

## Propuesta de automatización para el seguimiento de ventas en microempresas

*Fecha de recepción: 2021-07-31 • Fecha de aceptación: 2021-09-10 • Fecha de publicación: 2021-10-10*

**Juan Sebastián Reza Morales**

Tata Consultancy Services, Ecuador

[jsrezam@gmail.com](mailto:jsrezam@gmail.com)

<https://orcid.org/0000-0002-8318-3695>

### RESUMEN

Varias microempresas realizan gran parte de la comercialización de sus productos de forma manual, sin una gestión de información óptima. En muchas ocasiones, el almacenamiento de estos datos es nulo. Registrar el trabajo realizado a lo largo de los principales flujos y acciones que se dan al momento que el agente de ventas contacta al cliente, oferta el producto y registra una orden de compra, es clave a razón de generar una visión concreta e integra acerca del desempeño real de los colaboradores, sin dejar de lado que pone a disposición una estadística global del negocio. La automatización de los procesos que realizan los agentes para llevar a cabo el registro de órdenes incrementará la eficiencia y la productividad de este flujo. Toda esta información automatizada, gestionada y almacenada a través de un sistema, brindará a la gerencia facilidad en la toma de decisiones que potenciarán el auge del negocio.

**PALABRAS CLAVE:** automatización, sistema web, productividad.

### ABSTRACT

Several micro-companies carry out much of the marketing of their products manually without optimal information management. In many cases, the storage of this data is null. Registering the work carried

out along the main flows and actions that occur when the sales agent contacts the customer, offers the product, and registering a purchase order is key to generate a specific and integrated vision about performance real of the collaborators, without neglecting that it makes available a global statistics of the business. Automating the processes that agents carry out to carry out an order registration will increase the efficiency and productivity of this flow. All this automated information, managed and stored through a system, will provide management power in making decisions that will boost the business.

**KEYWORDS:** automation, web system, productivity.

## Introducción

En un entorno cada vez más competitivo, es inaceptable realizar procesos manuales, el depender de un almacenamiento de información volátil, poco confiable y en muchos de los casos, hasta indisponible, impide a las gerencias o mandos superiores del negocio conocer el estatus real del mismo (Trasobares, 2003). Este problema puede llegar a limitar el crecimiento u optimización de las actividades primordiales del flujo de ventas.

Actualmente existen pequeños negocios dedicados a la venta y comercialización de sus respectivos productos, y para automatizar sus procesos utilizan sistemas informáticos, en su gran mayoría, sistemas web (Riera, 2020). Esta actividad regularmente es realizada por los agentes vendedores, los cuales se desenvuelven bajo la supervisión de un agente senior, cuya función consiste en recopilar toda la información relacionada con el desempeño de sus supervisados a lo largo de la interacción con el flujo de ventas. La constante implementación de soluciones tecnológicas tiende a dejar atrás la sensación de ineficiencia, que se podría dar al momento de ejecutar las actividades necesarias para ofertar y vender los productos relacionados al negocio.

La literatura demuestra que la automatización mediante sistemas informáticos se ha explorado notablemente, así lo demuestran varios trabajos publicados. Inicialmente se puede mencionar a Trasobares (2003), que desarrolla un sistema para gestión de ventas en unidades móviles, permitiendo a la agencia de venta ofrecer una fuente de información a través de la web, actualmente por la pandemia COVID-19, la mayoría de empresas optaron por utilizar sistemas móviles para optimizar sus procesos. Reyes Rivera et al. (2020) mencionan la problemática de las pequeñas empresas en tiempos de pandemia y proponen una solución informática para que la empresa pueda realizar sus ventas en línea, Artieda Guachamin et al. (2021) proponen una aplicación móvil personalizada para ofertar transporte en la ciudad de Quito con características similares a Uber e Indriver, en aplicaciones móviles se puede mencionar varias soluciones para gestionar adopciones de mascotas callejeras, que en los últimos años en la ciudad de Quito son un gran problema para la sociedad (Burbano Ulloa et al., 2020).

En este sentido, el presente trabajo ha sido construido en base a una sólida especificación de requerimientos indispensables para resolver un problema que se da cotidianamente en las empresas dedicadas a la oferta y venta de productos. El desarrollo realizado para este trabajo se enfocó directamente a pequeños negocios, los cuales buscan una automatización económicamente accesible y acorde a su volumen de productos, clientes y generación de órdenes para flujos como: control y seguimiento de campañas de venta, registro de órdenes y contactabilidad al cliente. El seguir una arquitectura enfocada en el ágil desacoplamiento del sistema implica una mayor inversión de tiempo al momento de generar nueva funcionalidad, debido al mandatorio construcción de interfaces necesarias para establecer comunicación con las diferentes capas del software.

En contraste al aplicar este tipo de arquitectura, se genera también una mejor mantenibilidad en el software, y lo más importante, la deuda técnica del sistema disminuye considerablemente, ya que al seguir un mismo estándar de desarrollo especializado en conservar la independencia

de componentes, resulta en impactos leves para mantenimientos futuros, como, por ejemplo, el cambio de la herramienta ORM por cambio de tecnología *front-end*.

## Metodología

El proyecto se ejecutó bajo la metodología ágil Scrum, a razón de aprovechar que la misma involucra al cliente y ayuda a generar entregas parciales del desarrollo en un periodo de tiempo predeterminado llamado *sprint*.

Para la ejecución de la metodología Scrum se utilizó la herramienta “Azure Devops”, con el objetivo de aprovechar los servicios que ofrece para los equipos de soporte y desarrollo. Con esta herramienta se puede planificar el trabajo, generar colaboración en el desarrollo de código y crear e implementar aplicaciones (Microsoft, 2021).

En específico de todos los servicios que presenta, está Suit, se empleó la funcionalidad del servicio “Azure Boards”, el cual resulta beneficioso para generar la administración del proyecto, ya que maneja un soporte nativo para la ejecución de la metodología Scrum, sin dejar de lado que se apoya en el tradicional tablero Kanban útil para generar el respectivo seguimiento a las tareas de cada iteración.

