

## Solución basada en TICs para el proceso de pedidos a través de preventistas

MORALES-HERNÁNDEZ, Maricela\*†, ALTAMIRANO-CABRERA, Marisol, DOROTEO-CASTILLEJOS, Rubén y JIMÉNEZ-ALVAREZ, Jasiel.

Recibido Julio 7, 2016; Aceptado Septiembre 14, 2016

### Resumen

La "Solución basada en TICs para el proceso de pedidos a través de preventistas" se presenta como una solución para una empresa de bienes que requiera agilizar el proceso de pedidos a través de preventistas. Esta consiste de un sistema de levantamiento de pedidos con un servidor central, el cual incluye una aplicación móvil que permite la captura de pedidos y una aplicación web que es el medio para agregar los pedidos de la aplicación móvil al sistema central. La aplicación móvil ofrece portabilidad, rapidez y certeza en los productos sin depender de una conexión a internet, ya que la aplicación cuenta con una base de datos local en el dispositivo móvil que se sincroniza con la base de datos centralizada cuando el preventista lo decida o cuente con una conexión estable, de este modo tendrá acceso a información como lo son: los artículos, precios y sus existencias; artículos disponibles, etc, evitando con ello sobreventa de artículos. La metodología de desarrollo de software utilizada es XP (Extreme Programming), la aplicación web está programada en PHP del lado del servidor y con Javascript del lado del cliente, para la aplicación móvil se utiliza el lenguaje nativo de Android, que es SDK; y para caracterizar a los clientes se utiliza código QR.

**Sistema Web, Aplicación Móvil, Soluciones con TICs, Código QR**

### Abstract

The "Solution based on TIC's to orders process through sellers" is presented as a solution for a company that requires expedite the ordering process through sellers. This consists of a system of soliciting orders with a central server, which includes a mobile application that allows order entry and a web system which add orders from the mobile to the central system. The mobile application offers portability, speed and accuracy in products without relying on an internet connection because the application has a local database on the mobile device wich is synchronized with the central database when the seller decides or he/she has a stable connection, then seller can access information such as: items, prices and item's stock; available items, etc, thereby avoiding oversold articles. The software development methodology used is XP (Extreme Programming), the server side web application is programmed in PHP, while client side in JavaScript, mobile application is coded using SDK, the native language of Android and, QR code is used in order to characterize customers.

**Web system, mobile application, TICs solutions, QR code**

**Citación:** MORALES-HERNÁNDEZ, Maricela, ALTAMIRANO-CABRERA, Marisol, DOROTEO-CASTILLEJOS, Rubén y JIMÉNEZ-ALVAREZ, Jasiel. Solución basada en TICs para el proceso de pedidos a través de preventistas. Revista de Tecnologías de la Información 2016. 3-8: 49-59

\*Correspondencia al Autor (Correo Electrónico: moralesh.maricela@gmail.com)

† Investigador contribuyendo como primer autor.

## Introducción

El presente trabajo muestra una solución basada en TICs (Tecnologías de la Información y la Comunicación), para un proceso cotidiano en las empresas, que aún en algunas de ellas, se realiza de forma manual, tal como se explica en el planteamiento del problema en el presente documento, el cual se ha organizado para su mejor comprensión en nueve apartados, en el primero se introduce al lector al documento. En el segundo apartado, se describen los problemas encontrados en la empresa “Tecnología en Negocio” en el proceso de levantamiento de pedidos. En el apartado tres se aborda la metodología aplicada, en este caso se centra en la Metodología de Desarrollo de Software denominada XP (eXtreme Programming, por sus siglas en inglés), se describen los requisitos encontrados a partir de la técnica de historias de usuario, y la planeación hecha se basa en el número de historias encontradas. El desarrollo de la base de datos y de la programación se explica brevemente en los apartados cuatro y cinco.

