de datos ..... 73

**LISTA DE TABLAS**

Tabla. 3.1. Características SIM-900 .....	22
Tabla. 3.2. Características Arduino Mega .....	24
Tabla. 3.3. Actividades para el desarrollo de la propuesta .....	30
Tabla. 3.4. Costos de la implementación de la propuesta.....	31
Tabla. 4.1. Pruebas realizadas en el transmisor .....	70
Tabla. 4.2. Pruebas realizadas en el receptor.....	71

## RESUMEN

La comunicación es un aspecto fundamental en la actualidad y en este proyecto se busca la forma de comunicar dos áreas en distintas edificaciones. En la empresa Road Track S.A. se cuenta con un sistema que genera alertas cuando ocurren incidentes en las diferentes estaciones de las líneas del área de producción.

El sistema se llama Andon System y permite los técnicos de Herramientales brindar el debido soporte a las estaciones. Debido a la expansión del departamento de producción se separó a las dos áreas por falta de espacio. La comunicación del sistema no se lo puede realizar de manera remota, y como solución se propone implementar un sistema de transmisión para mantener las dos áreas comunicadas.

Para la implementación se propone elaborar dos dispositivos que funcionen como transmisor y receptor mediante el uso de la red GPRS. Los elementos que componen estos dispositivos son un Arduino Mega y un modem SIM-900. Estos dispositivos se encargan de transmitir la información que generan los módulos del Andon System.

Para establecer la comunicación entre los dos dispositivos es necesario contar un servidor en el cual se aloja una base de datos donde se muestre la información transmitida por parte de los módulos. Adicionalmente se envía un mensaje de texto al encargado del soporte técnico para las ocasiones en que no se encuentre frente a su computador ya que es ahí donde se visualiza la alerta generada.

**Palabras clave:** Arduino, Servidor, Transmisión, Módulos, Modem

## SUMMARY

Communication is a fundamental aspect at present and this project seeks to communicate two areas in different buildings. In the company Road Track S.A. there is a system that generates alerts when incidents occur in the different stations of the lines of the production area.

The system is called Andon System and allowed the technicians of Herramentales to provide the appropriate support to the stations. Due to the expansion of the production department, the two areas had to be separated due to lack of space. The communication of the system can not be done remotely, and as a solution it is proposed to implement a transmission system to keep the two areas communicated.

For the implementation, it is proposed to elaborate two devices that function as transmitter and receiver through the use of the GPRS network. The elements that make up these devices are an Arduino Mega and a SIM-900 modem. These devices will be responsible for transmitting the information generated by the Andon System modules.

To establish communication between the two devices it is necessary to have a server in which a database will be housed where the information transmitted by the modules will be displayed. Additionally, a text message is sent to the technical support manager for the occasions in which it is not in front of your computer since that is where the generated alert is displayed.

**Keywords:** Arduino, Server, Transmission, Modules, Modem

## INTRODUCCIÓN

### ANTECEDENTES

Las comunicaciones han permitido generar grandes avances tecnológicos en todos los ámbitos conocidos y sobre todo a nivel industrial, donde es innegable que las comunicaciones han concedido facilidades en ciertos procesos y actividades que en su momento fueron un problema para grandes empresas.

La empresa de servicios telemáticos Road Track S.A. es una organización que cuenta con quince años de funcionamiento en el mercado automotriz y son los proveedores directos de dispositivos de rastreo y seguridad vehicular de alta calidad a la ensambladora GM Ómnibus en Ecuador.

Estos dispositivos son validados, ensamblados y verificados por los operadores del área de Producción con ayuda de simuladores y distintos *software* que permiten evaluar las características y especificaciones de los equipos, con el objetivo de cumplir con los requerimientos del cliente.

Dichas pruebas son realizadas dentro del laboratorio, cuyo espacio físico fue compartido con el departamento de Herramientales, en donde, el personal de ésta área se encarga de realizar mantenimientos del tipo preventivo y correctivo de los elementos y dispositivos que permiten la realización de las pruebas funcionales de estos equipos.

Estas áreas fueron separadas debido a la expansión del área de Producción, lo que provocó que el área de Herramientales se traslade a una edificación ubicada a 80 metros de distancia, por lo que el personal encargado del mantenimiento no se encuentra al tanto de los incidentes suscitados en el sitio antes mencionado.

## **PLANTEAMIENTO**

En el área de Producción se cuenta con dos líneas principales de manufactura donde se procesan y se empacan los equipos de monitoreo y rastreo satelital. Cada línea tiene un número específico de estaciones que son manejadas por un operador. En cada una de las estaciones se cuenta con elementos electrónicos llamados *fixtures* los cuales permiten ejecutar las pruebas funcionales

Adicionalmente, en cada estación se cuenta con un módulo (*slave*) que forma parte del sistema de generación de alertas *Andon*, el cual permite monitorear y reportar inmediatamente sobre imprevistos ocurridos en el proceso de pruebas funcionales de los equipos de monitoreo vehicular.

Este módulo le permite al operador de la estación activar el sistema *Andon* mediante un pulsador, en el momento en que se genera un desperfecto con los dispositivos *fixtures*. La alerta que genera este sistema de supervisión informa al personal del área de Herramientales para tomar acciones correctivas en la estación donde se produce el inconveniente.

Este proceso se implementó dentro del laboratorio con el fin de evitar retrasos en la manufactura de los equipos, por lo cual el proceso de alertas generados por los operadores y el soporte inmediato brindado por los técnicos de herramientas generó fluidez en su momento. Cabe mencionar que el departamento de Herramientales compartió la misma edificación hasta que el área de producción se expandió

## **FORMULACIÓN DE LA PROPUESTA**

Se propone implementar un sistema de Tx y Rx que permita comunicar las áreas de Herramientales y Producción mediante el uso de la red GPRS para el control de los módulos Andon de la empresa Road Track S.A.



## **JUSTIFICACIÓN**

El proyecto permitirá comunicar el sistema *Andon* ubicado en el área de Producción con la estación del personal de Herramientales, y formará parte del sistema de alerta ante situaciones adversas que requieran acciones correctivas.

Esta solución evitará que se produzcan estancamientos en la elaboración de los productos y promoverá a fluidez en la línea de producción ante problemas relacionados directamente con el área de herramientas.

## **OBJETIVOS**

### **OBJETIVO GENERAL:**

- Implementar un sistema de Tx y Rx que permita comunicar las áreas de herramientas y producción mediante el uso de la red GPRS para el control de los módulos Andon de la empresa Road Track S.A.

### **OBJETIVOS ESPECÍFICOS:**

- Ensamblar los dispositivos transmisores que permitan establecer la comunicación entre los dos puntos.
- Desarrollar la programación que permita controlar los módems para transferir los datos que genera el sistema Andon.
- Desarrollar una aplicación para visualizar las alertas generadas que incluye una alerta vía mensaje de texto al delegado del área de Herramientales.
- Implementar una base de datos en un servidor web que permita alojar los eventos enviados por el modem del transmisor hacia el receptor.
- Realizar pruebas de funcionamiento para evaluar la comunicación.

# CAPÍTULO I

## FUNDAMENTACIÓN TEÓRICA

En el presente Capítulo se presentan los fundamentos teóricos que sustentan la implementación de un sistema de Tx y Rx que permitirá comunicar las áreas de Herramientales y Producción mediante el uso de la red GPRS para el control de los módulos *Andon* de la empresa Road Track S.A.

### 1.1 SISTEMA ANDON

Es un sistema de generación de alertas cuyo propósito tiene informar sobre hechos puntuales que se producen en una jornada laboral del departamento de Producción.

Anteriormente cuando que se presenta una anomalía en la estación el operador levanta una bandera de color amarillo con su mano para informar al supervisor que se presenta un inconveniente en su puesto de trabajo.

Una vez informado el suceso, el supervisor debe acudir al lugar para verificar el motivo por el cual se generó la alerta. Una vez identificada la avería el supervisor se dirige hacia el área de Herramientales a informar el hecho suscitado en la estación a los técnicos encargados del mantenimiento de los *fixtures*.

Este proceso se manejó de manera admisible pero el inconveniente es que se desperdicia demasiado tiempo, tomando en cuenta que las dos áreas aún se encuentran en la misma edificación. Debido a ello se planteó implementar un sistema automático que genere estas alertas con el propósito de agilizar el proceso.

La implementación del nuevo Andon trajo consigo avances muy importantes, que con el paso del tiempo lo convirtieron en un sistema muy sólido para la generación de alertas. Entre las mejoras se destaca la posibilidad de observar el accionamiento del sistema mediante una baliza que se enciende cuando se activa la alerta.

Además se añadió una alerta auditiva mediante un parlante que emite un sonido en formato MP3 con lo cual facilitó a los técnicos del área de herramientas percatarse de manera aún más rápida del momento en que surgen inconvenientes en las estaciones de las líneas de Producción.

### **1.1.1 ELEMENTOS DEL SISTEMA ANDON**

Se procede a detallar los elementos que componen el sistema de supervisión y alertas *Andon*. En donde se presentan las características tanto de *software* como de *hardware* y demás partes que lo conforman. Adicionalmente se presentan figuras que muestran cómo se encuentra realizado el montaje en cada una de las estaciones de trabajo.

Básicamente el cuerpo del sistema *Andon* radica en el principio de funcionamiento del módulo o *slave* como se muestra en la Figura 1.1. Este es un mecanismo desarrollado que en su interior cuenta con una placa electrónica cuyo elemento principal es un microcontrolador.



Figura. 1.1. Modulo o slave

Fuente: (Manual Andon, 2015)

Este *slave* es el dispositivo que permite generar las alertas cuando ocurren inconvenientes de distintas índoles en la estación de trabajo. Como se puede ver en la Figura 1.1 cuenta con pulsadores normalmente abiertos y cada uno de ellos representa un área operativa distinta diferenciada con un color. En este caso es el botón que interesa es el botón que representa el área de Herramientales.

La señal que envía este pulsador es una señal sincrónica que genera datos en paralelo y mediante comunicación UART (*Universal Asynchronous Receiver-Transmitter*) convertirlos en datos en serie para ser transmitidos al PC mediante comunicación RS-485, esto gracias al microcontrolador PIC 18f4620 con el que cuenta cada módulo (Manual Andon, 2015).

Esta señal tiene una velocidad de transmisión de 9600 baudios con un tamaño de 5 bytes. Cada byte tiene información importante acerca del evento generado, el primer byte se debe al dato de inicio de transmisión, el segundo byte tiene información acerca de la ubicación del módulo, es decir, el número de *slave* ID, y a la línea de ensamble a la que pertenece.

En tanto que el tercer byte indica el estado de los botones de alarma, mientras que el cuarto byte es un *check sum* que suma los primeros tres bits y verifica que la transmisión se realice de manera correcta, y por último el byte número cinco indica el final de la trama de transmisión (Manual Andon, 2015).

El dispositivo además cuenta con LED's (Diodo emisor de luz) indicadores que representan el *status* del módulo, es decir, si se encuentra trabajando correctamente. Este módulo se enciende con la ayuda de un *switch* ON/OFF.

Adicionalmente cuenta con un bus de datos y un LCD (*Liquid Cristal Display*) como se ilustra en la Figura 1.2, aquí es donde se muestra información del estado del módulo. Este elemento se conecta mediante el bus de datos al *slave* y en conjunto forman parte del sistema de alertas que se encuentran en cada una de las estaciones de las líneas de producción.



Figura. 1.2. LCD 16\*2

Fuente: (Manual Andon, 2015)

La conexión desde el módulo hacia el computador principal para la transferencia de la información es mediante la topología del tipo bus o de forma lineal, es decir, se conectan uno a continuación del otro, de esta forma se conectan todos los módulos conforman una línea de producción (Manual Andon, 2015).

Para ello se emplea el cable UTP (*Unshielded Twister Pair*) con terminales RJ-11. Estas extensiones van conectadas a cada *slave* con la topología mencionada anteriormente. Y para realizar la comunicación con el computador se cuenta con un conversor RS-485 a RS-232 o comunicación serial como se muestra en la Figura 1.3.



Figura. 1.3. Conversor RS-485 a RS-232

Fuente: (Manual Andon, 2015)

Adicionalmente, se cuenta con una baliza principal por cada línea y una baliza secundaria por cada estación. Estas balizas emiten una señal lumínica de color rojo, las cuales indican que existe una anomalía en el área de Producción de acuerdo a la señal que fue accionada.

La baliza principal se encuentra ubicada en la primera estación de cada línea, sitio previamente elegido para que el supervisor de producción tenga un ángulo de visión admisible para acudir de manera inmediata a la línea donde se generó la alerta.

En el caso de las balizas secundarias, se cuenta con una baliza por cada estación y al igual que en el primer caso le permite al supervisor identificar la estación de forma inmediata.

Para accionar las balizas se cuenta con una tarjeta de control que permite encender las tanto principales como secundarias. Este circuito se conecta mediante un cable USB (*Universal Serial Bus*) al computador principal donde se encuentra instalado el programa del sistema *Andon*. Es aquí donde se tiene el control de las señales que envían los *slaves* y de la manera en cómo se generan las alertas tanto visuales como auditivas.

## 1.2 COMUNICACIÓN INALÁMBRICA

Es la forma en la que dos más dispositivos pueden comunicarse entre sí sin la necesidad de establecer una conexión por un medio de propagación físico, sino que emplea la modulación de ondas electromagnéticas a través del espacio. Existen diferentes tecnologías que dependen de las frecuencias que utilizan, el alcance y la velocidad de transmisión.

La comunicación de forma inalámbrica permite la conectividad de dispositivos remotos que se encuentran a grandes distancias. Con el paso de los años han evolucionado y en la actualidad se convierte en una alternativa fiable para establecer conectividad. Se clasifican de la siguiente manera:

- WPAN: (Wireless Personal Area Network) es un tipo de red de cobertura personal en el que se incluye la tecnología Bluetooth y Zigbee.
- WLAN: (Wireless Local Area Network) es un tipo de red similar a la red de área local pero sin pasar por un medio físico donde se destaca el WI-FI.
- WMAN: (Wireless Metropolitan Area Network) se basa en la interoperabilidad mundial para el acceso por microondas.
- WWAN: (Wireless Wide Area Network) red de área extendida inalámbrica donde se destaca la tecnología para celulares GSM (*Global System for Mobile communications*) y GPRS (*General Packet Radio Service*).

### 1.2.1 GPRS

GPRS (Servicio general de paquetes vía radio) es un sistema de comunicación móvil creado en la década de los 80. Se derivó del sistema GSM (Sistema global para las comunicaciones móviles), pero con más ventajas y múltiples servicios e incorpora nuevas aplicaciones en el sector de la telefonía celular principalmente. (VIU, 2018)

Esta manera de establecer comunicación se la conoce también con el nombre de tecnología 2.5G, ya que es una extensión de la red GSM o 2G. El cambio se produce en la velocidad de transmisión de datos que emplea con el mismo ancho de banda que su antecesor y la capacidad de acceder al Internet mediante una IP (Protocolo de internet).

En la Figura 1.4 se describe de manera general la arquitectura de la red GPRS. Estos elementos permiten establecer la conexión del usuario hacia el internet a partir de un dispositivo móvil.



Figura. 1.4. Arquitectura de la red GPRS

Fuente: (Móvil fácil, 2016)

El usuario: es el punto de inicio de la comunicación y es el que emplea la red para transmitir información.

La estación terrestre: Uno o más transmisores o receptores, o una combinación de transmisores y receptores, donde se incluyen las instalaciones necesarias para asegurar un servicio de radiocomunicación en un lugar determinado. (Arcotel, 2017)

SGSN (*Serving GPRS Support Node*): es el elemento que se encarga de interactuar con la estación terrena, ya que guarda la información de los abonados de la red. (Móvil fácil, 2016)

GGSN (*Gateway GPRS Support Node*): es el punto donde se conecta la red GPRS con las redes IP externas en esta caso el Internet. (Móvil fácil, 2016)

Internet: es el medio por el cual la comunicación se establece sin tomar en cuenta la distancia entre los usuarios.



El espectro de frecuencia en que se encuentra ubicada la red GPRS es UHF (Ultra Alta Frecuencia) como se muestra en la Figura 1.5. El ancho de banda en que maneja este medio de comunicación se encuentra en el rango de 800 y 900 MHz que son destinados a la comunicación móvil.

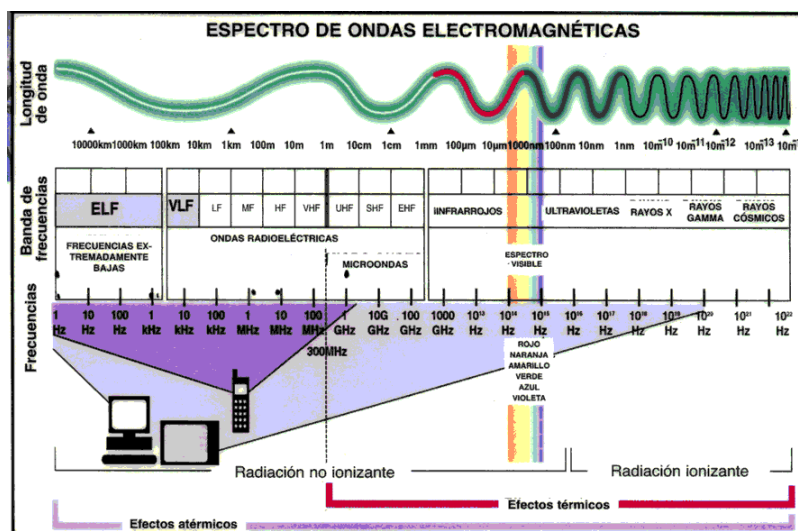


Figura. 1.5. Espectro de frecuencias

Fuente: (Blog de tecnologías Wireless, 2018)

El tipo de modulación que emplea es GMSK (*Gaussian Minimum Shift Keying*) un método de modulación digital proveniente de la modulación por desplazamiento de fase que permite reducir los requerimientos de ancho de banda empleado en la telefonía celular. Es un enfoque simple pero eficaz para la transmisión inalámbrica de datos. (Haykin, 2001)

La red GPRS permite la transferencia de datos a velocidades operativas desde 56 a 80 Kbps en comparación con los 9,6 Kbps que emplea GSM, es decir, hasta 10 veces más rápido que los servicios de conmutación de circuitos, lo que lo hace un sistema confiable y se ajusta a las necesidades de transmisión moderada de información.

La conmutación de paquetes significa que los recursos de la red GPRS son empleados únicamente cuando el usuario envía o recibe datos, esto permite a muchos abonados emplear el mismo ancho de banda y depende de la cantidad de información que transmite.

La forma de transmitir los datos se realiza en modalidad de paquetes con lo que se logra que la transferencia se efectúe en forma de bloques los cuales se agrupan posteriormente al llegar a su destino. Esta forma de mantener la comunicación permite utilizar diferentes canales en paralelo con lo que se logra una mayor velocidad y capacidad en la transmisión. (Entel Chile, 2018)

Un avance importante en el desarrollo de la comunicación que emplea la red GPRS fue la facturación de servicios respecto a la cantidad de datos transmitidos. Debido a que antes se debió cancelar el valor del servicio por el tiempo de conexión, lo que representó siempre un cobro excesivo por el servicio prestado.

En la Figura 1.6 se muestra la arquitectura de la comunicación mediante el uso de la red GPRS, donde se puede observar que un teléfono móvil o un computador pueden acceder al internet mediante el uso de una IP y a su vez estos datos transmitidos son guardados en una base de datos en un servidor.



Figura. 1.6. Esquema de comunicación mediante GPRS

Fuente: (VIU, 2018)

### 1.3 COMUNICACIÓN MÓVIL EN ECUADOR

Actualmente en el Ecuador se cuenta con tres operadoras que manejan el servicio de comunicaciones celulares, las cuales son: Conecel S.A. (Claro), Otecel S.A. (Movistar), y CNT S.A. están tres telefónicas compiten en el mercado nacional, pero hasta la actualidad se cuenta con un solo operador dominante.

Claro debió ser calificada como operador dominante hace varios años, con el propósito de impedir prácticas monopólicas, competencia desleal, acumulación de ganancias ilegales, y proteger al consumidor. Luego de constantes presiones de la CNT y Movistar, recién el 30 de julio del 2010, el Conatel declaró "operador dominante" a Claro. (Plan V, 2018)

La banda de frecuencias en las que operan es 850 MHz en el rango de 824 a 849 MHz de subida, y 869-894 MHz de bajada. Pero desde el año 2008 se entregó a Claro y Movistar la banda de 1900 MHz para SMA (Servicio Móvil Avanzado) en el rango de 1850 a 1910 MHz de subida y 1930 – 1990 MHz de bajada. (Plan V, 2018)

En la Figura 1.7 se muestra un estudio de medición realizado por la Agencia de Regulación y Control de las Telecomunicaciones del mes de Julio del 2017, donde se evidencia que la Operadora Conecel es la que más abonados tiene con un 64,86 % de usuarios.

En segunda instancia se encuentra la operadora Otecel con una intervención del 29,30 %, y por último CNT con el 5,86 %. Esto indica que los usuarios prefieren la operadora Claro por sobre las telefónicas Movistar y de la Corporación Nacional de Telecomunicaciones.