En la siguiente *Tabla 1* se visualizará la lista de requerimientos presentados para este proyecto.

**Tabla 1**

*Tabla de Requerimientos*

Tabla de Requerimientos	
Requerimientos	
1	El sistema debe contar un módulo que gestione la autenticación y autorización de los usuarios en el sistema.
2	El sistema debe contar con un módulo que gestione la carga, consulta y modificación de las categorías existentes en el sistema.
3	El sistema debe contar con un módulo que gestione la carga, consulta y modificación de los productos existentes en el sistema.
4	El sistema debe contar con un módulo que gestione la carga, consulta y modificación de los clientes existentes en el sistema.
5	El sistema debe contar con un módulo que gestione el estado y los privilegios de los usuarios existentes en el sistema.
6	El sistema debe contar con un módulo que gestione la creación y consulta de las órdenes a transaccionarse en el sistema.
7	El sistema debe contar con un módulo que gestione la creación, asignación, consulta y modificación de campaña de ventas.
8	El sistema debe contar con un <i>dashboard</i> que muestre la siguiente información sobre la actividad de ventas ejecutada: los productos más vendidos en el mes, los clientes más destacados en el mes, el agente más destacado, un <i>chart</i> con la información de ventas realizadas por cada agente y finalmente un <i>chart</i> que destaque el total de ventas por mes.

9	El sistema debe contar una aplicación web en la cual el agente pueda gestionar la contactabilidad al cliente y el progreso en sus campañas activas.
---	---

A su vez, en la *Tabla 2* se evidencian los roles e integrantes.

**Tabla 2**

*Tabla de Scrum Team*

Scrum Team		
1	Sebastián Reza	Scrum Master / Development Team Member

La construcción del presente proyecto tomó de tres iteraciones para abarcar todas las características y funcionalidad

### Primer *sprint*

El primer *sprint* fue ejecutado del 3 al 10 de mayo de 2021, cubriendo tareas iniciales indispensables a razón de establecer los fundamentos de la arquitectura limpia, la cual busca que el *core* del negocio que maneja la aplicación sea altamente desacoplable, tanto a los *frameworks* ORM, como a los *frameworks* que se especializan para construir *front-end*.

Otro hito importante para este *sprint* fue establecer las relaciones entre objetos de negocio para la gestión de usuarios, sin dejar de lado las validaciones de autenticación y autorización al sistema. A continuación, en la *Figura 1* se muestra parte del *backlog* ejecutado.

**Figura 1**

*Backlog para el primer sprint en Azure DevOps*

Order	ID	Title	Assigned To	State	Tags	
+	1	42	Crear módulo de cuentas y seguridad.	Sebastian Reza	Done	
		188	✓ Crear controlador para la gestión de cuentas y seguridad.	Sebastian Reza	Done	
		44	✓ Crear acción para registro de usuarios.	Sebastian Reza	Done	
		43	✓ Crear acción para inicio de sesión de usuarios.	Sebastian Reza	Done	
		189	✓ Implementar el inicio de sesión por medio JWT.	Sebastian Reza	Done	
		190	✓ Crear menú de navegabilidad.	Sebastian Reza	Done	
		78	✓ Crear página de registro.	Sebastian Reza	Done	
		77	✓ Crear página de inicio de sesión.	Sebastian Reza	Done	
		184	✓ Crear relación entre objetos de entidad en la base datos.	Sebastian Reza	Done	
		185	✓ Ejecutar primera migración para la creación de la BD.	Sebastian Reza	Done	
		183	✓ Configurar la unidad de trabajo para la transaccionalidad con...	Sebastian Reza	Done	
		186	✓ Añadir el repositorio para la entidad usuario a la unidad de tr...	Sebastian Reza	Done	

## Segundo *sprint*

El segundo *sprint* (Figura 2) fue ejecutado del 11 al 27 de mayo de 2021, con el objetivo de implementar la funcionalidad medular del sistema, la misma que se basa en habilitar al supervisor de ventas la posibilidad de generar campañas, las cuales cuenten con un número finito de clientes afines al negocio, para posteriormente ser contactados por un agente de ventas, a quien será asignada la nueva campaña creada por el supervisor.

Otros entregables importantes de esta iteración fueron la construcción de los módulos que gestionan los objetos de negocio como categorías, productos y clientes.

**Figura 2**

*Backlog para el segundo sprint en Azure DevOps*

Order	ID	Title	Assigned To	State	Tags
1	47	> Crear módulo de campañas.	Sebastian Reza	Done	
2	49	> Crear módulo de clientes.	Sebastian Reza	Done	
3	46	> Crear módulo de categorías.	Sebastian Reza	Done	
4	45	> Crear módulo de productos.	Sebastian Reza	Done	

## Tercer *sprint*

El tercer *sprint* (Figura 3) fue ejecutado del 28 de mayo al 30 de junio del 2021, llevando a cabo la construcción de la aplicación móvil, la misma que será utilizada como herramienta por el agente de ventas para desarrollar sus campañas asignadas y facilitar la contactabilidad a los clientes presentes, en su respectivo detalle de ofertas. Finalmente, la implementación de un tablero de reportes, el cual muestra información relevante acerca de la ejecución del flujo de ventas, añadiéndole la construcción del módulo que gestiona la creación y consulta de **órdenes**, tanto a nivel de agente, como de administrador.

