

## Diseño e Implementación de un Sistema de Facturación en Pista para Estaciones de Servicio

José Guarnizo, Bernardo Vallejo, Darwin Alulema y Evelio Granizo

**Resumen**— El proyecto presenta un sistema de facturación en pista para estaciones de servicio, que consta de tres elementos: un dispositivo facturador basado en un PIC18F67J11, un módulo de comunicación basado en la plataforma Arduino, y un software basado en JAVA que administra la comunicación del sistema de facturación y opera la información de la base de datos del sistema.

**Palabras claves**— Microcontrolador, Arduino, Base de Datos, JAVA, Web Service.

### I. INTRODUCCIÓN

En agosto del 2013 el SRI emite la resolución 00382, la cual establece que las estaciones de servicio deben eliminar la figura de consumidor final en sus ventas y garantizar la correcta emisión de facturas. Entonces, Teleautomatismos V&P S.A., empresa ecuatoriana dedicada en la rama de Ingeniería Electrónica, con el fin dar solución a la problemática descrita anteriormente y evitar multas por el SRI a las estaciones de servicio, realiza el estudio de un Sistema de Facturación en Pista, que consta de tres elementos:

- Un dispositivo, que se llamará dispositivo facturador, el cual permite realizar las tareas de facturación y está formado por los siguientes elementos: un teclado alfanumérico, un LCD gráfico y una tarjeta de control.
- Un software desarrollado en lenguaje JAVA, que está instalado en el servidor de la estación, el cual permite la interacción entre los dispositivos: facturadores, la base de datos y las impresoras. Además, utiliza el web Service Merlyna para la consulta online del nombre del cliente a través de su cédula.
- Un módulo de comunicación, que se basa en la plataforma Arduino, el cual permite la comunicación entre el dispositivo facturador y el servidor.

### II. DESARROLLO DE HARDWARE

La distribución de todos los componentes que forman parte del sistema de facturación en pista desarrollado, se muestra en la Fig. 1.

J. Guarnizo, B. Vallejo. Carrera de Ingeniería en Eléctrica, Automatización y Control, Departamento de Eléctrica y Electrónica, Universidad de las Fuerzas Armadas ESPE, Sangolquí, Ecuador, e-mails: jguarnizo90@gmail.com y bernardojvm1989@gmail.com.

D. Alulema, E. Granizo. Departamento de Eléctrica y Electrónica, Universidad de las Fuerzas Armadas ESPE, Sangolquí, Ecuador, e-mails: doalulema@espe.edu.ec y egranizo@espe.edu.ec.

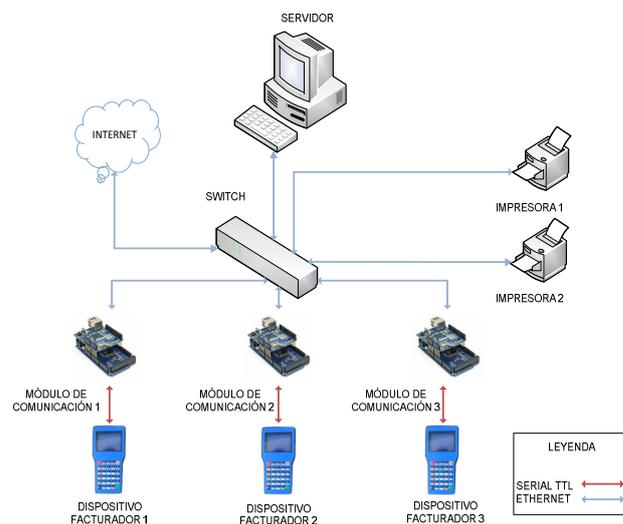


Fig. 1. Hardware del sistema de facturación en pista

A continuación, se detallan los dos elementos de hardware desarrollados: el dispositivo facturador y el módulo de comunicación.