En el apartado seis se muestran los resultados obtenidos para esta solución que consiste de dos aplicaciones: una web y otra móvil. En el apartado siete se presenta un glosario de términos técnicos, sobre algunos conceptos que al lector le pueden resultar desconocidos. En el apartado ocho se presentan las conclusiones a las que se llegan una vez finalizada la implementación del software, finalmente se presenta un glosario de términos que ayudará al lector del documento a la mejor comprensión del mismo. Las referencias consultadas para el desarrollo de la solución propuesta se presentan en el apartado nueve.

## Planteamiento del problema

La solución expuesta en este artículo responde a la problemática de “pérdida monetaria” en la empresa denominada “Tecnología en Negocio”, ésta presenta pérdidas monetarias y de clientes debido a la lentitud del proceso de levantamiento de pedidos, el cual se realiza de forma manual, es decir el preventista va a la ubicación del cliente y realiza la visita, anotando los pedidos en una libreta y posteriormente se la entrega al capturista, quien procesa los pedidos. En este proceso se identifica pérdida de tiempo al momento de registrar todos los productos de cada pedido levantado, causando en consecuencia el retraso en el abastecimiento y entrega de productos a sus clientes.

La empresa no tiene un control sobre el comportamiento de sus preventistas en la toma de sus pedidos, generando pérdidas ya que se desconoce si los preventistas cubrieron realmente su ruta completa visitando a todos los clientes que le son asignados, lo que genera la pérdida de clientes al no ser visitados estos clientes y tomar sus pedidos en los días planeados.

A pesar de que la empresa cuenta con un sistema para administrar sus pedidos este proceso es muy lento además que no cuenta con una herramienta para la administración del comportamiento de sus preventistas, se pretende agilizar el proceso de captura de pedidos y administrar el comportamiento de los vendedores y de esta forma evitando la pérdida monetaria y de clientes.

## Metodología

Para el desarrollo de esta solución basada en Tecnologías de la Información (TICs) se ha utilizado la Metodología de Desarrollo de Software Extreme Programming (XP), formulada por Kent Beck, esta se considera el proceso más destacado de los procesos ágiles de desarrollo de software. XP se diferencia de las metodologías tradicionales principalmente porque pone más énfasis en la adaptabilidad que en la previsibilidad. Los defensores de XP creen que adaptarse a los cambios de requisitos en cualquier punto de la vida de un proyecto es una aproximación más realista que intentar definir todos los requisitos al inicio del proyecto e invertir esfuerzos después en controlar los cambios en los requisitos (Beck & Andres, 2012).

## Historias de Usuario.

Para abordar el desarrollo del software aplicando XP se inicia con la definición de los requerimientos, en este caso todos los requerimientos se expresan como escenarios (llamados historias de usuario), los cuales se implementan directamente como una serie de tareas (Sommerville, 2005), para tal fin, primero se identificaron todos los roles o tipos de usuarios que interactúan en el proceso del levantamiento de pedidos, y se encontraron cinco tipos de usuarios, éstos se describen a continuación:

- a) **Preventista:** este usuario es el que realiza la cotización y el levantamiento del pedido a los clientes asignados. Este proceso lo realiza dirigiéndose a la ubicación física del cliente y mediante los catálogos de artículos realiza las notas del pedido para posteriormente entregarlas al capturista.
- b) **Cliente:** Es el que requiere de un producto y se lo solicita al preventista o a la empresa. Este proceso se realiza cuando el cliente requiere de artículos, por lo que cuando es visitado por el preventista el cliente solicita los artículos requeridos para que el preventista los capture en un pedido.
- c) **Administrador:** Es el que se encarga de asignarle los clientes a un preventista. Este proceso consiste en agregar o modificar los datos de los clientes que tendrá a cargo cada preventista.
- d) **Coordinador:** Es el que supervisa a los preventistas, y también puede realizar el levantamiento de pedidos de cualquier cliente. Este proceso consiste en supervisar la actividad de los preventistas.
- e) **Capturista:** Se encargan de ingresar los pedidos de los preventistas al sistema central. Este proceso consiste en ingresar todas las notas de pedidos de los preventistas en el sistema central.