**Figura 3**

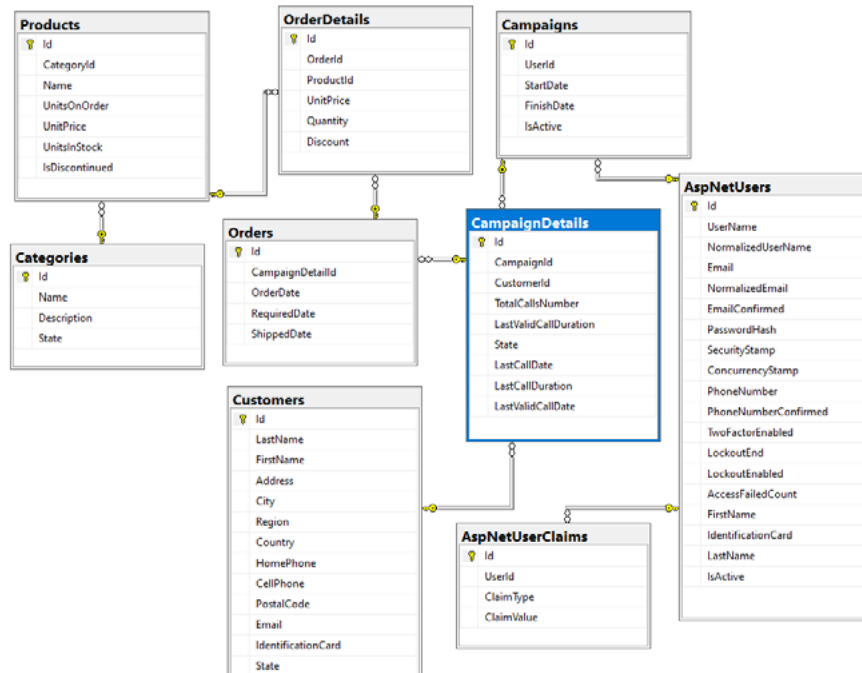
*Backlog para el tercer sprint en Azure DevOps*

Order	ID	Title	Assigned To	State	Tags
1	53	> Crear módulo de acciones para aplicación móvil.	Sebastian Reza	Done	
2	52	> Crear módulo de campaña para aplicación móvil.	Sebastian Reza	Done	
3	150	> Crear módulo de reportes.	Sebastian Reza	Done	
4	51	> Crear módulo de cuentas y seguridad para aplicación móvil.	Sebastian Reza	Done	
5	140	> Crear módulo de usuarios.	Sebastian Reza	Done	
6	50	> Crear módulo de órdenes.	Sebastian Reza	Done	

En la *Figura 4* se muestra el modelo físico de la base de datos empleada por el sistema para realizar la persistencia de datos.

**Figura 4**

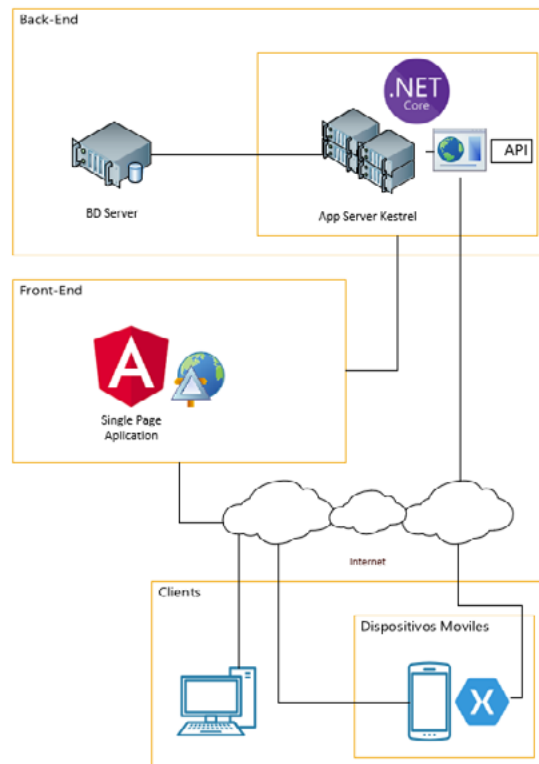
*Base de datos del sistema*



En la siguiente *Figura 5* se expone la arquitectura que emplea el sistema a nivel de hardware y software utilizados.

**Figura 5**

*Infraestructura de la solución*



A continuación, se presenta la arquitectura en la que se desenvuelve la transaccionalidad del sistema, básicamente se compone de los siguientes elementos de hardware y software:

- **Back-end:** es la parte de la solución que se conforma por todo el hardware y software encargado de gestionar la información que ingresa por las diferentes interfaces que expone el sistema. El *back-end* posibilita el acceso a las operaciones de configuración del sistema y de las funcionalidades puestas a disposición (Solon et al., 2015).
- **Servidor de base de datos:** es la infraestructura en la cual se apoya el motor de base de datos relacional, el cual se encarga de gestionar y persistir las transacciones que demande el sistema. Para este desarrollo se ha seleccionado SQL Server 2019 como motor de base de datos, estos componentes conforman la parte del *back-end* de la solución.
- **Servidor de aplicaciones:** es la infraestructura en la cual se apoya el API RESTful, el mismo que se encarga de exponer toda la funcionalidad que las interfaces de la aplicación consumen; con el objetivo de consultar, transaccionar y mostrar toda la información que se genera en los flujos del sistema, para la construcción de este componente su utilizo la tecnología .Net Core (Paco et al., 2009).
- **API:** es un acrónimo del término inglés Application Programming Interface (en español: interfaz de programación de aplicaciones). Su principal característica es que permiten que



sus productos se comuniquen con otros sin que estos conozcan su implementación. Su funcionamiento se considera que es parecido a un contrato, ya que, si el emisor envía una solicitud con una estructura conocida, el receptor podrá realizar la operación pertinente de forma adecuada y devolver una respuesta con un cierto formato establecido (de la Maza Valles, 20209).

