

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA



PONTIFICIA
UNIVERSIDAD
CATÓLICA
DEL PERÚ

**SISTEMA DE MONITOREO DE UNA RED DE BUSES DE
TRANSPORTE PÚBLICO E INFORMACIÓN PARA USUARIOS
EMPLEANDO TRANSCPTORES GPS/GSM**

Tesis para optar el Título de Ingeniero electrónico, que presentan los bachilleres:

Javier César Meza Romero

Victor Gabriel Leño Pariona

ASESOR: Dr. Manuel Augusto Yarlequé Medina

Lima, Octubre de 2017

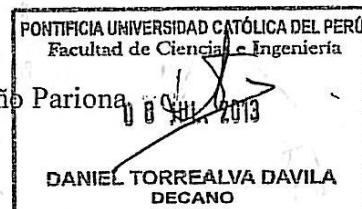
RESUMEN

Vivimos en una ciudad en donde la movilidad se ha convertido en un acto de violencia, en donde “orden”, en materia de transporte, no es más que una palabra que se repite una y otra vez sin que se haga algo al respecto. Se opera bajo un modelo urbanístico anticuado, donde el transporte privado se impone sobre el público, y donde la calle se convierte en un lugar violento, desprovisto de incentivos para que la gente camine y disfrute de su entorno. Usualmente la solución al tráfico y congestión es la de ampliar la oferta. Esto se traduce en la ampliación de avenidas, construcción de ‘bypass’ y nuevas vías expresas en zonas urbanas que no representan una solución sostenible. En lugar de ampliar la oferta vial, una alternativa es el modelo de ciudad que controla la demanda de viajes en vehículo privado y promueve la demanda del transporte sostenible. Este modelo representa una tendencia global en el urbanismo y ha sido exitoso en otras ciudades latinoamericanas como Bogotá o Curitiba. [1]

Por tanto el presente trabajo de tesis aborda el desarrollo de un sistema de monitoreo de una red de buses de transporte público integral, mediante el empleo de tecnologías modernas de comunicación y posicionamiento, con un sistema de procesamiento de información que permita al administrador del sistema la regulación de los flujos de tránsito, controlar las unidades de transporte, disminuir el riesgo de accidentes y ofrecer una mejor calidad de vida a los ciudadanos. El sistema de monitoreo se desarrolla principalmente para ser implementado en la flota de las empresas de transporte público, teniendo como finalidad apoyar a la creación de un sistema inteligente de transporte para nuestra capital. El empleo de transceptores GPS/GSM nos permitirá obtener datos de posición (latitud, longitud), fecha, hora y velocidad de desplazamiento del móvil monitoreado. Esta información será enviada a una central en la que será procesada junto a la data recopilada de otras fuentes. Una vez procesada la información, el administrador podrá tomar acciones correctivas necesarias, como aumentar o disminuir la frecuencia de los buses, controlar el exceso de velocidad, informar a los choferes de potenciales peligros y rutas alternativas, proveer la información de tiempo y distancia de llegada a los usuarios a través de mensajes de texto o aplicación para Smartphone. Se muestra las potencialidades del sistema mediante el desarrollo de un sistema de visualización que provea información de horas de arribo a los usuarios. Este sistema de monitoreo puede ser integrado a los sistemas actuales como el Metropolitano y el Metro de Lima, unificando así los mecanismos de transporte y lograr el tan ansiado Sistema Integrado de Transporte.

TEMA DE TESIS PARA OPTAR EL TÍTULO DE INGENIERO ELECTRÓNICO

Título : Sistema de monitoreo de una red de buses de transporte público e información para usuarios empleando transeceptores GPS/GSM.
Área : Comunicaciones # 1139
Asesor : Manuel Augusto Yarleque Medina
Alumno : Javier César Meza Romero / Victor Leañó Pariona
Código : 20072116 / 20074045
Fecha : 10/04/2013



Descripción y Objetivos

Actualmente Lima concentra el 68% del parque automotor del país y sus niveles de congestión son cada vez más preocupantes. Es por eso que las últimas gestiones municipales han enfocado gran parte de su presupuesto en implementar infraestructuras viales pero no contemplan alternativas de ordenamiento para la circulación del transporte público ni privado empleando recursos electrónicos. Por ende, el impulsar la formalización y modernización de las empresas de transporte público integrados a un sistema inteligente de transporte representa un gran aporte para una solución deseable a largo plazo.

Por tanto, se buscará gestionar la información proveniente de las unidades de transporte, bajo señales GSM, que contendrán datos como posición, velocidad, cantidad de pasajeros, entre otros. Estos datos serán procesados mediante un software que será desarrollado en una plataforma adecuada para el tranceptor, el cual brindará el tiempo estimado de llegada del bus a los paraderos, el tiempo de una unidad respecto a la otra y otros datos más que podrán ser empleados por la empresa de transporte y permitirá el acceso de los usuarios a esta información para que éste pueda decidir, estimar y ordenar su ruta.

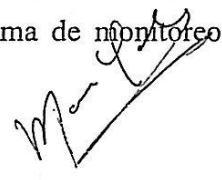
El Objetivo de la presente tesis es el desarrollo e implementación de un sistema de monitoreo para una red de buses de transporte público y la gestión de dicha información por software.

Los objetivos parciales de la tesis son: diseño del hardware de monitoreo basado en transeceptores GPS/GSM. Desarrollo del software de monitoreo de buses y procesamiento de información. Diseño e implementación de un módulo digital ubicado en los paraderos para la obtención de información del estado de ruta. Implementación del sistema de monitoreo y control de buses.



PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
SECCIÓN ELECTRICIDAD Y ELECTRÓNICA

Dr. Ing. BENJAMÍN CASTAÑEDA APHAN
Coordinador de la Especialidad de Ingeniería Electrónica



MÁXIMO 80 PÁGINAS

DEDICATORIA

“A Marta Romero, mi madre, quien es el motor de todos mis logros, nada hubiese logrado sin tu aliento, tu amor y tus más de mil sacrificios. Te amo.

A Francisco Meza, mi padre, que es mi mayor ejemplo de persona, cuyo trabajo diario e incansable me han permitido tener una carrera profesional.

Ojalá algún día pueda tener tu gran corazón, mi querido padre. A mis tíos, Romero Mayta, quienes estuvieron siempre a mi lado con sus consejos. Son mi más grande orgullo.”

Javier Cesar Meza Romero

“A Patricia Pariona, la mujer más importante en mi vida, quien me ha demostrado que la riqueza más grande no se guarda en el banco sino en el alma.

A Julián Leño, el hombre que me ha demostrado que se puede tener principios en una época carente de ellos. Más que un padre, un ejemplo de vida.

A Miriam, Rosana, Virginia, Carlos y Lucho: mis hermanos, mis ejemplos. A Adriana, Flor de María, Martín, Harold, Luis, Yamil, Carlos y Diego: casi unos hijos para mí.”

Victor Gabriel Leño Pariona

AGRADECIMIENTOS

A nuestros padres y familia en general, por su apoyo incondicional, aliento perenne y por su fe absoluta en nosotros. Nada de esto hubiese sido posible sin su cariño incondicional y su confianza depositada. Mil veces gracias.

Al Dr. Manuel Yarlequé Medina, por su apoyo constante, su disposición durante todo el desarrollo de la tesis, paciencia en escucharnos y su aporte con consejos de gran relevancia que potenciaron la estructura sistemática del presente trabajo.

A nuestros maestros y amigos de universidad, quienes estuvieron y estarán en todo nuestro trayecto profesional, en especial, a Natalia, una hermana, cuya amistad trascenderá las aulas de clase, y a nuestro gran amigo Julio, por su apoyo en la etapa final de este trabajo.

Y a todos aquellos que fueron parte directa o indirecta en nuestra carrera.

Muchas Gracias a todos.

ÍNDICE

INTRODUCCIÓN.....	1
CAPÍTULO 1	2
DESCRIPCIÓN DEL ESTADO ACTUAL	2
1.1. Marco Problemático	2
1.2. Municipalidad Metropolitana de Lima (MML) y las ETU.....	4
1.3. Contaminación ambiental, salubridad e inseguridad vial.	5
1.4. Iniciativa de la MML: reforma al transporte.....	6
1.5. Propuesta y objetivos	7
CAPÍTULO 2	9
BASE TECNOLÓGICA DEL SISTEMA DE MONITOREO DE TRANSPORTE	9
2.1. Tecnologías de Comunicación Móvil	9
2.1.1 Comunicaciones móviles	9
2.1.2. GSM.....	10
2.1.3. GPRS	12
2.2. Sistemas de posicionamiento global	12
2.3. Lenguaje de programación JAVA	14
2.4. Comandos AT	15
2.5. Base de datos MySQL	15
2.6. Transceptores de rastreo GPS	16
2.7. Sistemas de Transporte Inteligente	16
2.8. Algoritmo KNN (K – nearest neighbors).....	21
CAPÍTULO 3	23
PLANTEAMIENTO PARA EL DESARROLLO DEL HARDWARE Y SOFTWARE DE MONITOREO E INTERFAZ DE USUARIO	23
3.1. Diseño e implementación del Hardware	23
3.1.1. Requerimientos del Hardware.....	27
3.1.2. Selección de los componentes del hardware de monitoreo.....	29
3.1.3. Diseño del sistema de visualización de datos para el paradero.....	33
3.1.4. Programación de los transceptores GPS/GSM.....	38
3.1.5. Configuración de modos de operación	39

3.1.5.1.	Modo comunicación serial.....	40
3.1.5.2.	Modo lectura y envío de mensajes de texto.....	42
3.1.5.3.	Modo recepción de coordenadas GPS.....	44
3.2.	Diseño e implementación del software.....	44
3.2.1.	Requerimientos de algoritmo de monitoreo.....	44
3.2.2.	Módulos y elementos del sistema de control.....	45
3.2.3.	Módulo de comunicaciones.....	45
3.2.4.	Diseño de Software.....	46
3.2.5.	Programación en Visual Basic.....	47
3.2.5.1.	Base de datos.....	48
3.2.5.2.	Visualización de mapa.....	49
3.2.5.3.	Algoritmo de estimación.....	49
3.2.5.4.	Ajuste de los datos de predicción.....	51
3.2.5.6.	Interfaz de usuario.....	59
3.3.	Integración del hardware y del software del sistema de monitoreo.....	60
3.3.1.	Protocolo de comunicación.....	60
CAPÍTULO 4.....		62
PRUEBAS Y RESULTADOS.....		62
4.1.	Descripción de entornos para desarrollo de pruebas.....	62
4.2.	Pruebas por módulos.....	64
4.2.1.	Módulo de envío y recepción de mensajes.....	64
4.2.2.	Módulo de GPS.....	67
4.3.	Pruebas y resultados del sistema.....	68
4.3.1.	Pruebas del paradero.....	68
4.3.2.	Pruebas del envío de coordenadas geodésicas del bus.....	70
4.3.3.	Resultados iniciales del sistema.....	72
4.3.4.	Resultados finales del sistema.....	73
4.4.	Evaluación de proyecto de inversión.....	80
4.4.1.	VAN y TIR.....	80
4.4.2.	Estado financiero de flujo de efectivo.....	84
4.5.	Conclusiones y recomendaciones.....	91
BIBLIOGRAFIA.....		93

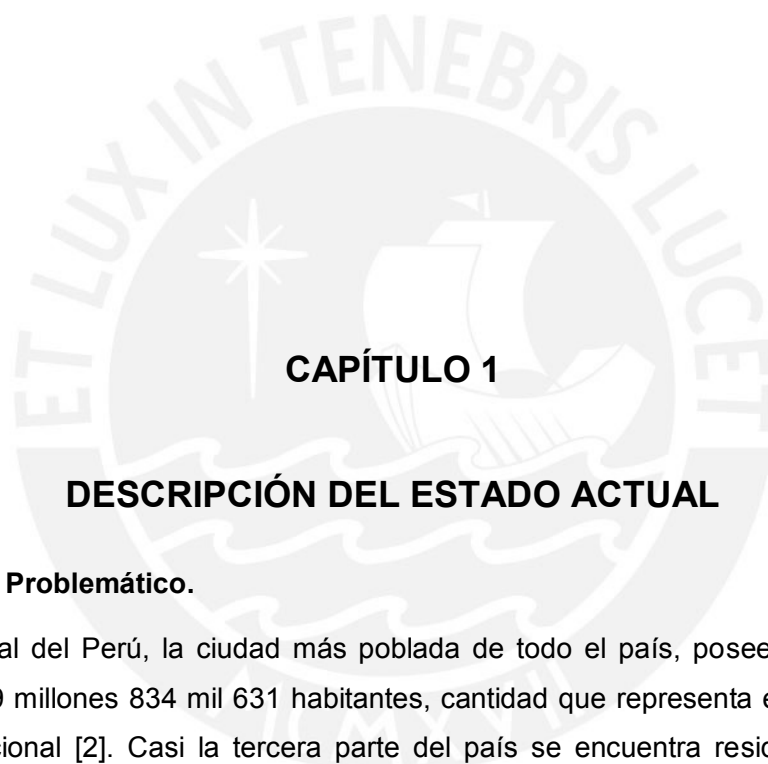
INTRODUCCIÓN

Según datos de la ONU, para 2030 dos tercios de la población mundial vivirá inmersa en ciudades. Bajo este panorama convenimos en que el transporte es un elemento básico de la organización de los asentamientos urbanos; las actividades económicas y productivas, gestión de suministros básicos, atención médica, abastecimiento de servicios públicos, capital humano, entre otros, todos ellos derivados del común de la vida urbana, conforman operaciones cruciales del día a día, las cuales no podrían ser articuladas sin la gestión necesaria de un medio de transporte urbano.

La ciudad de Lima, con una población próxima a los 10 millones de habitantes [2], está sufriendo grandes cambios ya que la ciudad crece año a año a un ritmo vertiginoso; bajo esta coyuntura es que nos preguntamos: ¿realmente es efectivo un sistema de transporte público como el que conocemos para una metrópoli que crece a un ritmo tan acelerado? es claro que el déficit de implementación tecnológica que presenta la Municipalidad, e incluso el MTC, sea un obstáculo para el desarrollo de la ciudad, y que la única solución que encuentran es construir corredores viales, elementos fundamentales de las reformas de transporte urbano, que si bien son una solución válida, no ofrecen una opción perdedera y que funcione a todo nivel en el marco problemático que tratamos de abordar. [3]

Por tanto, es indispensable el desarrollo de un sistema de monitoreo para el transporte público, mediante el empleo de tecnologías modernas de comunicación y posicionamiento; asimismo, es indispensable desarrollar el software de procesamiento de información para la central, para así regular los flujos de tránsito, controlar las unidades de transporte, disminuir el riesgo de accidentes y ofrecer una mejor calidad de vida a los ciudadanos.

Es por ello que el presente trabajo de tesis plantea una solución sistemática y articulada en el manejo de herramientas tecnológicas para monitorear los flujos, velocidad, orden y posición global de las unidades de transporte, procesados por un algoritmo desarrollado con herramientas informáticas. Esta solución permite al administrador del sistema adoptar medidas correctivas a tiempo real que aseguren una gestión inteligente del transporte y de esta forma mejorar la calidad de vida de las ciudadanos.



CAPÍTULO 1

DESCRIPCIÓN DEL ESTADO ACTUAL

1.1. Marco Problemático.

Lima es capital del Perú, la ciudad más poblada de todo el país, posee una población estimada de 9 millones 834 mil 631 habitantes, cantidad que representa el 31.57% de la población nacional [2]. Casi la tercera parte del país se encuentra residiendo en Lima Metropolitana, la densidad poblacional de Lima asciende a los 269.1 habitantes por kilómetro cuadrado que comparado con las densidades de 1993 y 2010, 186.2 y 252.1 hab/km² (presentes en la tabla 1.1) [4], da como resultado un incremento considerable; a esto hay que sumar la inserción de distritos que en los años 60 aún no existían. La población de Lima, de acuerdo al incremento tendencial de la misma, pronostica una población de aproximadamente 11 millones de habitantes para el 2022, como se puede visualizar en la tabla 1.2. Este crecimiento significativo de la población presenta grandes retos en lo que a transporte demanda, puesto que en un escenario no muy lejano convergerá en graves problemas viales que pueden solucionarse si se adoptan medidas correctivas tempranas.

Tabla 1.1. Tabla de densidad poblacional de Lima entre 1961 y 2015 según INEI [Elaboración propia], [4]

Año	Densidad (hab./km²)
1961	60,6
1981	143,5
1993	186,2
2010	252,1
2015	269,1

Tabla 1.2. Población de Lima entre el año 2000 y 2015 según INEI [Elaboración propia], [2]

Año	Población (habitantes)
2000	7,767,873.00
2001	7,913,690.00
2002	8,057,558.00
2003	8,199,172.00
2004	8,338,208.00
2005	8,474,342.00
2006	8,605,145.00
2007	8,730,820.00
2008	8,855,022.00
2009	8,981,440.00
2010	9,113,684.00
2011	9,252,401.00
2012	9,395,149.00
2013	9,540,996.00
2014	9,685,490.00
2015	9,834,631.00

Las metrópolis más grandes del mundo, en vista de la gran congestión de habitantes que presentan, han adoptado por un enfoque diferente al convencional, ya que la solución para el transporte urbano no se encuentra en la construcción masiva de arterias viales sino en un sistema integrado de transporte que se soporte en recursos tecnológicos y manejo de la información, que permita un flujo ordenado y eficiente. Países asiáticos como Japón lideran este tipo de tecnologías aplicadas, que en el tiempo le ha permitido ahorrar significativas sumas de dinero. Emplean sistemas que procesan información mediante programas informáticos complejísimos de monitoreo y control, empleo de transceivers (transceptores), sistemas de posicionamiento global, entre otros [5].

Por tanto es imperioso el estudio de soluciones urbanas para el transporte en nuestro país, es en ese contexto que este documento aborda dicho asunto de estudio, elaborando una solución sencilla a través del desarrollo de un sistema de monitoreo de unidades de transporte público basado en el manejo de información proveniente de las unidades, bajo el empleo de señales GSM/GPS. Estas señales transportan información como posición, velocidad, tiempo de aproximación, entre otros. Estos datos serán procesados mediante un software que ha sido desarrollado en una plataforma adecuada para el transceiver, puesto que cada modelo de éste posee un lenguaje y protocolo propio del integrado, el cual brindará el tiempo estimado de llegada a los paraderos, el de una unidad respecto a la otra y otros datos más que podrán ser empleados por la empresa de transporte en función y permitirá el acceso de los usuarios a esta información para que éste pueda decidir, estimar y ordenar su ruta.

1.2. Municipalidad Metropolitana de Lima (MML) y las ETU.

La relación entre la Municipalidad de Lima y las empresas de transporte urbano ha sido una constante de marchas, paros y acciones nada plausibles por parte de los transportistas; no obstante, la MML también ha jugado un rol en contra. A lo largo de más de 20 años, la Municipalidad no ejerció una labor coordinada o responsable frente a un problema que poco a poco se iba convirtiendo en un caos. Las limitadas coordinaciones con el Ministerio de Transportes y Comunicaciones (MTC), quien también fue un agente ajeno, incrementaron aún más el problema, puesto que al no poseer normas claras sobre el transporte, creció como pudo. Largas décadas sumidas en el terrorismo, volatilidad de la moneda (inflación), deudas internacionales y gobiernos corruptos, formaron parte de una receta, sin duda exitosa, para sumir al país en el socavón más profundo de Latinoamérica. Bajo estas circunstancias, gran cantidad de emprendedores, limeños y provincianos, vieron en el

transporte un negocio emergente, el cual, al no tener normas claras, permitió el crecimiento de este rubro bajo cánones netamente lucrativos, dejando de lado las condiciones de trabajo y las responsabilidades que esto implica.

Por tanto, es claro que, frente a una reforma, la mayoría de empresarios que ha trabajado bajo el enfoque de “arrendamiento de rutas” se oponga a la reforma, ya que sus utilidades se verían significativamente afectadas. Esto ha conllevado al sinnúmero de paros que las ETU han promovido a lo largo de estos años, puesto que saben el poder que tienen.

1.3. Contaminación ambiental, salubridad e inseguridad vial.

La contaminación del aire limeño es crítica, se ha excedido en más del 30% en los niveles de calidad permitido, pese a que en los últimos años el grado de contaminación ha disminuido. Esto ha ocasionado que más de 800 mil personas hayan sufrido o sufran, de una afección respiratoria. Los distritos más contaminados son los del centro de Lima; el Cercado por ejemplo, en el año 2010 registró niveles de 65 ug/m³ (microgramo por metro cúbico) según el Estándar Nacional de Calidad de Aire, cuyo límite establecido es de 50 ug/m³ [6]. Es en estos distritos que hay mayor concentración de flujo vehicular de transporte público, puesto que al frecuentar constantemente estas rutas, se genera congestión, las unidades de transporte se ven obligadas a reducir la velocidad, generando mayor emisión de partículas contaminantes; esto podría reducirse si se optimizara la frecuencia de las unidades y se modernizara las flotas que transcurren por esas zonas, ya que la mayoría de los buses en estos distritos supera los 20 años. Este problema no puede considerarse irrelevante puesto que según un estudio realizado en el año 2006 por la Comisión Nacional del Medio Ambiente, cerca de 6 mil ciudadanos fallecen en Lima por efecto de estos gases [7].

Por otra parte, en cuanto a los accidentes de tránsito, el Perú es el país con mayores accidentes de tránsito en toda América Latina, y posee uno de los mayores índices de accidentes a nivel mundial. La ciudad de Lima tiene una flota de transporte público de aproximada 42 mil unidades vehiculares, entre formales e informales, que genera una sobreoferta en el servicio según la GTU [8]. Las unidades de transporte no tienen un control adecuado ni están sujetas a regulaciones, ya que cada una simplemente es usuaria de la ruta que cubre, valiéndose en sí por intereses propios. Por tanto, al ser cada unidad indistinta y no directamente relacionada con otras de la misma empresa, trata de acaparar la mayor cantidad de clientes potenciales (pasajeros) haciendo competencia con la otra unidad. A estas carreras desenfundadas por la obtención de pasajeros se le ha llamado

coloquialmente como “correteo”, convirtiéndose esta famosa guerra del centavo en uno de los principales motivos de accidentes de tránsito. Esto sumado a la creciente informalidad en el manejo empresarial del transporte, entre otros, ha desembocado en alarmantes cifras, como las obtenidas en el 2002, donde se registraron más de 36 mil 800 accidentes de tránsito tan solo en lima de los cuales el 35% fue ocasionado por el Transporte Público [9]. De acuerdo a un estudio realizado por el MTC en el 2008, se determinó que el Perú, comparado con otros países de Europa y Asia (Figura 1.1), arrojaba una tasa elevada de mortalidad en comparación con el parque automotor del país.

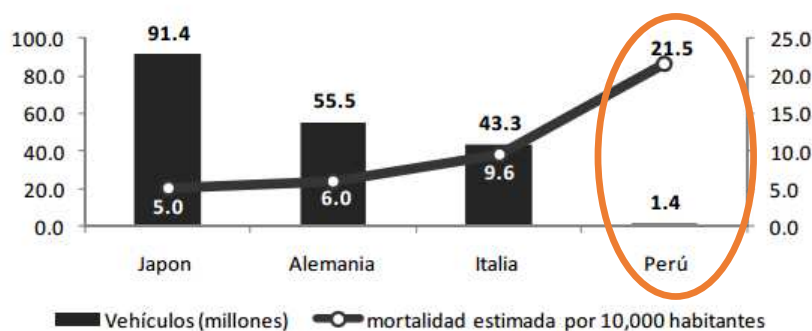


Figura 1.1. Tamaño del parque automotor y mortalidad estimada. Comparación con países seleccionados [10]

Asimismo, estos accidentes, más allá de las pérdidas humanas, también ocasionan gastos increíbles a la ciudad de Lima: más de 500 millones de dólares anuales [10].

1.4. Iniciativa de la MML: reforma al transporte.

El sistema con el que se manejan las empresas de transporte no exige a los choferes mayores responsabilidades que las de juntar el dinero para realizar el pago diario. Por tanto, toda falta grave que pueda cometer una unidad, no es asumida por la empresa, ya que para ellos, dicha falta está exenta de su responsabilidad. Es por ello que la Municipalidad de Lima ha emprendido una Reforma al Transporte, con la cual, las empresas deberán responsabilizarse de todas sus unidades, ofreciendo también los correspondientes beneficios laborales que la ley laboral demanda. No obstante, este cambio radical implicaría una significativa disminución sobre las utilidades actuales que las empresa de transportes perciben por alquiler de ruta, esto por tanto, es el principal motivo por el que los

transportistas se oponen a la mencionada reforma. No obstante es necesaria la planteada reforma, puesto que ello también demandará mayor control sobre cada una de sus unidades, parte en la cual el presente trabajo de tesis plantea una solución tecnológica para el monitoreo de las mismas, logrando así una mayor eficiencia en la gestión de los buses, permitiendo de esta manera un sistema ordenado y eficiente.

1.5. Propuesta y objetivos

Propuesta

La propuesta del presente proyecto de tesis fue desarrollar un sistema de monitoreo para unidades de transporte público mediante el empleo de tecnologías modernas de comunicación y posicionamiento, con un sistema de procesamiento de información para así monitorear los flujos de tránsito, disminuir el riesgo de accidentes y ofrecer una mejor calidad de vida a los ciudadanos. El sistema de monitoreo se desarrolló principalmente para ser implementado en las flotas de las empresas de transporte público, teniendo como finalidad aportar a la creación de un sistema inteligente de transporte para nuestra capital al instalar trancceptores GPS/GSM en los buses, los cuales nos permitirán obtener datos de posición (latitud, longitud), fecha, hora y velocidad de desplazamiento del móvil monitoreado; la información es enviada a una central en la que es procesada junto a la data recopilada de otras fuentes. Una vez procesada la información, el operador podrá tomar las acciones correctivas necesarias, ya sea aumentar o disminuir la frecuencia de los buses, controlar el exceso de velocidad, informar a los choferes de potenciales peligros y rutas alternativas, proveer la información de tiempo y distancia de llegada a los usuarios a través de mensajes de texto o aplicación para smartphone.

Objetivos

- Desarrollar un estudio comparativo de las tecnologías de comunicación entre una unidad móvil, una central y un paradero.
- Demostrar mediante la implementación del sistema que es posible la realización de un sistema de monitoreo.
- Verificar la recepción de información en el paradero de prueba.
- Demostrar que un sistema de estimación de tiempo de aproximación de un bus al paradero es realizable con un margen de error menor al
- Validar la funcionalidad de la integración del Hardware y software de monitoreo.

- Verificar la rentabilidad de un proyecto de implementación tecnológica en el sector Transporte.





CAPÍTULO 2

BASE TECNOLÓGICA DEL SISTEMA DE MONITOREO DE TRANSPORTE

En el presente capítulo se analizarán las bases tecnológicas del sistema de monitoreo planteado en el capítulo anterior, con el fin de implementar un prototipo funcional adecuado para su integración en nuestro sistema actual de transporte público.

2.1. Tecnologías de Comunicación Móvil

2.1.1 Comunicaciones móviles

Existen dos formas de comunicaciones móviles: inalámbrica y celular.

- **Comunicación inalámbrica:** El radio de acción de esta tecnología es muy limitado, de hecho los equipos móviles y los de transmisión-recepción deben estar situados en zonas geográficas muy cercanas, como por ejemplo, dentro de un mismo edificio. Sin embargo actualmente se vienen desarrollando sistemas VANET (Redes Vehiculares Ad hoc) las cuales implementan comunicación inalámbrica en vehículos y permiten tener nodos móviles ideales para la implementación de sistemas SIT (Sistemas Inteligentes de Transporte).

- **Comunicación celular:** Tiene una red totalmente definida que incluye protocolos para establecer y despejar llamadas, bajo nivel de ruido, algoritmos de corrección de errores, rastreo de las unidades móviles dentro de áreas geográficas definidas (células), siendo el tamaño de estas del orden de kilómetros. Es por esto que realizamos el prototipo funcional del sistema de monitoreo de transporte teniendo como base los sistemas de comunicación celular, debido a que se tienen redes privadas de telefonía celular ya instaladas en nuestro entorno, lo cual permite poder implementar el prototipo de una forma rápida, eficiente y a un bajo costo, con el fin de mostrar el alcance y la importancia de los sistemas inteligentes de transporte público.