Teniendo ya identificados a los usuarios del proceso, se obtienen los requisitos, y se identifican ocho historias de usuario, las cuales se dividen en tres iteraciones y generan la Lista Maestra de Historias de Usuario tal como se muestra en la Tabla 1.

Iteraciones	Historias de usuario a resolver
Iteración 1	<ol style="list-style-type: none"> <li>1. Levantamiento de pedido</li> <li>2. Captura de pedidos</li> <li>3. Visita inefectiva</li> <li>6. Administración de preventistas</li> </ol>
Iteración 2	<ol style="list-style-type: none"> <li>5. Búsqueda de clientes</li> <li>4. Captura de pedidos en el sistema</li> <li>8. Registro de nuevo cliente</li> </ol>
Iteración 3	<ol style="list-style-type: none"> <li>7. Reportes de visitas</li> </ol>

**Tabla 1** Lista maestra de historias de usuario

### Plan de Pruebas de Aceptación.

Una vez que se planifican las historias de usuario y el número de iteraciones en las que se resuelven, se planean las pruebas de aceptación para cada historia de usuario, éstas se muestran en la Tabla 2, para cada caso se detalla cuáles son las pruebas a las que se somete la funcionalidad para que sea aceptada por el cliente.

Historia de Usuario	Pruebas de aceptación
1. Levantamiento del pedido	<ol style="list-style-type: none"> <li>1. Sincronización con la información de la base de datos.</li> <li>2. Consulta de la lista de clientes</li> <li>3. Obtención de la geolocalización al iniciar la visita</li> <li>4. Impresión del pedido en un ticket con la impresora móvil.</li> <li>5. Captura de la duración de la visita.</li> </ol>
2. Captura de pedidos	<ol style="list-style-type: none"> <li>1. Consulta de artículos por clave o nombre.</li> <li>2. Consulta de la lista de precios.</li> <li>3. Edición o eliminación de artículos agregados al pedido.</li> <li>4. Cancelación un pedido.</li> </ol>
3. Visita inefectiva	<ol style="list-style-type: none"> <li>1. Marcación de una visita inefectiva.</li> <li>2. Selección del motivo del porqué es inefectiva.</li> </ol>
6. Administración de preventistas	<ol style="list-style-type: none"> <li>1. Alta de un vendedor.</li> <li>2. Edición de un vendedor.</li> <li>3. Baja de un vendedor.</li> <li>4. Consulta de los vendedores registrados.</li> </ol>

**Tabla 2** Pruebas de aceptación Historias de Usuarios

Historia de Usuario	Pruebas de aceptación
5. Búsqueda de clientes	<ol style="list-style-type: none"> <li>1. Generación de código QR de clientes en la aplicación.</li> <li>2. Escaneo de código QR en la aplicación móvil.</li> </ol>
4. Captura de pedidos en el sistema	<ol style="list-style-type: none"> <li>1. Guardar los pedidos en el sistema central a través del web-service.</li> </ol>
8. Nuevo cliente	<ol style="list-style-type: none"> <li>1. Captura de la información de un nuevo cliente desde la aplicación móvil.</li> <li>2. Almacenamiento de la información del nuevo cliente en el sistema central a través del web-service.</li> </ol>
7. Reportes de visitas	<ol style="list-style-type: none"> <li>1. Listado de las visitas realizadas cada día.</li> <li>2. Visualizar la geolocalización de las visitas de cada día en un mapa.</li> <li>3. Obtención de reportes gráficos de las visitas efectivas e inefectivas.</li> </ol>

**Tabla 2** Continuación. Pruebas de aceptación Historias de Usuarios

Después de tener la lista maestra y el plan de trabajo, se inicia con el desarrollo de los entregables para cada una de las iteraciones. A continuación, se describen cada uno de los componentes que incluye esta solución.

