



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

OPTIMIZACIÓN DE FRECUENCIAS EN SALIDAS HACIA LAS
RUTAS QUE REALIZAN LOS BUSES PERTENECIENTES A
TRANSESPOL E.P.

INFORME DE PROYECTO INTEGRADOR

Previo la obtención del Título de:

LICENCIADO EN REDES Y SISTEMAS OPERATIVOS

Presentado por:

ACOSTA ORTIZ WILMER HUGO

RIVERA VERA PEDRO ABEL

GUAYAQUIL - ECUADOR

AÑO: 2018

DEDICATORIA

Dedicado: a mis padres: María y Pedro, por el sacrificio y el esfuerzo, por darme sueños y aspiraciones, por creer en mi capacidad, a ellos que siempre han estado brindándome cariño, comprensión y sobre todo mucho amor.

A Pablo, por demostrarme que la confianza es una fuente invaluable que fortaleza los sueños y la esperanza, tu mi guardián, mi amigo, mi hermano, con el que se puede pasar muchas horas conversando sobre ideas que mejoren nuestra calidad de vida, en ellas encontré mi fuente motivacional.

A las hermanas, voluntarios, animadores, colaboradores y catequistas de la comunidad Salesiana, quienes sembraron la Fe en mí, y me enseñaron a ver, amar y confiar a Jesús en cada acción, en ellos descubrí que una vida de sacrificios es más placentera que una llena de lujos, a dar gracias en cada momento por la vida que se nos ha concedido, porque con ustedes aprendí a dar la vida por los demás como Él lo hizo.

A mis mentores Ignacio y Patricia, quienes inculcaron en mi la pasión por aprender, de no desmayar a pesar de las adversidades, a darme mi lugar como profesional a forjarme como un líder, gracias por creer en mi

Y sin dejar atrás también va para Anggie quién me da su luz cual faro a un barco que naufragó y busca refugio, por ser el arcoíris después de la tempestad, la que me enseña que la paciencia es una virtud que se recompensa con el tesoro que se esconde detrás de una sonrisa

Gracias a todos.

Pedro Abel.

AGRADECIMIENTOS

Dios, tu amor y tu bondad no tienen fin, me permites sonreír ante todos mis logros que son resultado de tu ayuda incondicional, por las pruebas y por el aprendizaje obtenido me doy cuenta de que buscas mi superación como humano.

Este proyecto lleva consigo un gran sacrificio y por ende en una gran bendición en todos los sentidos, y no cesas mis ganas de decir que todo es gracias a ti.

La vida se encuentra plagada de travesías, y sin duda una de ellas es la universidad. Enrolado en ella, me he dado cuenta de que más que un reto, es el alma máter de la sociedad conteniendo en ella vida y futuro.

Le agradezco a la institución y a cada uno de mis profesores por los esfuerzos invertidos en mí, y lograr así poder graduarme, como un profesional que colaborara a beneficio de la sociedad.

Pedro Abel.

DECLARACIÓN EXPRESA

“Los derechos de titularidad y explotación, nos corresponde conforme al reglamento de propiedad intelectual de la institución; *Wilmer Hugo Acosta Ortiz, Pedro Abel Rivera Vera* y damos nuestro consentimiento para que la ESPOL realice la comunicación pública de la obra por cualquier medio con el fin de promover la consulta, difusión y uso público de la producción intelectual”

Wilmer Hugo Acosta Ortiz

Pedro Abel Rivera Vera

TRIBUNAL DE EVALUACIÓN

Ing. Robert Andrade

PROFESOR DE LA MATERIA

Ing. Adriana Collaguazo

PROFESOR TUTOR

RESUMEN

Este documento presenta el estudio realizado por parte de los estudiantes Wilmer Acosta y Pedro Rivera, como parte de su proceso de titulación, que haciendo uso de la metodología de Design Thinking lograron encontrar entre las problemáticas que tiene la Empresa Pública TransESPOL, misma que es se encarga del servicio de transporte para estudiantes y docentes de la ESPOL, la posibilidad de implementar una solución tecnológica que les permita estimar la cantidad de personas que requieren su servicio en “tiempo real”, además de poder guardar información con la finalidad de tener un histórico de estos registros y poder hacer análisis estadísticos con la finalidad de obtener datos que permitan la optimización de los recursos, que le faculta a la empresa realizar estimaciones de uso, mejorar la logística de sus rutas y optimizar el uso de sus unidades.

Nuestra solución consiste en un dispositivo que contabiliza la cantidad de personas por medio del uso que estas hagan de teléfonos celulares, cabe recalcar que la solución que se propone tiene un algoritmo capaz de realizar un descarte de los dispositivos que se detecten y que no sean un teléfono celular, es decir, otros dispositivos electrónicos con conexión Wifi como lo son: tabletas, televisores, puntos de acceso, entre otros.

Palabras Clave: IoT, NodeMCU, Transporte público, TransESPOL. ESPOL

ABSTRACT

This document presents the study carried out by the students Wilmer Acosta and Pedro Rivera, as part of their titling process, which using the Design Thinking methodology managed to find among the problems that the Public Company TransESPOL has, which is In charge of the transport service for students and teachers of the ESPOL, the possibility of implementing a technological solution that allows them to estimate the number of people that require their service in real time, besides being able to save information with the purpose of having a history of these records and being able to make statistical analyzes in order to obtain data that allow the optimization of resources, which empowers the company to make use estimates, improve the logistics of its routes and optimize the use of its units.

Our solution is a device that counts the number of people through the use they make of cell phones, it should be noted that the solution we propose has an algorithm capable of discarding the devices that are detected and that are not a telephone cellular, that is, other electronic devices with Wi-Fi connection such as: tablets, televisions, access points, and others.

Keywords: IoT, NodeMCU, Public transport, TransESPOL, ESPOL

ÍNDICE GENERAL

RESUMEN.....	I
<i>ABSTRACT</i>	II
ÍNDICE GENERAL	III
ABREVIATURAS	V
SIMBOLOGÍA	VI
ÍNDICE DE FIGURAS.....	VII
ÍNDICE DE TABLAS.....	VIII
CAPÍTULO 1	9
1. INTRODUCCIÓN	9
1.1 Descripción del problema	10
1.2 Justificación del problema.....	10
1.3 Objetivos.....	12
CAPÍTULO 2.....	13
2. METODOLOGÍA	13
2.1 Fase 1 de Empatizar.....	13
2.2 Fase 2 de Definir	15
2.3 Fase 3 de Idear.....	15
2.4 Fase 4 de Prototipar	17
2.5 Fase 5 de Evaluar.....	18
CAPÍTULO 3.....	19
3. DISEÑO DE LA SOLUCIÓN	19
3.1 Diagrama de red	19
3.2 Diseño del contadorMCU.....	23
3.3 Comunicación con los Servidores.....	25

3.4	Servidores y Aplicativos.....	30
3.5	Pruebas con el ContadorMCU – Versión 1	36
	CAPÍTULO 4	47
4.	PLAN DE IMPLEMENTACIÓN Y COSTOS	47
4.1	Plan de Implementación	47
4.2	Costos.....	48
	CONCLUSIONES Y RECOMENDACIONES	50
	BIBLIOGRAFÍA.....	52
	ANEXOS.....	53

ABREVIATURAS

ACL	Access Control List
AP	Access Point
ESPOL	Escuela Superior Politécnica del Litoral
IDE	Integrated Development Environment
IEEE	Institute of Electrical and Electronics Engineers
MQTT	Message Queue Telemetry Transport
PLA	Ácido Poli-Láctico
RAM	Random Access Memory
RSSI	Received Signal Strength Indicator
RTC	Real Time Clock
TransESPOL E.P.	TransESPOL Empresa Publica
TV	Televisión
USB	Universal Serial Bus

SIMBOLOGÍA

bps	bit por segundos
GB	Gigabyte
GHz	Gigahercios
Km	Kilómetros
MB	Megabytes
mil	Milésima de pulgada
MHz	Megahercios
V	Voltios

ÍNDICE DE FIGURAS

Figura 2.1 – Mapa de empatía del Ing. Alejandro Chanaba.....	14
Figura 2.2 Prototipo de Solución Planteada al Cliente	17
Figura 3.1 – Diagrama de Red de la Solución.	20
Figura 3.2 – Imagen de un NodeMCU.	22
Figura 3.3 - Esquema Electrónico de Dispositivo.....	23
Figura 3.4 – Registros almacenados en la microSD.	24
Figura 3.5 – Diseño de caja protectora que contendrá el contadorMCU.	25
Figura 3.6 – Representación gráfica de la comunicación del protocolo MQTT.....	26
Figura 3.7 – Diseño de la interfaz para Pagina Web.....	31
Figura 3.8 – Diseño de la interfaz para Aplicación Móvil.	32
Figura 3.9 – Diseño de la Interfaz de Administración.....	34
Figura 3.10 – Diagrama de Despliegue.	35
Figura 3.11 – Ensamblaje del dispositivo.....	36
Figura 3.12 – Instalación del prototipo de pruebas.	38
Figura 3.13 – Inicio de sesión de la aplicación de pruebas.....	39
Figura 3.14 – Interfaz principal de la aplicación de prueba.....	40
Figura 3.15 – Interfaz principal y la barra de tareas del aplicativo de prueba.	42
Figura 3.16 – Interfaz de aplicación móvil.....	44
Figura 3.17 – Prueba del dispositivo y toma de datos.	46

ÍNDICE DE TABLAS

Tabla 2.1 – Matriz de decisión.	16
Tabla 3.1 – Detalles de las suscripciones de los nodos en los	27
Tabla 3.2 – Canales de comunicación que se usan en la solución.....	29
Tabla 4.1 - Plan de Implementación.	47
Tabla 4.2 – Detalle de Costo de componentes electrónicos del dispositivo.....	48
Tabla 4.3 - Costos de Implementación de la Solución.	49

CAPÍTULO 1

INTRODUCCIÓN

TransESPOL Empresa Pública (TransESPOL E.P.) dedicada a la prestación de servicio de transportación urbana de estudiantes, profesores, empleados y funcionarios de la Escuela Superior Politécnica del Litoral (ESPOL) buscando mejorar constantemente sus procesos, se ha propuesto optimizar la frecuencia de salida de sus buses pensando en la administración de sus recursos (Buses y conductores), abaratando costos operativos manteniendo una buena calidad en el servicio.

Según el cuerpo directivo de TransESPOL, no puede mejorarse un proceso sin datos que nos ayuden a validar un antes y después del mismo, se necesita información confiable y mucho mejor si esta se obtiene en “tiempo real”.

El proyecto consiste en proporcionar datos válidos y necesarios en “tiempo real” sobre la cantidad de estudiantes que se encuentran en las paradas de buses, así como también una estimación del tiempo total que estos últimos tuvieron que esperar el bus, esta información permitirá optimizar el uso de sus recursos y medir realmente la eficiencia del servicio de TransESPOL.

Las tecnologías de la información nos permiten usar múltiples alternativas para la captación de datos de acuerdo con la necesidad que se tenga, en este caso se hace uso de una herramienta en redes de datos conocida como Sniffer, la cual permite detectar tráfico de paquetes de son enviados por algún medio de comunicación.

En este proyecto se presenta un uso práctico para los programas Sniffer, capturando las tramas de los paquetes transmitidos, con las cuales se puede detectar la cantidad de dispositivos que se encuentran en un determinado lugar, como parte de las pruebas se implementará un dispositivo en la terminal de buses con el fin de almacenar datos que sirvan para un posterior análisis.

1.1 Descripción del problema

TransESPOL E.P. es la encargada de transportar, dentro y fuera de la Universidad, a los estudiantes y docentes que requieran el servicio, el problema es que solo cuentan con 25 buses para todos sus usuarios, de estos últimos 10 mil son estudiantes. El director de TransESPOL, Ing. Alejandro Chanaba, dice: “Es imposible que nosotros podamos servir a todo el mundo y peor aún que todo el mundo quede contento”, asegura que la empresa no puede ser la solución de transporte de los estudiantes. Además, las malas decisiones administrativas llevaron a TransESPOL a la quiebra, razón por la cual la empresa no cuenta con el dinero suficiente para adquirir más.

Otro problema de la empresa es que no cuenta con información histórica sobre el uso de su servicio razón por la cual no poseen datos estadísticos como: la cantidad de personas en las filas del Terminal de buses de ESPOL, días de mayor afluencia y horas pico, Datos necesarios para la toma de decisiones respecto a temas como las frecuencias de salida de los buses hacia los diferentes destinos.

1.2 Justificación del problema

Desde el año 2016 servicio de transporte público del cual dispone la ESPOL ha sufrido de inconformidad constante por parte de los estudiantes según lo demuestra la encuesta que se realizó a la población estudiantil tomando una muestra de 100 estudiantes, donde el 60% de estos mantiene que no ha tenido un buen servicio con respecto a la toma de decisiones que ha realizado la empresa, mismos que lo usan como método de entrada y salida a la ESPOL, es posible que esta eventualidad se deba a un incremento en la cantidad de estudiantes que anualmente ingresan a la universidad.

Según los datos estadísticos de la ESPOL en el 2016 fueron admitidos un total de 2107 estudiantes, mientras que el 2017 este número aumento a 2138, lo que representa un incremento del 1,47% [1].

Actualmente para conocer el número de estudiantes que usan el servicio de transporte, TransESPOL E.P. maneja registros manuales en fichas llenadas por los controladores, personas encargadas del envío de los buses a las diferentes rutas en la terminal de buses, agregando la información que les proveen los conductores al completar un circuito, según el Ing. Alejandro Chanaba, al manejar la información de esta manera se ve sujeta a tergiversaciones, lo cual resta validez a estos datos y crea en la empresa la necesidad de obtener una mejor manera para registrar la cantidad de estudiantes que usan el servicio, además de almacenar esta información digitalmente de manera histórica, lo que le permitiría realizar estimaciones de uso.

Por otro lado, en TransESPOL se han concretado estudios durante los últimos seis meses, tomando en consideración parámetros como los sectores de la ciudad en donde viven la mayor cantidad de estudiantes y la cantidad de buses necesarios para cubrir esta demanda. Los resultados de este estudio les permitió tomar la decisión de cambiar las rutas de los buses, reduciendo la cantidad de kilómetros recorridos en cada una, para así poder asegurar disponibilidad de buses, evitando en gran medida los retrasos de las unidades.

Sobre las mejoras implementadas por TransESPOL, se realizaron 10 entrevistas a estudiantes y docentes, usuarios del servicio, pero ellos mostraron insatisfacción con cambios del servicio. Sostienen el 80% de los estudiantes encuestados que “no fueron pensadas para ellos sino en el factor económico de la empresa, puesto que ahora toman entre dos o tres buses más para poder llegar a la universidad, y las rutas se han acortado”.

El presente proyecto plantea una solución a las necesidades que tiene la empresa para determinar la cantidad de estudiantes que se encuentran en la terminal de buses y las diferentes paradas, de esa manera poder mejorar la toma de decisiones por parte del administrador para optimizar las diferentes rutas (Controlador).

1.3 Objetivos

1.3.1 Objetivo General

Mejorar el control logístico sobre las frecuencias de salidas hacia las rutas que realizan los buses de ESPOL durante la salida del personal, mediante la estimación de la cantidad de usuarios que se encuentran en el terminal de buses TransESPOL E.P.

1.3.2 Objetivos Específicos

1. Diseñar una solución que requiera poca inversión monetaria que permita la estimación de usuarios en el terminal de buses.
2. Diseñar e implementar un dispositivo que estime la cantidad de personas que se encuentren en un lugar específico para optimizar el uso de los buses informando al controlador cuantos estudiantes se encuentran en la terminal.
3. Almacenar información histórica del uso del servicio de transporte en un servidor de base de datos.
4. Implementar una aplicación que active el dispositivo y muestre la información recolectada por el dispositivo de estimación.

CAPÍTULO 2

METODOLOGÍA

Para el desarrollo del proyecto se utilizó la metodología Design Thinking con el propósito de desarrollar una idea que nos permita mejorar los servicios de TransESPOL, haciendo uso de las Tecnologías de la Información (TI).

A continuación, describimos algunas de las herramientas utilizadas en las distintas fases de la metodología aplicada, además de los resultados obtenidos.

2.1 Fase 1 de Empatizar

En la primera fase de la metodología, se busca empatizar con nuestro cliente, TransESPOL E.P, para lo cual primero usamos la herramienta “mapa de actores” para establecer todas las personas que figuran en la organización, este se encuentra en el Anexo A. Con los actores establecidos, se procedió a utilizar la herramienta “Los Porque” el cual contiene las preguntas ordenadas, provenientes de una lluvia de ideas, que se usaron en las entrevistas, esta se encuentra en el Anexo B.

El director de TransESPOL el Ing. Alejandro Chanaba, nos explicó los cambios realizados en la misma y también los problemas que se les han venido presentando. En la figura 2.1 se detalla el cuadro de empatía, de manera resumida, perteneciente al Ing. Alejandro Chanaba, gran parte de la información que él nos proporciono sirvió para poder determinar que problemas está esperando resolver, en el Anexo C de este documento se encuentra el mapa de empatía de manera completa.