- **.Net Core:** es un marco multiplataforma de alto rendimiento y código abierto para crear aplicaciones modernas, habilitadas para la nube y conectadas a Internet (Microsoft, 2021).
- **Front-end:** es el componente del sistema que se encuentra disponible al público (Solon et al., 2015), encargado de recibir y enviar al *back-end* todas las transacciones que demanda el usuario y sin dejar de lado que es el mismo componente que se encarga de mostrar la información solicitada el mismo.
- **Single page application:** aplicación web que carga completamente todos los recursos en la solicitud inicial y luego los componentes de la página son reemplazados por otro componente dependiendo de la interacción del usuario es decir no se da un recargo de toda la página para cada petición (Jadhav et al., 2015).
- **Angular:** es una plataforma de desarrollo construida sobre TypeScript, con un marco basado en componentes para crear aplicaciones web escalables, cuenta con una colección de bibliotecas bien integradas que cubren una amplia variedad de características, que incluyen enrutamiento, administración de formularios, comunicación cliente-servidor entre algunas otras (Angular, s.f).
- **TypeScript:** es un lenguaje de programación basado totalmente en JavaScript que se enfoca en producir código seguro y predecible y este puede ser ejecutado por cualquier tiempo de ejecución de JavaScript. Su característica principal es la escritura estática, que hace que trabajar con JavaScript sea más predecible y cómodo para los programadores familiarizados con lenguajes tipados (Freeman, 2021).
- **Xamarin:** plataforma de código abierto para crear aplicaciones modernas y eficaces para iOS, Android y Windows con .NET. Xamarin es una capa de abstracción que administra la comunicación del código compartido con el código de la plataforma subyacente. Este patrón permite a los desarrolladores escribir toda su lógica empresarial en un solo idioma (o reutilizar el código de la aplicación existente) pero lograr un rendimiento, una apariencia y un funcionamiento nativos en cada plataforma (Microsoft, s.f).
- **Clean Architecture:** esta arquitectura se empleó específicamente para la construcción del API Rest, lo que busca es que el *core* del negocio que maneja la aplicación sea altamente desacoplable tanto a los *frameworks* ORM, como a los *frameworks* que se especializan para construir *front-end*. Uno de los beneficios importantes que podemos destacar de esta arquitectura es que al no tener lógica de negocio en la base de datos se puede tratar a este componente como un simple repositorio, al cual el núcleo del negocio se puede desvincular cuando la organización lo crea conveniente, de la misma forma funcionará el desacoplamiento al *framework* encargado de gestionar el *front-end*, porque de igual manera, la arquitectura direcciona a la inexistencia de lógica de especializada para el núcleo del negocio en la capa de presentación.

Aplicando *clean architecture* y respetando las normas establecidas, independientemente de si posteriormente se implementa mejor o peor, como mínimo obtendremos un software robusto, el cual nos proporcionará un software fácil de mantener, puesto que el desacoplamiento que proponen las capas brindará a nuestro programa del principio SOLID Open/Closed, lo que quiere decir que estará abierto a extensión, pero cerrado a modificación; conseguiremos que el software sea propio y no dependa de cambios externos (González, 2016).

## Resultados

El sistema se basó en dos perfiles de usuario indispensables para poder esquematizar, organizar y persistir almacenar la información del flujo de ventas. La siguiente *Tabla 3* muestra las actividades que desempeña cada perfil en el sistema.

**Tabla 3**

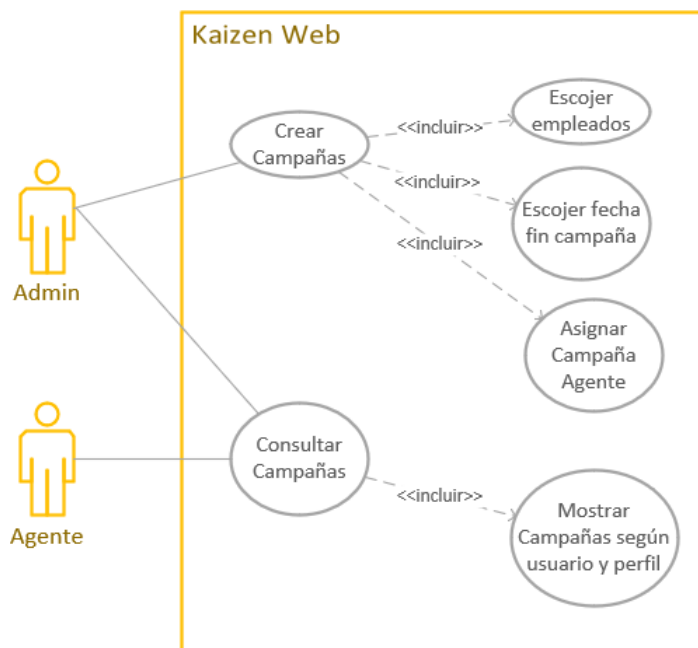
*Tabla de Perfiles de usuario*

Nombre de Usuario	Tipo de Usuario	Área Funcional	Actividad
Administrador	Administrador	Administración	Administrar clientes. Administrar categorías. Administrar productos. Administrar usuarios. Administrar campañas. Visualizar progreso campañas. Visualizar dashboard. Visualizar órdenes. Administrar clientes. Gestionar campañas asignadas.
Agente	Agente de Ventas	Ventas	Gestionar campañas asignadas. Sincronizar llamadas del día. Generar órdenes. Visualizar solamente sus órdenes generadas.

Con el fin de agilizar el proceso de asignación de clientes, que tiene por objetivo realizar la oferta de productos o promociones del negocio, el sistema propone el siguiente flujo que se muestra en la *Figura 6*:

**Figura 6**

*Casos de uso para la creación, asignación y consulta de campañas*



El usuario administrador ingresa al sistema para ejecutar la acción que lleva a cabo la creación de una nueva campaña. Esta acción debe cumplir con tres pasos subsecuentes los cuales son: elegir los empleados que conformarán la campaña, seleccionar una fecha fin para la campaña y escoger un agente a quien otorgarle la nueva campaña a crearse.

Con esta implementación se evita el manejo de archivos tipo hojas de cálculo, archivos planos, e incluso, libros de registro que estén destinados a manejar esta información de forma deficiente. De esta forma, el sistema es el único gestor de la información que fluye para los objetos principales de esta funcionalidad, como lo son las campañas y los clientes.

Otro punto importante que se gana con esta funcionalidad en el sistema es que se evita pérdidas de información, ya sea en una transmisión por correo electrónico o físicamente, referente a estas asignaciones de trabajo, ya que, de manera automática e inmediata, la campaña que asigna el administrador por medio del sistema aparecerá en la bandeja del usuario agente para que sea trabajada.