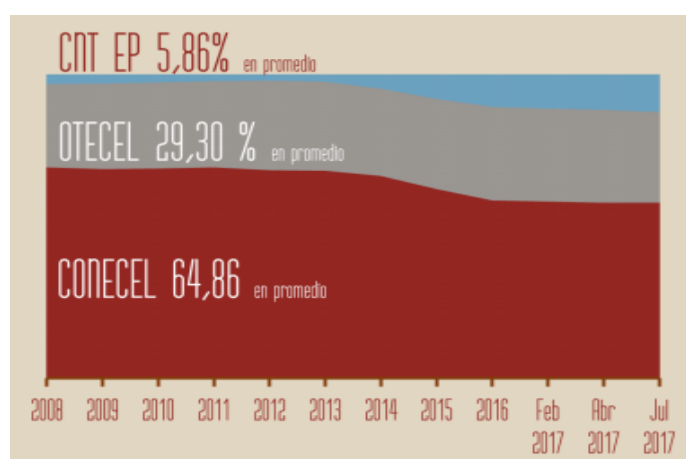


Figura. 1.7. Líneas activas en el Ecuador

Fuente: (Agencia de Regulación y Control de las Telecomunicaciones, 2017)

## **1.4 BASE DE DATOS**

Se establece como un conjunto de datos interrelacionados entre sí que pertenecen a un mismo contexto y que son almacenados de forma organizada para usos particulares. Un ejemplo práctico de base de datos es una hoja de Excel, en donde se guarda información de manera ordenada de acuerdo a las necesidades del usuario que la emplea.

Actualmente las bases de datos se encuentran en formato digital debido al desarrollo tecnológico en la informática y la electrónica, esto se debe a que ofrece un rango más elevado para almacenar información.

### **1.4.1 SISTEMA DE GESTIÓN DE BASE DE DATOS**

Se presenta como un tipo de software muy específico que sirve como interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan. Se compone de un lenguaje de programación estándar para la manipulación, definición y consulta de la información que se almacena en la misma. (Maestros del Web, 2018)

El SGBD (Sistema de gestión de base de datos) presenta varias ventajas para el almacenamiento de datos como, por ejemplo: mantener la integridad de la información almacenada, y la seguridad en el caso de que un usuario no autorizado intente vulnerar las seguridades de acceso.

### **1.4.2 TIPOS DE BASE DE DATOS**

Existen diferentes tipos de base datos donde cada una tiene ciertas características que permiten a los desarrolladores inclinarse por una de ellas depende de las necesidades que presenten. A continuación, se detallan las más utilizadas:

- MySQL: se caracteriza por su rapidez, aunque no es recomendable para la gestión de grandes cantidades de información.
- Oracle: excelente para la administración de información con gran cantidad de datos, aunque su costo es elevado.
- PostgreSQL: al igual que la anterior maneja grandes cantidades de datos y suelen ser utilizadas en intranets.
- Access: es una base de datos desarrollada por Microsoft y al igual que Oracle su costo es elevado. (Maestros del Web, 2018)

## 1.5 SERVIDOR

La principal función de un servidor Web es almacenar los archivos de un sitio y emitirlos por Internet. Básicamente, un servidor Web es una gran computadora que guarda y transmite datos vía el sistema de redes llamado Internet. (Duplika, 2018)

Se refiere a la máquina que almacena y maneja los sitios web utilizados por las compañías que ofrecen hostings, es decir, un sitio web que se puede utilizar para almacenar información al que se puede acceder con ciertos permisos, la cual, es la encargada de gestionar los datos que se desean enviar.

El trabajo de un servidor es almacenar y transmitir datos de un sitio Web según lo solicita el navegador de un usuario. Dentro de estas funciones el usuario puede configurar un servidor para controlar el acceso de ciertos usuarios. En la Figura 1.8 se muestra un esquema básico de una red de usuarios conectados a un servidor en internet.

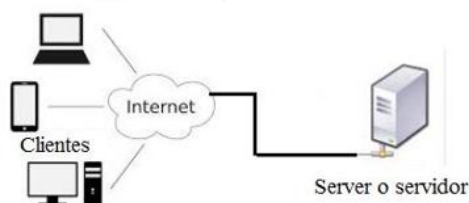


Figura. 1.8. Esquema de usuarios de un servidor

Fuente: (Área tecnología, 2018)

## **CAPÍTULO II**

### **METODOLOGÍA**

En el presente Capítulo se detallan las metodologías de investigación a las que se desea recurrir para resolver el problema planteado. Cada uno de estos métodos y tipos de investigación son explicados de forma breve de acuerdo al aporte que generan en cada evento para la ejecución del proyecto.

#### **2.1 MÉTODOS DE INVESTIGACIÓN**

##### **2.1.1 MÉTODO INDUCTIVO**

Muestra los inconvenientes que se presenta al momento de visualizar las alertas que se encuentren en el área de Producción y la respectiva solución que se expondrá para solucionar los inconvenientes que generaron el cambio de localidad del área de Herramientales. Consiste además en hechos reales de estudios de observación y vivencias que han desencadenado en la búsqueda de una solución al problema que se plantea.

Este método se emplea para visualizar la problemática que actualmente existe con el cambio de localidad del área de Herramientales y que se espera que las alertas que se generan en el área de Producción puedan ser visualizadas en la otra localidad y que facilite la comunicación entre los dos departamentos.

### **2.1.2 MÉTODO DEDUCTIVO**

Este método presenta un análisis que se realizó previamente para evidenciar la problemática que se generó con el cambio de localidad del departamento de Herramientales y de esa manera dar a conocer los procedimientos que actualmente se presentan y respecto a ello proponer una solución que permita establecer la comunicación entre las dos áreas.

### **2.1.3 MÉTODO ANALÍTICO SINTÉTICO**

Es un método que permite mostrar la problemática que actualmente existe en el área de Producción y de esa manera buscar la solución más adecuada para el problema planteado. Permite brindar una solución útil al problema de una manera sencilla y que permite que las alertas generadas se envíen a su respectivo receptor, en este caso desde el departamento de Producción al área de Herramientales.

## **2.2 TIPOS DE INVESTIGACIÓN**

### **2.2.1 INVESTIGACIÓN BIBLIOGRÁFICA**

Se emplea la investigación Bibliográfica debido a que el tema propuesto se encuentra enfocado en la tecnología, por lo tanto las investigaciones que se han realizado anteriormente permiten generar una posible solución al problema planteado. Ya que se puede partir de experiencias anteriores para proponer la solución más efectiva.

Esto permite buscar una solución acorde a las necesidades que presenta el área de Producción a partir de experiencias similares en diferentes ámbitos que permitan proponer una solución efectiva del problema planteado.

### **2.2.2 INVESTIGACIÓN DE CAMPO**

Se basa en la recopilación de datos de acuerdo al inconveniente que se presenta en el área de Producción. La misma que muestra las condiciones físicas para implementar una solución más conveniente, ya que esta solución debe abarcar las condiciones que actualmente se presentan en el departamento antes mencionado.

### **2.2.3 INVESTIGACIÓN EXPLICATIVA**

Permite describir el problema y las causas que lo generan. Ya que brinda la posibilidad de encontrar los fundamentos de la necesidad que genera mantener las dos áreas comunicadas. Y de esa manera buscar la solución más adecuada que garantice la solución definitiva del problema planteado.



## **CAPÍTULO III**

### **PROPUESTA**

#### **3.1 DESCRIPCIÓN BREVE DE LA PROPUESTA**

La presente propuesta se busca la manera de transmitir la información que se generan en los *slaves* dentro del área de producción de la empresa Road Track S.A. con los datos más relevantes acerca del inconveniente suscitado. Esta información se guarda en una base de datos que está alojada en un servidor ubicado en el área de Sistemas de la misma empresa.

La información del evento que se guarda en la base de datos se extrae mediante un receptor de datos que se ubica en el área de Herramientales, dicho departamento se encuentra en otra edificación, y mediante una interfaz gráfica se visualizan los datos que son enviados por el transmisor.

Con esto se busca promover la agilidad en el proceso de resolución de inconvenientes puesto que el personal que brinda el soporte no cuenta con la información específica acerca del evento suscitado.

### 3.2 ELEMENTOS DE LA PROPUESTA

Se procede a detallar los elementos que conforman la solución al problema planteado y donde se tomó en cuenta detalles técnicos, de funcionamiento y de costos. En este caso se optó por implementar un sistema que utilice la interfaz de comunicación GPRS.

Los elementos que componen la presente propuesta se detallan en la Figura 3.1, en donde se muestra el diagrama de bloques del sistema que se propone en este proyecto. El sistema se compone de cuatro elementos principales que son los encargados de transferir la información que generan los slaves del sistema Andon de la empresa Road Track S. A.

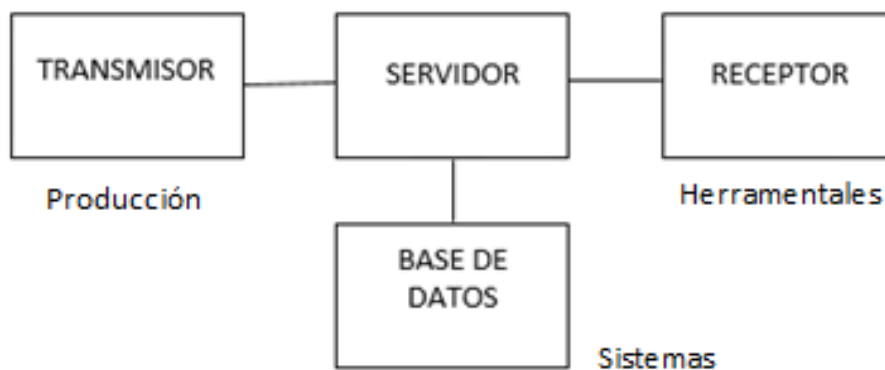


Figura. 3.1. Diagrama de bloques de la propuesta

Elaborado por: Luis Tito

El elemento transmisor recoge la información que generan los *slaves* cada vez que se generan anomalías en la estación y los presenta en una ventana que puede ser manejada por el Supervisor del Área de Producción. Esta información es transmitida y posteriormente guardada en un servidor que contiene una base de datos.

En cuanto se refiere a la parte del manejo de la información se cuenta con un servidor en el cual se tiene una base de datos, esta se encarga de gestionar y guardar los datos que envía el transmisor y que posteriormente serán extraídos por el receptor que está ubicado en la otra edificación.

Por su parte el elemento receptor se encarga de extraer la información del evento y posteriormente lo muestra en un aplicativo visual donde se despliega la información más relevante acerca del evento que se genera en el área de Producción. A continuación se procede a detallar brevemente cada elemento que conforma la propuesta:

### **3.2.1 TRANSMISOR**

Es el elemnto que se encargara de gestionar los datos que generan los *slaves* del área de Produccion y enviarlos mediante la red GPRS hacia una base de datos alojada en un servidor. Constan de dos elementos que operan en conjunto, el primero es un Modulo SIM-900 y el segundo es una arduino Mega. Estos elementos seran deatallados a continuacion:

#### **3.2.1.1 MODULO SIM-900**

Se trata de una tarjeta compacta de comunicación inalámbrica compatible con todos los modelos de Arduino. Es una tarjeta basada en el modulo SIM-900 que se encargará de enviar la señal mediante la interfaz GPRS hacia el receptor. El control del modem se lo realiza mediante comandos AT (que significa Atención).

En la Figura 3.2 se muestra la tarjeta SIM-900, que fue elegida porque responde a las características de la información que se desea transmitir. Se consideró la mejor opción de acuerdo a su bajo costo y por la simplicidad que tiene para codificar los mensajes y enviarlos hacia su receptor.



Figura. 3.2. Módulo SIM-900

Fuente: (Importadora Arduinel, 2018)

En la Tabla 3.1 se muestran las especificaciones técnicas del Módulo SIM-900.

Tabla. 3.1. Características SIM-900

Voltaje DC	3.4 V – 4.5 V
Corriente	1 A
Quad-Band	850/ 900/ 1800/ 1900 MHz
Interfaz	Serial
Velocidad de transmisión	171 Kb/s

Elaborado por: Luis Tito

Dentro de este elemento se debe incluir una tarjeta SIM (Subscriber Identify Module) de una operadora local que servirá como el identificador para el dispositivo, el cual permitirá establecer la comunicación desde el modem hacia un servidor en internet mediante el protocolo IP.

### 3.2.1.2 MODULO DE IDENTIFICACIÓN DE ABONADO

Para la presente propuesta se elige la operadora Claro debido a que es la empresa dominante en cuanto se refiere a telefonía móvil en el País. Esto debido a que en la implementación se requiere de un plan de datos mensual para transmitir los datos generados por el Andon System.

La telefónica Claro inicia sus operaciones en el país en el año 2011 y es la primera operadora en cuanto a líneas activas en el territorio Nacional. Se maneja en la banda de 850 MHz entre los rangos de 824 a 835 MHz, 845 a 846,5 MHz en enlace de subida. Y en la banda de 869 a 880 MHz a 890 a 891,5 MHz en enlace de bajada como se muestra en la Figura 3.3.

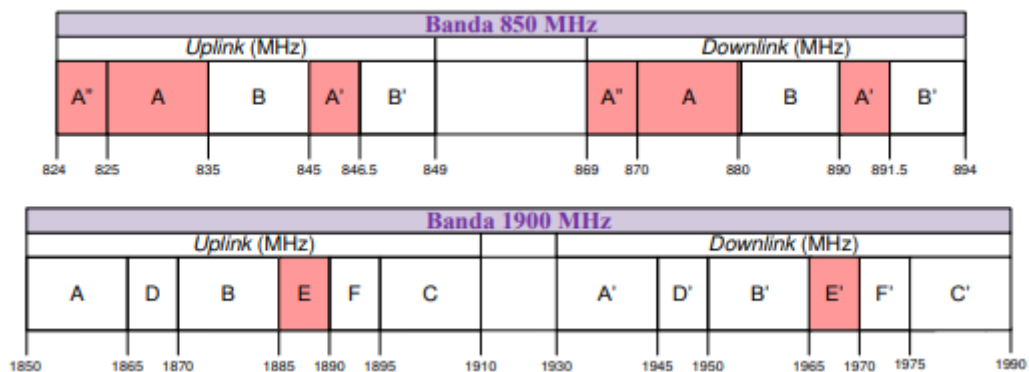


Figura. 3.3. Banda de frecuencias de la operadora Conecel S.A.

Fuente: (Secretaria Nacional de Telecomunicaciones, 2008)

En cuanto a la banda de 1900 MHz se maneja entre los rangos de subida de 1855 a 1890 MHz y de bajada entre los rangos de 1965 a 1970 MHz. En cuanto se refiere a la parte de control y alimentación del modem se cuenta con un microcontrolador de la familia Arduino.

### 3.2.1.3 ARDUINO MEGA

El Arduino Mega es un microcontrolador de la plataforma que lleva el mismo nombre. Este es un dispositivo muy versátil y debido a sus renovadas características respecto a los primeros controladores, lo hace ideal para la implementación de un sistema de transmisión de larga distancia ya que utiliza un protocolo distinto al de radiofrecuencia. (Arduino, 2018).

Para la parte del control del módulo de transmisión se requiere de este Arduino que permitirá suministrar el voltaje adecuado al módulo SIM-900 y que facilitará la conexión hacia la PC tanto del transmisor como del receptor para gestionar los datos que se obtienen en el módulo. En la Figura 3.4 se muestra el elemento que se va a utilizar.



Figura. 3.4. Arduino Mega

Fuente: (Hobby Tronics, 2018)

En la Tabla 3.2 se muestran las características técnicas que se han tomado en cuenta para la presente implementación.

Tabla. 3.2. Características Arduino Mega

Voltaje DC recomendado	7 V - 12 V
Corriente de entrada	40 mA
Microprocesador	Atmega 2560
Puertos Serial	4
Pines I/O	54
Corriente de salida	30 mA

Elaborado por: Luis Tito

Entre las ventajas de utilizar este dispositivo se debe a la cantidad de entradas de transmisión y recepción que posee, además de un mayor número de pines que permiten tener alcance a un control de más elementos de visualización de alertas. Esto en el caso en que se desee incrementar las características que ya posee el *Andon*.

Estos dos elementos permiten codificar la señal que envían los esclavos de cada estación y transmitirlos mediante el uso del protocolo GPRS, con el fin de gestionar y almacenar la información que será recibida en un computador del departamento a donde se desea enviar la información.

Un punto importante a tomar en cuenta para emplear esta plataforma es debido a que se trata de un software gratuito y de fácil manejo, ya que se pretende emplear tarjetas de la misma línea lo hace más versátil y más sencillo para desarrollar aplicaciones como la que se pretende integrar en este proyecto.

Para la elaboración de la tarjeta que permita ensamblar el módulo SIM-900 con el arduino Mega es indispensable utilizar un entorno que me permita desarrollar el circuito electrónico que posteriormente será transferido a una placa (baquelita), donde se acoplará el módulo que permite conectarlos al arduino.

### **3.2.2 SERVIDOR**

Es un elemento que permite la interacción entre los dispositivos de transmisión y recepción, ya que permite el tráfico de la información que se generan en los slaves. Dentro de este computador se sitúa la base de datos encargada de gestionar la información que envían desde el área de Producción.

Este medio sirve de puente entre los dos dispositivos y se ubica en el área de Sistemas de la empresa. Cuenta con una IP pública que permite transmisor y al receptor iniciar una conexión mediante la red GPRS para realizar el envío de los datos sin importar la distancia a la que se encuentre.

La comunicación se realiza con ayuda de los Modems que cuentan con el chip de la operadora elegida. La forma en la que se establece la transmisión hacia la base de datos es a través de comandos AT.

### 3.2.3 BASE DE DATOS

Para almacenar la información será indispensable crear una base de datos que permita guardar los datos generados por los *slaves*, pero antes de esto se deben seguir los pasos pertinentes para su elaboración para cumplir con los requerimientos que hacen que un sistema sea estable.

Esta base de datos será elaborada en la plataforma de MySQL. La razón de elegir este *software* es debido a la simplicidad para la obtención de información, y porque es un lenguaje de programación estándar.

MySQL es un sistema de gestión de base de datos relacional en código abierto y gratuito que se basa en el lenguaje de consulta estructurado. Una de las ventajas de este *software* es que se ejecuta en todas las plataformas incluso Linux. Su entorno se especializa en aplicaciones en la web y emplea PHP (*Hipertext Preprocessor*) como lenguaje de programación.

PHP es un lenguaje de código abierto dedicado al desarrollo web. La ventaja que tiene PHP sobre JAVA es la ejecución del código directamente en el servidor, donde únicamente se muestra el HTML (*Híper Text Markup Language*), es decir, se presenta solo el producto final más no el código de programación.

### 3.2.4 RECEPTOR

Este elemento posee características similares al transmisor que se detalló anteriormente, es decir, cuenta con un arduino Mega, una tarjeta Shield Sim-900 que en su interior posee un chip de una operadora telefónica local. Estos tres elementos funcionan de manera conjunta.

Este elemento se ubica en el área de Herramientales de la empresa, en un sitio que no interviene en las actividades laborales del personal.



### 3.2.5 VISUAL STUDIO

Es un entorno de programación que en este proyecto permitirá generar una librería que será integrada en el programa Andon System, esto permitirá controlar el modem del elemento SIM-900 que está conectado al Arduino.

La creación de este software permitirá enviar datos relevantes acerca de la alerta generada. La información que se desea transferir es la siguiente: número del slave, la estación donde se generó, el tipo de evento, fecha y hora.

Esta información será presentada mediante un mensaje en un computador en el receptor que incluirá un mensaje de texto a un teléfono celular de la persona encargada de realizar el soporte técnico de los simuladores y bancos de pruebas. Estos ajustes podrán ser modificados en la PC principal, es decir, en el transmisor.

Para la implementación de la propuesta se procede a desarrollar los elementos que permitirán dar solución al problema. Se describe también el ensamble de las partes que componen el sistema. En la Figura 3.14 se muestra el diagrama de flujo de la propuesta.

### 3.3 ASPECTOS TÉCNICOS

Los elementos que conforman este proyecto son:

- Transmisor de datos Andon
- Receptor de datos Andon
- Adaptadores de voltaje AC/DC 9V 1A
- Cables USB tipo B
- Servidor + Base de datos

Los dispositivos son elementos cuya alimentación se realiza con un adaptador de voltaje de 9V 1A, esto con el fin de proveer la suficiente corriente para que el modem de la tarjeta Shield funcione de manera correcta. Los adaptadores se conectan a una fuente de poder de 120VAC y provee de una alimentación de 9V en DC para el arduino.

La tasa de transferencia de datos del dispositivo se mide de acuerdo a la cobertura que provee la operadora y se encuentra en el rango de 31.4 Mbps en enlace de bajada y 21.13 Mbps en enlace de subida. Este dato es muy importante debido a que es la velocidad con la que se transfieren los datos del evento a la base.

Cada dispositivo cuenta con una latencia de 100ms que interfiere en la comunicación desde el modem hacia el servidor y una potencia de transmisión de 2 W. Este dato es obtenido desde el SIM con el que trabaja y se realiza en el punto donde se va a instalar el transmisor y el receptor.