2.1.2. GSM

Es un sistema de comunicación móvil digital de 2ª generación que se basa en células de radio. Su diseño estuvo enfocado para la transmisión de voz por lo que se basa en la conmutación de circuitos, por lo que los recursos quedan ocupados durante la comunicación y la tarificación es por tiempo de conexión [11].

Arquitectura de una red GSM.

Mobile Station (MS): Consta de dos elementos básicos que debemos conocer. Por un lado el terminal o equipo móvil y por otro lado el SIM (Subscriber Identity Module). El SIM es una pequeña tarjeta inteligente que identifica las características de nuestro terminal. Esta tarjeta se inserta en el interior del móvil y permite al usuario acceder a todos los servicios que haya disponibles por su operador. El SIM está protegido por un número de cuatro dígitos que recibe el nombre de PIN o Personal Identification Number.

Base Station Subsystem (BSS): Sirve para conectar a las estaciones móviles con los NSS, además de ser los encargados de la transmisión y recepción. Constan de dos elementos diferenciados: La Base Transceiver Station (BTS) o Base Station y la Base Station Controller (BSC). La BTS consta de transceivers y antenas usadas en cada célula de la red y que suelen estar situadas en el centro de la célula, generalmente su potencia de transmisión determinan el tamaño de la célula. Los BSC se utilizan como controladores de los BTS y tienen como funciones principales las de estar al cargo de los handovers, los frequency hopping y los controles de las frecuencias de radio de los BTS [11].

Network and Switching Subsystem (NSS): Este sistema se encarga de administrar las comunicaciones que se realizan entre los diferentes usuarios de la red; para poder hacer

este trabajo la NSS se divide en siete sistemas diferentes, cada uno con una misión dentro de la red: MSC, GMSC, HLR, VLR, AuC, EIR, GIWU [11].

1. Mobile Services Switching Center (MSC): Es el componente central del NSS y se encarga de realizar las labores de conmutación dentro de la red, así como de proporcionar conexión con otras redes [11].
2. Gateway Mobile Services Switching Center (GMSC): Un Gateway es un dispositivo traductor (puede ser software o hardware, que se encarga de interconectar dos redes haciendo que los protocolos de comunicaciones que existen en ambas redes se entiendan bien. La misión del GMSC es esta misma, servir de mediador entre las redes de telefonía fijas y la red GSM [11].
3. Home Location Register (HLR): Es una base de datos que contiene información sobre los usuarios conectados a un determinado MSC. Entre la información que almacena el HLR tenemos fundamentalmente la localización del usuario y los servicios a los que tiene acceso. El HLR funciona en unión con el VLR que vemos a continuación [11].
4. Visitor Location Register (VLR): Contiene toda la información sobre un usuario necesaria para que dicho usuario acceda a los servicios de red. Forma parte del HLR con quien comparte funcionalidad [11].
5. Authentication Center (AuC): Proporciona los parámetros necesarios para la autenticación de usuarios dentro de la red; también se encarga de soportar funciones de encriptación [11].
6. Equipment Identity Register (EIR): También se utiliza para proporcionar seguridad en las redes GSM pero a nivel de equipos válidos. La EIR contiene una base de datos con todos los terminales que son válidos para ser usados en la red. Esta base de datos contiene los International Mobile Equipment Identity o IMEI de cada terminal, de manera que si un determinado móvil trata de hacer uso de la red y su IMEI no se encuentra localizado en la base de datos del EIR no puede hacer uso de la red [11].
7. GSM Interworking Unit (GIWU): Sirve como interfaz de comunicación entre diferentes redes para comunicación de datos.

Características de las redes GSM para la transmisión de datos:

- Velocidad de transferencia de 9,6 Kbps.

- Tiempo de establecimiento de conexión, de 15 a 30 segundos. Además las aplicaciones deben ser reinicializadas en cada sesión.
- Pago por tiempo de conexión.
- Problemas para mantener la conectividad en itinerancia (Roaming).

2.1.3. GPRS

GPRS (General Packet Radio Service) unifica el mundo IP con el mundo de la telefonía móvil, creándose toda una red paralela a la red GSM y orientada exclusivamente a la transmisión de datos. Al sistema GPRS se le conoce también como GSM-IP ya que usa la tecnología IP (Internet Protocol) para acceder directamente a los proveedores de contenidos de Internet. GPRS es una tecnología que comparte el rango de frecuencias de la red GSM utilizando una transmisión de datos por medio de 'paquetes'. La conmutación de paquetes es un procedimiento más adecuado para transmitir datos, hasta ahora los datos se habían transmitido mediante conmutación de circuitos, procedimiento más adecuado para la transmisión de voz [12].

La tecnología GPRS, o generación 2.5, representa un paso más hacia los sistemas inalámbricos de Tercera Generación o UMTS. Su principal base radica en la posibilidad de disponer de un terminal permanentemente conectado, tarifando únicamente por el volumen de datos transferidos (enviados y recibidos) y no por el tiempo de conexión como hemos podido observar en un punto anterior.

Características de GPRS:

- Velocidad de transferencia de hasta 144 Kbps.
- Conexión permanente. Tiempo de establecimiento de conexión inferior al segundo.
- Pago por cantidad de información transmitida, no por tiempo de conexión.

Estas características han permitido que se pueden utilizar servicios como Wireless Application Protocol (WAP), servicio de mensajes cortos (SMS), Multimedia Messaging System (MMS), Internet; y para los servicios de comunicación como el correo electrónico y el World Wide Web (WWW).

2.2. Sistemas de posicionamiento global

Es un sistema global de navegación por satélite que permite determinar en todo el mundo la posición de un objeto, una persona, un vehículo o una nave. Tiene tres partes

fundamentales: el espacial (satélites), el de control (estaciones terrestres) y los usuarios (receptores). Las 5 estaciones de tierra, detalladas en la tabla 2.1 están distribuidas a distancias similares alrededor de la línea Ecuatorial (Isla Ascensión, Diego García, Kwajalainy, Hawaii y Colorado Springs) y tienen como fin monitorear el estado de los satélites (altura, estado de los relojes atómicos), realizar pequeños ajustes en sus órbitas y calcular las efemérides (posición de los satélites). Esta información es transmitida a los satélites, los cuales a su vez la retransmiten a los receptores en tierra al menos una vez al día. El tercer componente es el usuario quien recibe las señales enviadas por los satélites mediante el uso de un receptor equipado con una antena.

El sistema NAVSTAR-GPS está formado por 27 satélites artificiales, 24 activos y 3 de respaldo, con órbitas circulares de 12 horas, una inclinación de 55° con respecto al Ecuador terrestre y ubicados a una distancia de 26,560 Km de la Tierra. Los cuales están desplegados en 6 planos orbitales con 4 satélites por órbita a una velocidad de desplazamiento aproximada de 4km/s. Cada uno de los satélites posee dos relojes atómicos, uno de Cesio y otro de Rubidio, que sirven para generar señales precisas y sincronizadas [13]

Tabla 2.1. Cuadro de puesta en funcionamiento de los Sistemas de posicionamiento en el mundo [Elaboración propia]

AÑO DE LANZAMIENTO	SISTEMA	ORGANIZACIÓN/PAIS	Nº DE SATELITES	CARACTERISTICAS
1965-1996	TRANSIT	MARINA DE ESTADOS UNIDOS	6	Dejó de operar un año después que el sistema GPS entró en funcionamiento
1978 1995(Capacidad total)	NAVSTAR-GPS	DEPARTAMENTO DE DEFENSA DE EEUU	27	P= 900kg Altura orbital =26560 km
1982 1996(capacidad total)	GLONASS	FEDERACION RUSA	24	P=1400kg(1ª generación) P= 700kg(2ª generación) Altura orbital =19100 km
2005 2011(operativo)	GALILEO	UNION EUROPEA	30	P= 700kg Altura orbital =23616 km
2015	BEIDOU-2	CHINA	14	Sistema de Navegación en fase de proyecto

P= Peso promedio de los satélites

2.3. Lenguaje de programación JAVA

Java incluye una serie de características distintas a los lenguajes de programación convencionales, por ejemplo, una de sus particularidades más importantes es que los programas ejecutables no dependen de la arquitectura de la computadora, por tanto se puede ejecutar en una gran variedad de equipos y dispositivos electrónicos. También permite la escritura de applets, programas que pueden ser insertados en una página HTML; así también se puede desarrollar aplicaciones para redes locales, unión de redes o internet. Al ser un lenguaje de alto nivel, y al estar orientada a objetos, provee gran facilidad al momento de emplearlo, lo que le provee gran ventaja frente a otros lenguajes tediosos y complicados de usar [14].

POO (Programación Orientada a Objetos)

La POO es una evolución de la programación clásica, que permite modelar elementos realmente complejos del programa en módulos independientes que podrán ser ejecutados de acuerdo a los requerimientos del programa. En esta programación se definen objetos con una serie de características y operaciones. Pero ¿Qué es un objeto?, en un ámbito ajeno a la programación, un objeto es cualquier elemento que pueda ser percibido a nuestro alrededor, que posee características y comportamientos. De forma análoga en informática los objetos son componentes que poseen roles característicos y que pueden comunicarse entre ellos, sus características están establecidas en variables y define su comportamiento bajo métodos. Por tanto cada objeto se define en función de la cantidad de características que posee y la cantidad de operaciones que pueden realizar sobre él [15].

Propiedades de un lenguaje orientado a objetos

- *Encapsulamiento*: consiste en tener en una sola unidad (clase) las características y comportamientos. Esto se logra gracias a la abstracción y el ocultamiento, que es una propiedad que permite ocultar atributos y métodos a otras partes del programa.
- *Herencia*: es considerada un elemento muy importante de la POO. Permite “heredar” ATRIBUTOS y MÉTODOS de una clase predecesora a otra descendiente (subclases), lo cual no afecta sus atributos y métodos propios.
- *Polimorfismo*: Permite que clases distintas que hayan recibido el mismo mensaje, se comporten de forma distinta.

2.4. Comandos AT

Los comandos AT fueron desarrollados en 1977 por Dennis Hayes como un interfaz de comunicación con un MODEM, con el avance del baudio, fueron las compañías Microcomm y US Robotics las que siguieron desarrollando y expandiendo el juego de comandos hasta universalizarlo. Los comandos AT se denominan así por la abreviatura de attention. Son instrucciones codificadas que conforman un lenguaje de comunicación entre el hombre y un Terminal MODEM.

Aunque la finalidad principal de los comandos AT es la comunicación con módems, la telefonía móvil GSM también ha adoptado como estándar este lenguaje para poder comunicarse con sus terminales. De esta forma, todos los teléfonos móviles GSM poseen un juego de comandos AT específico que sirve de interfaz para configurar y proporcionar instrucciones a los terminales, permitiendo así la interacción entre estos y dando lugar a las llamadas de datos o de voz, recibir y enviar SMS entre otros [16].

2.5. Base de datos MySQL

MySQL es uno de los sistemas de gestión de base de datos relacionales más usado, caracterizado por su simplicidad y notable rendimiento. Esa facilidad de uso es una opción atractiva para el desarrollo de aplicaciones, de forma versátil y ligera, ya que su libre distribución en Internet bajo licencia GPL (General Public License) le otorgan un alto grado de libertad y celeridad en el desarrollo [17].

Esta herramienta resulta atractiva debido a su numerosa serie de características:

- Está desarrollado en C/C++ por lo que se puede desarrollar de forma rápida debido a la reusabilidad de códigos.
- Crear y usar nuevos tipos de datos es más fácil que en otros lenguajes como
- Se distribuyen ejecutables para cerca de diecinueve plataformas diferentes.
- La API se encuentra disponible en C, C++, Eiffel, Java, Perl, PHP, Python, Ruby y TCL.
- Está optimizado para equipos de múltiples procesadores.
- Es muy destacable su velocidad de respuesta.
- Se puede utilizar como cliente-servidor o incrustado en aplicaciones.
- Cuenta con un rico conjunto de tipos de datos.
- Soporta múltiples métodos de almacenamiento de las tablas, con prestaciones y rendimiento diferentes para poder optimizar el SGBD a cada caso concreto.

- Su administración se basa en usuarios y privilegios.
- Se tiene constancia de casos en los que maneja cincuenta millones de registros, sesenta mil tablas y cinco millones de columnas.
- Sus opciones de conectividad abarcan TCP/IP, sockets UNIX y sockets NT, además de soportar completamente ODBC.
- Los mensajes de error pueden estar en español y hacer ordenaciones correctas con palabras acentuadas o con la letra 'ñ'.
- Es altamente confiable en cuanto a estabilidad se refiere.

2.6. Transceptores de rastreo GPS

Son dispositivos que integran un módulo receptor de señales GPS junto a un módulo GSM. Son la base tecnológica para los sistemas inteligentes de monitoreo vehicular, capaces de proporcionar datos en tiempo real como: posicionamiento de la unidad, velocidad, fecha y hora, comunicación de voz e incluso, en los módulos más avanzados y por supuesto con mayor costo, es posible transmitir y recibir imágenes. Actualmente existen empresas especializadas en tecnología de rastreo que fabrican estos dispositivos.

2.7. Sistemas de Transporte Inteligente

Un ITS (Intelligent Transport System) es un conjunto de soluciones tecnológicas que involucra a las Telecomunicaciones y la Informática (esta unión se conoce con el término de Telemática), diseñadas para mejorar la operación y seguridad del transporte terrestre.

ITS en Japón

Japón es uno de los países que más se ha desarrollado en el campo de las comunicaciones y tecnologías de la información; esto le ha permitido desarrollar un sistema integral de transporte que es aplicado a nivel nacional. La congestión vehicular origina una pérdida de tiempo estimada en 5.3 mil millones de horas al año que corresponde en un impacto negativo en la economía de ese país: pérdidas de hasta 12 trillones de yenes en la economía nacional [18]. Por otro lado, la tasa de mortalidad debida al tráfico vehicular llega a los diez mil ciudadanos por año, sin olvidarnos claro del terrible impacto sobre el medio ambiente. Muchos países alrededor del mundo han gastado millones de dólares por esta causa, pero es en Japón donde la aplicación comercial de las ITS es concretada. Existen 9 áreas que fueron desarrolladas en el ITS de Japón: Sistemas avanzados de navegación, recolección electrónica de peaje, asistencia para la conducción segura, optimización de la

gestión del tráfico, aumento de la eficiencia en la gestión vial, apoyo para el transporte público, aumento de la eficiencia en anuncios, soporte para peatones, apoyo para operaciones de emergencia.

En Japón, el ITS es un sistema integrado que incluye los VICS (Sistema de comunicación e información vehicular), ETC (recolección electrónica de peaje), AHS (sistema avanzado de vías rápidas) y UTMS (sistema universal de gestión de transporte) todos por sus siglas en inglés. El sistema de navegación emplea tecnología satelital, que conforma un bloque importantísimo en el desarrollo de los ITS, gracias a la red de satélites que orbitan la Tierra, los que, en su mayoría, le pertenecen a los Estados Unidos de América. La implementación de este sistema inteligente conllevó un trabajo sistemático y progresivo; la primera fase de esta etapa se inició aproximadamente en 1995, fue en estas épocas en que los sistemas de navegación fueron incluidos, fue durante este período que los sistemas de información de tráfico empezaron a emplearse. La segunda fase se inició alrededor de 2005, el “Ministerio de Tierra, Infraestructura y Transporte” decidió que se incluya información sobre el servicio y sobre los destinos del transporte público, lo que facilitaba de forma óptima la estimación en tiempos de la ruta en cuestión. En la tercera fase, alrededor del 2010, el MTIT manifestó que el aumento de la infraestructura y equipamiento a bordo de vehículos implementara el establecimiento de los ITS como un sistema social. Los sistemas legales y sociales participaron en esta marcha para que los efectos fuesen a nivel nacional. Ya para la cuarta fase, a partir del 2010 en adelante, todos los sistemas de los ITS estaban en funcionamiento, empleando para ello sistemas avanzados de comunicación digital, soportados en las redes nacionales de fibra óptica. En un futuro se espera que las ciudades más pobladas disminuyan la congestión vehicular, lo que permitirá aliviar el entorno y mejorar las condiciones de vida al disminuir la polución.

VICS (Figura 2.1) es un sistema de comunicación digital de datos que provee información reciente sobre estados de tránsito. El VICS center edita y procesa información acerca de los estados del tráfico, la cual será enviada posteriormente a los equipos de navegación de los vehículos; es en esta central en la que toda la información referente al tráfico es colectada, luego de ser procesadas son enviadas empleando rayos infrarrojos en las autopistas y ondas de radio en las pistas de la ciudad. La emisión de señales FM multiplexadas se realiza a través de ondas radiales [18]. El procesamiento de la información se puede describir en cuatro funciones imprescindibles: colección de información,

procesamiento y edición de la información, proveimiento de la información, empleo de la información

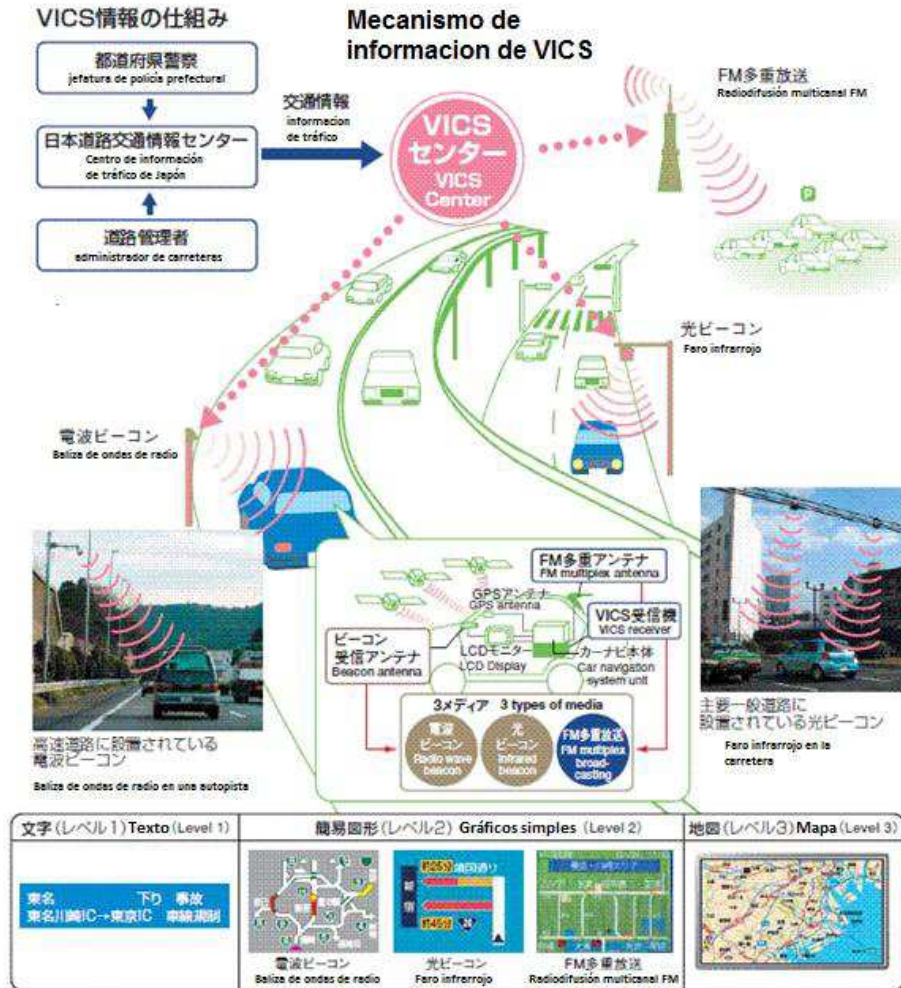


Figura 2.1. Mecanismos de información del VICS [19]

Sistema Integrado de Control de Tráfico (ITCS): El sistema de control de tráfico consta de tres funciones básicas: colección, procesamiento y presentación de la información del tráfico. Una red de sensores colocados alrededor de las pistas, proporciona la información del tráfico; por otro lado, la policía de tránsito, helicópteros y patrullas supervisan las calles, con lo cual proveen información adicional a la central (figura 2.2). El procesamiento de la información, se hace de la siguiente forma:

- Controlando los intervalos de la señal (rojo / verde) en relación al tráfico.
- Mostrando los atascos de tráfico en los paneles de la central (figura 2.3)

- Actualización permanente del servicio de información telefónica y de transmisores de noticias de tráfico.
- Viendo qué accidentes de tránsito ocurren y qué calles están cerradas.
- Intercambiando información sobre el tráfico con los Centros de Control de vecinos.



Figura 2.2. Central de Control de Tráfico de Tokio [20]



Figura 2.3. Paneles de Central de Control de Tráfico [20]

La presentación de la información del tráfico se realiza mediante los siguientes dispositivos: tableros eléctricos de anuncios (información de caracteres, mensaje actual, tiempo de viaje); señales de tránsito; transmisor Traffic News (transmisores de radio en carretera: los conductores pueden acceder a esta información sobre la radio de su auto a 1620kHz); servicio de información del tráfico a través del teléfono y fax; terminales de información que se instalan en los centros públicos para la comodidad del usuario; emisión radial; e información visualizada en automóviles.

ITS en Brasil

Brasil es el país más grande de Latinoamérica con una población de 200 millones de personas, todos los días se enfrenta a un problema vital: el transporte. Mucho se puede decir de los problemas que presentan los diferentes estados del país para movilizar a sus ciudadanos, no obstante, un escenario es distinto y presenta un flujo de transporte eficiente que ha sido no solo un modelo ejemplo para todo el país sino también para el mundo, este es el escenario que se presenta en Curitiba.

Curitiba, capital del estado de Paraná, al sur de Brasil, residen casi 1.800.000 personas y por día circulan 3.300.000, contando a las personas que ingresan desde las afueras. Esta ciudad vio nacer allá por la década de los 70 un sistema efectivo en referencia al transporte

público, un sistema que se desarrolla en base a una red integrada de transporte (figura 2.4) que fue diseñado por el arquitecto y urbanista Jaime Lerner. Este es un sistema trinario, que tiene una vía central, en la cual se encuentran las canaletas exclusivas para buses, con dos vías lentas a los costados, para otros vehículos. Hacia cada lado de esta vía central, hay dos avenidas paralelas de tránsito rápido, una de entrada y otra de salida de la ciudad, por donde circula el resto, mayormente autos particulares. Para permitir el desplazamiento de los más de dos millones de pasajeros que el sistema transporta a diario, las seis líneas rápidas de biarticulados (azul, rojo, naranja, gris, verde y amarillo) operan en forma coordinada con colectivos comunes, estos tienen diferentes funciones: los alimentadores conectan las 22 terminales de integración con los barrios de la zona; los ómnibus verdes, los barrios con las terminales, sin pasar por el centro; los convencionales, los barrios con el centro, sin integración, y los troncales, las terminales con el centro, utilizando vías compartidas. Una sola línea con tarifa diferenciada realiza los viajes cortos por el centro, para aquellas personas que no deseen caminar. Para el resto de los colectivos se paga un boleto único de 1,90 reales (\$0.33 aproximadamente), costo por el cual se puede viajar por toda la ciudad haciendo las distintas combinaciones. Los estudiantes sólo pagan la mitad de la tarifa, los discapacitados abonan un porcentaje y la gente mayor viaja gratis [21].

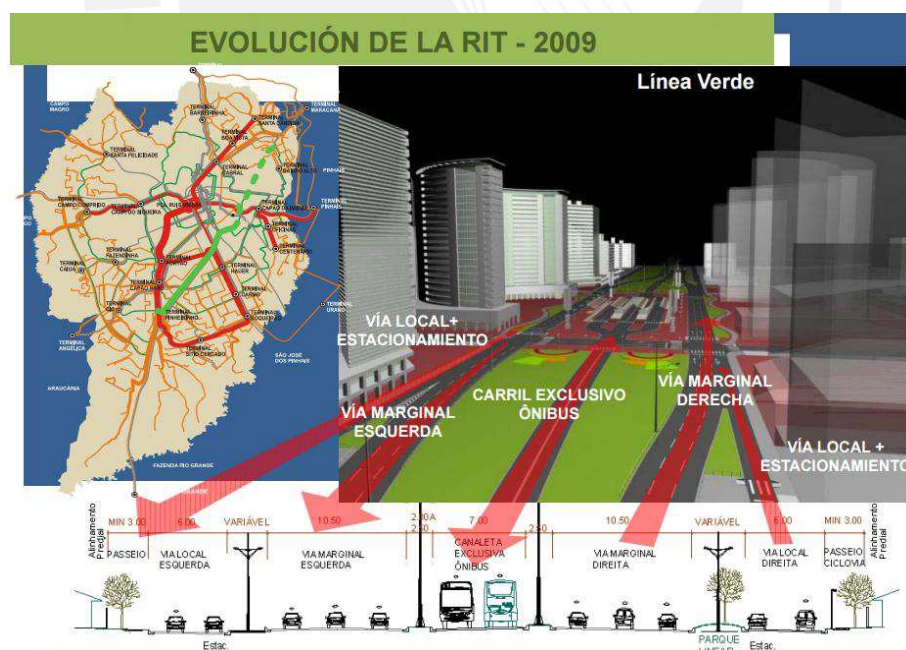


Figura 2.4. Red Integral de Transporte [22]

Este sistema es funcional puesto que los tiempos promedios utilizados en los desplazamientos casa - trabajo realizados en la región metropolitana de Curitiba, según el estudio “Indicadores de Movilidad Urbana de la PNAD 2012”, elaborado por el IPEA, variaron de 30,2 minutos para 32,0 minutos entre los años de 1992 y 2012 [22].

2.8. Algoritmo KNN (K – nearest neighbors)

El Algoritmo KNN es un método no paramétrico tanto de clasificación como de estimación que se basa en el aprendizaje por casos. Este algoritmo permite estimar el valor de un objeto tomando como base su proximidad a los datos más frecuentes en un radio de acción dentro de una gráfica de dispersión (Figura 2.5), es decir, que este método se basa en la suposición de que los prototipos más cercanos al punto de observación tienen una probabilidad más alta de poseer características similares con el dato observado [23]; por ejemplo, si tenemos un conjunto de datos que poseen las características “a” y “b”, y un objeto “X” que posee también las características “a” y “b”, podemos suponer que dicho objeto poseerá un comportamiento similar a los objetos del conjunto de datos.

En resumen, este algoritmo se basa en la búsqueda en un conjunto de prototipos de los K datos más cercanos al patrón a clasificar. Para este fin se determina la distancia de todos los miembros de la base de datos y se toma un número particular de vecinos (K) que son los que tuvieron distancias menores (vecinos más cercanos) del nuevo punto de observación hacia los demás puntos observados. Con los K – vecinos más cercanos se establece que el nuevo punto de observación tendrá mayor probabilidad de poseer características similares a aquellos datos que preponderan en mayor cantidad dentro del total de datos analizados, es decir, que si existen dos características, “n” y “m” y hay mayor cantidad “n” de datos cercanos al punto de observación, existe una alta probabilidad de que este nuevo valor posea la característica “n”.

