

**FACULTAD DE INFORMÁTICA Y CIENCIAS APLICADAS
TÉCNICO EN INGENIERÍA DE HARDWARE**



TEMA:

**SISTEMA CONFIGURABLE VÍA APP ANDROID PARA EL DESPLIEGUE
DE MENSAJES EN EL LABORATORIO DE HARDWARE DE LA
UNIVERSIDAD TECNOLÓGICA DE EL SALVADOR (UTEC).**

TRABAJO DE GRADUACIÓN PRESENTADO POR:

**CÉSAR ARNOLDO HERNÁNDEZ Crespín
RENÉ MAURICIO VILLALTA ESCOBAR**

PARA OBTAR AL GRADO DE:

TÉCNICO EN INGENIERÍA DE HARDWARE

ABRIL, 2015

SAN SALVADOR, EL SALVADOR CENTROAMÉRICA

PAGINA DE AUTORIDADES

ING. NELSON ZÁRATE SÁNCHEZ
RECTOR

LIC. JOSÉ MODESTO VENTURA
VICERRECTOR ACADEMICO

ING. FRANCISCO ARMANDO ZEPEDA
DECANO

JURADO EXAMINADOR

LIC. CARLOS ANTONIO AGUIRRE AYALA
PRESIDENTE

ING. JOSÉ MAURICIO RIVERA
PRIMER VOCAL

ING. JULIO CÉSAR ROSALES
SEGUNDO VOCAL

ABRIL, 2015

SAN SALVADOR, EL SALVADOR, CENTROAMERICA



ACTA DE EXAMEN PROFESIONAL

HABIÉNDOSE REUNIDO EL JURADO CALIFICADOR INTEGRADO POR:

Ing. José Mauricio Rivera, Ing. Julio César Rosales, Lic. Carlos Antonio Aguirre Ayala, a las 12:30m. del día Jueves, 4 de Diciembre de dos mil catorce,

Y LUEGO DE HABER DELIBERADO SOBRE EL EXAMEN PROFESIONAL DE LOS ALUMNOS:

- | | |
|------------------------------------|---------------------|
| 1- César Arnoldo Hernández Crespin | Carnet 28-1601-2012 |
| 2- René Mauricio Villalta Escobar | Carnet 28-3951-2011 |

QUIENES PRESENTARON DEFENSA DE SU TRABAJO DE GRADUACION TITULADO:
"Sistema configurable vía APP Andoid para el despliegue de mensajes en el laboratorio de Hardware de la Universidad Tecnología de El Salvador (UTEC)"

PARA OPTAR AL GRADO DE:

TÉCNICO EN INGENIERIA DE HARDWARE

Y DEL CUAL TAMBIEN EVALUARON LOS CONOCIMIENTOS RELACIONADOS CON EL TEMA DEL MISMO. POR LO QUE ESTE JURADO RESUELVE DECLARAR EL EXAMEN COMO:

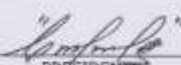
APROBADO

YA QUE CUMPLE CON LOS REQUISITOS ESTABLECIDOS EN EL REGLAMENTO DE GRADUACION DE LA UNIVERSIDAD.

San Salvador, 4 de Diciembre de 2014

F. 
PRIMER VOCAL
Ing. José Mauricio Rivera

F. 
SEGUNDO VOCAL
Ing. Julio César Rosales

F. 
PRESIDENTE
Lic. Carlos Antonio Aguirre Ayala

Agradecimientos

Primeramente gracias a Dios por ser nuestro Guía, a Jesús por él es el mejor ejemplo de amor en este mundo y nuestro padres por ser el mejor ejemplo de vida a seguir.

A nuestra familia fuente de apoyo incondicional en toda mi vida y más aún en mis duros años de carrera profesional y especialmente quiero expresar mis agradecimiento a mis padres que sin su ayuda hubiera sido imposible culminar nuestra carrera universitaria y el apoyo tanto moral como económicamente para seguir estudiando y seguir el objetivo trazado para un futuro mejor y ser un orgullo para la familia.

A la Universidad Tecnológica de el salvador (UTEC) por la cálida y el compromiso educativo que ofrece a sus estudiantes, a los docentes que fueron partes de nuestra formación porque gracias a ellos se formó el desarrollo de esta información.

A nuestro asesor julio cesar rosales y asesor asociado Ing. Omar Otoniel Flores por sus correcciones y dedicaciones para seguir con nuestro objetivo y poder lograr el desarrollo e esta tesis.

A los jurados de tesis Lic. Carlos Aguirre e Ing. Mauricio rivera los cuales nos ayudan a ver nuestros errores y pulir nuestras ideas para que al final sea un trabajo bien desarrollado.

ÍNDICE

Introducción.....	i
CAPITULO I SITUACIÓN ACTUAL	
1.1 Situación Problemática.....	1
1.2 Enunciado del Problema.....	2
1.3 Justificación.....	2
1.4 Objetivos.....	3
1.4.1 Objetivo General.....	3
1.4.2 Objetivo Específico.....	4
1.5 Delimitaciones.....	4
1.5.1 Delimitación Geográfica.....	4
1.5.2 Delimitación Temporal.....	5
1.5.3 Delimitación Organizacional.....	5
1.6 Alcances (Entregables).....	7
1.7 Estudio de Factibilidad.....	8
1.7.1 Factibilidad Económica.....	9
1.7.2 Factibilidad Técnica.....	10
1.7.3 Análisis General de los Componentes.....	14
CAPÍTULO II DOCUMENTACIÓN TÉCNICA	
2.1 Marco Teórico De Referencia.....	15
2.1.1 Domótica E Inmótica.....	15
2.1.2 Área de Aplicación de la Domótica e Inmótica.....	16
2.1.3 Componentes de un Sistema Domótico.....	17
2.1.4 Características que debe cumplir un buen Sistema Inmótico.....	19
2.1.5 Beneficios de un Sistema Inmótico.....	20
2.1.6 Sistema Operativo Android.....	21
2.1.7 Versiones de Android.....	23
2.1.8 Plataforma de desarrollo Electrónico Arduino.....	28
2.1.9 Plataforma de Desarrollo APP Inventor.....	30
2.1.10 Categorías de la APP.....	30

2.2 Marco Teórico de Solución.	35
2.3 Marco Teórico Conceptual.	42
2.4 Documentación Técnica.	47
2.4.1 Características de los Componentes del Proyecto.	48
CAPITULO III DESARROLLO DE LA SOLUCIÓN	
3.1 Propuesta de Solución.....	53
3.1.1 Firmware de control general.....	55
3.1.2 Código Arduino	56
3.1.3 Flujograma de Funcionamiento del Sistema Configurable Vía APP Android para el Despliegue de Mensajes.....	70
3.1.4 Diagrama Electrónico del Sistema Configurable Vía APP Android para el Despliegue de Mensajes.....	71
3.1.5 APLICACIÓN MOVIL (APP)	72
3.1.6 Pantalla principal	72
3.1.7 Pantalla de ayuda.....	73
3.1.8 Pantalla de ingreso de contraseña.....	74
3.1.9 Pantalla de envió de mensajes	75
3.1.10 Implementaciones	76
3.1.11 Elementos del prototipo	77
3.1.12 Ensamble del prototipo	80
3.1.13 Fotos del prototipo funcionando correctamente.....	81
3.2 Conclusión	82
3.3 Recomendaciones.....	83
3.4 Bibliografía.....	84
3.5 Anexos	85

INTRODUCCIÓN

Actualmente en las Universidades consta de muchos laboratorios informáticos donde necesita de un sistema de despliegue para exteriores que cumpla con la labor de informar las actividades diarias de dicho laboratorio, por lo cual surgió la necesidad de tener un sistema de despliegue para exteriores que cumpla con dicha función de brindar mensajes. Debido a esta necesidad se ha optado realizar e implementar un “sistema Domótico con tecnología Android para el despliegue de mensajes a exteriores como apoyo a las labores informativas del laboratorio de hardware de la Universidad Tecnológica de El Salvador.” Para solventar esta problemática y poder brindarle al encargado una herramienta de comunicación para los exteriores.

En el presente documento el lector tendrá a su disposición información sobre el desarrollo, diseño y construcción del proyecto llamado: Sistema configurable vía APP Android para el despliegue de mensajes en el laboratorio de Hardware de la Universidad Tecnológica de El Salvador. El primer capítulo, contiene características detalladas acerca de la formulación del proyecto, se especifican los objetivos perseguidos, los beneficios a obtener con la implementación, así como los estudios de factibilidad tanto económica como técnica, de los componentes principales

del proyecto. De antemano se destacan las principales características: funcionalidad, facilidad de uso, precisión y exactitud la cual considera como principal atractivo ya que en ellas se encierran todos los beneficios y soluciones que se pretende dar a la Universidad. Este documento también incluye la carta de aceptación donde se estipula que la Universidad Tecnológica de El Salvador, permite que se implemente en el laboratorio de Hardware el presente proyecto de graduación, además se anexa a este la matriz de congruencia donde se muestran los objetivos, alcances y el producto final.

Para el diseño e implementación de un sistema configurable vía APP Android para el despliegue de mensajes en el laboratorio de hardware de la Universidad Tecnológica de El Salvador, el capítulo dos se divide en cuatro sub contenidos que son:

1. **Marco teórico de referencia:** donde se describe las áreas de aplicación en que se fundamenta el proyecto.
2. **Marco teórico de la solución:** es donde se presenta la teoría referente a la solución provista por el proyecto, dando a conocer los componentes del circuito, así como sus implementaciones en las instalaciones de la Universidad tecnológica.

3. **Marco teórico conceptual:** donde se presentan una serie de conceptos teóricos que se van a utilizar en el desarrollo del proyecto.
4. **Documentación técnica:** contiene las características técnicas generales de los componentes eléctricos de todo el proyecto desarrollado.

En el capítulo tres se muestra el desarrollo e implementación del sistema configurable vía APP Android para el despliegue de mensajes en el laboratorio de hardware como solución a la problemática planteada por la Universidad Tecnológica, así como de la elaboración del flujograma para mayor comprensión del usuario, ya que se muestra el comportamiento que tendrá en circuito, el desarrollo del firmware o programa de control del circuito que será cargado al arduino Uno a través de un programador y con el apoyo del software Arduino.ino.

Se presenta el diagrama del circuito como una guía en la fabricación y a su vez para el ensamble de este, así se podrá verificar la forma de conectar cada uno de los componentes que serán utilizados en la fabricación de este proyecto.

Se dan a conocer las recomendaciones y conclusiones que se tienen al finalizar este proyecto, además de los anexos.

CAPITULO I SITUACIÓN ACTUAL

1.1 Situación Problemática.

En la calle Arce (17Avenida Norte) en el departamento de San Salvador se encuentra la Universidad Tecnológica de El Salvador (UTEC), Universidad privada, la cual dispone de 9 edificios y entre ellos se encuentra el edificio Francisco Morazán, la cual dispone de 5 laboratorios informáticos, de las cuales 3 poseen puertas de vidrio y 2 son puertas tradicionales de madera. Entre estas se encuentra el laboratorio de hardware, laboratorio informático utilizado para impartir clases, seminarios, prácticas y diplomados. Por lo cual surge la necesidad de un sistema de mensajes para exteriores, En la cual cabe mencionar la falta de información visual al usuario que quiera tener ingreso al laboratorio, ya que no se puede saber si hay actividades o está disponible. Siendo esto algo muy importante que no se puede obviar, pero lastimosamente en la Universidad no dispone sistema que cumpla con estas necesidades, pues actualmente disponen de un pequeño deposito cartelero ubicado cerca de la puerta pero no cumple con todas la necesidades.

Por otro lado, los problemas que surgen a raíz de esta situación es que en ocasiones a la hora de ir a buscar el laboratorio asignado para sus clases, suele pasar que hay confusión y no es el correcto y por la cantidad de alumnos y distinta forma de llegada no logran brindarles la

información correcta al instante a todos y muchos no logran conocer la información actual.

Tampoco se puede obviar que el catedrático asignado para impartir las clases en el laboratorio tiene diversas actividades diarias, por la cual pueden presentarse situaciones en donde se vuelve complicado en el momento de atender a los alumno/as.

Es bueno dejar claro que si no se realiza algo para resolver esta situación, siempre será una problemática constante tanto para los encargados como también para los alumnos/as.

1.2 Enunciado del Problema.

¿Cómo brinda una herramienta al laboratorio de hardware, para el anuncio de sus actividades a los estudiantes y que sea de fácil configuración vía un dispositivo móvil Android?

1.3 Justificación

Es necesario que en la Universidad Tecnológica de El Salvador, posea un sistema Domótico, el cual facilite la labor de información en las actividades diarias del laboratorio de hardware. El proyecto ayudará al mantener informado al estudiante acerca de las actividades que se están realizando o a realizarse en el laboratorio, así mismo a su encargado o

catedráticos para que cumpla con sus labores diarias de manera más tranquila y eficiente.

Además de ser un proyecto innovador y de gran confort es accesible por su bajo costo. Así mismo cuenta con un sistema personalizado de control y funcionamiento.

Algunos beneficios que el laboratorio obtendría son:

- Mantener informado al estudiante
- Poder modificar los mensajes vía APP Android
- La APP será de desarrollo propio y uso privado del laboratorio (creada por estudiante y asesor).
- Contar con una tecnología de bajo costo y actual.

1.4 Objetivos.

1.4.1 Objetivo General.

Implementar un sistema de despliegue para exteriores de mensajes alfanuméricos vía APP Android como apoyo en las labores informáticas del laboratorio de Hardware de la Universidad Tecnológica de El Salvador.

1.4.2 Objetivo Específico

- Crear un sistema electrónico que sea capaz de visualizar las actividades diarias del laboratorio de Hardware de la Universidad Tecnológicas de El Salvador (UTEC).
- Desarrollar un firmware que permita la funcionabilidad del circuito adecuadamente a la misma vez pueda recibir mensajes enviados por medio de un teléfono móvil o tablet por medio de una APP de sistema Android.
- Desarrollar una aplicación o APP para sistema operativo Android que permita la configuración de mensajes del sistema desde el dispositivo móvil.
- Integrar en un solo circuito electrónico el Hardware y Software para la realización del proyecto final.

1.5 Delimitaciones

1.5.1 Delimitación Geográfica

A realizarse en el laboratorio de Hardware, de la Universidad Tecnológica de El Salvador del departamento de San Salvador, El Salvador. C.A.



Figura 1.1 Mapa de ubicación de la Universidad Tecnológica de El Salvador (UTEC)

Fuente: Google Maps.