#### A. Dispositivo Facturador

El dispositivo facturador es portable, de fácil operación, y permite el ingreso y la visualización de datos; sus principales componentes son:

- PIC18F67J11
- Teclado matricial 6 filas x5 columnas
- GLCD de 128 x64 pixeles
- Módulo RFID-12LA

La PCB (PrintedCircuitBoard) que integra estos componentes se observa en la Fig. 2. Para su diseño se tomó en cuenta las siguientes consideraciones:

- Un diseño sencillo usando pistas cortas y una distribución organizada de los componentes.
- Las pistas se realizan con ángulos de 90 grados.
- Los componentes se ubican en una posición paralela a los bordes de la placa.
- El ancho de pista utilizado es de 5.44 mils para pistas externas y de 21.78 mils para pistas internas, valores determinados en base a la norma IPC-2221.

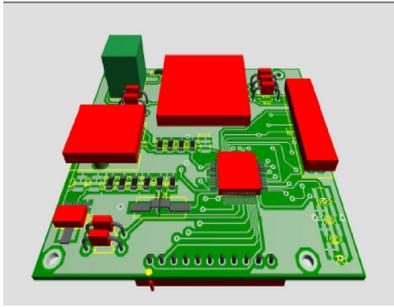


Fig. 2. Circuito Impreso en 3 dimensiones.

Por último, la carcasa del dispositivo facturador en pista que contiene los elementos de hardware antes mencionados, se diseña en base a delineamientos de controles móviles existentes en el mercado, pero se ajusta al tamaño de los elementos de hardware con los que se cuenta. Para su diseño se utilizó el software Solidworks y para su fabricación se utilizó un servicio de impresión 3D. El dispositivo una vez ensamblado se observa en la Fig. 3.



Fig. 3. Dispositivo facturador en pista

La programación del PIC18F67J11 se realiza en lenguaje C utilizando el compilador PCWHD. En la Fig. 4 se observa el diagrama de flujo principal del programa embebido en el PIC, el mismo que se encuentra formado por varios subprocesos secuenciales, en los que intervienen: el servidor, el cliente y el operador del sistema.

A continuación, se describen los subprocesos del diagrama de flujo de la Fig. 4.

- **En Configuración Inicial**, se especifica el orden en que se realizan las declaraciones de configuración del microcontrolador.
- **En Inicio de Sesión**, se realiza el proceso que identifica qué usuario inicia sesión, el cual puede ser mediante una clave de usuario ingresada por teclado o a través de tarjetas de proximidad.
- **En Menú**, se despliega una pantalla con 4 opciones que permite seleccionar: el lado del dispensador de combustible con el que se va a trabajar ('A' o 'B'), un reinicio y el cierre de sesión

- **En Selección del Tipo y Búsqueda de Cliente**, identifica que tipo de cliente realiza la compra y se comunica con el servidor para buscar sus datos.
- **En Proceso de Venta**, se registran los siguientes datos: el producto seleccionado, la cantidad de producto y la forma de pago.
- **En Forma de Pago**, se selecciona el tipo de transacción realizada, la cual puede ser: efectivo, cheque o tarjeta, y se imprime el comprobante.

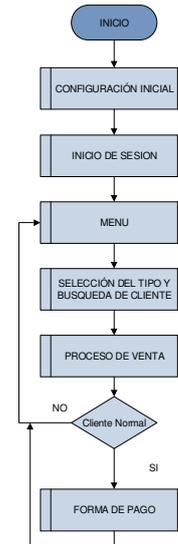


Fig. 4. Diagrama de Flujo del Programa del Microcontrolador

### B. Módulo de comunicación

La función de este módulo es comunicar al servidor con el dispositivo facturador, ya que el servidor posee una interfaz Ethernet mientras que el dispositivo una interfaz serial TTL. El módulo consta de una tarjeta Arduino Mega 2560 (Fig. 5) que por sus tres puertos seriales ofrece flexibilidad al sistema y además es compatible con la tarjeta Arduino Ethernet Shield.