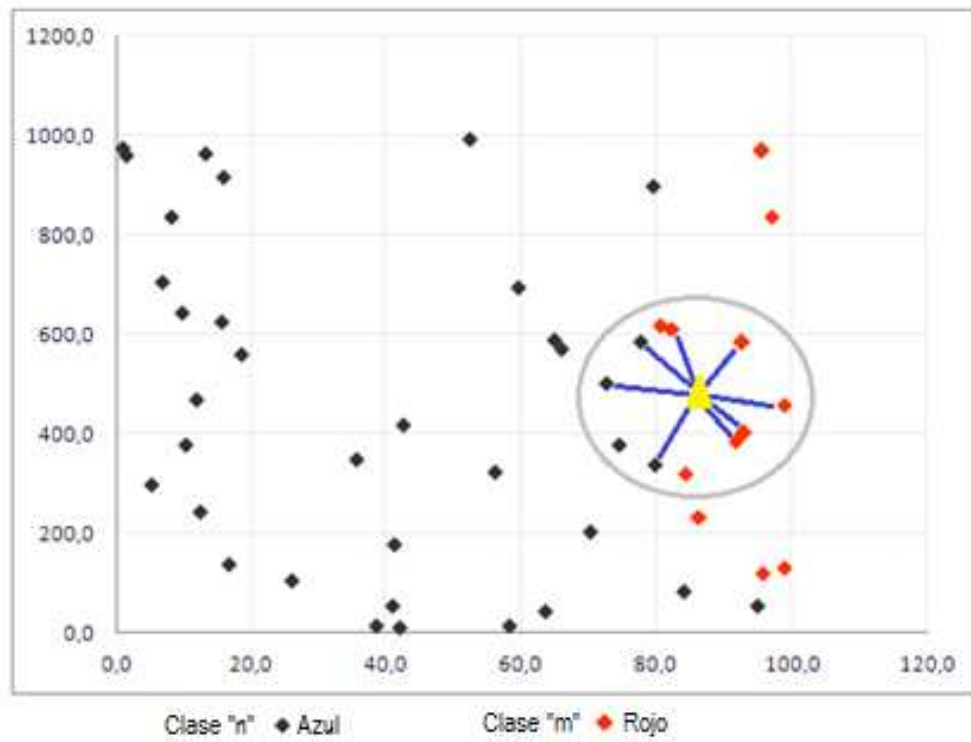


Figura 2.5. Grafica de dispersión de datos
[Elaboración Propia]



CAPÍTULO 3

PLANTEAMIENTO PARA EL DESARROLLO DEL HARDWARE Y SOFTWARE DE MONITOREO E INTERFAZ DE USUARIO

3.1. Diseño e implementación del Hardware

El diseño del hardware de nuestro sistema de monitoreo está enfocado principalmente para ser implementado en una empresa de transporte público como punto de partida en la creación de un sistema inteligente de transporte para nuestra capital.

Este sistema consta de las siguientes partes: un paradero que permite la visualización de datos mediante una pantalla LCD y un receptor GSM/GPRS, el cual recibirá información procesada por la central de monitoreo mediante mensajes SMS; el hardware de monitoreo, instalado en la unidad de transporte público, compuesto principalmente por un transceptor GPS/GSM-GPRS, el cual envía mediante SMS datos de posición (latitud, longitud), fecha, hora y velocidad de desplazamiento del móvil hacia la central de monitoreo, la cual esta implementada con el software de procesamiento que hemos desarrollado y un transceptor GSM/GPRS que le permite comunicarse con el bus y con el paradero.

Cabe precisar que no todas las partes del hardware necesitan tener un transceptor con GPS y GSM/GPRS a la vez; sin embargo en la etapa de diseño del prototipo hemos escogido un mismo equipo transceptor GPS/GSM-GPRS para cada módulo del hardware, cuya función deberá ser configurada de acuerdo a la etapa en la que opera, se implementó de esta forma para evitar tener modelos distintos de equipos en el sistema y así evitar un mayor costo de desarrollo.

Para demostrar la funcionalidad del sistema de monitoreo se diseñó un piloto cuyos componentes han sido seleccionados de tal forma que la arquitectura implementada (Figura 3.1) no tenga un alto costo pero que permita demostrar el alcance y beneficios de un sistema de monitoreo en unidades transporte público y además nos permita desplegarlo de forma rápida en cualquier escenario dada su versatilidad.

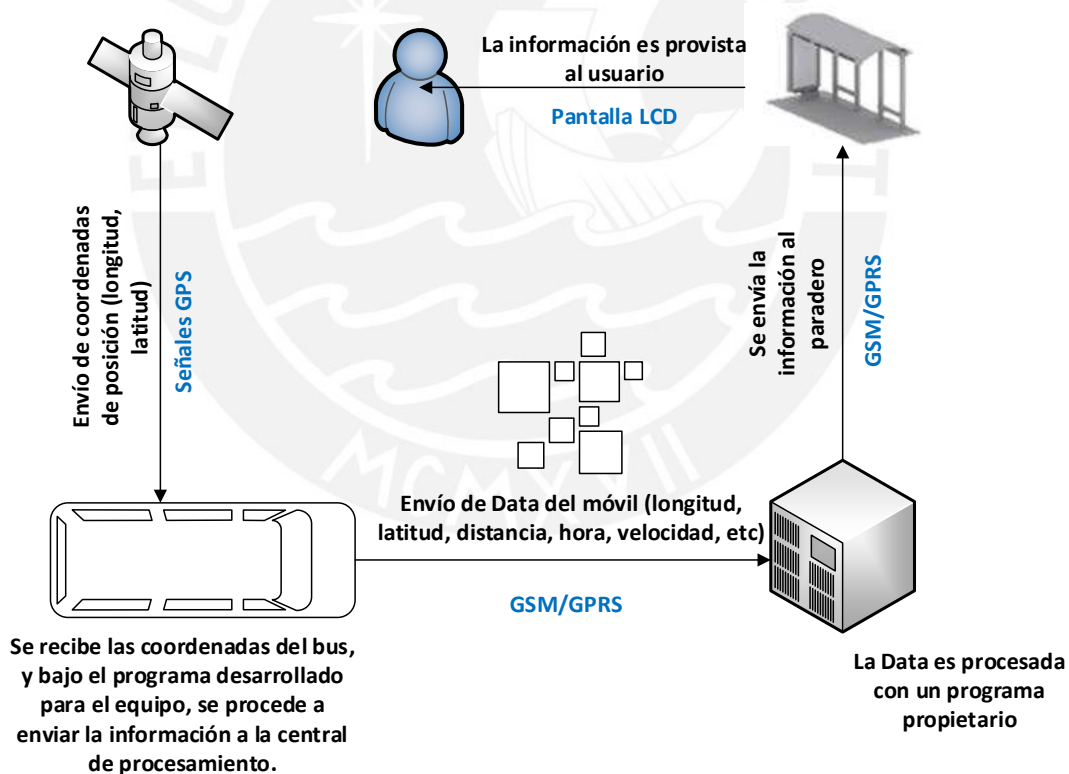


Figura 3.1. Red del sistema de monitoreo [Elaboración propia]

Como se puede apreciar en la figura 3.1 solo el móvil hace uso de señales GPS/GSM ya que es el único que provee posicionamiento, en tanto que el paradero y la central son puntos fijos que tan solo necesitan comunicación GSM.

El Hardware del Sistema de Monitoreo Piloto tiene las siguientes partes:

➤ **Paradero con visualización de datos:**

Permite acceder visualmente a los datos de una o varias líneas de transporte público que pasan por determinado paradero, esta información es proporcionada por la central de monitoreo mediante el envío de mensajes de texto (GSM/GPRS). Los datos pueden ser: distancia al paradero, tiempo de llegada, ruta, número de línea, etc.

Un sistema con múltiples paraderos requiere que la comunicación con la central sea rápida y que permita mostrar datos actualizados de manera constante, una alternativa viable es usar un medio físico con comunicación TCP/IP que permita tener altas velocidades de transmisión y un ancho de banda aceptable; sin embargo, para nuestro sistema piloto de monitoreo realizaremos esta comunicación mediante GSM/GPRS debido a que buscamos que nuestro piloto sea desplegable y nos permita realizar pruebas en múltiples ubicaciones. Las partes del Hardware de Visualización de Datos, está compuesto por:

- **Un transceptor GSM:** El cual se configurara para recibir los mensajes de texto de la central mediante GSM y retransmitirlos vía comunicación serial al micro controlador.
- **Una antena GSM:** La cual se conectara al transceptor y permitirá la recepción de los SMS, esto debido a que la señal GSM no es homogénea en la ciudad.
- **Una pantalla LCD de 128x64:** Se hace uso de una pantalla LCD para poder visualizar los datos que llegan al transceptor del paradero.
- **Un microcontrolador:** Se emplea como interface entre el transceptor y la pantalla LCD debido a que el transceptor no tiene el número de puertos necesarios para la gestión de una pantalla LCD; además, será el microcontrolador quien inicie la aplicación JAVA en el transceptor del paradero. El microcontrolador recibirá los datos proporcionados por el transceptor mediante comunicación serial y seguidamente los procesará para generar códigos de datos y retransmitirlos a una pantalla LCD.

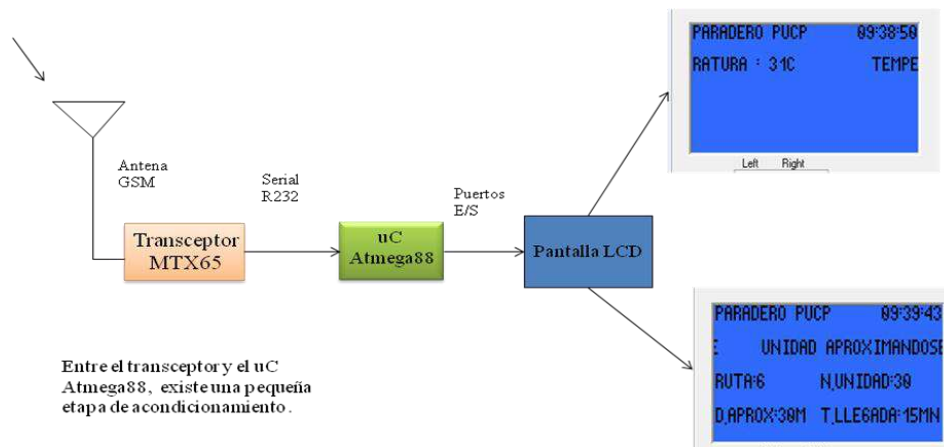


Figura 3.2. Diagrama de bloques del paradero interactivo. [Elaboración propia]

➤ **Monitoreo:**

Se instalará un transceptor GPS/ GSM-GPRS en el vehículo monitoreado, el cual se configura para calcular su posición mediante el uso del sistema de posicionamiento global (GPS), obteniendo coordenadas geodésicas, las cuales son enviadas a la central mediante SMS. Este proceso se realizará de forma constante teniendo como variable la frecuencia de envío de cada posición, el equipo transceptor se energizará con la batería del mismo vehículo (12VDC).

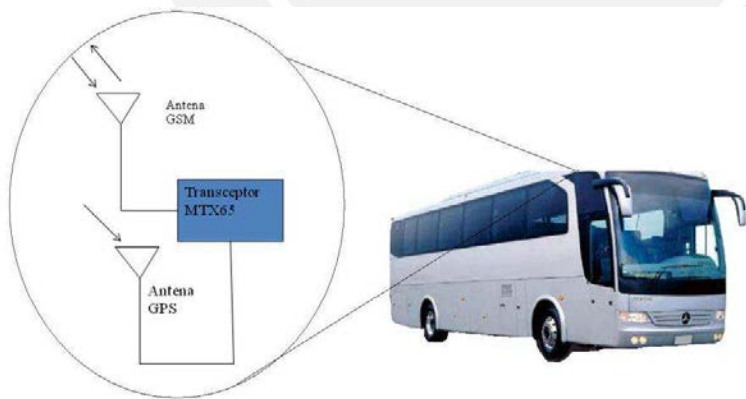


Figura 3.3 Diagrama de bloques del hardware de monitoreo instalado en el bus. [Elaboración propia]

➤ **Interface de comunicación con la Central de Monitoreo:**

La central de monitoreo es la encargada de comunicarse tanto con buses como con los paraderos, por ende el medio de comunicación a escoger entre ellos debe poder soportar un alto tráfico de información; no obstante, para la implementación de nuestro piloto utilizaremos comunicación GSM ya que al tener un solo bus de prueba y un solo paradero el tráfico de datos no será elevado y nos permitirá poder realizar el análisis de factibilidad de la implementación de un sistema de monitoreo.

Es por ello que la arquitectura de nuestra central de monitoreo piloto deberá tener un módulo transceptor GPS/GSM-GPRS, el cual se configuró como interface de comunicación (Figura 3.4) para que pueda gestionar los mensajes de texto provenientes de los buses u otros vehículos monitoreados; del mismo modo, le permite enviar datos procesados a cada uno de los buses y paraderos para proporcionar información relevante.

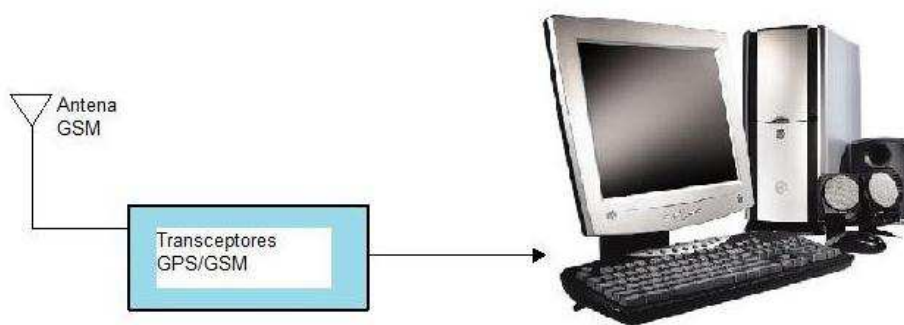


Figura 3.4. Diagrama de bloques de la interface de comunicación. [Elaboración propia]

3.1.1. Requerimientos del Hardware

El correcto funcionamiento del Sistema de Monitoreo está sujeto a los siguientes requerimientos:

Requerimientos del transceptor

Banda GSM de trabajo: Hace referencia a la frecuencia de comunicación que se utilizó. Ésta varía dependiendo la región en la cual se está operando. Es indispensable verificar la

banda de trabajo de los equipos transceptores para no tener problemas en la comunicación de datos.

Tiempo de adquisición: Es el tiempo que le toma al transceptor para poder obtener las señales de los satélites que orbitan la Tierra y obtener su posición. A mayor tiempo de adquisición los datos que se tomen serán menos exactos. Por ende se realizó una comparación entre todas las opciones de transceptores del mercado y ponderar este parámetro de forma importante.

Sensibilidad de recepción GPS: La sensibilidad se define como el nivel de potencia más baja a la cual cualquier receptor inalámbrico produce una tasa de error de bits mínima deseada ó Bit Error Rate (BER). Este valor está relacionado con la SNR_i del receptor y con el ruido térmico del ambiente por el cual se propagan las señales GPS [24]. La selección de este parámetro se realizó por comparación teniendo en cuenta la potencia con la cual la señal GPS llega a la superficie terrestre.

Número de canales para la recepción de señales GPS: Es un parámetro de los transceptores. A mayor número de canales el dispositivo puede triangular con mayor exactitud los datos de posicionamiento.

Requerimientos del LCD

Tamaño de la pantalla de visualización: Este parámetro es importante ya que está directamente relacionado con la cantidad y los tipos de datos que se desean mostrar. Para nuestro modelo piloto bastó con escoger una pantalla LCD de tamaño intermedio que permitía visualizar hasta 3 líneas de texto.

Requerimientos del microcontrolador

- Capacidad para gestionar los números de pines que requiere el control de la pantalla LCD a elegir.
- Soportar comunicación serial RS232.
- Memoria no volátil para almacenar y mostrar los datos del paradero. Debe poder almacenar como mínimo hasta 10 mensajes de 10 caracteres (100 bytes).
- Multi tarea, que permitan al microcontrolador realizar la actualización de los mensajes sin perder sincronismo con la comunicación serial.

Adicional a todos los requerimientos particulares de cada módulo, como requerimiento general del sistema se solicita:

- **Usar componentes electrónicos disponibles:** debido a que se desea tener un producto que se pueda fabricar en serie, sus componentes deben de estar disponibles en el mercado nacional.
- **Portabilidad y bajo costo:** se desea diseñar un hardware de bajo costo que pueda ser atractivo a las empresas de transporte público, por ende la selección de cada uno de los componentes deberán cumplir con este parámetro.

3.1.2. Selección de los componentes del hardware de monitoreo

Terminal GPS/GSM

Para determinar el transceptor adecuado se compararon varios equipos en el mercado (Tabla 3.1) y se realizó el análisis de sensibilidad teniendo en cuenta la potencia con la cual llega una señal GPS a la superficie terrestre, la cual tiene un valor de 1×10^{-16} W esto equivale -130dBm [24].

Tabla 3.1. Cuadro Comparativo de marcas y modelos de Transceptores.
[Elaboración propia]

Marca	TRACKER	MatriX	Teltonika
Modelo	TK-102B	MTX – 65	FM1100
Banda GSM	850/900/1800/1900Mhz	850/900/1800/1900Mhz	900/1800Mhz
Tiempo de Adquisición	Primera 45 s Reconexión 35 s	Primera 35 s Reconexión 2 s	Primera 30 s Reconexión 2 s
Sensibilidad de Recepción	-150dBm	-157 dBm	-161 dBm
Número de Canales	10	16	20
Programación	Software programación parametrizado	JAVA	JAVA
Costo (*)	37.5 USD	45 USD	75 USD

(*) Los costos no consideran gasto de importación

En el siguiente cuadro se muestra la comparación de sensibilidad de los equipos analizados, para comprobar que la sensibilidad de nuestros equipos puede recibir de manera adecuada la señal de GPS.

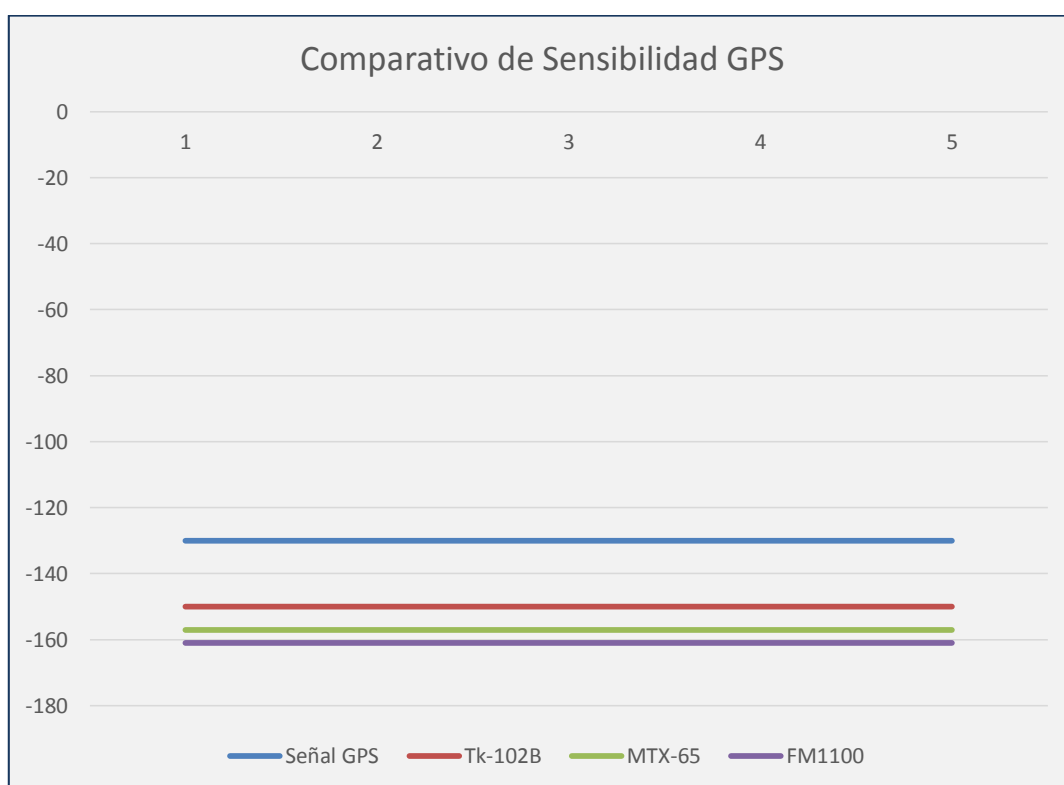


Figura 3.5. Comparación de sensibilidad. [Elaboración propia]

Finalmente se determinó el empleo del Terminal MTX65+G+B.V5 (Figura 3.5) debido a que tiene un costo intermedio y sus características cumplen con los parámetros necesarios de selección. Además nos permite la programación del mismo mediante JAVA con lo cual podríamos personalizar las funciones del equipo y realizar algoritmos de operación únicos. El presente equipo posee las siguientes características (Tabla 3.2):

Tabla 3.2. Tabla de características del MTX - 65. [Elaboración propia]

Chip GPS	Ublox de ATMELGPS integrado
Canales	16
Bandas GMS	Cuatribanda
Batería	Li-Poly, 1600mA/H,3.7V
Alimentación	3.7 a 5 VDC
Puertos	Serial Rx y Tx, Mini USB, DB15
Comunicación	GSM/GPRS
Programación	Comandos AT
Sensibilidad de GPS	-157dBm
Puertos de Propósito general	6
Programación	JAVA(J2ME)
Servicios de internet	TCP, UDP, HTTP, FTP, SMTP, POP3



Figura 3.5. Transceptor MTX65. [Elaboración propia]

Microcontrolador

El microcontrolador escogido para el desarrollo del proyecto es el ATmega88 (Figura 3.6) ya que sus características cumplen con los requerimientos mínimos que exige el sistema, además los laboratorios de la PUCP nos brindan los recursos necesarios, tales como simuladores y licencias para hacer pruebas y simulaciones de los programas que se puedan utilizar.

Características:

- Memoria Flash: 4k Bytes
- Memoria EEPROM : 512 Bytes
- Puerto Serie : RXD ,TXD (PD0,PD1)
- Baud Rate : 2400,4800,9600,...62.5kbs
- Numero de pines de propósito general: 22



Figura 3.6. Atmega88A.
[Elaboración propia]

Pantalla LCD

- Debido a que se necesitará mostrar múltiples datos al mismo tiempo se optó por buscar pantallas LCD gráficas, las cuales nos permiten digitar todo tipo de símbolos o figuras.
- Los tamaños de las pantallas LCDs que se encontraron en el mercado local son: 128x64, 256x128, 160x128, sin embargo el precio de estos dispositivos es elevado y proporcional al tamaño.
- Se escogió la pantalla LCD de 128x64, que nos permitió simular un monitor como el que se plantea usar en el paradero real y nos brinda el espacio necesario para lo que deseamos mostrar.
- Es en este escenario se realizó una búsqueda en el mercado local y se eligió la pantalla LCD 128x64 de la marca HANTROIX (Figura 3.7), la cual posee además una hoja de datos muy detallada y es muy sencilla de implementar.

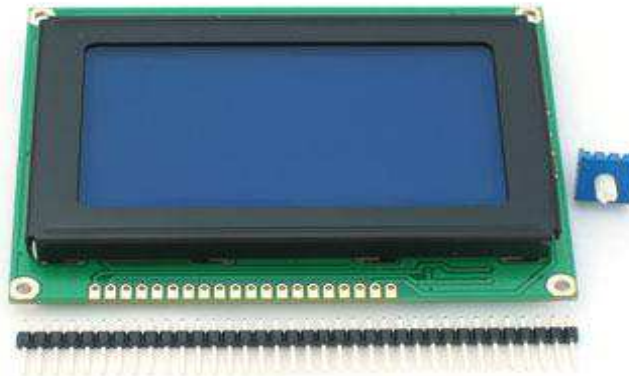


Figura 3.7. LCD 128x64 Hantronix. [Elaboración propia]

3.1.3. Diseño del sistema de visualización de datos para el paradero.

Después de la etapa de selección, lo siguiente es la implementación del Hardware de visualización el cual está compuesto de 3 etapas: Recepción de data a cargo del transceptor MTX65 vía mensajes de texto; adaptación de señales seriales entre el MTX65 (RS232) y el microcontrolador (TTL) ; y por último la etapa de visualización compuesto por la pantalla LCD y el microcontrolador ATmega88 quien actualizara los datos mostrados .

➤ Recepción de data

El Transceptor MTX 65 enviara la data recibida por mensajes de texto mediante su puerto serial RS232 hacia el microcontrolador, para lo cual se debe de implementar la etapa de adaptación que se explicara a continuación.

➤ Adaptación de señales seriales entre el transceptor y el microcontrolador

El transceptor posee un puerto serial con estándar RS232 con niveles de voltaje de +5,-5V (figura 3.10).

El micro controlador atmega88 es de tecnología TTL (*transistor-transistor logic*) y los puertos seriales que posee trabajan con niveles de voltaje de 0 y 5 V, para "0" y "1" lógico respectivamente (figura 3.10).

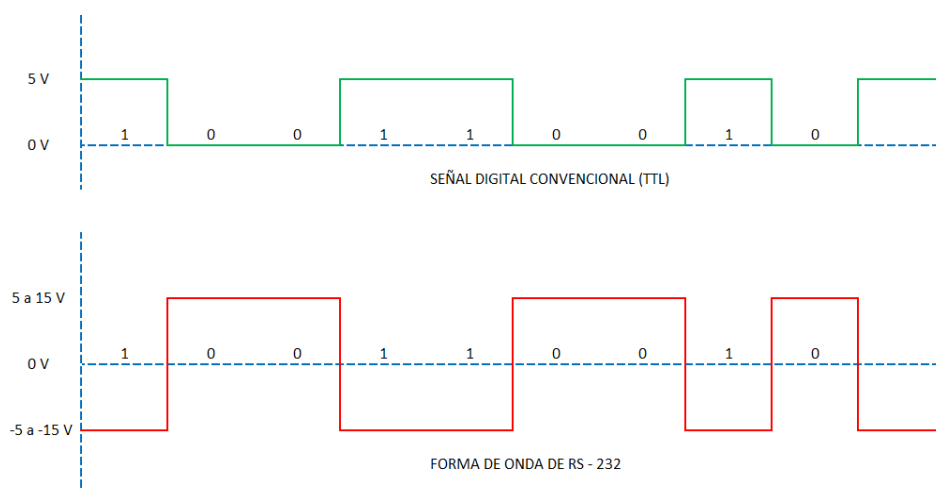


Figura 3.10. Comparación de niveles de comunicación RS232 y TTL
[Elaboración propia]

- Para el acondicionamiento de las señales recortaremos la onda, que nos brinda el transceptor a través de sus puertos de comunicación serial con estándar RS232, detallados en la tabla 3.3, mediante un diodo y seguidamente la invertiremos para poder obtener una onda que el microcontrolador pueda leer.

Tabla 3.3. Cuadro de puertos del transceptor MTX65
[Elaboración propia]

Pin	Señal	Tipo	niveles de voltaje
2	TD 0	Out	$\pm 5V$
3	RD 0	In	VILmax=0.6V VIHmin=2.4V
14	GND	-	0V
10	VCC	-	3.3VDC

Debido a que tanto el microcontrolador y la pantalla LCD tienen alimentaciones distintas, se empleará un optocoplador para la adaptación de niveles.

Componentes utilizados

- 1 diodo 1N4001: se seleccionó este diodo debido a que el voltaje inverso que tendrá como máximo es de -5 VDC y este diodo soporta hasta -50VDC.
- 1 CI 74HC04: Se eligió este circuito debido a que el voltaje que alimenta al transceptor es de 3.3 V, lo cual hace que la tecnología Half-Cmost sea ideal para esta aplicación.
- 1 CI 4N35, optocoplador.
- Las resistencias se calcularon de acuerdo a las hojas de datos de los CI 74HC04 y 4N33.