Después de la reunión con el Ing. Chanaba, se realizó una encuesta con 16 usuarios frecuentes del servicio de la compañía: 11 estudiantes y el resto fue público en general incluyendo profesores, también se hizo una entrevista con el controlador de buses, los modelos de preguntas de las encuestas están basadas en los cuadros de análisis de entorno que se trabajó dentro de la metodología, podemos encontrarlos en el Anexo B de este documento.

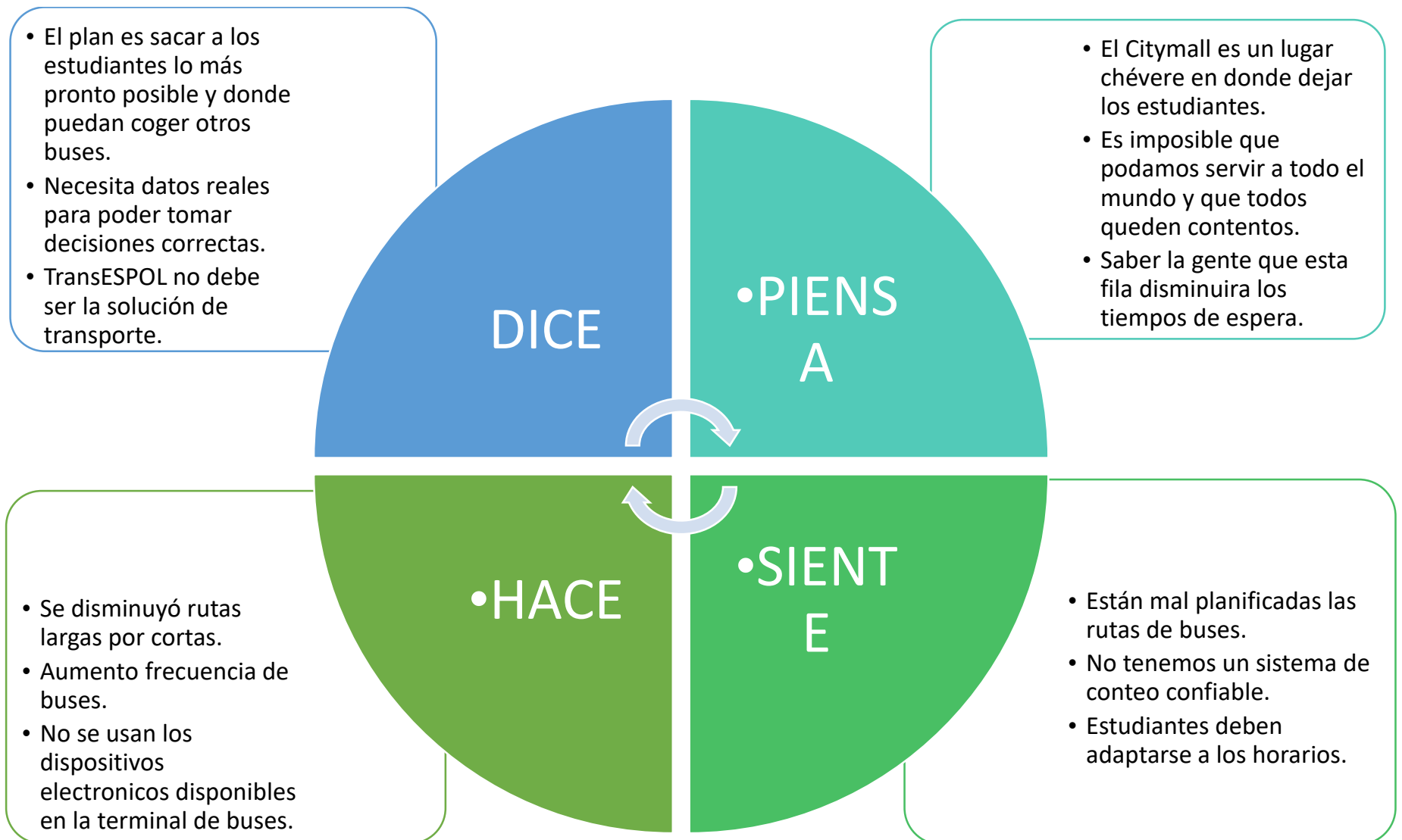


Figura 2.1 – Mapa de empatía del Ing. Alejandro Chanaba.

2.2 Fase 2 de Definir

Para la segunda fase definimos los problemas de TransESPOL haciendo uso de las entrevistas realizadas a los actores de la organización y usando la herramienta “Espina de Pescado”, puede encontrarlo en el Anexo D, logramos encontrar varios problemas en la empresa:

Para los estudiantes:

- El tiempo de espera de los buses, al momento de salir a las diferentes rutas, es demasiado alto

Para el personal de TransESPOL:

- Cuentan con solo 25 buses para la totalidad de la población ESPOL (Estudiantes, docentes, personal Administrativos).
- No se cuenta con registros históricos del uso de los servicios de la empresa, quedando sin herramientas para realizar análisis estadísticos y de logística.

En el capítulo 1 en la sección 1.1 de Descripción del Problema, vimos más a detalle los diferentes problemas de TransESPOL, y en la sección de 1.2 presentamos la Justificación del problema, con las consecuencias que provocan los problemas encontrados.

2.3 Fase 3 de Idear

Una vez definido el problema, se procedió con la tercera fase, donde propusimos varias posibles soluciones al problema. A continuación, se enumeran las ideas que más sobresalientes que fueron presentadas en la segunda reunión con los directivos de la empresa:

1. Desarrollo de una aplicación que encueste la tentativa de uso del servicio.
2. Hacer uso de visión por computadora mediante cámaras para contabilizar los estudiantes que se encuentran esperando buses.
3. Usar un equipo que permita contabilizar la cantidad de estudiantes en las paradas.

Tabla 2.1 – Matriz de decisión.

Alternativas	Aplicación para que TransESPOL pueda reportar a los estudiantes la ubicación de los buses	Sistema que encueste a los estudiantes para estimar las horas en las que se requieren más buses	Sistema que permita saber cuándo es necesario enviar buses, contando los estudiantes en los paraderos	Eliminar todas las rutas de buses y destinar todos los buses a una sola ruta estratégicamente establecida	Que se compren más buses para ESPOL
Necesidades					
Viable Económicamente	1	1	1	1	
Optimizar el uso del recurso buses		1	1	1	
Aumentar la frecuencia de salida de buses		1	1	1	1
Permitir guardar datos históricos del uso del servicio			1		
Ayudar a la toma de decisiones al personal administrativo	1	1	1		1
Mejorar la Imagen de la empresa	1	1	1		
Reducir el tiempo de espera de buses para los estudiantes	1	1	1	1	1
Asegurar la disponibilidad de buses cuando son necesarios			1		
TOTALES	4	6	8	4	3

En la tabla 2.1 se muestra la matriz de decisión que se realizó tratando de poder abarcar con la solución que se propuso la mayor cantidad de problemas detectados, de tal manera que se pueda desarrollar una y se obtenga el máximo impacto posible para todas las partes interesadas.

Después de la reunión con el directivo de TransESPOL E.P., se estableció que el proyecto consistiría en implementar un dispositivo que desde ahora se denominará contadorMCU y tenga la característica principal de estimar la cantidad

de estudiantes que se encuentran en las diferentes paradas y terminal de buses dentro del campus Gustavo Galindo.

2.4 Fase 4 de Prototipar

En la figura 2.2 donde presentamos la solución planteada al Ing. Alejandro Chanaba, en esta podemos ver representadas 4 paradas de buses donde vemos unos dispositivos con antenas Wifi que se encargarían de contar a los estudiantes que se encuentren en el lugar, para luego enviar esta información haciendo uso de la red ESPOL a un arreglo de servidores donde se implementara:

1. Un servidor de base de datos, el cual almacenara la información recolectada por los dispositivos.
2. Un servidor de aplicación que recepte toda la información enviada por los dispositivos y los envíe al servidor mosquito y el de base de datos.
3. Un servidor mosquito, para poder presentar la información recibida por el servidor de aplicación en tiempo real a los teléfonos de los controladores o algún computador, usando un punto de acceso para la comunicación.

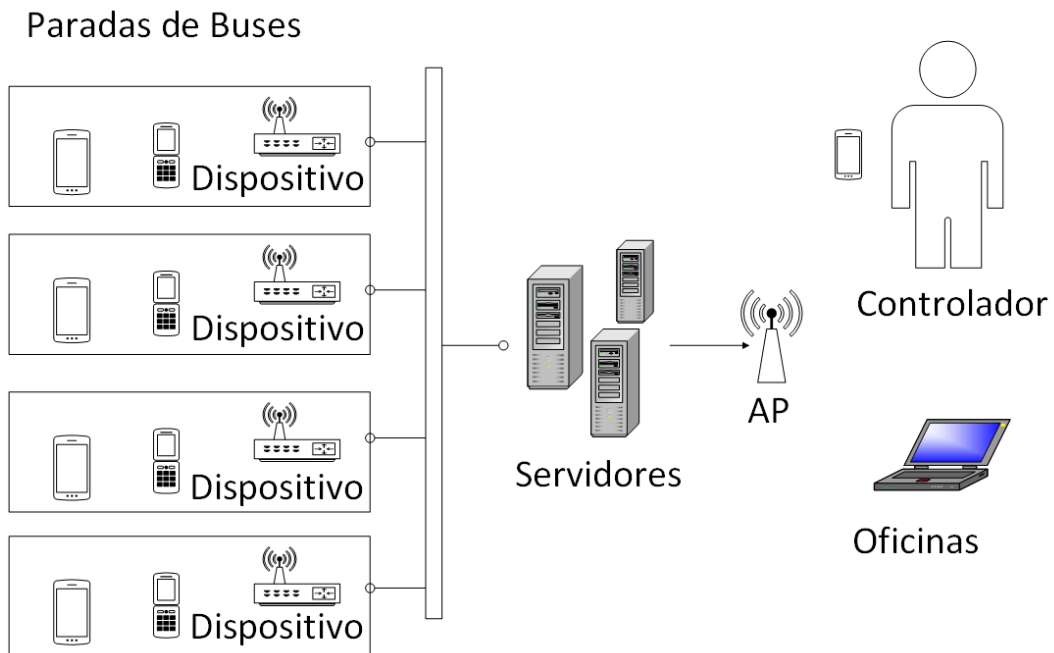


Figura 2.2 – Prototipo de Solución Planteada al Cliente

En el capítulo 3 se encuentra más detallada la información del diseño de la solución y las pruebas que se realizaron con nuestro prototipo.

2.5 Fase 5 de Evaluar

Una vez presentado el prototipo se nos solicitó que se tome en consideración algunos factores que podrían afectar la contabilización de personas en la parada, por ejemplo: puntos de acceso, impresoras inalámbricas, televisores y otros dispositivos electrónicos que tienen la capacidad de conectarse a internet por medio de wifi, puestos que también serán detectados. Además, puede darse el caso en que algunas personas suelen llevar más de un dispositivo como un segundo celular o usan tabletas, laptops u otros medios electrónicos con capacidad de conectarse a la red vía wifi.

Para solucionar los problemas planteados en el párrafo anterior se tuvo que agregar una lista blanca de direcciones MAC conocidas donde se agregarían los teléfonos del personal de TransESPOL y los equipos instalados en las paradas (Televisores, Puntos de Acceso, etc.). También agregar un filtrado de las direcciones MAC que nos permita diferenciar celulares de tabletas, laptops u otros dispositivos que suelen llevar consigo los estudiante, para eso se usará el listado de MAC asignadas por fabricante de la IEEE. [2]

CAPÍTULO 3

DISEÑO DE LA SOLUCIÓN

Usando como premisa: “cada persona posea al menos un teléfono celular”, se propone la implementación de un dispositivo, al que denominamos contadorMCU, con tecnología wifi 802.11b/g/n, capaz de captar los paquetes IP permitiéndole enumerar la cantidad de teléfonos en el lugar, esto es posible gracias a las direcciones físicas denominadas MAC's, que se envían dentro de la trama de los paquetes IP, durante la búsqueda que realiza dispositivo, con wifi, de un nodo en la red que le permita enlazarse a la red de datos.

Los ContadoresMCU que se implementarán en cada una de las paradas podrán transmitir en “tiempo real”, la cantidad de celulares que detecte y transmitirla por medio de la red de ESPOL a un arreglo de servidores que se encargarán de almacenar y presentar la información obtenida por los dispositivos, además de administrar los dispositivos. A continuación, describiremos los diferentes componentes necesarios para la implementación completa de nuestra solución.

3.1 Diagrama de red

En la Figura 3.1 el diagrama de red con los componentes necesarios para la implementación de la solución, cada uno de ellos identificados con la siguiente numeración:

1. Servidores: de Base de datos, Mosquito y el Receptor.
2. Computadora que contiene la interfaz de administración de los dispositivos.
3. Conmutador para conectar la solución a la infraestructura de ESPOL.
4. Una nube en representación de la Red ESPOL.
5. Punto de Acceso de TransESPOL al cual estará conectado el ContadorMCU, de manera inalámbrica.
6. ContadorMCU, encargado de generar y enviar los registros al receptor.
7. Punto de Acceso ESPOL.
8. Dispositivos Inalámbricos de personal de TransESPOL y Estudiantes

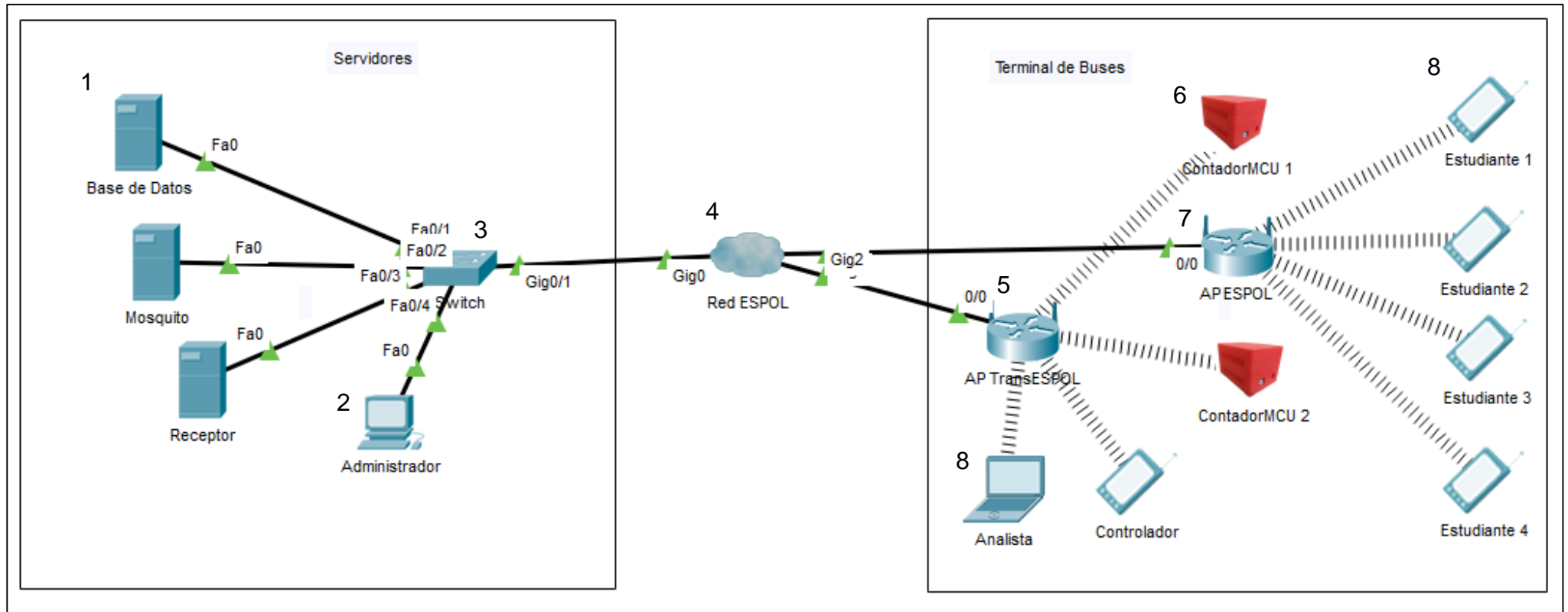


Figura 3.1 – Diagrama de Red de la Solución.

El punto de acceso perteneciente a la red interna de TransESPOL, el cual se encuentra en la terminal de Buses, es de uso exclusivo del personal de la empresa.

Como una recomendación para incitar a que los estudiantes usen la red de ESPOL estando en la terminal de buses, se planteó la idea a los directivos de TransESPOL E.P. la cual consiste en crear una aplicación móvil que permita la visualización de los horarios de salida de los buses y un temporizador mostrando el tiempo restante para la salida del siguiente, de esta manera podemos reducir el error producido durante la contabilización de dispositivos.

