

Aplicación móvil para la búsqueda y realización de pedidos de restaurante a domicilio



Colaboración

Juan Antonio Rivero Fierro; Juan Miguel Hernández Bravo; Mario Hernández Hernández; Francisco Javier Gutiérrez Mata, Tecnológico Nacional de México / IT Acapulco

Fecha de recepción: 14 de octubre del 2022

Fecha de aceptación: 02 de mayo del 2023

RESUMEN: El presente trabajo tiene como objetivo diseñar una solución que permita realizar una aplicación móvil que facilite al personal del restaurante Ensaladas Green, automatizar los procesos de despacho de pedidos que realizan los comensales, para hacer su entrega en forma y tiempo. La principal problemática es la falta de una App, que actualmente realizan la toma de pedidos a mano mediante formatos hechos para ello, pero que no es gestionado a través de ninguna App, en la cual se tiene la probabilidad de errores en la captura de los pedidos y pérdida de tiempo en la corrección de los mismos. Así mismo, generan retrasos al momento de dar a conocer al área de expedir el envío del pedido.

PALABRAS CLAVE: Aplicación Móvil, Automatización, Gestión, Pedido.

ABSTRACT: The purpose of this work is to design a solution that allows the creation of a mobile application that facilitates the staff of the Ensaladas Green restaurant, receiving automatic orders from the dispatch processes originated or facilitated by diners, to make their delivery in a timely manner. The main problem is that there is no app, which currently takes orders by having employees in the business, but that is not managed through any app. This situation can lead to having errors in capturing the orders and loss of time in correcting them. Likewise, they create delays at the time of making known to the area of issuing the shipment of the order.

KEY WORDS: Mobile Application, Automation, Management, Order.

INTRODUCCIÓN

El restaurante Ensaladas Green, actualmente realizan la toma de pedidos a mano mediante formatos hechos para ello, pero que no es gestionado a través de ninguna App, en la cual se tiene la probabilidad de errores en la captura de los pedidos y pérdida de tiempo en la corrección de los mismos. Así mismo, generan retrasos al momento de dar a conocer al área de expedir el envío del pedido. El presente trabajo tiene como objetivo diseñar una solución que

permita realizar una aplicación móvil que facilite al personal del restaurante Ensaladas Green, automatizar los procesos de despacho de pedidos que realizan los comensales, para hacer su entrega en forma y tiempo.

En los negocios de alimentos y bebidas, las aplicaciones de servicio de entrega (delivery service) son muy atractivas para los usuarios y permiten a los dueños de restaurantes anunciar sus productos de una manera más intuitiva y completa. Los empresarios que inician tienen una necesidad precisa de construir una cartera de clientes y darlos a conocer en su campo de actividad. El mercado ha evolucionado y la sociedad se adaptó de inmediato por conveniencia y economía. Existe un nuevo modelo de negocio y atención al cliente a través del uso de internet, en la actualidad el método tradicional de esperar a los clientes detrás del mostrador ha cambiado drásticamente, el uso de dispositivos inteligentes se ha incrementado a pasos agigantados y con ello el uso de aplicaciones que permiten recibir todo tipo de ofertas, comparar y elegir las mejores.

METODOLOGÍA

El desarrollo de este trabajo se lleva a cabo a través del método ágil Scrum, elegido porque se basa en prácticas de autogestión y organización del proyecto, fomenta una mayor interacción con los clientes, aumenta la tasa de aceptación del producto final a través de la entrega continua, lo que nos ayuda a facilitar el análisis, el diseño, la evolución y luego implementar el sistema final.

El modelado ágil (MA) es un método basado en ejercicios para modelar y documentar sistemas de software. En pocas palabras, es un conjunto de estándares, principios y prácticas que permiten que un modelo de software se use de manera efectiva y eficiente en proyectos de desarrollo de software. Los modelos ágiles son más eficientes que los modelos tradicionales, porque son más prácticos sin pretender ser perfectos, ya que se pueden entregar avances, sin esperar al final del término del proyecto [1].

Definición de los roles de Scrum

Esta metodología considera tres roles: Product Owner, Scrum Master, El equipo de desarrollo, de los cuales se distribuyen de la siguiente manera, ver Tabla 1.