Es importante recalcar que el administrador puede consultar y ver el progreso de cualquier campaña perteneciente a cualquier agente en el ámbito del sistema, lo cual resulta sumamente beneficioso para que el administrador cuente con un conocimiento real del avance de sus colaboradores.


Las siguientes *Figuras 7 y 8* muestran el aspecto del formulario para crear y asignar campañas del lado del usuario administrador, como de la página que muestra la bandeja que expone estas campañas asignadas del lado del usuario agente.

**Figura 7**

*Página creación y asignación de campañas perfil administrador*

Kaizen Home Administration Customers Orders hello! jsrezam@kaizen.com Logout

### New Campaign



**Sthefanny Ramirez**  
ssrg\_2104@kaizen.com  
1716652191

Agent: Sthefanny Ramirez

Finish Campaign: 2021-08-31

[Create Campaign](#)

Customers in the new campaign:

Search options:

Generate Random Campaign: Enter max random range [Generate](#)

Max Range available: 9 customers

[Select All](#)

First Name	Last Name	ID	Email	Cell Phone	Phone	Country	Region	City	Zip	Address	
<input checked="" type="checkbox"/>	Leonardo	Padilla	1715987453	lpadilla@gmail.com	0987451236	2798371	Ecuador	Pichincha	Quito	1234w1	Av. 1234 MST strett
<input checked="" type="checkbox"/>	Andre	Lopez	1748785919	alopez@gmail.com	0987456321	2365147	Ecuador	Guayas	Milagro	128745	Av. Mariscal

**Figura 8**

*Página bandeja de campañas asignadas perfil agente*

Kaizen Home Customers My Campaigns Orders hello! ssrg\_2104@kaizen.com Logout

### My campaigns

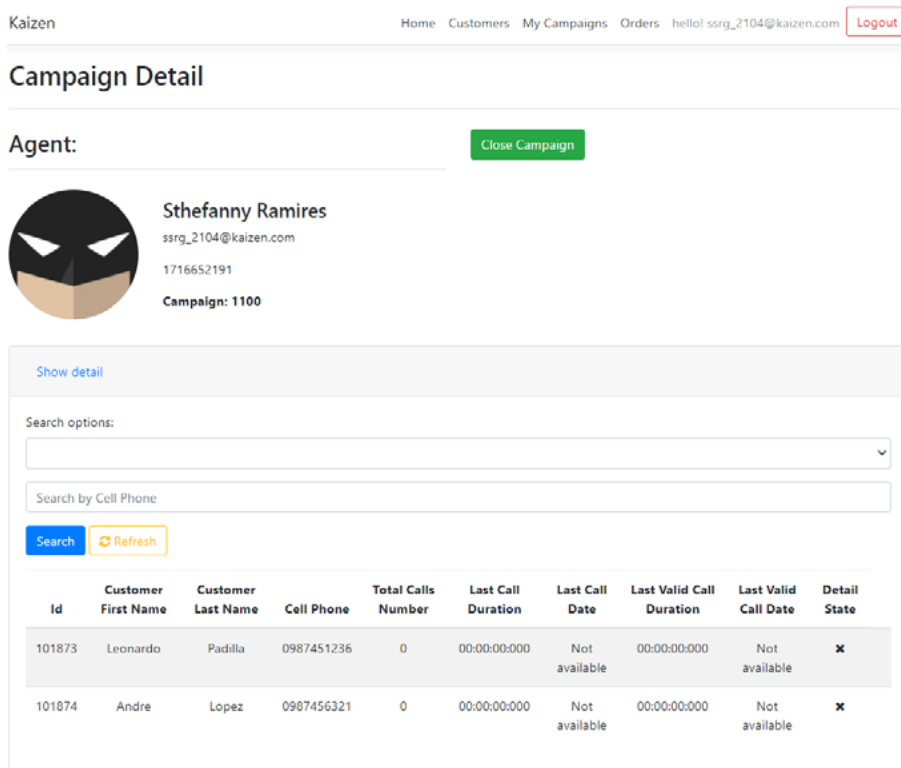
Campaign	Start Date	Finish Date	State	Progress
1100	8/24/21, 11:56 PM	8/31/21, 5:00 AM	Open	<div style="width: 100%; height: 10px; background-color: #ccc;"></div>

[View Detail](#)

El perfil agente podrá ver el detalle de su campaña a desarrollar en la página de desglose de campaña (*Figura 9*), la misma que mostrará para cada oportunidad presente en la campaña los clientes a contactar.

**Figura 9**


*Página desglose de campaña perfil agente*



Kaizen Home Customers My Campaigns Orders hello! ssrcg\_2104@kaizen.com Logout

### Campaign Detail

Agent: Close Campaign

 **Sthefanny Ramires**  
 ssrcg\_2104@kaizen.com  
 1716652191  
 Campaign: 1100

Show detail

Search options:

Search by Cell Phone

Search Refresh

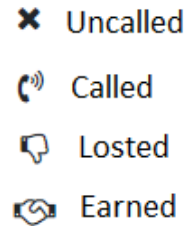
Id	Customer First Name	Customer Last Name	Cell Phone	Total Calls Number	Last Call Duration	Last Call Date	Last Valid Call Duration	Last Valid Call Date	Detail State
101873	Leonardo	Padilla	0987451236	0	00:00:00:000	Not available	00:00:00:000	Not available	✘
101074	Andre	Lopez	0987456321	0	00:00:00:000	Not available	00:00:00:000	Not available	✘

Existen los siguientes cuatro estados (*Figura 10*) que pueden tomar las oportunidades según las circunstancias que se exponen a continuación:

- **Uncalled:** la oportunidad tiene este estado cuando el agente aún no se ha contactado con el cliente presente en la oferta.
- **Called:** la oportunidad tiene este estado cuando el cliente ya se contactó con el cliente presente en la oferta.
- **Losted:** la oportunidad tiene este estado cuando la campaña se cerró y la oferta no concretó una generación de orden.
- **Earned:** la oportunidad tiene este estado cuando la oferta se concretó con una generación de orden.