Por el tamaño de la terminal de buses se implementan por ahora solo dos contadorMCU para tratar de capturar la mayor cantidad de teléfonos celulares que están conectados a la red, cabe recalcar que para aumentar la precisión de la solución se requiere agregar otros.

En la actualidad las personas están acostumbradas a tener acceso a internet en cualquier lugar por medio de una red inalámbrica o bien sea a la red celular de operador telefónico, pero un usuario convencional (persona con pocos conocimientos en el funcionamiento de las redes de datos) es transparente la manera en la que sus dispositivos logran acceder a internet, en este caso refiriéndonos a los procesos que realizan los teléfonos móviles buscando conectarse a las redes wifi-disponibles.

Para comprender el funcionamiento del contadorMCU que se presenta en este proyecto se explicará brevemente el proceso de la conexión que realizan los teléfonos móviles a las redes wifi.

Cuando los teléfonos móviles son configurados en una red wifi, guardan la información de conexión de esta red en las bases de datos internas del sistema operativo, mismas que son consultadas concurrenciamente cuando se pierde la comunicación buscando una nueva que esté disponible o a su vez consultando por medio inalámbrico una que le permita acceder a internet.

Esto implica que cuando se conecta en otro lugar a la red wifi, el teléfono automáticamente intenta establecer comunicación con un nodo de la red que le conceda el acceso, para poder lograrlo emite mensajes de sincronización con los puntos de accesos (AP, por sus siglas en inglés) lo que crea un hueco de seguridad, el cual aprovecha el contadorMCU.

El contadorMCU cuentan con dos NodeMCU, presentado en la figura 3.2, microcontroladores que usan procesadores Tensilica L106 32-bit con una velocidad de 80 – 160MHz, capacidad de memoria máxima de 16MB y opera con 3.3V según sus especificaciones técnicas. Este microcontrolador utiliza un chip llamado ESP8266 que vienen con un módulo wifi integrado bajo estándar 802.11 b/g/n con una frecuencia de 2,4GHZ, el cual podemos usar en modo promiscuo para ejecutar un aplicativo Sniffer, que nos permite capturar las tramas de datos transmitidos en la red inalámbrica que se encuentre dentro del alcance del chip wifi, se copia la dirección física MAC y se descartan los demás campos de la trama. También registramos la intensidad con la que está captando la señal de los emisores (RSSI).

La placa del NodeMCU posee pines que pueden ser configurados tanto de entrada como de salida, mismos que son usados para incorporar sensores y de esta manera añadir nuevas funcionalidades.

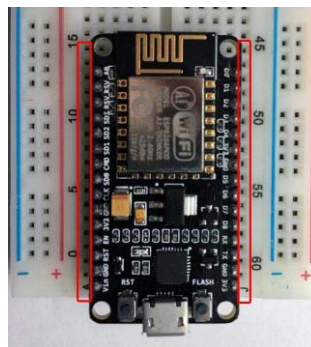


Figura 3.2 – Imagen de un NodeMCU.

En la tarjeta microSD se guarda la siguiente estructura de datos, siguiendo el mismo orden aquí descrito:

1. RSSI,
2. Dirección MAC,
3. Fecha con la hora, y
4. El lugar donde se encuentra el dispositivo.

Cuando se configura un contadorMCU se establece por medio de código la ubicación en la que estará tomando datos.

Para el almacenamiento de los datos se usa una tarjeta de 16GB (15.8 GB capacidad real de la tarjeta) dentro de esta se tiene un archivo de texto llamado "REPORTE.TXT"; cada registro dentro de este archivo está conformado con la estructura de datos que se mencionó en el párrafo anterior, un registro tiene pesa en total 50 bytes, lo que permite estimar que en esta se puede almacenar un aproximado de 374'616.591 registros.

La figura 3.4 nos muestra 4 registros que le llegarán al servidor de procesos, indicando que se han detectados direcciones MAC's con el tiempo y lugar donde fue detectada.

```
-94,44:33:4c:b8:8c:7d,16/07/2018 01:05:16,terminal  
-91,44:33:4c:b8:8c:7d,16/07/2018 01:05:31,terminal  
-98,44:33:4c:b8:8c:7d,16/07/2018 01:05:02,terminal  
-79,ec:f3:5b:0c:24:47,16/07/2018 01:05:25,fcsh
```

Figura 3.4 – Registros almacenados en la microSD.

Como se puede notar, los 3 primeros registros le pertenecen a la misma dirección física, variando solo el RSSI y el tiempo; los diferentes RSSI que se tienen depende de factores climáticos como lo es el viento, la distancia entre el ContadorMCU y el dispositivo o movimientos que ha tenido el usuario, no está dentro del alcance de este proyecto profundizar en el análisis de las causas de variabilidad, pero si es necesario aclarar que mientras menor sea el valor del RSSI más alejado está el teléfono celular del contadorMCU.

Para protección y la manipulación del contadorMCU, se diseña una caja protectora, que cumpliera con los siguientes requerimientos:

1. Que nos permita proteger las partes electrónicas.
2. Que cuente con una plataforma interna donde reposaría la placa de prototipado.
3. Que permita la ventilación a los equipos electrónicos.
4. Que permita la conexión de punto de alimentación externo.

A continuación, en el diseño que tiene la caja, cuenta con dos aberturas en la parte frontal inferior ambas especialmente para permitir la alimentación que tendrán los dispositivos y dos grupos de franjas a los lados para permitir la ventilación del dispositivo, como se puede visualizar en la figura 3.5.

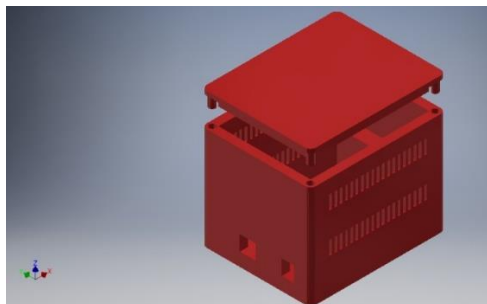


Figura 3.5 – Diseño de caja protectora que contendrá el contadorMCU.

Tomar en cuenta que para las pruebas pilotos que se realizarán en la terminal de buses, los contadoresMCU serán ubicados en la parte posterior de dos de los cinco televisores que se tiene (uno por cada TV.).

3.3 Comunicación con los Servidores

Cada contadorMCU envía los registros hacia el servidor de datos, este último almacena en memoria las direcciones MAC's, analizando cada registro que le llega actualiza la fecha y hora de la dirección, en caso de que una dirección MAC no haya sido actualizada durante los últimos 3 minutos se procede a guardar el registro en la base de datos y descartarla de la memoria del servidor.

Se implementó un servidor que llamamos Mosquito en el que se configuró el protocolo MQTT (se explica a detalle su funcionamiento en los siguientes párrafos), para poder transmitir la estimación de personas hacia el teléfono del controlador y la máquina de administración de la compañía.

La topología con la que trabaja MQTT es de tipo estrella, misma que permite poder ubicar un nodo principal denominado Broker1 y replicar la comunicación a los demás nodos que se agreguen a la red, como buenas prácticas de seguridad solo usuarios autorizados pueden solicitar esta información.

Este tipo de arquitectura tiene consigo una característica importante, nos permite la comunicación: uno a uno o a su vez de uno a muchos.

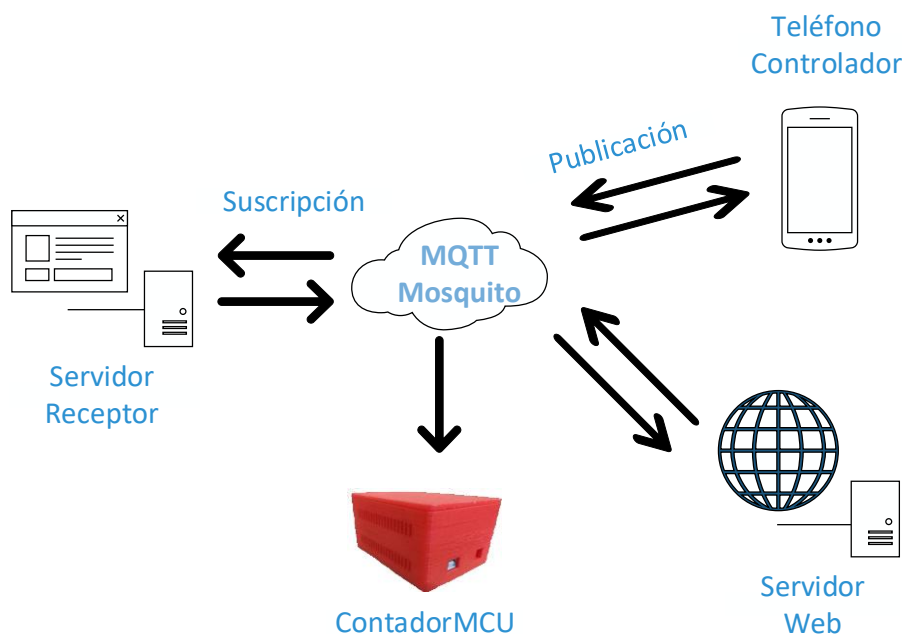


Figura 3.6 – Representación gráfica de la comunicación del protocolo MQTT.

La figura 3.6 muestra el flujo de comunicación que tiene el protocolo MQTT, las flechas con dirección hacia el servidor mosquito representan la publicación de mensajes y cuando la flecha tiene dirección hacia los demás nodos representa a

¹ El servidor Mosquito es el Broker en la topología estrella que se usa en la solución.

la suscripción donde todos los nodos afiliados al tema o canal de comunicación reciben los mensajes correspondientes.

El protocolo MQTT opera bajo el modelo publicación-suscripción, es decir, tomar en consideración que los canales de comunicación con el servidor mosquito, se establecen por medio de un puerto que se destina a este tipo de comunicación, para el proyecto el puerto que se habilito fue 14943.

Cuando un equipo de la red pública por medio de un canal de comunicación este mensaje es receptado por el servidor mosquito y reenviado a todos los equipos suscritos a este canal de comunicación, solo se requiere de dos campos para poder transmitir la información por medio de MQTT: 1) canal de comunicación y 2) el mensaje a comunicar.

Los mensajes solo serán reenviados en los canales para los que fueron destinados, es preciso mencionar que los nodos que no estuvieran suscritos al canal o a su vez que no tuviesen los permisos necesarios no podrían recibir los mensajes.

Los canales de comunicación pueden ser configurados mediante ACL, los cuales se puede establecer a un usuario para que tenga permiso de escritura o lectura sobre un canal específico.

Tabla 3.1 – Detalles de las suscripciones de los nodos en los diferentes canales de comunicación.

Nodo	Suscrito a:
Servidor Receptor	<ul style="list-style-type: none">• Administrador/admin
Aplicación móvil, y web.	<ul style="list-style-type: none">• Administrador/tenemos• Administrador/status
Dispositivo	<ul style="list-style-type: none">• Administrador/admin

En la tabla 3.1 se especifica la suscripción a los distintos canales por parte de los equipos informáticos que se usan en la solución, lo que permite que estén en constante comunicación.

Cabe recalcar que sin los permisos necesarios un equipo de la red pudiese publicar en cualquier canal de comunicación, para la implementación de esta solución se consideró los posibles escenarios y se aplicaron las políticas necesarias para prevenir que los equipos pudiesen publicar en canales en los que no están autorizados, asignando a cada dispositivo en la red un usuario y contraseña, de esta manera se limita la publicación indebida.

En la tabla 3.2 se muestra el flujo de comunicación que tiene el servidor receptor, como se puede consultar en la tabla 3.1 en la página 26, el servidor receptor solo está suscrito al canal de administración, donde recibe un mensaje que se detalla a continuación:

- Cuando recibe el número 1, algún dispositivo ya sea el teléfono del controlador o la página web está solicitando que se publique la cantidad de dispositivos que tiene activo, es decir, los que tiene en memoria. La publicación se realiza a través del canal "Administrador/tenemos" según la estructura: cantidad de dispositivos activos seguido del lugar de ubicación del contadorMCU.
- Cuando recibe el número 2, la página web está solicitando que se envíe la disponibilidad de los ContadoresMCU y la ubicación. La publicación se realiza a través del canal "Administrador/status" según la estructura: 'on' cuando el dispositivo está activo, 'off' cuando no lo está seguido del lugar de la ubicación del contadorMCU.
- Cuando recibe el número 3, la página web está solicitando que se envíe el nombre y la ubicación del contadorMCU. La publicación se realiza por el mismo canal según la estructura: nombre del dispositivo seguido de la ubicación.

Tabla 3.2 – Canales de comunicación que se usan en la solución.

CANAL DE COMUNICACIÓN	EMISOR	RECEPTOR	MENSAJE	DESCRIPCIÓN
"Administrador/admin"	Servidor Mosquito	Servidor receptor	1	Solicita que se publique la cantidad de dispositivos activos.
			2	Solicita disponibilidad de los dispositivos.
			3	Solicita La ubicación de los dispositivos
	Servidor Receptor	Servidor Mosquito	{Dispositivo, Ubicación}	Estructura de datos de los dispositivos y su ubicación.
"Administrador/tenemos"	Servidor Receptor	Servidor Mosquito	{Cantidad, ubicación}	Estructura de datos con la cantidad de dispositivos y la ubicación
"Administrador/status"	Servidor Receptor	Servidor Mosquito	{status, ubicación}	El estado del dispositivo puede ser on u off.

3.4 Servidores y Aplicativos.

A continuación, se describe cada uno de los servidores de la solución, los servicios que deberían implementarse en cada uno y los aplicativos con sus propuestas de diseño:

3.4.1 Servidores

1. Servidor Mosquito (protocolo MQTT)

Para este proyecto se puede utilizar un servidor mosquito ofrecido por la empresa Cloud MQTT se usó el plan gratis mismo que nos permite conectar hasta 10 dispositivos.

2. Servidor de Base de datos

Para la implementación del servidor de base de datos se usará MySQL en su versión 8.0 sobre un sistema operativo Linux, distribución Debian 9 [6].

El servidor puede ser hospedado en la plataforma Azure, la cual ofrece una base de datos de prueba para estudiantes totalmente gratuita siempre y cuando no se exceda los 5 MB.

3. Servidor Web

Para el aplicativo web se usará también tecnología Microsoft, en este caso ASP Net Core en su versión 2.0, con lenguaje de compilación C#. El marco de trabajo fue NET Estándar Framework en su versión 2.0, se usará este tipo de tecnología puesto que el servidor este trabajado con el mismo lenguaje de programación lo que simplifica la comunicación entre los aplicativos. [7]

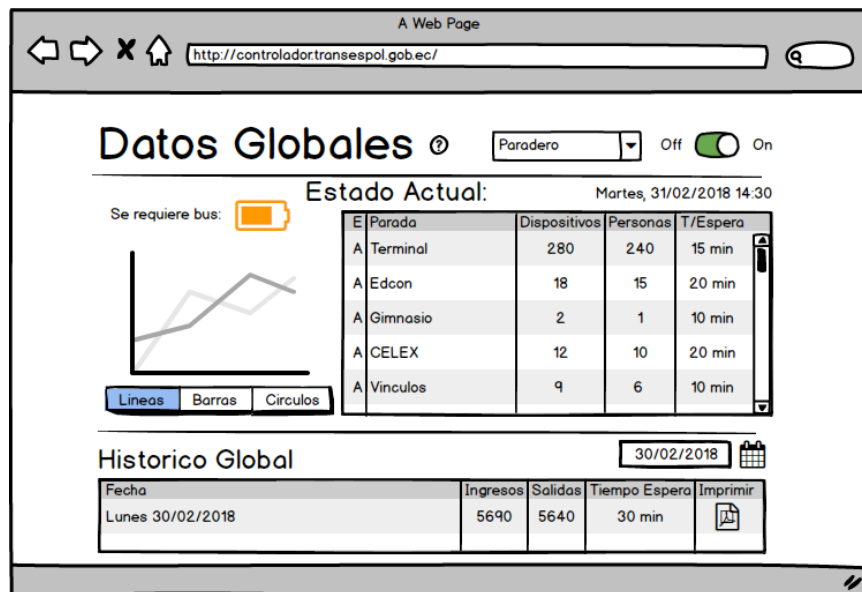


Figura 3.7 – Diseño de la interfaz para Pagina Web.

En la figura 3.7 presentamos el diseño para la página web con la cual el personal de TransESPOL podrá realizar el análisis de la información y generación de reportes.

En la parte superior de la interfaz web propuesta se mostraría una barra despegable donde el usuario puede elegir el paradero que desea consultar, el indicador de encendido y apagado indica si se está recibiendo la información en tiempo real, la parte inferior posee dos secciones la primera que muestra la cantidad de dispositivos encontrados en una tabla y una figura (en la interfaz de la figura, se muestra los datos globales por lo que se ve en la tabla la columna “E” de estado, donde la “A” significa activo y la “I” de inactivo, en el Anexo F puede encontrar también la interfaz para paradas), la información de la tabla principal muestra la cantidad de personas detectadas en las paradas y el tiempo de espera promedio, también se muestra de manera gráfica del lado derecho, en la sección inferior encontramos la segunda tabla que nos permitirá descargar los reportes diarios del histórico de registros en formato pdf.