Tabla 1. Roles de Scrum.

Rol	Responsabilidades
Product Owner	Ensaladas Green Acapulco
Scrum Master	Director de la Tesis.
El equipo de desarrollo	Autor de la tesis

- Director de la Tesis: Designado por la institución Tecnológico de Acapulco, para la revisión y avance del proyecto, por lo cual es asignado al rol de Scrum Master.
- Ensaladas Green Acapulco: Él es la persona que se comunica directamente dentro de Restaurante Ensaladas Green y tiene un conocimiento profundo de las

necesidades dentro de las instalaciones de las que se derivan las historias de los usuarios.

- Desarrollador del proyecto: Es la persona encargada de realizar el levantamiento y codificación de las tareas de las historias de usuario.

Creación de las historias de usuario

Las historias de usuario se utilizan de forma flexible en la especificación de requisitos, que es una breve descripción de la funcionalidad del software percibida por los usuarios [2].

Describen funciones que brindan una solución a la necesidad o problema de un cliente o usuario, representan funciones que se construirán y están escritas en una historia de una o dos oraciones utilizando la lengua franca del usuario.

Las historias de usuario se obtienen durante reuniones directas con los jefes de todas las operaciones involucradas en el restaurante, donde se utilizará la aplicación móvil, utilizando un formato para obtener información, ver Tabla 2 [3].

Tabla 2. Consulta de la carta del establecimiento.

Historia de usuario	
Número: HU01	Usuario: Comensal.
Nombre historia: Consulta de la carta del establecimiento.	
Prioridad en negocio: ALTA	Riesgo en desarrollo: BAJA
Puntos estimados: 5	Iteración asignada: 1
Programador responsable: Juan Antonio Rivero Fierro.	
Descripción: Como comensal, la aplicación debe permitirme acceder a la carta del menú.	
Validación:	
✓ Consultar todos los platillos de la carta del menú.	
✓ Consultar precios	

- Número: Es asignado a cada historia y es utilizado como identificador.
- Usuario: Persona que proporciona la necesidad y hará uso de ella.
- Nombre de historia: Título descriptivo de la historia de usuario.
- Prioridad en el negocio: Es el valor de negocio que aporta una historia de usuario. Se usa los términos bajo, medio y alta.
- Riesgo en desarrollo: Riesgo funcional en el desarrollo de la historia. Se usa los términos bajo, medio y alto.
- Puntos estimados: Es la estimación de esfuerzo dada a la historia, se usó la secuencia Planning Poker.
- Iteración asignada: Es el número asignado de acuerdo al Sprint donde se plantea codificarla.
- Programador responsable: Persona que se encarga de la codificación, dado a los roles asignados previamente todas son asignadas a la misma persona en este caso.
- Descripción: Descripción general de lo que trata la historia de usuario, siguiendo el patrón "Como [rol del usuario], quiero [objetivo], para poder [beneficio]".
- Validación: Criterios de aceptación a evaluar por el usuario final.

Estructuración del Product Backlog

Las historias de usuarios se priorizan en el Backlog en función de las prioridades del establecimiento, los riesgos de desarrollo y los puntos estimados. Uso de la herramienta MoSCoW [3].

Uno de los métodos de priorización más utilizados se denomina método MoSCoW. Es un método de priorización de requisitos basado en que, si bien todos los requisitos se consideran importantes, se deben seleccionar y considerar obligatorios aquellos que aporten más valor al negocio para que el producto o servicio no se ponga en producción si no es así. Esto nos ayudará a monitorear nuestro progreso de cerca.

El enfoque **MoSCoW** no se limita a indicar si una necesidad es baja, media o importante. Este proceso nos ayudó a dividir las necesidades en cuatro grandes categorías [4].

- **Must Have** – Debe tener (es necesario): Requisitos fundamentales y obligatorios para que proyecto se considere como éxito, deben de llevarse a cabo como prioridad y no es negociable.
- **Should Have** – Debería de tener (es recomendable): Requisitos que son críticos, pero no imprescindibles en nuestro proyecto, la aplicación no fallará si no se implementa.
- **Could Have** – Podría tener (podría implementarse): Requisitos que son interesantes tenerlos, ya que mejorarían el rendimiento de la aplicación, pero podrían ser eliminados fácilmente.
- **Won't** – No tendrá (no lo queremos...quizá en algún futuro): Requisitos que se ha decidido no implementar de momento, pero que serán tomados en cuenta en un futuro.