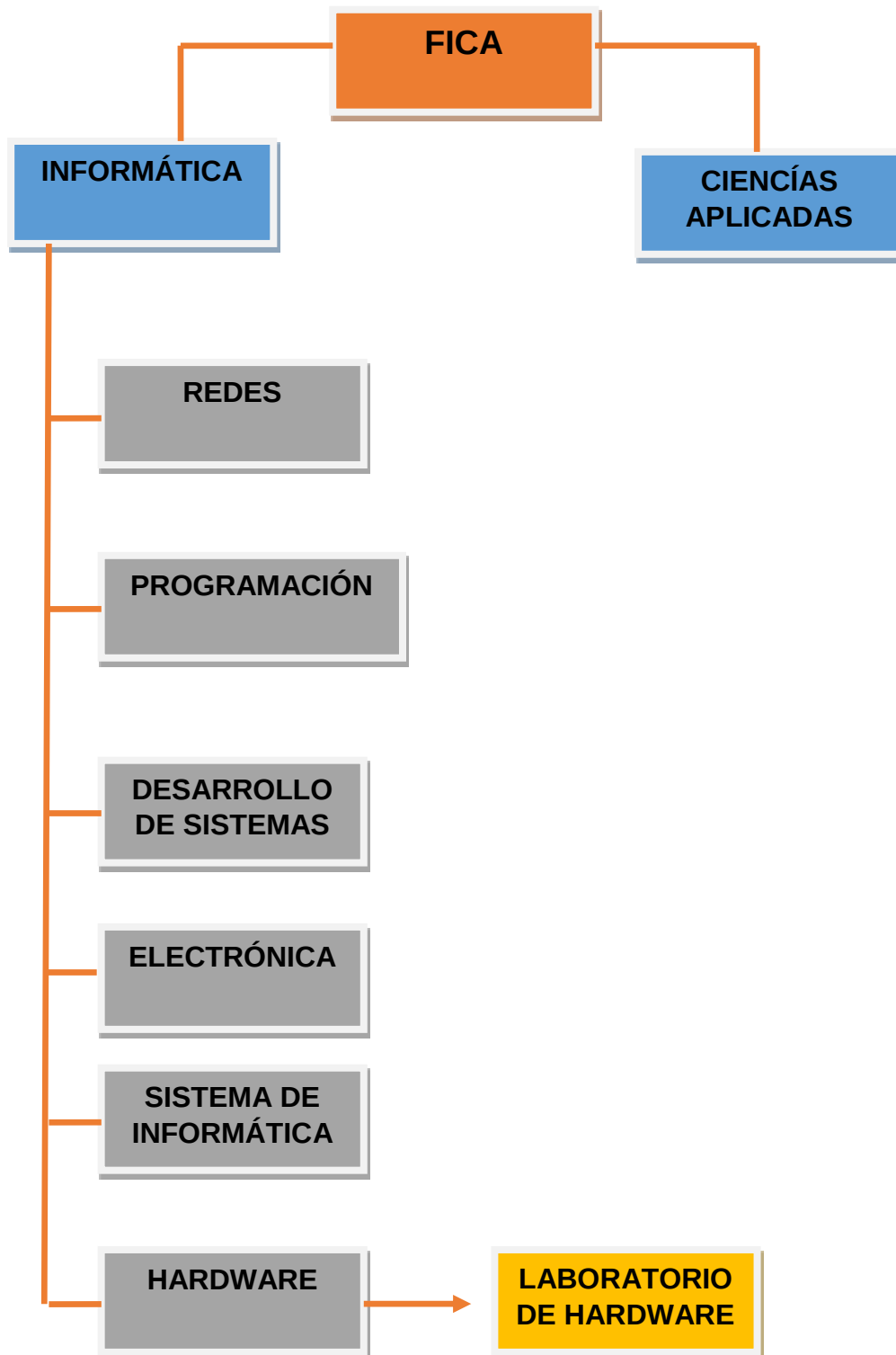
1.5.2 Delimitación Temporal.

Programado para el periodo del 20 de septiembre al 10 de octubre del 2014.

1.5.3 Delimitación Organizacional

En el siguiente Organigrama se muestra como está estructurado el laboratorio de Hardware de la Universidad Tecnológica de El Salvador ya que depende de la Facultad de Informática y Ciencias Aplicadas y sus ramas como la Escuela de Informática.

Figura 1.2 Esquema que muestra la organización de la Universidad Tecnológica de El Salvador en el área de informática.



1.6 Alcances (Entregables)

Durante la realización de este proyecto los alcances propuestos darán paso a convertirse en promesas en donde estas se encuentran en el siguiente cuadro juntamente con los productos para el desarrollo de este proyecto y su finalización.

PROMESAS	PRODUCTO
Crear un circuito electrónico de primera tecnología, implementadas tales como la plataforma arduino y la tecnología Android para así poder elaborar un sistema electrónico que sea capaz de visualizar las actividades diarias del laboratorio de hardware.	Sistema electrónico de despliegue (pantalla).
Desarrollar un firmware que permita la funcionalidad del circuito para poder cumplir con las labores de información visual para los exteriores de dicho laboratorio.	Software diseñado para el control de pantalla de despliegue.

Desarrollar una aplicación para sistema operativo Android, lo cual permite la configuración de mensajes.	Aplicación con sistema operativo Android de diseño propio.
Integrar el hardware y software para la realización de un solo circuito electrónico juntamente con el sistema Android.	Proyecto funcional y con una aplicación Android con la cual podrá enviar mensajes para la visualización de actividades en el laboratorio de Hardware.

1.7 Estudio de Factibilidad.

Es necesario el estudio técnico y a la vez económico para la elaboración de un dispositivo electrónico, para tener una cotización y saber que componentes son más recomendables tanto económico y funcional que estos dispositivos electrónicos presenten. Para ello se cotizara precios y marcas que conforman el proyecto final.

1.7.1 Factibilidad Económica.

Para el desarrollo del dispositivo electrónico se hacen todas las comparaciones de los precios de los diferentes componentes a utilizar con el objetivo de apreciar la factibilidad económica de cada componente electrónico.

Seguidamente se presentarán una serie de tablas, en las que se mostrarán los precios y lugar de cotización de los elementos electrónicos que se utilizarán en el desarrollo del proyecto.

MICROCONTROLADOR			
Características.	PICAXE28	PIC16F84A	ATMEGA328
Precio de cotización	\$12.00	\$4.75	\$3.39
Lugar de cotización	www.ebay.com	www.mouser.com	www.mouser.com

Elección de micro-controlador: para este proyecto, el precio accesible y tomado en cuenta del aprendizaje adquirido en la Universidad Tecnológica de El Salvador (UTEC) fue el motivo por el cual se decide por el micro-controlador ATMEGA328, la decisión se tomó ya que es el de menor costo entre las ofertas presentadas.

MÓDULO DE COMUNICACIÓN BLUETOOTH			
Características.	HC-06	HC-05	VS1838
Precio de cotización	\$7.05	\$5.90	\$8.00
Lugar de cotización	www.bangood.com	www.bangood.com	www.ingnac.com

Elección del módulo de comunicación Bluetooth: se decide por el módulo HC-06 ya que está entre medio del rango de precios de cotización.

PANTALLA DE VISUALIZACIÓN			
Características.	DISPLAY DE LED DE CUATRO DIGITOS	MATRIZ DE LED	LIQUID CRYSTAL DISPLAY (LCD)
Precio de cotización	\$3.39	\$6.78	\$6.80
Lugar de cotización	www.ebay.com	www.ebay.com	www.ebay.com

Elección de la pantalla de visualización: se decide por la matriz de led ya que su visualización es muy buena y el costo anda entre lo estimado.

1.7.2 Factibilidad Técnica.

Se hace el análisis y elección de los componentes desde un punto de vista técnico, tomando en cuenta características de cada uno, para elegir el que es mejor y que sea posible el correcto funcionamiento, tomando en cuenta su rendimiento y lo económico.

MICROCONTROLADOR							
SE REQUIERE UN PORSENTAJE DE ACEPTACIÓN MINIMO DE 90%							
CARACTERÍSTICAS	PONDERACIÓN	PICAXE28		PIC16F84A		ATMEGA328	
	%	VALOR	%	VALOR	%	VALOR	\$
ENCAPSULADO	10%	1	10%	1	10%	1	10%
VELOCIDAD	20%	1	20%	2	40%	2	40%
FACILIDAD DE PROGRAMACIÓN	15%	1	15%	2	30%	2	30%
MEMORIA	20%	2	40%	1	20%	2	40%
LENGUAJE DE PROGRAMACIÓN	20%	1	20%	2	40%	2	40%
VIA DE PROGRAMACIÓN	15%	2	30%	2	30%	2	30%
TOTAL	100%		135%		170%		190%
PORCENTAJE FINAL	TOTAL/2		67.5%		85%		95%

VALORES: MALO = 0, MUY BUENO = 1, EXCELENTE = 2

Elección del micro-controlador: Atmega328 es debido a que cumple con todas las características técnicas necesarias para el excelente funcionamiento de nuestro proyecto.

MÓDULO DE COMUNICACIÓN BLUETOOTH							
SE REQUIERE UN PORSENTAJE DE ACEPTACIÓN MINIMO DE 90%							

CARACTERÍSTICAS	PONDERACIÓN	HC-06		HC-05		VS1838	
		VALOR	%	VALOR	%	VALOR	\$
DISTANCIA DE BLUETOOTH	30%	2	60%	2	60%	2	60%
TAMAÑO	20%	1	20%	1	20%	1	20%
FRECUENCIA	25%	2	50%	1	25%	1	25%
FACILIDAD DE COMUNICACIÓN	25%	2	50%	2	50%	1	25%
TOTAL	100%		180%		155%		130%
PORCENTAJE FINAL	TOTAL/2		90%		77.5%		65%

VALORES: MALO = 0, MUY BUENO = 1, EXCELENTE = 2

Elección del módulo de comunicación: según los valores mostrados en la tabla es más factible utilizar el módulo HC-06, pues cumple con todas las características necesarias para la implantación del proyecto.

PANTALLA DE VISUALIZACIÓN							
SE REQUIERE UN PORSENTAJE DE ACEPTACIÓN MINIMO DE 90%							
CARACTERÍSTICAS	PONDERACIÓN	DISPLAY DE LED DE CUATRO DIGITOS		MATRIZ DE LED		LIQUID CRYSTAL DISPLAY (LCD)	
		VALOR	%	VALOR	%	VALOR	%
N° DE PINES	20%	1	20%	2	40%	2	40%
COLOR	30%	1	30%	2	60%	1	30%
TAMAÑO	20%	2	40%	2	40%	2	40%
CONFIGURACION	30%	1	30%	2	60%	1	30%
TOTAL	100%		120%		200%		140%
PORCENTAJE FINAL	TOTAL/2		60%		100%		70%

VALORES: MALO = 0, MUY BUENO = 1, EXCELENTE = 2

Elección de la pantalla de visualización: utilizaremos la matriz de led's debido a que cumple con los requerimientos técnicos necesarios para la implementación del sistema.

1.7.3 Análisis General de los Componentes.

ANÁLISIS GENERAL DE LOS COMPONENTES QUE SE UTILIZARÁN EN EL PROYECTO	
PRODUCTO	DESCRIPCIÓN DEL PRODUCTO
MICROCONTROLADOR ATMEGA 328	Se utilizará este micro-controlador por su precio accesible y es más fácil de encontrar donde este posee un alto mercado en el país y además poseemos conocimiento de programación adquirido en nuestra carrera.
MÓDULO DE BLUETOOTH HC-06	Este módulo se utilizará por su bajo costo que posee y por su nivel de conexión inalámbrica que nos brinda y nos facilitará la comunicación con el dispositivo Android a utilizar.
MATRIZ DE LED	Se utilizará en nuestro proyecto por su factibilidad económica y su mayor confort en la visualización para mensajes.

CAPÍTULO II DOCUMENTACIÓN TÉCNICA

2.1 Marco Teórico De Referencia.

Este apartado presenta información importante de cada área en las que se fundamentará el proyecto denominado “Sistema configurable vía APP Android para el despliegue de mensajes en el laboratorio de Hardware de la Universidad Tecnológica de El Salvador.

2.1.1 Domótica E Inmótica.

Según (Cristóbal, Fracisco, & Carlos, 2011) el término Domótica es ampliamente utilizado en la actualidad, aunque a veces de forma incorrecta, ya que se usa casi siempre para indicar cualquier tipo de automatización. Donde la palabra “domo” etimológicamente proviene del latín domus que significa casa y el sufijo tica proviene de la palabra automática. Aunque algunos autores también diferencian entre “TIC” de Tecnologías de la Información y de la Comunicación y “a” de Automatización.

Así mismo se debe de tener en cuenta que existen múltiples definiciones de Domótica elaboradas por distintos expertos del tema pero en la gran mayoría se destaca la idea de mejorar la calidad de vida de los usuarios de estos sistemas.

Ya que según (Millares., 2005) de forma más rigurosa se puede definir la domótica como ciencia y los elementos y servicios desarrollados por ella que proporcionan algún nivel de automatización o automatismo de forma integrada dentro de la casa y capaz de satisfacer las necesidades básicas de seguridad, comunicación, gestión energética y confort, del hombre y de su entorno más cercano. Etimológicamente, la palabra domótica fue acuñada en Francia y procede de la unión de domus (casa en latín) y robotique (robótica).

Por Inmótica se entiende la incorporación de sistemas que proporcionan algún nivel de automatización dentro del equipamiento de las edificaciones del sector terciario, como son hospitales, edificios de oficinas, grandes superficies, parques tecnológicos, etc. De forma óptima e integrada proporciona a los distintos controles y automatismos que se incluyen en el edificio, comunicación, control, monitorización, gestión y mantenimiento de los mismos.

2.1.2 Área de Aplicación de la Domótica e Inmótica.

Según (Romero, Vazquez, & Castro, Domótica e inmótica, 2011) estos sistemas se están haciendo parte de nuestras vidas cotidianas y sus aplicaciones son diversas, algunas se detallan a continuación:

Confort: Se encarga de facilitar al usuario la obtención de un mayor nivel de comodidad en las actividades que desarrolle dentro de la vivienda o edificio.

Comunicación: Esta se encarga de captar, transportar, almacenar, procesar y difundir datos o información.

Ahorro energético: El ahorro energético no es algo tangible, sino un concepto al que se puede llegar de muchas maneras. Se encarga de gestionar el consumo de energías mediante temporizadores y otros.

Seguridad: Consiste en una red de seguridad encargada de proteger tanto los Bienes Patrimoniales, la seguridad personal e incidentes y averías. Por ejemplo: Alarmas.

Accesibilidad: en este se incluyen las aplicaciones o instalaciones de control remoto del entorno que favorecen la autonomía de personas con limitaciones funcionales, o discapacidad.

2.1.3 Componentes de un Sistema Domótico.

Si bien los componentes de un sistema Domótico pueden variar en cada caso, según (Romero, Vazquez, & de Castro, 2011) se pueden diferenciar algunos generales:

Sensor: La Misión de un sensor es la conversión de magnitudes de una determinada naturaleza a otra y también es el dispositivo que monitoriza el entorno captando información que transmite al sistema

Controlador: En este gestiona toda la instalación, recibe señales que proporciona los sensores y emite las señales que llegarán a los actuadores.

Actuador: Dispositivo electromecánicos que actúan sobre el medio exterior y son capaz de recibir una orden y ejecutarla.

Bus: Es el medio de comunicación que transporta la información entre los distintos dispositivos, ya sea por una red propia, o por la redes de otros sistemas.

Interfaz: Señal que entrega un controlador, ya sea analógico o digital también es un dispositivos en que se muestra la información del sistema para los usuarios y donde ellos mismos pueden interactuar con el sistema.



Imagen 2.1 Componentes de sistema Domótico.

Fuente: Diseño Propio

2.1.4 Características que debe cumplir un buen Sistema Inmótico.

Integración: Es la propiedad fundamental de un edificio inteligente. Es lo que diferencia un edificio inteligente de un edificio automatizado. En una instalación automatizada, los diversos autómatas actúan de forma aislada. Al integrar el conjunto de sensores, controles, actuadores. El edificio es capaz de detectar lo que ocurre en su interior y en su alrededor y actuar en consecuencia.

Flexibilidad: El sistema debe ser capaz de adaptarse con facilidad a la incorporación de nuevos subsistemas en su arquitectura. Resulta fundamental que tras una inversión inicial que puede resultar importante, se pueda actualizar de forma rápida y cómoda el sistema con tecnologías futuras.

Fiabilidad: El número de funciones que controla el sistema será elevado por lo que es necesario reducir los errores al mínimo para que las consecuencias ocasionadas sean irrelevantes.

Manejo sencillo: El sistema será controlado por más de un empleado y generalmente, será personal no cualificado. Por ello, es necesario que el funcionamiento que permite controlar el sistema sea de fácil uso y rápida comprensión a la hora de aprender a usarlo.

2.1.5 Beneficios de un Sistema Inmótico.

Reduce el consumo de energía: El edificio inteligente controla de forma óptima el uso de la energía, provocando un ahorro económico considerable. Además, contribuye a proteger el medio ambiente.

Aumenta el confort: Un edificio inteligente proporciona a los ocupantes del mismo un ambiente más confortable, lo que provoca mejores condiciones de trabajo y favorece la producción de los empleados.