3.4.2 Aplicativos

1. Aplicativo Móvil

Para el aplicativo móvil se usará Android Studio en su versión 3.0, con lenguaje de programación basado en JAVA. Para poder consumir el servicio MQTT se debe establecer una conexión al puerto dedicado en el servidor y conectar el aplicativo a internet, puesto que, el servicio está configurado en un servidor en internet, se hizo uso de la librería MQTT en su versión 3.0 desarrollada por Paho-Eclipse.

La aplicación será desarrollada para ser usada de forma intuitiva la cual contiene solo dos botones para cumplir con su objetivo:

- “Conectar” permite establecer la conexión entre el aplicativo móvil y el servidor Mosquito.
- “Solicitar Datos” pide la información al servidor Mosquito.

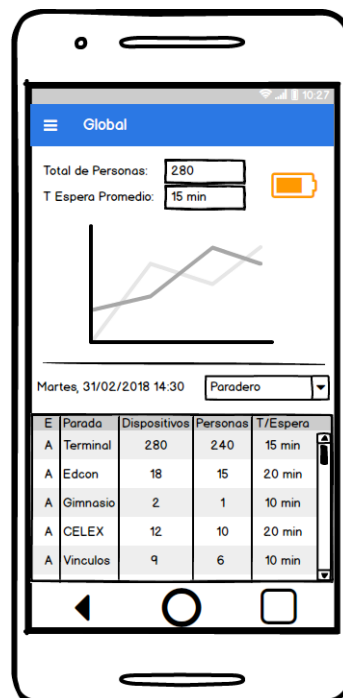


Figura 3.8 – Diseño de la interfaz para Aplicación Móvil.

En la figura 3.8 presentamos el diseño de la interfaz para la aplicación móvil. Esta interfaz móvil es parecida a la Aplicación web de la figura

3.7, muestra en la parte superior información en tiempo real del estado de la parada seleccionada mediante una gráfica con el detalle de la información representada en la parte inferior.

2. Aplicación del contadorMCU

Para el dispositivo se usará Visual Studio Code con el entorno de trabajo PlatformIO, permite usar de una mejor manera las librerías del lenguaje C (usado para extraer la dirección MAC) que su contraparte Arduino IDE, otra ventaja que se encontró es la compatibilidad con las librerías que se necesitaba para el control de los módulos de la tarjeta de memoria y la frecuencia de reloj.

Para controlar el módulo de la tarjeta MicroSD se usará la librería SD desarrollada por Adafruit y su versión 1.2.2, en la configuración electrónica del dispositivo está conectado a la placa por medio de los pines que permiten I2C.

Para controlar el módulo de la frecuencia de reloj (RTC) se usará la librería DS1302 desarrollada por Jeelabs en su versión 1.0.0 en la configuración electrónica del dispositivo está conectada a la placa por medio de los pines I2C.

3. Aplicativo en el servidor Receptor

Para el desarrollo de la aplicación que mantiene en memoria la cantidad de los teléfonos móviles que se encuentran activos se usará como lenguaje nativo C# en su versión 7.0 y también se usará el IDE de Visual Studio 2017 edición desarrolladora, siendo el entorno de trabajo fue NET Framework en su versión 4.7 desarrollados todos por Microsoft.

Se usará la librería M2MQTT desarrollada por Paho-Eclipse, en su versión 3.1 para aplicativos desarrollados en entorno NET Framework, esta librería permite poder trabajar con el protocolo MQTT, de esta

manera poder enviar y recibir los mensajes, cabe recalcar que la comunicación se establece por medio de un puerto en el que se da la comunicación de manera bidireccionalmente.

En la figura 3.9 se presenta el diseño para la aplicación del servidor receptor.

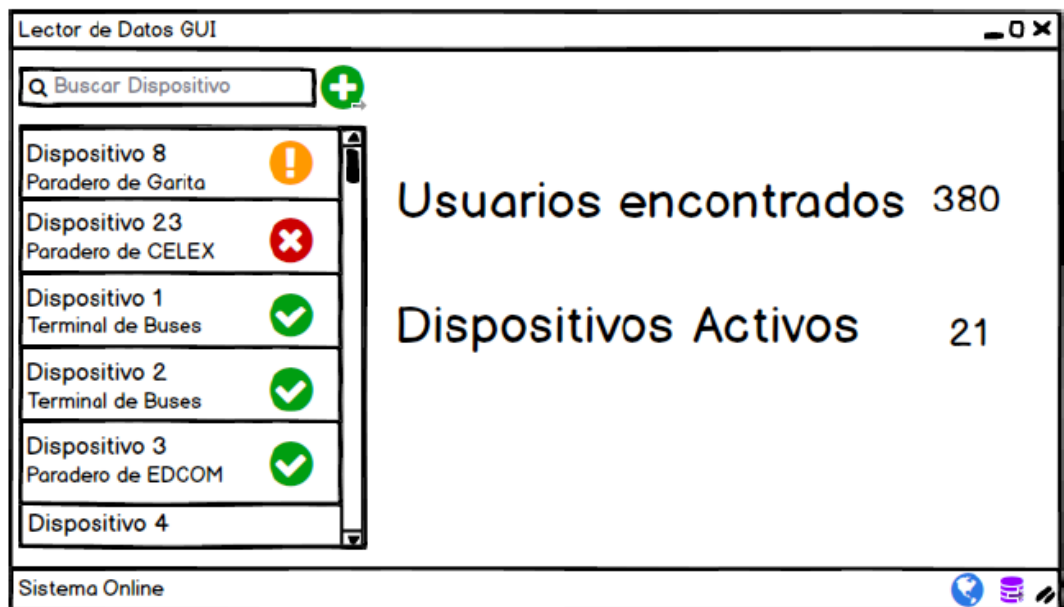


Figura 3.9 – Diseño de la Interfaz de Administración.

Esta Interfaz está pensada para que el Administrador pueda ingresar nuevos dispositivos con facilidad, además de ver el estado actual de cada uno con iconos indicadores (Advertencia, para informar posibles fallas como recalentamiento, la exís, fuera de línea y el visto, para todo está bien), también en la parte inferior indicadores del servidor de base de datos y pagina web.

En la figura 3.10 se muestra las especificaciones técnicas ya antes descritas, en esta también podemos observar que la comunicación entre los equipos conectados a la red se realiza por medio del protocolo TPC/IP.

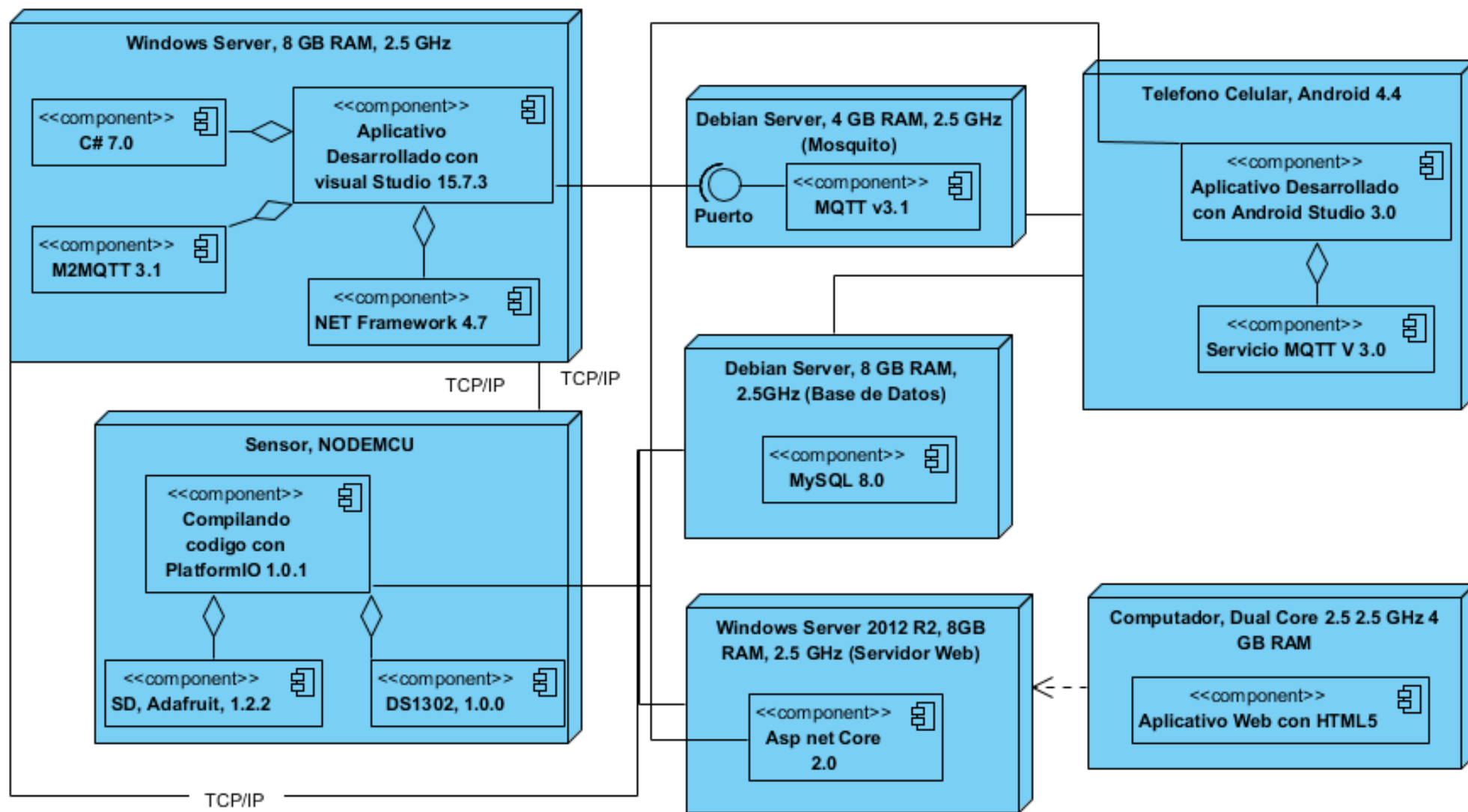


Figura 3.10 – Diagrama de Despliegue.

3.5 Pruebas con el ContadorMCU – Versión 1

En esta sección se describe el proceso de implementación del dispositivo en la terminal de buses, para las pruebas de funcionamiento.

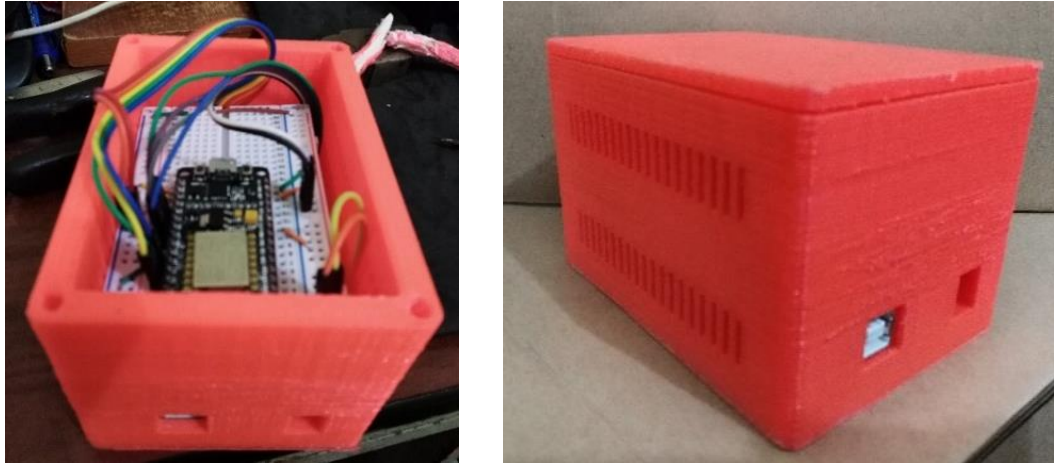


Figura 3.11 – Ensamblaje del dispositivo.

En la figura 3.11: la imagen de la **izquierda** muestra los componentes electrónicos dentro de la caja protectora y como se encuentran ubicados, mientras que en la imagen de la **derecha** muestra cómo queda el prototipo una vez que se ha cerrado la caja, y de esta manera estaría listo para ser colocado en la parada de buses.

Se determinó que inicialmente el dispositivo será puesto a prueba inicialmente una semana en la terminal de buses, para esto se habló con los directivos de la compañía para poder solicitar los permisos correspondientes y especificar detalles como punto de alimentación y envió de información hacia el servidor receptor de la información.

Pensando en dejar el contadorMCU como equipo desatendido por lo menos un mes, se detectaron inconvenientes para la colocación del dispositivo:

1. Lugar específico de la colocación.
2. Transmisión de información hacia el servidor de datos.

Para solucionar el problema de la ubicación se discutió con el personal de mantenimiento general de la ESPOL, puesto que por la experiencia de su trabajo pueden darnos una mejor opinión sobre un lugar óptimo, que el lugar apropiado

es la parte anterior de los televisores, favoreciendo el soporte con el que se agarra el equipo televisivo. Luego se pensó en la ruta que se quería monitorear para esto se habló con los controladores de la compañía, puesto que podrían dar una opinión que no se debería descartar, el mejor lugar sería monitorear la ruta Citymall, puesto que constantemente tiene usuarios que esperan el servicio.

No se pudo solucionar el problema de la transmisión hacia el servidor receptor de los datos, debido a la seguridad en las redes de datos que tiene ESPOL misma que nos impide poder comunicarnos hacia internet con el servidor mosquito, se limitó a que los datos sean almacenados en la tarjeta microSD

Se espera obtener una cantidad masiva de datos con los que se podría hacer análisis estadísticos entre ellos cálculo de varianzas entre otros. Estos análisis no estarán dentro del alcance de este proyecto, pero esta demás decir que se pueden generar proyectos posteriores en el área de Big Data.

En la figura 3.12 se muestran fotografías del proceso de instalación del contadorMCU

1. Se monitorea que el dispositivo este tomando datos correctos, puesto que, estará un mes ni supervisión.
2. Personal de servicios generales de la ESPOL identificando la ubicación idónea para colocar el contadorMCU.
3. Personal de servicios generales de la ESPOL, arreglando el cable de alimentación eléctrica.
4. ContadorMCU instalado satisfactoriamente.



Figura 3.12 – Instalación del prototipo de pruebas.

3.5.1 Aplicativo de escritorio de prueba para monitorear funcionamiento de dispositivo.

En esta sección se detalla brevemente la aplicación que se desarrolló para visualizar los datos que registra el contadorMCU, compilada para ser ejecutada en escritorios con sistemas operativos tipo Windows versiones superiores a igual a Windows 7, usando C# como lenguaje de programación nativo y como marco de trabajo .Net en su versión 4.7

A continuación, se muestran las interfaces que se diseñaron tomar en consideración que fue pensada para hacer pruebas del contadorMCU, la aplicación que se desarrollaría para la puesta en producción a la solución será web y se describe en el Anexo F de este documento.

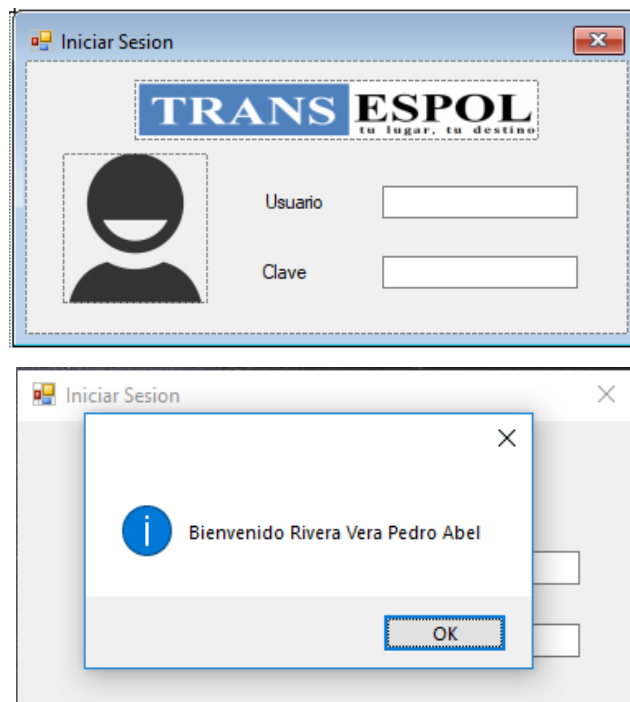


Figura 3.13 – Inicio de sesión de la aplicación de pruebas.

Existe una interfaz de inicio de sesión, figura 3.13, para impedir el acceso a personas no autorizadas, el usuario y la contraseña están almacenadas en la base de datos.