Cabe mencionar que los dispositivos tanto Transmisor como Receptor trabajan en el rango de UHF desde 900 MHz hasta 1900 MHz. En campo de la telefonía se refiere a que utilizan la tecnología 2,5 G de transmisión de datos y tecnología de primera generación para el envío de SMS.

El periodo de envío de mensajes de texto es de 10 segundos hasta que llega al destinatario, mientras que el periodo para la transferencia de la información de los slaves hacia la base de datos oscila entre 20 a 25 segundos. Esto depende de la cobertura ya que de acuerdo a este aspecto responde el modem.

El tiempo en establecer la conexión a la base de datos responde a que se envían comandos AT al modem y se realiza en un espacio de 2 segundos por respuesta. Esto quiere decir que por cada segmento de la comunicación hacia la base de datos se puede visualizar mediante el LED un doble parpadeo indicando la respuesta del modem.

Para realizar la comunicación se emplean 8 comandos AT similar a la comunicación que emplea un teléfono celular cuando se enciende hasta que empieza a recibir la cobertura de la telefónica con la que trabaja. Debido a esto es el periodo que tarda en ingresar los datos del evento en la base de datos.

La base de datos se encuentra en un servidor y cuenta con una IP pública, esto quiere decir que los dispositivos se pueden conectar hacia la base sin importar la distancia a la que se encuentren. Este tipo de comunicación es posible ya que el SIM adopta una IP que lo provee la operadora y de esa manera puede conectarse hacia la base de datos.

### **3.4 COSTOS**

Para la implementación de la presente propuesta se optó por la manera de utilizar la menor cantidad de recursos, debido a ello se emplearán programas de entorno gratuito. Para la parte de hardware se debe adquirir los Módems que forman parte del SIM-900 que debe alimentarse mediante un Arduino y la elaboración de un PCB que los ensamble.

Por último, se debe manejar el gasto que generan las dos tarjetas SIM que serán adquiridas para cada uno de los módulos SIM-900, donde el costo como se explicó en el Capítulo 1 dependerá de la cantidad de datos que se envíen y no por el tiempo en que permanece conectado al servicio.

Para elaborar el presupuesto de la presente propuesta se debe partir por detallar cada una de las actividades que se formarán parte de la ejecución e implementación del sistema de transición mediante el uso de la red GPRS.

Para la realización del cronograma se empleó el programa Project donde se indican cada una de las actividades realizadas empleadas tanto en la elaboración de la parte física como también la parte de redacción del documento que avala la realización del proyecto. Cada actividad cuenta con la debida fecha de duración desde el inicio hasta el fin de la actividad.

En la Tabla 3.3 se describen las actividades que se realizarán para desarrollar la propuesta, cada una de estas tiene estrecha relación con los objetivos específicos descritos en la Introducción del presente trabajo monográfico. Cabe destacar que el horario de la ejecución de las actividades se inicia a las 6 de la tarde con una duración de dos horas diarias.

Tabla. 3.3. Actividades para el desarrollo de la propuesta

Nombre de tarea	Duración	Comienzo	Fin
<b>Proyecto</b>	<b>96 días</b>	<b>mar 18/09/18</b>	<b>jue 07/02/19</b>
Adquisición de materiales	6 días	mar 18/09/18	mar 25/09/18
Elaboración del Hardware	30 días	mié 26/09/18	jue 08/11/18
Programación del Software	20 días	vie 09/11/18	lun 10/12/18
Implementación en el sitio	10 días	mar 11/12/18	mié 26/12/18
Pruebas funcionales	30 días	jue 27/12/18	jue 07/02/19
<b>Redacción PIC</b>	<b>16 días</b>	<b>lun 11/02/19</b>	<b>vie 08/03/19</b>
Redacción del PIC	16 días	lun 11/02/19	vie 08/03/19

Elaborado por: Luis Tito

En la Figura 3.5 se puede apreciar el Diagrama de Gantt de la propuesta, en donde se puede verificar de manera cronológica la ejecución de cada una de las actividades para la realización de la implementación y la redacción de la parte escrita del informe.

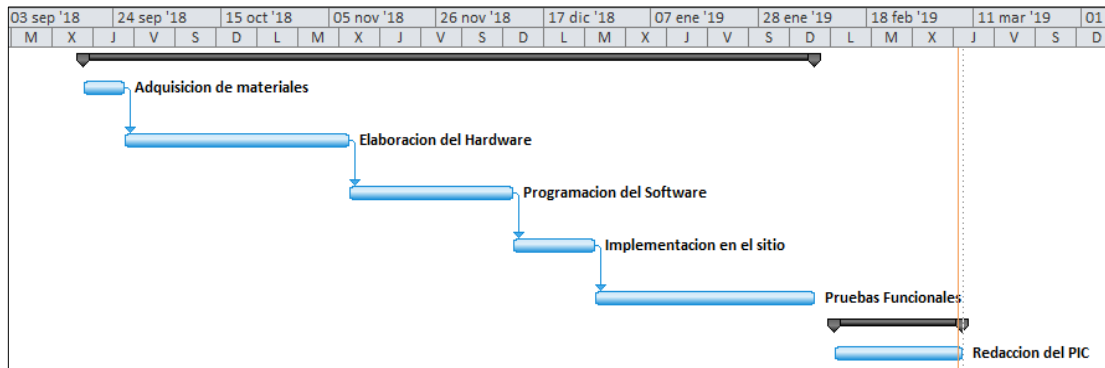


Figura. 3.5. Diagrama de Gantt de la propuesta

Elaborado por: Luis Tito

Mientras tanto en la Tabla 3.4 se muestra el detalle de los gastos que generan la implementación de la propuesta y adicionalmente se cuenta con el número de horas dedicadas a la elaboración de la misma con su respectivo costo por hora.

Tabla. 3.4. Costos de la implementación de la propuesta

Ítem	Cantidad	P. Unitario	Total
Arduino Mega Genérico	2	\$ 20.53	\$ 41.06
GPRS-GSM Shield SIM-900	2	\$ 41.07	\$ 82,14
Caja tipo 0	2	\$ 3.00	\$ 6.00
Baquelita Layout impreso	2	\$ 15.00	\$ 30.00
Resistencia 220 ohmios	2	\$ 0.05	\$ 0.10
Resistencia 4,7 Kohmios	2	\$ 0.05	\$ 0.10
Tip 122	2	\$ 0.40	\$ 0.80
Diodo Emisor de luz	2	\$ 0.10	\$ 0.20
Sim Card	2	\$ 2.00	\$ 4.00
Número de horas dedicadas a la elaboración de la propuesta	45	\$15.00	\$720.00
		<b>Costo Total</b>	\$ 869.40

Elaborado por: Luis Tito

### 3.5 VENTAJAS DEL PROYECTO

Este proyecto se considera una solución a un inconveniente que tiene el personal de Herramientales para estar al tanto de los inconvenientes que suceden en el área de Producción. Ya que al momento no se sabe con certeza cuáles son los problemas que ocurren en cierto momento.

Lo que se busca es que las personas que se encuentran ligadas a estos inconvenientes puedan conocer de manera más precisa y con mayor rapidez sobre las anomalías que ocurren en las estaciones de trabajo. Las ventajas del proyecto se detallan a continuación:

- Promueve la fluidez en la comunicación entre departamentos.
- Muestra información precisa acerca de los inconvenientes suscitados en Producción.
- El personal autorizado recibe una notificación vía mensaje de texto a su teléfono con detalles más específicos acerca del evento.
- Es muy sencillo de manipular.
- Es amigable con el ambiente ya que no desprende desechos.
- No afecta a las actividades cotidianas.
- Es silencioso.
- Es robusto ya que cumple con los objetivos para los que fue desarrollado.

## **CAPÍTULO IV**

### **IMPLEMENTACIÓN**

En este Capítulo se describen los pasos que se tomaron para realizar la implementación del sistema propuesto. Se toma en cuenta la elaboración del hardware, la programación de los dispositivos encargados de la transmisión y recepción de los eventos y las respectivas pruebas funcionales en la empresa.

#### **4.1 DISEÑO DE LOS DISPOSITIVOS**

En primera instancia de deben fabricar los dispositivos capaces de transmitir información mediante la red GPRS. Y para ello se recurre a la elaboración de una placa que permita acoplar al Arduino con las SIM-900 ya que por sus características físicas no se pueden superponer la una sobre la otra.

Para ello se desarrolla una adaptación entre los dos elementos en la plataforma Proteus Professional cuyas características se muestran en la Figura 4.1. En la imagen se muestra el esquema de conexiones que hace que el arduino provea de energía a la tarjeta Shield SIM-900 y adicionalmente se muestra la conexión del LED para visualizar el estado de envío de la información.

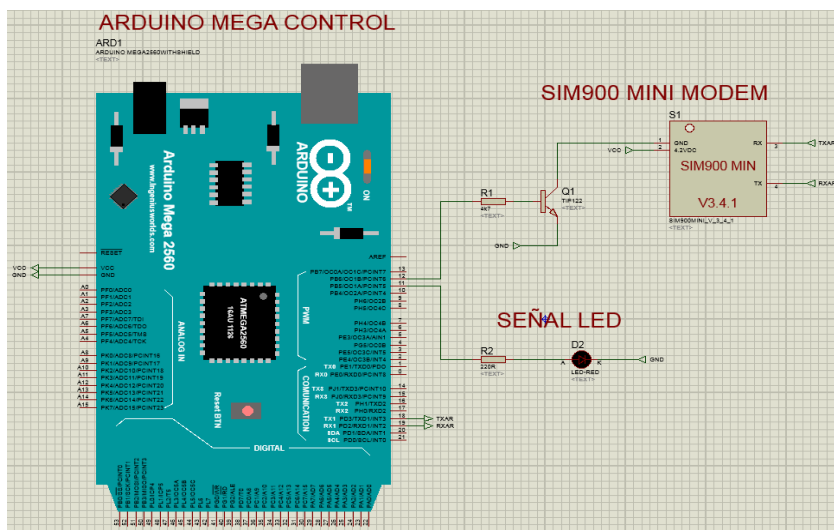


Figura. 4.1. Diagrama de conexiones

Elaborado por: Luis Tito

Según las características técnicas del arduino mega cada Pin de salida PWM manejan una corriente de salida de 30 mA Este dato sirve para realizar el cálculo de las resistencias tanto para el LED como para el TIP122.

En la parte de visualización se ubica un LED que permite monitorear la conexión a la base de datos, para el cálculo de la resistencia se emplean los datos conocidos de las características del arduino mega y los datos de la caída de tensión en un LED de color amarillo de la Figura 4.2.

$$V_d = 2,4V$$

$$I = 30 \text{ mA}$$

$$E = 9V$$

$$R = \frac{E - V_d}{I}$$

$$R = \frac{9V - 2,4V}{30 \text{ mA}}$$

$$R = 220\Omega$$



$$P = V * I$$

$$P = 2,4V * 40mA$$

$$P = 0,096W$$



Figura. 4.2. Diagrama de conexiones indicador

Elaborado por: Luis Tito

En cuanto a la estabilización del modem se emplea un TIP 122 mismo que provee la suficiente corriente en modo de saturación al circuito como se muestra en la Figura 4.3. Obteniendo los datos del datasheet del transistor se tienen los datos para calcular la resistencia de base:

$$I_C = 5A$$

$$I_B = 2mA$$

$$V_{BE} = 0,7V$$

$$E = 9V$$

$$R = \frac{E - V_d}{I}$$

$$R = \frac{9V - 0,7V}{2mA}$$

$$R = 4150\Omega$$

$$R = 4700\Omega$$

$$P = V * I$$

$$P = 0,7V * 2mA$$

$$P = 1,4W$$

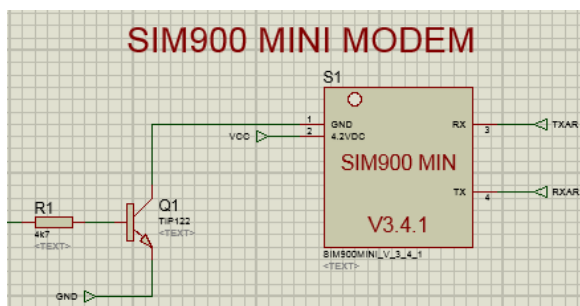


Figura. 4.3. Diagrama de conexiones modem

Elaborado por: Luis Tito

Como se mencionó en el apartado de los elementos que permiten diseñar el Hardware en el Capítulo 3 se recurre al diseño del PCB que permitirá acoplar los dos elementos citados en el párrafo anterior con lo que se desea elaborar un solo dispositivo más reducido tanto para la parte de transmisión como para la recepción.

Posteriormente se debe diseñar el PCB de la placa que permitirá unificar las dos tarjetas. Y para ello se emplea el entorno ARES que permite realizar simulaciones virtuales sobre el funcionamiento de los elementos antes de ser diseñados en la placa. La parte del diseño de PCB se muestra en la Figura 4.4.

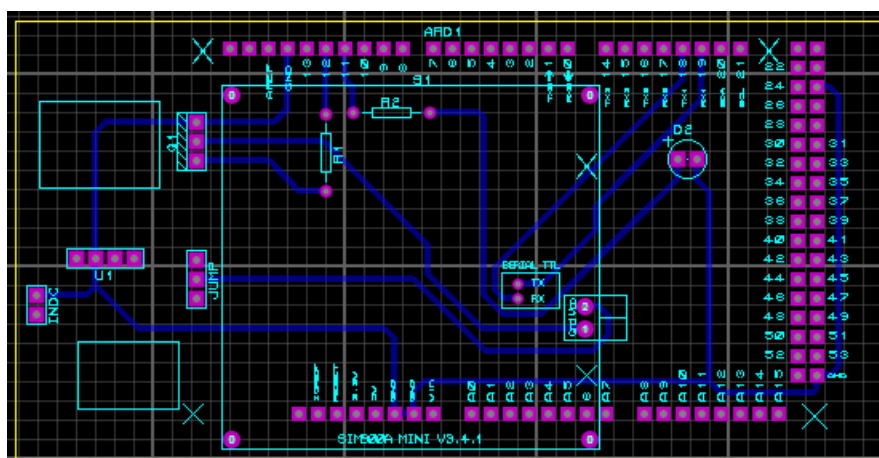


Figura. 4.4. Layout PCB

Elaborado por: Luis Tito

En la Figura 4.5 se muestra una ilustración en 3D del circuito terminado, en éste se evidencia la forma en que se ubicarán cada uno de los elementos utilizados. Cabe recalcar que se colocarán pines que permitirán al modem acoplarse a la baquelita y para adicional se colocaran tornillos para permitir que se mueva.

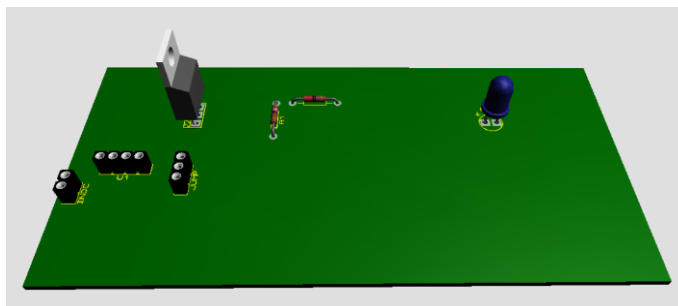


Figura. 4.5. Esquema 3D del circuito terminado

Elaborado por: Luis Tito

Una vez que se cuenta con la placa impresa en una baquelita se dispone a ensamblar las dos partes para conformar un elemento más reducido. Esto permitirá conectar las partes fundamentales del modem con el arduino. Los elementos que se deben conectar son la fuente de alimentación para el modem y los pines que permiten realizar la transmisión de datos.

Las dimensiones de la placa son 11,5 cm \* 5,8 cm. En la Figura 4.6 se muestra la ubicación de los elementos sobre la placa y además se puede ver que se realizaron cortes sobre las esquinas, esto con el fin de permitir a la placa acoplarse a la cubierta de plástico que evitará que el dispositivo sufra golpes o afectación por el polvo u otros contaminantes.

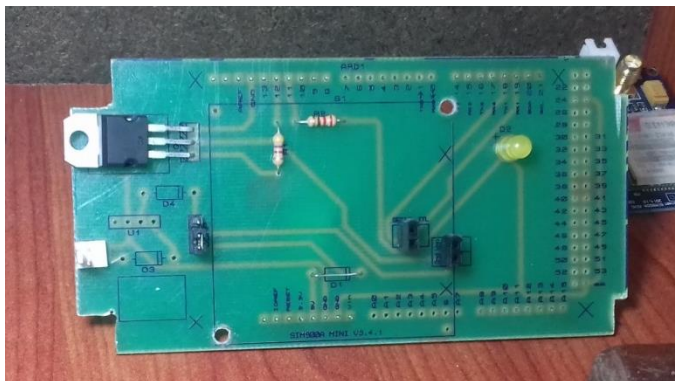


Figura. 4.6. Ubicación de elementos en la placa

Elaborado por: Luis Tito

Como se aprecia en la imagen se realizaron perforaciones sobre la baquelita con el fin de acoplar el modem a la placa debido a que no se puede sobreponer directamente sobre el Arduino. Esto permitirá brindar un soporte estable del modem sobre la baquelita.

En la Figura 4.7 se muestra la etapa de soldadura de los elementos con ayuda de un caufín. Se puede observar que se ubicaron pines alrededor de la baquelita, esto se realizó con el fin de permitir acoplar a la placa con el arduino, y adicionalmente se aprovecha para energizar al Modem desde el arduino.

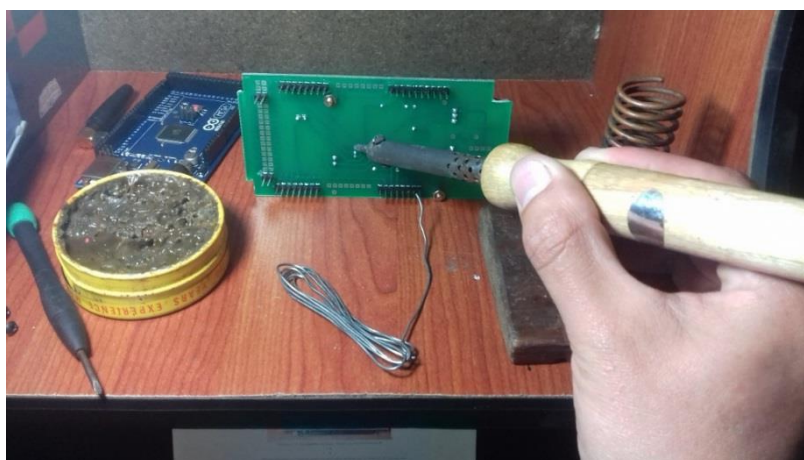


Figura. 4.7. Soldadura de los elementos sobre la placa

Elaborado por: Luis Tito

Posteriormente se procede a realizar la ubicación del modem sobre la baquelita con ayuda de los elementos de conexión. Estos pines permiten conectar los elementos de comunicación del Modem para alimentarlo posteriormente con la ayuda del Arduino. Además se observan los postes que permitirán sujetar al modem con la baquelita como se muestra en la Figura 4.8.

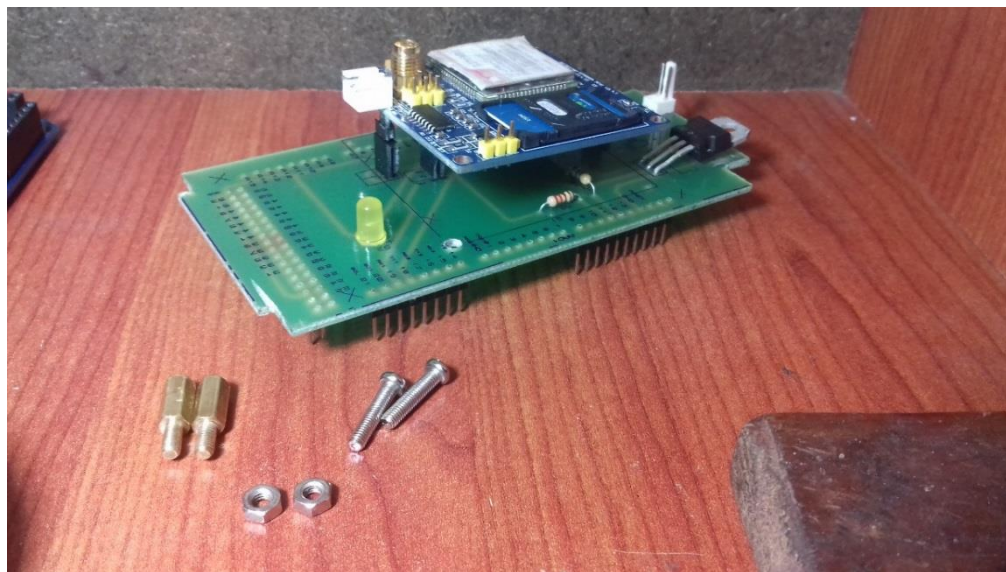


Figura. 4.8. Ensamble de la placa con el modem

Elaborado por: Luis Tito

Una vez ensambladas las dos partes se dispone a colocar la baquelita sobre el Arduino y para ello hay que basarse en los pines de conexión que tiene la placa que encajan en los pines de conexión del Arduino como se muestra en la Figura 4.9. Esto lo hace un dispositivo más reducido para ubicarlo posteriormente en un contenedor plástico.