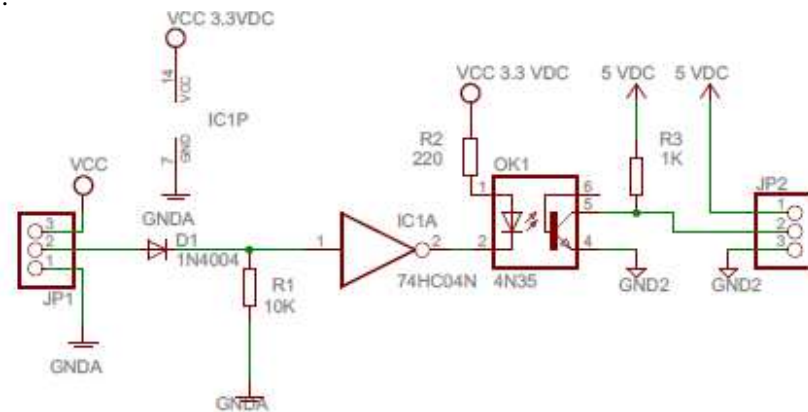


Figura 3.11. Acondicionamiento de señal RS232 a TTL [Elaboración propia]

➤ Visualización en pantalla LCD

La pantalla LCD (128x64) posee 20 pines de control, por lo que se escogió los puertos adecuados en el microcontrolador para su respectiva implementación; además, se debe de considerar la fuente de alimentación que energizará tanto al microcontrolador y a la pantalla LCD. El circuito esquemático de la conexión entre la pantalla LCD y el microcontrolador se detalla en la figura 3.8.

Componentes utilizados:

- Microcontrolador Atmega88.
- Pantalla LCD 128x64.
- Regulador de voltaje (Circuito integrado L7805).

- 2 Potenciómetros de 20K Ω (Estos son componentes que recomienda la hoja de datos de la pantalla Hantronix).
- 1 condensador de 1 μ F (Componentes recomendado en la hoja de datos del CI LM7805).
- 1 condensador de 10 μ F (Componentes recomendado en la hoja de datos del CI LM7805).

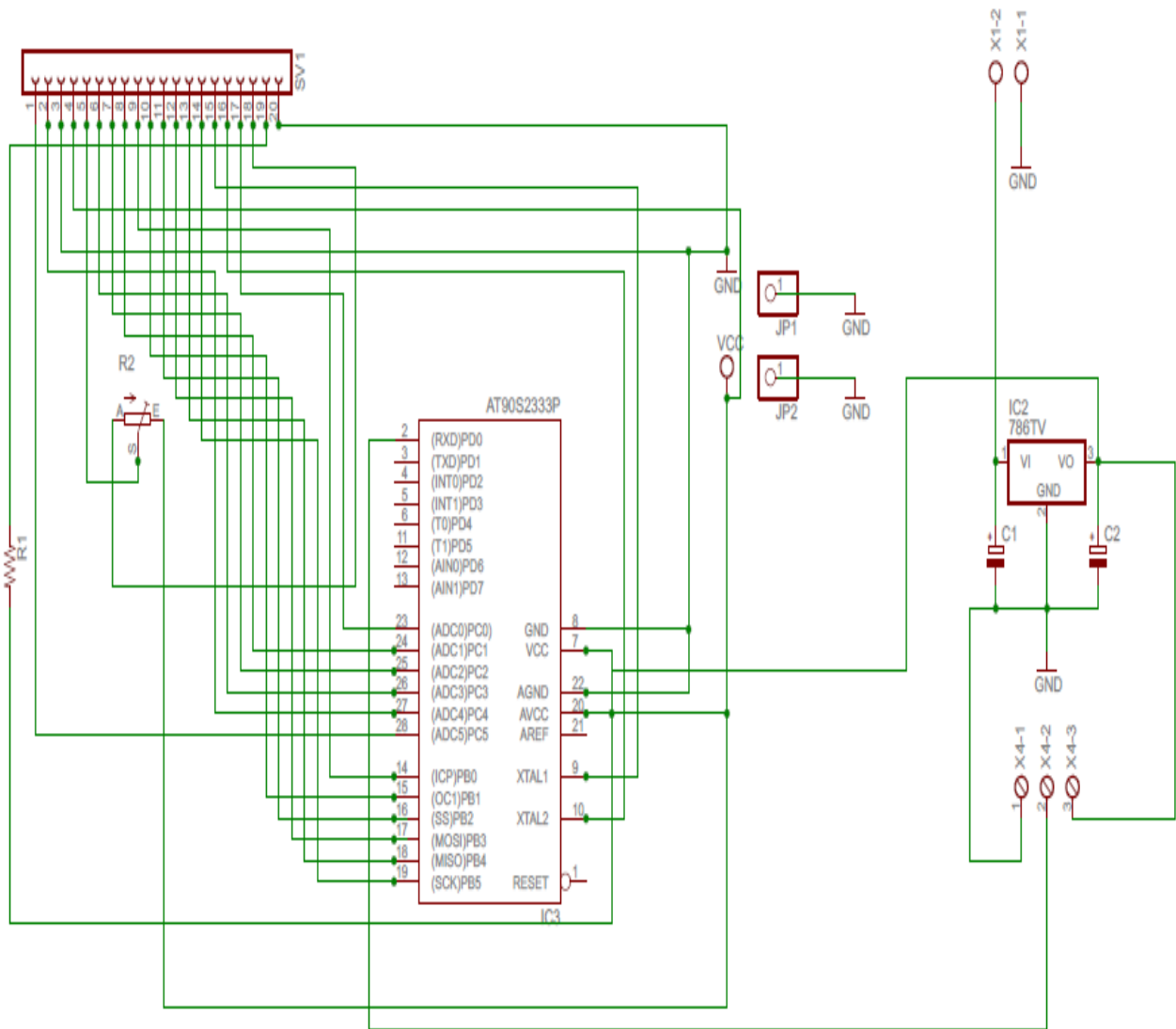


Figura 3.8. Circuito esquemático de la conexión entre la pantalla LCD y el microcontrolador [Elaboración propia]

Campos de la Interface gráfica LCD.

Una vez configurado adecuadamente el transceptor se visualizará en la pantalla LCD el mensaje de bienvenida inicial (figura 3.9), que tiene los siguientes campos:

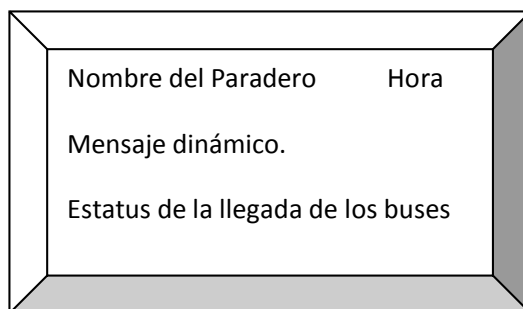


Figura 3.9. Modelo de la interface gráfica [Elaboración propia]

Antenas GPS y GSM

Los transceptores MTX65 necesitan estar conectados a dos tipos antenas: una para poder acceder a la red GSM y la otra para la recepción de datos GPS. Para el proyecto de la presente tesis no todos los transceptores necesitan estar conectados a las dos antenas, esto dependía de la función que desempeñaban. Las antenas no tienen mayores restricciones solo deben de cumplir lo siguiente:

- Antena GPS de conexión SMA hembra (figura3.12).
- Antena GSM de conexión FME macho (figura3.13).



Figura 3.12. Antena GPS
[Elaboración propia]



Figura 3.13. Antena GSM
[Elaboración propia]

3.1.4. Programación de los transceptores GPS/GSM

Para la programación de los transceptores el fabricante (MATRIX ELECTRONICS) nos brinda el Cinterion Mobility Toolkit Installation CD, el cual posee MES (Module Exchange Suite, figura 3.14), este software nos permite acceder a la memoria flash del transceptor como si fuese una memoria externa para grabar programas embebidos con JAVA para la gestión de los comandos AT a través del puerto serial o el puerto mini USB (figura 3.15) del MTX65.

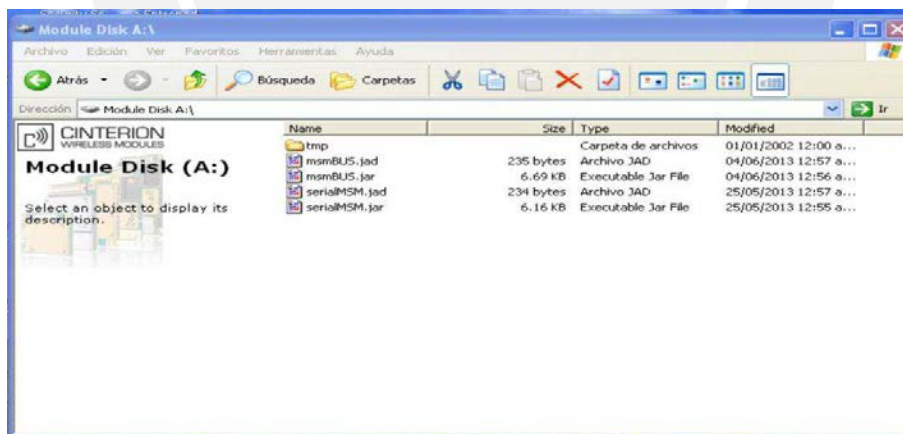


Figura 3.14. MES (Module Exchange Suite)
[Elaboración propia]

3.1.5. Configuración de modos de operación

El transceptor MTX65 permite su configuración mediante comunicación serial a través del puerto mini USB del equipo, haciendo uso de determinados comandos AT. Esta comunicación se ejecuta a través del hyperterminal de Windows.



Figura 3.15. Puerto mini USB del transceptor
[Elaboración propia]

Algunos comandos AT:

AT, ATI, AT+CSQ, AT

Ejemplo: Método para el envío de mensaje de texto.

AT+CMGF=1 : Configura la forma de comunicación móvil a utilizar (GSM, UDP)

OK : El transceptor responde con un OK, Si no existe error.

AT+CMGS= #numero#

>mensaje »

»CMGS: 12 : El lugar que ocupa en la bandeja de salida.

OK

Debido a que se tiene que estar enviando constantemente comandos AT para que el transceptor ejecute cierto tipo de función deseada, se gestionó los comandos AT mediante el uso de aplicaciones JAVA, las cuales se crearon con NetBeans IDE 6.0.1 (figura 3.16), el cual nos permite acceder a librerías del transceptor para las creación de los archivos .JAR y .JAD, estos son grabados directamente al transceptor mediante el MES (Module Exchange Suite).

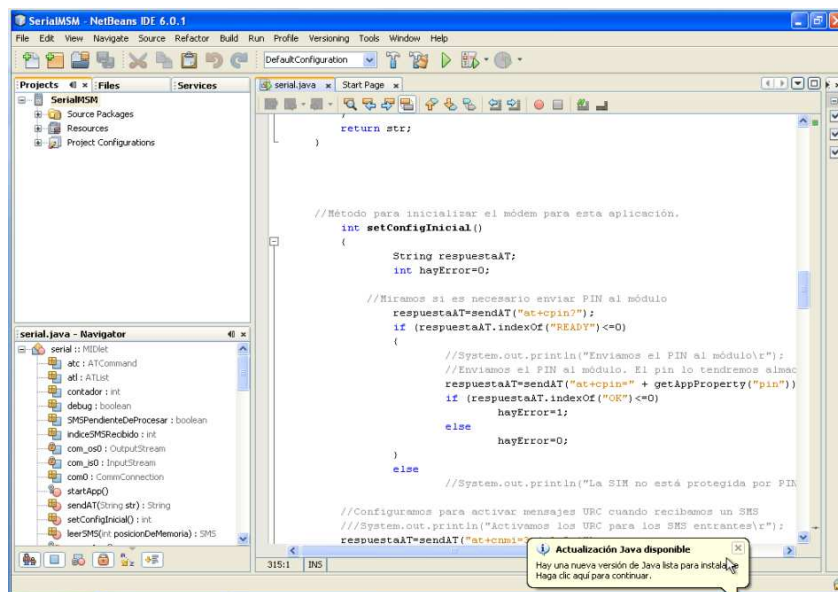


Figura 3.16. Entorno de programación JAVA en NetBeans IDE 6.0.1
[Elaboración propia]

3.1.5.1. Modo comunicación serial

Este modo de operación se utilizará en el paradero piloto como interface entre el transceptor y el microcontrolador; del mismo modo, en la central de monitoreo como interface de comunicación entre la PC y el transceptor. El primer paso para crear cualquier modo de funcionamiento es crear una clase especial para la gestión de los comandos AT, para esto se deben declarar las siguientes librerías:

import com.siemens.icm.io.ATCommand;

import com.siemens.icm.io.ATCommandFailedException;

import com.siemens.icm.io.ATCommandListener;

Método propietario para enviar los comandos AT.

```
String sendAT(String str) {
    try {
        str = atc.send(str + "\r".toUpperCase());
    } catch (IllegalStateException e) {
        e.printStackTrace();
    } catch (IllegalArgumentException e) {
        e.printStackTrace();
    }
}
```

```

    } catch (ATCommandFailedException e) {
        e.printStackTrace();
    }
    if (debug) {
        System.out.println(str);
    }
    return str;
}

```

Para permitir la comunicación serial de nuestro transceptor se debe de incorporar en el programa las siguientes librerías:

```

import java.io.InputStream;
import java.io.OutputStream;
import javax.microedition.io.*;

```

Luego de declarar correctamente las librerías de comunicación serial se procede a escribir las líneas de código necesarias que nos permitirán configurar los parámetros (número de bits, paridad, bits de parada) de la comunicación serial.

Estas líneas son para la entrada de datos.

```

try{
    com0 = (CommConnection) Connector.open("comm:com0;" +
        "baudrate=9600;" +
        "bitsperchar=8;" +
        "stopbits=1;" +
        "parity=none;" +
        "blocking=off;" +
        "autocts=off;" +
        "autorts=off");
    //Asignamos los streams de salida y entrada del puerto ASC0
    com_os0 = com0.openOutputStream();
    com_is0 = com0.openInputStream();
    // System.out.println("configuracion de serial exitosa");

}catch(IOException e){e.printStackTrace();}

```


Dónde:

com_os0= stream asignado para la escritura de datos.

com_is0 = stream asignado para la lectura de datos.

3.1.5.2. Modo lectura y envío de mensajes de texto

Este modo de operación se empleará en el bus para el envío de coordenadas, del mismo modo para el paradero piloto, para recibir los mensajes enviados por la central de monitoreo. Para el envío de mensaje de texto se debe de crear un sub programa que configure el transceptor con los comandos necesarios para recibir mensajes de texto, así mismo, se debe escoger el tipo de mensajería que se utilizará. Como se mencionó previamente, en el modo de comunicación serial se debe de crear la clase de gestión de comandos AT, la cual nos ayudará a la programación en este modo de funcionamiento. La lógica de configuración, lectura y envío se detallan en las figuras 3.17, 3.18 y 3.19.

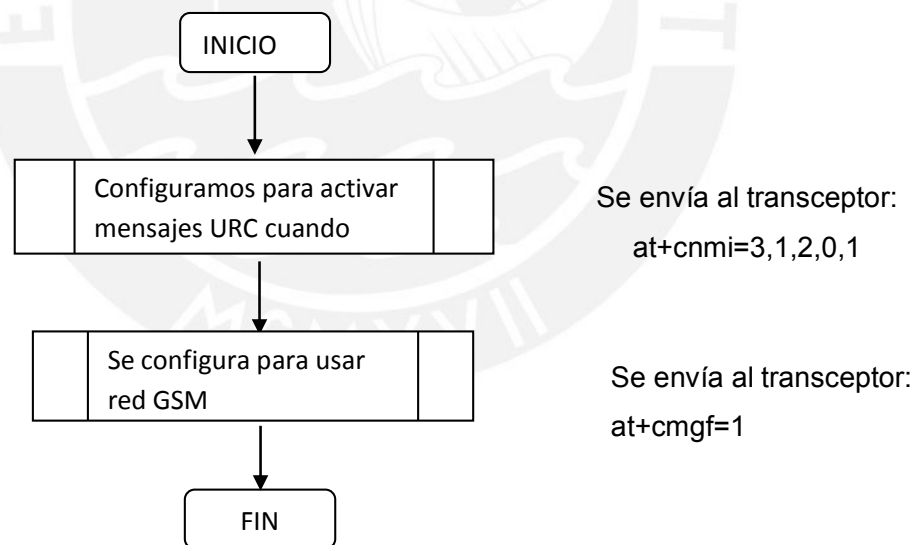


Figura 3.17. Diagrama de bloques de configuración GSM
[Elaboración propia]

Lectura de mensaje de texto

Diagrama de bloques del subprograma de lectura de mensaje de texto.

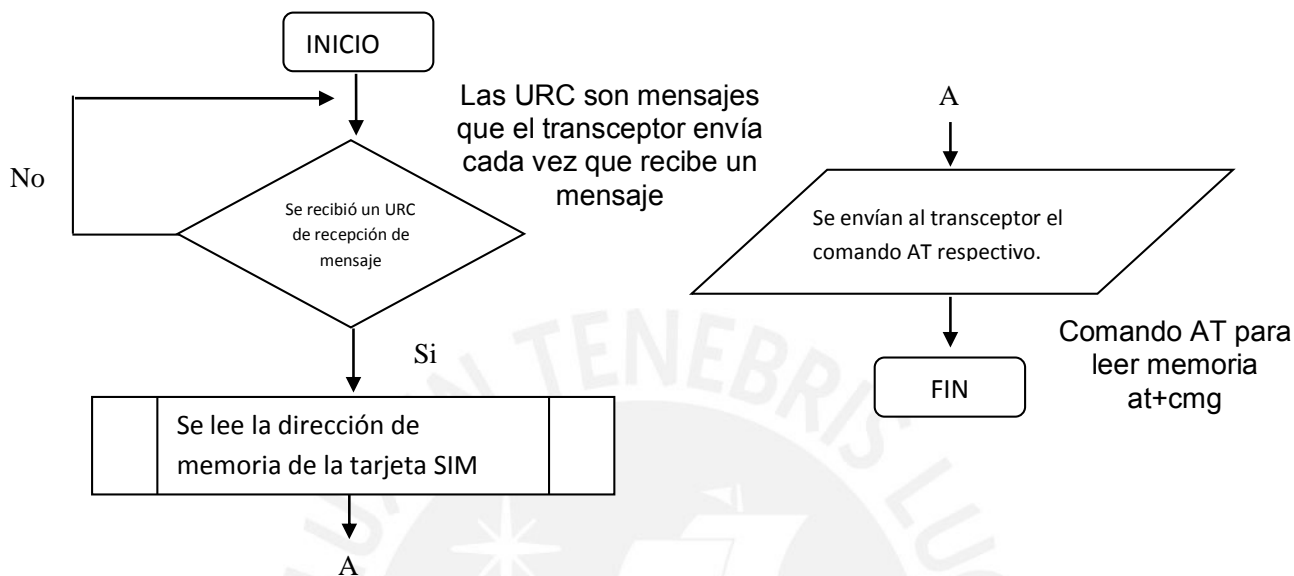


Figura 3.18. Diagrama de bloques de lectura de mensaje [Elaboración propia]

Envío de mensaje de texto

Diagrama de bloques del sub programa de envío de mensajes de texto.

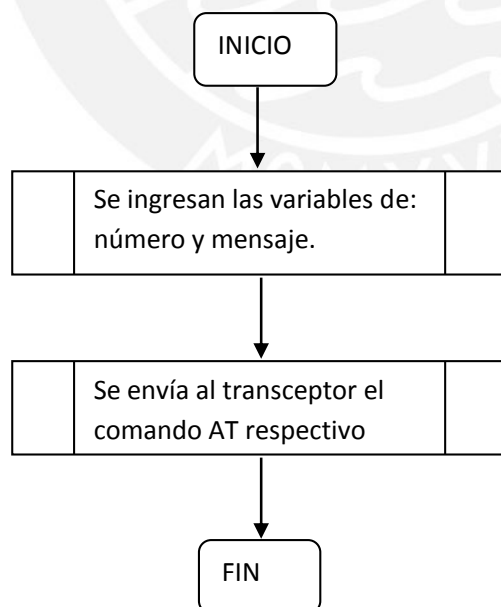


Figura 3.19. Diagrama de bloques de envío de mensajes [Elaboración propia]

3.1.5.3. Modo recepción de coordenadas GPS

El siguiente modo de operación se utiliza para la adquisición de coordenadas geodésicas de cada unidad monitoreada (figura 3.20), las cuales se enviarán a la central de monitoreo vía mensaje de texto a una determinada frecuencia.

AAAA/MM/DD,hh:mm:ss,XX.XXXXXXX,S,YYY.YYYYYYY,W

Fecha Hora Latitud Longitud

Figura 3.20. Formato de coordenadas geodésicas adquiridas por el MTX65 [Elaboración propia]

Se utilizarán los comandos:

Activación del servicio de GPS del transceptor: $AT^SGPSS=1,0$

Frecuencia de envío del mensaje: $AT^SGPSP=tiempo$

3.2. Diseño e implementación del software

3.2.1. Requerimientos de algoritmo de monitoreo

El procesamiento de la información se puede describir en cuatro funciones imprescindibles:

- Colección de información.
- Procesamiento y edición de la información.
- Envío y empleo de la información.

La colección de la información se realiza en bases de datos. Estos datos son recibidos por el transceptor y comunicados a la computadora, almacenados y manipulados de tal forma que permitan proveer datos relevantes como:

- Estado de congestión: nombre de la vía, la longitud de la congestión y el grado de congestión.
- Los tiempos de viaje: nombres de carreteras y tiempos de viaje.
- Información de incidentes: nombres de las calles, ubicación, congestión, etc.

El programa debe cumplir con los siguientes requerimientos:

- Fácil programación.
- Fácil manipulación para la creación de Interfaz de usuario (GUI).
- Conexión con gestores de base de datos (MySQL).
- Comunicación RS232.
- Compatibilidad con entorno web para gestión de mapas por empleo del api de google maps.

3.2.2. Módulos y elementos del sistema de control

Para el desarrollo del sistema de monitoreo fueron necesarios los siguientes elementos:

- Transceptor MTX65+G+B.V5
- PC, RAM 2GB, Hard drive de 500 GB y procesador de 1.7 GHz.
- Netbean 7.3
- Visual Basic
- Notepad++, Apache

El programa se desarrolló con VB y netbeans ya que permitieron la elaboración de la interfaz gráfica de forma sencilla.

3.2.3. Módulo de comunicaciones.

La comunicación entre el transceptor y el programa es mediante RS232 puesto que es un modo versátil y sencillo para la comunicación entre la PC y el dispositivo. La configuración establecida fue: 115200 bps, 8 bits, sin paridad, 1 bit de stop (figuras 3.21 y 3.22). Dicha velocidad corresponde a la configuración por defecto del puerto mini USB del transceptor, lo cual establece la velocidad de trabajo. Esta comunicación es requerida ya que el transceptor recibe información cada 60 segundos a través de señales GSM, las cuales son procesadas por el software de monitoreo y enviadas al paradero a través del mismo transceptor.



Figura 3.21. Descripción de conexión [Elaboración propia]

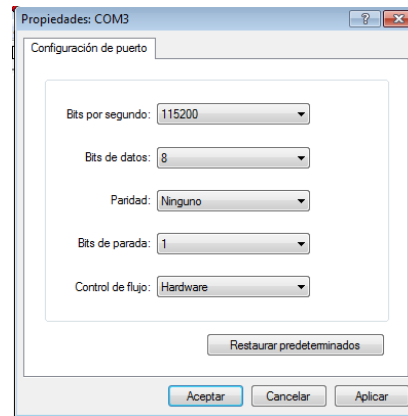


Figura 3.22. Configuración de hyperterminal [Elaboración propia]

3.2.4. Diseño de Software.

Para diseñar el Software debemos establecer el flujo de información, presente en la figura 3.23, por tanto se desarrolló el siguiente diagrama de flujo:

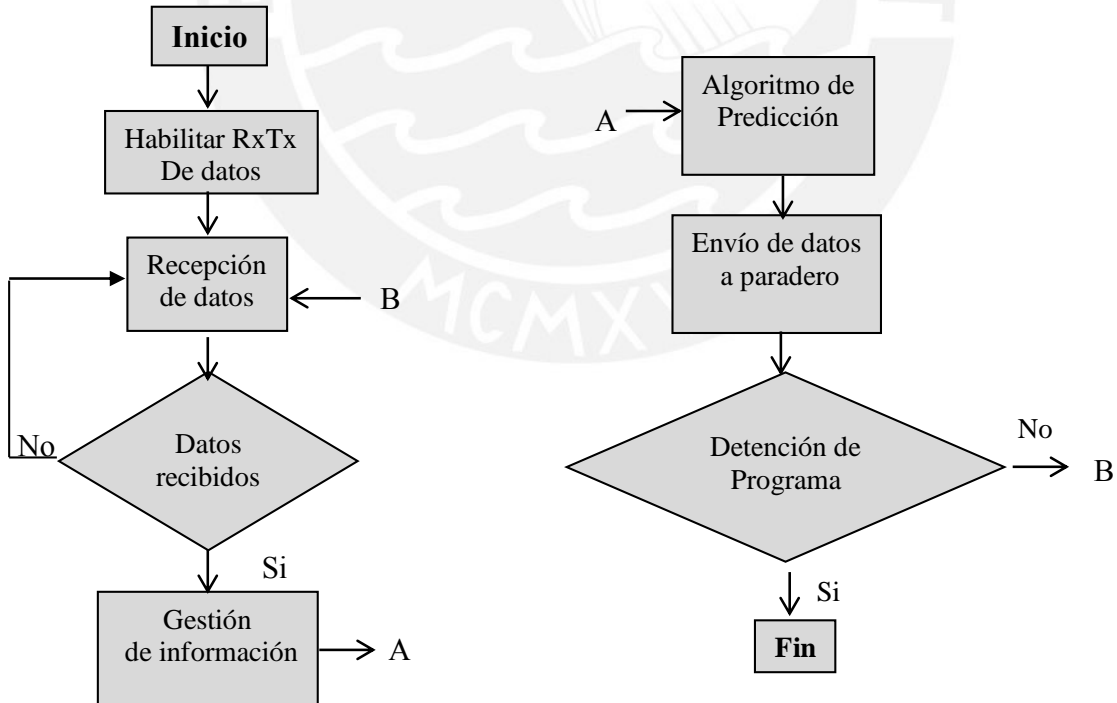


Figura 3.23. Diagrama de flujo general del procesamiento de información [Elaboración propia]

3.2.5. Programación en Visual Basic.

La central de monitoreo demandaba el desarrollo de un software de monitoreo con una interfaz sencilla que pueda ser empleada por un usuario promedio. El programa que se desarrolló fue en Visual Basic, puesto que el empleo de la API de Google maps era mucho más sencillo bajo este lenguaje; sin embargo, si se deseara expandir el programa, se recomienda JAVA puesto que empresarialmente es mucho más conveniente debido a su orientación multiplataforma ya que se emplea con gran frecuencia en el mundo web, por la gran cantidad de herramientas que posee, librerías, etc. Al emplear la API de google maps, se demanda de una conexión a internet para poder cargar los mapas y obtener los valores predictivos de esta plataforma; las pruebas realizadas para esta tesis se realizaron con una velocidad de 1Mbps y no se tuvo problemas con la carga de los mapas, por ello se infiere que con una velocidad de por lo menos 2Mbps se podrá trabajar sin ningún inconveniente.

Como todo software este demanda una interfaz inicial de logueo para el usuario con las credenciales necesarias para emplear el programa. Es por ello que el primer formulario desarrollado fue el de inicio (figura 3.24). Este formulario fue construido con una imagen de google maps que fue animada por código. Asimismo se gestionó el ingreso mediante la creación de un usuario y un password para el acceso del usuario a la plataforma. El código completo se especifica en el **anexo 1**.

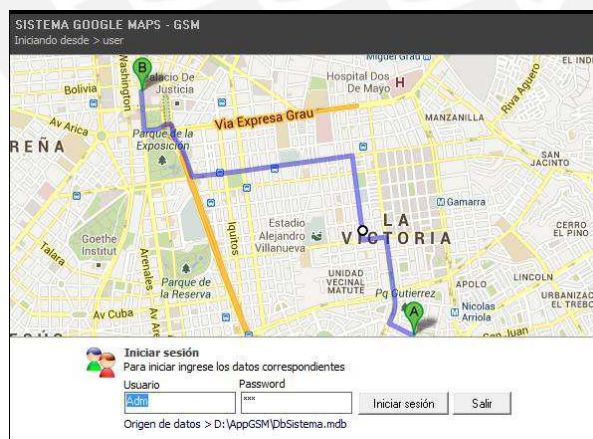


Figura 3.24. Imagen de inicio del programa
[Elaboración propia]

Para la creación de un usuario y password, se creó el siguiente formulario (figura 3.25) en el cual el operario podrá crear el código de ingreso para el usuario en gestión. La codificación se detalla en el **anexo 2**.