Cabe mencionar que al ser una aplicación de pruebas la administración de usuarios esta ejercida directamente en la base de datos, dos usuarios se mantienen estipulados de manera local, para poder hacer depuración en el sistema cuando no existe conexión a internet.

Luego de iniciar sesión se nos presentara siguiente y última pantalla la denominada “Pantalla Principal” constituida de varios componentes, tal como se muestra en la figura 3.14, necesarios para hacer una depuración directamente sobre el dispositivo.

Tomar en cuenta que se espera que el dispositivo no tenga acceso a internet durante las pruebas por lo que se agrega la recepción de los datos conectando el contadorMCU a un puerto USB disponible en el computador.

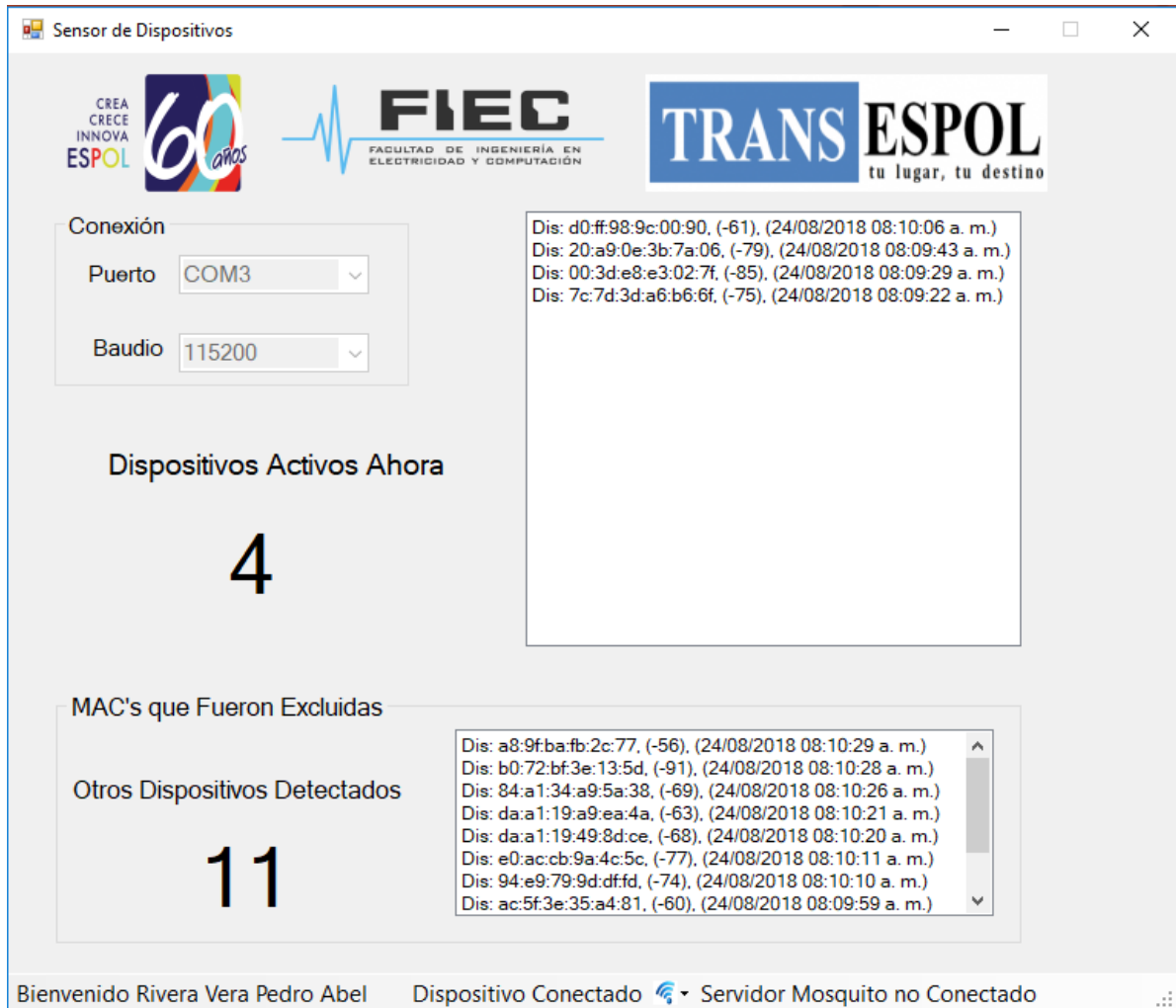


Figura 3.14 – Interfaz principal de la aplicación de prueba.

Bloques de componentes que tiene el aplicativo de pruebas:

1. Control de conexión con el contadorMCU por medio de USB. – permite la selección del puerto USB y la velocidad de transmisión de los datos (baudios), se deberá escoger la máxima 115200 bps, para evitar perder los datos.
2. La cantidad de dispositivos activos. – Nos muestra un valor numérico correspondiente al conteo realizado, tomar en consideración que la aplicación acceder a un archivo de texto donde se han estipulado los 6 primeros caracteres que debe considerara para determinar si es

una dirección MAC es válida, para efecto de estas pruebas, el archivo contiene las direcciones MAC que no debe considerar. Este valor será el enviado cuando se haga la petición por medio del protocolo MQTT.

3. Una lista de los dispositivos que se están registrando. – esta lista contiene los registros que están en memoria, cada registro contiene el RSSI, la MAC del dispositivo y la última vez que se actualizo (tiempo), este control representado por el rectángulo blanco, que está puesto en la parte superior derecha del formulario.
4. Otros dispositivos detectados. – nos muestra un valor numérico que representa la cantidad de dispositivos que fueron excluidos del contador general. Se debe recordar que esta exclusión es debido a que fueron MAC's que pertenecen a tabletas, puntos de Accesos u otros dispositivos que tienen un chip Wifi activo
5. Una lista de los dispositivos excluidos. – opera de igual manera que su control descrito en el ítem 3, siendo el rectángulo blanco, ubicado en la parte inferior del formulario.

Se incorporó una barra de tareas, tal como se muestra en la figura 3.15, esta muestra campos descriptivos, enfocados en la revelación de información fundamental para las pruebas que se están tomando, campos que va desde el nombre del usuario que inicio sesión, si el dispositivo está o no conectado, si se ha podido establecer la conexión con el servidor mosquito.

También se dispone de tres opciones las cuales: permiten cambiar la conexión serial del contadorMCU, y la conexión con el servidor mosquito, a continuación, se entra en breve detalles cada opción.

Conectar Dispositivo: nos permite establecer la comunicación serial con el dispositivo, mientras se reciben los datos los controles quedan

inhabilitados, se regresan a su estado anterior una vez desconectado tanto física o lógicamente el contadorMCU

Desconectar Dispositivo: nos permite interrumpir la conexión lógica con el dispositivo y habilita los controles de para poder seleccionar el USB y los baudios.

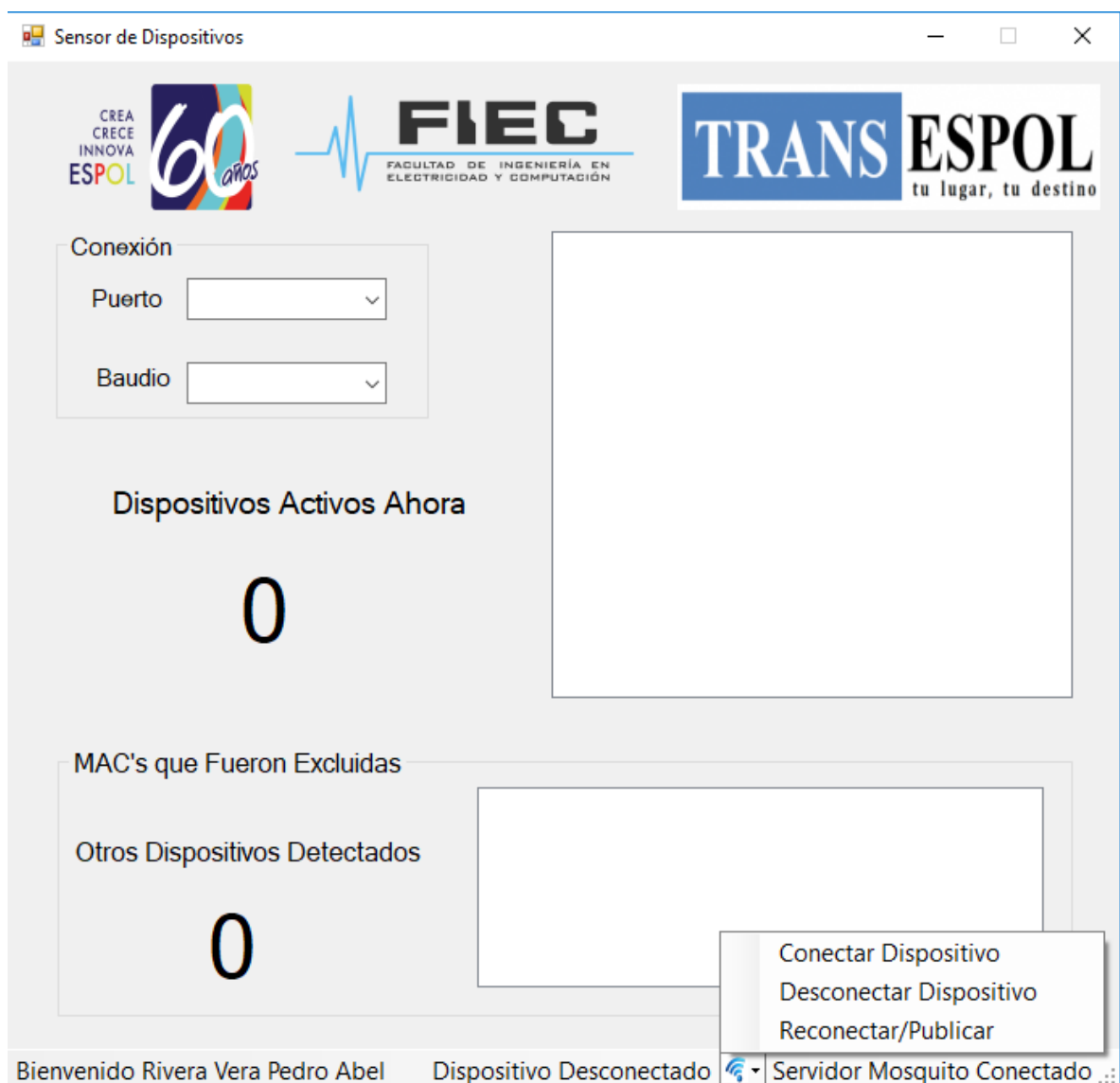


Figura 3.15 – Interfaz principal y la barra de tareas del aplicativo de prueba.

Reconectar/Publicar: esta opción tiene la función de interconectar el aplicativo con el servidor mosquito. Cabe mencionar que la aplicación

desde su ejecución inicial intenta establecer la comunicación con el servidor mosquito, de no conseguirlo se podría hacer durante la ejecución y monitoreo de los datos por medio de este botón.

Para terminar la ejecución del programa se debe hacer desde la ventana de inicio de sesión, cuando se cierra la pantalla principal se cerrará la conexión con el puerto USB, con el servidor MQTT, y presentará la ventana de inicio de sesión.

3.5.2 Aplicativo Móvil de Prueba

Se desarrolló una aplicación para dispositivos móviles Android para que pueda ser ejecutada desde la versión 4.4, con la cual se muestra la información que fue enviada al servidor mosquito solicitando la cantidad de personas que se encuentran en un lugar ya sea en las paradas o el terminal de buses. Cabe mencionar que esta no será la aplicación definitiva solo se desarrolló para fines académicos de demostración al momento de realizar las pruebas con el prototipo, las imágenes de cómo debe quedar la aplicación para el controlador se detalla en los anexos de este documento.

La figura 3.16 muestra las dos interfaces que se realizaron para las pruebas del contadorMCU, la imagen que está a la izquierda es la interfaz de inicio de sesión y la imagen que se encuentra a la derecha es la interfaz principal que muestra la información de la cantidad estimada de personas en la parte central de la pantalla.

El usuario y la contraseña para iniciar sesión en la aplicación están establecidos en la base de datos, no almacena usuarios de manera local, al contrario de la aplicación de escritorio previamente descrita, puesto que se requiere que el teléfono esté conectado a internet para recibir datos del servidor MQTT.

En la interfaz principal existen solo dos opciones para el funcionamiento:

Conectar: este nos permite que el teléfono establezca comunicación con el servidor mosquito y pueda recibir los mensajes por el canal “Administración/tenemos” los valores que le llegan, en la tabla 3.1 y 3.2 se describen los funcionamientos de los canales de comunicación.

Representado en la aplicación en forma de “nube desconectada” (una barra lateral de izquierda a derecha pasa sobre esta) según los iconos del propio sistema operativo, cuando se logra la comunicación con el servidor mosquito esta cambia a la forma “nube conectada”.

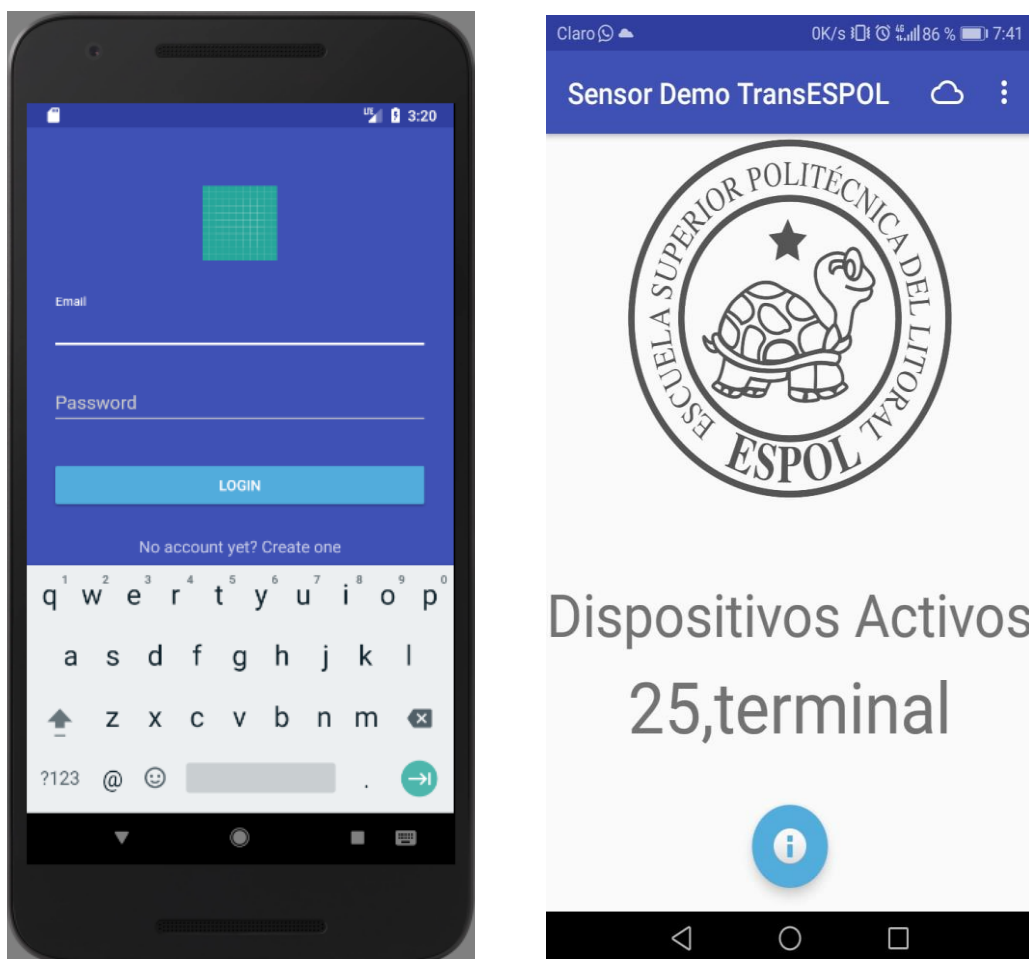


Figura 3.16 – Interfaz de aplicación móvil.

Pedir información: este envía por el canal de comunicación “Administración/Admin” con mensaje 1, para recibir la información se

escucha el canal "Administración/tenemos", se espera el valor de la cantidad de personas detectadas y el lugar. Representada en la aplicación como la "i".

En el aplicativo se toman en consideración las siguientes validaciones:

1. No puede pedir información si no se ha establecido la comunicación con el servidor mosquito.
2. Cada aplicación tiene un usuario interno para autenticarse al servidor mosquito, este usuario tiene aplicada las ACL, correspondiente para evitar fallas de seguridad.
3. Ante de pedir información se envía un mensaje de consulta para determinar si hay un contadorMCU emitiendo información, no darse el caso, que no existiese alguno enviando información al servidor de proceso la aplicación no abrirá el canal de comunicación para solicitar información.

3.5.3 Datos recolectados

Se realizó muestras de captación de datos durante 3 días consecutivos, Se comenzó desde el martes 7 de agosto de 2018, usando como periodo de 4 horas por día: comenzando a las 12:00 PM.