Esta categorización nos ayuda a marcar un producto mínimo aceptable, definir los requisitos que se deben incluir en el desarrollo del servicio. Es muy fácil de entender y aplicar en la práctica para todas las partes interesadas, porque definimos claramente los requisitos básicos con la letra M (Must), sin los cuales no se puede producir el servicio de la aplicación móvil.

Planeación de los Sprint

El equipo de desarrollo está formado por una sola persona, por lo que las tareas se distribuyen en 4 Sprints, como se muestra en la Tabla 3.

El elemento esencial es la estimación de dificultad atribuida a cada historia de usuario, donde los puntos de dificultad para el sprint uno es de 18 puntos, 24 puntos para el sprint dos, 38 puntos para el sprint tres, 30 puntos para el sprint final, el sprint uno sirva como base para entender los 3 restantes por la distribución de puntos de dificultad, como se muestra en la Tabla 3 para que sea más clara esta explicación.

Tabla 3. Planeación de los Sprint.

HISTORIA DE USUARIO		OTROS DATOS DE LA ÉPICA O HISTORIA DE USUARIO				
ID	Nombre	Criterios de aceptación	Prioridad	Estimación en horas.	Sprint	Estado
HP01	Análisis del sistema.		1	8	1	Realizado
HU01	Consulta de la carta del establecimiento.	1.- Consultar todos los platillos de la carta. 2.- Consultar precios	2	8	1	Realizado
HU02	Selección entre "Recoger" el pedido o "A domicilio."	1.- El Comensal debe de seleccionar entre dos opciones de entrega. 2.- Validar que se halla registrado la dirección de entrega.	3	10	1	Realizado
HU03	Añadir platos al carrito.	1.- El Comensal debe poder añadir platos al carrito.	4	8	2	Realizado
HU04	Eliminar platos del carrito.	1.- El Comensal debe poder eliminar platos al carrito	5	8	2	Realizado
HU05	Consultar los platos del carrito	1.- El Comensal debe poder consultar platos del carrito	6	8	2	Realizado
HU06	El usuario debe poder cerrar el pedido cuando haya terminado las operaciones sobre él.	1.- El Comensal debe poder cerrar el pedido, para poder hacer su envío.	7	8	3	Realizado
HU07	Consultar estatus del pedido.	1.- El Comensal debe poder monitorear el estatus del pedido.	8	8	3	Realizado
HU08	Modificar el estado del pedido.	1.- El administrador puede modificar el estado del pedido.	9	10	3	Realizado
HU09	Consultar información del pedido.	1.- El administrador puede consultar el estado del pedido.	10	10	3	Realizado
HU10	El encargado podrá consultar los identificadores y la hora a la que se realizó cada pedido.	1.- El administrador puede consultar los identificadores y la hora del pedido.	11	10	3	Realizado
HU11	Recibir una notificación al producirse un nuevo pedido.	1.- El administrador podrá atender los nuevos pedidos generados.	12	10	4	Realizado
HU12	El usuario determinado (Gerente o Repartidor) podrá indicar al sistema que un pedido determinado ha sido entregado y así permitir al cliente realizar un nuevo pedido cuando así lo considere.	1.- El usuario determinado (Gerente o Repartidor) podrá indicar al sistema que un pedido se entregó.	13	10	4	Realizado
HU13	El sistema debe permitir a cada usuario acceder a los ingredientes del producto seleccionado.	Cada usuario podrá acceder a los ingredientes del producto seleccionado.	14	10	4	Realizado

Descripción del Método

Modelo de negocio

En este trabajo se elaboró el diagrama de un modelado de negocio, en el cual se muestra los procesos que se llevan a cabo en cada sección del Restaurante Ensaldas Green Acapulco en el levantamiento del pedido de alimento, ver Figura 1.

A continuación, se explica en resumen en que consiste cada proceso del modelado de negocio:

- **Elegir platillo del menú:** En este proceso el comensal elige la comida del menú disponible en ese momento, para después realizar el pedido al restaurante.
- **Realizar pedido:** En este proceso el restaurante toma el pedido, para después solicitar el pago al comensal del pedido solicitado.