Aumenta la seguridad: Una de las áreas a la que más importancia da un sistema inmótico es la seguridad, ya que generalmente el edificio contará con un equipamiento caro y con información que deberán ser protegidos ante intrusiones y alarmas técnicas (inundaciones, incendios, etc.).

Gestión remota: Disponiendo de un acceso a Internet, desde cualquier rincón del mundo se puede controlar y variar cualquier parámetro del sistema.

Buena impresión: La introducción de tecnología en edificios de oficinas provoca buena imagen ante los clientes.

2.1.6 Sistema Operativo Android.

Según (Girones, 2013) Android es un sistema operativo basado en Linux, diseñados principalmente para dispositivos móviles con pantalla táctil, como los teléfonos inteligente o Tabletas, google, tv y otros dispositivos, junto con aplicaciones middleware (software que asiste a una APP para interactuar o comunicarse con otras aplicaciones, software, hardware, redes, y/o sistemas Operativos).

Este sistema por lo general maneja aplicaciones como Market o su actualización, Play Store. Es desarrollado por la open Handset Alliance, la cual es liderada por google.

Fue desarrollado principalmente por Android Inc. Una firma comprada por Google en el 2005. Es principal producto de Open Handset Alliance, un conglomerado de fabricantes y desarrolladores de Hardware, software y operadores de servicios.

(Yerika, 2012) Android se ubica como uno de los principales en Estado unidos por su venta de teléfonos inteligente con sistemas Android dentro del segundo y tercer trimestre de 2010.

A nivel mundial alcanzó una cuota de mercado del 50,9% durante el cuarto trimestre de 2011, más del doble que el segundo sistema operativo (IOS de iPhone) con más cuota.

Tiene una gran comunidad de desarrolladores escribiendo aplicaciones para extender la funcionalidad de los dispositivos. A la fecha, se han sobrepasado las 400.000 aplicaciones (de las cuales, dos tercios son gratuitas) disponibles para la tienda de aplicaciones oficial de Android: Google Play, sin tener en cuenta aplicaciones de otras tiendas no oficiales para Android, como pueden ser la App Store de Amazon o la tienda de aplicaciones Samsung Apps de Samsung Google Play es la tienda de aplicaciones en línea administrada por Google, aunque existe la posibilidad de obtener software externamente. Los programas están escritos en el lenguaje de programación Java. No obstante, no es un sistema operativo libre de malware, aunque la mayoría de ello es

descargado de sitios de terceros. El anuncio del sistema Android se realizó el 5 de noviembre de 2007 junto con la creación de la Open Handset Alliance, un consorcio de 78 compañías de hardware, software y telecomunicaciones dedicadas al desarrollo de estándares abiertos para dispositivos móviles. Google liberó la mayoría del código de Android bajo la licencia Apache, una licencia libre y de código abierto.

2.1.7 Versiones de Android.

A continuación se muestran las versiones Android, desde la primera hasta la actual, según (Gironés, El Gran Libro de Android , 2013).

ANDROID 1.0 (Apple Pie, Septiembre 2008)

Es la primera versión del software, aunque nunca se utilizó comercialmente, por lo tanto no tiene caso desarrollar para esta plataforma.

ANDROID 1.1 (Banana Bread, Febrero 2009)

En esta actualización hubo mejoras y agregaron apenas unas funcionalidades cambio a la API y una serie de características pero no obstante es la opción a escoger si queremos desarrollar una APP

compatible con todos los dispositivos Android aunque existen usuarios con esta versión.

Compatible con todos los dispositivos Android aunque existen usuarios con esta versión.

Android 1.5 (Cup cake, Abril 2009)

Esta versión fue lanzada basada en el núcleo de Linux 2.6.27 la actualización incluye correcciones de interfaz de usuario y nuevas características, es la primera versión con algunos usuarios, aunque en la actualidad ya quedan pocos. Como novedades se incorpora la posibilidad del teclado en la pantalla con predicción de texto, ya los terminales ya no tienen que tener un teclado físico también aparecen widgets de escritorio y live folders así mismo la capacidad de grabación avanzada de audio y video.

Android 1.6 (Donut, Septiembre 2009)

Esta versión fue lanzada basado en Linux 2.6.29 y permite capacidades de búsqueda avanzada en todo el dispositivo. Aparece un nuevo atributo XML, onClick, que puede especificarse en una vista. Play Store antes, Android Market se mejora permitiendo una búsqueda más sencilla de aplicaciones, también se facilita que una aplicación pueda trabajar con

diferentes densidades de pantalla. Soporte para una resolución de pantalla WVGA.

Android 2.0/2.1 (Éclair, Octubre 2009)

Esta versión de API cuenta con algunos usuarios, dado que la mayoría de fabricantes pasaron directamente de la versión anterior hasta esta, ósea del 1.6 - 2.1 como novedades cabria mencionar que incorpora una API para manejar el Bluetooth 2.1 nueva funcionalidad que permite sincronizar adaptadores para conectarlo a cualquier dispositivo.

Android 2.2.x (Froyo, Mayo 2010)

Fue lanzado, basado en el núcleo Linux 2.6.32, y como característica destaca que se puede indicar la mejora de velocidad de ejecución de las aplicaciones. También se añaden varias mejoras relacionadas con el navegador web, como el soporte de adobe Flash y el motor JavaScript V8 utilizado en Chrome.

Android 2.3.x (Gingerbread, Diciembre 2010)

Esta versión incorpora un nuevo interfaz de usuario con un diseño actualizado, y mejor la funcionalidad de cortar, copiar y pegar.

Debido al éxito de Android en las nuevas tabletas ahora soportan mayores tamaños de pantalla y resoluciones (WXGA y superiores).

Android 3.0.x (Honeycomb, Febrero 2011)

En esta versión se destaca la nueva interfaz de usuario que ha sido completamente rediseñada con paradigmas nuevos para la interacción y navegación. Versión 3.0 optimizada para dispositivos con pantallas grandes.

Android 3.1 - 3.2 (HoneyComb, Mayo-Julio 2011)

En esta versión permite el manejo de dispositivos por USB y la sincronización desde SD.

Android 4.0 (Ice Cream Sandwich, Octubre 2011)

En esta versión se introduce una nueva interfaz de usuario totalmente renovado

También unifican las dos versiones anteriores para celulares y para las tabletas tres. Así en una sola hacen la compatibilidad para cualquier tipo de dispositivo.

Android 4.0.3 (Ice Cream Sandwich, Diciembre 2011)

En esta versión se introducen nuevas mejoras en algunas API, incluyendo redes sociales, bases de datos, revisor ortográfico entre otros.

Android 4.1 (Jelly Bean, Julio 2012)

En esta versión siguen las mejoras y la principal es la fluidez del interfaz de usuario. Con este propósito se incorporan varias técnicas, una de ellas es: que aumenta la velocidad del procesador al tocar la pantalla.

También hay mejoras en las notificaciones con un sistema de información expandible personalizada, el dictado de voz puede hacerse sin conexión a internet aunque en esta versión todavía en inglés.

Android 4.2 (Jelly Bean, Noviembre 2012)

En esta versión una de las cosas más relevante es la creación de varias cuentas de usuario en un mismo dispositivo, y cada cuenta podrá tener sus propias APP y configuraciones.

Android 4.3 (Jelly Bean, Julio 2013)

En esta versión existen mejoras en distintas áreas. Uno de ella están los perfiles restringidos, disponible solo en tabletas. Esta permite controlar el derecho de los usuarios para el control a datos específicos.

Android 4.4 (Kit Kat, Octubre 2013)

En esta versión su actualización es grande ya que desea hacer Android disponibles en una gama más amplia de dispositivos. Todos los componentes principales de Android han sido recortados para reducir sus requerimientos de memoria una nueva API permite adaptar el comportamiento de la app en dispositivos con poca memoria.

2.1.8 Plataforma de desarrollo Electrónico Arduino.

Arduino nació como un proyecto educativo allá por el año 2005 sin pensar que algunos años más tarde se convertiría en líder del mundo DIY (Do It Yourself).

Su nombre viene del nombre del Bar Bar di Re Arduino donde Massimo Banzi pasaba algunas horas, el cual a su vez viene del nombre de un antiguo rey europeo allá por el año 1002. (Flores, 2013).

(Arduino, 2014) Arduino puede detectar el medio ambiente mediante la recepción de la entrada de una variedad de sensores y puede afectar a sus alrededores por las luces de control, motores, y otros actuadores. El microcontrolador en la placa se programa mediante el lenguaje de programación Arduino (basado en Wiring) y el entorno de desarrollo Arduino (basado en Processing). Proyectos de Arduino pueden ser

autónomos o que se puedan comunicar con el software que se ejecuta en un ordenador (por ejemplo, Flash, Processing, MaxMSP).

Las tablas pueden ser construidas a mano o comprados pre ensamblado; el software se puede descargar de forma gratuita. Los diseños de referencia de hardware (archivos CAD) están disponibles bajo una licencia de código abierto, usted es libre de adaptarlos a sus necesidades.

Arduino recibió una Mención Honorífica en la sección Digital Communitie de la Ars Electrónica Prix 2006.

Los fundadores de Arduino son: Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino y David Mellis.

El software de Arduino es libre, de código abierto y está disponible para Windows, Mac OS y Linux.

Banzi dice que nunca surgió como una idea de negocio, es más nació por una necesidad de subsistir ante el eminente cierre del Instituto de diseño Interactivo IVREA en Italia. Es decir, al crear unos productos open hardware (de uso público) no podría ser embargado. Es más hoy en día Arduino tiene la difícil tarea de subsistir comercialmente y continuar en crecimiento.

A la fecha se han vendido más de 250 mil placas en todo el mundo sin contar las versiones clones y compatibles.

2.1.9 Plataforma de Desarrollo APP Inventor.

Según el profesor (Wolber, 2010), Esta plataforma está basada en un lenguaje de desarrollo grafico lo cual no se escribe ninguna sola fuente, lo que permite ser más fácil de utilizar con tan solo arrastrar una serie de bloques para poder desarrollar distintas acciones.

2.1.10 Categorías de la APP.

Según (Play, 2014) clasifica las APP por categorías que son:

Categoría	Ejemplo
Libros y referencia	Lectores de libros, libros de referencia, libros de texto, diccionarios, tesauros, wikis
Economía	Lectores y editores de documentos, seguimiento de paquetes, escritorio remoto, administración de correo electrónico, búsqueda de empleo
Cómics	Personajes de cómics, libros de cómics.
	Mensajes, chats/mensajería

Comunicaciones	instantánea, marcadores, libretas de direcciones, navegadores, administración de llamadas
Educación	Preparación de exámenes, ayudas para el estudio, vocabulario, juegos.
Ocio	Reproductor de vídeo, películas, programas de TV y contenido de ocio interactivo
Finanzas	Banca, pagos, buscadores de cajeros, noticias financieras, seguros, impuestos, cartera y comercio, calculadoras de propinas
Salud y bienestar	Bienestar personal, seguimiento de ejercicios, dietas y consejos nutricionales, salud y seguridad, etc.
Bibliotecas y demos	Bibliotecas de software, demostraciones técnicas
Estilo de vida	Recetas, guías de estilo

Multimedia y vídeo	Servicios de películas de suscripción, mandos, reproductores de vídeo/multimedia
Medicina	Referencias clínicas y farmacológicas, calculadoras, manuales para el personal sanitario, noticias y revistas médicas.
Música y audio	Servicios musicales, radios, reproductores de música
Noticias y revistas	Periódicos, agregadores de noticias, revistas, blogs
Personalización	Fondos de pantalla, fondos de pantalla animados, pantalla de inicio, pantalla de bloqueo, tonos de llamada
Fotografía	Cámaras, herramientas de edición fotográfica, aplicaciones para administrar y compartir fotos

Productividad	Blocs de notas, listas de tareas, teclados, impresión, calendarios, copias de seguridad, calculadoras, herramientas de conversión
Compras	Compras online, subastas, cupones, comparaciones de precios, listas de la compra, reseñas de productos
Sociedad	Redes sociales, registros de visitas, blogs
Deportes	Comentarios y noticias deportivas, seguimiento de resultados, administración de equipos virtuales, cobertura de partidos
Herramientas	Herramientas para dispositivos Android
Transporte	Transporte público, herramientas de navegación, conducción

Viajes y guías	Mapas, guías de ciudades, información sobre empresas locales, herramientas de administración de viajes
Tiempo	Informes meteorológicos

La aplicación Android que controlará y configurará los mensajes a desplegar en la Pantalla de Led, será de tipo privado o licencia privada (no publica), es decir será una aplicación desarrollada específicamente para este proyecto, no estará disponible vía ningún medio electrónico como Google Play. Y solo se instalará en el dispositivo Android de la persona a cargo del Laboratorio de Hardware de la Universidad Tecnológica de El Salvador (UTEC). Además, basado en la clasificación de Google para desarrolladores de aplicaciones para S.O. Android, la de este proyecto se clasifica como: Nivel de Madurez: Para todos (sin uso de lenguaje maduro, odio o referencia a drogas)

Categoría: Multimedia y Video (Servicios de películas de suscripción, mandos, reproductores de vídeo/multimedia).

2.2 Marco Teórico de Solución.

La solución a presentar en la Universidad Tecnológica De El Salvador (Laboratorio De Hardware) es un sistema configurable vía APP Android para el despliegue de mensajes, controlado por el encargado del laboratorio de Hardware. Los dos bloques que constituyen el proyecto son:

- Pantalla Electrónica.

Estará constituido por dos Módulos los cuales son: Módulo de Bluetooth, Módulo de Matriz de Led y la implementación por medio del ATMEGA328. Para visualizar los mensajes tendrá la Matriz de Led lo cual será manipulado por medio de un dispositivo con sistema Operativo Android.



Imagen 2.2 Despliegue de mensajes vía APP Android
<https://encryptedtbn2.gstatic.com/images>.

Se utilizará una pantalla matriz de leds el cual será de 16X128 así como la muestra la imagen siguiente.



Imagen 2.3 Matriz de Leds

Fuente: <https://data:image/jpeg;base64>

En este apartado se mostrara la realización del firmware donde permitirá la funcionalidad del circuito para poder cumplir con las labores de información visual.