**Figura 10**

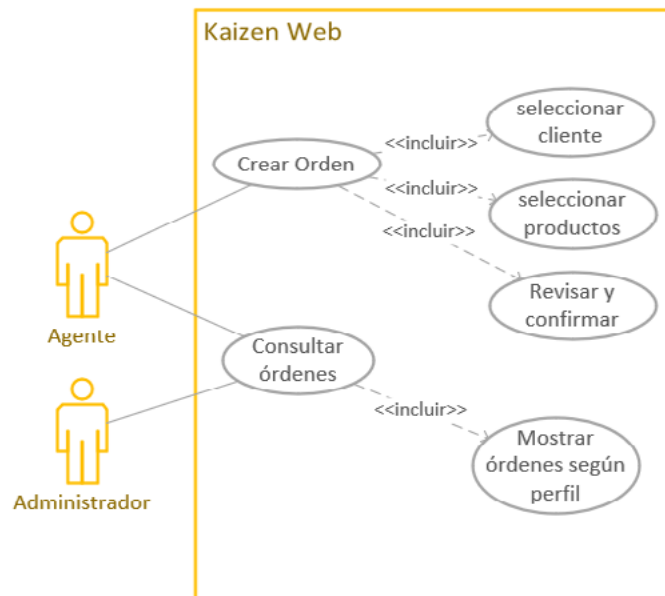
*Estados de la oferta*



Otra funcionalidad que aporta significativamente al negocio es cuando el agente crea o concreta pedidos (*Figura 11*). Con la correcta estructuración y gestión de la información que propone el sistema, este brinda agilidad, seguridad, integridad y disponibilidad a estas transacciones, lo cual mejora significativamente a comparación del trato manual con estas acciones, ya que, al manejar archivos sueltos sin una debida seguridad, ni un confiable almacenamiento, existe un riesgo bastante grande de perdida de información o modificación no autorizada de datos.

**Figura 11**

*Casos de uso para generar órdenes*



Para crear una orden el agente tendrá que escoger un cliente que exista en su campaña en progreso, asignarle los productos en los que esté interesado, y como paso final, validar el detalle

generado con sus respectivos montos, cantidades e información básica del cliente en una interfaz de *check-out*.

Como se puede observar en el caso de uso el usuario administrador, este tendrá acceso *read-only* a cualquier orden generada por cualquier agente existente en el ámbito del sistema, a diferencia del usuario agente de ventas que solamente podrá consultar sus propias ordenes generadas.

Las siguientes *Figuras 12, 13 y 14* muestran el aspecto de las interfaces para generar una orden en sus tres etapas: selección de cliente, selección de productos y *check-out* de orden.

## Figura 12

*Página para generar órdenes interfaz de selección de cliente*

The screenshot displays the 'New Order' page in the Kaizen system. At the top, the navigation bar includes 'Kaizen', 'Home', 'Customers', 'My Campaigns', 'Orders', and a user profile 'hello! srrg\_2104@kaizen.com' with a 'Logout' button. The main heading is 'New Order'. Below it, a progress indicator shows two steps: '1 Select Customer' (active) and '2 Select Products'. The 'Select Customer' section features a 'Search options:' dropdown menu, a text input field containing '0987451236', and a blue 'Search' button. To the right, the 'Selected Customer:' information is shown, including a circular profile picture of a person with a black mask, the name 'Leonardo Padilla', the phone number '0987451236', and a table with 'Campaign: 1100' and 'Offer: 101873'.

**Figura 13**

*Página para generar órdenes interfaz de selección de producto*

Kaizen Home Customers My Campaigns Orders hello! ssrc\_2104@kaizen.com Logout

## New Order

1 Select Customer 2 **Select Products** 3 Checkout

Search options:

**Cart Resume:**  
 Items: 1  
 Total (USD): **\$773.97**

Product Id	Category	Name	Unit Price	Stock	Units on Order
82875	Technology	Lenovo Chromebook S330 portátil	\$257.99	75	24

**Items:**

**Lenovo Chromebook S330 portátil**

Unit Price: **\$257.99**

Quantity: **3**

Import: **\$773.97**

**Figura 14**

*Página para generar órdenes interfaz de check-out*

Kaizen Home Customers My Campaigns Orders hello! ssrc\_2104@kaizen.com Logout

## New Order

1 Select Customer 2 Select Products 3 **Checkout**

**Customer Info:**

**Leonardo Padilla**  
 Ubication: Av. 1234 MSt strett Quito - Pichincha - Ecuador  
 Cell Phone: 0907451236  
 Home Phone: 2790371  
 Postal Code: 1234w1  
 Id: 1715907453  
 Email: lpadilla@gmail.com  
 Campaign Number: 1100  
 Detail Number: 101073

**Customer's Cart:**

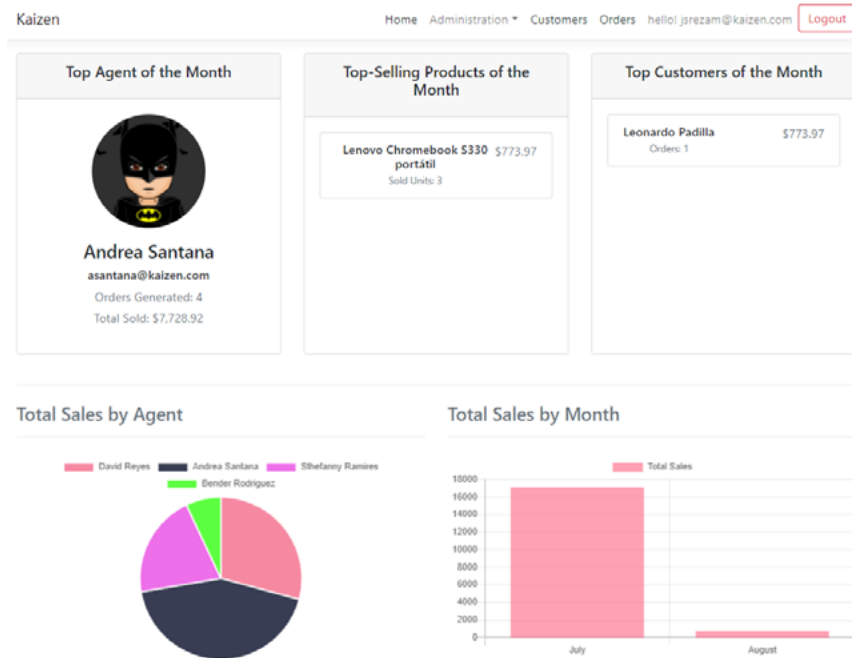
Lenovo Chromebook S330 portátil	\$773.97
\$257.99 x 3	
<b>Total (USD)</b>	<b>\$773.97</b>