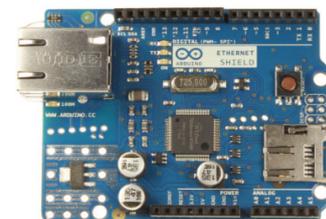


Fig. 5. Tarjeta Arduino Mega 2560

Además, el módulo de comunicación posee una tarjeta Ethernet Shield que proporciona una interfaz Ethernet a la tarjeta Arduino Mega 2560; lo que permite al dispositivo facturador conectarse a la red del sistema de facturación en pista.

### III. DESARROLLO DE SOFTWARE

El software es la parte central en el sistema de facturación en pista, porque controla todo el flujo de comunicación del sistema. El software se implementa en lenguaje JAVA a través

del IDE Netbeans versión 7.1. Se elige JAVA por ser un lenguaje de programación open-source y este IDE porque facilita el desarrollo de interfaces gráficas.

Al software se lo divide en dos aplicaciones a fin de evitar que los usuarios cierren el software, y por ende el sistema de facturación deje de funcionar, dichas aplicaciones se describen a continuación:

#### A. Aplicación Administrador de Datos

Esta aplicación no posee una interfaz gráfica, se ejecuta en segundo plano para garantizar que siempre se encuentre activa, y gestiona la comunicación entre dispositivos facturadores, base de datos e impresoras. El Web Service Merlyna es utilizado en esta aplicación y se muestra la utilización del mismo en la Fig. 6.

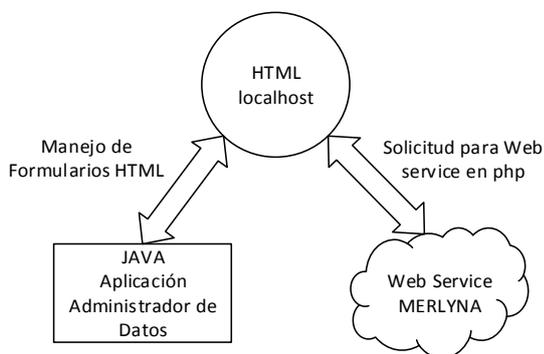


Fig. 5. Utilización de Web Service

Para la implementación del Web Service se crea una aplicación en lenguaje PHP en el IDE Netbeans. La aplicación PHP se ejecuta en el localhost, por tal motivo se configura un servidor HTTP, en este caso Apache Tomcat. Esta aplicación permite usar el web service cada vez que se pulsa el botón “Enviar” y en la aplicación Administrador de Datos se crea un método llamado “consumir\_webservice” que permite el manejo de páginas HTML. Finalmente, para realizar una consulta la aplicación Administrador de Datos llena automáticamente los campos en la aplicación PHP y obtiene la información que retorna la página.

#### B. Aplicación Cliente-Servidor Units

La aplicación Cliente-Servidor posee una interfaz gráfica y su acceso está restringido por clave. Esta aplicación se encarga de realizar las siguientes tareas:

- Hacer Reportes
- Ingreso y Edición de Clientes
- Anulación de Facturas
- Cierre de Caja
- Configuración

El diseño de la interfaz gráfica se rige por el esquema que se muestra en la Fig. 7.

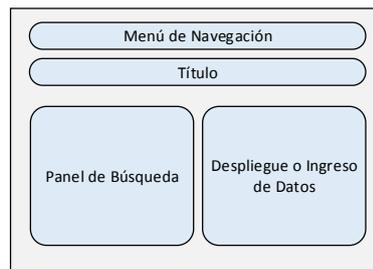


Fig. 7. Esquema General

Este esquema es usado para diseñar la mayoría de las pantallas de la interfaz y cuenta con la siguiente distribución:

- **Menú de Navegación.** Permite al usuario del sistema elegir que pantalla desea desplegar.
- **Título.** Indica que pantalla es.
- **Panel de Búsqueda.** Se colocan los campos necesarios para realizar una búsqueda.
- **Despliegue o Ingreso de Datos.** Despliega los datos a consultar y en algunos casos permite la modificación de los datos a través de botones de control.