### Desarrollo de las bases de datos

En el desarrollo de esta solución, primero se crean las nuevas tablas que se añaden al sistema central el cual se encuentra en un servidor IIS (Web-Internet Information Server) con sistema operativo Windows server 2012 y la base de datos se encuentra en el gestor de base de datos Firebird versión 2.5, y el juego de caracteres está configurado con el estándar "ISO8859\_1" y con la librería "fbclient.dll" la que permite la conexión entre cliente y servidor.

Antes de llevar a cabo la aplicación móvil se realiza el diseño de la base de datos. Para esto se hace uso de la documentación de SQLite, historias de usuario y el diseño de la base de datos que se encuentra en el servidor central. En la figura 1 se muestra el diagrama UML (Lenguaje Unificado de Modelado- Unified Modeling Language) de las tablas que se crearon para la aplicación móvil.

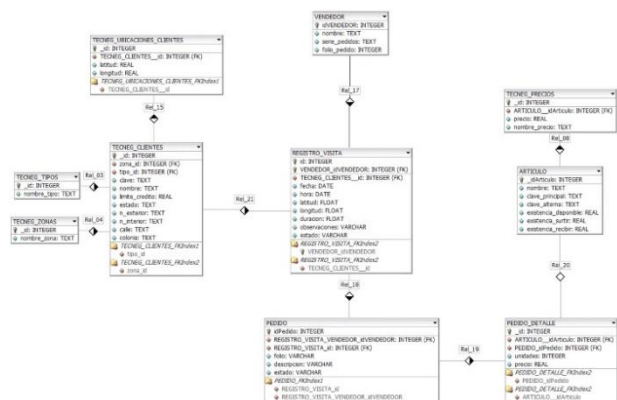


Figura 1 Base de datos para la aplicación móvil.

Debido a que ya se tiene una base de datos en el sistema central de la empresa donde se ha desarrollado el software, solamente se agregan algunos datos que se consideran importantes para lograr integrar las TIC's al proceso de pedidos, estos datos se representan en el diagrama UML de la Figura 2.

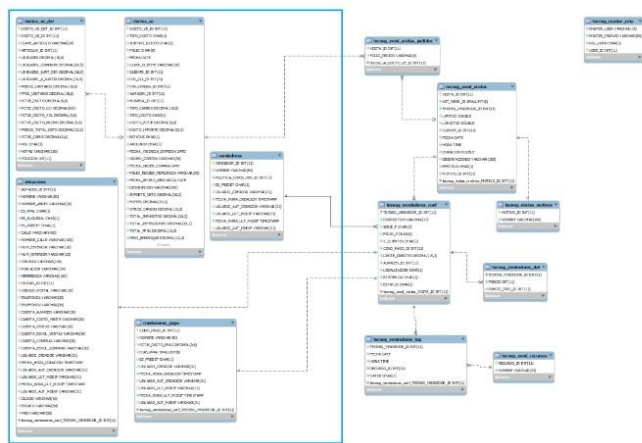


Figura 2 Base de datos en la aplicación web.

Desarrollo de la programación

Después de tener la base de datos tanto para la aplicación web, como para la aplicación móvil se desarrolla la programación del software. Para la programación se requiere:

- a) El lenguaje de programación PHP (Hypertext Preprocessor) versión 5.6 en el que está desarrollado el Web-Service y la aplicación Web. Para la configuración de PHP se necesita:
  - Añadir la extensión "pdo\_firebird", esta es una librería que permite la conexión entre php y el gestor firebird para poder realizar las consultas.
  - Agregar la librería fbclient.dll a la carpeta php.

- b) El framework Slimp versión 2.5 en la plataforma del servidor IIS para el desarrollo del web-services. Para que el framework funcione bajo la plataforma del servidor IIS se agrega el archivo de configuración "Web.config" el cual permite que el servidor IIS redirecciones todas las peticiones derivadas de la URL de la aplicación con su respectiva función

Para el inicio de sesión se crea una interfaz con un formulario donde se ingresa el nombre del usuario y su contraseña, para la seguridad y validación de estos datos se necesita:

- a) La librería y función mymd5 de jQuery, a la cual se le pasa la contraseña del usuario y esta se encripta antes de ser transmitida por la red, con el fin de establecer seguridad en las conexiones al sistema.
- b) Una función php específica en el servidor para la autenticación de estos datos, la cual valida si existe el usuario, y si la contraseña es correcta.
- c) Una cookie con php para mantener la sesión activa con los datos del usuario y eliminar la cookie al cerrar sesión.