Figura 3.25. Cuadro para gestión de nuevos usuarios
[Elaboración propia]

3.2.5.1. Base de datos

De acuerdo a la información brindada en el Capítulo 2 sobre MySQL como Sistema de Gestión de Datos, se entiende la necesidad de su empleo. Cabe indicar que debido a que se trabaja con Visual Basic, y fundamentándonos en su eficiente desempeño con un entorno Windows, en particular con Office, la primera base de datos se gestionó con Access. Posteriormente, se usó MySQL, lo cual conllevó a que se creara el archivo DbSistema, en el cual se guardarían los datos provenientes del bus y posteriormente serían procesados por el programa. Asimismo, se creó el archivo DbcocienteK que es la base de datos en donde se almacenarán los valores que serán tomados para determinar el K (cociente de ajuste) y el posterior ajuste del valor de aproximación. Para este fin se creó el formulario FrmDB (figura 3.26) con el cual se procedería a seleccionar la base de datos con la que deseáramos trabajar, abriendo la posibilidad de poder cambiar de base de datos si así lo requiriésemos. La codificación de este formulario se adjunta el **anexo 3**.

Figura 3.26. Cuadro para seleccionar base de datos
[Elaboración propia]

3.2.5.2. Visualización de mapa

Los procesos más complicados de toda la programación sin duda recayeron en la gestión de la información y visualización de datos en el mapa. En un inicio se pensó en emplear archivos KML, que en esencia son ficheros que almacenan una característica como posición o imagen, que es empleado para representar datos geográficos en Google Earth. No obstante, manejar la información de aquella forma no hubiera sido conveniente para el software ya que una necesidad esencial del sistema radica en su simplicidad y trabajar con ficheros KML implicaría gestionar un generador de archivos de esta clase, crear un hipervínculo con Google Earth y visualizarlo con este programa, lo cual es un proceso engorroso y para nada eficiente. Por ende resultó mucho más conveniente trabajar con la API de Google maps. Dicha herramienta del buscador líder del internet provee una gran variedad de beneficios, entre ellos su sencillez y facilidad de empleo.

VB nos brinda la facilidad de poder “incrustar” un browser dentro de nuestro formulario (figura 3.27), ello provee gran manejabilidad del API dentro del programa, sin necesidad de hipervínculos o abrir programas paralelos al software de monitoreo. Dicha codificación se adjunta en el **Anexo 4**.

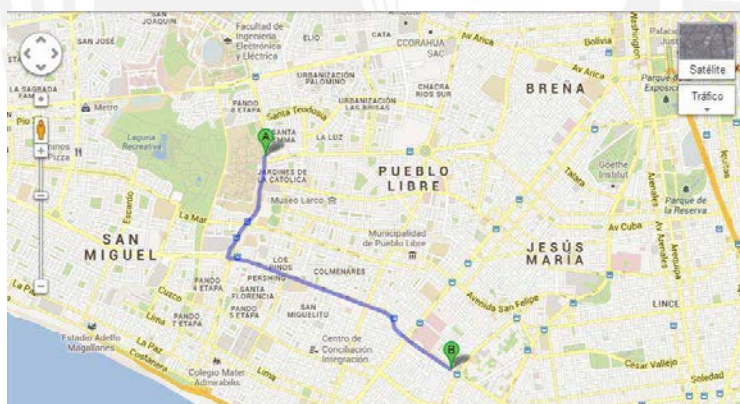


Figura 3.27. Mapa de Google con ruta marcada
[Elaboración propia]

3.2.5.3. Algoritmo de estimación

Google maps ofrece herramientas realmente útiles, una de ellas es el cálculo del tiempo estimado de recorrido entre dos puntos (figura 3.28). Dicha herramienta es realmente muy útil ya que se puede estimar el tiempo que demoraremos para ir de un punto a otro; no

obstante, el software que se ha desarrollado está orientado a trabajar con buses de transporte público y no con carros particulares, lo cual repercute en los tiempos estimados de llegada a un punto en específico (paradero) y por ende demandaba un ajuste.

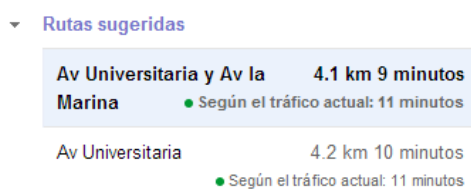


Figura 3.28. Estimación de tiempo de recorrido por google maps
[Elaboración propia]

Por tanto ha sido necesaria la implementación de un algoritmo de estimación fundamentado en la cantidad de tiempos variados que existen en las rutas de Lima. Para fines prácticos, se optó por la ruta que cubre el recorrido entre el paradero de la PUCP (coordenadas en latitud y longitud respectivamente: -12.06798350690308° , -77.07791890632933°) y el paradero ubicado en la intersección de las avenidas Sánchez Carrión y Gregorio Escobedo (coordenadas: $-12.089742473332548^\circ$, -77.05830659401244°). Dichos tiempos fueron tomados y procesados durante casi todo un mes (desde el lunes 03/06/2013 al viernes 28/06/2013, figura 3.29). La información fue tomada en diferentes intervalos de tiempo ya que nos percatamos que el tráfico vehicular variaba dependiendo de la hora. Se pudo apreciar que el tiempo marcaba una tendencia, con una varianza que oscila entre 1 y 2 (**anexo 4**), por tanto dedujimos que se puede estimar un tiempo de viaje determinado dependiendo de la hora de tránsito de los vehículos. Para tal fin se empleó la mediana del total de datos, debido a que esta media es poco sensible a valores muy grandes, muy pequeños o fuera del rango.

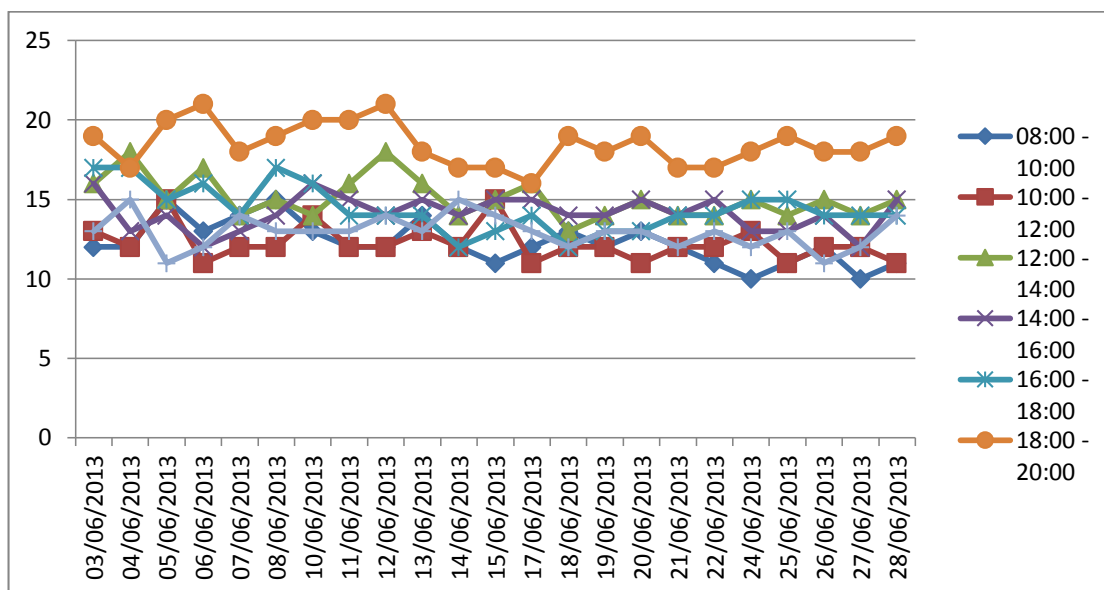


Figura 3.29. Gráfica de dispersión de tiempos tomados en minutos
[Elaboración propia]

3.2.5.4. Ajuste de los datos de predicción

Para la predicción del tiempo nos apoyaremos en la plataforma de Google maps. No empleamos la plataforma Waze ya que en la fecha de realización del programa esta plataforma no ofrecía una API gratuita como la que nos ofrece Google maps; no obstante, se empleó Waze como uso referencial, para hacer la comparación entre este aplicativo y nuestro sistema predictivo.

Cuando ubicamos dos puntos en el mapa de la plataforma de Google maps, éste nos provee un tiempo estimado de viaje entre punto y punto, a este tiempo lo denominaremos T_{google} . El tiempo que provee esta plataforma es un tiempo inexacto puesto que su algoritmo de predicción se basa en el la distancia y la velocidad promedio de un vehículo al recorrer la ruta establecida, sin embargo, este tiempo no es preciso ya que dependiendo de la hora en la que se haga la consulta de predicción, el tráfico puede variar debido a múltiples factores. Es por este motivo que el valor de google debe ser ajustado por un factor (K_h). Este factor se obtendrá del valor de la mediana (Me) dividido entre T_{google} , cuyo cociente será denominado como “cociente de ajuste” ($K=Me/ T_{google}$). El tiempo de aproximación ($T_{aprox.}$) es un valor que irá disminuyendo a medida que el bus se vaya acercando al paradero, por ende, el ajuste se tendrá que aplicar a cada valor que google maps provea de forma

actualizado ($T_{\text{google actual}}$), el valor de $T_{\text{google actual}}$ irá disminuyendo a medida que el bus se aproxime al paradero, como se aprecia en la tabla 3.4, dicho valor será arreglado por el cociente de ajuste para que se pueda obtener un tiempo de aproximación más real. La fórmula empleada será.

$$T_{\text{aprox.}} = (T_{\text{google actual}}) * K_h$$

h: hora en la que se efectúa la estimación.

Por ejemplo:

Para un periodo de 4 pm – 6pm, se obtuvo 23 datos (los datos han sido tomados en minutos y han sido redondeados). Para obtener la mediana de estos datos, se empleó la siguiente fórmula.

$$Me = L_{i-1} + \frac{\frac{N}{2} - F_{i-1}}{f_i} \cdot a$$

Donde:

- L_{i-1} = Límite del intervalo mediana
- a = Amplitud del intervalo mediana
- F_{i-1} = Frecuencia acumulada anterior al intervalo mediana
- f_i = Frecuencia absoluta del intervalo mediana
- N = total de datos

Entonces:

Me = MEDIANA (17,17,15,16, 14,17,16,14,14,14,12,13,14,12,13,13,14,14,15,15,14,14,14)

Me = 14

Luego, el cociente de ajuste será $K = 14/12 = 1.17$

Con este cociente de ajuste se realizó una prueba de predicción en un recorrido desde la intersección de la avenida F. Sánchez Carrión con Gregorio Escobedo hasta el paradero principal de la Pontificia Universidad Católica del Perú. Los resultados de la prueba se presentan en la tabla 3.4.

Acrónimos de la tabla 3.4:

MGPD: Minutos que indica Google a punto destino

K: Cociente de ajuste

DA: Distancia de aproximación

TABP: Tiempo ajustado del bus al paradero

TABPW: Tiempo aproximación del bus al paradero según Waze

TRA: Tiempo real de aproximación

Tabla 3.4. Tabla de ajuste de valores entre el paradero PUCP y Pershing para un intervalo entre 4 pm y 6 pm. [Elaboración propia]

MGPD (min)	K	DA (km)	TABP (min)	TABPW (min)	TRA (min)
13	1.17	4.3	15	13	14
12	1.17	4.0	14	12	13
11	1.17	3.6	13	12	12
10	1.17	3.3	12	11	11
8	1.17	2.9	9	7	8
6	1.17	2.3	7	6	7
4	1.17	1.7	5	4	6
3	1.17	1.3	4	3	4
2	1.17	0.9	2	2	2
1	1.17	0.25	1	1	1

3.2.5.5. Aprendizaje

Para este fin se empleó el algoritmo KNN, descrito en el capítulo 2. Este algoritmo se sustenta en información histórica, por tanto, se tuvo que recopilar una gran cantidad de datos obtenidos con el sistema inicial de predicción, tomados entre las 8:00 horas y 20:00 horas del mes de octubre del 2013 en intervalos de 2 horas. De la amplia gama de datos

obtenidos, se obtuvieron datos que acertaron en la predicción de aproximación y otros que fallaron. En base al índice de acierto, y al margen absoluto de diferencia que existía, se le otorgó una categoría probabilística de acierto, que dependía del rango de hora en la que se tomó la muestra. Esta categoría se describe en la tabla 3.5, en la que se aprecia que entre las 12 y 14 horas del día se obtuvo 28 minutos totalizados de diferencia entre el tiempo real y el tiempo predicho por el sistema, mientras que entre las 18-20 horas se obtuvo un valor de 52 minutos totalizados; por consiguiente, se procedió a categorizar de 1 a 6, en la escala de bueno a malo respectivamente.

Tabla 3.5. Categoría de probabilidad de datos obtenidos.
[Elaboración propia]

horario (horas)	Total (min)	valor asignado
12-14	28	1
10-12	31	2
8-10	33	3
14-16	34	4
16-18	36	5
18-20	52	6

Luego se procede a realizar una recopilación de datos que acertaron junto con datos que no lo hicieron. En vista que hubo una gran variedad de datos que no acertaron, ya que fue la primera muestra, se escogió aleatoriamente una cantidad similar a las que sí lo hicieron, de esta forma se obtuvo la tabla 3.6, en donde se visualiza los minutos predichos, la probabilidad de acierto que tenía conforme al periodo de tiempo en donde se realizó la prueba, y el resultado de la prueba: “a tiempo” o “demorado”.

Tabla 3.6. Categoría de probabilidad de datos obtenidos.

[Elaboración propia]

	Dia	Minutos	Prob. Acierto	status	
Buenos	miércoles	14	6	A tiempo	
	viernes	15	3	A tiempo	
	miércoles	16	2	A tiempo	
	jueves	26	1	A tiempo	
	jueves	16	3	A tiempo	
	miércoles	12	6	A tiempo	
	viernes	12	6	A tiempo	
	martes	16	5	A tiempo	
	martes	17	2	A tiempo	
	jueves	16	5	A tiempo	
	lunes	14	4	A tiempo	
	Malos	martes	14	2	demorado
		miércoles	11	4	demorado
martes		21	1	demorado	
lunes		13	6	demorado	
viernes		12	4	demorado	
martes		13	4	demorado	
lunes		17	3	demorado	
lunes		16	6	demorado	
lunes		19	2	demorado	
lunes		13	3	demorado	
martes		13	3	demorado	
lunes		25	1	demorado	

Gracias a esta tabla se obtuvo la gráfica de dispersión de la probabilidad de acierto versus el tiempo de aproximación, en la figura 3.30 se muestra los datos dispersos de datos buenos y malos. En la misma figura se aprecia el dato muestra que se va a analizar (en color amarillo), tomado entre las 14 y 16 horas, con el cual se aplicará el algoritmo a fin de saber si la estimación tiene mayor probabilidad de ser certera.

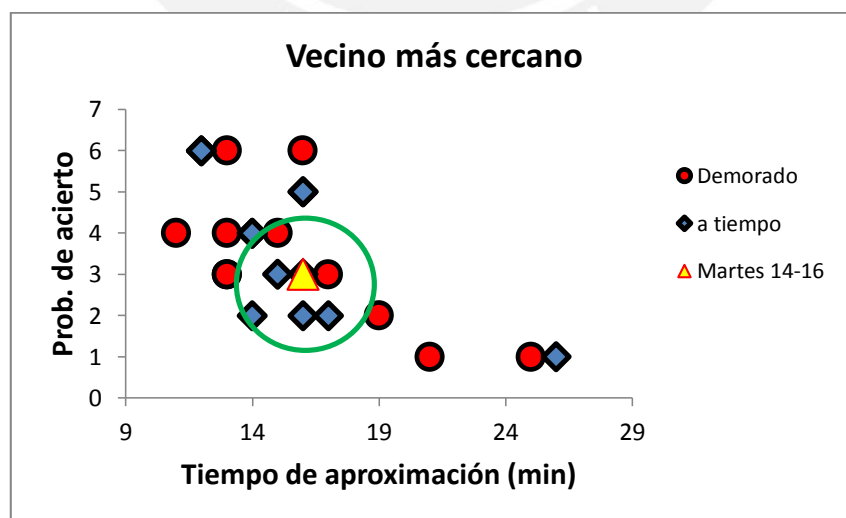


Figura 3.30. Gráfica de vecinos más cercanos [Elaboración propia]

Sabemos que esta técnica se sustenta en analizar la distancia entre los datos dispersos y el dato nuevo que esta evaluado, pero hay un problema, los datos están medidos en escalas diferentes ya que el tiempo de aproximación se mide en minutos y la probabilidad de acierto es adimensional, cuyo valor varía entre 1 y 6; por tanto, es necesario normalizar los datos para que estén en la misma escala. Una forma es transformar tanto los minutos predichos como la probabilidad de acierto a la escala [0-1], para ello se deben tomar los datos mínimos de la escala de minutos y probabilidad de acierto: 11 y 1; y los valores máximos respectivos: 26 y 6. La normalización se realizará empleando la siguiente fórmula:

$$x_{normalizada} = \frac{x - \min(x)}{\max(x) - \min(x)}$$

Con esta fórmula se normalizó los datos de la muestra como se muestra en la tabla 3.7.

Tabla 3.7. Normalización de datos de la muestra [Elaboración propia]

Dia	Minutos	Prob. Acierto	IngresosmesN	EstudiosN
martes	16	3	0.33	0.40

Posteriormente se procede a normalizar el resto de datos tal como se muestra en la tabla 3.8.

Tabla 3.8. Normalización de datos y obtención de distancia con la muestra [Elaboración propia]

Dia	Minutos	Prob. Acierto	status	MinutosN	Prob. AciertoN	Distancia con	El más cercano	K vecinos más cercanos	El vecino es...
miércoles	14	6	A tiempo	0.20	1.00	0.615			
viernes	15	3	A tiempo	0.27	0.40	0.067		Cercano	A tiempo
miércoles	16	2	A tiempo	0.33	0.20	0.200		Cercano	A tiempo
jueves	26	1	A tiempo	1.00	0.00	0.777			
jueves	16	3	A tiempo	0.33	0.40	0.000	El más cercano	Cercano	A tiempo
miércoles	12	6	A tiempo	0.07	1.00	0.657			
viernes	12	6	A tiempo	0.07	1.00	0.657			
martes	16	5	A tiempo	0.33	0.80	0.400			
martes	17	2	A tiempo	0.40	0.20	0.211		Cercano	A tiempo
jueves	16	5	A tiempo	0.33	0.80	0.400			
lunes	14	4	A tiempo	0.20	0.60	0.240			
martes	14	2	demorado	0.20	0.20	0.240			
miércoles	11	4	demorado	0.00	0.60	0.389			
martes	21	1	demorado	0.67	0.00	0.521			
lunes	13	6	demorado	0.13	1.00	0.632			
viernes	12	4	demorado	0.07	0.60	0.333			
martes	13	4	demorado	0.13	0.60	0.283			
lunes	17	3	demorado	0.40	0.40	0.067		Cercano	demorado
lunes	16	6	demorado	0.33	1.00	0.600			
lunes	19	2	demorado	0.53	0.20	0.283			
lunes	13	3	demorado	0.13	0.40	0.200		Cercano	demorado
martes	13	3	demorado	0.13	0.40	0.200		Cercano	demorado
lunes	25	1	demorado	0.93	0.00	0.721			

Para determinar cuán lejos se encuentra la muestra de sus vecinos se calcula la distancia euclidiana, para lo cual se emplea la siguiente fórmula:

$$d(x, y) = \sqrt{\sum_{k=1}^n (x_k - y_k)^2}$$

Una vez determinada la distancia de todos los valores, tenemos que elegir una cantidad “K” de valores vecinos que se encuentren más cerca de la muestra. Se tomaron siete valores vecinos, que representa la tercera parte de la data total, para realizar el análisis de los valores próximos. De estos datos, se obtuvieron 4 vecinos buenos y 3 malos, lo cual nos indica que el valor predicho califica como bueno tal como se muestra en la tabla 3.9.

Tabla 3.9. Número de vecinos buenos y malos [Elaboración propia]

Vecinos malos -->	3	→	Es clasificado como:	Bueno
Vecinos buenos -->	4			

Gracias a esta calificación, podremos mejorar la probabilidad de la data que se almacenará en la base de datos. Esta data predicha se contrasta con el valor real, a fin de catalogar al dato con un índice de acierto. Este índice nos permite emplear el dato predicho dentro de la data total con la finalidad de ajustar el cociente K.

Ajuste de cociente

Al ser clasificado como bueno, tiene una alta probabilidad de que éste valor predicho sea correcto, por tanto, se debe considerar los 4 escenarios descritos en la tabla 3.10. En dicha tabla se aprecia que de los cuatro escenarios, el dato que ingresará a la base de datos DbcocienteK de nuestro sistema es aquel cuya clasificación sea calificado como buena y que además el dato haya acertado en la estimación, de esta forma nos aseguramos que el dato que se ingrese a la base de datos DbcocienteK sea el más adecuado para ajustar el cociente de ajuste K. El dato ingresa a la base de datos y el K se actualiza con los últimos 23 datos, de esta forma el valor del cociente de ajuste irá mejorando conforme mejores resultados se obtengan.

Tabla 3.10. Clasificación del dato predictivo [Elaboración propia]

	¿Dato acertó?	Resultado
clasificación buena	si	dato se guarda en base de valores para K
clasificación buena	no	dato no se guarda en base de valores para K
clasificación mala	si	dato no se guarda en base de valores para K
clasificación mala	no	dato no se guarda en base de valores para K

Una vez determinado la clasificación del dato evaluado, se procede a su ingreso a las base de datos DbSistema con la categorización respectiva (bueno o malo). En resumen el diagrama de flujo de la figura 3.31 describe la secuencia de la programación del aprendizaje dentro de nuestro programa.

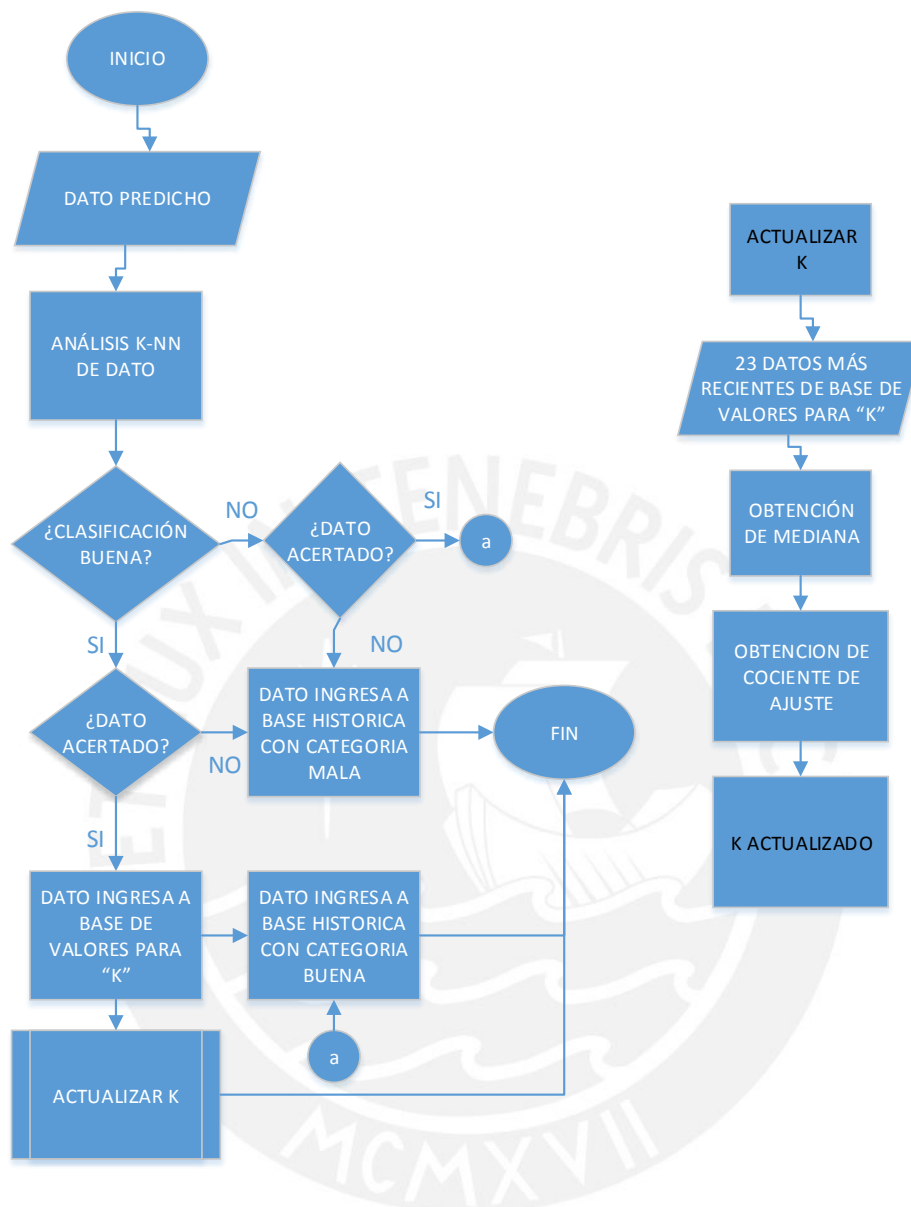


Figura 3.31. Diagrama de flujo general del algoritmo de estimación con aprendizaje [Elaboración propia]

3.2.5.6. Interfaz de usuario.

Con un algoritmo adecuado de estimación de tiempo de aproximación, la visualización del bus y el destino, lo único que queda es juntar todas las partes y establecer la interfaz final que el operario manejará para poder monitorear el bus (figura 3.32). Dicha programación se adjunta en el **anexo 6**. Con la información procesada, se procede a habilitar la transmisión de datos y se envía la información de los datos al paradero.

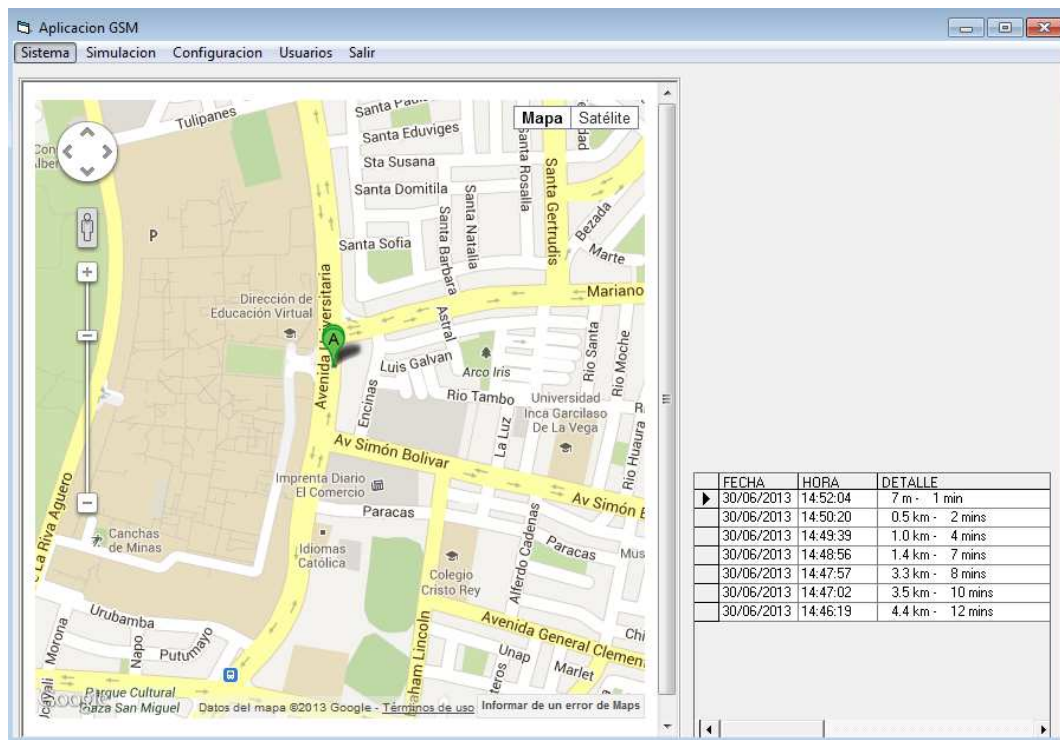


Figura 3.32. Interfaz final desarrollada para monitoreo de buses.
[Elaboración propia]

3.3. Integración del hardware y del software del sistema de monitoreo.

3.3.1. Protocolo de comunicación

Para hacer posible la comunicación entre el hardware y software de monitoreo se desarrolló un protocolo propietario para los mensajes de texto (figura 3.33), de tal manera que los mensajes sean interpretados tanto por el transceptor como el software. Otro beneficio de usar un protocolo de comunicación es la simplificación y la extensión de cada mensaje, con lo cual se evita información innecesaria y el gasto inadecuado de recursos.