- El promedio de registros que se obtuvo durante las 4 horas de recolección de datos fue de casi 138.000.
- Entre las 14:01 y las 15:00 se obtuvo aproximadamente un poco más de 31.000 registros lo que nos representa un 23%.
- Entre las 15:01 y las 16:00 se obtuvo aproximadamente casi 28.000 registros lo que representa un 21%.

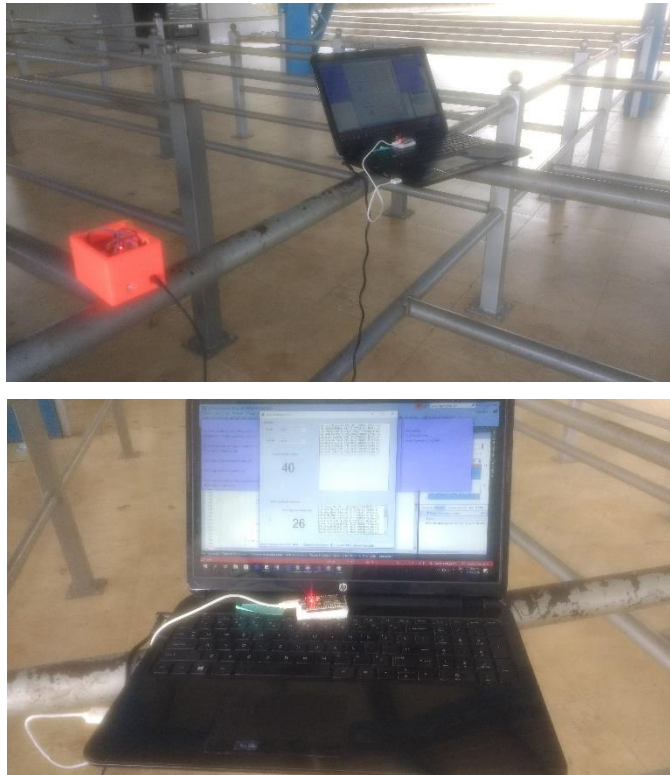


Figura 3.17 – Prueba del dispositivo y toma de datos.

En la figura 3.17 se puede visualizar el contadorMCU ubicado en el andén de la ruta Sur junto con la computadora que tiene el aplicativo en ejecución, tomando correctamente los datos, se estableció una comunicación serial entre ambos equipos.

No se usó el computador por los días que se estuvieron monitoreando datos solo durante el primero, por problemas de la alimentación eléctrica del lugar donde se tomó datos fue en el carril central de la terminal de buses, lado derecho, siendo el único punto eléctrico habilitado.

CAPÍTULO 4

PLAN DE IMPLEMENTACIÓN Y COSTOS

4.1 Plan de Implementación

La implementación completa de la solución se realizará en 66 días. En la tabla 4.1 presentamos en detalle las actividades a realizar para la implementación.

Tabla 4.1 - Plan de Implementación.

Id	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras
1	Implementación de Solución	66 días	mié 12/9/18	mié 12/12/18	
2	Levantamiento de la Información	10 días	mié 12/9/18	mar 25/9/18	
3	Reunión con personal de TransESPOL	1 día	mié 12/9/18	mié 12/9/18	
4	Reunión con personal de ESPOL	1 día	jue 13/9/18	jue 13/9/18	3
5	Toma de mediciones y ubicación de los equipos	4 días	vie 14/9/18	mié 19/9/18	4
6	Diagrama físico de la Solución	2 días	jue 20/9/18	vie 21/9/18	5
7	Aprobación y correcciones en el Diagrama físico	2 días	lun 24/9/18	mar 25/9/18	6
8	Adquisición de Equipos	15 días	mié 26/9/18	mar 16/10/18	
9	Cotización de Equipos y componentes electrónicos	5 días	mié 26/9/18	mar 2/10/18	7
10	Adquisición de Equipos y componentes electrónicos	10 días	mié 3/10/18	mar 16/10/18	9
11	Adecuación de Cuarto de Servidores	7 días	mié 17/10/18	jue 25/10/18	
12	Instalación del cableado	1 día	mié 17/10/18	mié 17/10/18	10
13	Instalación de Servidores	2 días	jue 18/10/18	vie 19/10/18	12
14	Instalación y configuración de conmutador	1 día	lun 22/10/18	lun 22/10/18	13
15	Configuración de Servidores	2 días	mar 23/10/18	mié 24/10/18	14
16	Pruebas de funcionamiento	1 día	jue 25/10/18	jue 25/10/18	15
17	Instalación de ContadorMCU	24 días	vie 26/10/18	mié 28/11/18	
18	Ensamblaje de ContadoresMCU	12 días	vie 26/10/18	lun 12/11/18	16
19	Instalación de los ContadoresMCU	5 días	mar 13/11/18	lun 19/11/18	18
20	Configuración de los Contadores	1 día	mar 20/11/18	mar 20/11/18	19
21	Pruebas de funcionamiento	6 días	mié 21/11/18	mié 28/11/18	20
22	Puesta en Producción	10 días	jue 29/11/18	mié 12/12/18	
23	Presentación de Solución Funcionando	1 día	jue 29/11/18	jue 29/11/18	21
24	Capacitaciones	5 días	vie 30/11/18	jue 6/12/18	23
25	Entrega de documentación técnica	4 días	vie 7/12/18	mié 12/12/18	24

Es importante mencionar que algunos hitos pueden llegar a tardar más tiempo del estimado, ya que depende de factores externos, como por ejemplo las reuniones con el personal de TransESPOL y ESPOL, que son necesarias para establecer los equipos con los que se cuenta y los que la universidad nos puede brindar para ahorrar costos.

El hito 10 de Adquisición de equipos y componentes electrónicos, también considera el mandado a hacer de las 12 placas para los contadoresMCU, para las diferentes paradas de buses. Gracias a que los componentes pueden ser conseguidos localmente sin problemas, no es necesario importarlos.

El ensamblaje de cada contadorMCU tardaría alrededor de 4 horas, pero al ser 12 hemos considerado el establecer un tiempo de 1 día para cada uno, por posibles problemas.

4.2 Costos

A continuación, se detallan los valores del dispositivo en la tabla 4.2 y de la implementación de la solución como tal en la tabla 4.3:

Tabla 4.2 – Detalle de Costo de componentes electrónicos del dispositivo.

Cantidad	Descripción	V. Unitario	Valor
2	NodeMCU	\$ 10,00	\$ 20,00
1	Caja Protectora ²	\$ 51,20	\$ 51,20
1	Módulo SD Card	\$ 2,50	\$ 2,50
1	Módulo RTC	\$ 4,00	\$ 4,00
1	Construcción de circuito electrónico PCB	\$ 40,00	\$ 40,00
Total			\$ 117,70

² Se puede abaratar costos dependiendo del lugar de la imprenta en 3D, por medio de una suscripción.

Tabla 4.3 - Costos de Implementación de la Solución.

Cantidad	Descripción	V. Unitario	Valor
12	<ul style="list-style-type: none"> Dispositivo (Vea tabla 4.2) 	\$ 117,70	\$ 1404,70
1	<ul style="list-style-type: none"> Wifi 802.11 de 2,4 GHZ 	\$ 50,00	\$ 50,00
4	Servidores <ul style="list-style-type: none"> RAM 8 GB Almacenamiento 1024 GB Procesador Xeon v5 3 GHZ Marca DELL, tipo torre^{3 4} 	\$ 957,00	\$ 3828,00
	Mano de Obra (Instalación y configuración de equipos)	\$ 1320,00	\$1320,00
		Total	\$ 6602,70

En el Anexo G se detalla las especificaciones técnicas que deberían tener los servidores físicos.

³ Se puede abaratar costos, manteniendo varios servicios en un mismo servidor, aunque no es lo más aconsejable, otra manera que se puede reducir el valor del coste de infraestructura es mantener los servicios en la nube.

⁴ En los anexos se establece el detalle técnico de los servidores.

CONCLUSIONES Y RECOMENDACIONES

Durante la implementación del prototipo se logró poner en práctica conocimientos académicos que adquirimos a lo largo de la carrera tales como programación orientada a objetos, implementación y administración de servidores, fundamentos de redes inalámbricas y comunicación de datos.

Se evidenció que podemos crear soluciones tecnológicas para problemas reales. Se diseñó e implementó de manera exitosa un prototipo capaz de estimar la cantidad de personas en un determinado lugar eficiente y económica, permitiéndole a TransESPOL, satisfacer sus necesidades con una mínima inversión.

Conclusiones

Se logró implementar un programa Sniffer con el que se contabilizó los estudiantes gracias a sus teléfonos inteligentes con wifi, esta información, aunque no es exacta, resultó ser una muy buena aproximación a la cantidad real de estudiantes en la parada.

En las pruebas de funcionamiento captamos datos desde las 12:00 hasta las 16h00 y con la información recolectada se logró determinar que la mayor concentración de registros estaba entre las 12:00 y las 14:00 siendo el 56% del total contabilizado. Además, hubo una mayor concentración de estudiantes esperando los buses para poder irse a sus casas en los primeros días de la semana, mientras que más se acercaba el viernes reducía en menor cantidad los estudiantes esperando bus en el terminal.

La cantidad de información que se recoge y su estructura de datos que captaran los dispositivos es muy densa pudiéndose dar lugar a nuevos estudios sobre todo estadísticos que nos permitan determinar factores que tienen que ver la eficiencia de la compañía.

Dada la capacidad de la solución de generar un histórico de personas por fecha y hora, es posible, con los análisis necesarios la predicción de la cantidad de personas que estarán en un lugar determinado.

Recomendaciones

El proyecto puede servir como base para la implementación de diferentes soluciones tecnológicas, estadísticas y logísticas. Para la presentación de la información a los clientes interesados, herramientas para análisis de datos o estudios que permitan mejorar el servicio brindado por TransESPOL.

Se recomienda hacer la implementación de nuevos contadores dentro de la terminal de buses para aumentar la precisión de la cantidad de personas.

Actualmente se establece por medio de código la ubicación del dispositivo, se recomienda poder actualizarla a través del aplicativo web, cada vez que se cambie su lugar de captura de datos.

Aunque en un solo servidor físico se pueden implementar varios de los servicios requeridos, esto no es recomendable, lo que se recomienda es usar implementaciones de pago por uso en la nube donde se obtendría una reducción considerable del presupuesto de implementación de la solución, incluyendo que no se necesita gastos extras de infraestructura física adecuada para los servidores físicos.

Se recomienda dejar por lo menos un mes el dispositivo capturando datos para poder luego hacer análisis de varianza con respecto al tiempo promedio que un estudiante espera un bus y determinar la eficiencia de envío de los buses.

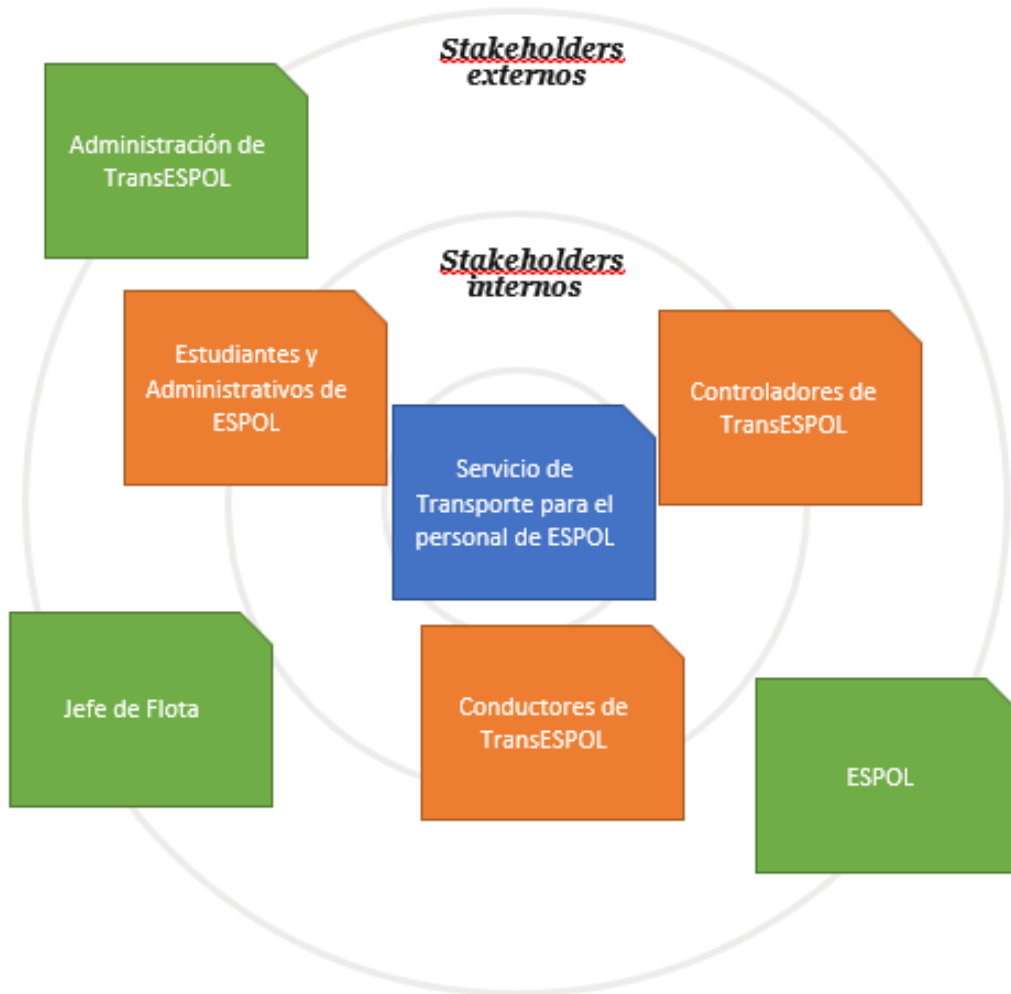
BIBLIOGRAFÍA

- [1] F. O. Espinoza, K. Bustamante, N. Naranjo y M. F. Ronquillo, «RENDICIÓN DE CUENTAS 2016,» *POLIGRÁFICA C.A.*, p. 22, 2016.
- [2] IEEE Standards Association, «IEEE Standards Association Registration Authority,» [En línea]. Available: <https://regauth.standards.ieee.org/standards-raweb/pub/view.html#registries>.
- [3] Espressif Systems, «ESP8266ex Datasheet,» 2018. [En línea]. Available: https://www.espressif.com/sites/default/files/documentation/0a-esp8266ex_datasheet_en.pdf.
- [4] Handsontec Technology, «ESP8266 NodeMCU WiFi DevKit,» [En línea]. Available: http://www.handsontec.com/pdf_learn/esp8266-V10.pdf.
- [5] I. Kravets, «PlatformIO is an open source ecosystem for IoT development,» PlatformIO, 17 Julio 2018. [En línea]. Available: <https://docs.platformio.org/en/latest/what-is-platformio.html>. [Último acceso: 17 Agosto 2018].
- [6] Oracle Corporation and/or its affiliates , «MySQL 8.0 Reference Manual,» Oracle Corporation, 1 Agosto 2018. [En línea]. Available: <https://dev.mysql.com/doc/>. [Último acceso: 19 Agosto 2018].
- [7] D. Roth, R. Anderson y S. Luttin, «Introduction to ASP.NET Core,» Microsoft, 02 Febrero 2018. [En línea]. Available: <https://docs.microsoft.com/en-us/aspnet/core/?view=aspnetcore-2.1>. [Último acceso: 18 Agosto 2018].
- [8] L. E. Aguilar, F. Ortiz, K. Bustamante, N. Naranjo y S. Troya, «Informe de Rendición de Cuentas 2017,» *ESPOL*, pp. 16-17, 2017.

ANEXOS

ANEXO A

Mapa de Actores



ANEXO B

Cuadro interrogativo para estudiantes.

¿Qué?	¿Cómo?	¿Por qué?
<ul style="list-style-type: none"> • ¿Cuántos años tienes siendo estudiante de ESPOL? • ¿Qué es lo que han escuchado de TRANSESPOL? • ¿Qué es lo que han Visto de TRANESPOL? • ¿En qué parte de la Ciudad Vive el estudiante? • ¿Cuál es tu opinión respecto al cambio de las rutas? • Si pudieras cambiar algo respecto al servicio de TRANSESPOL, ¿Qué cambiarías? 	<ul style="list-style-type: none"> • ¿Frecuencia de uso con el servicio de TRANSESPOL? • Califica tu nivel de satisfacción usando el servicio de TRANSESPOL • ¿En qué crees que fallan los choferes? • ¿Cuál Crees que sea una solución a corto plazo para la actitud de los choferes? • desde el inicio de clases ¿Consideras que los buses ahora salen más rápido? • ¿cuánto tiempo te ha tocado esperar en la parada o terminal de buses? • ¿Conoces las fuentes de información que ofrece TRANESPOL? • En general, cuál es tu opinión respeto al servicio de TRANSESPOL 	<ul style="list-style-type: none"> • Podrías Darnos una opinión sobre el servicio de TRANESPOL • ¿prefieres coger el bus en la parada o en el terminal de buses? ¿Por qué? • ¿Por qué razones no utilizarías los buses de TRANSESPOL

Cuadro interrogativo para controlador y conductores de buses.