Figura. 4.9. Ensamble de la placa con el Arduino

Elaborado por: Luis Tito

En la Figura 4.10 se muestra la conexión de los dos elementos para conformar un elemento más reducido. Se puede apreciar que la Shield SIM-900 se ubica en la parte superior de la placa mientras que los pines inferiores permiten que se pueda ensamblar con el Arduino Mega y así aprovechar las entradas de voltaje para el modem.

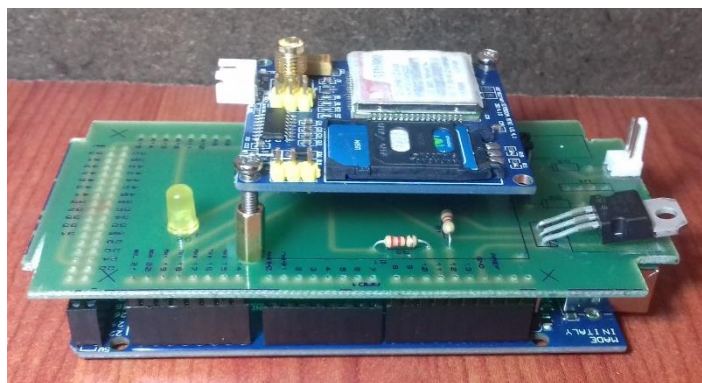


Figura. 4.10. Ensamblaje Arduino Mega con Shield SIM-900

Elaborado por: Luis Tito

Posteriormente con el fin de mantener protegido al elemento se dispone de una carcasa de un material resistente a golpes como se ve en la Figura 4.11, cuyas dimensiones son: 6,68 cm de ancho, 13 cm de largo, y 4,3 cm de altura. Esta es una caja de material plástico que evitará también que el elemento se contamine con materiales que perjudiquen su funcionamiento.



Figura. 4.11. Carcasa de protección

Elaborado por: Luis Tito

Finalmente se debe realizar un orificio que permita que la antena sobresalga de la carcasa. Como se puede apreciar en la Figura 4.12. De igual manera se debe ubicar un par de orificios más para ubicar los indicadores luminosos.



Figura. 4.12. Ensamble final del dispositivo

Elaborado por: Luis Tito

## 4.2 DESARROLLO DEL PROGRAMA

Para entender la lógica del manejo de la parte física con el software se desarrolla un diagrama para verificar los estados de cada etapa de la comunicación hasta la recepción del evento, como se muestra en la Figura 4.13.

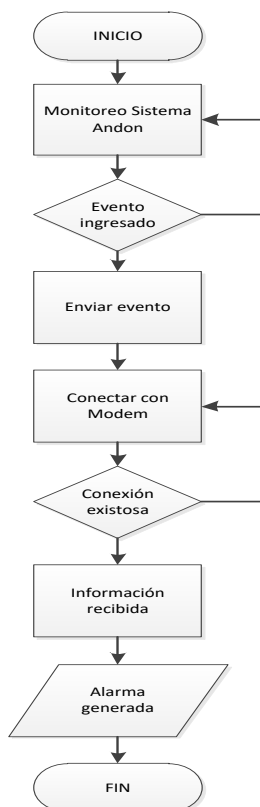


Figura. 4.13. Diagrama de flujo de la propuesta

Elaborado por: Luis Tito

En el diagrama se puede observar el desarrollo de la propuesta de manera general y para ello a continuación se describen los pasos que se siguieron para elaborar los elementos del diseño. Además se detallan los programas que permitieron verificar el funcionamiento del sistema.



Para empezar la ejecución de la propuesta es necesario contar con los programas necesarios y por ello se debe instalar en un computador. Inicialmente se tiene que incorporar un servidor web libre llamado Xampp, el cual es una colección de programas con software libre, que son los intérpretes de la programación del proyecto.

Este programa consiste básicamente en un sistema gestión de base de datos en MYSQL, que además cuenta con un servidor Apache, y con los intérpretes del lenguaje de programación en PHP. Una vez incorporado en la PC se ejecuta y posteriormente se despliega una ventana como se muestra en la figura 4.14.

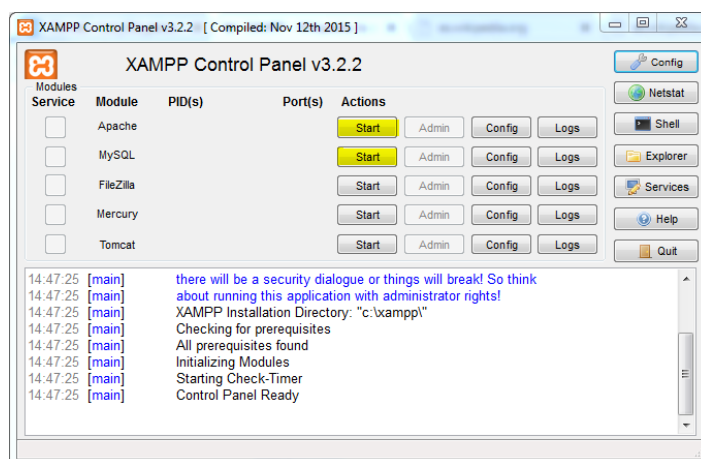


Figura. 4.14. Panel de control XAMPP

Elaborado por: Luis Tito

En esta ventana se deben activar los servicios de MySQL y Apache mediante un clic en el botón Start. Este paso es muy importante ya que permite abrir puertos de comunicación y convertir al computador en un servidor web para la gestión de datos. Esto permitirá almacenar la información en una base que se creará posteriormente.

#### 4.2.1 DISEÑO DE LA BASE DE DATOS

Para realizar el diseño de una base de datos se debe partir por el tipo de información que se desea almacenar, para el caso de los slaves se trata de datos relacionados a un incidente puntual en un momento y en el lugar específico.

Estos datos son: la identidad del slave, la hora, la fecha, la estación, el código y el tipo de evento. Y responden a las necesidades que el cliente espera para que la información manejada sea clara y esté al alcance del usuario.

La manera de ordenar los registros de cada evento se queda determinada de la siguiente manera:

Id_slave	Hora	Fecha	Estación	Código	Evento
65	18:45	21/01/2019	Estación 0	Perfiles	Herramientales
66	07:24	15/12/2018	Estación 5	Mensajería	Herramientales
78	11:40	01/02/2019	Estación 7	CARE	Herramientales
85	13:58	30/01/2019	Estación 4	Automáticas	Herramientales
99	16:19	18/11/2018	Estación 1	GP12	Herramientales

#### 4.2.2 DESARROLLO DE LA BASE DE DATOS

Para la creación de la base de datos se debe abrir el navegador en una PC e ingresar la palabra `localhost` o la dirección IP `127.0.0.1` en la barra de direcciones. Esto permite ingresar a la página web de `Xampp` y permite comprobar que los servicios activados funcionan de manera adecuada. En la figura 4.16 se muestra el entorno de la página web de `Xampp`.



Figura. 4.15. Página web de Xampp

Elaborado por: Luis Tito

Una vez que se tiene acceso a la página web se procede a dar clic en la pestaña phpMyAdmin que dará paso al entorno de creación de bases de datos. En el panel de navegación se encuentra la opción Nueva, como se muestra en la Figura 4.16, y enseguida se presenta la opción de crear una nueva base de datos.

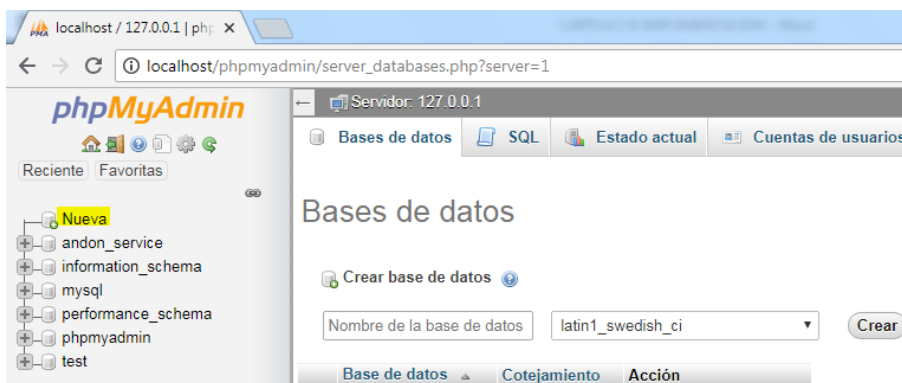



Figura. 4.16. Creación de base de datos

Elaborado por: Luis Tito

Ahora se procede a crear una tabla con los datos que se desea guardar en la base, como se muestra en la Figura 4.17. Se procede a nombrar la tabla en este caso se utiliza en nombre `slaves_info` y el número de columnas representa la estructura de datos que se desea elaborar, es decir, con 6 datos.

⚠ No se han encontrado tablas en la base de datos.

 Crear tabla

Nombre:  Número de columnas:

Figura. 4.17. Tabla de datos

Elaborado por: Luis Tito

En la Figura 4.18 se muestra la tabla creada con los nombres de los datos que se desean almacenar. Como se mencionó en el capítulo 3, los datos que se desea guardar son: número del slave, la estación donde se generó, el tipo de evento, fecha, hora y el último dato que se agregó es un código que permitirá desactivar la alerta generada.

Nombre de la tabla:  Agregar  columna(s)

Nombre	Tipo	Longitud/Valores	Predeterminado	Cotejamiento	Atributos
<input type="text" value="id_slave"/> <small>Seleccionar desde las columnas centrales</small>	CHAR	<input type="text" value="2"/>	Ninguno	<input type="text"/>	<input type="text"/>
<input type="text" value="estacion"/> <small>Seleccionar desde las columnas centrales</small>	VARCHAR	<input type="text" value="100"/>	Ninguno	<input type="text"/>	<input type="text"/>
<input type="text" value="evento"/> <small>Seleccionar desde las columnas centrales</small>	VARCHAR	<input type="text" value="100"/>	Ninguno	<input type="text"/>	<input type="text"/>
<input type="text" value="hora"/> <small>Seleccionar desde las columnas centrales</small>	TIME	<input type="text" value="10"/>	Ninguno	<input type="text"/>	<input type="text"/>
<input type="text" value="fecha"/> <small>Seleccionar desde las columnas centrales</small>	DATE	<input type="text" value="10"/>	Ninguno	<input type="text"/>	<input type="text"/>
<input type="text" value="codigo"/> <small>Seleccionar desde las columnas centrales</small>	VARCHAR	<input type="text" value="20"/>	Ninguno	<input type="text"/>	<input type="text"/>

Figura. 4.18. Parámetros de la tabla de datos

Elaborado por: Luis Tito

El tipo de variable y la longitud depende del dato que se desea almacenar. En el caso del ID del slave se toma un tipo de variable **CHAR** debido a que es un valor fijo de dos caracteres. Para el caso de la estación, el evento y el código se optó por una variable tipo **VARCHAR** por ser caracteres de longitud variable.

Y en el caso de la fecha y la hora se toman tipos de variables dedicadas a la información que se va a almacenar. En el caso de la hora una variable tipo **TIME**, mientras que para la fecha se emplea una variable del tipo **DATE**.

En la figura 4.19 se muestra como queda estructurada la tabla. Se puede ver que se muestran las configuraciones iniciales del tipo de variables y la longitud. Adicionalmente se muestra un campo de **cotejamiento**, esto no tiene que ver con el almacenamiento de datos, solo me indica tipo de escritura que se mostrará en la tabla.



#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra	Acción
<input type="checkbox"/>	1	id_slave	char(2)	latin1_swedish_ci	No	Ninguna			Cambiar
<input type="checkbox"/>	2	estacion	varchar(100)	latin1_swedish_ci	No	Ninguna			Cambiar
<input type="checkbox"/>	3	evento	varchar(100)	latin1_swedish_ci	No	Ninguna			Cambiar
<input type="checkbox"/>	4	hora	time		No	Ninguna			Cambiar
<input type="checkbox"/>	5	fecha	date		No	Ninguna			Cambiar
<input type="checkbox"/>	6	codigo	varchar(20)	latin1_swedish_ci	No	Ninguna			Cambiar

Figura. 4.19. Estructura de la tabla de datos

Elaborado por: Luis Tito

Para brindar protección a los datos se crea un usuario y una contraseña como se ilustra en la figura 4.20. El nombre de usuario que se va a utilizar es **andon\_service** y la contraseña es **service2018**, el nombre del host es **localhost**.

## Agregar cuenta de usuario

Información de la cuenta

Nombre de usuario: Use el campo de texto ▼ andon\_service

---

Nombre de Host: Local ▼ localhost 🔗

---

Contraseña: Use el campo de texto ▼ ..... Strength:  Muy débil

---

Debe volver a escribir: .....|

---

Complemento de autenticación: Autenticación de MySQL nativo ▼

---

Generar contraseña: Generar

Figura. 4.20. Seguridad de la tabla

Elaborado por: Luis Tito

La información que se presenta en esta tabla está disponible para los técnicos de Herramientales y debido a ello para la creación del usuario de la base de datos se debe configurar con todos los privilegios, como se muestra en la Figura 4.21.

Privilegios globales  Seleccionar todo

Nota: Los nombres de los privilegios de MySQL están expresados en inglés.

Datos

- SELECT
- INSERT
- UPDATE
- DELETE
- FILE

Estructura

- CREATE
- ALTER
- INDEX
- DROP
- CREATE TEMPORARY TABLES
- SHOW VIEW
- CREATE ROUTINE
- ALTER ROUTINE
- EXECUTE
- CREATE VIEW
- EVENT
- TRIGGER

Administración

- GRANT
- SUPER
- PROCESS
- RELOAD
- SHUTDOWN
- SHOW DATABASES
- LOCK TABLES
- REFERENCES
- REPLICATION CLIENT
- REPLICATION SLAVE
- CREATE USER

Límites de recursos

Nota: si cambia los parámetros de estas opciones a 0 (cero), remueve el límite.

MAX QUERIES PER HOUR

MAX UPDATES PER HOUR

MAX CONNECTIONS PER HOUR

MAX USER\_CONNECTIONS

Figura. 4.21. Privilegios globales

Elaborado por: Luis Tito

### 4.2.3 SERVICIOS WEB

Un servicio web es un sistema diseñado para establecer interacción entre dos puntos o clientes mediante el uso de la red de internet. Emplea lenguajes de programación de tipo estándar lo que le permite interactuar con distintos entornos de desarrollo. Para la base de datos se deben crear los siguientes servicios web: un **POST** y un **GET** (W3C, 2018).

Un servicio de tipo **POST** permite ingresar los datos a la base mediante el código de programación en **PHP**, para ello se debe crear el archivo que va a contener la programación del servicio web. En la Figura 4.22 se muestra la carpeta de destino con el nombre del archivo `insert_event`.

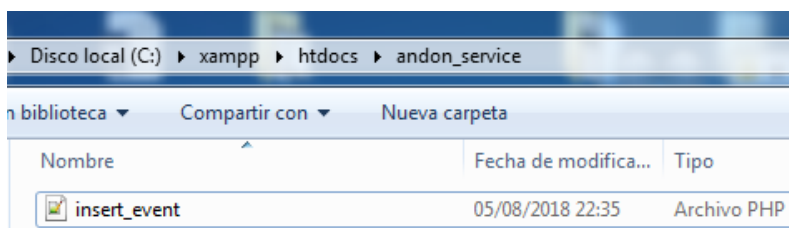


Figura. 4.22. Servicio Web Tipo POST

Elaborado por: Luis Tito

El primer servicio web del tipo **POST** se desarrolla en el entorno de **PHP** y contiene el código que se muestra en el Anexo 1. En este archivo se muestran la información del servidor y la forma en que va a guardar los datos en la tabla. Para verificar que el servicio web fue levantado correctamente se emplea el programa **Postman** como se muestra en la figura 4.23.



Figura. 4.23. Verificación de servicios web

Elaborado por: Luis Tito

En la barra de direcciones de este programa se escribe la ubicación de la carpeta que se creó anteriormente con el archivo que contiene la programación del servicio web de insertar evento. La cadena que recibe el `andon_service` es un JSON (JavaScript Object Notation), una forma de datos de formato muy ligero como se muestra en la Figura 4.24.

```
1 {"id_slave":"1234", "estacion":"alarmas", "evento":"herramientales", "hora":"14:15", "fecha":"2017-12-20", "codigo":"12312322"}
```

Figura. 4.24. Cadena de recepción tipo JSON

Elaborado por: Luis Tito

Una vez que se realiza un *request* en el Postman se recibe la notificación desde la base datos donde se muestra que la información generada por el servicio *web* fue ingresada correctamente en la base de datos. En la Figura 4.25 se muestra la respuesta del servidor con respecto a la información agregada.

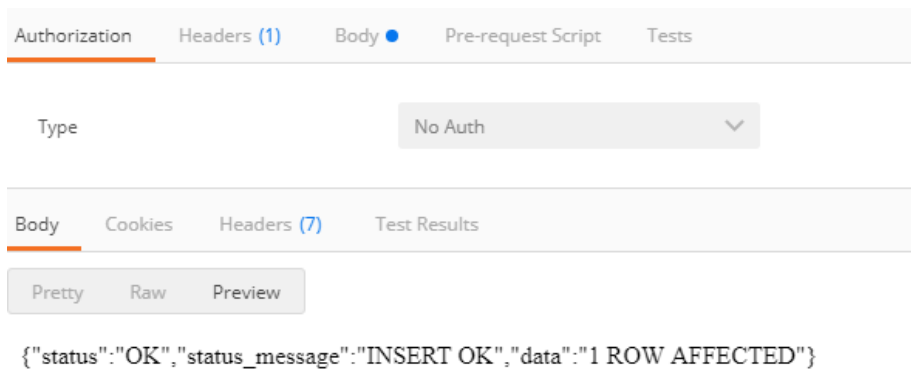


Figura. 4.25. Verificación con ayuda del Postman

Elaborado por: Luis Tito

Mientras tanto en el servidor se muestra que se agregó correctamente la información a la tabla de datos como se muestra en la Figura 4.26. Se puede observar que el evento cuenta con las características que se detallaron en el Capítulo 2, es decir, que cuenta con la información requerida.



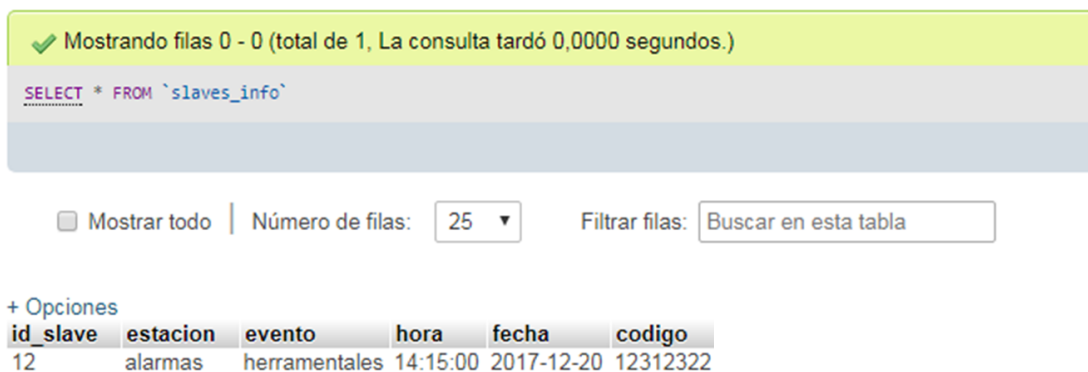


Figura. 4.26. Evento ingresado correctamente

Elaborado por: Luis Tito

Ahora para leer los eventos ingresados en la base de datos del servidor es necesario crear otro servicio *web*. Para ello es necesario elaborar un nuevo archivo con formato PHP que contenga la programación para visualizar los eventos que se agregaron a la base de datos. El nombre del archivo es `get_event.php` como se muestra en la Figura 4.27. Y cuya programación se muestra en el Anexo 2.

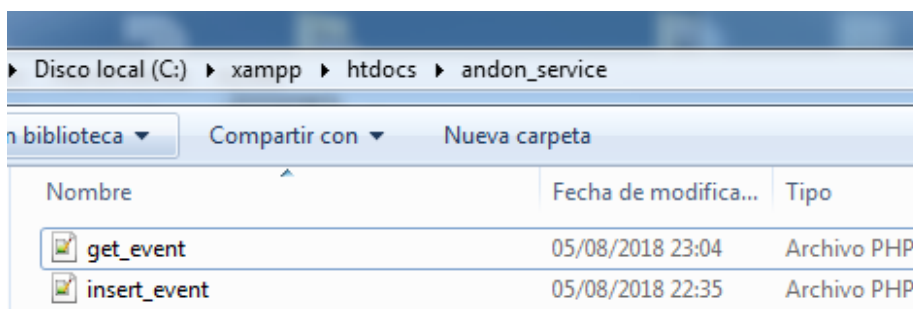


Figura. 4.27. Servicio web tipo GET

Elaborado por: Luis Tito

Esto devuelve un JSON con los datos de los eventos que se agregaron a la base de datos. Para verificar que el servicio GET funciona correctamente se debe abrir nuevamente el POSTMAN e ingresar la dirección de la carpeta donde se guardó el archivo como se muestra en la Figura 4.28.