Comunicación entre la central de monitoreo y el paradero.

- Mensajes para cambiar el mensaje dinámico de la pantalla LCD.
 - m10** > PROBLEMAS EN LA RUTA
 - m20** > ET "XXXXXXXX"
 - m30** > TEMPERATURA XXC°
 - m40** > T.UNICA 1.50 S/

b > Borra pantalla.

- Cambio de la hora del paradero.

El mensaje que cambia la hora del reloj del paradero tiene 7 dígitos, el primero es un indicador “h” y los siguientes dígitos corresponden a la hora.

hHHMMSS

Ejemplo: Si son las 6:30:20, entonces el mensaje sería h063020

- Datos de unidad que se aproxima

Cuando la central

uABCDEFGHI

Comunicación entre los buses y la central de monitoreo

AAAA/MM/DD, hh:mm:ss, XX.XXXXXXX, S, YYY.YYYYYYY, W

Fecha Hora Latitud Longitud

Figura 3.33. Formato de coordenadas geodésicas adquiridas por el MTX65 [Elaboración propia]

CAPÍTULO 4.

PRUEBAS Y RESULTADOS

En este capítulo se procederán a describir y explicar las pruebas realizadas con los 3 transeptores adquiridos para el desarrollo del presente proyecto. Estos transeptores se emplearon individualmente para el paradero, móvil y para la central de monitoreo; asimismo, en el presente capítulo se describe las pruebas realizadas con el software de monitoreo, así como la integración de este con el hardware.

Se presentan los comentarios de las simulaciones realizadas, los resultados obtenidos y el análisis financiero para un proyecto de implementación tecnológica en el rubro “transporte urbano” para la ciudad de Lima y la viabilidad del mismo para los inversores interesados.

4.1. Descripción de entornos para desarrollo de pruebas.

Las pruebas fueron desarrolladas independientemente y en dos escenarios distintos:

- Escenario de prueba 1: Ambientes de la Pontificia Universidad Católica del Perú (PUCP), figura 4,1.
- Escenario de prueba 2: Ruta que cubre el recorrido entre el paradero de la PUCP (coordenadas en latitud y longitud respectivamente: -12.06798350690308°, -

77.07791890632933°) y el paradero ubicado en la intersección de las avenidas Sánchez Carrión y Gregorio Escobedo (coordenadas: -12.089742473332548°, -77.05830659401244°), figura 4.2.



Figura 4.1. Instalaciones del Pabellón V de la PUCP.

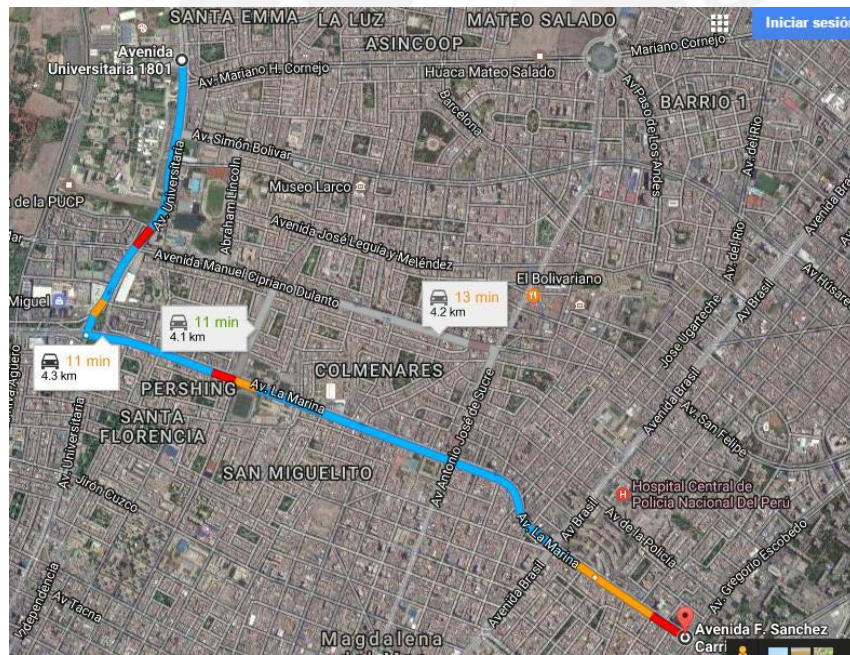


Figura 4.2. Ruta entre Paradero PUCP y el paradero ubicado en la intersección de las avenidas Sánchez Carrión y Gregorio Escobedo. [Elaboración propia]

4.2. Pruebas por módulos

En la siguiente parte se mostrarán todas las pruebas realizadas del sistema de monitoreo. Para las pruebas del transceptor, se tuvo que abrir el equipo (figura 4.3) para conectar la batería interna. Además de contar con una tarjeta SIM (figura 4.4), para poder acceder a la banda GSM.

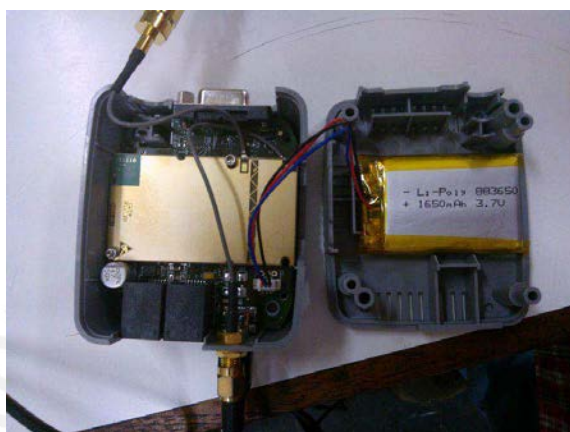


Figura 4.3. Transceptor MTX65 internamente. [Elaboración propia]



Figura 4.4. Transceptor MTX65 con chip movistar.
[Elaboración propia]

4.2.1. Módulo de envío y recepción de mensajes.

Las pruebas realizadas en este punto se hicieron, enviando comandos AT a través del puerto serial del transceptor, para poder ser visualizadas en el hyperterminal de Windows (figura 4.5).



Figura 4.4. Comunicación serial del transceptor con una PC.
[Elaboración propia]

Pruebas de señal

Para saber la buena calidad de señal se utilizó el comando AT: “AT+CSQ”, el cual nos responde con un :

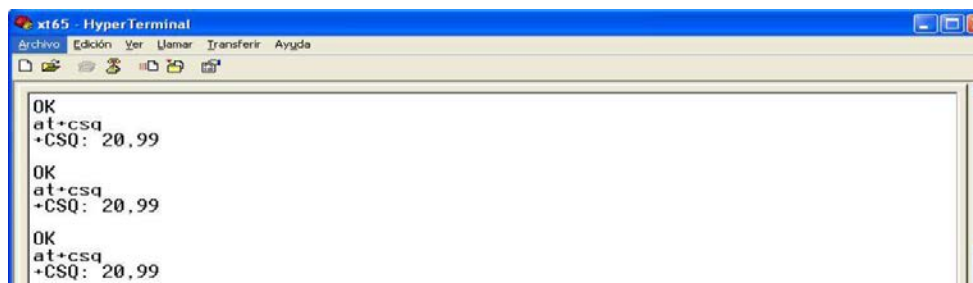
+CSQ:<RSSI>,<BER>

RSSI: Es una escala de referencia (en relación a 1 mW, presente en la tabla 4.1) para medir el nivel de potencia de las señales recibidas por un dispositivo en las redes inalámbricas (típicamente GPS, WIFI, o telefonía móvil).

Tabla 4.1. Cuadro de escalas de RSSI [Elaboración propia]

RSSI	Valor nominal
0	"-113dBm o menor.
1	"-111 dBm
2...30	"-109...-53dBm
31	"-51dBm o mayores
99	No detectada

BER: Esta variable nos expresa la relación entre los bits errados y la totalidad de los bits recibidos. Esta variable se encuentra entre el 0.14% y el 18.10% según las especificaciones técnicas de GSM. En la figura 4.6 se presenta una captura del nivel de señal del transceptor.



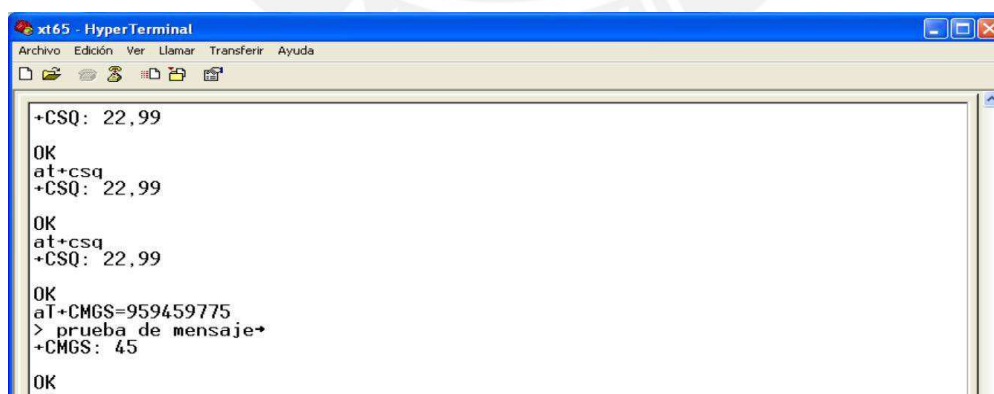
```
xt65 - HyperTerminal
Archivo  Edición  Ver  Llamar  Transferir  Ayuda
OK
at+csq
+CSQ: 20,99
OK
at+csq
+CSQ: 20,99
OK
at+csq
+CSQ: 20,99
```

**Figura 4.6. Captura del nivel de señal del transceptor en el laboratorio V101
[Elaboración propia]**

Como se puede notar en la captura el nivel de BER es muy alto, lo cual se consultó con el fabricante y nos indicó que el equipo presentaba un error en la toma de este parámetro; no obstante, en la siguiente prueba (Figura 4.7) de envío de mensaje de texto se puede notar que los mensajes están siendo recibidos correctamente

Envío de mensaje de texto.

Luego de haber constatado la correcta señal de la red GSM, se puede comenzar hacer las pruebas del envío de mensaje de texto (figura 4.7). Para lo cual se utilizó el comando AT: “AT+CMGS=numero”.



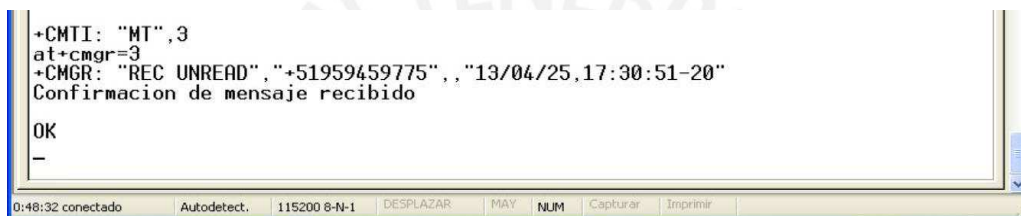
```
xt65 - HyperTerminal
Archivo  Edición  Ver  Llamar  Transferir  Ayuda
+CSQ: 22,99
OK
at+csq
+CSQ: 22,99
OK
at+csq
+CSQ: 22,99
OK
aT+CMGS=959459775
> prueba de mensaje→
+CMGS: 45
OK
```

Figura 4.7. Captura del envío de un mensaje de texto. [Elaboración propia]

+CMGS: 45, nos determina la dirección de memoria en donde se encuentra almacenado el mensaje enviado. En este caso el valor es 45.

Lectura de un mensaje recibido.

El transceptor nos permite leer mensajes almacenados tanto en la memoria del equipo como en la memoria del chip SIM a través del comando $AT+CMGR="índice"$ (figura 4.8), donde "índice" es la dirección de memoria en donde se encuentra el mensaje que deseamos leer. Se puede saber aquel "índice" debido a que el transceptor nos enviará a través del hyperterminal: $+CMTI: "MT",3$. Donde "3" nos indica la posición de memoria que ocupa el mensaje recibido.

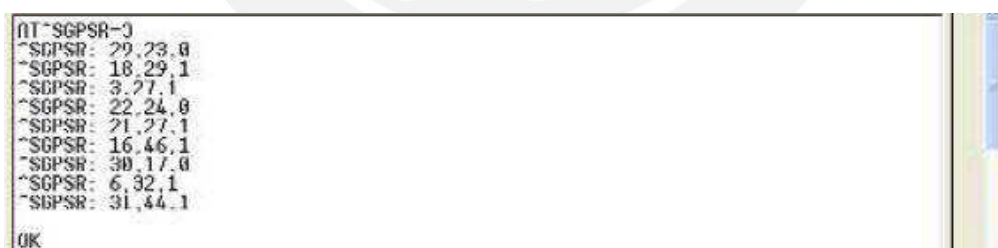


```
+CMTI: "MT",3
at+cmgr=3
+CMGR: "REC UNREAD", "+51959459775", "13/04/25,17:30:51-20"
Confirmacion de mensaje recibido
OK
-
```

Figura 4.8. Captura de la lectura de un mensaje. [Elaboración propia]

4.2.2. Módulo de GPS.

Para poder realizar las pruebas de posicionamiento se tiene que habilitar al GPS interno con el comando: $AT^SGPSR=1,0$. Posteriormente se verifica el número de satélites sincronizados, con el comando: $AT^SGPSR=3$ (figura 4.9).

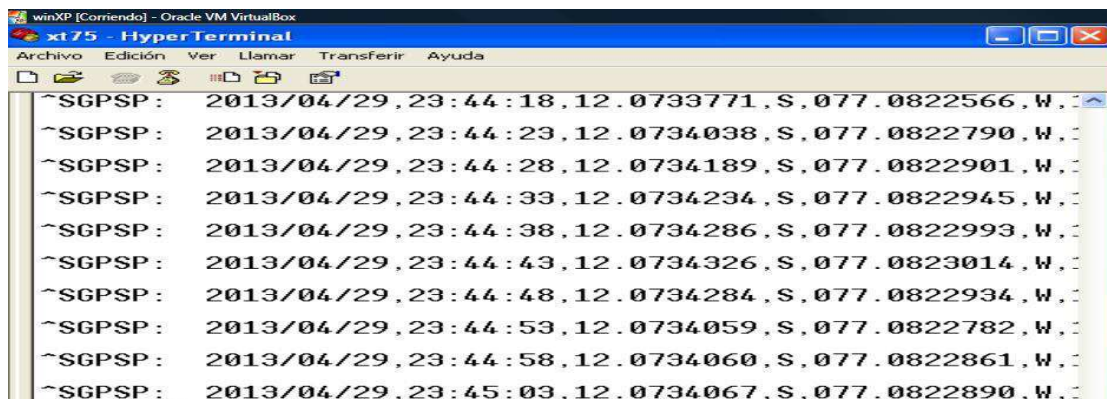


```
AT^SGPSR=3
^SGPSR: 29,23,0
^SGPSR: 18,29,1
^SGPSR: 3,27,1
^SGPSR: 22,24,0
^SGPSR: 21,27,1
^SGPSR: 16,46,1
^SGPSR: 30,17,0
^SGPSR: 6,32,1
^SGPSR: 31,44,1
OK
```

Figura 4.9. Captura de la sincronización de satélites con el transceptor [Elaboración propia]

Los transceptores enviarán coordenadas geodésicas a la Central de Monitoreo cada cierto tiempo con el comando $AT^SGPSP=Tiempo$, donde "Tiempo" es la variable del periodo que configura la frecuencia con la que se enviarán las coordenadas. La prueba se realizó en el

laboratorio V101 de la PUCP, por lo que las coordenadas que se muestran corresponden a dicho recinto. Además se configuró el tiempo a 5 segundos, como se aprecia en la figura 4.10.



```
winXP [Corriendo] - Oracle VM VirtualBox
xt75 - HyperTerminal
Archivo Edición Ver Llamar Transferir Ayuda
^SGPSP: 2013/04/29,23:44:18,12.0733771,S,077.0822566,W,...
^SGPSP: 2013/04/29,23:44:23,12.0734038,S,077.0822790,W,...
^SGPSP: 2013/04/29,23:44:28,12.0734189,S,077.0822901,W,...
^SGPSP: 2013/04/29,23:44:33,12.0734234,S,077.0822945,W,...
^SGPSP: 2013/04/29,23:44:38,12.0734286,S,077.0822993,W,...
^SGPSP: 2013/04/29,23:44:43,12.0734326,S,077.0823014,W,...
^SGPSP: 2013/04/29,23:44:48,12.0734284,S,077.0822934,W,...
^SGPSP: 2013/04/29,23:44:53,12.0734059,S,077.0822782,W,...
^SGPSP: 2013/04/29,23:44:58,12.0734060,S,077.0822861,W,...
^SGPSP: 2013/04/29,23:45:03,12.0734067,S,077.0822890,W,...
```

Figura 4.10. Captura del envío de coordenadas cada 5 seg. [Elaboración propia]

4.3. Pruebas y resultados del sistema.

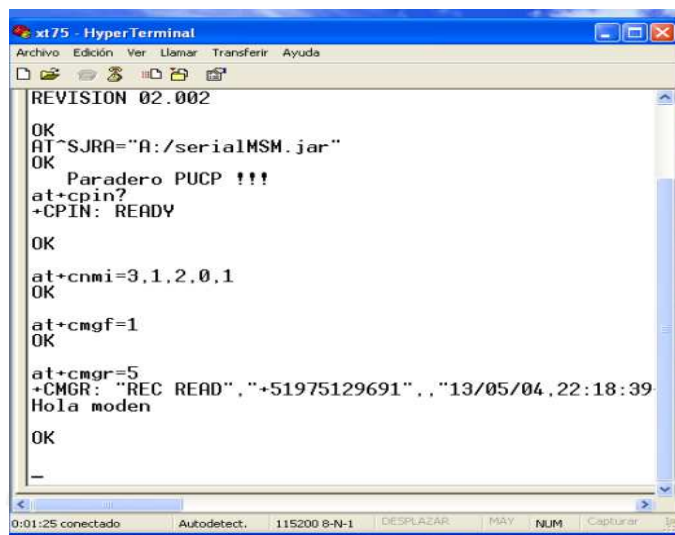
4.3.1. Pruebas del paradero

Las pruebas realizadas para el paradero consistieron básicamente en la recepción de los mensajes de texto y en la visualización de datos en la pantalla LCD. Para esta prueba se enviaron mensajes de texto al equipo transceptor del paradero, el cual retransmitía el mensaje al microcontrolador a través de su puerto serial, quien decodificaba el mensaje y modificaba el texto en la pantalla LCD. (Figura 4.11).



Figura 4.11. Transceptor integrado con el LCD [Elaboración propia]

Se ejecutó en el transceptor el programa serialMSN.jar, desarrollado en JAVA, creado para el paradero y se ejecutó el archivo .JAR mediante el comando AT: "AT ^SJRA=A:/serialMSN.jar. (Figura 4.12).



```
xt75 - HyperTerminal
Archivo Edición Ver Llamar Transferir Ayuda
REVISION 02.002
OK
AT^SJRA="A:/serialMSM.jar"
OK
Paradero PUCP !!!
at+cpin?
+CPIN: READY
OK
at+cnmi=3,1,2,0,1
OK
at+cmgf=1
OK
at+cmgr=5
+CMGR: "REC READ", "+51975129691", "13/05/04,22:18:39"
Hola moden
OK
_
```

Figura 4.12. Captura de cómo se ejecuta el aplicativo serialMSM.jar
[Elaboración propia]

El mensaje de bienvenida de la pantalla LCD se muestra en la figura 4.13.



Figura 4.13. Foto de la pantalla LCD con el mensaje de bienvenida.
[Elaboración propia]

Luego se envió el mensaje de texto (vía GSM) "m10", que de acuerdo al protocolo de comunicación propietario corresponde a "PROBLEMAS EN LA RUTA" (figura 4.14).



Figura 4.14. Pantalla LCD con el mensaje de “PROBLEMAS EN LA RUTA”.

[Elaboración propia]

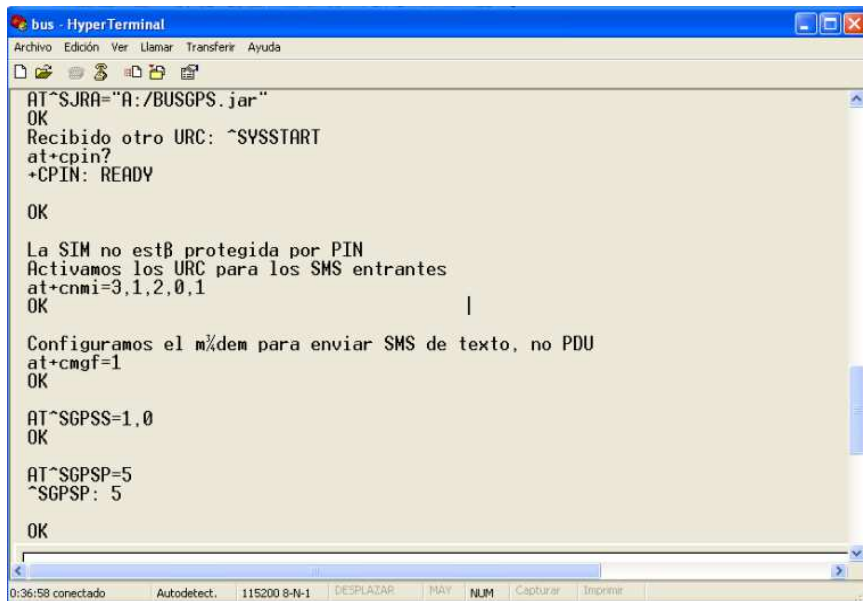
Con esta prueba se comprobó que el paradero recibe de forma efectiva los mensajes y actualiza el texto en la pantalla LCD

4.3.2. Pruebas del envío de coordenadas geodésicas del bus.

En primer lugar se configura uno de los transceptores (el que irá dentro del móvil) para que pueda trabajar en modo de envío de coordenadas geodésicas, para lo cual se ejecuta la aplicación JAVA: BUSGPS.jar (figura 4.15). Esta aplicación configura el transceptor con los parámetros de la red GSM y envía los comando AT para la activación del GPS interno del transceptor, además, esta aplicación permite enviar coordenadas geodésicas de una unidad hacia la central cada cierto periodo de tiempo; para nuestra prueba inicial el periodo fue configurado con una periodicidad de 5 segundos.

El programa fue desarrollado para ejecutarse en cuanto sea energizado, para ello se utiliza el comando AT: *SYSTART* que permite el autoarranque del programa BUSGPS.jar de forma automática.

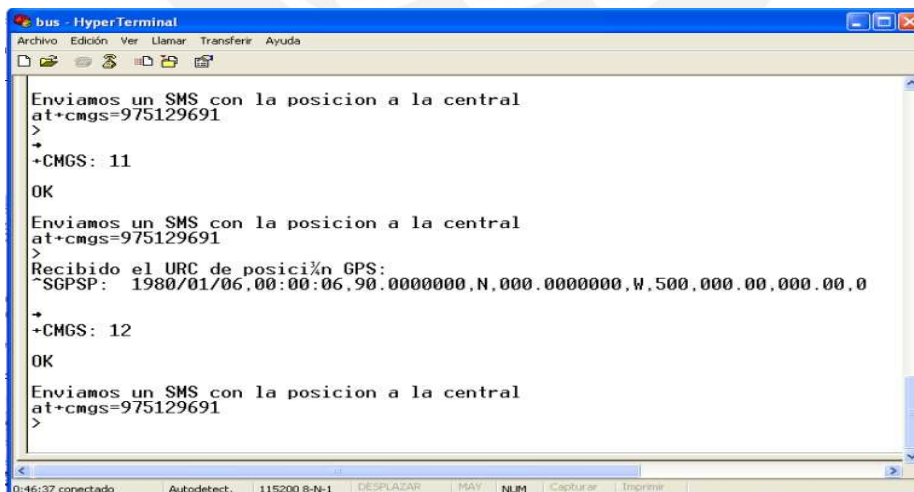
Cabe aclarar que si bien el transceptor no tiene ninguna interfaz dentro del móvil, para esta primera prueba, y para hacer captura del programa corriendo, se comunicó el transceptor por RS232 con una laptop, y a través del hyperterminal se hizo una captura del programa corriendo tal como se muestra en la figura 4.15.



```
bus - HyperTerminal
Archivo Edición Ver Llamar Transferir Ayuda
AT^SJRA="A:/BUSGPS.jar"
OK
Recibido otro URC: ^SYSSTART
at+cpin?
+CPIN: READY
OK
La SIM no está protegida por PIN
Activamos los URC para los SMS entrantes
at+cnmi=3,1,2,0,1
OK
Configuramos el m%dem para enviar SMS de texto, no PDU
at+cmgf=1
OK
AT^SGPSS=1,0
OK
AT^SGPSP=5
^SGPSP: 5
OK
0:36:58 conectado Autodetect. 115200 8-N-1 DESPLAZAR MAY NUM Capturar Imprimir
```

Figura 4.15. Captura de la ejecución de la aplicación BUSGPS.jar
[Elaboración propia]

Como se había mencionado anteriormente, la aplicación comienza a recibir las coordenadas de los satélites con los cuales se está sincronizando, para posteriormente reenviar estas coordenadas vía mensajes de texto hacia la central de monitoreo, que en este caso tiene el número 975129691 (figura 4.16), ya que hay un transceptor en la central, el cual recibe dichos mensajes.



```
bus - HyperTerminal
Archivo Edición Ver Llamar Transferir Ayuda
Enviamos un SMS con la posición a la central
at+cmgs=975129691
>
+
+CMGS: 11
OK
Enviamos un SMS con la posición a la central
at+cmgs=975129691
>
Recibido el URC de posición GPS:
^SGPSP: 1980/01/06,00:00:06.90.0000000,N,000.0000000,W,500.000.00.000.00,0
+
+CMGS: 12
OK
Enviamos un SMS con la posición a la central
at+cmgs=975129691
>
0:46:37 conectado Autodetect. 115200 8-N-1 DESPLAZAR MAY NUM Capturar Imprimir
```

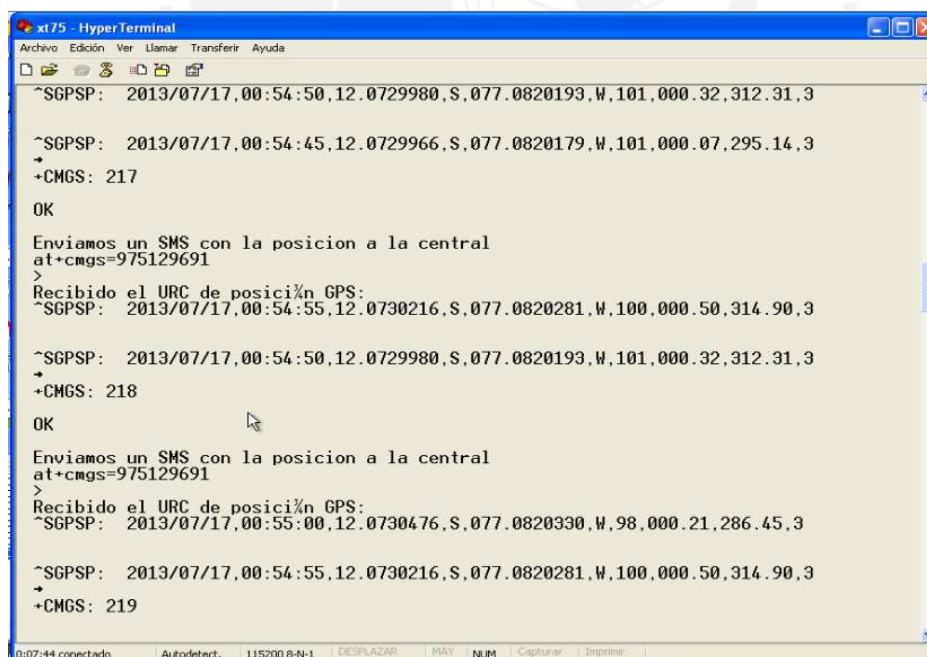
Figura 4.16. Prueba de envío de coordenadas [Elaboración propia]

Se puede notar que el transceptor nos muestra la coordenadas iniciales de sincronización con los satélites, asimismo, se visualiza los envíos de estas coordenadas hacia la central de monitoreo.