¿Qué?	¿Cómo?	¿Por qué?
<ul style="list-style-type: none"> • ¿Qué tan grande es la carga laboral en TRANSESPOL? • ¿Qué ha cambiado en los últimos años? • ¿Qué cambiaría usted para mejorar la situación actual de la empresa? • ¿Cuánto tiempo tiene trabajando para TRANSESPOL? • ¿Ha trabajado en alguna otra cooperativa de transporte? • ¿Qué ha escuchado que dicen los estudiantes de la empresa? • ¿Cuántos cursos, taller, seminarios o demás actividades le han patrocinado la empresa? 	<ul style="list-style-type: none"> • ¿Cómo se siente trabajando en TRANSESPOL? • ¿Cómo han influido en su trabajo los cambios en las rutas de buses? • ¿Tiene experiencia trabajando en alguna otra cooperativa de transporte urbano? • ¿Cómo se diferencia TRANSESPOL de estas? • ¿cómo catalogaría el comportamiento de los estudiantes hacia la empresa? 	<ul style="list-style-type: none"> • ¿Por qué razón está trabajando en TRANSESPOL? • ¿Cuáles cree usted que son las causas que provocan el malestar general de los estudiantes respecto al servicio brindado por TRANSESPOL? • ¿Por qué cree usted que TRANSESPOL está en quiebra? • ¿Por qué razones desistiría de trabajar en TRANSESPOL? • ¿Considera que se mejoró el Sistema de transportación de los estudiantes con las nuevas rutas?

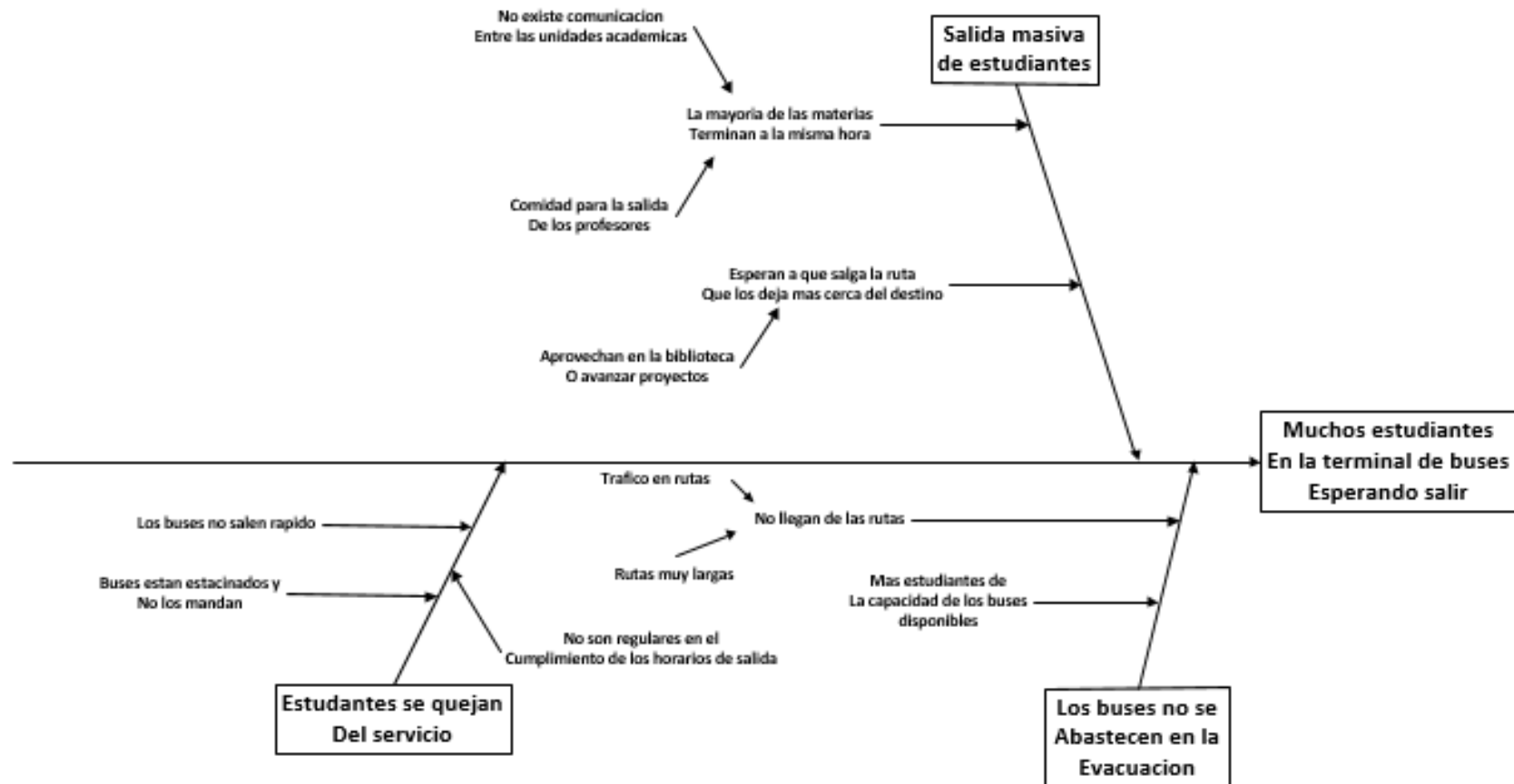
ANEXO C

Mapa de Empatía de la entrevista

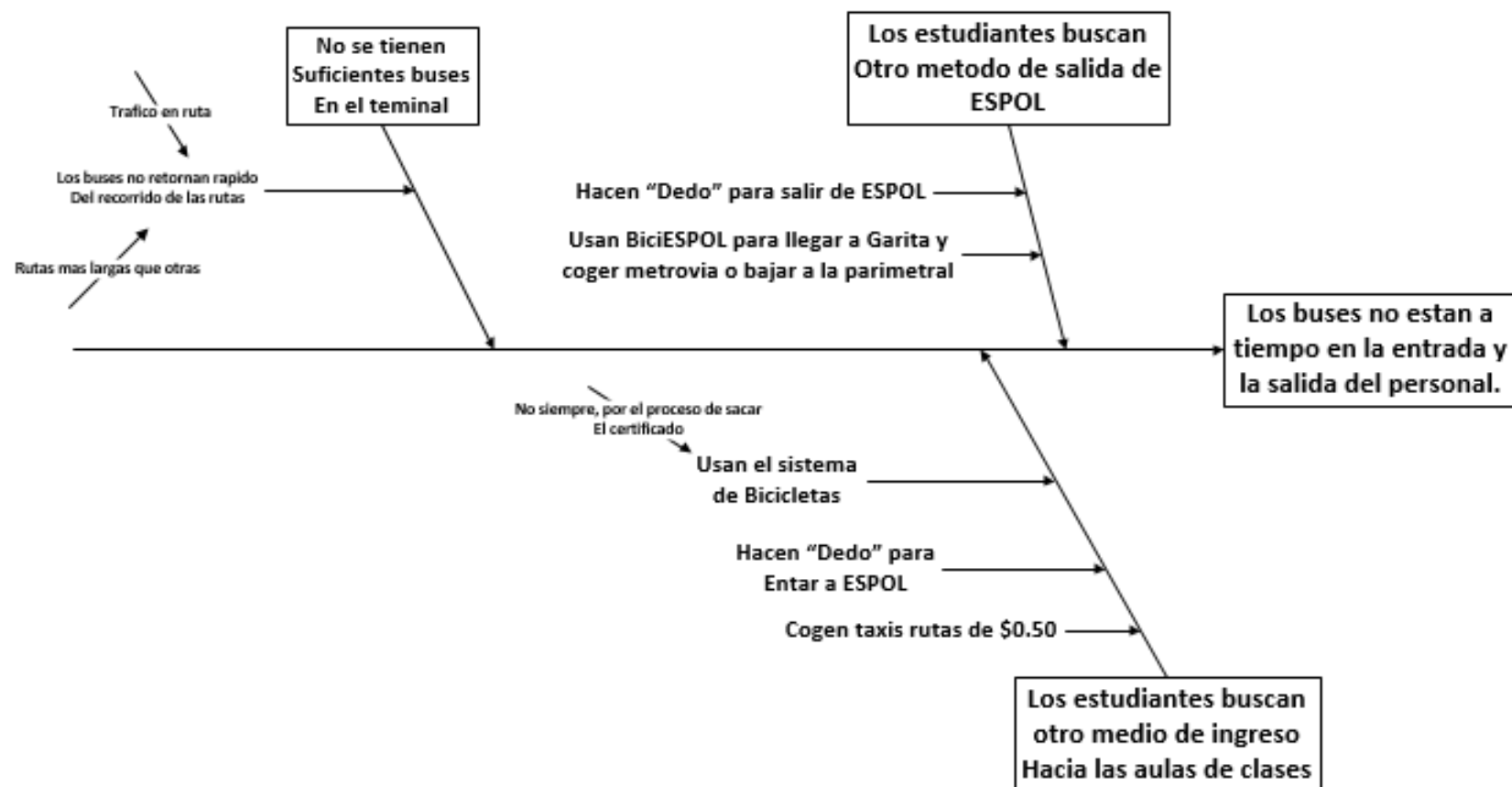
Alejandro Chanaba Presidente del Directorio de TransESPOL	
DICE	PIENSA
<ul style="list-style-type: none"> El plan es sacar a los estudiantes lo más pronto posible El plan es sacarlos a un lugar donde puedan coger más buses. El Sistema funcionará con el carnet del estudiante El estudiante tendrá que recargar para poder subirse al bus Paquetes de recargas con descuentos Todos los buses tendrán aire acondicionado y Wifi El Sistema está financiado con una alianza publico privado Necesita datos reales para poder tomar decisiones correctas TransESPOL no debe ser la solución de transporte El Sistema estará listo en agosto 	<ul style="list-style-type: none"> Se debe pagar por usar los parqueaderos El sistema puede ayudar a la toma de listas El CityMall es un lugar chévere en donde dejar los estudiantes Debe evitar que los conductores manejen dinero ESPOL debe hacer las cosas bien hechas Es imposible que podamos servir a todo el mundo y que todos queden contentos. Saber la gente en fila disminuiría los tiempos de espera Debe apuntar a las nuevas generaciones de estudiantes, porque a los antiguos ya los perdió.
HACE	SIENTE
<ul style="list-style-type: none"> Disminuyo las rutas largas por unas más cortas. Aumento la frecuencia de buses Incremento el número de veces que un carro sale. Cambio de rutas validado con datos de un estudio de 6 meses Transportan 8mil estudiantes balanceado entre interno y el resto de las rutas No se están usando los electrónicos disponibles en la Terminal TransESPOL no tiene datos estadísticos, aparte de la que él hizo La compañía está en quiebra 	<ul style="list-style-type: none"> Estaban mal planificadas las rutas de buses La Administración anterior dejo todo mal, hicieron las cosas a lo loco, a dedo. Le gusta la metodología de desing thinking El sistema evitara la evasión de la entrega del dinero No tenemos un sistema de conteo confiable Los estudiantes deben educarse y adaptarse a los horarios de los buses Existen mucha gente que se queja del sistema Se siente mal porque Cuenca se adelantó con lo del sistema. Los estudiantes ven a la compañía con mala reputación