Finalizando con la parte web del sistema, las siguientes imágenes muestran el tablero informativo (*Figura 15*) sobre las ventas efectuadas en el flujo. Esta interfaz da a conocer varios puntos importantes para definir el estatus actual del negocio.

**Figura 15**

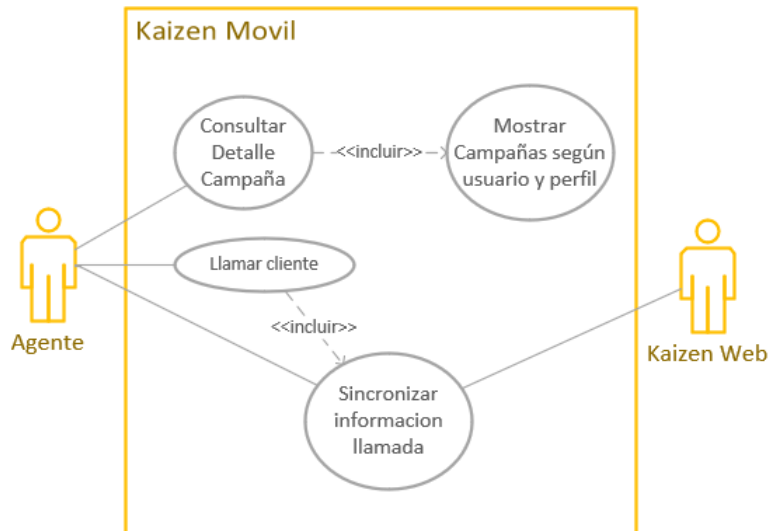
*Página que muestra el dashboard informativo*



Con el fin de generar veracidad al momento de contactar con los diferentes clientes, se presentan las oportunidades de la campaña, se ha desarrollado un aplicativo móvil, el cual tiene por objetivo sincronizar la información que se genera en el dispositivo móvil al momento de realizar llamadas (*Figura 16*), con esta funcionalidad, el administrador podrá saber cuántas veces intento el agente contactarse con el cliente y cuánto duro su última llamada.

**Figura 16**

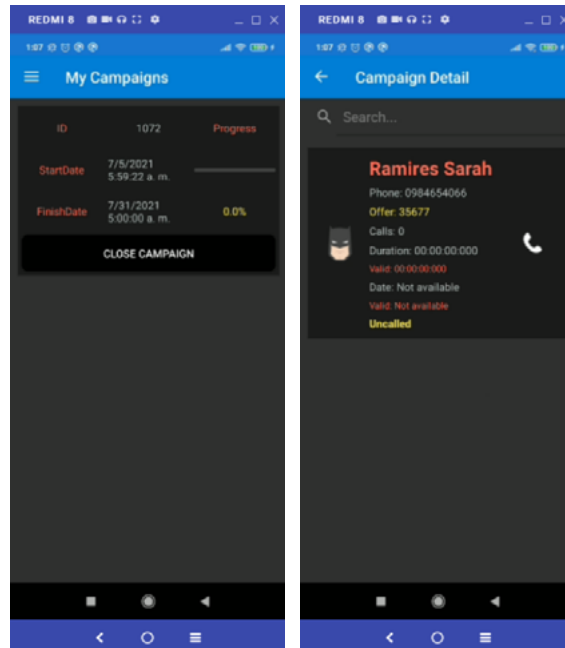
*Casos de uso para generar la sincronización de la información de llamadas*



Como se puede observar en el caso de uso, el aplicativo web, al utilizar el mismo API que gestiona la información en el mismo repositorio de datos, también quedará actualizado con la información de las llamadas realizadas por el agente, al momento de ejecutar estas acciones de sincronización desde la aplicación del dispositivo móvil. Las siguientes *Figuras 17 y 18* muestran el aspecto de las interfaces del aplicativo móvil, al momento que se realiza una sincronización de llamadas.