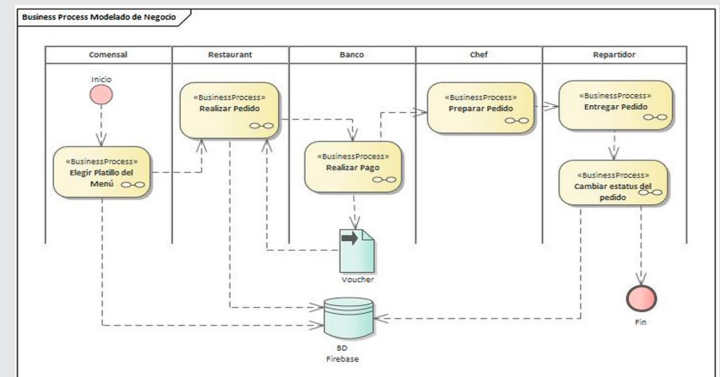


Figura 1. Modelado del Levantamiento de un Pedido de Alimento.

- **Realiza pago:** Una vez que el restaurante tenga el pago del pedido, este lo manda al chef para su preparación.
- **Preparación del pedido:** El cocinero realiza la preparación del pedido asignado, para después hacer la entrega al repartidor.

• Entrega de pedido: En este proceso el repartidor realiza la entrega al comensal para después hacer el cierre del pedido.

Caso de uso principal

Caso de uso – Proceso para crear un pedido

Las personas a las cuales están enfocado el uso de esta aplicación son a los comensales, así como el personal del restaurante de ensalada Green Acapulco.

En la Figura 2 se muestra la interacción entre el usuario y el administrador, es decir, son las acciones que puede realizar cada cliente y las funciones del administrador.

DCU-Iniciar sesión en el sistema de pedidos

• Comensal. Es el usuario que podrá crear, consultar, modificar, eliminar, el pedido del menú del restaurante que esté disponible en ese momento.

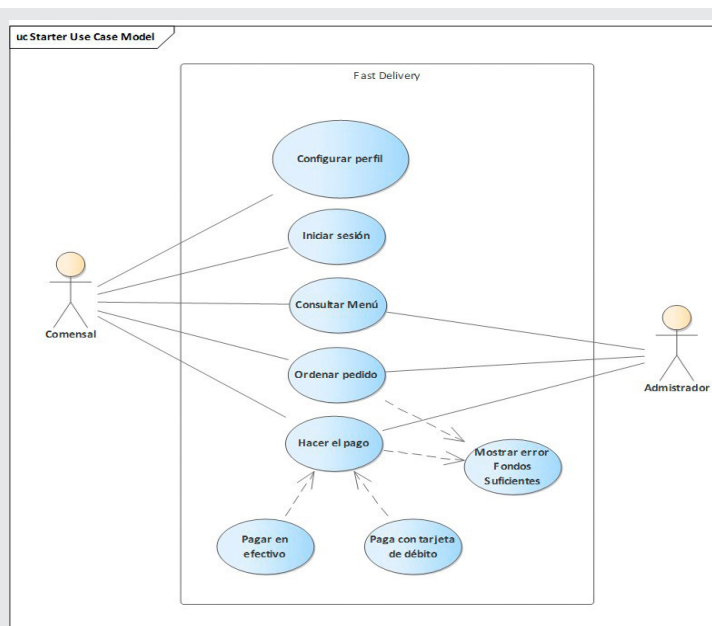


Figura 2. CDU- Cliente.

Fuente: Elaboración propia.

• Administrador. Este usuario es el que podrá dar de alta platillos, así como la disponibilidad de estos en el menú, así también como recibir pedidos realizados, pagos del comensal y mandar la preparación de estos con el chef, y así como marcar el pedido como terminado.

• Configurar perfil. Aquí el comensal puede darse de alta, modificar sus datos para iniciar sesión.

• Iniciar sesión. Aquí el comensal inicia la sesión para poder realizar un pedido.

• Consultar Menú. Aquí el comensal puede hacer la consulta del Menú de comidas disponibles en el momento.