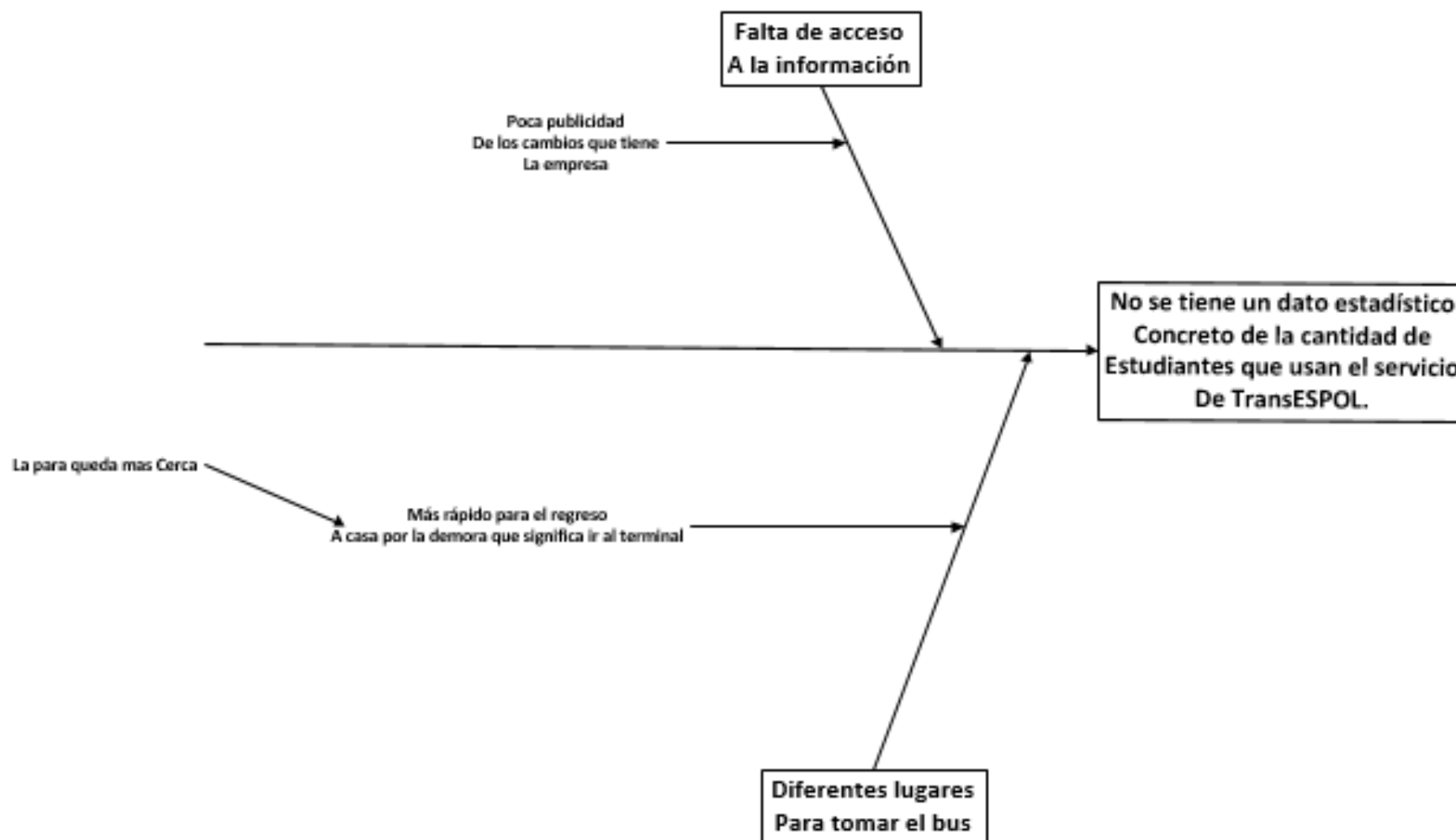
ANEXO D

Espina de Pescado de Estudiantes



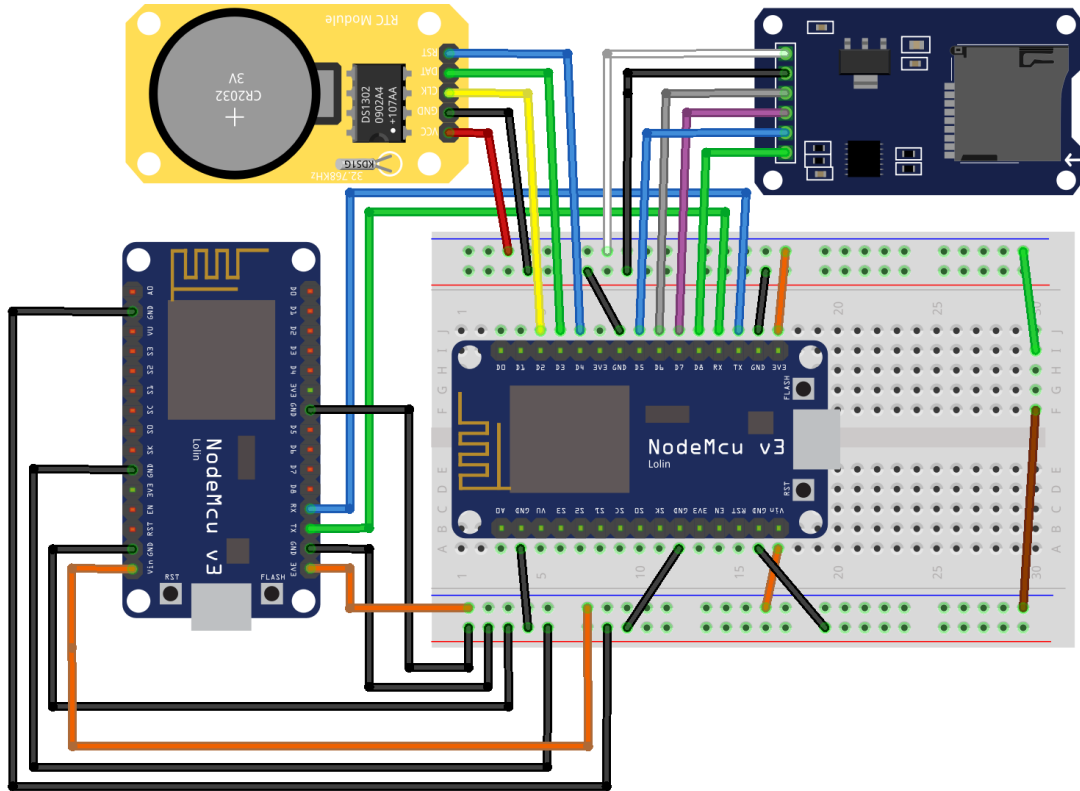
Espina de Pescado de TransESPOL



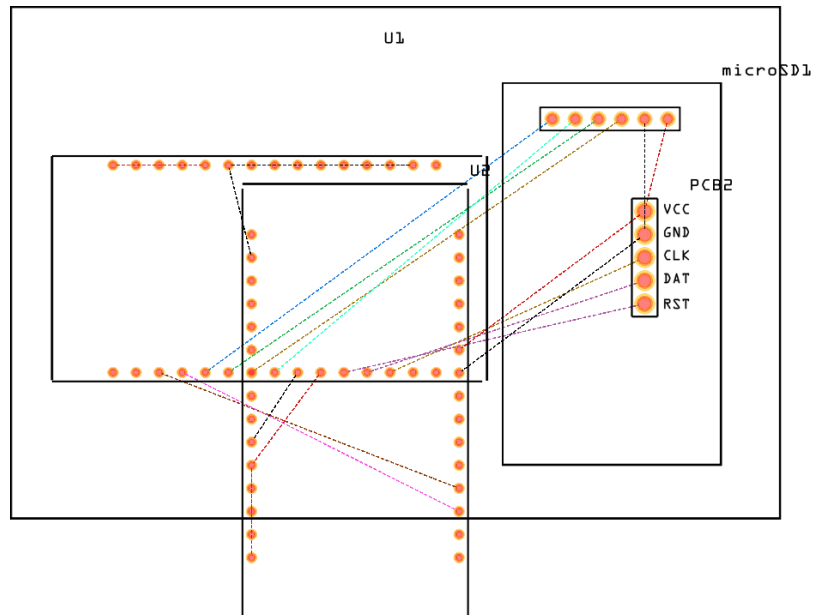


ANEXO E

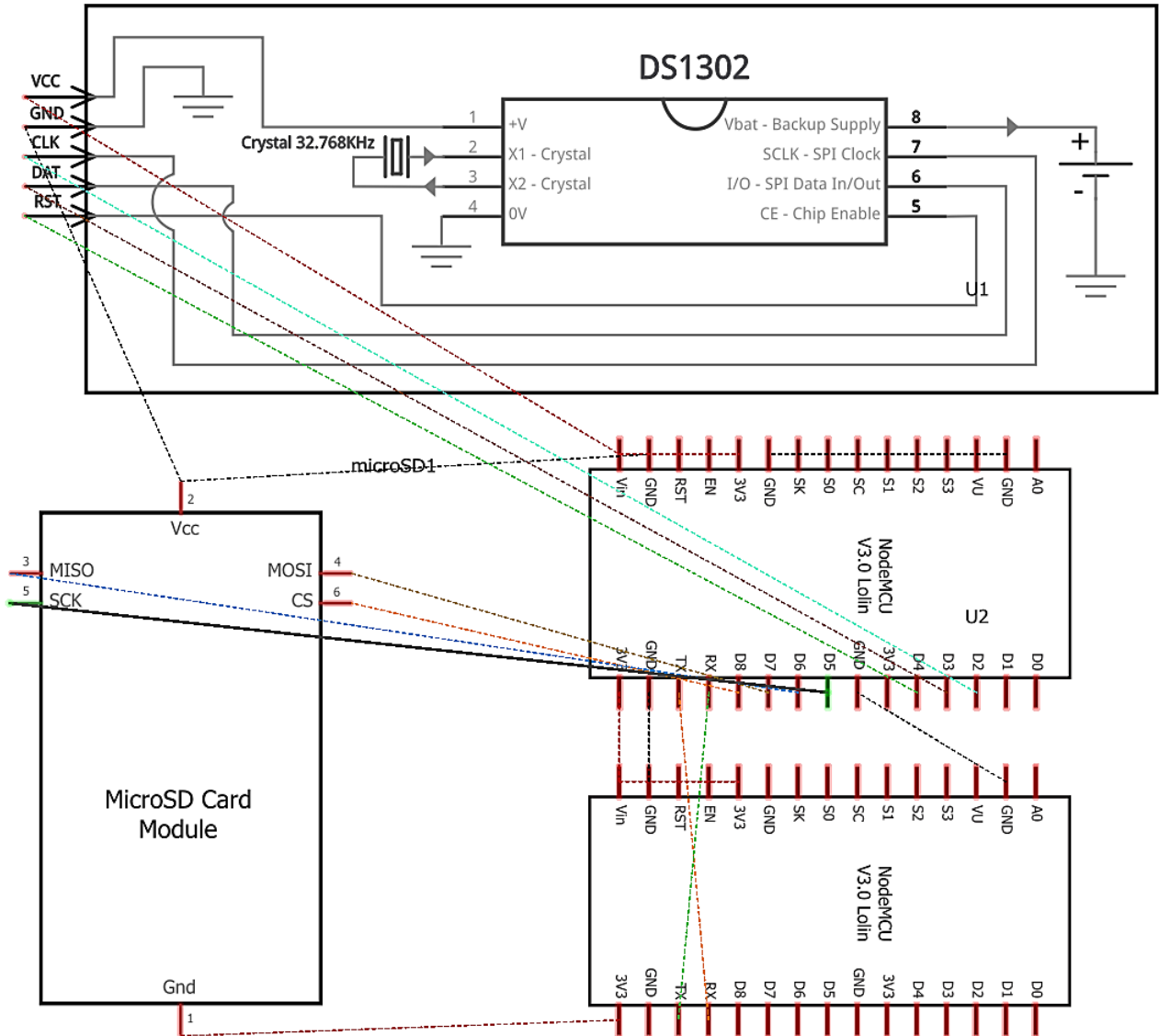
Diagramas del dispositivo hechos en Fritzing



fritzing



fritzing

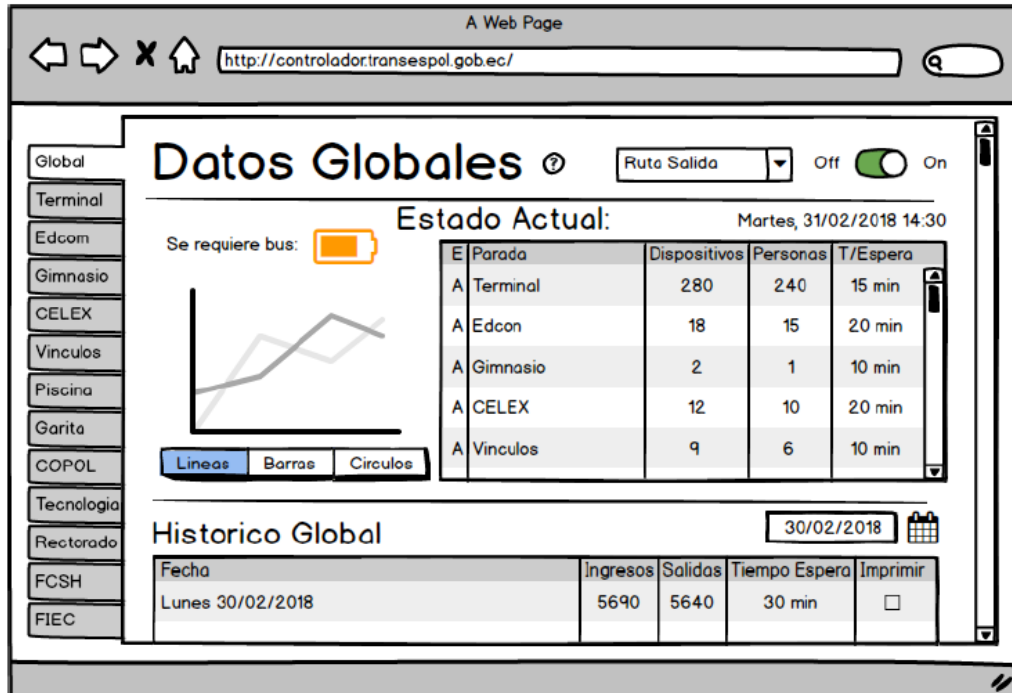


fritzing

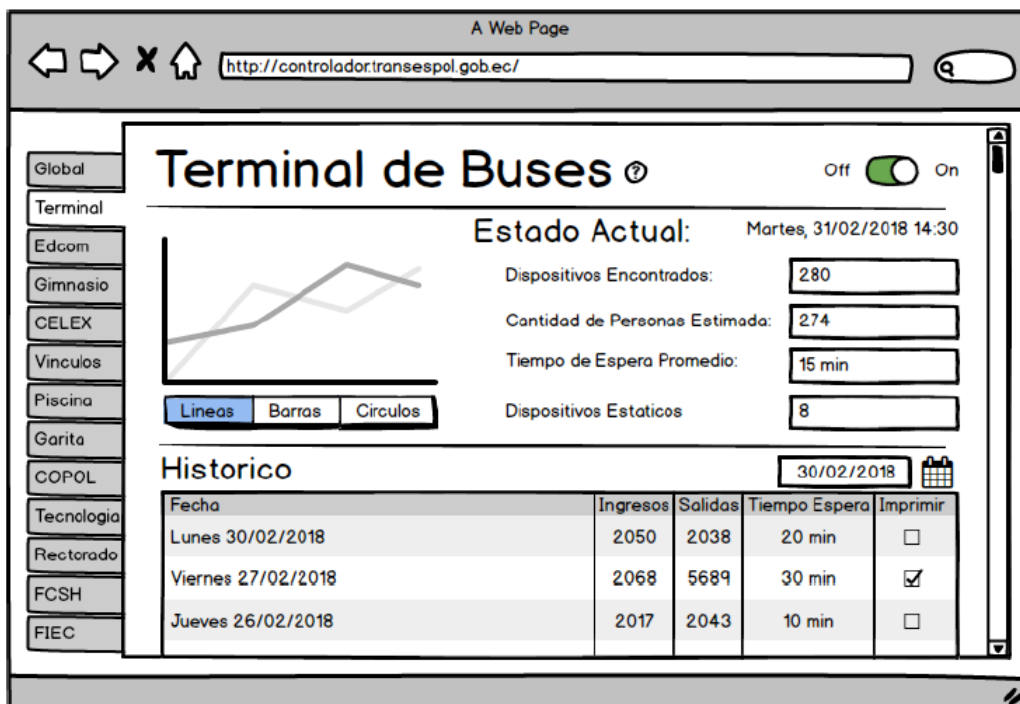
ANEXO F

Interfaz de Usuario en Oficinas de TransESPOL

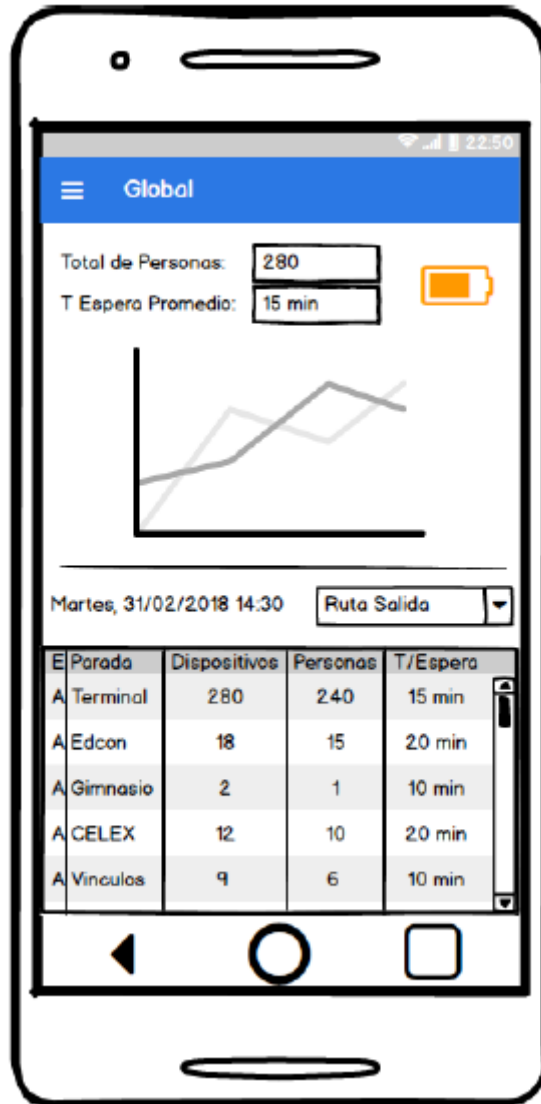
Interfaz de Información Global



Interfaz de Información de una Parada



Interfaz del Controlador en el Celular



ANEXO G

Características de los servidores en forma de torre.

Características	Especificaciones Técnicas	
Factor de forma	Servidor en mini torre	
Dimensiones	Alt.: 36.0 cm (14.17 in) An.: 17.5 cm (6.89 in) Prof.: 43.5 cm (17.12 in)	
Procesador	1 procesador de las siguientes familias de productos: · Procesador Intel® Xeon® de la familia de productos E3-1200 v6 · Intel Pentium® · Intel Core™ i3 · Intel Celeron®	
Chipset	Intel C236	
Sistema operativo	Microsoft® Windows Server® 2012 Microsoft Windows Server 2012 R2, x64 Microsoft Windows Server 2016	Red Hat® Enterprise Linux® VMware® vSphere® ESXi® SUSE Linux Enterprise Server
Memoria	Arquitectura: Hasta 2,400 MT/s de DIMM DDR4 Tipo de memoria: UDIMM	Sockets de módulo de memoria: 4 Cantidad máxima de RAM: Hasta 64 GB
Compatibilidad con hipervisores	Microsoft Windows Server con Hyper-V® VMware vSphere ESXi®	
Almacenamiento	Discos duros SATA Enterprise de 3.5 in y 7,200 r/min Discos duros SAS nearline de 3.5 in y 7,200 r/min Unidades SATA de entrada de 3.5 in y 7,200 r/min	
Bahías de unidades	4 discos duros cableados de 3.5 in	
Slots	1 de 8 PCIe 3.0 (con 16 conectores) 1 de 4 PCIe 3.0 (con 8 conectores) 1 de 4 PCIe 3.0 (con 8 conectores)	

	1 de 1 PCIe 3.0 (con 1 conector)	
Controladoras RAID	PERC S130, PERC H330, PERC H730, PERC H830	
Controladora de redes	Broadcom® BCM5720	
Comunicaciones	2 LOM de 1 GbE Haga clic aquí para información sobre compatibilidad del T130 con tarjetas de interfaz de red (NIC) y tarjetas de adaptadores de bus del host (HBA), y desplácese hacia la sección “Tarjetas de red adicionales”.	
Alimentación	PSU cableado de 290 W	
Administración	<p>Administración de sistemas:</p> <ul style="list-style-type: none"> · Compatible con IPMI 2.0 · Dell OpenManage Essentials · Dell OpenManage Mobile · Dell OpenManage Power Center <p>Administración remota:</p> <ul style="list-style-type: none"> · iDRAC8 con Lifecycle Controller, iDRAC8 Express (predeterminado), iDRAC8 Enterprise (actualización), medios vFlash de 8 GB (actualización), medios vFlash de 16 GB (actualización) 	<p>Conexiones de Dell OpenManage:</p> <ul style="list-style-type: none"> · HP Operations Manager, IBM Tivoli® Netcool® y CA Network and Systems Management, plug-in de Dell OpenManage para Oracle® Database Manager <p>Integraciones de Dell OpenManage:</p> <ul style="list-style-type: none"> · Suite de integración de Dell OpenManage para Microsoft System Center, integración de Dell OpenManage para VMware vCenter®
Acceso a los dispositivos	8 puertos USB en total: <ul style="list-style-type: none"> · Parte posterior: 2 USB 3.0 más 4 USB 2.0 · Parte frontal: 1 USB 2.0 más 1 USB 3.0 	

Fuente tomada de: <http://www.dell.com/ec/empresas/p/poweredge-t130/pd>