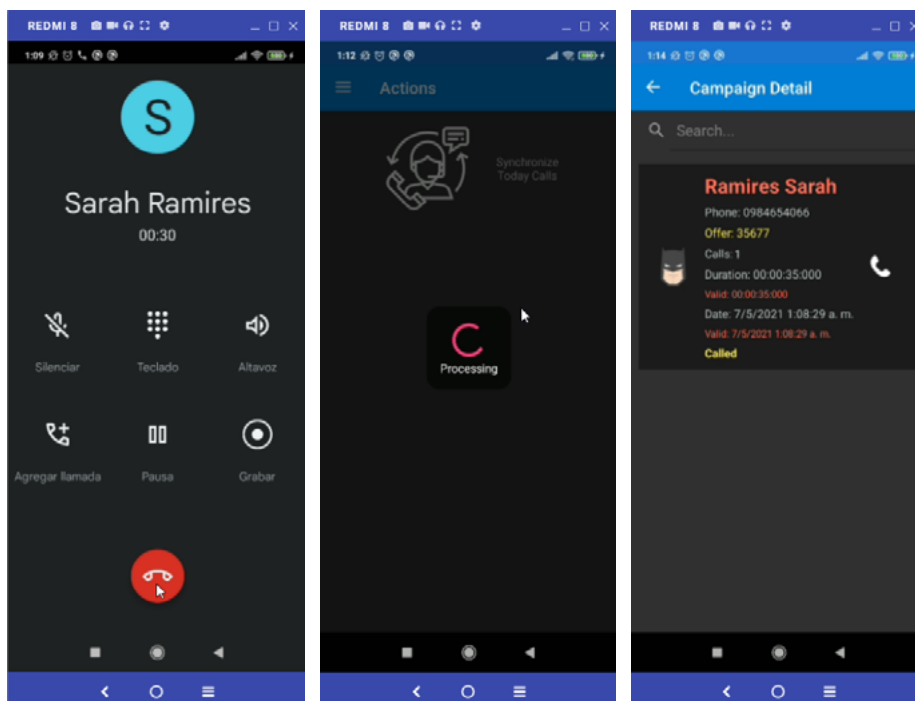
**Figura 17**

*Pantallas que muestran la campaña con su respectivo desglose*



**Figura 18**

*Sincronización de información de llamadas*



Consumando el resultado positivo para cada uno de los siguiente puntos clave que busca cubrir el sistema:

- Reducción de trabajo manual.
- Organización y estructuración correcta de la información que genera el negocio.
- Asegurar la integridad en la información que genera el negocio.
- Brindar seguridad a la información y funciones que resguarda el sistema.

## Conclusiones

Sin duda, el sistema construido aliviana un alto porcentaje el trabajo manual que existe en el ámbito de ventas. También se enfoca en remover las inconsistencias que produce un flujo operado sin una debida seguridad administrable ni una gestión de persistencia apropiada de los datos. Este software, en su operación diaria, generará información valiosa y confiable, que, a su tiempo, ayudará a las gerencias de estos pequeños negocios con la toma de decisiones que los enruten al siguiente paso tecnológico.

Respaldar y solidificar la funcionalidad de la aplicación es uno de los puntos medulares de cualquier proyecto de software. Es de imperiosa necesidad asegurar que el producto este probado bajo un contexto correcto de carga y estrés transaccional, con el objetivo de evitar lentitud, y en el peor de los casos, la indisponibilidad del sistema en un ambiente productivo.

## Referencias

- Angular. (s. f.). What is Angular? <https://angular.io/guide/what-is-angular>
- Artieda Guachamin, C., Reza Morales, J., & Rojas Cisneros, H. (2021). Sistema móvil para la oferta de recorridos con rutas fijas. *Revista ODIGOS*, 2(1), 21-40. <https://doi.org/10.35290/ro.v2n1.2021.381>
- Jadhav, M. A., Sawant, B. R., & Deshmukh, A. (2015). Single page application using angularjs. *International Journal of Computer Science and Information Technologies*, 6(3), 2876-2879.
- Burbano Ulloa, E., Constante Molina, M., Hidalgo Guamán, L., & Moya Chiluzia, F. (2020). Prototipo móvil para la geolocalización de mascotas callejeras. *Revista ODIGOS*, 1(3), 77-96. <https://doi.org/10.35290/ro.v1n3.2020.372>
- de la Maza Valles, A. (2020). *API REST para la gestión de partidas multijugador de un juego serio*. [Tesis de grado, Universidad de Valladolid]. UVaDOC <http://uvadoc.uva.es/handle/10324/44417>
- Freeman, A. (2021). Understanding typescript. In *Essential TypeScript 4* (pp. 35-41). Apress. [https://doi.org/10.1007/978-1-4842-4979-6\\_2](https://doi.org/10.1007/978-1-4842-4979-6_2)
- González Díez, M. (2016). *Clean architecture y RxJava en Android*. [Tesis de grado, Universitat de Barcelona]. <http://hdl.handle.net/2445/102790>
- Microsoft. (22 de enero de 2021). What is Azure DevOps? <https://docs.microsoft.com/en-us/azure/devops/user-guide/what-is-azure-devops?view=azure-devops>
- Microsoft. (s.f). Xamarin documentation. <https://docs.microsoft.com/en-us/xamarin/>
- Paco Blanco, J., Fumero, A., Werterski, A., & Rodríguez, P. (2009). Metodología de desarrollo ágil para sistemas móviles Introducción al desarrollo con Android y el iPhone. <http://radar.oreilly.com/2008/07/inte-rest-in-the-iPhone-android.html>
- Reyes Rivera, A. J., Vallejos Cango, M. de L., & Quintana García, D. A. (2020). Nueva realidad: compras en línea y a domicilio. *Revista ODIGOS*, 1(3), 47-60. <https://doi.org/10.35290/ro.v1n3.2020.370>
- Riera, E. G. (2020). Sistema de gestión documental para la empresa Logikard. *Revista ODIGOS*, 1(3), 61-76. <https://doi.org/10.35290/ro.v1n3.2020.371>
- Solon Guimarães Jr, C., Bigolin Wiedemann, R., Rubio-Tamayo, L., & Ventura Bayan Henriques, R. (2015). Propuesta de Front-end y Back-end para un Servicio Web Aplicado en Tecnología Asistiva.
- Trasobares, A. H. (2003). Los sistemas de información: evolución y desarrollo. *Proyecto social: Revista de relaciones laborales*, (10), 149-165.

Copyright (c) 2021 Juan Sebastián Reza Morales



Este texto está protegido bajo una licencia internacional [Creative Commons](#) 4.0.

Usted es libre para Compartir—copiar y redistribuir el material en cualquier medio o formato — y Adaptar el documento — remezclar, transformar y crear a partir del material—para cualquier propósito, incluso para fines comerciales, siempre que cumpla las condiciones de Atribución. Usted debe dar crédito a la obra original de manera adecuada, proporcionar un enlace a la licencia, e indicar si se han realizado cambios. Puede hacerlo en cualquier forma razonable, pero no de forma tal que sugiera que tiene el apoyo del licenciante o lo recibe por el uso que hace de la obra.

[Resumen de licencia](#) – [Texto completo de la licencia](#)