### Resultados

Como resultado se tiene la Solución basada en TICs para el proceso de pedidos a través de preventistas, se presentan las pantallas más importantes de esta solución que tiene dos componentes:

- a. La aplicación Web: La cual está programada en PHP para el lado del servidor y el lado del cliente está en Javascript.
- b. La aplicación móvil: Se programa en lenguaje nativo de Android (SDK).

### Aplicación Web

En la figura 3 se presenta la pantalla principal a la que tiene acceso el administrador de la aplicación web.



Figura 3 Interfaz del Administrador de la Aplicación Web.

El administrador puede visualizar a los vendedores activos, así como dar de alta nuevos vendedores, como puede observarse en la figura 4.

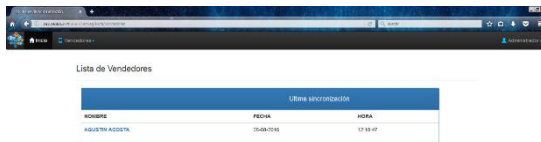


Figura 4 Lista de vendedores de la aplicación web.

El administrador configura algunos datos del vendedor como son: nombre del vendedor, dirección MAC de su equipo móvil, folio de los pedidos, etc., tal como se observa en la figura 5.

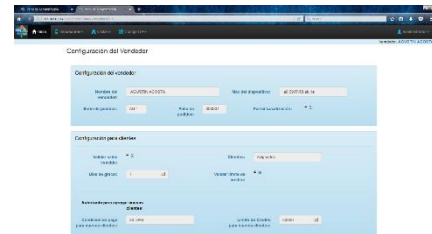


Figura 5 Cofiguración del vendedor.

Uno de los beneficios importantes que logra la empresa con esta solución basada en TICs es que se facilita el seguimiento de las visitas del vendedor a las ubicaciones físicas del cliente, con información sobre las mismas, si éstas fueron efectivas o no, y el monto global de los pedidos que logró con los clientes asignados, como puede observarse en la figura 6.

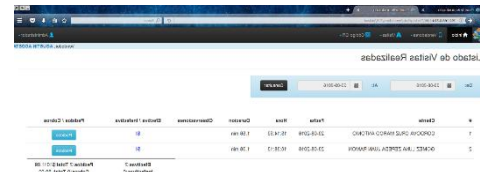


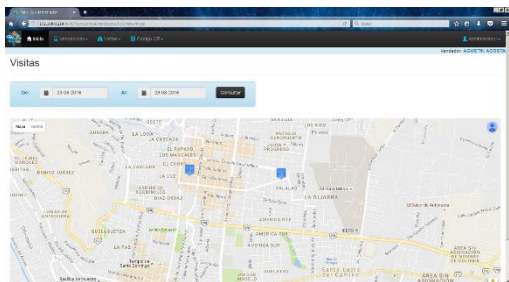
Figura 6 Listado de visitas realizadas.

Si se requiere ver con detalle los pedidos que cada vendedor levanta en su ruta de visitas a los clientes, en la aplicación se tiene un botón azul que dice “Pedidos”, con éste se puede visualizar el detalle de cada pedido, como se muestra en la figura 7.



Figura 7 Detalle de los pedidos que levanta el vendedor.

La aplicación web cuenta con un módulo para visualizar la ruta realizada por un preventista, en este módulo se marcan los puntos donde se realizan las visitas de cada ruta, estos puntos son marcados en un mapa de "Google maps". Se muestran detalles de cada visita; el tiempo que se lleva en esa visita, algunas observaciones y los pedidos que se realizan. Esto trae como beneficio a la empresa conocer el comportamiento de los preventistas en su ruta diaria y así analizar el desempeño de los mismos, esta funcionalidad se observa en la figura 8.