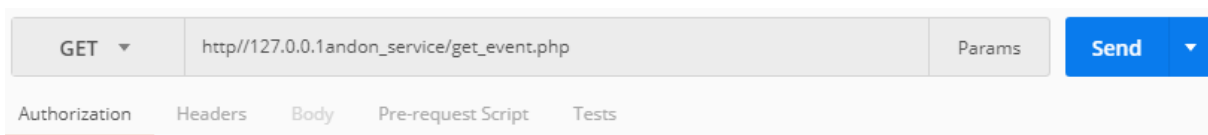


Figura. 4.28. Verificación servicio GET

Elaborado por: Luis Tito

Posteriormente se presiona **Send** para que devuelva la información que posee la base de datos con los eventos que se agregaron anteriormente. En esta parte deben intervenir los Módems ya que son los encargados de enviar y extraer los datos de la base. En la Figura 4.29 se muestra los datos que el módem receptor obtiene de la base en el servidor.



Figura. 4.29. Respuesta del servicio GET

Elaborado por: Luis Tito

Hasta el momento se dispone de dos servicios que permiten ingresar y visualizar los eventos generados por los *slaves* de las estaciones del departamento de producción, esta información se guarda en la base de datos del servidor que se creó anteriormente tal como se muestra en la Figura 4.30.



Figura. 4.30. Eventos guardados en la base de datos

Elaborado por: Luis Tito

#### 4.2.4 PROGRAMACIÓN ARDUINO

Para que los módems puedan realizar el ingreso y la eliminación de los datos del servidor necesitan un código de programación que permita realizar estas operaciones y para ello se emplea el Arduino Mega en conjunto con las Shield SIM-900. La programación de los Arduinos se muestra en el Anexo 5.

Tanto para el dispositivo que realiza la transmisión como el que realiza la recepción se desarrollan programaciones similares ya que ambos tienen la capacidad de enviar y recibir los datos de la base y va a depender de la actividad que realice el Modem junto con la SIM-CARD asignada al mismo.

La codificación de los Modems se realiza mediante comandos AT, un tipo de codificación que significa atención y fue desarrollado por la compañía Hayes Communications, esto permite enviar mensajes al Modem desde las librerías que se crearán posteriormente. Debido a esto se permite que el modem ejecute las operaciones desde la librería sin tener que accionarlo manualmente.

En el Anexo 4 se muestra la codificación con comandos AT que son enviados al modem, donde se visualiza los pasos que se ejecutan para conectarse con el servidor, esto es muy necesario ya que posteriormente permite que los datos obtenidos del *Andon* sean enviados a la base de datos que se configuró anteriormente.

Finalmente se debe realizar el entorno de programación que permitirá que los Modems realicen las funciones de transmisión de los eventos generados por el *Andon*. Esto se realiza en el programa Visual Studio y para ello se debe crear un proyecto con modelo de programación Visual C# como se ilustra en la Figura 4.31.

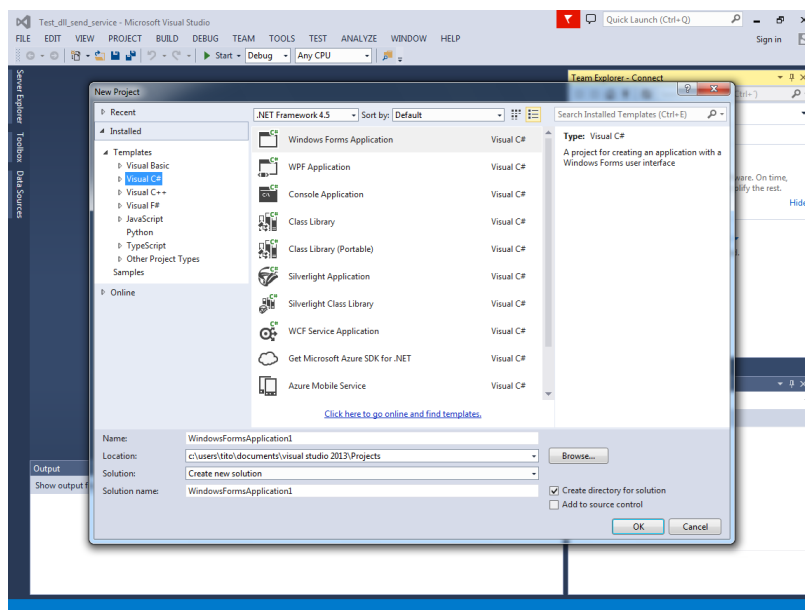


Figura. 4.31. Creación de las librerías para control de Modems

Elaborado por: Luis Tito

En esta ventana se debe desarrollar el programa para controlar los Modems mediante comandos AT y esto generará una librería que será añadida al Andon System. La programación de estas librerías ese muestra en los Anexos 5 y 6. Estos archivos contienen una subrutina donde se extraen los servicios Web creados en PHP.

Esto permite invocar a los tres servicios Web creados en la base de datos y permiten controlar los Modems para que sean los encargados transmitir las tramas del Andon y que posteriormente serán alojadas en la base creada en el servidor. Además se configura el acceso al envío de un mensaje de texto a un número telefónico de los técnicos de herramientas.

Esta librerías serán añadidas al programa que contiene la parte grafica del entorno que se propuso para la solución del problema. Para ello se debe crear un nuevo proyecto en el cual se desarrollará la parte grafica del sistema, es decir, lo que podrá ser visualizado por los técnicos encargados del soporte técnico.

En la Figura 4.32 se muestra como añadir las librerías al nuevo proyecto que permiten enviar la trama con los datos que genera cada uno de los *slaves*. Inicialmente se agregan el servicio de envío y posteriormente se debe agregar el servicio que permite borrar los datos ingresados en la base en un nuevo proyecto.

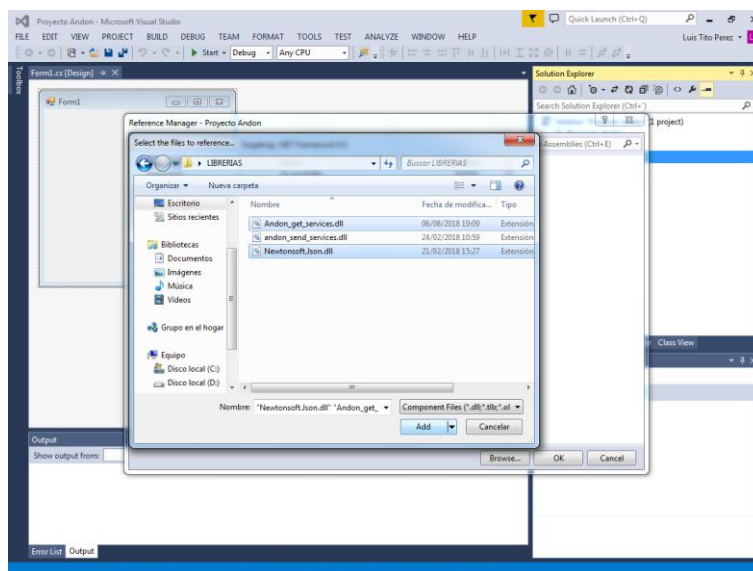


Figura. 4.32. Librerías añadidas al programa

Elaborado por: Luis Tito

Ahora en lo que se refiere a la parte grafica que contiene los datos que generan la alerta del Andon se debe añadir elementos como botones y casillas de texto que contengan la información del evento que invocó la alerta y el número telefónico del técnico encargado de brindar el soporte a las líneas de producción. Como se muestra en la Figura 4.33.

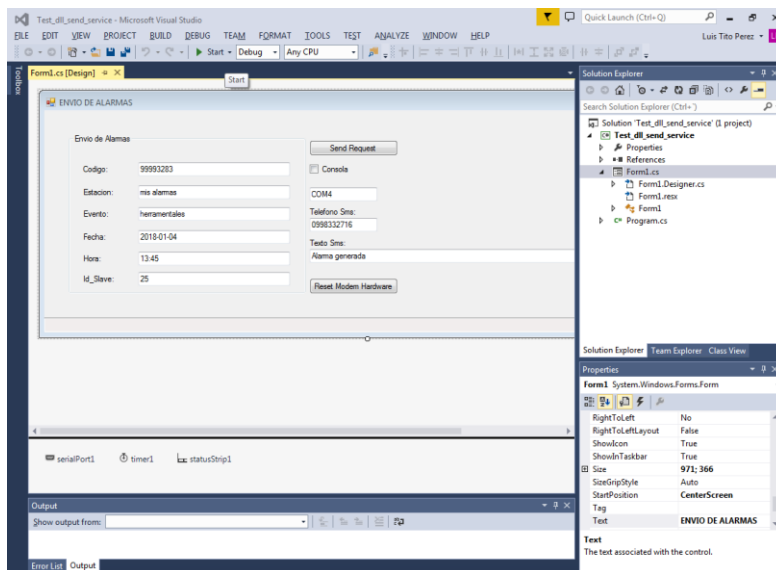


Figura. 4.33. Diseño de ventana de control

Elaborado por: Luis Tito

El mensaje que será visualizado en el computador del área de herramientas contiene la información del slave que lo activo y que será almacenado en la base de datos. Una vez que el dato es alojado en la base del servidor el modem se encarga de borrar el dato con un mensaje similar al del transmisor. Es por eso que los dos dispositivos son similares.

Para realizar esta operación es indispensable crear un nuevo proyecto en Visual Studio que contenga la biblioteca que permita borrar los datos que son alojados en la base del servidor. Y al igual que en el caso anterior se parte por añadir el archivo que contiene la librería que le permite actuar al Modem que será ubicado en el área de herramientas.

En la Figura 4.34 se muestra el entorno que se encargará de borrar el dato de la base con la posibilidad de que la ejecución de las acciones que se realizan para visualizarlas por medio de consola. Adicional se presenta un cuadro de dialogo que informa que los eventos han sido borrados correctamente.

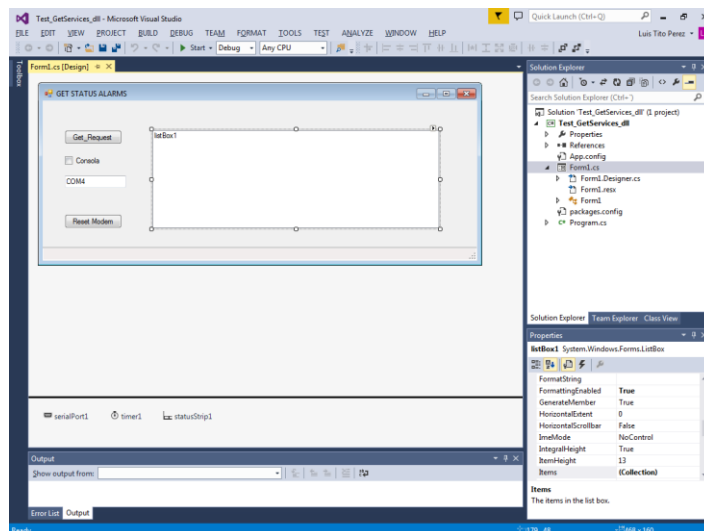


Figura. 4.34. Ventana de control eliminación de datos

Elaborado por: Luis Tito

### 4.3 PRUEBAS DE FUNCIONAMIENTO

El lugar escogido para la implementación es directamente en el departamento de producción ya que es ahí donde se generan las alertas del Andon. Mientras tanto el otro dispositivo será instalado en el área de herramientas para que sus técnicos puedan observar cuando se generan las alertas.

Para ello se debe conectar el Arduino a un puerto disponible del computador donde se encuentra alojado el Andon System. El voltaje suministrado por el puerto del computador no permite trabajar de manera correcta con el modem es por eso que se recurre a un adaptador de 5 Voltios para alimentar al Arduino.

Para realizar las primeras pruebas se debe contar con dos SIM CARD en los Modems para consumir los datos con los que cuentan. Se debe conectar el Arduino al computador tanto del departamento de Producción como en el de Herramientales con su respectivo adaptador de voltaje.

Debido a que los archivos que se crearon deben formar parte del Andon System se puede proceder a realizar la comunicación una vez que se genere la alerta de los *slaves*. La biblioteca creada en el sistema permitirá controlar los dos Modems que posteriormente enviarán esa señal para guardarla en la base de datos.

En la Figura 4.35 se muestra un *slave* conectado al sistema Andon y mediante el LCD se puede detectar su estado. Cuando uno de los botones es presionado debido a un incidente en la estación genera la alerta en el sistema y es recibido por el modem para transmitir esa información a la base de datos.



Figura. 4.35. Alerta generada por el Andon System

Elaborado por: Luis Tito

Cuando el modem recibe la información procede a ejecutar los comandos con los que está configurado a través del archivo que fue añadido al Andon System. En la figura 4.32 se muestra el dispositivo conectado al computador en el área de Producción. Una vez que envía la información a la base de datos se recibe un mensaje en el departamento de Herramientales.



### 4.3.1 PUESTA A PUNTO

Con el fin de realizar las pruebas de funcionamiento con los dispositivos que se pueden observar en la Figura 4.35 en el sitio se debe cumplir con los siguientes pasos:

Se debe seguir los siguientes pasos:

1. Conexión de los dispositivos
  - a. Conectar el adaptador de 9 VDC 1A a la entrada del dispositivo y posteriormente a la toma de 110 VCA para proveer de energía al dispositivo.
  - b. Verificar el LED de Modem Status hasta que el periodo de parpadeo sea de 1 segundo de encendido y tres segundos de apagado. Esto se realiza para verificar el nivel de la recepción de la operadora con la que está trabajando el dispositivo. Este periodo varía entre 10 segundos hasta 1 minuto.
  - c. Conectar el cable USB tipo B al dispositivo en el lugar indicado y el conector USB 2.0 a un puerto disponible del computador donde se encuentra instalado el programa Andon System.
  - d. Verificar el puerto con el que identifica el *tester* en el Administrador de dispositivos.



Figura. 4.36. Elementos del proyecto

Elaborado por: Luis Tito

### 4.3.2 CONEXIÓN A LA BASE DE DATOS

- a. Ejecutar el aplicativo Test Receptor que se encuentra en el escritorio para verificar la conexión del dispositivo con la base de datos que se encuentra en el área de Sistemas, en dicha base se almacenan los datos que transmiten los eventos que se generan en cada estación.
- b. Una vez iniciado el aplicativo se debe cambiar el puerto con el que está trabajando el dispositivo por ejemplo: COM7 como se muestra en la Figura 4.37.

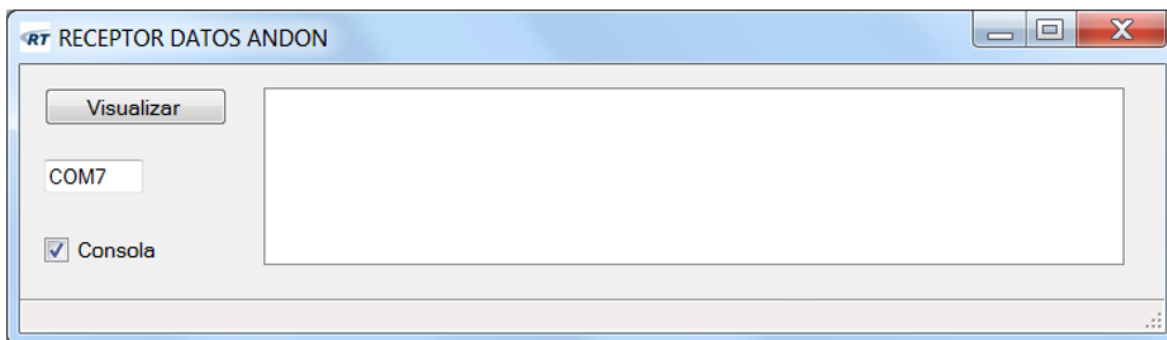


Figura. 4.37. Software de verificación de conexión a la base de datos

Elaborado por: Luis Tito

- c. Adicional se debe poner check en el recuadro de consola para verificar la ejecución de los comandos AT y de esa manera comprobar el funcionamiento del modem. La información que aparece en la ventana de consola muestra como se establece la comunicación desde el modem hacia la base de datos.
- d. Dar clic en la pestaña Visualizar para obtener el último registro que fue ingresado en la base de datos. En la parte inferior de la ventana aparece el mensaje Procesando Request, lo que nos indica que la solicitud a la base de datos se está ejecutando. Esto se puede observar en la ventana de consola.

- e. Esperar 20 segundos aproximadamente hasta que en la parte inferior de la ventana se despliegue el mensaje Proceso OK, y se despliegue la información del último evento registrado en la base.
- f. Durante el tiempo en el que la información se envía se debe verificar el comportamiento del LED EVENT RECEIVED STATUS. Este LED realiza 2 parpadeos por cada respuesta del modem en cada tramo del establecimiento de la comunicación mediante comandos AT. (8 tramos)
- g. En caso de que se no se pueda establecer la conexión con la base de datos el modem deja de enviar respuestas y el LED deja de parpadear, si se presenta este inconveniente se debe desconectar el dispositivo del puerto de comunicación y de alimentación de voltaje posteriormente volver al paso 1.

### 4.3.3 FUNCIONAMIENTO DEL TRANSMISOR

1. El dispositivo se encuentra ubicado en la estación que maneja el Andon System como se puede observar en la Figura 4.38, este no afecta a las labores habituales y es un lugar preciso para verificar la conexión hacia la base de datos.

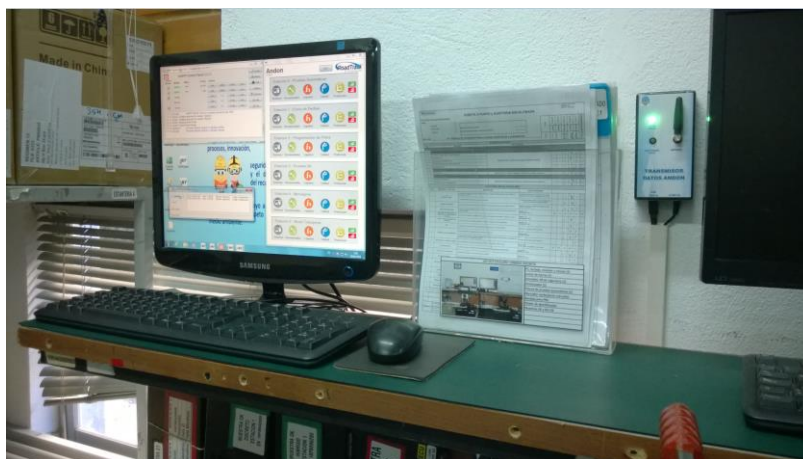


Figura. 4.38. Ubicación del transmisor de datos Andon

Elaborado por: Luis Tito



Figura. 4.39. Transmisor de datos Andon

Elaborado por: Luis Tito

2. Iniciar la aplicación Andon\_supervisor\_winform que se encuentra en el escritorio similar a la versión que maneja el Andon System. En la Figura 4.40 se muestra la ventana de inicio del programa.



Figura. 4.40. Condiciones iniciales de funcionamiento

Elaborado por: Luis Tito

3. Verificar el status de cada uno de los slaves, esto quiere decir que se encuentren funcionales y a punto en caso de producirse un inconveniente en la estación de trabajo de la línea de Producción
4. En el momento de producirse un inconveniente relacionado con el área de Herramientales el operador debe accionar el slave indicando que en la estación se está produciendo una anomalía. Posteriormente el Supervisor se acerca al sitio a constatar el inconveniente.
5. Mientras tanto en el computador donde se está ejecutando el programa se genera una ventana similar a la que se presenta en la Figura 4.41, donde se recoge la información acerca del incidente producido en la estación.

TRANSMISOR DATOS ANDON

Datos del evento

Id\_Slave: 70

Hora: 16:10

Fecha: 2019-02-27

Estacion: Estacion5

Evento: Herramientales

Codigo: |

Enviar evento

Consola

COM10

Telefono Sms: 0998332716

Texto Sms: Alarma generada en la estacion

Enviar Sms

Reiniciar modem

toolStripStatusLabel1

Figura. 4.41. Datos del incidente producido en la estación

Elaborado por: Luis Tito

6. Después de constatar el inconveniente el Supervisor debe escribir el nombre de la estación en la casilla de código. Esto es muy importante ya que es un dato que se muestra en el aplicativo de recepción de datos ubicado en el área de Herramientales.
7. Una vez completados los datos se debe dar clic en la pestaña de Enviar evento para subir la información a la base de datos.

8. En la parte inferior de la ventana se despliega el mensaje Subiendo información..., indicando el estado del envío de los datos del incidente, tal como se muestra en la Figura 4.42.