4.3.3. Resultados iniciales del sistema.

Cuando las unidades monitoreadas envían sus coordenadas a la central de monitoreo (figura 4.17), ésta última muestra al usuario la posición de dicha unidad en un mapa híbrido de google maps que además posee la opción de poder acceder a una vista satelital o vista convencional. A continuación se muestran las pruebas del sistema de monitoreo para la cual se hicieron las pruebas con un automóvil en las pistas traseras de la PUCP, cercanas al pabellón V. Tal y como se aclaró en el punto 4.3.2, el transceptor del móvil estaba conectado con una PC portátil para realizar la captura del programa corriendo dentro del transceptor mientras el móvil se encontraba en movimiento. De esta prueba se pudo capturar las coordenadas (figura 4.17), que se estaban enviando a razón de 5 segundos.

La frecuencia de envío se eligió de tal manera que la trayectoria en el mapa pueda ser más apreciable.



```
xt75 - HyperTerminal
Archivo Edición Ver Llamar Transferir Ayuda
~SGPSP: 2013/07/17,00:54:50,12.0729980,S,077.0820193,W,101,000.32,312.31,3
~SGPSP: 2013/07/17,00:54:45,12.0729966,S,077.0820179,W,101,000.07,295.14,3
+CMGS: 217
OK
Enviamos un SMS con la posicion a la central
at+cmgs=975129691
>
Recibido el URC de posici%n GPS:
~SGPSP: 2013/07/17,00:54:55,12.0730216,S,077.0820281,W,100,000.50,314.90,3
~SGPSP: 2013/07/17,00:54:50,12.0729980,S,077.0820193,W,101,000.32,312.31,3
+CMGS: 218
OK
Enviamos un SMS con la posicion a la central
at+cmgs=975129691
>
Recibido el URC de posici%n GPS:
~SGPSP: 2013/07/17,00:55:00,12.0730476,S,077.0820330,W,98,000.21,286.45,3
~SGPSP: 2013/07/17,00:54:55,12.0730216,S,077.0820281,W,100,000.50,314.90,3
+CMGS: 219
0:07:44 conectado Autodetect. 115200 8-N-1 DESPLAZAR MAY NUM Capturar Imprimir
```

Figura 4.17. Captura del envío de coordenadas hacia la central de monitoreo [Elaboración propia]

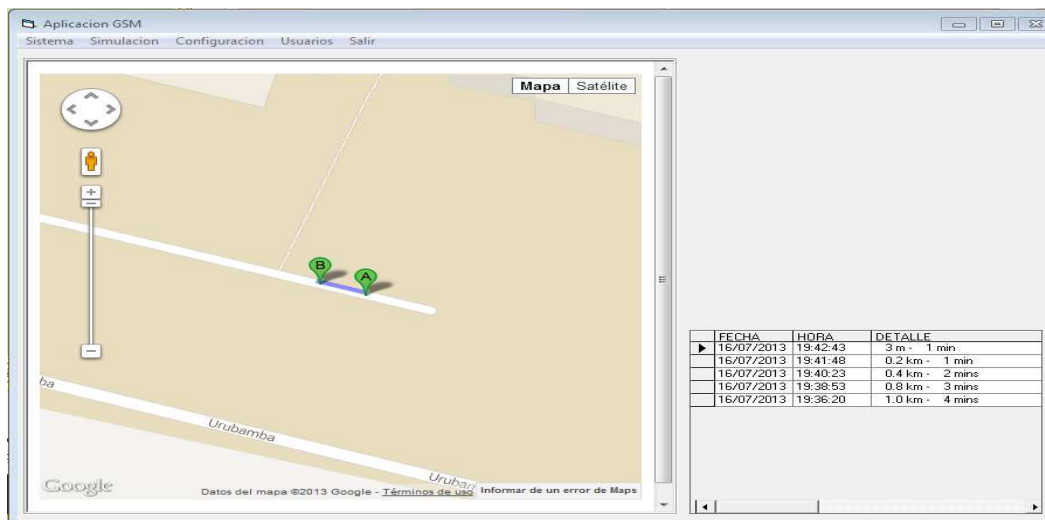


Figura 4.17. Captura de la interfase de usuario después de procesar los datos de las unidades monitoreadas. [Elaboración propia]

4.3.4. Resultados finales del sistema.

Como se había mencionado anteriormente, las pruebas del sistema de monitoreo se realizaron en dos ambientes distintos: las primeras se realizaron dentro de las instalaciones de PUCP; y las pruebas finales se hicieron en un tramo de ruta que cubre el recorrido entre el paradero de la PUCP (coordenadas en latitud y longitud respectivamente: -12.06798350690308° , -77.07791890632933°) y el paradero ubicado en la intersección de las avenidas Sánchez Carrión y Gregorio Escobedo (coordenadas: $-12.089742473332548^\circ$, -77.05830659401244°). Se escogió este tramo por su proximidad con la PUCP y porque el tráfico en su trayectoria presenta variaciones apreciables dependiendo de la hora del día, ello permitía que se pueda evaluar el sistema con condiciones diversas de tráfico.

Para estas pruebas, se realizaron 6 marchas durante un periodo de cuatro semanas en un intervalo de 2 horas entre marcha y marcha, esto permitió poder ver la efectividad del sistema y poder apreciar como mejoraba de semana en semana. Para ello se mostraran las pruebas entre semana y semana.

Semana 1:

Para estas pruebas nos tuvimos que subir a los buses de transporte público que recorrían la ruta mencionada en 6 intervalos distintos del día desde las 8:00 horas hasta las 20:00 horas entre el 30/09/2013 y el 25/10/2013. Una vez en el bus se encendía el transceptor

para poder iniciar la toma de datos y el envío de estos hacia la central tal como se muestra en la figura 4.18. Esta marcha se inició en el intervalo de 8 am – 10 am, configurando previamente el tranceptor con un periodo de envío de 60 segundos, esto nos permitió poder evaluar el programa en marcha y poder registrar la primera data en condiciones reales. En la figura 4.19 se puede apreciar la marcha ya concluida, con un tiempo real de recorrido de 12 minutos. En esta primera marcha no se tuvo un acierto en la estimación ya que el tiempo estimado del programa fue de 13 minutos; no obstante, a medida que se fue empleando el sistema, el sistema mejoró tal y como se muestra en la data recopilada de las semanas consecutivas.

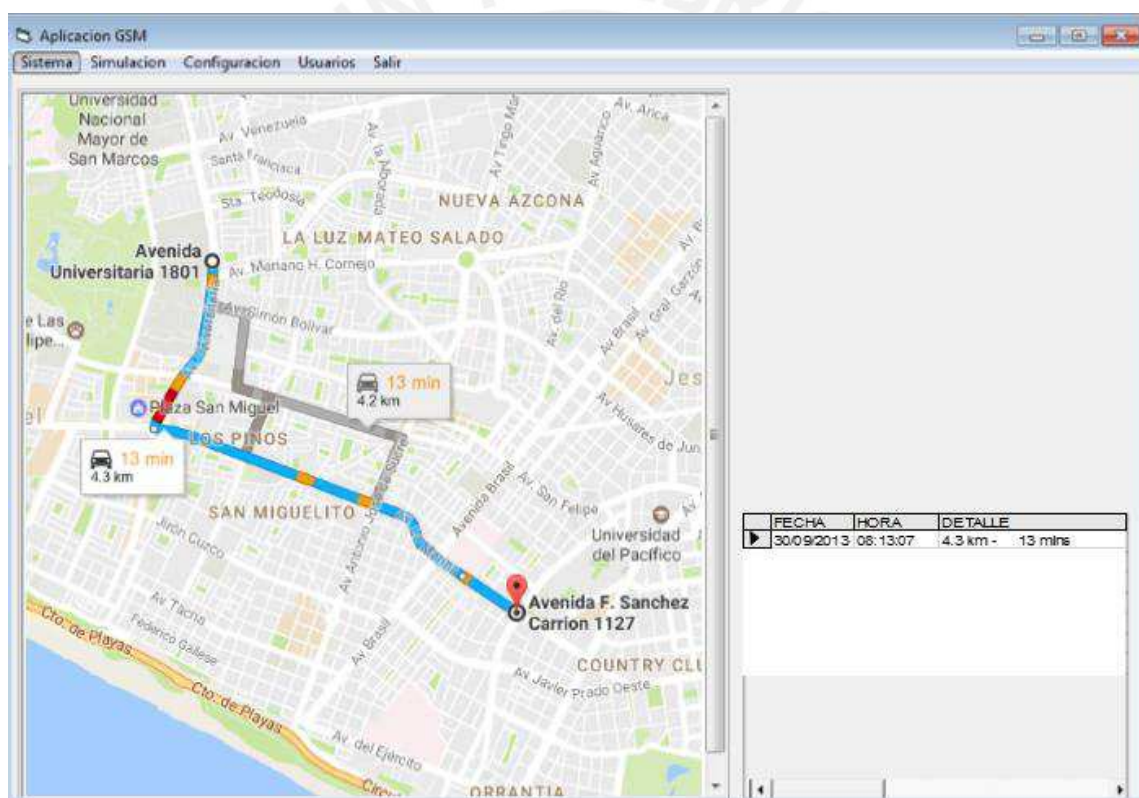


Figura 4.18. Captura de pantalla del programa ejecutándose en el inicio del recorrido. [Elaboración propia]

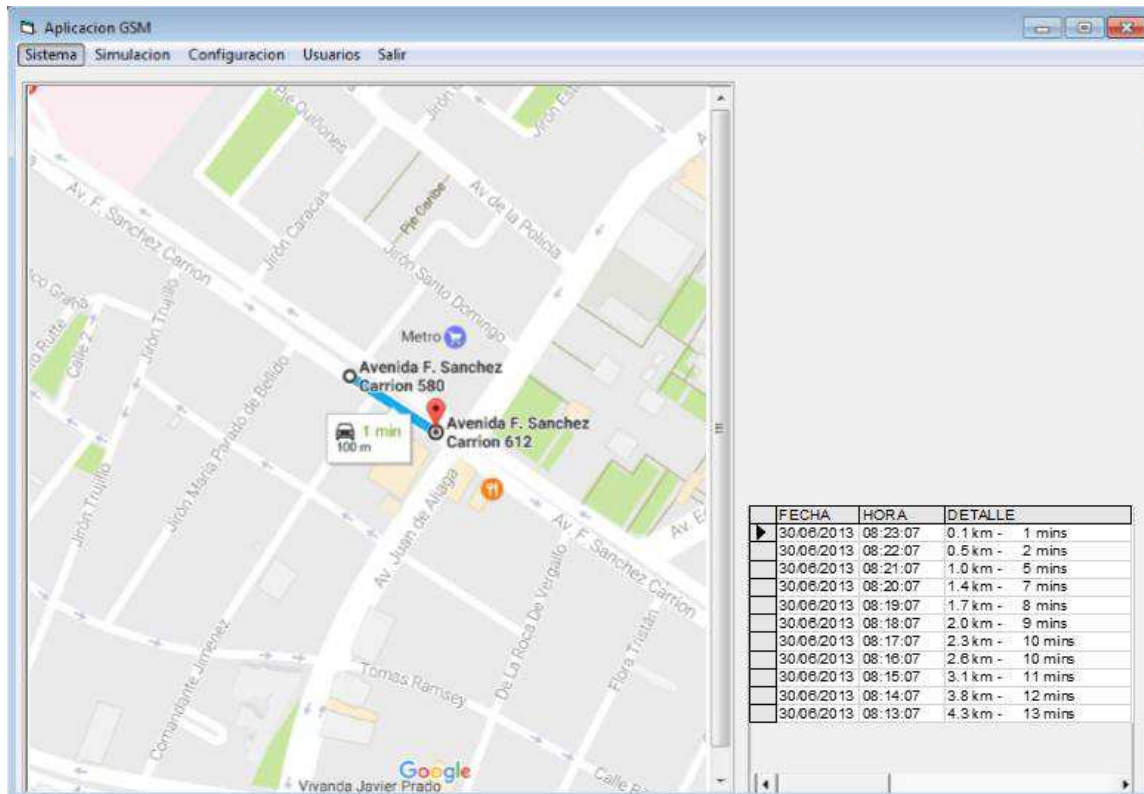


Figura 4.19. Captura de pantalla del programa ejecutándose al final del recorrido. [Elaboración propia]

Estas pruebas se tuvieron que realizar en 6 tiempos distintos durante el transcurso del día, y del mismo modo, se realizaron estas pruebas durante el transcurso de toda la semana, solo en los días hábiles ya que no consideramos los fines de semana dado que en estos días por lo general el tráfico no presentaba una tendencia homogénea. Una vez concluida estas pruebas, se procedió a recopilar la data y determinar la tasa de acierto de nuestro programa tal como se muestra en la tabla 4.2. De toda la data recopilada en la primera semana, se apreció una tasa de acierto muy baja, solo 2 de 30 intentos, lo cual nos da un porcentaje del 6.7%, a comparación de la estimación de google que tuvo 6 aciertos de 30, que en porcentaje es un 20%. Asimismo, se puede apreciar que hay una marcada diferencia entre los valores reales y los valores predichos por el sistema, por ende, se consideró tomar la suma de los valores absolutos de estos datos (columna "Dif. Con nuestro sistema" de la tabla 4.2), otorgándonos un total de 52 minutos totales, este valor se comparó con el obtenido en la cuarta semana, con la finalidad de verificar la eficacia del sistema.

Tabla 4.2. Datos recopilados en la primera semana de prueba.

[Elaboración propia]

Fecha	Día de la sem.	Tiempo google	Tiempo real	Tiempo estimado	Dif. con google	Dif. con nuestro sistema	Horario
30/09/2013	lunes	11	12	13	1	-1	8-10
01/10/2013	martes	13	14	15	1	-1	8-10
02/10/2013	miércoles	11	12	13	1	-1	8-10
03/10/2013	jueves	12	12	14	0	-2	8-10
04/10/2013	viernes	13	15	15	2	0	8-10
30/09/2013	lunes	15	15	17	0	-2	10-12
01/10/2013	martes	12	16	14	4	2	10-12
02/10/2013	miércoles	14	14	16	0	-2	10-12
03/10/2013	jueves	13	17	15	4	2	10-12
04/10/2013	viernes	14	13	16	-1	-3	10-12
30/09/2013	lunes	14	13	16	-1	-3	12-14
01/10/2013	martes	14	15	16	1	-1	12-14
02/10/2013	miércoles	11	15	13	4	2	12-14
03/10/2013	jueves	14	13	16	-1	-3	12-14
04/10/2013	viernes	13	16	15	3	1	12-14
30/09/2013	lunes	13	12	15	-1	-3	14-16
01/10/2013	martes	14	12	16	-2	-4	14-16
02/10/2013	miércoles	10	14	12	4	2	14-16
03/10/2013	jueves	13	13	15	0	-2	14-16
04/10/2013	viernes	13	15	15	2	0	14-16
30/09/2013	lunes	15	14	17	-1	-3	16-18
01/10/2013	martes	15	16	17	1	-1	16-18
02/10/2013	miércoles	14	14	16	0	-2	16-18
03/10/2013	jueves	15	15	17	0	-2	16-18
04/10/2013	viernes	11	14	13	3	1	16-18
30/09/2013	lunes	21	22	24	1	-2	18-20
01/10/2013	martes	22	24	26	2	-2	18-20
02/10/2013	miércoles	20	25	23	5	2	18-20
03/10/2013	jueves	21	26	25	5	1	18-20
04/10/2013	viernes	23	28	27	5	1	18-20

Semana 2:

En la segunda semana, se tuvo una mejora apreciable tal como se aprecia en la tabla 4.3, ya que el índice de acierto fue de 20% (6 de 30), mientras que en la tasa de google fue de 23.3% (7 de 30). Se resalta en gris los aciertos de nuestro sistema.

Tabla 4.3. Datos recopilados en la segunda semana de prueba.

[Elaboración propia]

Fecha	Día de la sem.	Tiempo google (min)	Tiempo real(min)	Tiempo estimado (min)	Dif. con google (min)	Dif. con nuestro sistema (min)	Horario (horas)
07/10/2013	Lunes	10	12	12	2	0	8-10
08/10/2013	Martes	15	15	17	0	-2	8-10
09/10/2013	miércoles	14	14	16	0	-2	8-10
10/10/2013	Jueves	12	13	14	1	-1	8-10
11/10/2013	Viernes	12	14	14	2	0	8-10
07/10/2013	Lunes	15	16	17	1	-1	10-12
08/10/2013	Martes	14	14	16	0	-2	10-12
09/10/2013	miércoles	14	14	16	0	-2	10-12
10/10/2013	Jueves	11	13	13	2	0	10-12
11/10/2013	Viernes	12	15	14	3	1	10-12
07/10/2013	Lunes	13	15	15	2	0	12-14
08/10/2013	Martes	12	16	14	4	2	12-14
09/10/2013	miércoles	14	15	16	1	-1	12-14
10/10/2013	Jueves	15	14	17	-1	-3	12-14
11/10/2013	Viernes	14	13	16	-1	-3	12-14
07/10/2013	Lunes	12	11	14	-1	-3	14-16
08/10/2013	Martes	13	14	15	1	-1	14-16
09/10/2013	miércoles	11	16	13	5	3	14-16
10/10/2013	Jueves	11	16	13	5	3	14-16
11/10/2013	Viernes	12	12	14	0	-2	14-16
07/10/2013	Lunes	14	16	16	2	0	16-18
08/10/2013	Martes	11	15	13	4	2	16-18
09/10/2013	miércoles	11	12	13	1	-1	16-18
10/10/2013	Jueves	12	16	14	4	2	16-18
11/10/2013	Viernes	14	14	16	0	-2	16-18
07/10/2013	Lunes	24	27	28	3	-1	18-20
08/10/2013	Martes	22	23	25	1	-2	18-20
09/10/2013	miércoles	25	25	29	0	-4	18-20
10/10/2013	Jueves	23	28	27	5	1	18-20
11/10/2013	Viernes	23	27	27	4	0	18-20

Semana 3:

En la tercera se pudo apreciar otra mejora en la cantidad de aciertos tal como se aprecia en la tabla 4.4, el índice de acierto fue de 23.3% (7 de 30), mientras que en la tasa de google fue de 20% (6 de 30).

Tabla 4.4. Datos recopilados en la primera semana de prueba.

[Elaboración propia]

Fecha	Día de la sem.	Tiempo google (min)	Tiempo real (min)	Tiempo estimado (min)	Dif. con google (min)	Dif. con nuestro sistema (min)	Horario (horas)
14/10/2013	Lunes	12	15	14	3	1	8-10
15/10/2013	Martes	12	14	14	2	0	8-10
16/10/2013	miércoles	12	12	14	0	-2	8-10
17/10/2013	Jueves	13	10	15	-3	-5	8-10
18/10/2013	Viernes	11	16	13	5	3	8-10
14/10/2013	Lunes	13	16	15	3	1	10-12
15/10/2013	Martes	12	13	14	1	-1	10-12
16/10/2013	miércoles	13	13	15	0	-2	10-12
17/10/2013	jueves	12	15	14	3	1	10-12
18/10/2013	viernes	14	16	16	2	0	10-12
14/10/2013	lunes	12	13	14	1	-1	12-14
15/10/2013	martes	12	13	14	1	-1	12-14
16/10/2013	miércoles	13	17	15	4	2	12-14
17/10/2013	jueves	15	15	17	0	-2	12-14
18/10/2013	viernes	12	16	14	4	2	12-14
14/10/2013	lunes	12	16	14	4	2	14-16
15/10/2013	martes	11	16	13	5	3	14-16
16/10/2013	miércoles	13	13	15	0	-2	14-16
17/10/2013	jueves	11	13	13	2	0	14-16
18/10/2013	viernes	11	16	13	5	3	14-16
14/10/2013	lunes	11	17	12	6	5	16-18
15/10/2013	martes	11	12	12	1	0	16-18
16/10/2013	miércoles	12	12	14	0	-2	16-18
17/10/2013	jueves	11	15	12	4	3	16-18
18/10/2013	viernes	15	17	17	2	0	16-18
14/10/2013	lunes	23	25	26	2	-1	18-20
15/10/2013	martes	20	20	23	0	-3	18-20
16/10/2013	miércoles	26	29	29	3	0	18-20
17/10/2013	jueves	25	30	28	5	2	18-20
18/10/2013	viernes	27	30	30	3	0	18-20

Semana 4:

Y por último, en la cuarta semana se pudo apreciar una mejora considerable frente a la primera semana, ya que la tasa de aciertos, tal como se aprecia en la tabla 4.4, se incrementó a 30% (9 de 30), mientras que en la tasa de google fue de 10% (3 de 30). Esto demuestra, que nuestro programa mejora a medida que se vaya empleando, es decir que

se puede evidenciar el aprendizaje de nuestro programa; no obstante, cabe resaltar que la cantidad de datos tomados es una pequeña muestra, se espera que a medida que más datos se obtengan, el programa mejore el algoritmo de estimación de forma gradual. Asimismo, se realizó la suma los valores absolutos la columna “Dif. Con nuestro sistema” de la tabla 4.5, otorgándonos un total de 43 minutos totales, lo cual evidencia una mejora significativa frente a la primera semana.

Tabla 4.5. Datos recopilados en la primera semana de prueba.
[Elaboración propia]

Fecha	Día de la sem.	Tiempo google (min)	Tiempo real (min)	Tiempo estimado (min)	Dif. con google (min)	Dif. con nuestro sistema (min)	Horario (horas)
21/10/2013	Lunes	12	12	14	0	-2	8-10
22/10/2013	Martes	12	13	13	1	0	8-10
23/10/2013	miércoles	13	15	15	2	0	8-10
24/10/2013	Jueves	12	13	14	1	-1	8-10
25/10/2013	Viernes	12	14	14	2	0	8-10
21/10/2013	Lunes	15	17	17	2	0	10-12
22/10/2013	Martes	14	13	16	-1	-3	10-12
23/10/2013	miércoles	12	13	14	1	-1	10-12
24/10/2013	Jueves	13	17	15	4	2	10-12
25/10/2013	Viernes	13	17	15	4	2	10-12
21/10/2013	Lunes	15	13	17	-2	-4	12-14
22/10/2013	Martes	12	13	13	1	0	12-14
23/10/2013	miércoles	13	16	15	3	1	12-14
24/10/2013	Jueves	12	14	14	2	0	12-14
25/10/2013	Viernes	15	15	17	0	-2	12-14
21/10/2013	Lunes	14	16	16	2	0	14-16
22/10/2013	Martes	12	15	14	3	1	14-16
23/10/2013	miércoles	12	13	14	1	-1	14-16
24/10/2013	Jueves	14	13	16	-1	-3	14-16
25/10/2013	Viernes	11	16	12	5	4	14-16
21/10/2013	Lunes	14	17	16	3	1	16-18
22/10/2013	Martes	14	12	16	-2	-4	16-18
23/10/2013	miércoles	13	16	15	3	1	16-18
24/10/2013	Jueves	15	17	17	2	0	16-18
25/10/2013	Viernes	14	15	16	1	-1	16-18
21/10/2013	Lunes	24	25	27	1	-2	18-20
22/10/2013	Martes	21	24	24	3	0	18-20
23/10/2013	miércoles	26	27	29	1	-2	18-20
24/10/2013	jueves	24	24	27	0	-3	18-20
25/10/2013	viernes	22	27	25	5	2	18-20

4.4. Evaluación de proyecto de inversión

El proyecto que se plantea a continuación tiene por nombre “*Implementación de un sistema de gestión tecnológico en el sector Transporte Urbano*”. El presente proyecto tiene como finalidad ofrecer herramientas tecnológicas a la municipalidad de lima con el fin de poder tener una gestión sofisticada y sobre todo a la vanguardia de las grandes metrópolis del mundo. Procederemos a la evaluación financiera del proyecto, ya que el marco teórico, detalles técnicos y composición del sistema han sido detallados previamente en el presente trabajo de tesis. Para evaluar el proyecto procederemos a aplicar criterios de evaluación que permitirán a los inversionistas determinar las bondades del mismo o, en caso contrario, descartarlo. Estos dos criterios son los siguientes:

4.4.1. VAN y TIR

El VAN nos permite determinar el valor presente del proyecto, mediante un valor monetario que es el resultado de la resta de los flujos descontados a la inversión inicial. Compara todos los Ingresos y Egresos esperados del proyecto un solo momento del tiempo. En otras palabras, mide la contribución neta, en valor presente de todos los flujos del proyecto [25]. Para proceder a la evaluación debemos estimar el costo oportunidad del accionista (K_e) o el costo ponderado de capital (WACC). Procederemos a aplicar el modelo de valuación de activos de capital (CAPM). Calcularemos la tasa de descuento aplicable a un proyecto (en términos reales) y se le suma el riesgo país [26]. Emplearemos la siguiente fórmula:

$$K_e = R_f + \beta_{\text{proy}} * [\text{Prima de Riesgo}] + RP$$

Dónde:

- K_e : Costo de capital propio
- R_m : Rendimiento esperado del mercado.
- R_f : Tasa libre de riesgo.
- $R_m - R_f$: Prima de riesgo
- β_{proy} : Sensibilidad en el rendimiento de un activo con respecto al rendimiento del mercado (beta)
- RP : Riesgo país Perú.

El primer paso es hallar el β_{proy} . Empezaremos con un β des-apalancado del rubro de las firmas del sector de Tecnologías de la Información (IT services) para el mercado estadounidense (figura 4.20).