Para las tareas de “Configuración” y “Cierre de Caja” se mantiene igual los campos de “Menú de Navegación” y “Título”, pero se generan campos adicionales específicos para cada una de las tareas. Por otro lado, para el resto de reportes se sigue al esquema mostrado en la Fig. 8, que cuenta con el Menú de Navegación, en el cual se selecciona que reporte se ha de desplegar y en el cuadro inferior “Reporte en objeto PDF”, se da una vista previa del reporte solicitado y permite guardar o imprimir.

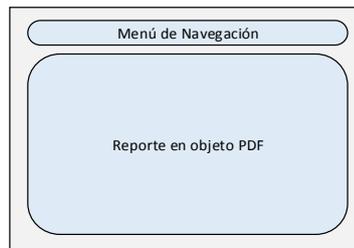


Fig. 8. Esquema para el Resto de Reportes

## IV. BASE DE DATOS

El diseño de la base de datos del Sistema de Facturación en Pista se basa en el modelo entidad-relación. Como gestor de la base de datos se elige PostgreSQL en su versión 9.2 por ser open-source, y para el diseño de la misma se elige el software Power Designer en su versión 16; porque facilita el diseño de la base de datos y permite pasarlo a PostgreSQL.

Además, para el diseño de la base de datos se utiliza la metodología orientada a objetos, por lo tanto se muestra a continuación: el diagrama de casos de uso, se explica el diagrama de clases y finalmente se pasa al modelo lógico de la base de datos.

### A. Diagrama de Casos de Uso

El diagrama de casos de uso del Sistema de Facturación en Pista, Fig. 9, permite visualizar el comportamiento del sistema, así como identificar que actores se ven involucrados en el mismo; se observan dos actores: un operador y un administrador. Todas las acciones del diagrama se plasman en la base de datos del sistema.

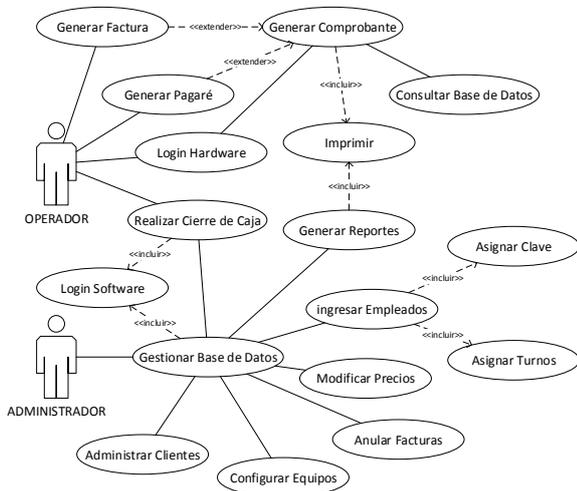


Fig. 9. Diagrama de Casos de Uso

### B. Diagrama de Clases

El diagrama de clases describe la estructura del sistema a través de una metodología orientada a objetos. Todas las actividades y relaciones mostradas en el diagrama de casos de