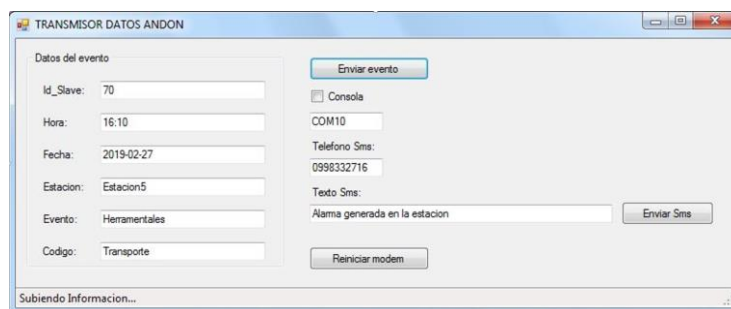


Figura. 4.42. Estado del envío de la información

Elaborado por: Luis Tito

9. Mientras se envía la información se visualiza el estado del envío a través del LED EVENT SENDING STATUS, como se menciona en el literal (f) del paso 2.
10. Una vez que la información es almacenada en la base de datos se muestra el mensaje: Evento Ingresado Correctamente, como se puede apreciar en la Figura 4.43, constatando la conexión hacia la base de datos ubicada en el área de Sistemas.

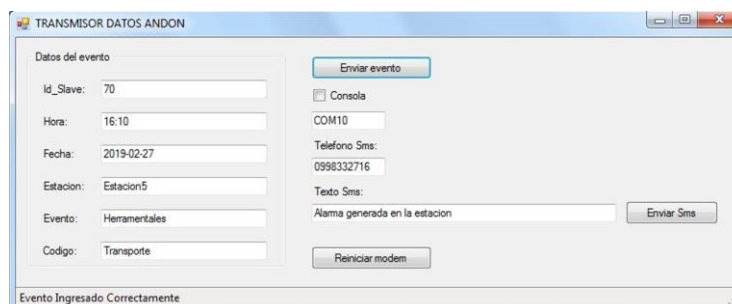


Figura. 4.43. Evento ingresado correctamente

Elaborado por: Luis Tito

11. Posteriormente el Supervisor tiene la posibilidad de enviar un mensaje de texto a la persona encargada del soporte al área de Producción con una idea rápida acerca del inconveniente producido en la línea. Este mensaje puede incluir hasta 160 caracteres y son enviados a un número telefónico que puede ser ingresado manualmente. Para ello debe llenar el texto en la casilla correspondiente y una vez terminada la redacción dar clic en la pestaña Enviar mensaje.
12. En la Figura 4.44 se muestra la respuesta del programa una vez que el mensaje ha sido enviado al destinatario.



Figura. 4.44. Estado del mensaje de texto

Elaborado por: Luis Tito

13. Una vez que se envía el mensaje el supervisor debe cerrar la ventana que contiene la información acerca del evento y posteriormente debe desactivar la alarma normalmente, es decir, ingresando el usuario y la contraseña. Como se muestra en la Figura 4.45.

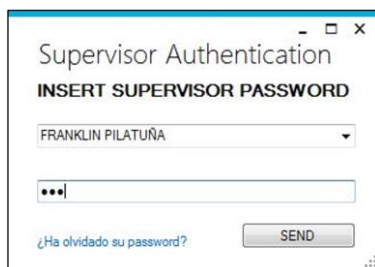


Figura. 4.45. Desactivación de alarmas

Elaborado por: Luis Tito

#### 4.3.4 FUNCIONAMIENTO DEL RECEPTOR

1. Para esta parte se cuenta con un dispositivo de similares características al transmisor y para iniciar la operación se debe realizar los pasos de puesta a punto mencionados anteriormente en los pasos 1 y 2 como se muestra en la Figura 4.46.

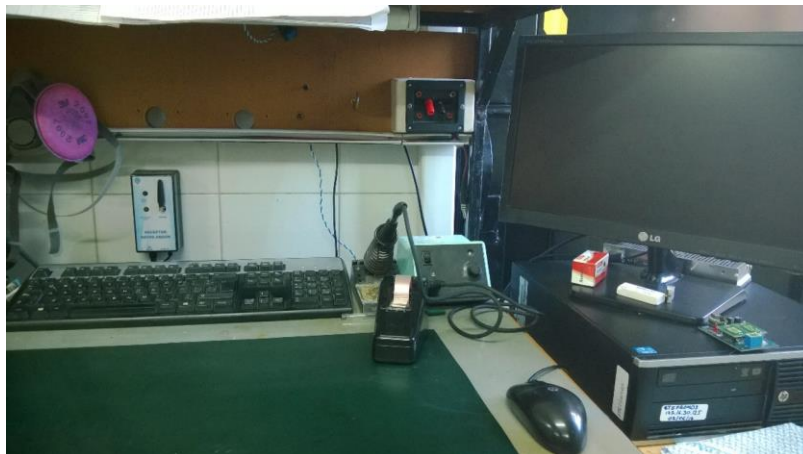


Figura. 4.46. Ubicación dispositivo Receptor de datos Andon

Elaborado por: Luis Tito



Figura. 4.47. Receptor de datos Andon

Elaborado por: Luis Tito



2. En el área de Herramientales se cuenta con el aplicativo que pertenece al Andon System que se muestra en la Figura 4.48, esta ventana permite visualizar el momento en que se produce un incidente en la línea de Producción.

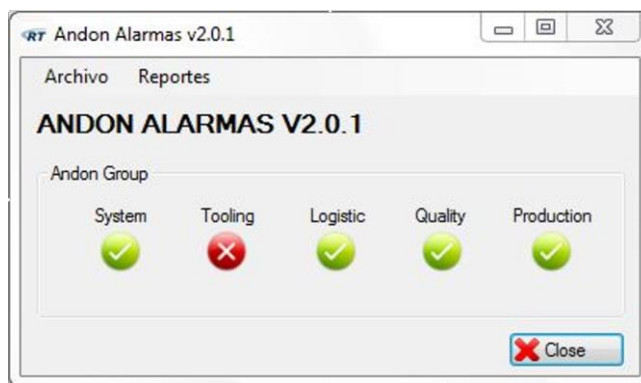


Figura. 4.48. Visor de alarmas Andon System

Elaborado por: Luis Tito

3. El aplicativo que permite visualizar los datos en el receptor se denomina Receptor Datos Andon y se encuentra en el escritorio como acceso directo. En la Figura 4.49 se puede apreciar el aplicativo que al igual que en el transmisor se debe elegir un puerto de comunicación antes de entrar en funcionamiento.

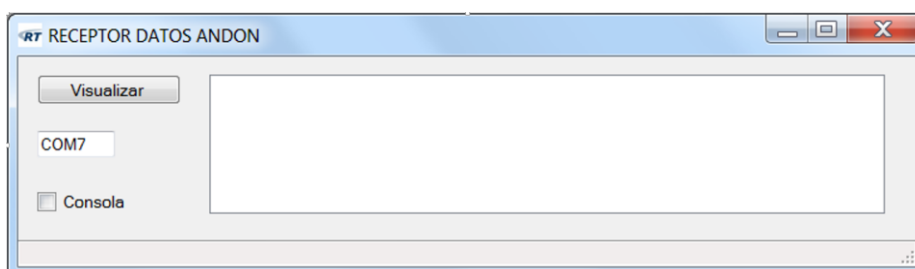


Figura. 4.49. Receptor de datos Andon

Elaborado por: Luis Tito

- Una vez que se visualiza la alerta en el área de Herramientales se debe accionar la pestaña Visualizar para que el dispositivo receptor extraiga la información de la base con los datos del evento que se produjo en el área de Producción.
- En la Figura 4.50 se observa el mensaje Procesando Request... en la parte inferior del aplicativo y al igual que en el caso del transmisor, se debe esperar aproximadamente 20 segundos para obtener la visualización de los datos del evento.



Figura. 4.50. Estado de recepción del evento

Elaborado por: Luis Tito

- De igual manera el LED EVENT RECEIVED STATUS indica el estado del proceso de comunicación del dispositivo receptor con la base de datos. Finalmente el aplicativo muestra el resultado con los datos que se obtienen de los slaves: Fecha, Hora, Estación y Código como se puede apreciar en la Figura 4.51.



Figura. 4.51. Información extraída de la base de datos

Elaborado por: Luis Tito

Se debe mencionar que la información está ingresando en un servidor físico que se encuentra en el área de Sistemas de la empresa. En la Figura 4.52 se muestra la imagen del servidor que contiene la base de datos.



Figura. 4.52. Servidor ubicado en el área de Sistemas

Elaborado por: Luis Tito

#### **4.4 MEDICIONES**

Para realizar la medición de datos del sistema se realizaron 20 pruebas tanto al transmisor como al receptor en distintos tiempos para verificar que los Modems funcionen de manera adecuada, ya que se puede observar que pueden sufrir intermitencias al momento de realizar la comunicación.

Para evidenciar que los dispositivos trabajen de manera óptima se realizaron mediciones para verificar su funcionamiento. En la tabla 4.1 se muestran los datos obtenidos de las pruebas realizadas al transmisor en distintos promedios de tiempo en donde se evidencia que en ciertos momentos se pueden presentar intermitencias en el modem.

Tabla. 4.1. Pruebas realizadas en el transmisor

Numero de prueba	Resultado		Periodo de tiempo	Observaciones
		No deseado		
1	X		60 seg	
2		X		
3	X			
4	X			
5	X		30 seg	
6		X		
7	X			
8		X	15 seg	
9	X			
10	X			
11	X			
12	X		10 seg	
13	X			
14	X			
15	X			
16	X		5 seg	
17	X			
18	X			
19	X			
20		X		Reinicio de modem

Elaborado por: Luis Tito

En la tabla 4.2 se muestran los datos obtenidos de las pruebas realizadas en el receptor

Tabla. 4.2. Pruebas realizadas en el receptor

Numero de prueba	Resultado		Periodo de tiempo	Observaciones
	Satisfactorio	No deseado		
1		X	60 seg	
2		X		
3	X			
4	X			
5	X		30 seg	
6	X			
7	X			
8	X			
9		X	15 seg	
10		X		
11	X			
12	X			
13	X		10 seg	
14	X			
15		X		
16	X			
17	X		5 seg	
18	X			Reinicio de modem
19	X			
20		X		

Elaborado por: Luis Tito

#### 4.5 ANÁLISIS DE RESULTADOS

Los datos obtenidos de las pruebas del funcionamiento de los dispositivos muestran que el sistema presenta momentos en que el modem debe ser reiniciado. En el grafico 4.53 se puede observar que de 20 pruebas realizadas el modem responde en 16 veces de manera satisfactoria, es decir, un 80 %. Por otra parte se puede ver que 4 de las 20 pruebas realizadas no son satisfactoria, es decir, un 20 %.

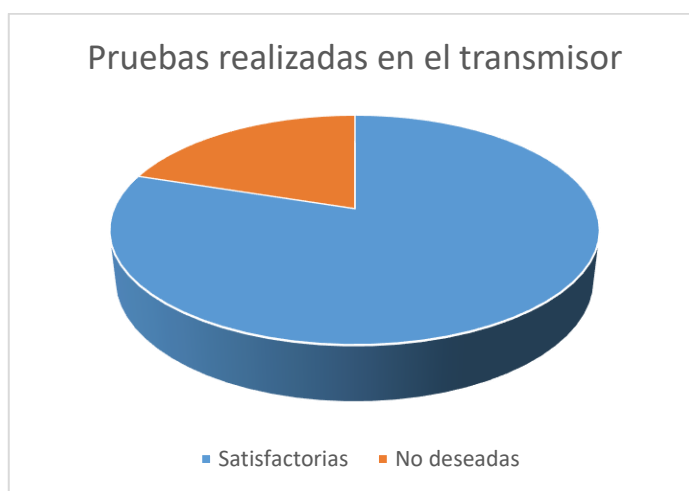


Figura. 4.53. Información extraída de la base de datos

Elaborado por: Luis Tito

Por otra parte se realizan las pruebas en el receptor con lo que se obtuvieron los siguientes resultados: de 20 pruebas realizadas 15 se producen de manera satisfactoria, es decir, un 75%. Por otra parte se producen 5 pruebas no deseadas lo que equivales a 25%. Como se puede observar en la Figura 4.54.

El sistema genera un promedio de dos alarmas al día que son brevemente atendidas por el personal encargado. Se verifica que el tiempo que tardan los técnicos de herramientas en brindar la respectiva asistencia a las líneas se ha reducido y por ende las métricas del área de Producción se cumplen sin sobresaltos



Figura. 4.54. Información extraída de la base de datos

Elaborado por: Luis Tito

Los elementos diseñados funcionan de manera correcta y el sistema se debe implementar en las actividades diarias del departamento. Se puede evidenciar que la facultad de enviar mensajes a los técnicos de Herramientales es un buen valor agregado ya que en ocasiones el personal no se encuentra en el área de trabajo.

Debido a que los chips consumen saldo se puede establecer la cantidad de datos que se envían cada vez que se generan las alertas, esta información es importante ya que permite realizar la adquisición de un paquete de datos que brinde la posibilidad de mantener la red de una manera permanente.

Al momento de realizar las pruebas el SIM del transmisor cuenta con 299.90 Megas para el consumo de datos y después de 20 pruebas consumieron 0,02 Mb, es decir, 20 Kb por lo que se puede decir que por cada evento enviado se produce un gasto de 1 Kb.

De igual manera en el receptor se realiza la medición de consumo con un valor inicial de 299,95 Mb para la transmisión de datos, y posterior a las 20 pruebas realizadas se obtuvo una diferencia de 299,93 Mb, es decir, se consumieron 0,02 Mb, lo que equivale a 20 Kb. Al igual que en el caso anterior se puede concluir que por cada evento ingresado en la base se reduce una tasa de 1 Kb.

## CONCLUSIONES

El ensamblaje de los dispositivos de transmisión y recepción tuvo su mediana dificultad, debido a que las características físicas de las tarjetas no permiten combinarlas por lo que fue necesario solucionar este inconveniente con la elaboración de un PCB en una placa. Esto permitió unir a las dos tarjetas de una manera sencilla que utilizó únicamente los pines importantes que se tomaron en cuenta para la solución del problema.

Para evitar que se generen intermitencias en el funcionamiento de los Modems se tomó la decisión de alimentar al Arduino Mega con una fuente externa de 5 Voltios y evitar aplicar el voltaje que genera el computador al estar conectado por medio del puerto de comunicación ya que el modem requiere más cantidad de corriente.

La elaboración de librerías es un proceso bastante complicado ya que se trata de hacerlo lo más simplificado posible para evitar que se produzcan daños en el programa nativo. Debido a ello se realizó los servicios Web de manera independiente para mantener un orden en el funcionamiento y evitar que se generen confusiones al momento de guardar y extraer información de la base de datos.

La programación de los dos Arduinos se realizó de manera similar ya que las librerías creadas en el Virtual Studio son las encargadas de designar a las tarjetas los nombramientos de tarjeta transmisora y receptora.

La manera más adecuada de crear entornos de programación sencillos se debe a que las librerías permiten englobar la mayor cantidad de información en forma de subrutinas que consumen menor cantidad de espacio del archivo.



Los comandos AT son muy importantes ya que de estos depende que la comunicación entre el modem y la base de datos se realice de manera correcta. Es por ello que se tomaron en cuenta los comandos que se acoplan a la solución del problema.

El ensamblaje de los modem con el arduino debe ser muy seguro ya que el exceso de manipulación puede generar que se presenten desprendimientos en el interior del transmisor y con ello generar intermitencias en su funcionamiento.

La propuesta se pudo implementar de forma adecuada ya que se cumplió con el requerimiento inicial, el cual era comunicar dos áreas separadas y con la ventana a expandir la red debido a que su funcionalidad fue satisfactoria.

## RECOMENDACIONES

Con el fin de conseguir un mayor crecimiento en el proyecto se recomienda adquirir tarjetas SIM-900 con su respectivo Board ya que eso permitirá que los pines de conexión se adapten perfectamente a los puertos del Arduino. O en su defecto investigar la manera de suprimir el Arduino ya que únicamente emplea los puertos de transmisión en el caso de que no se requieran generar alertas que necesiten emplear distintas salidas del microcontrolador.

Generar un archivo compacto que permita alojar todas las características del programa, ya que al hacerlo por separado es posible perder información o que en su defecto se omitan ciertas características que son importantes para el funcionamiento del sistema.

Exportar de forma minuciosa el servidor de la base de datos ya que el evitar parámetros de configuración genera que el destino de los datos no se encuentre en la red lo que provoca bloqueos en la conexión entre los dos puntos de transmisión.

Desarrollar un entorno que permita enviar correos electrónicos con la información acerca de los eventos que se generan en el departamento de producción con el fin de generar reportes para establecer un control en el mantenimiento de los mecanismos que permiten elaborar las pruebas funcionales a los equipos.

Elaborar una tabla de datos con novedades registradas en los demás departamentos que tienen estrecha relación con el área de producción y hacer que se guarden en el servidor de la misma manera que se realizó con el área de herramientas.

El ensamblaje y la implementación de los transmisores deben ejecutarse en una zona donde no exista mayor manipulación por parte de personal no autorizado ya que se corre el riesgo de provocar daños en la antena externa o peor aún en el interior de la carcasa que cubre y protege al modem.

Verificar que el funcionamiento de las SIM-CARD sea el más adecuado antes de colocarlo en la en la tarjeta ya que se corre el riesgo de que se desactive de la red de la operadora contratada.

La parte de la implementación final deber ser desarrollada por el personal encargado de manipular las funcionalidades que tiene el Andon System. Debido a ello se cuenta con la información necesaria para acoplar los Modems a la operación que se buscó con esta propuesta.

## BIBLIOGRAFÍA

Sánchez, Juan. (2005) Análisis y estudios de Redes GPRS. Valdivia

Universidad Internacional de Valencia VIU (2018): Sistemas GPRS, concepto principal y usos principales. Disponible en web: <https://www.universidadviu.com/sistemas-gprs-concepto-general-usos-principales/> [Consulta: 25 de Junio del 2018]

Micayael (2018): Métodos GET y POST del HTTP. Disponible en web: <http://blog.micayael.com/2011/02/09/metodos-get-vs-post-del-http/> [Consulta: 24 de Junio del 2018]

IBM (2016): Conceptos de los servicios web JSON. Disponible en web: [https://www.ibm.com/support/knowledgecenter/es/SSGMCP\\_5.1.0/com.ibm.cics.ts.mobileextensions.doc/concepts/concepts\\_json.html](https://www.ibm.com/support/knowledgecenter/es/SSGMCP_5.1.0/com.ibm.cics.ts.mobileextensions.doc/concepts/concepts_json.html) [Consulta: 27 de Junio del 2018]

Programando (2015): Tipos de datos en MySQL. Disponible en web: <https://advenis.wordpress.com/2010/04/21/tipos-de-datos-en-mysql/> [Consulta: 21 de Junio del 2018]

IBM (2016): Servicios Web. Disponible en web: [https://www.ibm.com/support/knowledgecenter/es/SS7K4U\\_9.0.0/com.ibm.webspHERE.zseries.doc/ae/cwbs\\_wbs2.html](https://www.ibm.com/support/knowledgecenter/es/SS7K4U_9.0.0/com.ibm.webspHERE.zseries.doc/ae/cwbs_wbs2.html) [Consulta: 25 de Julio del 2018]

Entel (2018): GPRS. Disponible en web: [http://personas.entel.cl/PortalPersonas/appmanager/entelpcs/personas?\\_nfpb=true&\\_pageLabel=P11800567291273156038130](http://personas.entel.cl/PortalPersonas/appmanager/entelpcs/personas?_nfpb=true&_pageLabel=P11800567291273156038130) [Consulta: 25 de Julio del 2018]

GsmSpain (2018): Como funciona la red GPRS. Disponible en web:  
[https://www.gsmSpain.com/info\\_tecnica/gprs/index.php](https://www.gsmSpain.com/info_tecnica/gprs/index.php) [Consulta: 23 de Junio del 2018]

Instructables (2018): Sim-900 GSM GPRS Shield con Arduino. Disponible en web:  
<https://www.instructables.com/id/SIM900-GSM-GPRS-SHIELD-CON-ARDUINO-UNO/> [Consulta: 23 de Junio del 2018]

Tecmikro (2018): Módulo GSM Sim-900 Shield Arduino. Disponible en web:  
<https://tecmikro.com/transmisores-receptores/483-modulo-gsm-sim900-shield-arduino.html> [Consulta: 12 de Junio del 2018]