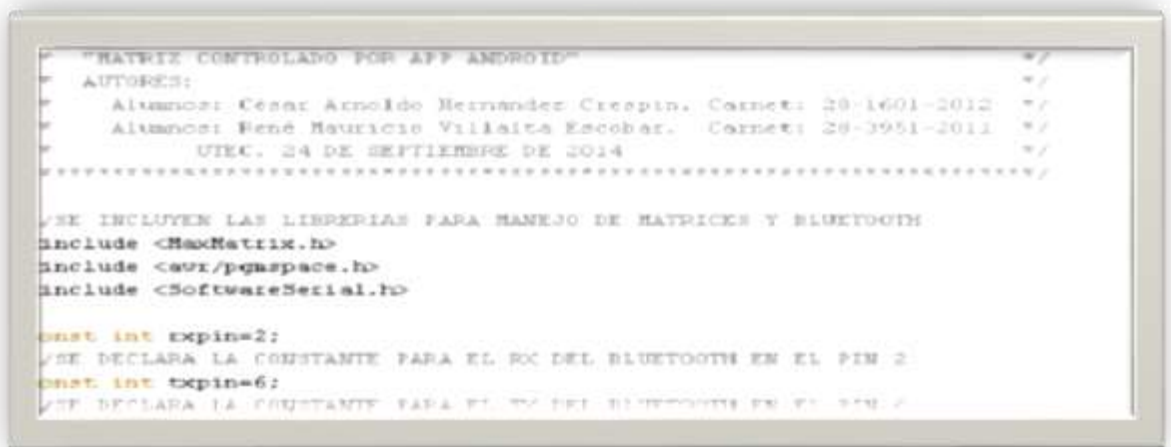


Imagen 2.4 Firmware Compilado

Fuente: Diseño Propio


```

//SE DEFINE UNA VARIABLE QUE ALMACENARÁ LA LUMINOSIDAD DE LA MATRIX, SE INICIALIZA A 5
int inten=5;
//SE DEFINE UNA VARIABLE QUE LEERÁ EL VALOR DEL POTENCIÓMETRO DE VELOCIDAD, SE INICIALIZA A 3
int valpot=0;
//SE DEFINE UNA VARIABLE QUE LEERÁ EL VALOR DEL POTENCIÓMETRO DE LUMINOSIDAD, SE INICIALIZA A 3
int valpot2=6;
//SE DEFINE LAS VARIABLES DE CONFIGURACION DE LA MATRIX, SE INICIALIZAN CON UN NUMERO DE PIN
int data = 5;
int load = 3;
int clock = 4;
//SE DEFINE UNA VARIABLE QUE ESPECIFICA LA CANTIDAD DE MATRICES SE UTILIZADAS, SE INICIALIZA A 4
int maxInUse = 4;
//SE DEFINE UN OBJETO "a" DE TIPO MacMatrix Y SE PASAN LAS VARIABLES DE CONFIGURACION UTILIZADAS ANTERIORMENTE
MacMatrix a(data, load, clock, maxInUse);
//SE DEFINE UN ARREGLO "buffer" DE TIPO BYTE, LLEVARÁ EL CARACTER QUE SE IMPRIME EN CADA INSTANTE
char buffer[10];
//SE DEFINE UN ARREGLO "array" DE TIPO BYTE, CONTENEDOR DE LA FORMA INDICADA DENTRO LA DECLARACION DEPENDIENDO

```

Imagen 2.7 Firmware Compilado

Fuente: Diseño Propio

```

void loop()
{
    //VERIFICA SI HAY DATOS EN EL BLUETOOTH
    if(blueetooth.available() > 0)
    {
        //LLAMA A LA FUNCION QUE LEE EL BLUETOOTH
        leebluetooth();
    }
    //SI NO HAY DATOS EN EL BLUETOOTH
    }else{
        //SE LIMPIA LA MATRIX
        m.clear();
        //LLAMA A LA FUNCION QUE IMPRIME EL MENSAJE RECIBIDO
        imprimemensaje();
    }
}

```

Imagen 2.8 Firmware Compilado

Fuente: Diseño Propio



Imagen 2.9 Firmware Compilado

Fuente: Diseño Propio

Además constará por un controlador, una tarjeta Arduino Mega en el cual será la fuente principal de la logística del funcionamiento como lo muestra la imagen siguiente.

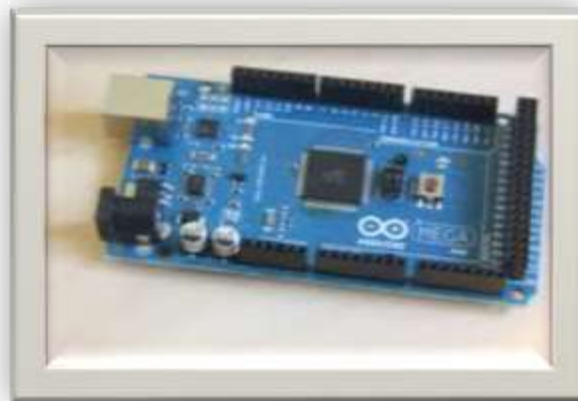


Imagen 2.10 Arduino Mega 2560

Fuente: <https://encrypted-tbn1.gstatic.com/images?>

También se implementara el uso del módulo Bluetooth, lo cual hará la conectividad con el sistema Android en el dispositivo móvil



Imagen 2.11 Módulo Bluetooth HC-06

Fuente: <https://data:image/jpeg;base64>

- Aplicación APP en Android



Imagen 2.12 Dispositivo Android con APP instalada

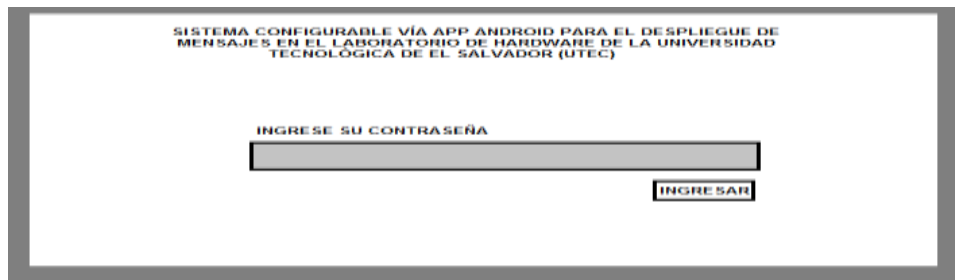
<https://encrypted-tbn1.gstatic.com/images>

La aplicación a utilizar en este proyecto será para sistemas operativo Android desde la versión 2.3 en adelante.

Además la aplicación será de uso privado (solo para el encargado del laboratorio de hardware) no estará disponible para para ningún

sitio web como google play ó play store. La APP será dada e instalada en el dispositivo móvil.

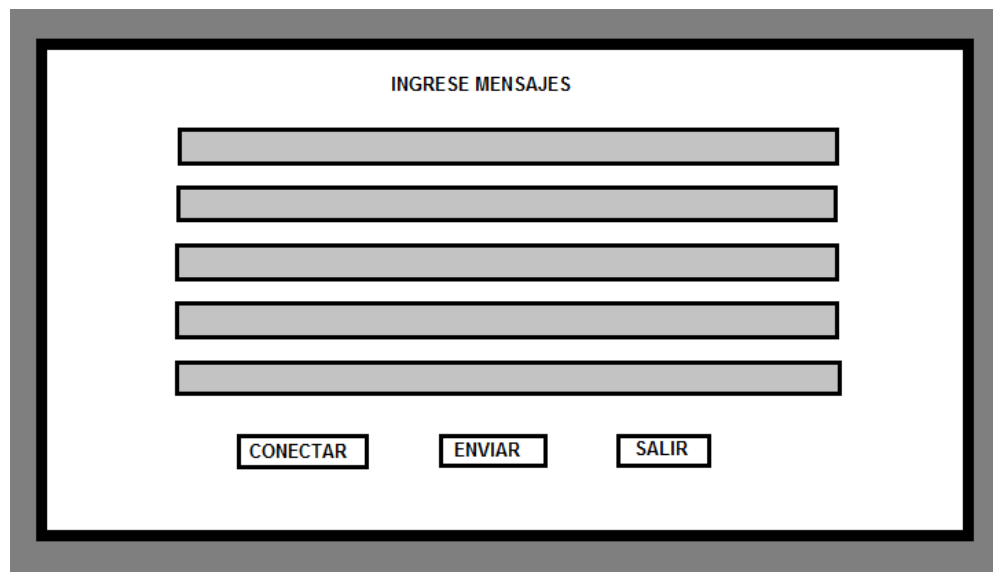
La aplicación funcionará solo al acceder desde un usuario (contraseña) dado exclusivamente al encargado del laboratorio, como lo muestra la imagen siguiente.



The screenshot shows a login interface with the following text at the top: "SISTEMA CONFIGURABLE VÍA APP ANDROID PARA EL DESPLIEGUE DE MENSAJES EN EL LABORATORIO DE HARDWARE DE LA UNIVERSIDAD TECNOLÓGICA DE EL SALVADOR (UTEC)". Below this, the prompt "INGRESE SU CONTRASEÑA" is displayed above a single-line text input field. To the right of the input field is a button labeled "INGRESAR".

Imagen 2.13 APP ingreso de usuario.
Fuente: Diseño Propio

Ya al ingreso a la aplicación mostrará una serie de cuadros en donde se podrá manipular la configuración de mensajes. Además contará con tres botones: Conectividad, enviar y Salida, como los muestra la imagen siguiente.



The screenshot shows a configuration interface with the title "INGRESE MENSAJES" at the top. Below the title are five horizontal text input fields stacked vertically. At the bottom of the screen, there are three buttons labeled "CONECTAR", "ENVIAR", and "SALIR" arranged horizontally.

Imagen 2.14 ingreso a la configuración de mensajes
Fuente: Diseño Propio

2.3 Marco Teórico Conceptual.

A continuación una serie de conceptos, que se han utilizado en el presente trabajo:

Actuador: es un dispositivo capaz de transformar energía hidráulica, neumática o eléctrica en la activación de un proceso con la finalidad de generar un efecto sobre un proceso automatizado.

Android: es el sistema operativo basado en el kernel de Linux diseñado principalmente para dispositivos móviles con pantalla táctil, como teléfonos inteligentes o tabletas, y también para relojes inteligentes, televisores y automóviles, inicialmente desarrollado por Android.

APP Android: es un tipo de programa informático diseñado como herramienta para permitir a un usuario realizar uno o diversos tipos de trabajos.

Alfanuméricos: es un término colectivo que se utiliza para identificar letras del alfabeto latino y de números arábigos.

Arduino: es una plataforma de hardware libre, basada en una placa con un micro-controlador y un entorno de desarrollo, diseñada para facilitar el uso de la electrónica en proyectos multidisciplinarios.

Bus: Es un sistema digital que transfiere datos entre los componentes de una computadora o entre varias computadoras.

Cable: Conductor (generalmente cobre) o conjunto de ellos generalmente recubierto de un material aislante o protector, si bien también se usa el nombre de cable para transmisores de luz (cable de fibra óptica) o esfuerzo mecánico (cable mecánico).

Comunicación: Es el proceso mediante el cual se puede transmitir información de una entidad a otra.

Comunicación inalámbrica: Es aquella en la que extremos de la comunicación (emisor/receptor) no se encuentran unidos por un medio de programación físico, sino que se utiliza la modulación de ondas electromagnéticas a través del espacio.

Controlador: elemento software utilizado en diversos sistemas operativos, también llamado manejador de dispositivo, de vice driver o driver.

Conexión: Acción de conectar un aparato, una máquina u otro sistema eléctrico a su fuente de energía o dos aparatos o partes de un sistema eléctrico entre sí.

Dato: Es un valor o referente que recibe el computador por diferentes medios, los datos representan la información que el programador manipula en la construcción de una solución o en el desarrollo de un algoritmo.

Domótica: la palabra domótica etimológicamente viene de la unión de dos palabras, del latín “DOMUS” que significa casa y AUTOMÁTICA, por lo tanto se denomina como el área de aplicación que tiene por objetivo la automatización de procesos dentro del hogar, además de ser un área multidisciplinaria que requiere conocimientos de electrónica e informática para su implementación.

Electrónica: área de la ingeniería y de la física aplicada al diseño de dispositivos, por lo general circuitos electrónicos, cuyo funcionamiento depende del flujo de electrones haciendo posible la transmisión, recepción almacenamiento de información. Esta información consistir en voz (señales de voz) o en música en un receptor de radio, en una imagen en una pantalla de televisión, o en números u otros datos en una computadora.

Firmware: es un sistema operativo de bajo nivel similar al BIOS en el que todas las aplicaciones más importantes utilizan su funcionalidad. Aunque a diferencia del BIOS en el firmware no hay una versión común disponible, ya que cada firmware es diseñado específicamente para un modelo, y no hay compatibilidad con otras marcas.

Funcionabilidad: Las funciones son aquellas que tiene como objetivo satisfacer las necesidades implícitas o explícitas.

Hardware: es la parte tangible, lo que se puede ver y tocar, de un determinado sistema o equipo electrónico o informático.

Id Arduino: el entorno del código abierto Arduino hace fácil escribir código y cargarlos a la placa E/S. funciona en Windows, Mac OS X y LINUX. El entorno está escrito en java y basado en processing, avr-gcc y otros programas también de código abierto.

Informática: La informática se define como la ciencia que estudia el tratamiento de la información mediante medios automáticos, es decir la ciencia de la información automática.

Interfaz: se utiliza para nombrar a la conexión física y funcional entre dos sistemas o dispositivos de cualquier tipo dando una comunicación entre distintos niveles.

Laboratorio: es un lugar dotado de los medios necesarios para realizar investigaciones, experimentos, prácticas y trabajos de carácter científico, tecnológico o técnico; está equipado con instrumentos de medida o equipos con que se realizan experimentos, investigaciones o prácticas diversas, según la rama de la ciencia a la que se dedique.

Matriz de Led: está constituido por Led dispuestos en filas y columnas, cada Led se conecta a una fila y a una columna.

Memoria: es un dispositivo diseñado y fabricado específicamente para guardar en ella información. La capacidad dependerá del tipo de equipo que se esté implementando, y puede ser que la memoria sea externa o que ya venga incorporada en uno de los dispositivos que se están usando, estas pueden ser ROM, PROM, EPROM, EEPROM, RAM, entre otras.

Mensajes: Está definido como la información que el emisor envía al receptor a través de un canal de comunicación o medio de comunicación determinado (como el habla o la escritura, por ejemplo).

Middleware: Es un software que asiste a una aplicación para interactuar o comunicarse con otras aplicaciones, software, redes, hardware y/o sistemas operativos.

Periférico: son los dispositivos de entrada y salida que permiten la conexión con el entorno físico, como entradas de información y salidas, es decir donde se muestra la función final del equipo o dispositivo.

Prototipo: es un primer modelo o molde original que se fabrica y de este modelo se parte para la elaboración de lo que se pretende, un objeto, herramienta, dispositivo electrónico.

Sensor: es un dispositivo capaz de detectar magnitudes físicas o químicas, llamadas variables de instrumentación, y transformarlas en variables eléctricas.

Sistema: conjunto de partes o cosas coordinadas por reglamentos diseñados para un fin, o una aplicación.

Software: es la parte interna que en algunos casos no se puede ver y no se puede tocar; es el bloque lógico encargado de indicar que es lo que se debe hacer, esto según lo que se le haya programado al dispositivo.

Tecnología: es el conjunto de conocimientos técnicos, científicamente ordenados, que permiten diseñar y crear bienes y servicios que facilitan la adaptación al medio ambiente y satisfacer tanto las necesidades esenciales como los deseos de la humanidad.

Teléfono inteligente: teléfono móvil construido sobre una plataforma informática móvil.

Visualización: es la generación de una imagen mental o una imagen real de algo abstracto o invisible.

2.4 Documentación Técnica.

En la documentación técnica se describen todas las características que poseen cada uno de los componentes que se van a utilizar en este proyecto.

2.4.1 Características de los Componentes del Proyecto.

- Arduino Mega 2560.



Imagen 2.15 Arduino Mega 2560

Fuente: <http://www.bricogeeek.com/arduino-mega-2560.jpg>

Las características principales son:

- Microprocesador ATmega2560
- Tensión de alimentación (recomendado) 7-12V
- Integra regulación y estabilización de +5Vcc
- 54 líneas de Entradas/Salidas Digitales (14 de ellas se pueden utiliza como salidas PWM)
- 16 Entradas Analógicas
- Máxima corriente continua para las entradas: 40 mA
- Salida de alimentación a 3.3V con 50 mA
- Memoria de programa de 256Kb (el bootloader ocupa 8Kb)
- Memoria SRAM de 8Kb para datos y variables del programa
- Memoria EEPROM para datos y variables no volátiles
- Velocidad del reloj de trabajo de 16MHz
- Reducidas dimensiones de 100 x 50 m

- **Módulo de Bluetooth HC-06**



Imagen 2.16 Módulo de Bluetooth HC-06

Fuente: <http://mco-s1-p.mlstatic.com/modulo-bluetooth-hc-06>

El Módulo de Bluetooth funciona como maestro y esclavo o host

- Servicio SPP.
- Configuración vía comando AT.

Características:

- Compatible con el protocolo de bluetooth v2.0.
- Voltaje de alimentación : 1.8v
- Voltaje de operación : 3.3 VDC
- Baud rate ajustable: 1200, 2400, 4800, 19200, 38400, 57600, 115200.
- Tamaño: 1,73 in x 0,63 in x 0.28 in (4.4 cm x 0.7 cm).
- Corrientes de operacion: <40 mA.
- Corrientes modo sleep.

- **Matriz de Led**



Imagen 2.17 Módulo de Matriz de Led

Fuente: <http://www.puntoflotante.net/MLMONWALL3.jpg>

Características:

- Módulo matriz de 7x10 Led fabricado con 2 módulos de 7x5.
- Conexión directa
- sin necesidad de cables al conector para LCD de 14 pines
- compatible con sistema EB88 y Bolt 18F2550.