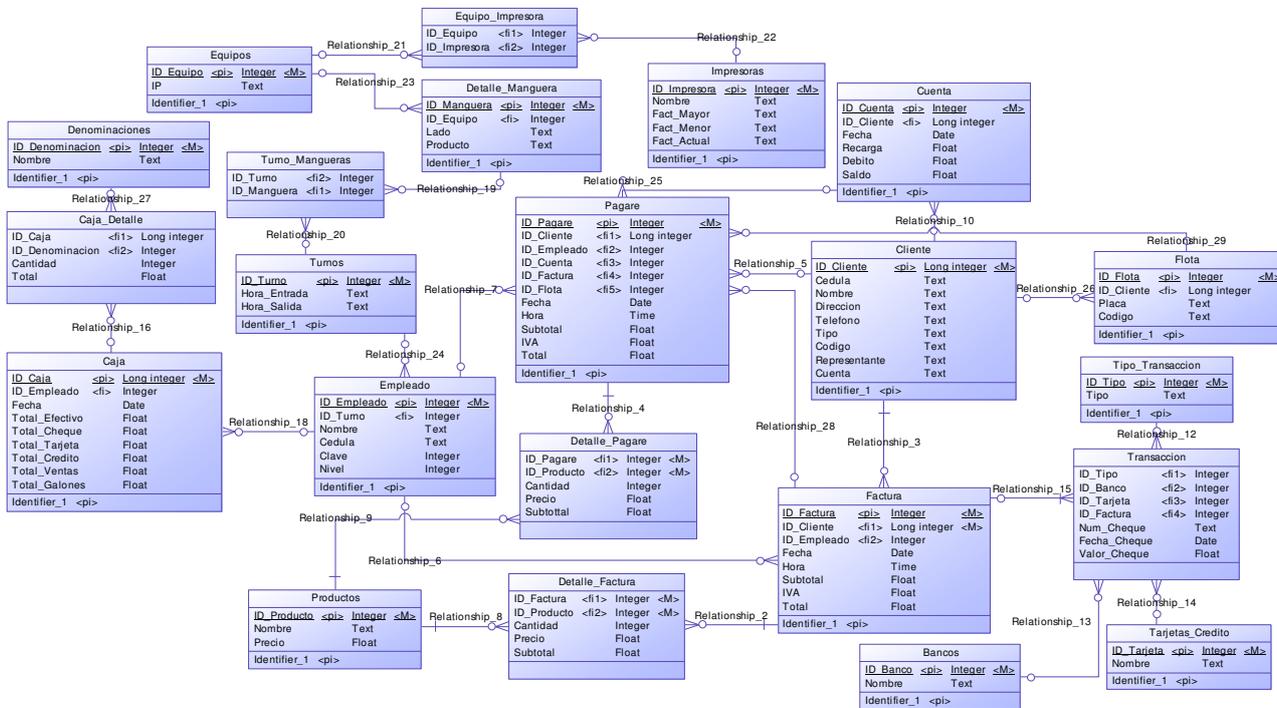


Fig.9. Modelo Lógico de la Base de Datos

uso sirven de guía para estructurar el diagrama de clases de la base de datos, el cual ya muestra un modelo entidad-relación porque al ser un sistema pequeño cada clase se convierte en una tabla para la base de datos.

### C. Modelo Lógico

El modelo lógico de la base de datos (Fig. 9) parte esencialmente del diagrama de clases. En él se definen las entidades que corresponden a las clases del diagrama anterior, y se definen los atributos de cada entidad, su clave primaria y la relación entre cada entidad.

El software Power Designer, es donde se diseña el modelo lógico de la base de datos, posee una herramienta que permite pasar automáticamente del modelo lógico al modelo físico, y una vez en el modelo físico se puede generar el script para la creación de la base de datos en PostgreSQL.

## V.RESULTADOS

Las pruebas se realizaron en la estación de servicio ESNAL 2 ubicada en Tandapi, Km 42 Santo Domingo de los Tsáchilas, que cuenta con las siguientes partes: 3 dispensadores de gasolina, un servidor, instalación de red y demás trabajos de obra civil necesarios para la instalación del sistema. Las pruebas realizadas son las siguientes:

### A. Conectividad de los módulos de comunicación a la red

La prueba se realiza con un PING desde el servidor a la dirección IP de cada módulo, enviando 4 paquetes de tamaño de 32 Bytes y esperando a que estos sean devueltos.

### B. Impresión de facturas válidas

Para esta prueba se considera una muestra de 3900 facturas, equivalente a un mes de trabajo, y se analiza que cada factura cumpla con las siguientes características: sea emitida, que no tenga errores, y que los datos mostrados correspondan a la información del cliente.