**Figura 8** Geolocalización en un mapa de las visitas realizadas por el preventista.

La aplicación web dando solución a los problemas de la empresa, implementa el uso de códigos QR, los cuales tienen las siguientes características:

- a. Versión 1, de 25 x 25 pixeles.
- b. El nivel de corrección: Level H (High).



**Figura 9** Código QR utilizado.

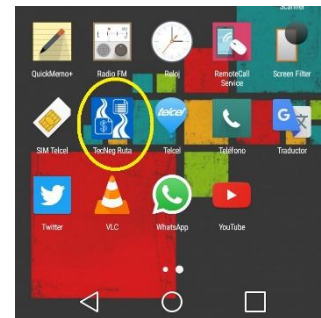
La aplicación web imprime los códigos QR, los cuales tienen la función de identificar todos los datos de un cliente (ver figura 10). Estos códigos son leídos por la aplicación móvil.



**Figura 10** Código QR de los Clientes.

### Aplicación móvil

En el dispositivo móvil del vendedor o preventista se muestra un ícono de la aplicación desarrollada para la parte móvil, tal como se muestra en la figura 11.



**Figura 11** Ícono de la aplicación móvil.

Al iniciar la aplicación móvil, la pantalla principal muestra un menú para el preventista QR el que puede tener acceso al catálogo de artículos, la lista de los clientes que le fueron asignados por el administrador de la empresa, las visitas que realiza, así como el menú de sincronización, como puede observarse en la figura 12.

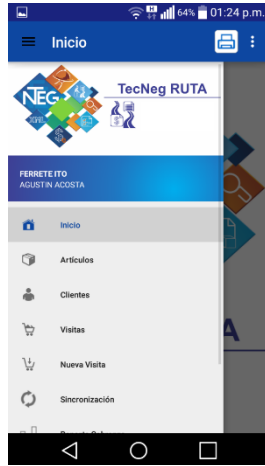


Figura 12 Pantalla de inicio de la aplicación móvil.

El catálogo de productos puede ser consultado por el preventista en su dispositivo móvil. Este catálogo tiene la apariencia que se muestra en la figura 13. Del mismo modo el preventista consulta los clientes que le fueron asignados, el resultado de esta consulta se ilustra en la figura 14.



Figura 14 Lista de clientes en la aplicación móvil.

En la aplicación móvil se captura el código del artículo para desplegar todos los detalles de ese artículo en la pantalla, se agrega la cantidad y automáticamente se va calculando el monto total del pedido, de acuerdo con los artículos que el cliente solicita, esta pantalla se puede ver en la figura 15.

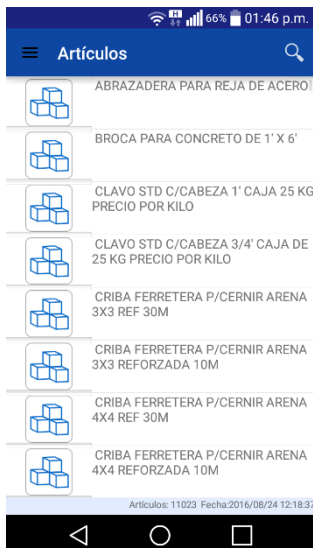


Figura 13 Catálogo de artículos.

Para tener un comprobante del pedido que se levanta, el vendedor tiene la opción de imprimir un ticket, utilizado una impresora móvil, por lo que la aplicación le pide confirmar si desea imprimir el pedido, como puede observarse en la figura 16.



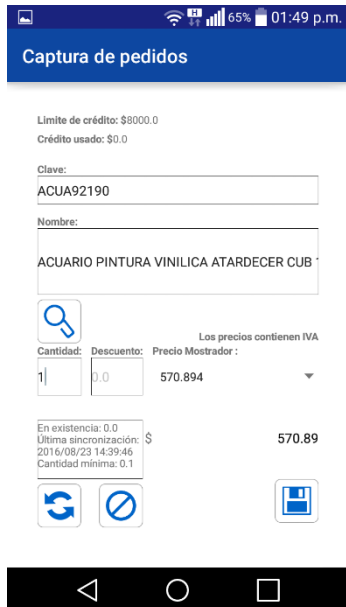


Figura 15 Captura de pedidos..