# ANEXOS

## ANEXO 1 Programación GET

```

<?php
if(strcasecmp($_SERVER['REQUEST_METHOD'], 'POST') != 0){ //metodo verifica que el contenido sea tipo POST
    throw new Exception('Request method must be POST!');
}
//La siguiente linea de codigo lee el contenido de la cabecera del request
$contentType = isset($_SERVER["CONTENT_TYPE"]) ? trim($_SERVER["CONTENT_TYPE"]) : '';
if(strcasecmp($contentType, 'application/json') != 0){
    throw new Exception('Content type must be: application/json'); //verifica que el contenido sea JSON
}
$content = trim(file_get_contents("php://input")); //toma el contenido que viene en el post
$decoded = json_decode($content, true); //decodifica el JSON a array
if(!is_array($decoded)){
    throw new Exception('Received content contained invalid JSON!'); //verifica que sea un array
}

/* echo "el numero de slave es: ".$decoded["id_slave"]."<br>";
echo "la estacion es: ".$decoded["estacion"]."<br>";
echo "el tipo de evento es: ".$decoded["evento"]."<br>";
echo "Hora: ".$decoded["hora"]."<br>";
echo "Fecha: ".$decoded["fecha"]."<br>";
echo "el codigo para desactivar la alerta es: ".$decoded["id_slave"]."<br>"; */

try
{
//Insertar datos en la base

$servername = "127.0.0.1"; //variable tipo string
$username = "andon_service"; //variable tipo string
$password = "service2018"; //variable tipo string
$dbname = "andon_service"; //nombre de la base de datos en MySQL
$conn = new mysqli($servername, $username, $password, $dbname); //conexion entre PHP y MySQL

//La siguiente fila verifica que los datos obtenidos son los correctos: servername, username, password, dbname
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error); //si lo datos son incorrectos envia el mensaje: Connection failed: ???
}
//echo "Conexion MySQL OK"; //muestra que la conexion se realizo correctamente

$id_slave=$decoded['id_slave'];
$estacion=$decoded['estacion'];
$evento=$decoded['evento'];
$hora=$decoded['hora'];
$fecha=$decoded['fecha'];
$codigo=$decoded['codigo'];

// tomamos los valores del array convertido para hacer la SQL

$sql = "INSERT INTO slaves_info (id_slave, estacion, evento, hora, fecha, codigo) VALUES ('$id_slave','$estacion','$evento

echo "La query es: ".$sql; //el query que genera el codigo php

if ($conn->query($sql) === TRUE) //permite enviar el query: si esta bien redactado permite armar el siguiente array
{
$response['status']="OK";
$response['status_message']="INSERT OK";
$response['data']="1 ROW AFFECTED";

$json_response = json_encode($response); //se convierte el array en un JSON
echo $json_response;
} else // caso contrario armar un array con la respuesta error
{
$response['status']="Error";
$response['status_message']=""+$sql+"Error: "+$conn->error;
$response['data']="0 ROW AFFECTED";
$json_response = json_encode($response); //se convierte el array en un JSON
echo $json_response;
}
$conn->close();

} catch(Exception $e)
{
//Que hacer si no se ingresa el dato correctamente
}

?>

```

## ANEXO 2 Programación POST

```

<?php
try
{
get_data(); //llama a la funcion
} catch (Exception $e)
{}
//funcion
function get_data()
{
//variables del programa
$servername = "127.0.0.1"; //variable tipo string
$username = "andon_service"; //variable tipo string
$password = "service2018"; //variable tipo string
$dbname = "andon_service"; //variable tipo string

$conn = new mysqli($servername, $username, $password, $dbname); //conector entre el php y el MySQL
if ($conn->connect_error) { //verifica que los datos obtenidos son los correctos: servername, username, password, dbname
die("Connection failed: " . $conn->connect_error); //si lo datos son incorrectos envia el mensaje: Connection failed: ???
}
$sql = "SELECT * FROM slaves_info"; //permite seleccionar de slaves_info ya guardar en la variable sql
$result = $conn->query($sql); //es una variable que va a contener la respuesta de la base de datos
if ($result->num_rows > 0) { //SI EL NUMERO DE FILAS ES MAYOR A CERO ENTRA AL WHILE

//bucle de verificacion de filas de la base de datos
$i=0;
while($row = $result->fetch_assoc()) {
$respuesta[$i]['id_slave']=$row["id_slave"];
$respuesta[$i]['estacion']=$row["estacion"];
$respuesta[$i]['evento']=$row["evento"];
$respuesta[$i]['hora']=$row["hora"];
$respuesta[$i]['fecha']=$row["fecha"];
$respuesta[$i]['codigo']=$row["codigo"];
$i=$i+1;
}
$json_response = json_encode($respuesta); //se convierte en un array de json dependiendo del numero de filas
echo $json_response;
} else {
echo "0 results"; //si no hay filas en la base me devuelve el valor de 0 results
}
$conn->close(); //cierra la conexion
}
?>

```



## ANEXO 3 Programación Arduino

```

✓ → 📄 ⬆️ ⬇️ Verificar
Programas $
//Variables globales

int rele=12;
int led=11;

//Valores predeterminados
void setup() {
  // initialize both serial ports:

  Serial.begin(9600); //iniciar comunicacion serial al computador
  Serial1.begin(9600); //iniciar comunicacion serial para comunicar los modems

  pinMode(rele, OUTPUT); //Definir pines como salida
  pinMode(led, OUTPUT); //Definir pines como salida

  for (int i=0; i <= 10; i++){ //bucle de inicio
    digitalWrite(led, HIGH);
    delay(100);
    digitalWrite(led, LOW);
    delay(100);
  }
}

//Bucle del proyecto

void loop() {

  // read from port 1, send to port 0:
  /* Los datos que envia el modem son receptados
  por el puerto 1 y enviados a l puerto 0 de la PC */

  if (Serial1.available()) {
    digitalWrite(led, HIGH);
    int inByte = Serial1.read();
    Serial.write(inByte);
  }

  // read from port 0, send to port 1:

  /* Los datos que se envian mediante la PC son captados
  por el puerto serial 0 y posteriormmete son enviados
  por el puerto 1 al modem*/

  if (Serial.available()) {
    digitalWrite(led, HIGH);
    int inByte = Serial.read();

  /*Esta parte permite reinicaira al modem en el caso de que se resetee*/
    if (inByte==1)
    {
      digitalWrite(rele, HIGH);
    }
    else
    {
      {
        if (inByte==2)
        {
          digitalWrite(rele, LOW);
        }
        else
        {
          Serial1.write(inByte);
        }
      }
    }
  }
  digitalWrite(led, LOW);
}

```

**ANEXO 4 Comandos AT**

AT+CGATT=1

Response: OK

Luego

AT+CSTT= "internet.movistar.com.ec", "movistar", "movistar"

Response: OK

Luego

AT+CIICR

Response: OK

AT+CIFSR

Response: Ip value

AT+CIPSTATUS

Response: OK – STATE: IP STATUS

AT+CIPHEAD=1

Response: OK

AT+CIPSTART= "TCP", "186.4.148.194", "8045"

Response: CONNECT OK

AT+CIPSEND

POST /andon\_service/insert\_event.php HTTP/1.1

Host: 186.4.148.194

Content-Length: 126

Content-Type: application/json

Accept: \*/\*

Accept-Encoding: gzip, deflate

Accept-Language: es-ES;es;q=0.9

(Dos enter 0D 0A)

## ANEXO 5 Librería GET para Andon

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using Andon_get_services;

namespace Test_GetServices_dll
{
    public partial class Form1 : Form
    {
        Get_service get_ser = null;
        String respuesta = "";
        String status = "";
        public Form1()
        {
            InitializeComponent();
        }

        private void get_servicio(slaves_data[] sla, String status)
        {
            Console.WriteLine("Estado de respuesta: " + status);
            for (int i = 0; i < sla.Length; i++)
            {
                Console.WriteLine("Valor " + sla[i].Codigo + ": " + sla[i].Estacion + " " + sla[i].Fecha + '
                respuesta = "Valor " + sla[i].Codigo + ": " + sla[i].Estacion + " " + sla[i].Fecha + " " + :
                this.status = status;
                this.Invoke(new EventHandler(displaytext));
            }
            get_ser.send_response -= new Get_service.response(get_servicio);
            get_ser.finalize();
            get_ser = null;
        }
        private void displaytext(object sender, EventArgs e)
        {
            listBox1.Items.Add("" + respuesta);
            toolStripStatusLabel1.Text = "" + this.status;
        }
        private void button1_Click(object sender, EventArgs e)
        {
            try
            {
                serialPort1.Close();

                serialPort1.PortName = txt_com.Text;

                if (checkBox1.Checked == true)
                    get_ser = new Get_service(serialPort1, timer1, true);
                else
                    get_ser = new Get_service(serialPort1, timer1, false);

                get_ser.Opera.Apn = "internet.movistar.com.ec";
                get_ser.Opera.User = "movistar";
                get_ser.Opera.Pass = "movistar";

                toolStripStatusLabel1.Text = "Procesando Request...";

                get_ser.Net.Ip = "186.4.148.194";
                get_ser.Net.Protocolo = "TCP";
                get_ser.Net.Puerto = "8045";

                get_ser.Rest.Host = "Host: 186.4.148.194";
            }
        }
    }
}

```

```
get_ser.Get_val = "GET /andon_service/get_events.php HTTP/1.1"; ;

/*get_ser.Slave.Codigo = "99993283";
get_ser.Slave.Estacion = "mis alarmas";
get_ser.Slave.Evento = "herramientales";
get_ser.Slave.Fecha = "2018-01-04";
get_ser.Slave.Hora = "13:45";
get_ser.Slave.Id_slave = "25";*/

get_ser.test_dll();
get_ser.send_response += new Get_service.response(get_servicio);
get_ser.get_alarm();
}catch(Exception ex)
{

}

}

private void button2_Click(object sender, EventArgs e)
{
    try
    {
        serialPort1.Open();
        get_ser.reset_modem_hardware();
    }
    catch (Exception ex)
    {

    }

}
}
```

## ANEXO 6 Librería SEND para Andon

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using andon_send_services;

namespace Test_dll_send_service
{
    public partial class Form1 : Form
    {
        Andon_service miclase = null;
        bool flag_ini = false;

        public Form1()
        {
            InitializeComponent();

            //miclase = new Andon_service(serialPort1, timer1, false);
        }

        private void button1_Click(object sender, EventArgs e)
        {
            try
            {
                serialPort1.Close();

                serialPort1.PortName = txt_com.Text;

                if (chk_consola.Checked == true)
                    miclase = new Andon_service(serialPort1, timer1, true);
                else
                    miclase = new Andon_service(serialPort1, timer1, false);

                toolStripStatusLabel1.Text = "Subiendo Informacion...";

                miclase.Opera.Apn = "internet.movistar.com.ec";
                miclase.Opera.User = "movistar";
                miclase.Opera.Pass = "movistar";

                miclase.Net.Ip = "186.4.148.194";
                miclase.Net.Protocolo = "TCP";
                miclase.Net.Puerto = "8045";

                miclase.Rest.Host = "Host: 186.4.148.194";
                miclase.Rest.Post = "POST /andon_service/insert_event.php HTTP/1.1";

                miclase.Slave.Codigo = txt_codigo.Text;
                miclase.Slave.Estacion = txt_estacion.Text;
                miclase.Slave.Evento = txt_evento.Text;
                miclase.Slave.Fecha = txt_fecha.Text;
                miclase.Slave.Hora = txt_hora.Text;
                miclase.Slave.Id_slave = txt_slave.Text;

                miclase.test_dll();

                miclase.send_response += new Andon_service.response(get_response);

                miclase.send_alarm();
            }
            catch { }
        }
    }
}

```

```
    }catch(Exception ex)
    {
        MessageBox.Show("error");
    }

}

private void get_response(String response)
{
    //Termino la libreria
    Console.WriteLine("Service Response: " + response);
    toolStripStatusLabel1.Text = ""+response;
    miclase.send_response -- new Andon_service.response(get_response);
    //timer1.Enabled = false;
    miclase.finalize();

    miclase = null;

}

private void button2_Click(object sender, EventArgs e)
{
    /*if (chk_consola.Checked == true)
        miclase = new Andon_service(serialPort1, timer1, true);
    else
        miclase = new Andon_service(serialPort1, timer1, false);*/
    try
    {
        serialPort1.Open();
        miclase.reset_modem_hardware();
    }catch(Exception ex)
    {

    }

}

private void btn_send_Click(object sender, EventArgs e)
{
    try
    {
        serialPort1.Close();

        serialPort1.PortName = txt_com.Text;

        if (chk_consola.Checked == true)
            miclase = new Andon_service(serialPort1, timer1, true);
        else
            miclase = new Andon_service(serialPort1, timer1, false);

        toolStripStatusLabel1.Text = "Enviando mensaje...";

        miclase.test_dll();
        if (miclase.send_message(txt_telefono.Text, txt_mensaje.Text) == true)
            MessageBox.Show("Mensaje enviado");
        else
            MessageBox.Show("Mensaje No enviado");

        miclase.finalize();

        miclase = null;
    }catch(Exception ex)
    {

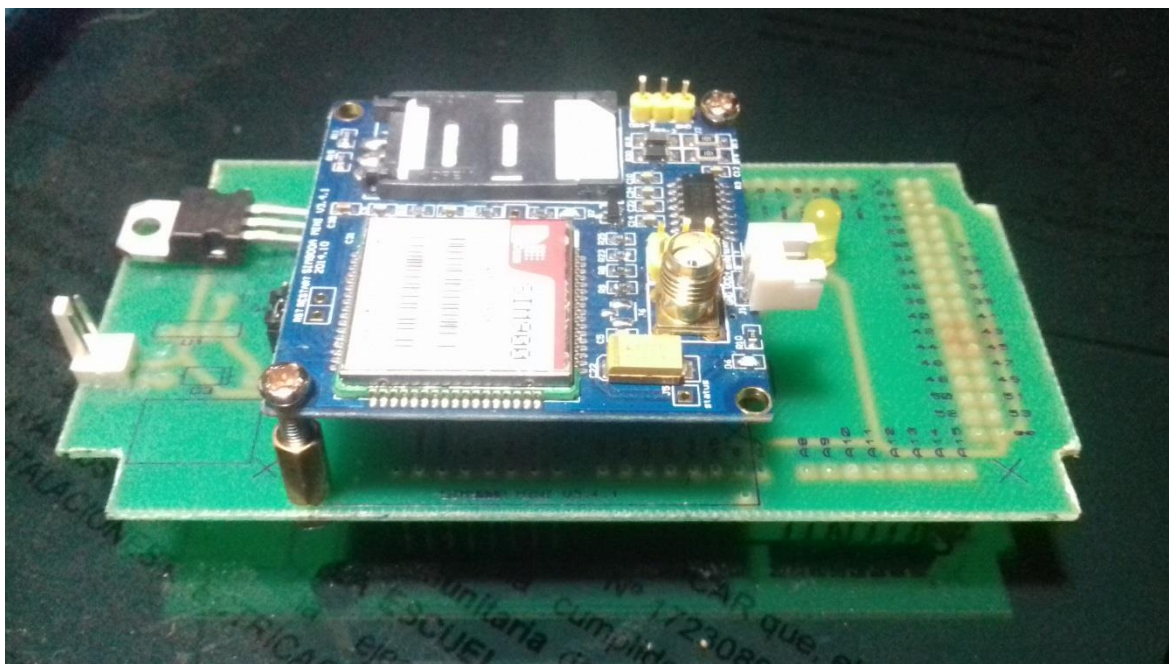
    }

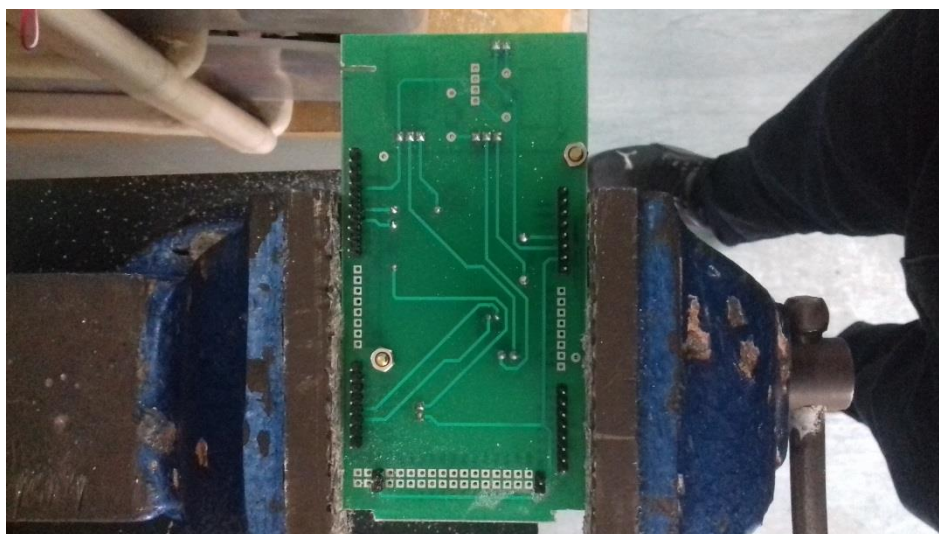
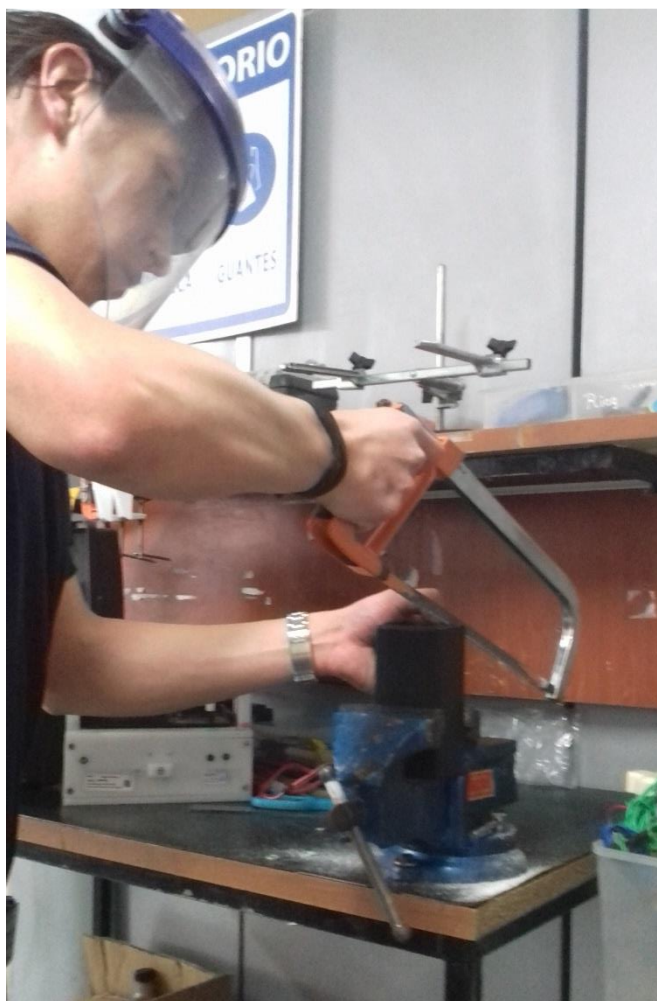
}
```

### ANEXO 7 Desarrollo del proyecto



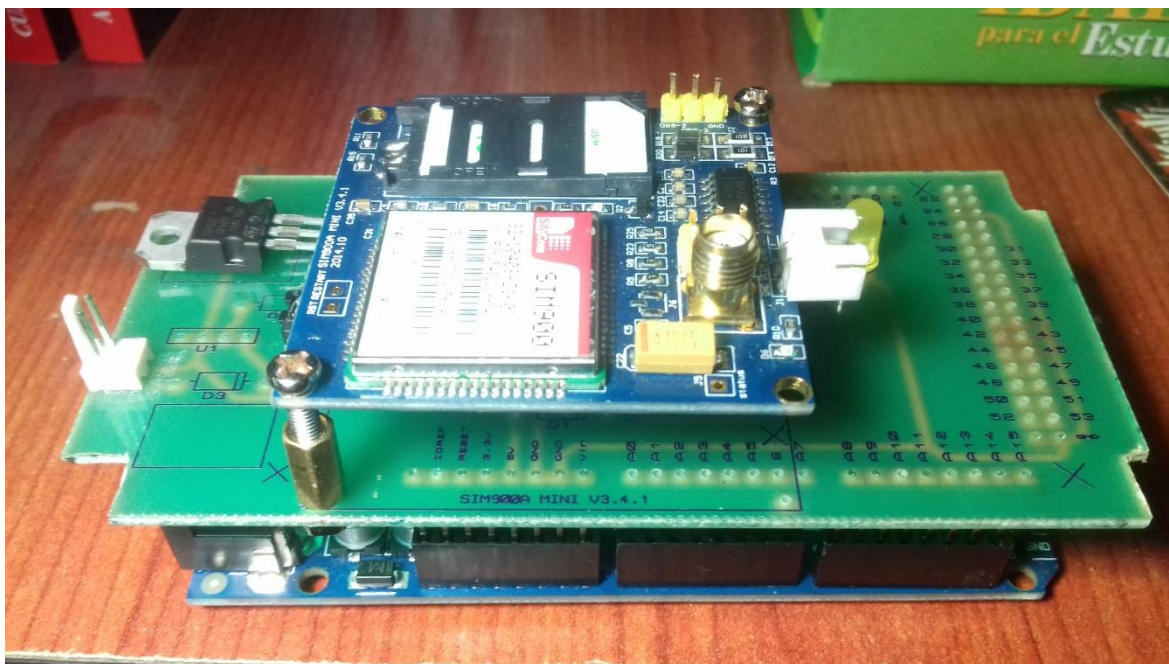
#### ANEXO 7.1



**ANEXO 7.2****ANEXO 7.4**



### ANEXO 7.3



### ANEXO 7.5



### ANEXO 7.6



### ANEXO 7.7



## ANEXO 8 Hoja de datos Tip 122



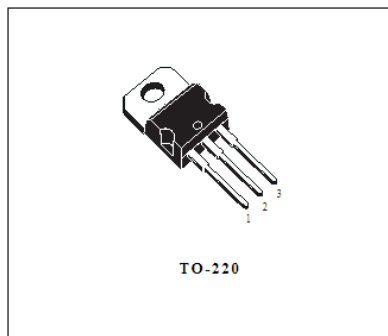
**TIP120/121/122**  
**TIP125/126/127**

COMPLEMENTARY SILICON POWER  
DARLINGTON TRANSISTORS

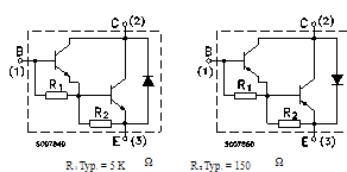
· STMicroelectronics PREFERRED  
SALESTYPES

**DESCRIPTION**

The TIP120, TIP121 and TIP122 are silicon Epitaxial-Base NPN power transistors in monolithic Darlington configuration mounted in Jedec TO-220 plastic package. They are intended for use in power linear and switching applications. The complementary PNP types are TIP125, TIP126 and TIP127, respectively.