- **Dispositivo Android.**



Imagen 2.18 teléfono inteligente

Fuente: [http://198.101.200.156/imagesPosts/galeria/1%20\(56\).jpg](http://198.101.200.156/imagesPosts/galeria/1%20(56).jpg)

Características:

- Os: Android os, v2.3 gingerbread
- Navegador : HTML
- 3g hsdpa 7.2 Mbps/hsupa 5.76 Mbps
- Bluetooth V4.0 azdp

- Cable UTP.



Imagen 2.19 Cables de conexión

Fuente: <http://static.mercadoshops.com/cables-para-protoboard-pic-arduino.jpg>

Características:

- Impedancia $100 \text{ ohmios} \pm 5\%$ a 100 MHz
- Conductores de cobre
- Cable delgado y flexible, fácil para cruzar

CAPITULO III DESARROLLO DE LA SOLUCIÓN

3.1 Propuesta de Solución

Al haber estudiado a profundidad la situación problemática en el Laboratorio de Hardware de La Universidad Tecnológica de El Salvador (UTEC), se logró identificar que carece de la falta de un sistema de visualización de mensajes que cumpla como una herramienta de apoyo a las necesidades de información de los alumnos/as con las actividades diarias que se presentaran en dicho laboratorio. Se logró identificar que para evitar inconvenientes en los profesores, encargado del Laboratorio de Hardware ó encargado de poner la ficha informativa en la parte superior de dicho laboratorio. Una solución más viable para evitar la problemática de información del estudiante, es la implementación del proyecto denominado “Sistema Configurable Vía APP Android para el Despliegue de Mensajes en el Laboratorio de Hardware de La Universidad Tecnológica de El Salvador (UTEC)”.

Este proyecto consiste en la visualización de mensajes en una pantalla Led de las actividades diarias que se presentarán en el laboratorio de Hardware, teniendo como producto principal tecnología de última generación combinando hardware y software, Arduino y Android además se contará con una APP que podrá ser instalada en cualquier dispositivo

móvil que posea sistema operativo Android, y que permitirá configurar los mensajes a mostrar en la pantalla Led.

El sistema estará compuesto por una serie de dispositivos electrónicos entre los cuales se pueden mencionar.

- Tarjeta de desarrollo electrónico Arduino Uno.
- Pantalla Led de 2048 pixeles.
- Sensor receptor de señales Bluetooth modelo HC-06.

El funcionamiento que presentará el prototipo estará controlado por la placa electrónica Arduino Uno, por lo cual será programada para poder tener el funcionamiento correcto de los demás componentes y así poder leer de una forma adecuada el módulo de Bluetooth HC-06 y capturar los comandos provenientes del dispositivo móvil, seguido del envío a la pantalla de Led, donde se presentará la serie de mensajes a desplegar, además el sistema tendrá la facilidad de recibir nuevos mensajes por medio del Módulo de Bluetooth, por lo cual se creó una APP diseñada para dispositivos móviles con sistema Android lo cual se contará con la creación de una serie de mensajes y enviarlas al sistema, siendo posible sincronizar los dos dispositivos Bluetooth (El del sistema y el del dispositivo Android móvil a utilizar).

Esto fue implementado gracias a conocimientos obtenidos durante toda una carrera en Técnico En Ingeniería de Hardware, donde las asignaturas como Micro-controladores, Principios de Robótica y Domótica, han ayudado poder desempeñar este proyecto ya que durante esas materias impartidas se vieron una serie de proyectos similares.

Para poder implementar este sistema se tuvo que tomar en cuenta los conocimientos requeridos en cuanto a electrónica, Sistemas Embebidos y Sistemas Domótico siendo implementados con la plataforma Arduino Uno. Y teniendo conocimientos sobre la creación de APP Android.

3.1.1 Firmware de control general

Como se puede observar en todo circuito domótico o inmótico debe de contar con un microcontrolador por lo cual es necesario crear un firmware con sus instrucciones bien detalladas para obtener un buen funcionamiento con los diferentes componentes electrónicos a utilizarse, es por ello que además para que nuestro proyecto funcione a la perfección fue necesario la creación de un firmware en el software de Arduino para poder permitir la detección de cada componente, a continuación se presentara el código programado que se utilizará en la tarjeta Arduino.

3.1.2 Código Arduino

```
//////*****  
//Envío de Mensajes a Pantalla Led  
//Código en lenguaje Arduino C  
//Estudiantes:  
//Cesar Arnoldo Hernández Crespín  
//René Mauricio Villalta Escobar  
//Asesor Técnico:  
// Ing. Julio Cesar Rosales  
// Arduino Uno  
  
#include <pgmspace.h>  
#include <SoftwareSerial.h>  
  
SoftwareSerial BTSerial(14, 15); // RX | TX  
#define COLUMNS 16  
#define HEIGHT 16  
int indice=0;  
// Connections to board  
const byte latchPin=8;  
const byte clockPin=12;  
const byte data_R1=10;  
const byte data_R2=11;  
const byte en_74138=2;  
const byte la_74138=3;
```

```

const byte lb_74138=4;
const byte lc_74138=5;
const byte ld_74138=6;
byte ScanRow = 0;
unsigned long counter;

```

```

char message[] = "Laboratorio Hardware UTEC---Laboratorio Hardware
UTEC---Laboratorio Hardware UTEC---Laboratorio Hardware UTEC---
Laboratorio Hardware UTEC---Laboratorio Hardware UTEC---";

```

```

// Mensajes por defecto

```

```

byte buffer[256] = { //buffer a escaner en la pantalla

```

```

    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,

```

```

    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,

```

```

    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,

```

```

    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,

```

```

    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,

```

```

    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,

```

```

    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,

```

```

    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,

```

```

    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,

    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,

    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,

    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,

    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,

    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,

    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00,

};

// -----
--

// Fuentes de Carácteres

const uint8_t font8x16_basic[] PROGMEM = {

    /* (32) */ 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

    /* ! (33) */ 0x00, 0x00, 0x00, 0x18, 0x3C, 0x3C, 0x3C, 0x18, 0x18,
    0x18, 0x00, 0x18, 0x18, 0x00, 0x00, 0x00,

    /* " (34) */ 0x00, 0x00, 0x66, 0x66, 0x66, 0x24, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

    /* # (35) */ 0x00, 0x00, 0x00, 0x00, 0x6C, 0x6C, 0xFE, 0x6C, 0x6C,
    0x6C, 0xFE, 0x6C, 0x6C, 0x00, 0x00, 0x00,

    /* $ (36) */ 0x00, 0x18, 0x18, 0x7C, 0xC6, 0xC2, 0xC0, 0x7C, 0x06,
    0x06, 0x86, 0xC6, 0x7C, 0x18, 0x18, 0x00,

```

/* % (37) */ 0x00, 0x00, 0x00, 0x00, 0x00, 0xC2, 0xC6, 0x0C, 0x18,
0x30, 0x60, 0xC6, 0x86, 0x00, 0x00, 0x00,

/* & (38) */ 0x00, 0x00, 0x00, 0x38, 0x6C, 0x6C, 0x38, 0x76, 0xDC,
0xCC, 0xCC, 0xCC, 0x76, 0x00, 0x00, 0x00,

/* ' (39) */ 0x00, 0x00, 0x30, 0x30, 0x30, 0x60, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

/* ((40) */ 0x00, 0x00, 0x00, 0x0C, 0x18, 0x30, 0x30, 0x30, 0x30,
0x30, 0x30, 0x18, 0x0C, 0x00, 0x00, 0x00,

/*) (41) */ 0x00, 0x00, 0x00, 0x30, 0x18, 0x0C, 0x0C, 0x0C, 0x0C,
0x0C, 0x0C, 0x18, 0x30, 0x00, 0x00, 0x00,

/* * (42) */ 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x66, 0x3C, 0xFF,
0x3C, 0x66, 0x00, 0x00, 0x00, 0x00, 0x00,

/* + (43) */ 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x18, 0x18, 0x7E,
0x18, 0x18, 0x00, 0x00, 0x00, 0x00, 0x00,

/* , (44) */ 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x18, 0x18, 0x18, 0x30, 0x00, 0x00,

/* - (45) */ 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFE,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

/* . (46) */ 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x18, 0x18, 0x00, 0x00, 0x00,

/* / (47) */ 0x00, 0x00, 0x00, 0x00, 0x00, 0x02, 0x06, 0x0C, 0x18,
0x30, 0x60, 0xC0, 0x80, 0x00, 0x00, 0x00,

/* 0 (48) */ 0x00, 0x00, 0x00, 0x3C, 0x66, 0xC3, 0xC3, 0xDB, 0xDB,
0xC3, 0xC3, 0x66, 0x3C, 0x00, 0x00, 0x00,

/* 1 (49) */ 0x00, 0x00, 0x00, 0x18, 0x38, 0x78, 0x18, 0x18, 0x18,
0x18, 0x18, 0x18, 0x7E, 0x00, 0x00, 0x00,

/* 2 (50) */ 0x00, 0x00, 0x00, 0x7C, 0xC6, 0x06, 0x0C, 0x18, 0x30,
0x60, 0xC0, 0xC6, 0xFE, 0x00, 0x00, 0x00,

/* 3 (51) */ 0x00, 0x00, 0x00, 0x7C, 0xC6, 0x06, 0x06, 0x3C, 0x06,
0x06, 0x06, 0xC6, 0x7C, 0x00, 0x00, 0x00,

/* 4 (52) */ 0x00, 0x00, 0x00, 0x0C, 0x1C, 0x3C, 0x6C, 0xCC, 0xFE,
0x0C, 0x0C, 0x0C, 0x1E, 0x00, 0x00, 0x00,

/* 5 (53) */ 0x00, 0x00, 0x00, 0xFE, 0xC0, 0xC0, 0xC0, 0xFC, 0x06,
0x06, 0x06, 0xC6, 0x7C, 0x00, 0x00, 0x00,

/* 6 (54) */ 0x00, 0x00, 0x00, 0x38, 0x60, 0xC0, 0xC0, 0xFC, 0xC6,
0xC6, 0xC6, 0xC6, 0x7C, 0x00, 0x00, 0x00,

/* 7 (55) */ 0x00, 0x00, 0x00, 0xFE, 0xC6, 0x06, 0x06, 0x0C, 0x18,
0x30, 0x30, 0x30, 0x30, 0x00, 0x00, 0x00,

/* 8 (56) */ 0x00, 0x00, 0x00, 0x7C, 0xC6, 0xC6, 0xC6, 0x7C, 0xC6,
0xC6, 0xC6, 0xC6, 0x7C, 0x00, 0x00, 0x00,

/* 9 (57) */ 0x00, 0x00, 0x00, 0x7C, 0xC6, 0xC6, 0xC6, 0x7E, 0x06,
0x06, 0x06, 0x0C, 0x78, 0x00, 0x00, 0x00,

/* : (58) */ 0x00, 0x00, 0x00, 0x00, 0x00, 0x18, 0x18, 0x00, 0x00,
0x00, 0x18, 0x18, 0x00, 0x00, 0x00, 0x00,

/* ; (59) */ 0x00, 0x00, 0x00, 0x00, 0x00, 0x18, 0x18, 0x00, 0x00,
0x00, 0x18, 0x18, 0x30, 0x00, 0x00, 0x00,

/* < (60) */ 0x00, 0x00, 0x00, 0x00, 0x06, 0x0C, 0x18, 0x30, 0x60,
0x30, 0x18, 0x0C, 0x06, 0x00, 0x00, 0x00,

/* = (61) */ 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x7E, 0x00, 0x00,
0x7E, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

/* > (62) */ 0x00, 0x00, 0x00, 0x00, 0x60, 0x30, 0x18, 0x0C, 0x06,
0x0C, 0x18, 0x30, 0x60, 0x00, 0x00, 0x00,

/* ? (63) */ 0x00, 0x00, 0x00, 0x7C, 0xC6, 0xC6, 0x0C, 0x18, 0x18,
0x18, 0x00, 0x18, 0x18, 0x00, 0x00, 0x00,

/* @ (64) */ 0x00, 0x00, 0x00, 0x00, 0x7C, 0xC6, 0xC6, 0xDE,
0xDE, 0xDE, 0xDC, 0xC0, 0x7C, 0x00, 0x00, 0x00,

/* A (65) */ 0x00, 0x00, 0x00, 0x10, 0x38, 0x6C, 0xC6, 0xC6, 0xFE,
0xC6, 0xC6, 0xC6, 0xC6, 0x00, 0x00, 0x00,

/* B (66) */ 0x00, 0x00, 0x00, 0xFC, 0x66, 0x66, 0x66, 0x7C, 0x66,
0x66, 0x66, 0x66, 0xFC, 0x00, 0x00, 0x00,

/* C (67) */ 0x00, 0x00, 0x00, 0x3C, 0x66, 0xC2, 0xC0, 0xC0, 0xC0,
0xC0, 0xC2, 0x66, 0x3C, 0x00, 0x00, 0x00,

/* D (68) */ 0x00, 0x00, 0x00, 0xF8, 0x6C, 0x66, 0x66, 0x66, 0x66,
0x66, 0x66, 0x6C, 0xF8, 0x00, 0x00, 0x00,

/* E (69) */ 0x00, 0x00, 0x00, 0xFE, 0x66, 0x62, 0x68, 0x78, 0x68,
0x60, 0x62, 0x66, 0xFE, 0x00, 0x00, 0x00,

/* F (70) */ 0x00, 0x00, 0x00, 0xFE, 0x66, 0x62, 0x68, 0x78, 0x68,
0x60, 0x60, 0x60, 0xF0, 0x00, 0x00, 0x00,

/* G (71) */ 0x00, 0x00, 0x00, 0x3C, 0x66, 0xC2, 0xC0, 0xC0, 0xDE,
0xC6, 0xC6, 0x66, 0x3A, 0x00, 0x00, 0x00,

/* H (72) */ 0x00, 0x00, 0x00, 0xC6, 0xC6, 0xC6, 0xC6, 0xFE, 0xC6,
0xC6, 0xC6, 0xC6, 0xC6, 0x00, 0x00, 0x00,

/* I (73) */ 0x00, 0x00, 0x00, 0x3C, 0x18, 0x18, 0x18, 0x18, 0x18,
0x18, 0x18, 0x18, 0x3C, 0x00, 0x00, 0x00,

/* J (74) */ 0x00, 0x00, 0x00, 0x1E, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C,
0xCC, 0xCC, 0xCC, 0x78, 0x00, 0x00, 0x00,

/* K (75) */ 0x00, 0x00, 0x00, 0xE6, 0x66, 0x66, 0x6C, 0x78, 0x78,
0x6C, 0x66, 0x66, 0xE6, 0x00, 0x00, 0x00,

/* L (76) */ 0x00, 0x00, 0x00, 0xF0, 0x60, 0x60, 0x60, 0x60, 0x60,
0x60, 0x62, 0x66, 0xFE, 0x00, 0x00, 0x00,

/* M (77) */ 0x00, 0x00, 0x00, 0xC3, 0xE7, 0xFF, 0xFF, 0xDB, 0xC3,
0xC3, 0xC3, 0xC3, 0xC3, 0x00, 0x00, 0x00,

/* N (78) */ 0x00, 0x00, 0x00, 0xC6, 0xE6, 0xF6, 0xFE, 0xDE, 0xCE,
0xC6, 0xC6, 0xC6, 0xC6, 0x00, 0x00, 0x00,

/* O (79) */ 0x00, 0x00, 0x00, 0x7C, 0xC6, 0xC6, 0xC6, 0xC6, 0xC6,
0xC6, 0xC6, 0xC6, 0x7C, 0x00, 0x00, 0x00,

/* P (80) */ 0x00, 0x00, 0x00, 0xFC, 0x66, 0x66, 0x66, 0x7C, 0x60,
0x60, 0x60, 0x60, 0xF0, 0x00, 0x00, 0x00,