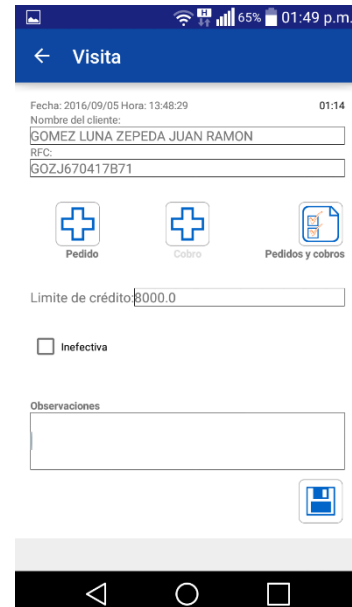


Figura 17 Seguimiento de visitas.

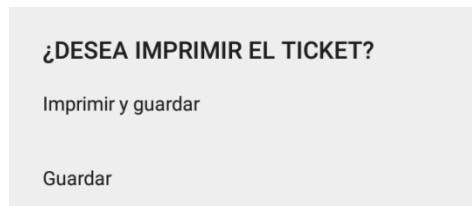


Figura 16 Ventana de impresión del pedido

También en la aplicación móvil, el preventista visualiza las visitas hechas a sus clientes, teniendo en la pantalla visibles los pedidos hechos por ese cliente, así como la cobranza de los pedidos entregados. Un ejemplo de esta funcionalidad se presenta en la figura 17.

Finalmente, la aplicación móvil cuenta con el menú de sincronización, el cual se requiere cuando sea necesario vaciar la información que el preventista ha recabado durante las visitas a los clientes. Esta funcionalidad en el dispositivo móvil se muestra en la figura 18.

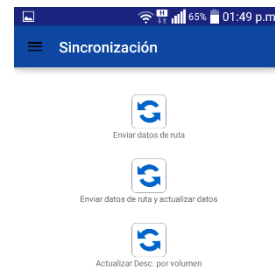


Figura 18 Menú de sincronización de datos.

## Anexos

### Glosario de términos

**Android:** Sistema operativo basado en el núcleo Linux. Diseñado principalmente para su uso en dispositivos móviles con pantalla táctil.

**API:** Interfaz de programación de aplicaciones, abreviada como API (del inglés: Application Programming Interface), son un conjunto de subrutinas, funciones y procedimientos, que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

**Código QR:** Quick Response Code o código de respuesta rápida, es un módulo para almacenar información en una matriz de puntos o en un código de barras bidimensional.

**Cookie:** Es una pequeña información enviada por un sitio web y almacenada en el navegador del usuario, de manera que el sitio web puede consultar la actividad previa del usuario.

**Framework:** (marco de trabajo) es una estructura conceptual y tecnológica de soporte definido, normalmente con artefactos o módulos de software concretos, que puede servir de base para la organización y desarrollo de software.

**Geolocalización:** Conocimiento de la propia ubicación geográfica de modo automático en un sistema de coordenadas determinado.

**Google maps:** Es un servidor de aplicaciones de mapas en la web que pertenece a Google. Ofrece imágenes de mapas desplazables, así como fotografías por satélite del mundo e incluso la ruta entre diferentes ubicaciones o imágenes a pie de calle (Google Street View).

**ISO 8859-1:** es una norma de la ISO que define la codificación del alfabeto latino, incluyendo los diacríticos (como letras acentuadas, ñ, ç), y letras especiales (como ß, Ø).

**Java:** Lenguaje de programación de propósito general, concurrente y orientado a objetos.

**Json:** Es un formato ligero para el intercambio de datos. JSON es un subconjunto de la notación literal de objetos de JavaScript que no requiere el uso de XML.