Los resultados, mostrados en la Fig. 10, advierten un total de 3814 (97.8%) facturas que fueron emitidas correctamente, la muestra incluye clientes que piden más de un producto y su forma de pago varía entre las tres opciones: efectivo, tarjeta, cheque.

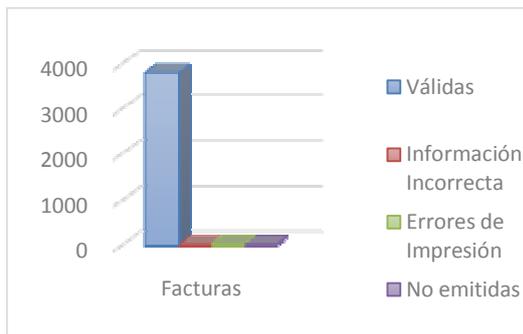


Fig. 10. Facturación de un día promedio de trabajo

### C. Módulo RFID

Para evaluar la funcionalidad del módulo RFID integrado en los dispositivos facturadores, se realizan dos pruebas diferentes que se listan a continuación:

- *La primera.* Es el tiempo de respuesta para usuarios del sistema y clientes, comparado con el tiempo de ingreso manual. En la Tabla I se observa los resultados y se demuestra que el uso de tarjetas RFID es 3 veces más eficiente que el método de ingreso manual.

Tabla I. Resultados prueba de funcionamiento módulo RFID

	Tiempo de respuesta (segundos)	Cantidad de ingresos al mes	Tiempo total mensual (segundos)
<b>Tarjetas RFID</b>			
Usuarios del sistema	4	60	240
Clientes	6	30	180
<b>Ingreso Manual</b>			
Usuarios del sistema	13	60	780
Clientes	16	30	480

- *La segunda.* Evalúa la cantidad de lecturas válidas versus no válidas al primer intento y la cantidad de tarjetas no registradas validadas por el sistema, para lo cual se realiza: registrar 10 tarjetas de usuario del sistema, registrar 10 tarjetas de clientes y se mantienen 10 tarjetas sin registrar. La Fig. 11 muestra los resultados de la prueba y se valida que el sistema no da paso a tarjetas que no hayan sido registradas previamente.

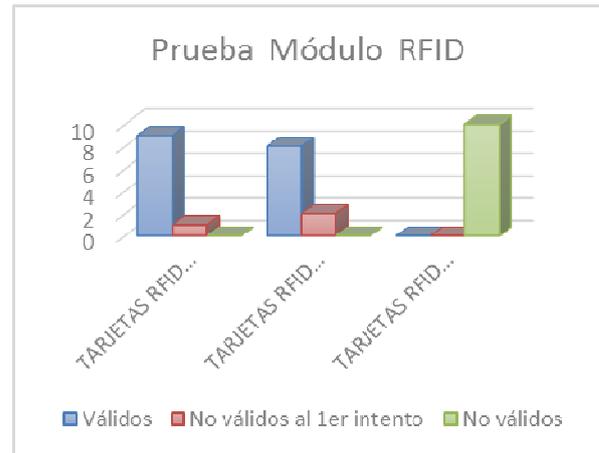


Fig. 11. Resultado prueba de reconocimiento de los dispositivos

### D. Módulo RFID

Esta prueba compara el tiempo que lleva realizar una factura entre el sistema de facturación desarrollado y el ingreso manual. La prueba se la realiza en dos días para una muestra de 10 clientes, en el primer día se toma el tiempo que tarda realizar cada factura de la forma manual y en el segundo día se lo realiza con el sistema desarrollado.

Al concluir la prueba se obtiene los resultados mostrados en la Fig. 12, de donde se concluye que el tiempo promedio que lleva hacer una factura de la forma manual tradicional, es superior al tiempo promedio que lleva hacerlo con el sistema de facturación desarrollado; lo que implica un mayor tiempo en el proceso de despacho de combustible y al final del día un menor número de clientes atendidos.