/* Q (81) */ 0x00, 0x00, 0x00, 0x7C, 0xC6, 0xC6, 0xC6, 0xC6, 0xC6,
0xC6, 0xD6, 0xDE, 0x7C, 0x0C, 0x0E, 0x00,

```

/* R (82) */ 0x00, 0x00, 0x00, 0xFC, 0x66, 0x66, 0x66, 0x7C, 0x6C,
0x66, 0x66, 0x66, 0xE6, 0x00, 0x00, 0x00,

/* S (83) */ 0x00, 0x00, 0x00, 0x7C, 0xC6, 0xC6, 0x60, 0x38, 0x0C,
0x06, 0xC6, 0xC6, 0x7C, 0x00, 0x00, 0x00,

/* T (84) */ 0x00, 0x00, 0x00, 0xFF, 0xDB, 0x99, 0x18, 0x18, 0x18,
0x18, 0x18, 0x18, 0x3C, 0x00, 0x00, 0x00,

/* U (85) */ 0x00, 0x00, 0x00, 0xC6, 0xC6, 0xC6, 0xC6, 0xC6, 0xC6,
0xC6, 0xC6, 0xC6, 0x7C, 0x00, 0x00, 0x00,

/* V (86) */ 0x00, 0x00, 0x00, 0xC3, 0xC3, 0xC3, 0xC3, 0xC3, 0xC3,
0xC3, 0x66, 0x3C, 0x18, 0x00, 0x00, 0x00,

/* W (87) */ 0x00, 0x00, 0x00, 0xC3, 0xC3, 0xC3, 0xC3, 0xC3,
0xDB, 0xDB, 0xFF, 0x66, 0x66, 0x00, 0x00, 0x00,

/* X (88) */ 0x00, 0x00, 0x00, 0xC3, 0xC3, 0x66, 0x3C, 0x18, 0x18,
0x3C, 0x66, 0xC3, 0xC3, 0x00, 0x00, 0x00,

/* Y (89) */ 0x00, 0x00, 0x00, 0xC3, 0xC3, 0xC3, 0x66, 0x3C, 0x18,
0x18, 0x18, 0x18, 0x3C, 0x00, 0x00, 0x00,

/* Z (90) */ 0x00, 0x00, 0x00, 0xFF, 0xC3, 0x86, 0x0C, 0x18, 0x30,
0x60, 0xC1, 0xC3, 0xFF, 0x00, 0x00, 0x00,

/* [ (91) */ 0x00, 0x00, 0x00, 0x3C, 0x30, 0x30, 0x30, 0x30, 0x30,
0x30, 0x30, 0x30, 0x3C, 0x00, 0x00, 0x00,

/* \ (92) */ 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0x70, 0x38,
0x1C, 0x0E, 0x06, 0x02, 0x00, 0x00, 0x00,

/* ] (93) */ 0x00, 0x00, 0x00, 0x3C, 0x0C, 0x0C, 0x0C, 0x0C, 0x0C,
0x0C, 0x0C, 0x0C, 0x3C, 0x00, 0x00, 0x00,

/* ^ (94) */ 0x00, 0x10, 0x38, 0x6C, 0xC6, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

/* _ (95) */ 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0x00,

/* ` (96) */ 0x00, 0x30, 0x30, 0x18, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

```

```

/* a (97) */ 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x78, 0x0C, 0x7C,
0xCC, 0xCC, 0xCC, 0x76, 0x00, 0x00, 0x00,

/* b (98) */ 0x00, 0x00, 0x00, 0xE0, 0x60, 0x60, 0x78, 0x6C, 0x66,
0x66, 0x66, 0x66, 0x7C, 0x00, 0x00, 0x00,

/* c (99) */ 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x7C, 0xC6, 0xC0,
0xC0, 0xC0, 0xC6, 0x7C, 0x00, 0x00, 0x00,

/* d (100) */ 0x00, 0x00, 0x00, 0x1C, 0x0C, 0x0C, 0x3C, 0x6C,
0xCC, 0xCC, 0xCC, 0xCC, 0x76, 0x00, 0x00, 0x00,

/* e (101) */ 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x7C, 0xC6, 0xFE,
0xC0, 0xC0, 0xC6, 0x7C, 0x00, 0x00, 0x00,

/* f (102) */ 0x00, 0x00, 0x00, 0x38, 0x6C, 0x64, 0x60, 0xF0, 0x60,
0x60, 0x60, 0x60, 0xF0, 0x00, 0x00, 0x00,

/* g (103) */ 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x76, 0xCC, 0xCC,
0xCC, 0xCC, 0xCC, 0x7C, 0x0C, 0xCC, 0x78,

/* h (104) */ 0x00, 0x00, 0x00, 0xE0, 0x60, 0x60, 0x6C, 0x76, 0x66,
0x66, 0x66, 0x66, 0xE6, 0x00, 0x00, 0x00,

/* i (105) */ 0x00, 0x00, 0x00, 0x18, 0x18, 0x00, 0x38, 0x18, 0x18,
0x18, 0x18, 0x18, 0x3C, 0x00, 0x00, 0x00,

/* j (106) */ 0x00, 0x00, 0x00, 0x06, 0x06, 0x00, 0x0E, 0x06, 0x06,
0x06, 0x06, 0x06, 0x06, 0x66, 0x66, 0x3C,

/* k (107) */ 0x00, 0x00, 0x00, 0xE0, 0x60, 0x60, 0x66, 0x6C, 0x78,
0x78, 0x6C, 0x66, 0xE6, 0x00, 0x00, 0x00,

/* l (108) */ 0x00, 0x00, 0x00, 0x38, 0x18, 0x18, 0x18, 0x18, 0x18,
0x18, 0x18, 0x18, 0x3C, 0x00, 0x00, 0x00,

/* m (109) */ 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE6, 0xFF,
0xDB, 0xDB, 0xDB, 0xDB, 0xDB, 0x00, 0x00, 0x00,

/* n (110) */ 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xDC, 0x66, 0x66,
0x66, 0x66, 0x66, 0x66, 0x00, 0x00, 0x00,

/* o (111) */ 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x7C, 0xC6, 0xC6,
0xC6, 0xC6, 0xC6, 0x7C, 0x00, 0x00, 0x00,

```

```

/* p (112) */ 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xDC, 0x66, 0x66,
0x66, 0x66, 0x66, 0x7C, 0x60, 0x60, 0xF0,

/* q (113) */ 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x76, 0xCC, 0xCC,
0xCC, 0xCC, 0xCC, 0x7C, 0x0C, 0x0C, 0x1E,

/* r (114) */ 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xDC, 0x76, 0x66,
0x60, 0x60, 0x60, 0xF0, 0x00, 0x00, 0x00,

/* s (115) */ 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x7C, 0xC6, 0x60,
0x38, 0x0C, 0xC6, 0x7C, 0x00, 0x00, 0x00,

/* t (116) */ 0x00, 0x00, 0x00, 0x10, 0x30, 0x30, 0xFC, 0x30, 0x30,
0x30, 0x30, 0x36, 0x1C, 0x00, 0x00, 0x00,

/* u (117) */ 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xCC, 0xCC,
0xCC, 0xCC, 0xCC, 0x76, 0x00, 0x00, 0x00,

/* v (118) */ 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC3, 0xC3, 0xC3,
0xC3, 0x66, 0x3C, 0x18, 0x00, 0x00, 0x00,

/* w (119) */ 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC3, 0xC3, 0xC3,
0xDB, 0xDB, 0xFF, 0x66, 0x00, 0x00, 0x00,

/* x (120) */ 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC3, 0x66, 0x3C,
0x18, 0x3C, 0x66, 0xC3, 0x00, 0x00, 0x00,

/* y (121) */ 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC6, 0xC6, 0xC6,
0xC6, 0xC6, 0xC6, 0x7E, 0x06, 0x0C, 0xF8,

/* z (122) */ 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFE, 0xCC, 0x18,
0x30, 0x60, 0xC6, 0xFE, 0x00, 0x00, 0x00,

/* { (123) */ 0x00, 0x00, 0x00, 0x0E, 0x18, 0x18, 0x18, 0x70, 0x18,
0x18, 0x18, 0x18, 0x0E, 0x00, 0x00, 0x00,

/* | (124) */ 0x00, 0x00, 0x00, 0x18, 0x18, 0x18, 0x18, 0x00, 0x18,
0x18, 0x18, 0x18, 0x00, 0x00, 0x00,

/* } (125) */ 0x00, 0x00, 0x00, 0x70, 0x18, 0x18, 0x18, 0x0E, 0x18,
0x18, 0x18, 0x18, 0x70, 0x00, 0x00, 0x00,

/* ~ (126) */ 0x00, 0x00, 0x00, 0x76, 0xDC, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

```

```

        /* BLK(127)*/ 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
        0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
    };

```

```

byte tam;

```

```

// Rutina de Carga de Mensaje en Buffer

```

```

void drawChar(uint16_t x, uint16_t y, byte n, byte font, boolean inverse) {
    byte charbytes[16], fontrows;
    int index;
    if (0 != (x % 8)) return;
    if ((n>9) && (n<32)) return;
    if (font==2) fontrows=16; else fontrows=8;
    if ((n>=0) && (n<=9)) index=(n+16)*fontrows; else index=(n-
32)*fontrows;
    byte *pDst = buffer + (y*COLUMNS) + (x>>3);
    for (byte i=0;i<fontrows;i++) {
        if (font==0) charbytes[i] = pgm_read_byte(&(font8x16_basic[index+i]));
        if (font==2) charbytes[i] = pgm_read_byte(&(font8x16_basic[index+i]));
        if (font != 2) {
            charbytes[i] = (charbytes[i] & 0xF0) >> 4 | (charbytes[i] & 0x0F) << 4;
            charbytes[i] = (charbytes[i] & 0xCC) >> 2 | (charbytes[i] & 0x33) << 2;
            charbytes[i] = (charbytes[i] & 0xAA) >> 1 | (charbytes[i] & 0x55) << 1;
        };
        if (inverse) charbytes[i] = ~charbytes[i];
    };
    const byte *pSrc = charbytes;

```

```

    for (byte i=0; i<fontrows; i++) {
        *pDst = *pSrc;
        pDst += COLUMNS;
        pSrc++;
    }
};

// Rutina para efecto Scroll
void moveLeft(byte pixels, byte rowstart, byte rowstop) {
    byte row, column;
    short unsigned int address;
    for (column=0;column<COLUMNS;column++) {
        for (row=rowstart;row<=rowstop;row++) {
            address = (row*COLUMNS) + column;
            if (column==(COLUMNS-1))
                buffer[address] = buffer[address]<<pixels;
            else {
                byte incomingchar = buffer[address+1];
                buffer[address] = buffer[address]<<pixels;
                for (byte x=0;x<pixels;x++) { buffer[address] += ((incomingchar &
(128>>x)) >> (7-x))<<(pixels-x-1); };
            }
        }
    }
};

void setPixel(byte x, byte y, byte colour) {
    bitWrite(buffer[(y*COLUMNS)+(x>>3)],7-(x&7),colour); };

```

```

void drawRect(byte x1, byte y1, byte x2, byte y2, byte colour) {
    for (byte x=x1;x<=x2;x++) { setPixel(x,y1,colour); setPixel(x,y2,colour); };
    for (byte y=y1;y<=y2;y++) { setPixel(x1,y,colour); setPixel(x2,y,colour); };
};

void shiftOut(byte row) { //envio de mensaje a la pantalla
byte rtimesc = row*COLUMNS;
    for(byte column=0;column<COLUMNS;column++) {
        byte index = column + rtimesc;
        for(byte i=0;i<8;i++) {
            PORTB &=~(3<<(data_R1-8));
            PORTB &=~(1<<(clockPin-8));
            PORTB |= !((buffer[index]>>(7-i)) & 0x01) << (data_R1-8);
            if (HEIGHT == 32) PORTB |= !((buffer[index+128]>>(7-i)) & 0x01) <<
(data_R2-8);
            PORTB |= 1<<(clockPin-8);
        };
    };
};

ISR(TIMER2_COMPA_vect){
    cli();
    digitalWrite(en_74138, HIGH);
    shiftOut(ScanRow);
    digitalWrite(latchPin, LOW);
    digitalWrite(latchPin, HIGH);
    PORTD = (ScanRow << 3) | (PORTD & 0X87);
}

```

```

    digitalWrite(en_74138, LOW);
    ScanRow++;
    ScanRow &= 0x0F;
    sei();
};

void setup() {

    tam = sizeof(message);
    Serial.begin(9600);
    cli();
    TCCR2A = 0; TCCR2B = 0; TCNT2 = 0;
    TCCR2B |= (1 << CS12) | (1 << CS10);
    OCR2A = 97;
    TCCR2A |= (1 << WGM21); TIMSK2 |= (1 << OCIE2A);

    pinMode(latchPin,OUTPUT); pinMode(clockPin,OUTPUT);
    pinMode(data_R1,OUTPUT); pinMode(data_R2,OUTPUT);

    pinMode(en_74138,OUTPUT);
    pinMode(la_74138,OUTPUT); pinMode(lb_74138,OUTPUT);
    pinMode(lc_74138,OUTPUT); pinMode(ld_74138,OUTPUT);

    digitalWrite(en_74138, LOW);
    digitalWrite(data_R1, HIGH); digitalWrite(data_R2, HIGH);
    counter = millis();
    sei(); //permitir interrupciones

```

```

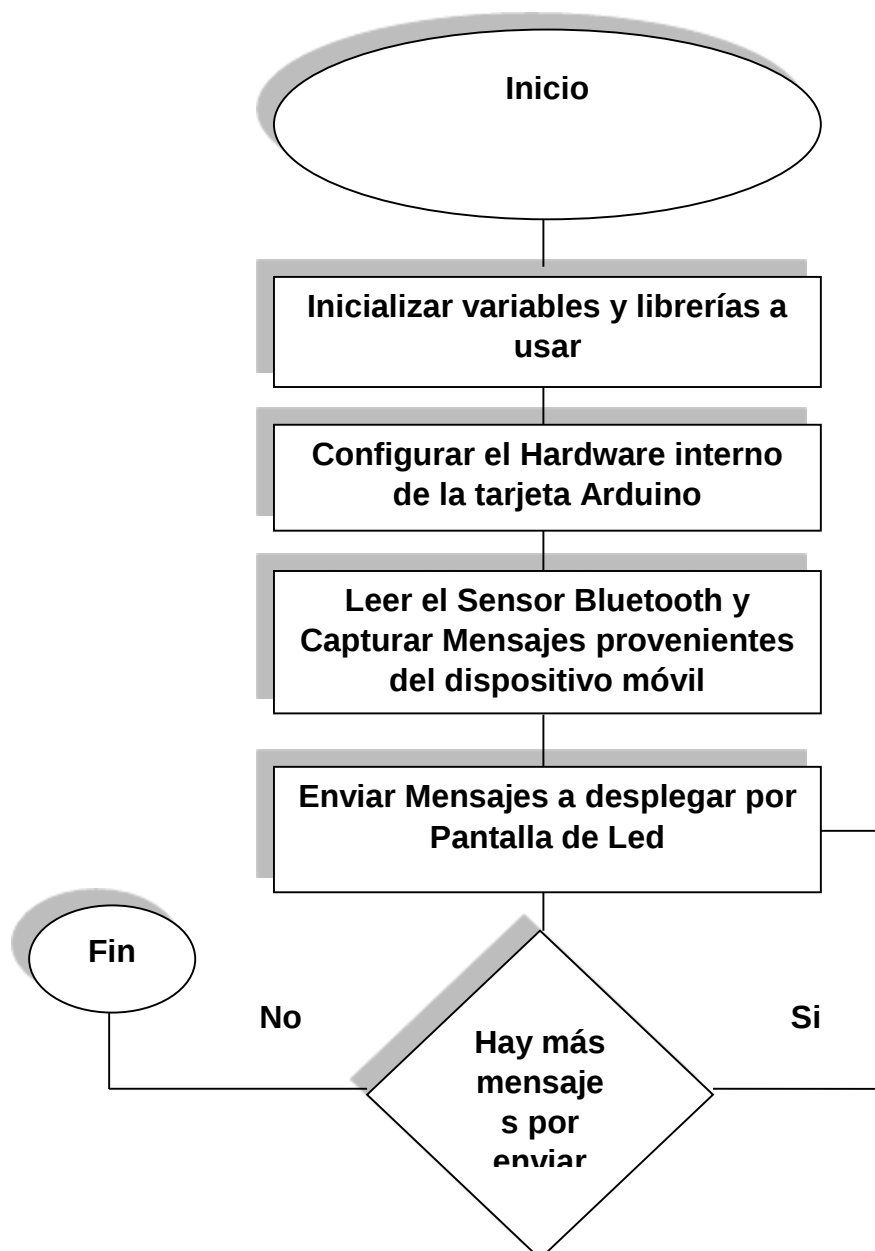
};
void loop() {
  if (Serial.available())
  {
    for (byte indicador=0; indicador<tam; indicador++) {
      message[indicador]=' ';
    };
    tam = Serial.readBytesUntil('.', message, 100);
  }
  drawChar(120,0,message[count%(tam-1)],2,false);
  count = (count + 1);
  for (byte i=0; i<8; i++) {
    delay(15);
    moveLeft(1,0,15);
  };
};

```

```
//FIN DEL CODIGO FUENTE
```

3.1.3 Flujograma de Funcionamiento del Sistema Configurable Vía APP Android para el Despliegue de Mensajes.

A continuación se presenta un flujo grama detallando la estructura para el buen funcionamiento del firmware.