**INTERNAL SCHEMATIC DIAGRAM**



**TIP120/TIP121/TIP122/TIP125/TIP126/TIP127**

**THERMAL DATA**

R <sub>th(j-case)</sub>	Thermal Resistance Junction-case	Max	1.92	°C/W
R <sub>th(j-amb)</sub>	Thermal Resistance Junction-ambient	Max	62.5	°C/W

**ELECTRICAL CHARACTERISTICS** (T<sub>case</sub> = 25 °C unless otherwise specified)

Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit
I <sub>CEO</sub>	Collector Cut-off Current (I <sub>b</sub> = 0)	for TIP120/125 V <sub>CE</sub> = 30 V for TIP121/126 V <sub>CE</sub> = 40 V for TIP122/127 V <sub>CE</sub> = 50 V			0.5 0.5 0.5	mA mA mA
I <sub>CBO</sub>	Collector Cut-off Current (I <sub>b</sub> = 0)	for TIP120/125 V <sub>CE</sub> = 60 V for TIP121/126 V <sub>CE</sub> = 80 V for TIP122/127 V <sub>CE</sub> = 100 V			0.2 0.2 0.2	mA mA mA
I <sub>EB0</sub>	Emitter Cut-off Current (I <sub>c</sub> = 0)	V <sub>EB</sub> = 5 V			2	mA
V <sub>CE(sat)*</sub>	Collector-Emitter Sustaining Voltage (I <sub>b</sub> = 0)	I <sub>c</sub> = 30 mA for TIP120/125 for TIP121/126 for TIP122/127	60 80 100			V V V
V <sub>CE(sat)*</sub>	Collector-Emitter Saturation Voltage	I <sub>c</sub> = 3 A I <sub>b</sub> = 12 mA I <sub>c</sub> = 5 A I <sub>b</sub> = 20 mA			2 4	V V
V <sub>BE(on)*</sub>	Base-Emitter Voltage	I <sub>c</sub> = 3 A V <sub>CE</sub> = 3 V			2.5	V
h <sub>FE*</sub>	DC Current Gain	I <sub>c</sub> = 0.5 A V <sub>CE</sub> = 3 V I <sub>c</sub> = 3 A V <sub>CE</sub> = 3 V	1000 1000			

\* Pulsed: Pulse duration = 300 μs, duty cycle < 2%

For PNP types voltage and current values are negative.

	<b>MANUAL TÉCNICO</b>	Fecha: 23/05/2011 Revisión: 02
M-3SA-03-XX	Pruebas de Transmisión y Recepción de datos Andon	Página: 1/8
Elaborado por: Luis Tito	Revisado por:	Aprobado por:

## ANEXO 9 Manual de usuario

### PUESTA A PUNTO

En la Figura 1 se pueden ver los elementos que conforman este proyecto:

Se debe seguir los siguientes pasos:

2. Conexión de los dispositivos
  - a. Conectar el adaptador de 9 VDC 1A a la entrada del dispositivo y posteriormente a la toma de 110 VCA para proveer de energía al dispositivo.
  - b. Verificar el LED de Modem Status hasta que el periodo de parpadeo sea de 1 segundo de encendido y tres segundos de apagado. Esto se realiza para verificar el nivel de la recepción de la operadora con la que está trabajando el dispositivo. Este periodo varía entre 10 segundos hasta 1 minuto.
  - c. Conectar el cable USB tipo B al dispositivo en el lugar indicado y el conector USB 2.0 a un puerto disponible del computador donde se encuentra instalado el programa Andon System.
  - d. Verificar el puerto con el que identifica el *tester* en el Administrador de dispositivos.

	<b>MANUAL TÉCNICO</b>	Fecha: 23/05/2011 Revisión: 02
M-3SA-03-XX	Pruebas de Transmisión y Recepción de datos Andon	Página: 1/8
Elaborado por: Luis Tito	Revisado por:	Aprobado por:



Figura 1. Elementos del proyecto

## CONEXIÓN A LA BASE DE DATOS

- a. Ejecutar el aplicativo Test Receptor que se encuentra en el escritorio para verificar la conexión del dispositivo con la base de datos que se encuentra en el área de Sistemas, en dicha base se almacenan los datos que transmiten los eventos que se generan en cada estación.
  
- b. Una vez iniciado el aplicativo se debe cambiar el puerto con el que está trabajando el dispositivo por ejemplo: COM7 como se muestra en la Figura 2.

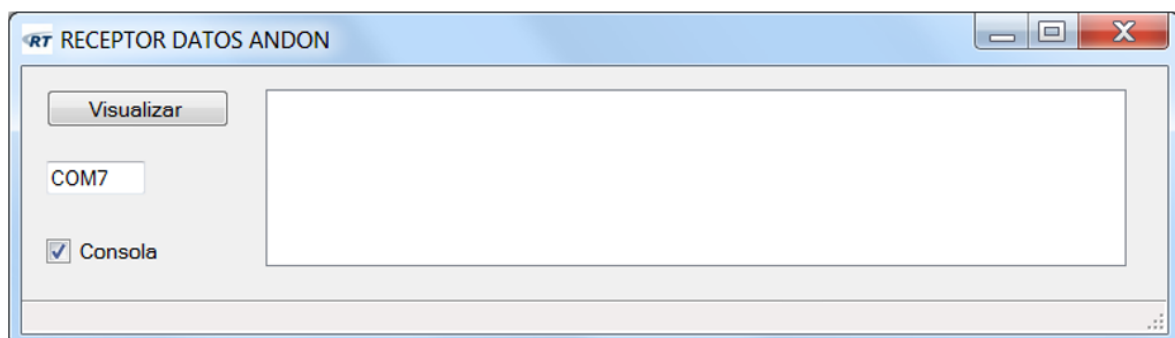


Figura 2. Software de verificación de conexión a la base de datos

	<b>MANUAL TÉCNICO</b>	Fecha: 23/05/2011 Revisión: 02
M-3SA-03-XX	Pruebas de Transmisión y Recepción de datos Andon	Página: 1/8
Elaborado por: Luis Tito	Revisado por:	Aprobado por:

- c. Adicional se debe poner check en el recuadro de consola para verificar la ejecución de los comandos AT y de esa manera comprobar el funcionamiento del modem. La información que aparece en la ventana de consola muestra como se establece la comunicación desde el modem hacia la base de datos.
  
- d. Dar clic en la pestaña Visualizar para obtener el último registro que fue ingresado en la base de datos. En la parte inferior de la ventana aparece el mensaje Procesando Request, lo que nos indica que la solicitud a la base de datos se está ejecutando. Esto se puede observar en la ventana de consola.
  
- e. Esperar 20 segundos aproximadamente hasta que en la parte inferior de la ventana se despliegue el mensaje Proceso OK, y se despliegue la información del último evento registrado en la base.
  
- f. Durante el tiempo en el que la información se envía se debe verificar el comportamiento del LED EVENT RECEIVED STATUS. Este LED realiza 2 parpadeos por cada respuesta del modem en cada tramo del establecimiento de la comunicación mediante comandos AT. (8 tramos)
  
- g. En caso de que se no se pueda establecer la conexión con la base de datos el modem deja de enviar respuestas y el LED deja de parpadear, si se presenta este inconveniente se debe desconectar el dispositivo del puerto de comunicación y de alimentación de voltaje posteriormente volver al paso 1.

	<b>MANUAL TÉCNICO</b>	Fecha: 23/05/2011 Revisión: 02
M-3SA-03-XX	Pruebas de Transmisión y Recepción de datos Andon	Página: 1/8
Elaborado por: Luis Tito	Revisado por:	Aprobado por:

## FUNCIONAMIENTO DEL TRANSMISOR

1. El dispositivo se encuentra ubicado en la estación que maneja el Andon System como se puede observar en la Figura 3, este no afecta a las labores habituales y es un lugar preciso para verificar la conexión hacia la base de datos.

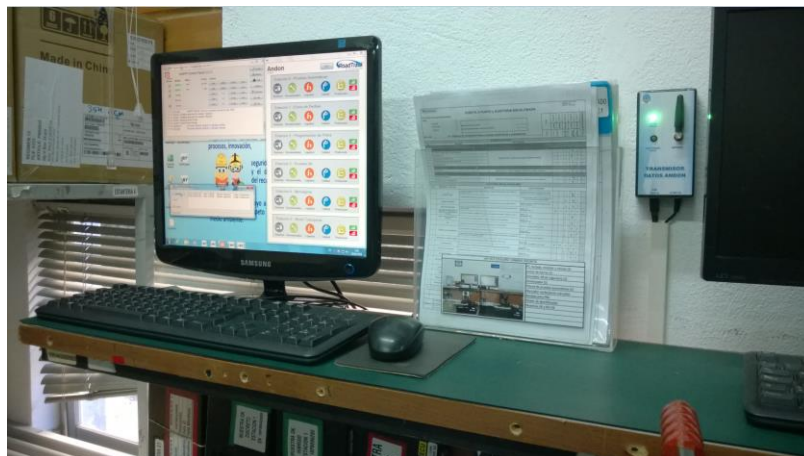


Figura 3. Ubicación del transmisor de datos Andon



Figura 4. Transmisor de datos Andon

	<b>MANUAL TÉCNICO</b>	Fecha: 23/05/2011 Revisión: 02
M-3SA-03-XX	Pruebas de Transmisión y Recepción de datos Andon	Página: 1/8
Elaborado por: Luis Tito	Revisado por:	Aprobado por:

2. Iniciar la aplicación Andon\_supervisor\_winform que se encuentra en el escritorio similar a la versión que maneja el Andon System. En la Figura 5 se muestra la ventana de inicio del programa.



Figura 5. Condiciones iniciales de funcionamiento

3. Verificar el status de cada uno de los slaves, esto quiere decir que se encuentren funcionales y a punto en caso de producirse un inconveniente en la estación de trabajo de la línea de Producción
4. En el momento de producirse un inconveniente relacionado con el área de Herramientales el operador debe accionar el slave indicando que en la estación se está produciendo una anomalía. Posteriormente el Supervisor se acerca al sitio a constatar el inconveniente.



	<b>MANUAL TÉCNICO</b>	Fecha: 23/05/2011 Revisión: 02
M-3SA-03-XX	Pruebas de Transmisión y Recepción de datos Andon	Página: 1/8
Elaborado por: Luis Tito	Revisado por:	Aprobado por:

- Mientras tanto en el computador donde se está ejecutando el programa se genera una ventana similar a la que se presenta en la Figura 6, donde se recoge la información acerca del incidente producido en la estación.

Figura 6. Datos del incidente producido en la estación

- Después de constatar el inconveniente el Supervisor debe escribir el nombre de la estación en la casilla de código. Esto es muy importante ya que es un dato que se muestra en el aplicativo de recepción de datos ubicado en el área de Herramientales.
- Una vez completados los datos se debe dar clic en la pestaña de Enviar evento para subir la información a la base de datos.
- En la parte inferior de la ventana se despliega el mensaje **Subiendo información...**, indicando el estado del envío de los datos del incidente, tal como se muestra en la Figura 7.

	<b>MANUAL TÉCNICO</b>	Fecha: 23/05/2011 Revisión: 02
M-3SA-03-XX	Pruebas de Transmisión y Recepción de datos Andon	Página: 1/8
Elaborado por: Luis Tito	Revisado por:	Aprobado por:

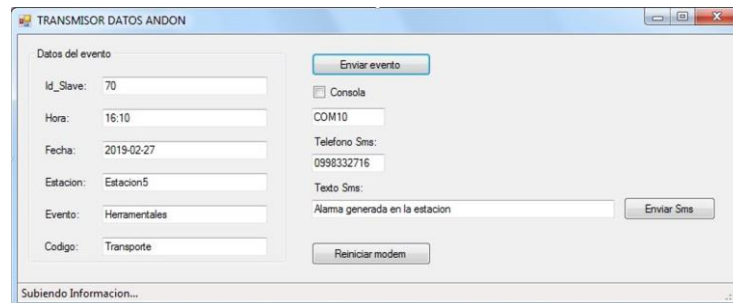


Figura 7. Estado del envío de la información

9. Mientras se envía la información se visualiza el estado del envío a través del LED EVENT SENDING STATUS, como se menciona en el literal (f) del paso 2.
10. Una vez que la información es almacenada en la base de datos se muestra el mensaje: Evento Ingresado Correctamente, como se puede apreciar en la Figura 8, constatando la conexión hacia la base de datos ubicada en el área de Sistemas.

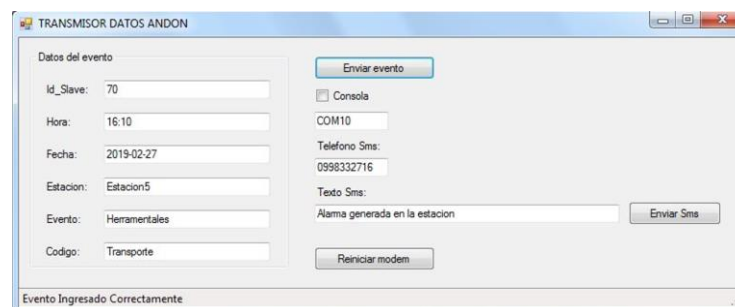


Figura 8. Evento ingresado correctamente

11. Posteriormente el Supervisor tiene la posibilidad de enviar un mensaje de texto a la persona encargada del soporte al área de Producción con una idea rápida acerca del inconveniente producido en la línea. Este mensaje puede incluir hasta 160 caracteres y son enviados a un número telefónico que puede ser ingresado manualmente. Para ello debe llenar el texto en la casilla correspondiente y una vez terminada la redacción dar clic en la pestaña Enviar mensaje.

	<b>MANUAL TÉCNICO</b>	Fecha: 23/05/2011 Revisión: 02
M-3SA-03-XX	Pruebas de Transmisión y Recepción de datos Andon	Página: 1/8
Elaborado por: Luis Tito	Revisado por:	Aprobado por:

12. En la Figura 9 se muestra la respuesta del programa una vez que el mensaje ha sido enviado al destinatario.



Figura 9. Estado del mensaje de texto

13. Una vez que se envía el mensaje el supervisor debe cerrar la ventana que contiene la información acerca del evento y posteriormente debe desactivar la alarma normalmente, es decir, ingresando el usuario y la contraseña. Como se muestra en la Figura 10.

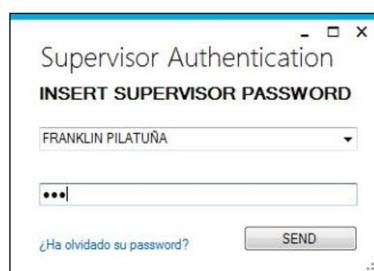


Figura 10. Desactivación de alarmas

## FUNCIONAMIENTO DEL RECEPTOR

1. Para esta parte se cuenta con un dispositivo de similares características al transmisor y para iniciar la operación se debe realizar los pasos de puesta a punto mencionados anteriormente en los pasos 1 y 2 como se muestra en la Figura 11.

	<b>MANUAL TÉCNICO</b>	Fecha: 23/05/2011 Revisión: 02
M-3SA-03-XX	Pruebas de Transmisión y Recepción de datos Andon	Página: 1/8
Elaborado por: Luis Tito	Revisado por:	Aprobado por:



Figura 11. Ubicación dispositivo Receptor de datos Andon



Figura 12. Receptor de datos Andon

2. En el área de Herramientales se cuenta con el aplicativo que pertenece al Andon System que se muestra en la Figura 13, esta ventana permite visualizar el momento en que se produce un incidente en la línea de Producción.

	<b>MANUAL TÉCNICO</b>	Fecha: 23/05/2011 Revisión: 02
M-3SA-03-XX	Pruebas de Transmisión y Recepción de datos Andon	Página: 1/8
Elaborado por: Luis Tito	Revisado por:	Aprobado por:

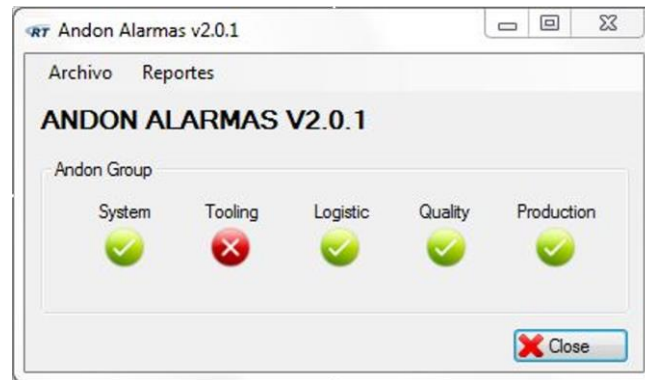


Figura 13. Visor de alarmas Andon System

3. El aplicativo que permite visualizar los datos en el receptor se denomina Receptor Datos Andon y se encuentra en el escritorio como acceso directo. En la Figura 14 se puede apreciar el aplicativo que al igual que en el transmisor se debe elegir un puerto de comunicación antes de entrar en funcionamiento.

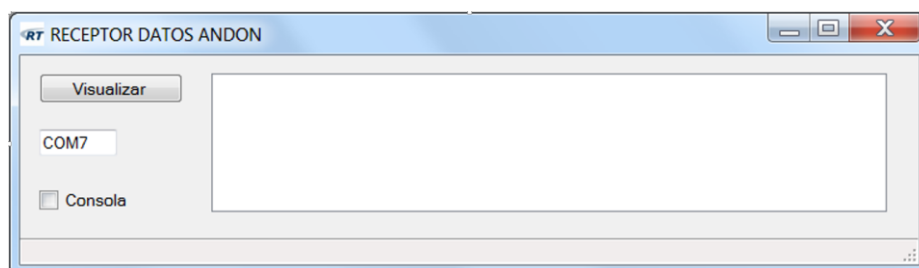


Figura 14. Receptor de datos Andon

4. Una vez que se visualiza la alerta en el área de Herramientales se debe accionar la pestaña Visualizar para que el dispositivo receptor extraiga la información de la base con los datos del evento que se produjo en el área de Producción.

	<b>MANUAL TÉCNICO</b>	Fecha: 23/05/2011 Revisión: 02
M-3SA-03-XX	Pruebas de Transmisión y Recepción de datos Andon	Página: 1/8
Elaborado por: Luis Tito	Revisado por:	Aprobado por:

5. En la Figura 15 se observa el mensaje Procesando Request... en la parte inferior del aplicativo y al igual que en el caso del transmisor, se debe esperar aproximadamente 20 segundos para obtener la visualización de los datos del evento.

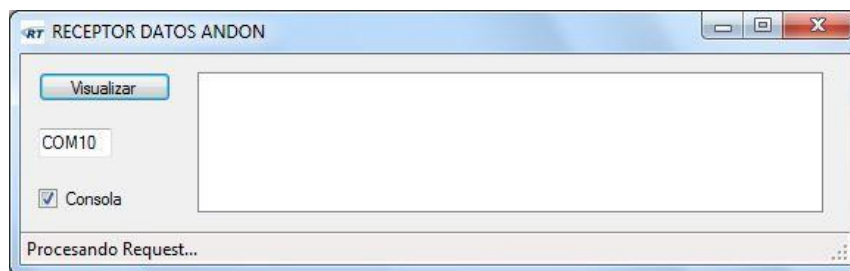


Figura 15. Estado de recepción del evento

6. De igual manera el LED EVENT RECEIVED STATUS indica el estado del proceso de comunicación del dispositivo receptor con la base de datos. Finalmente el aplicativo muestra el resultado con los datos que se obtienen de los slaves: Fecha, Hora, Estación y Código como se puede apreciar en la Figura 16.

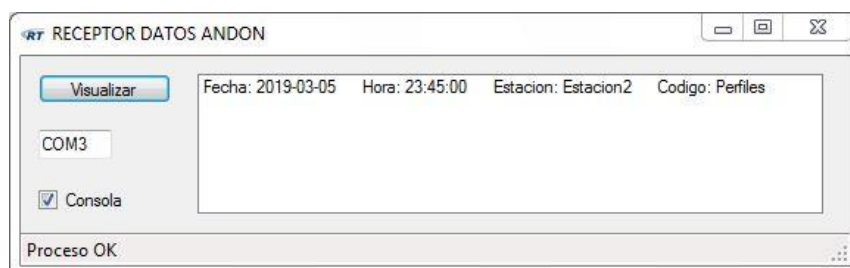


Figura 16. Información extraída de la base de datos