Industry Name	Number of Firms	Average Beta	Market D/E Ratio	Tax Rate	Unlevered Beta	Cash/Firm value	Unlevered Beta corrected for cash
Funeral Services	6	1.12	49.48%	28.66%	0.83	2.94%	0.85
Furn/Home Furnishings	32	1.63	22.06%	16.69%	1.37	6.52%	1.47
Healthcare Information	20	0.97	11.62%	20.31%	0.88	10.06%	0.98
Heavy Truck & Equip	23	1.80	47.30%	22.74%	1.32	9.15%	1.45
Homebuilding	22	1.55	49.37%	7.12%	1.06	15.14%	1.25
Hotel/Gaming	57	1.65	44.75%	17.52%	1.21	6.43%	1.29
Household Products	27	0.98	18.11%	24.66%	0.86	2.52%	0.88
Human Resources	25	1.38	10.80%	26.61%	1.28	12.49%	1.46
Industrial Services	136	0.97	36.56%	20.19%	0.75	9.21%	0.83
Information Services	38	1.25	38.46%	18.33%	1.01	3.48%	1.05
Insurance (Life)	32	1.44	55.85%	21.09%	1.00	28.80%	1.41
Insurance (Prop/Cas.)	62	0.85	21.48%	10.73%	0.71	20.86%	0.90
Internet	194	1.17	2.29%	8.43%	1.15	12.04%	1.31
Investment Companies	31	1.27	7.65%	2.26%	1.18	2.64%	1.21
IT Services	63	1.06	5.72%	16.27%	1.00	10.31%	1.11
Machinery	84	1.26	17.39%	22.73%	1.11	6.04%	1.18
Maritime	51	1.51	181.21%	7.92%	0.57	6.95%	0.60
Med Supp Invasive	87	0.87	16.46%	12.60%	0.76	7.36%	0.82
Med Supp Non-Invasive	143	1.07	12.89%	10.61%	0.96	13.33%	1.10
Medical Services	118	0.84	50.23%	17.72%	0.59	10.82%	0.66
Metal Fabricating	25	1.63	23.59%	23.30%	1.38	11.91%	1.56
Metals & Mining (Div.)	77	1.62	15.42%	11.24%	1.42	7.45%	1.54
Natural Gas (Div.)	31	1.28	41.88%	22.02%	0.97	4.36%	1.01
Natural Gas Utility	27	0.46	66.15%	28.80%	0.31	2.41%	0.32
Newspaper	14	1.86	39.29%	18.35%	1.41	5.91%	1.50
Office Equip/Supplies	22	1.43	72.14%	22.66%	0.91	13.12%	1.05
Oil/Gas Distribution	12	1.02	53.39%	18.11%	0.71	1.26%	0.72
Oilfield Svcs/Equip.	81	1.66	27.72%	18.20%	1.35	6.40%	1.45
Packaging & Container	27	1.20	56.17%	23.09%	0.84	5.10%	0.88
Paper/Forest Products	32	1.37	43.81%	11.43%	0.99	7.35%	1.07
Petroleum (Integrated)	26	1.17	28.47%	30.34%	1.02	7.03%	1.10

Figura 4.20. Betas por sector en el mercado Estadounidense. [27]

Este valor (1.00) deberá ser apalancado tomando la consideración de aporte de los accionistas (municipalidad) de un 65 % del capital y un préstamo a 5 años para completar el saldo de la inversión del proyecto.

$$\beta_{\text{proy}} = (1 + (35\%/65\%) * (1-30\%)) * (1.00)$$

$$\beta_{\text{proy}} = 1.377$$

Ahora determinaremos los parámetros R_f y $R_m - R_f$. Para hallar el valor de R_f podemos recurrir al reporte de rendimiento de un bono del tesoro americano a un plazo de 5 años (figura 4.21), cuyo valor es 1.637%. La prima por riesgo de mercado asciende a 9.76% [28].

Hallamos el COK del proyecto:

$$\text{COKproy.} = 1.637\% + 1.377 * 9.76\%$$

$$\text{COKproy.} = 15.08\%$$

Estados Unidos »

Curva de rendimiento



Figura 4.21. Tasa de riesgo de inversión. [29]

Hemos hallado el valor del COK del proyecto, sin embargo, esta tasa es la que pedirían los accionistas si el proyecto se llevara a cabo en los Estados Unidos. Por tanto, ajustaremos la tasa considerando el riesgo país del Perú (figura 4.22). Por ello consultamos el riesgo país del Perú, obteniendo un valor de 149 pbs (puntos básicos), ello llevado a porcentaje nos ofrece un porcentaje de 1.49%.



Figura 4.22. Riesgo país del Perú. [30]

$$\text{COKproy.} = 15.08\% + \text{RP (riesgo país)}$$

$$\text{COKproy.} = 15.08\% + 1.49\% = 16.57\%$$

Con este valor es claro que podemos obtener el costo ponderado de capital (WACC), para ello debemos considerar el apalancamiento con financiamiento de una entidad bancaria (deuda), para ello es preciso considerar un préstamo a un plazo de 5 años, con una tasa de 11,61, valor que se puede visualizar en la página de la SBS. Consideraremos la siguiente relación:

$$\text{WACC} = (1-T) * i * D / (E+D) + \text{COKproy} * E / (E+D)$$

i: interés

E: Fondos Propios

D: Deuda Financiera

T: Tasa impositiva (impuesto a la renta)

$$\text{WACC} = (1-28\%) * 11.61\% * 35\% + 16.74\% * 65\%$$

$$\text{WACC} = 13.8\%$$

Con los valores de costo oportunidad del capital (COK) y el coste promedio ponderado del capital (WACC) procederemos a realizar el análisis financiero para un proyecto de inversión a 5 años; con este análisis podremos determinar la rentabilidad del mismo gracias a indicadores como el valor actual neto (VAN) y la tasa interna de retorno (TIR).

4.4.2. Estado financiero de flujo de efectivo

El proyecto a evaluar consiste en la implementación de un plan de negocio que involucra el monitoreo de unidades de transporte público en la ciudad de Lima. Para ello, el negocio se concentra en 5 puntos principales:

- Venta de transceptores.
- Instalación de transceptores.
- Gestión de información para monitoreo de unidades de transporte.
- Venta de hardware de paraderos.
- Instalación de hardware de paraderos.

El primer paso para analizar el proyecto es determinar los costos asociados a cada una de estas actividades, para ello debemos considerar todos los costos involucrados de forma directa e indirecta en los que se incurran, de esta forma, con el correcto registro de los costos, podremos valorizar y determinar las tarifas unitarias que deberemos cobrar por cada unidad para poder obtener un margen positivo de utilidad. El parque automotor de unidades de transporte al 2015 bordeaba las 14,840 unidades [31], este valor es importante para la estimación ya que se analizará el proyecto considerando una pequeña parte de este total de unidades (aproximadamente el 30%). Se considerará un crecimiento constante y lineal a lo largo de 60 meses, es por ello que se estima un vegetativo mensual de 84 unidades de transporte (aproximadamente un 0.5% del total) en las que se instarán transceptores y se integrarán a la red de monitoreo. Asimismo, no se tiene un registro exacto de la cantidad de paraderos en Lima, pero se considerará un vegetativo mensual de 9 paraderos en los que se instalará sistemas de visualización de información de buses. De esta forma, se obtiene el stock descrito en la tabla 4.6.

Tabla 4.6. Stock de transceptores y paraderos (montos en USD). [Elaboración

	Mes 12	Mes 24	Mes 36	Mes 48	Mes 60
# Transceptores					
Ingreso Vegetativo mensual	84	84	84	84	84
Stock Final	1,008	2,016	3,024	4,032	5,040
# Paraderos					
Ingreso Vegetativo mensual	9	9	9	9	9
Stock Final	108	216	324	432	540

Los costos asociados a cada unidad del negocio se describen en el **Anexo 7**, de forma sintetizada se muestran en la tabla 4.7.

Tabla 4.7. Tabla general de costos (montos en USD). [Elaboración propia]

	Mes 12	Mes 24	Mes 36	Mes 48	Mes 60
Transceptores					
Pu Transceptores	28,224	28,224	28,224	28,224	28,224
Costo de Importación	4,234	4,234	4,234	4,234	4,234
Antenas	15,120	15,120	15,120	15,120	15,120
Remuneración personal	5,727	5,727	5,727	5,727	5,727
Instalación Transceptores					
Suministros eléctricos	15,120	15,120	15,120	15,120	15,120
Remuneración personal	16,364	16,364	16,364	16,364	16,364
Gestión Información					
Plan tarifario	117,936	335,664	553,392	771,120	988,848
Reportes	32,760	93,240	153,720	214,200	274,680
Remuneración personal	8,864	17,045	33,409	41,591	57,273
Hardware - Paraderos					
Pantallas LCD	43,200	43,200	43,200	43,200	43,200
Dispositivos varios	5,400	5,400	5,400	5,400	5,400
Otros	2,160	2,160	2,160	2,160	2,160
Remuneración personal	2,455	2,455	2,455	2,455	2,455
Instalación HW paradero					
Remuneración personal	8,182	8,182	8,182	8,182	8,182
Costos Comunes					
Alquiler de oficina	36,000	36,000	36,000	36,000	36,000
Agua	436	436	436	436	436
Energía	6,000	6,000	6,000	6,000	6,000
TOTAL	348,181	634,571	929,142	1,215,532	1,509,422

Asimismo, es preciso considerar los equipos contemplados en la inversión inicial, la reinversión en el transcurso de los 5 años y la depreciación respectiva de estos activos. El

detalle de inversiones se detalla en el **anexo 7**, de forma sintetizada se muestra en la tabla 4.8.

Tabla 4.8. Cuadro general de inversiones (montos en USD). [Elaboración propia]

Inversión	AÑO 0	AÑO 1	AÑO 2	AÑO 3	AÑO 4	AÑO 5
Servidor CPU 1T	6,000	2,000	2,000	2,000	2,000	2,000
Cisco ASA	2,000	-	-	3,000	-	-
Switch	3,600	-	-	5,400	-	-
Workstations	15,000	-	-	22,500	-	-
Monitores	7,000	-	-	10,500	-	-
Gabinete	1,800	-	-	1,200	-	-
Mobiliario	10,000	-	-	10,000	-	-
Total	45,400	2,000	2,000	54,600	2,000	2,000

Se considera la depreciación de activos de forma lineal, se muestra el detalle de forma resumida en la tabla 4.9.

Tabla 4.9. Tabla general de costos (montos en USD). [Elaboración propia]

Inversión	AÑO 0	AÑO 1	AÑO 2	AÑO 3	AÑO 4	AÑO 5
Servidor CPU 1T		600	800	1,000	1,200	1,400
Cisco ASA		667	667	667	1,000	1,000
Switch		1,200	1,200	1,200	1,800	1,800
Workstations		5,000	5,000	5,000	7,500	7,500
Monitores		2,333	2,333	2,333	3,500	3,500
Gabinete		360	360	360	600	600
Mobiliario		2,000	2,000	2,000	4,000	4,000
Total		12,160	12,360	12,560	19,600	19,800

Ahora que tenemos los costos asociados, determinaremos las tarifas respectivas para los productos y servicios que se brindarán, para ello distribuiremos todos los costos en los diferentes servicios y los dividiremos entre las cantidades totales para determinar el costo unitario por servicio y equipo. El análisis del primer año procede conforme a la tabla 4.10.

Tabla 4.10. Tabla general de costos (montos en USD). [Elaboración propia]

	Transceptores	Instalación	Gestión Información	Harw.Paraderos	Harw.Paraderos Instal.
COSTO VARIABLE	47,578	15,120	150,696	50,760	
Otros COSTOS FIJOS	8,487	8,487	8,487	8,487	8,487
PERSONAL	5,727	16,364	8,864	2,455	8,182
DEPRECIACIÓN	1,867	1,867	4,693	1,867	1,867
OTROS	2,238	1,512	6,486	2,268	713
TOTAL CF + CV	65,897	43,350	179,227	65,836	19,249
GA&GV	5,967	4,032	17,297	6,048	1,901
Q - Anual (unidades)	1,008	1,008	6,552	108	108
COSTO UNITARIO	USD 71	USD 47	USD 30	USD 666	USD 196
TARIFA UNITARIA	USD 74	USD 50	USD 33	USD 700	USD 220

Análisis de estado de flujo

Una vez que tenemos las tarifas unitarias, se pueden obtener las ventas totales bajo una relación PxQ (precio unitario x cantidad). El resultado de los ingresos se refleja en la tabla 4.11.

Tabla 4.11. Tabla ingresos (montos en USD). [Elaboración propia]

	AÑO 1	AÑO 2	AÑO 3	AÑO 4	AÑO 5
Ventas Transceptores	74,592	74,592	74,592	74,592	74,592
Ventas Servicio Instalación	50,400	50,400	50,400	50,400	50,400
Ventas Gestión Información	216,216	615,384	1,014,552	1,413,720	1,812,888
Ventas por Hardware - Paraderos	75,600	75,600	75,600	75,600	75,600
Ventas Servicio Instalación HW paradero	23,760	23,760	23,760	23,760	23,760
TOTAL	440,568	839,736	1,238,904	1,638,072	2,037,240

Una vez definidos los ingresos y egresos, lo primero que determinaremos es el margen de contribución, que es la diferencia entre el precio de venta y los costos variables. Este

margen se detalla en la tabla 4.11. Este margen representa el exceso de ingresos con respecto a los costos variables. El análisis detallado puede revidarse en el anexo 7.

Tabla 4.12. Margen de contribución en periodo de 5 años en USD. [Elaboración propia]

Concepto	AÑO 1	AÑO 2	AÑO 3	AÑO 4	AÑO 5
VENTAS	440,568	839,736	1,238,904	1,638,072	2,037,240
COSTO VARIABLE	264,154	542,362	820,570	1,098,778	1,376,986
MARGEN DE CONTRIBUCION	176,414	297,374	418,334	539,294	660,254

Luego de determinar el margen de contribución determinaremos la ganancia bruta que representa la diferencia entre los ingresos totales y los costos directos asociados con la producción, dicho margen se contempla en la tabla 4.13.

Tabla 4.13. Utilidad bruta en periodo de 5 años en USD. [Elaboración propia]

Concepto	AÑO 1	AÑO 2	AÑO 3	AÑO 4	AÑO 5
Costos Fijos	109,404	129,761	158,300	185,497	213,354
Personal	41,591	49,773	66,136	74,318	90,000
Depreciación	12,160	12,360	12,560	19,600	19,800
Otros Costos Fijos	55,653	67,628	79,603	91,579	103,554
Utilidad Bruta	67,010	167,613	260,035	353,798	446,901

Ahora que ya se han contemplado los gastos fijos, se procede a determinar la utilidad operativa y utilidad antes de impuestos para poder obtener la utilidad neta. La utilidad operativa resulta luego de restar a la utilidad bruta los conceptos de gastos administrativos y gastos de ventas. Una vez contempladas estas utilidades, se procede a determinar la utilidad neta, que es la resultante de restar a la utilidad antes de impuestos los conceptos por participación del personal e impuesto a la renta (tabla 4.14).

Tabla 4.13. Utilidad neta en periodo de 5 años (montos en USD). [Elaboración propia]

Concepto	AÑO 1	AÑO 2	AÑO 3	AÑO 4	AÑO 5
GASTOS ADMINISTRATIVOS (5%Ventas)	22,028	41,987	61,945	81,904	101,862
GASTOS DE VENTAS (3%Ventas)	13,217	25,192	37,167	49,142	61,117
UTILIDAD OPERATIVA	53,593	142,221	222,667	304,456	385,584
UTILIDAD ANTES DE IMPUESTOS	53,793	142,421	222,867	304,656	385,784
PARTICIPACIONES (5% U. antes de imp.)	15,869	42,014	65,746	89,873	113,806
IMPUESTO A LA RENTA (29.5%)	2,690	7,121	11,143	15,233	19,289
UTILIDAD NETA	35,234	93,286	145,978	199,549	252,688

Una vez determinada la utilidad neta se procede a determinar el flujo financiero, para lo cual debemos considerar los ingresos con el 18% de concepto de IGV. Empezaremos determinando el superávit, que representa la cantidad en que los ingresos superan a los gastos; para este fin, se restará los sueldos, egresos y el pago de impuestos a los ingresos con IGV. Una vez determinado el superávit, le restamos la amortización y los intereses acumulados por la deuda bancaria, así como la recuperación del capital de trabajo al quinto año. De esta forma se obtiene el flujo financiero que se empleará para analizar el TIR y el VAN. El flujo financiero se detalla en la tabla 4.14.

Tabla 4.14. Flujo financiero en periodo de 5 años en USD. [Elaboración propia]

Concepto	AÑO 0	AÑO 1	AÑO 2	AÑO 3	AÑO 4	AÑO 5
Ingresos Operativos		519,870	990,888	1,461,907	1,932,925	2,403,943
Egresos Operativos		447,104	925,229	1,311,032	1,793,361	2,231,317
Costo Variable		311,701	639,987	968,272	1,296,558	1,624,843
Personal		63,619	91,760	128,082	156,222	191,862
Otros Costos Fijos		65,671	79,802	93,932	108,063	122,193
Otros Gadm&ventas		15,596	29,727	43,857	57,988	72,118
Imp. a la renta (30%)		15,869	42,014	65,746	89,873	113,806
Participaciones	-	2,690	7,121	11,143	15,233	19,289
IGV	-	-28,042	34,820	-	69,425	87,205
Superávit Comercial	-	72,766	65,659	150,874	139,564	172,626

Una vez obtenido el superávit, procederemos a armar el cuadro de flujos del proyecto, para ello se debe considerar el superávit previamente obtenido; las inversiones en los 5 años del proyecto; el capital de trabajo, que es el resultado de una fracción del ingreso operativo del año en progreso considerando los días de cobro a 60 días. El proyecto se desarrolla para continuar después de los 5 años iniciales, por ello es que es necesario considerar una perpetuidad, que básicamente consiste en el flujo del año anterior dividido entre el COK del proyecto (16.57%), a este valor se le aplicará un porcentaje de castigo del 60%. Finalmente se obtiene la tabla 4.15.

Tabla 4.15. Flujo de efectivo en periodo de 5 años (montos en USD). [Elaboración propia]

Concepto	AÑO 0	AÑO 1	AÑO 2	AÑO 3	AÑO 4	AÑO 5
Superávit Comercial	-	72,766	65,659	150,874	139,564	172,626
Inversiones	-140,217	-80,863	-80,863	-142,931	-80,863	-2,360
Activos	-53,572	-2,360	-2,360	-64,428	-2,360	-2,360
Capital de Trabajo	-86,645	-78,503	-78,503	-78,503	-78,503	
Perpetuidad						212,557
Flujo de Periodo	-140,217	-8,097	-15,204	7,943	58,701	382,824

Con los flujos claros, procedemos a evaluar el VPN considerando un COK igual al costo ponderado de capital (16.57%) se tiene: **VPN = \$ 56,308**

Así también estimamos la **TIR = 24.50%**

4.5. Conclusiones y recomendaciones

Conclusiones

Luego de desarrollada la implementación se llegaron a las siguientes conclusiones:

- Se demostró mediante la implementación del sistema y las pruebas realizadas que es posible la realización de un sistema de monitoreo, empleando comunicación GSM, posicionamiento satelital (GPS), procesamiento por software, gestión de base de datos y envío de información al usuario a través del paradero.
- Se verificó la recepción de información en el paradero de prueba, y por ende se concluye que es posible proveer información al usuario en tiempo real sobre el estado de la ruta.
- Se demostró que un sistema de estimación de tiempo de aproximación de un bus al paradero es realizable. Se pudo ajustar los valores con datos reales a fin de que la estimación sea lo más exacta posible. Se mejoró la tasa de acierto de estimación de 6.7% en la primera semana a 30% en la cuarta semana. Asimismo, se aprecia que el error promedio disminuye de la primera semana a la cuarta (1.8 min. a 1.43 min. respectivamente). Esto demuestra que el sistema tiene la capacidad de mejorar a medida que más se emplee.
- Se cotejó la funcionalidad de la integración del Hardware y software de monitoreo, se verificó los resultado del trabajo en conjunto y se pudo comprobar la eficacia del sistema, ya que la suma total de los valores absolutos de diferencia entre los valores reales y los valores predichos por nuestro sistema mejoró de la primera semana a la cuarta, de 52 min a 43 min respectivamente.
- Se pudo verificar la rentabilidad del proyecto de implementación tecnológica en el sector Transporte, con los valores de VPN y TIR se logró comprobar que un proyecto de inversión en referencia al modelo planteado ofrecerá rentabilidad a futuro.

Recomendaciones

- El transceptor MTX – 65 es un dispositivo funcional y práctico pero que por el hecho de no ser muy comercial en nuestro país posee periféricos con salidas poco convencionales, lo que ocasionó que al buscar antenas GSM y GPS no encontráramos estos elementos con facilidad. Por tanto se recomienda que en trabajos posteriores se proceda a importar con las antenas.

- El dispositivo MTX – 65 es compatible con versiones de Windows XP de 32 bits para su programación; sin embargo, esta no es la única plataforma de trabajo (pero sí la más fácil) ya que puede ser desarrollado en otros entornos como Windows Vista o 7. Se recomienda investigar con estos entornos, ya que es altamente probable que Windows XP se encuentre obsoleto dentro de algunos años.
- El transceptor se comunica con el computador a través del puerto serial RS232, por tanto se recomienda que toda computadora con la que se comunique posea este tipo de puerto; asimismo, se recomienda que la computadora tenga una conexión a internet de por lo menos 2 MB.
- El software de procesamiento ha sido desarrollado para una unidad en movimiento acercándose a un paradero; sin embargo el sistema está desarrollado para poder ser escalable y poder enviar la información a otros paraderos dentro de la ruta del bus. Se recomienda realizar las pruebas respectivas para la ampliación del sistema.
- El programa piloto ha sido desarrollado en Microsoft Visual Basic, no obstante, en un marco más empresarial, y para mayor alcance del sistema de procesamiento, se recomienda JAVA, puesto que posee gran cantidad de librerías y comunicación con diversos tipos de dispositivos.
- El sistema ha sido desarrollado para monitorear tiempos y distancias; sin embargo, debido a la versatilidad del transceptor y del software desarrollado, estos permiten tomar registro de variables adicionales como velocidad de buses, cantidad de pasajeros, etc. Por tanto se recomienda ampliar la robustez del software para procesar esta nueva data.
- El sistema trabajado ha sido desarrollado con tecnología GSM/GPRS por su facilidad y versatilidad, no obstante, la arquitectura del sistema puede ser compatible con tecnologías 3G y 4G, por lo que se recomienda el uso de estas tecnologías para una etapa posterior, también se recomienda incluso el uso de una red propietaria basadas en conexiones por fibra óptica, todo dependerá de la complejidad del sistema a elaborar.

BIBLIOGRAFIA

- [1] BANCO INTERAMERICANO DE DESARROLLO
Junio 2015 “Casos de estudio comparativo de tres proyectos de transporte urbano apoyado por el BID”. Oficina de Evaluación y Supervisión, Washington DC, Estados Unidos.
- [2] “Población 2000 al 2015” Instituto Nacional de Estadística e Informática INEI. Ingresado el 14 de septiembre de 2016. <
<http://proyectos.inei.gob.pe/web/poblacion/>>
- [3] JUAN CARLOS DEXTRE
“Ciudad, transporte y calidad de vida” PALESTRA portal de asuntos públicos de la PUCP, Lima, Perú.
- [4] “Densidad poblacional” Instituto Nacional de Estadística e Informática INEI. Ingresado el 14 de septiembre de 2016. <
https://www.inei.gob.pe/media/MenuRecursivo/publicaciones_digitales/Est/Lib0015/cap-512.htm>
- [5] JAMIE HOUGHTON, JOHN REINERS Y COLIN LIM.
Junio 2009 “transporte inteligente: Cómo mejorar la movilidad en las ciudades.” IBM Global Business Services. Route 100 Somers, NY 10589, U.S.A.
- [6] INEI
2013 “Compendio Estadístico del Perú”, Tomo N° 1, Lima, Perú.
- [7] DIRECCION GENERAL DE SALUD AMBIENTAL
2011 “Política Nacional de Salud Ambiental 2011 – 2020”, Ministerio de Salud, Lima, Perú.
- [8] “Concesión de rutas: Problemática”. Página oficial de la Gerencia de Transporte urbano. Ingresado el 15 de Octubre de 2012. <
http://www.gtu.munlima.gob.pe/proyectos/concesionrutas_problematika.htm>
- [9] JORGE YESHAYAHU GONZALES-LARA

“El Perú, el país con mayor accidentes de tránsito en América Latina”. La Diáspora – Latino Digital Magazine. Ingresado el 15 de Octubre de 2012. <<http://peruimmigrationdocumentationproject.blogspot.com/2010/06/el-peru-el-pais-con-mayor-accidentes-de.html>>

[10] LUIS CHÍA RAMÍREZ Y SANDRO HUAMANÍ ANTONIO

Setiembre 2010 “Accidentes de Tránsito en el Perú: Casualidad o causalidad” Ministerio de Transportes y comunicaciones, cuadernos de Infraestructura e inclusión social, Lima – Perú. Pg. 13.

[11] REDL SIEGMUND M.

1995 An Introduccion to GSM. Boston: Artech House Publisers

[12] THEODORE S. RAPPAPORT

2002 WIRELESS communications. Principles & practice. Segunda edición. Prentice Hall PTR.

[13] BENNETT, V.

1990 GPS program status report. In. Protecting Natural Resources wiht remote sensing. Proceedings of the Third Forest Service Remote Sensing Applications Conference-April 9-13, pp.334-338 American Society for Phtogrammetry and Remote Sensing. Bethesda, Maryland USA.

[14] BEUCHOT, MAURICIO

2007 “Introducción a las ciencias de la computación con Java”, Universidad Autónoma de México, 1ra edición

[15] ALFREDO WEITZENFELD

2005 “Ingeniería de Software Orientada a Objetos Con Uml, Java E Internet”, Cengage Learning Editores

[16] EDWARD V. RAMÍREZ, MELVYN WEISSPAG.

1986 “Introducción a los microprocesadores: Equipo y sistemas”, primera edición, Editorial LIMUSA, Mexico. pp183

- [17] JORGE ARMANDO DE JESÚS RAYAS ALEMÁN
2016, “Instalación y configuración del sistema gestor de bases de datos en distintas plataformas”, Guanajuato – México.
- [18] SASKIA HOLLBORN
2002 “Intelligent Transport Systems in Japan”, TECHNISCHE UNIVERSITÄT DARMSTAD, Tokio.
- [19] “Mechanisms of VICS information” Page of Highway Industry Development Organization of Japan, Ingresado el 20 de Septiembre de 2012. <http://www.hido.or.jp/ITSHP_e/08its/deve/vics1.html>
- [20] MICHAEL KEFERL
“Inside the Tokyo Traffic Control Center”, ShiftEast. Ingresado el 20 de Septiembre de 2012. <<http://www.shifteast.com/inside-the-tokyo-traffic-control-center/>>
- [21] “Curitiba, un ícono del transporte público”. Página oficial del diario La Nación. Ingresado el 15 de Julio de 2013. <<http://www.lanacion.com.ar/1063450-curitiba-un-icno-del-transporte-publico>>
- [22] CLÉVER UBIRATAN TEIXEIRA DE ALMEIDA
“La movilidad urbana en Curitiba”. Universidad de Córdoba. Ingresado el 15 de Julio de 2013. <<http://www.uco.edu.co/extension/territoriales/Pensamiento%20Regional/Documents/MOVILIDAD%20URBANA%20EN%20CURITIBA.pdf>>
- [23] CRISTINA GARCÍA CAMBRONERO, IRENE GÓMEZ MORENO
“Algoritmos de aprendizaje: KNN & KMEANS”, Universidad Carlos III de Madrid, España.
- [24] NATIONAL INSTRUMENTS
2007 GPS Receiver Testing. Consulta 3 de Agosto de 2013. <<http://www.ni.com/white-paper/7189/en/>>

[25] CARMEN QUIROZ

Marzo 2011 “Ingeniería económica”, Facultad de Ciencias e Ingeniería, sección Industrial – PUCP.

[26] PAUL LIRA BRICEÑO

“La tasa de descuento de un proyecto en la práctica” Diario Gestión. Ingresado el 10 de Julio de 2013. <<http://blogs.gestion.pe/deregresoalobasico/2012/03/la-tasa-de-descuento-de-un-pro.html>>

[27] “Betas by sector” página web de A. Damodaran. Ingresado el 15 de Julio de 2013. <http://pages.stern.nyu.edu/~adamodar/New_Home_Page/datafile/Betas.html>

[28] “S&P 500 INDEX”. CNN Money international. Ingresado el 15 de Agosto de 2017. <<http://money.cnn.com/data/markets/sandp/>>

[29] “Estados Unidos - Bonos del Estado”. Portal de Investing. Ingresado el 15 de Agosto de 2017. <https://es.investing.com/rates-bonds/usa-government-bonds?maturity_from=40&maturity_to=130>

[30] “Consulta a series estadísticas del BCRP”. Página oficial del BANCO CENTRAL DE RESERVA DEL PERÚ. Ingresado el 15 de Julio de 2013. <<https://estadisticas.bcrp.gob.pe/estadisticas/series/diarias/resultados/PD04709XD/html>>

[31] LIMA COMO VAMOS

2016 “CÓMO VAMOS EN MOVILIDAD, sexto informe de resultados sobre calidad de vida”. Lima, Perú.