3.1.4 Diagrama Electrónico del Sistema Configurable Vía APP Android para el Despliegue de Mensajes

En el siguiente diagrama, se presenta como está estructurado cada componente electrónico y de qué forma irán las conexiones para el buen funcionamiento del dispositivo

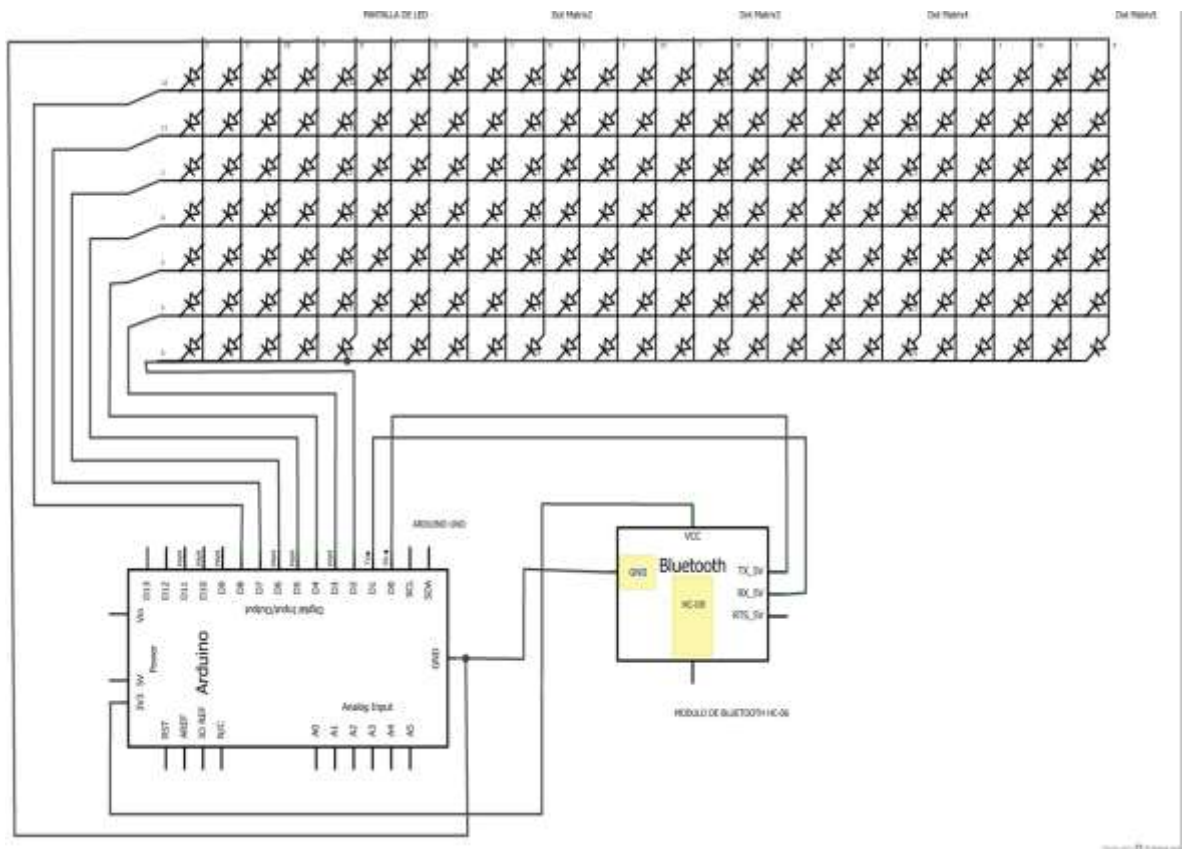


Imagen 3.1 Diagrama del Prototipo implementado

Fuente: Diseño Propio.

3.1.5 APLICACIÓN MOVIL (APP)

APP para administrar los mensajes en la pantalla Led

La APP será desarrollada únicamente para este proyecto, ya que estará desarrollada para el despliegue y configuración de mensajes en el laboratorio de Hardware de La Universidad Tecnológica de El Salvador (UTEC), basado en la plataforma de Arduino, por eso es muy bien tomar en cuenta hacer un poco dinámico la configuración de mensajes a través de una APP que será instalada en un dispositivo móvil Android (Smartphone), de la persona encargada del laboratorio de Hardware. Lo que se pretende es que al final se pueda modificar los mensajes sin necesidad de asistencia técnica.

Además la aplicación fue desarrollada usando el IDE APP Inventor, por lo cual se muestra la forma de cómo se desarrollo la APP.

3.1.6 Pantalla principal

En la pantalla principal se mostrará como inicio, la bienvenida.

También en la primera pantalla nos mostrará lo que son tres botones lo que serán los encargados de ingresar como primer opción al botón de Ayuda, como segunda opción, al botón de mensajes donde al darle clic

mostrará una pantalla de acceso de usuario para poder ingresar después a la pantalla de envío de mensajes, también tendrá un botón para poder salir de la APP.



Imagen 3. 2 pantalla y código principal de la APP

Fuente: Diseño Propio.

3.1.7 Pantalla de ayuda

En esta pantalla se mostrará la forma adecuada de cómo poder utilizar la APP para este proyecto ya que es muy necesario hacer los pasos requeridos para no tener ningún inconveniente a la hora de instalarla en su dispositivo móvil Android.



Imagen 3.3 pantalla de ayuda y código de la APP

Fuente: Diseño Propio.

3.1.8 Pantalla de ingreso de contraseña

En esta pantalla se mostrará el ingreso de contraseña, donde debe de ingresar la contraseña adecuada ya que esta será la que llevará a la siguiente pantalla donde se escribirán los mensajes que se enviarán a la pantalla Led, la clave solo puede ser cambiada vía código APP.

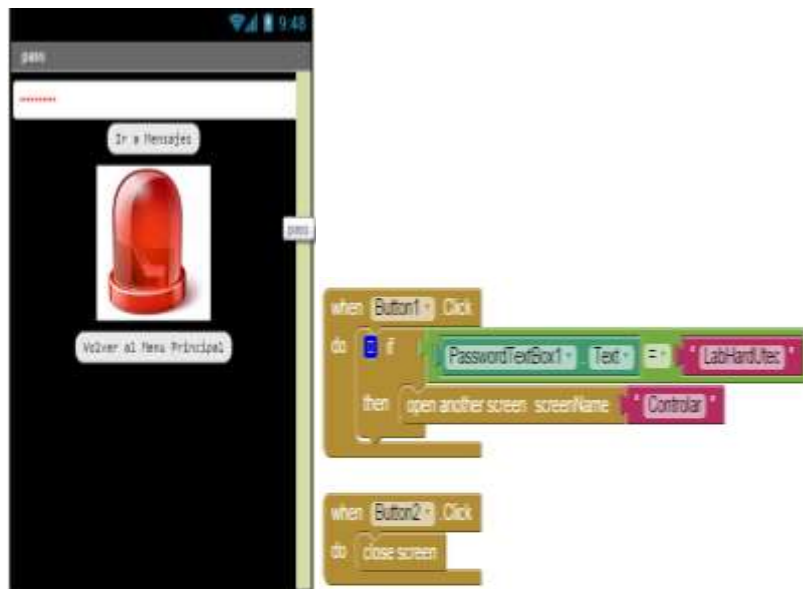


Imagen 3.4 Pantalla de ingreso de contraseña y código de la APP

Fuente: Diseño Propio.

3.1.9 Pantalla de envío de mensajes

En esta pantalla se mostrará una serie de cuadros para poder introducir los mensajes á enviar a la pantalla Led, por lo cual contará con la utilización de 3 botones que serán conectar, enviar y menú principal.

Con el primer botón se obtendrá la vinculación entre el dispositivo móvil Android y el módulo de Bluetooth, con el segundo botón se enviaran los mensajes asía la pantalla Led y como tercer botón tendremos para ir al menú principal.



Imagen 3.5 Pantalla de envío de mensajes y código de la APP

Fuente: Diseño Propio

3.1.10 Implementaciones

El siguiente proyecto se implementará en el laboratorio de Hardware instalado correctamente y funcionando adecuadamente para las necesidades diarias a utilizar, ya que el trabajo siguiente es denominado como “Sistema configurable vía APP Android para el despliegue de mensajes en el laboratorio de Hardware de la Universidad Tecnológica (UTEC)” será implementado adecuadamente.

3.1.11 Elementos del prototipo

- **Arduino Uno:** este componente es el cerebro de todo el circuito electrónico ya que es donde esta implementado el firmware para que cada uno de los componentes tenga un correcto funcionamiento.



Imagen 3.6 Arduino Uno
Fuente: Diseño Propio

- **Pantalla Led:** tendrá como objetivo brindar el despliegue de mensajes para los recursos de Alumnos/as y catedráticos.



Imagen 3.7 Pantalla Led
Fuente: Diseño Propio

- **Módulo de Bluetooth:** es el que brindará la interfaz con el dispositivo móvil Android.



Imagen 3.8 Modulo Bluetooth
Fuente: Diseño Propio.

- **Dispositivo móvil Android:** el dispositivo móvil a utilizar será de versión 4.2.2 Jelly Bean.



Imagen 3.9 Móvil Android

Fuente: Diseño Propio.

- **APP Android:** esta APP será la encargada de poder desplazar los mensajes a la Pantalla de Led, y se podrá ocupar con cualquier dispositivo Android desde la versión 2.3.6 hasta la última versión.



Imagen 3.10 APP instalada en el Móvil Android

Fuente: Diseño Propio

3.1.12 Ensamble del prototipo

Teniendo todos los elementos físicos (Arduino Uno, Pantalla Led, módulo Bluetooth, dispositivo móvil Android, APP) y lógicos (diagrama, flujo grama, firmware, código APP) el siguiente paso es construir el circuito, lo primero que se debe de hacer es identificar cada uno de los componentes, identificar sus pines, positivo, negativo y señal, y guiarse con el diagrama para conectar cada componente como es debido.

Para la colocación de cada uno de los componentes se debe colocar el Arduino Uno tomando en cuenta las dimensiones para los demás componentes, una vez teniendo listo el Arduino Uno se comenzará a unir cada componente según el diagrama electrónico, tratando de hacerlo de una forma más ordenada posible, para poder tener un prototipo más vistoso.

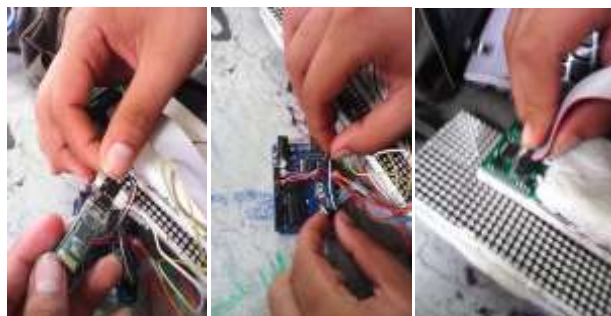


Imagen 3.11 Ensamble del prototipo

Fuente: Diseño Propio

3.1.13 Fotos del prototipo funcionando correctamente

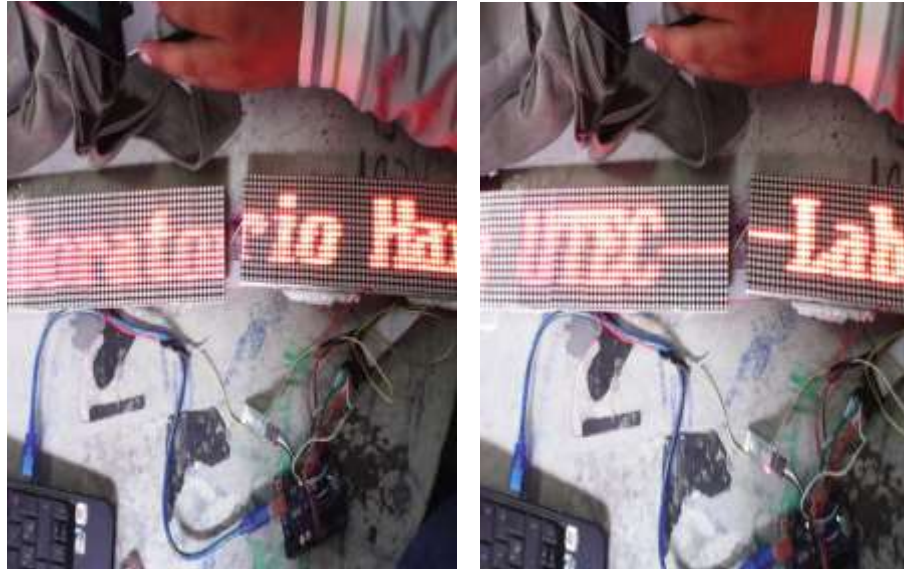


Imagen 3.12 prototipo funcionando al 100%

Fuente: Diseño Propio

3.2 Conclusión

En este trabajo de investigación se dio a conocer de forma escrita y paso a paso, la implementación de un Sistema Configurable Vía APP Android para el Despliegue de Mensajes en el Laboratorio de Hardware de la Universidad Tecnológica de El Salvador (UTEC), partiendo de conceptos y teorías.

Así mismo es de gran importancia ya que brinda gran confort al encargado del laboratorio de Hardware ya que poseerá una mejor estabilidad al momento de desplazar los mensajes ya que por medio del dispositivo móvil solo entrara a la APP del sistema y podrá configurar los mensajes ya que podrá dejarlos durante las actividades que tendrá el laboratorio.

Con lo anterior se cumplió con el diseño, construcción e implementación que se tenía como promesa en la descripción escrita de dicho proyecto, dando a conocer su principal aplicación y funcionamiento, el cual está destinado para las instalaciones del laboratorio de Hardware. Todo esto se logro luego del estudio realizado para la elección de cada componente electrónico para tener un buen funcionamiento del sistema Domótico por eso mismo se creó un prototipo funcional que demuestra nuestro trabajo y nuestro profesionalismo cumpliendo con las expectativas de solución.

Por lo tanto fue un placer contribuir con conocimientos adquiridos en nuestra carrera de Técnicos en Ingeniería de Hardware, ya que gracias a ello el laboratorio de Hardware cuenta con un sistema domótico de Despliegue de Mensaje y controladas por un dispositivo móvil Android.

3.3 Recomendaciones

Para este proyecto se tomaron en cuenta las siguientes recomendaciones para el uso y manejo y evitar cualquier percance que pueda dañar el circuito electrónico.

- Instalar correctamente la APP al dispositivo móvil Android.
- Mantener los dispositivos protegidos en un chasis.
- Mantener la APP de uso privado y no público.
- Saber la regulación adecuada del voltaje del proyecto.
- No hacer ninguna conexión sobre el cableado del dispositivo.