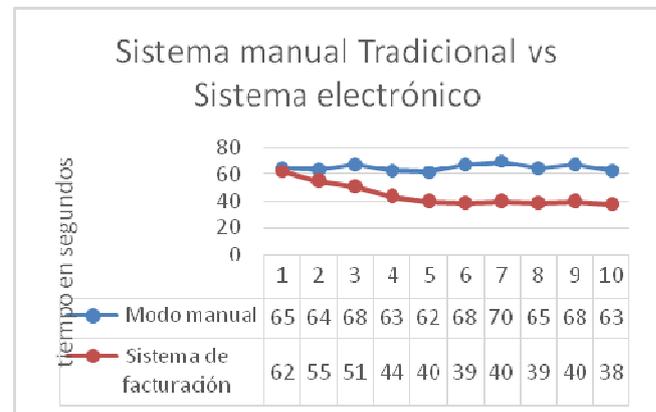


Fig. 12. Comparación de tiempo de facturación Ingreso manual vs. Sistema de facturación instalado.

## VI. CONCLUSIONES

La utilización de la norma IPC-2221 para el diseño de PCBs, permite determinar el ancho de las pistas, en base a la corriente máxima que las atraviesa, y las condiciones ambientales existentes en el lugar donde la tarjeta vaya a ser instalada.

El diseño del módulo de comunicación se facilita gracias al uso de la plataforma Arduino, útil para la implementación de prototipos debido a su flexibilidad en software y hardware, porque cuenta con placas que pueden ser montadas para extender su funcionalidad; como es el caso de la Ethernet Shield utilizada para la comunicación Ethernet; además, posee librerías dedicadas para comunicación Ethernet y serial RS232 indispensables para el desarrollo del sistema.

El uso de Java como lenguaje de programación para el desarrollo de la aplicación del servidor, permite crear una aplicación multitareas que cumpla todas las funciones necesarias para un proceso de facturación, gracias a la cantidad de librerías disponibles, nativas e independientes que permiten la implementación del protocolo TCP/IP, conexión con bases de datos, manejo de archivos pdf y manejo de páginas web.

La implementación de tecnología RFID, gracias a su velocidad de respuesta y facilidad de manipulación de los Tags o tarjetas, reduce el tiempo dedicado por el usuario del sistema para el ingreso de información del cliente, siendo tres veces más rápido que el método de ingreso manual de acuerdo a las pruebas y el análisis realizado.

La implementación de la base de datos con el modelo de entidad relación, permite obtener un mejor registro de las ventas realizadas en la estación de servicio, y facilita la generación de reportes, al asegurar las relaciones con entidades válidas y al evitar la duplicidad en los registros.

#### REFERENCIAS

- [1] Arduino. (2 de mayo de 2014). *Arduino*. Obtenido de <http://www.arduino.cc/es/>
- [2] Ibrahim, D. (2008). *PIC Microcontroller Projects in C: Basic to Advanced*. UK: Elsevier.
- [3] IPC. (Febrero de 1998). IPC-2221. *Generic Standard on Printed Board Design*.
- [4] Maldonado, P. (16 de Septiembre de 2013). El SRI ahora exige número de cédula al comprar combustible. *EL COMERCIO*.
- [5] Microchip. (2009). *PIC18F87J11 Family Data sheet*.
- [6] Saffirio, M. (4 de mayo de 2014). *¿Qué son los Web Services?* Obtenido de <http://msaffirio.wordpress.com/2006/02/05/%C2%BFque-son-los-web-services/>
- [7] Valdés Pérez, F., & Pallàs Areny, R. (2007). *Microcontroladores: Fundamentos y Aplicaciones con PIC*. España: MARCOMBO, S.A.
- [8] Guarnizo Poma, J. J., & Vallejo Mancero, B. J. (2014). *Diseño e Implementación de un Sistema de Facturación para Estaciones de Servicio*. Sangolquí, Pichincha, Ecuador: Universidad de las Fuerzas Armadas - ESPE.