**Librería:** Conjuntos de implementaciones funcionales, codificadas en un lenguaje de programación, que ofrece una interfaz bien definida para la funcionalidad que se invoca.

**MAC:** La dirección MAC (siglas en inglés de “Media Access Control”; en español “Control de Acceso al Medio”) es un identificador de 48 bits (6 bloques hexadecimales) que corresponde de forma única a una tarjeta o dispositivo de red.

**MD5:** Abreviatura de Message Digest Algorithm 5, Algoritmo de Resumen del Mensaje 5, el cual es un algoritmo de reducción criptográfico de 128 bits.

**PHP:** Lenguaje de programación de uso general para programar aplicaciones del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico.

**Preventista:** Es la persona encargada de visitar a comercios minoristas, levanta pedidos de dichos comercios, para luego realizar la venta de los artículos solicitados en los pedidos.

**SDK:** Son las siglas en inglés de Software Development Kit. Es un kit de desarrollo, se compone del conjunto de herramientas de desarrollo de software que permiten a los programadores crear aplicaciones para un sistema concreto.

**XML:** Por sus siglas en inglés significa eXtensible Markup Language ('lenguaje de marcas extensible'), es un lenguaje de marcas desarrollado por el World Wide Web Consortium (W3C) utilizado para desalmacenar datos en forma legible.

### Conclusiones

En esta solución basada en TICs para el proceso de pedidos a través de preventistas o vendedores se obtuvieron dos aplicaciones para el levantamiento de pedidos, una aplicación móvil y una aplicación web que permite a la empresa Tecnología en Negocios S.A de C.V agilizar el proceso del levantamiento de sus pedidos.

El proceso de desarrollo del proyecto se fundamenta en el uso de la metodología XP desarrollando las tres iteraciones planeadas, se logran cumplir los requisitos iniciales de los diferentes usuarios de la herramienta.

El desarrollo en sentido técnico de la aplicación web se realiza mediante el lenguaje PHP para el lado del servidor y para el lado del cliente con el lenguaje Javascript, y para la aplicación móvil se utiliza el lenguaje nativo de Android SDK.

Como resultado de la creación de estas aplicaciones se espera darle solución a la problemática de la empresa y que ésta reduzca las pérdidas monetarias, así como expandir su cartera de clientes mediante la implantación de las aplicaciones y observar si el proceso de pedidos utilizando preventistas mejora sustancialmente.

Las aplicaciones inician una etapa de mantenimiento por parte de los usuarios, para mejorar sus funcionalidades.

El siguiente paso es realizar una evaluación de campo de las aplicaciones web y móvil, y a través de herramientas estadísticas, poder determinar el grado de impacto en los procesos globales de la empresa.

### Referencias

- Asuni, Nicola (2015). TCPDF - PHP class for PDF. Recuperado de <https://sourceforge.net/projects/tcpdf/files/>
- Beck, Kent & Andres, Cynthia. (2012). Extreme Programming Explained: Embrace Change. Stoughton, Massachusetts: Addison-Wesley Professional.
- Google Maps/Google Earth APIs. (2015). Terms of Service. Recuperado de <https://developers.google.com/maps/terms?hl=es>
- Joskowicz, José. (2008). Reglas y Prácticas en eXtreme Programming. Recuperado de <http://iie.fing.edu.uy/~josej/docs/XP%20-%20Jose%20Joskowicz.pdf>
- Javascript (2015). jQuery API. Recuperado de <http://api.jquery.com/>
- Manual de firebird (2009). Recuperado de <http://firebirdmanual.com/firebird/es/firebird-manual/2>
- Sánchez, Miguel Angel. (2012). Javascript. Recuperado de [https://books.google.com.mx/books?id=3x09sewjaHIC&dq=maza+javascript&source=gbs\\_navlinks\\_s](https://books.google.com.mx/books?id=3x09sewjaHIC&dq=maza+javascript&source=gbs_navlinks_s)
- Sommerville, Ian. (2005). Ingeniería del software. Madrid, España: Pearson Educación.