3.4 Referencias

- Arduino. (7 de abril de 2014). *Arduino*. Obtenido de Arduino:
<http://www.arduino.cc/en/pmwiki.php?n=>
- Cristóbal, R., Fracisco, V., & Carlos, d. C. (2011). Domótica e inmótica. En R. Cristóbal, V. Fracisco, & d. C. Carlos, *Domótica e inmótica*. (pág. 23). Madrid, España: RA-MA Editorial.
- Fernandez, J., & Labarta, F. (2009). *Como crear una marca, Manual de uso y gestion* . Madrid: Almozara.
- Fernandez, P. (2013). *Gestion del marketing de servicio*. madrid: Granica.
- Flores, O. (2013). *Aprende Arduino*. San Salvador: Ingenieria nacional .
- Gironés, J. T. (2013). *El gran libro de Android* . mexico: Alfaomega.
- Gironés, J. T. (2013). *El Gran Libro de Android* . Mexico: Alfaomega.
- Girones, J. T. (2013). *El gran libro de Android*. mexico: Alfaomega.
- Gironés, J. T. (2013). *El gran libro de Android*. mexico: Alfaomega.
- Millares., M. J. (31 de marzo de 2005). *Panorámica de los sistemas domóticos e inmóticos* . Obtenido de Panorámica de los sistemas domóticos e inmóticos :
<http://bibing.us.es/proyectos/abreproy/11116/fichero/Volumen1%252F>
- Otoniel, F. (2011). *Aprende Arduino*. San Salvador: TM.
- Play, G. (24 de septiembre de 2014). *Google Play*. Obtenido de Google play:
https://support.google.com/googleplay/android-developer/answer/113475?hl=es&ref_topic=3450987
- Romero, C., Francisco, v., & castro, C. (2011). *Domótica e inmótica*. mexico: Alfaomega.
- Romero, C., Vazquez, F., & Castro, c. (2011). *Domótica e inmótica*. mexico: Alfaomega.
- Romero, C., Vazquez, F., & de Castro, C. (2011). *Domótica e Inmótica*. Mexico: Alfaomega.
- Wolber, D. (12 de julio de 2010). *Herramientas de desarrollo de software*. Obtenido de Herramientas de desarrollo de software: http://centrodeartigos.com/articulos-informativos/article_69772.html
- Yerika, E. A. (23 de Mayo de 2012). *Tecnológica Android*. Obtenido de Tecnológica Android: <http://tecnologiasandroid.blogspot.com/search?updated-min=2012-01-01t00:00:00-08:00&updated-max=2013-01-01t00:00:00-08:00&max-results=4>

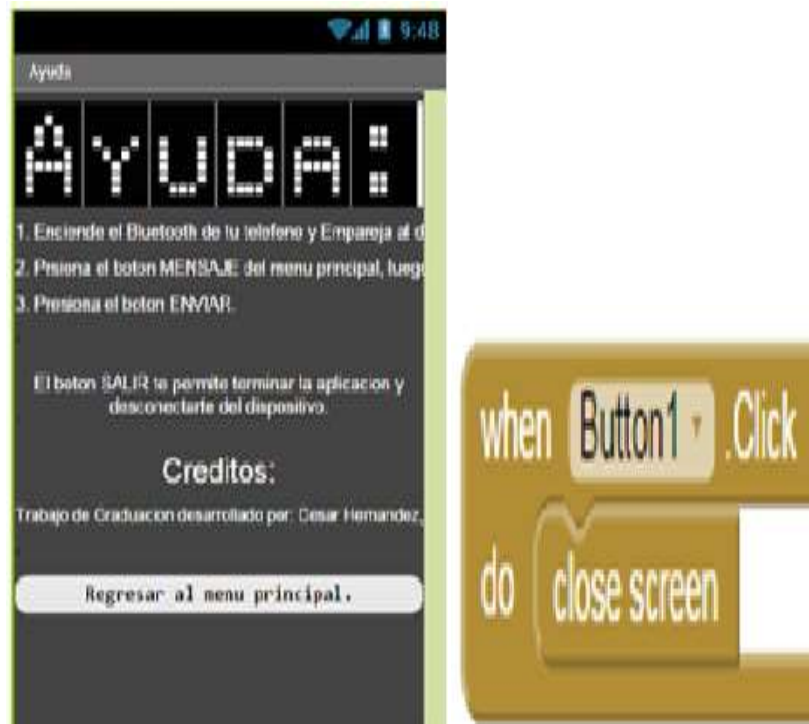
3.5 Anexos



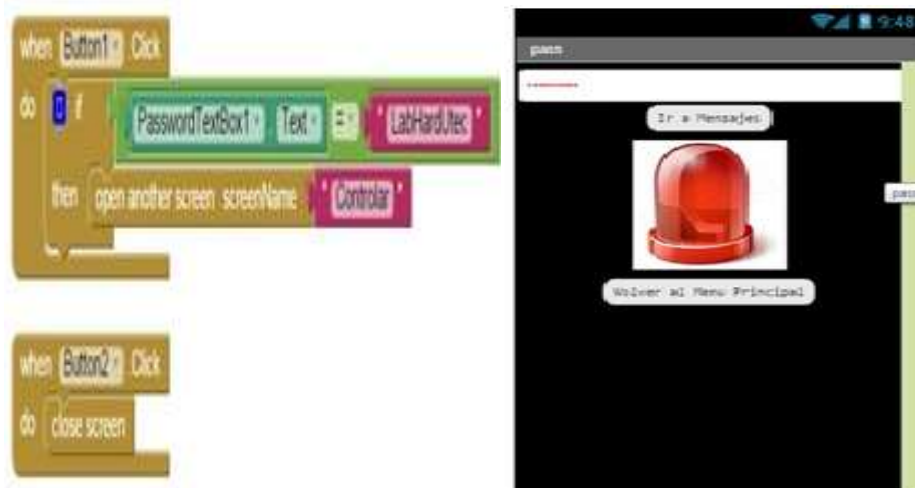
Dispositivo Móvil Android



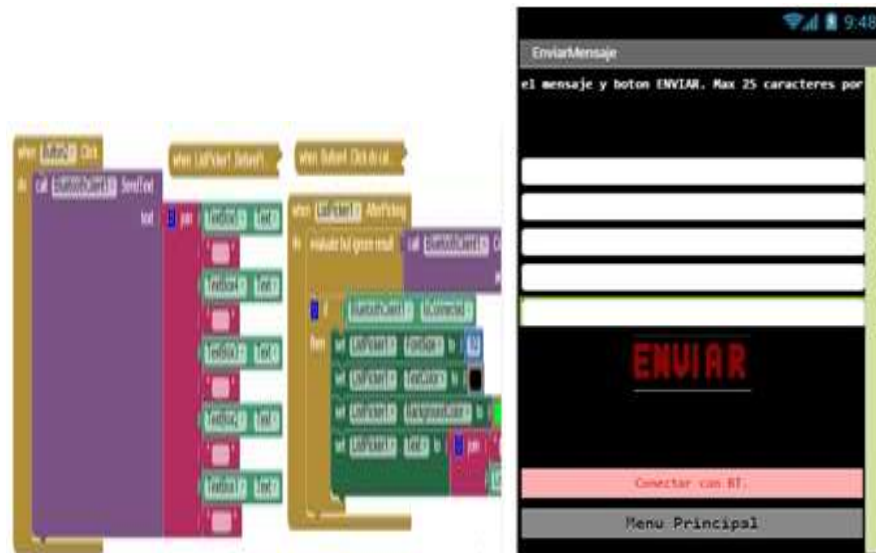
Pantalla de inicio y código fuente de la APP



Pantalla de ayuda y código de fuente APP



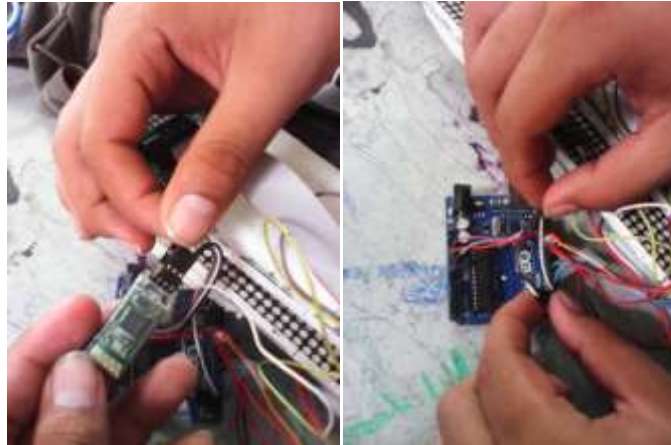
Pantalla de ingreso de contraseña y código de la APP



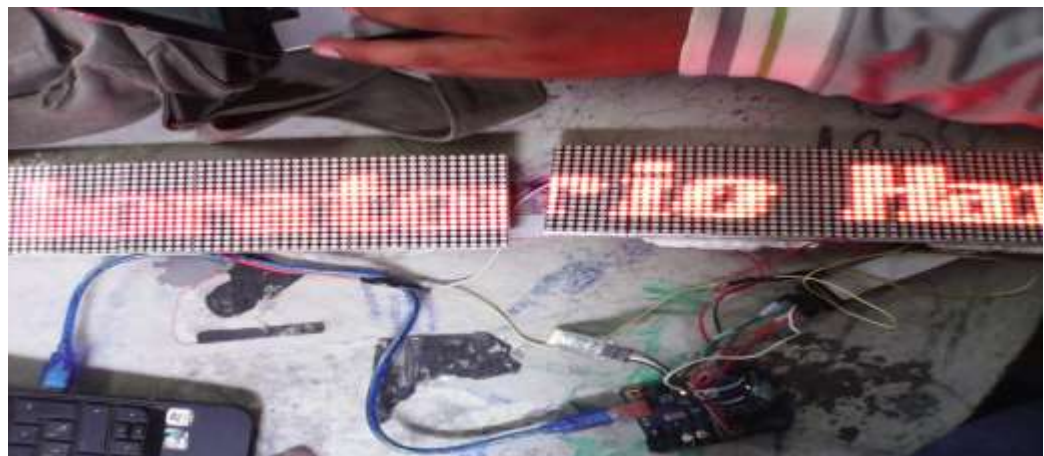
Pantalla de envío de mensajes y código de la APP



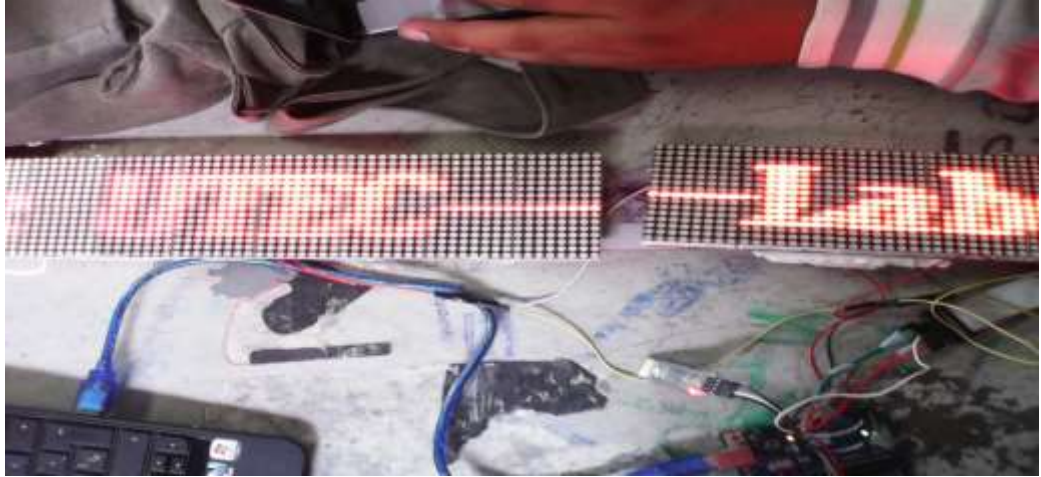
APP instalada en el Móvil Android



Ensamble del prototipo



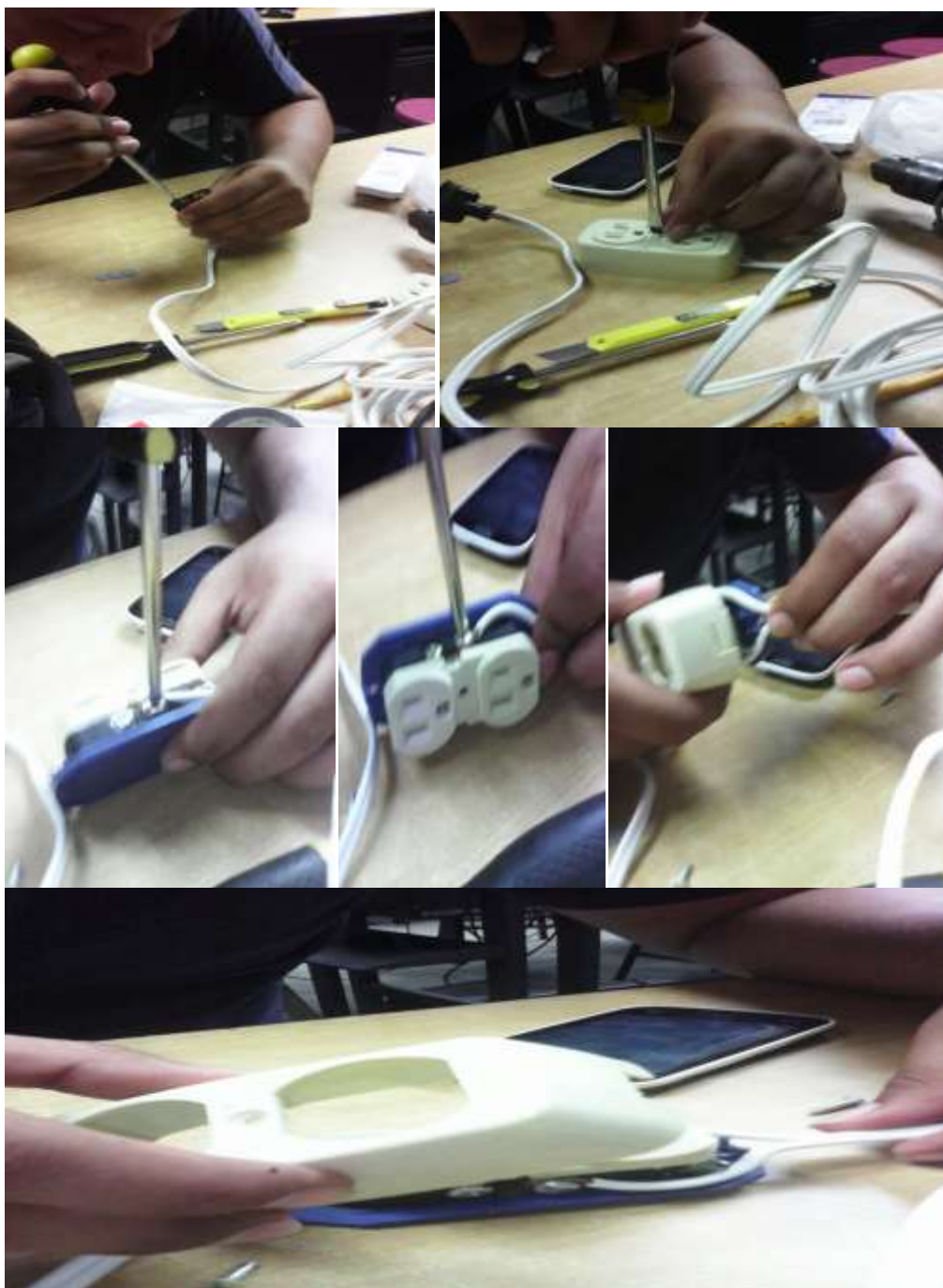
Prototipo funcionando al 100%



Prototipo funcionando al 100%



Ubicación del proyecto



Instalación Eléctrica del Prototipo