• Ordenar pedido. Aquí el comensal realiza el pedido de su comida seleccionada.

• Hacer el pago. El comensal realiza su pago con tarjeta.

Diagrama de clases

Como se muestra en la Figura 3. Se muestran las clases responsables de las funciones que realiza el sistema, donde se observan los elementos visibles y los procesos invisibles que realizan.

Estos diagramas de clases describen lo que es una vista estática de lo que es el modelo donde se describen lo que son los atributos y comportamiento que tienen a lugar para detallar lo que son los métodos y las operaciones a realizar.

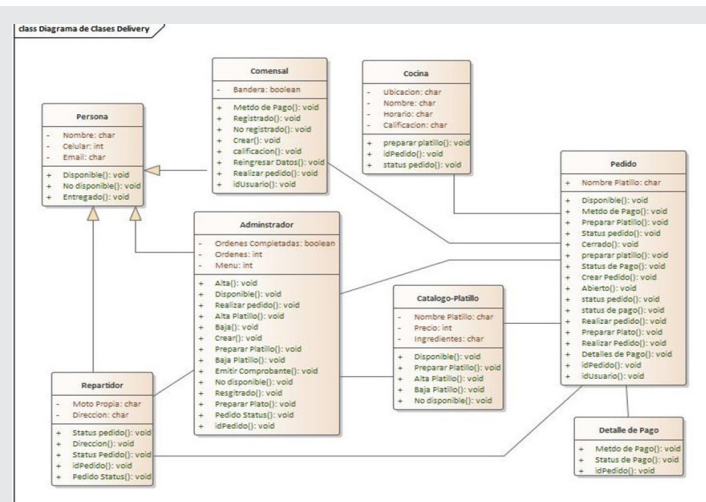


Figura 3. Diagrama de Clases.

Fuente: Elaboración propia.

A continuación, se describen cada una de las clases.

• Persona. Esta clase representa la información que hereda a otras clases.

• Comensal. Esta clase representa la información de los usuarios registrados, así como las compras realizadas.

• Administrador. Esta clase representa la persona que se encarga de dar de alta, modificar, baja del menú de platillos, así como mandar a preparar los platillos generados.

• Cocina. Esta clase representa donde preparan el pedido y le dan salida para su entrega.

• Catálogo-Platillo. Esta clase representa el menú de alimentos disponibles en la cocina.

• Pedido. Esta clase representa los pedidos generados por los comensales, así como el estatus de pendiente, terminado.

• Detalle de pago. Esta clase representa los métodos de pagos de los pedidos.

• Repartidor. Esta clase representa la persona que entrega el pedido al comensal, así como el cambio de estatus del pedido.

Diagrama de secuencia

Se elabora un diagrama de secuencia. Para la realización de pedido desde el levantamiento hasta la entrega del mismo, ver Figura 4.

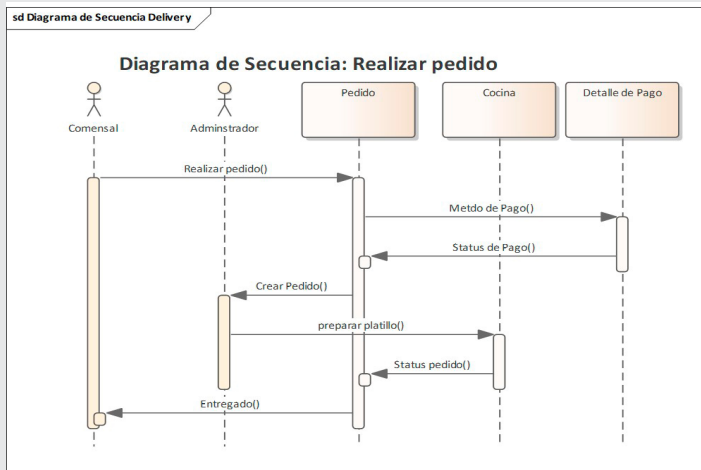


Figura 4. Diagrama de secuencia.
Fuente: Elaboración propia.

- Comensal. Realiza el pedido en el sistema.
- Comensal. Realiza el pago.
- Administrador. Recibe la notificación para crear un pedido nuevo y manda a cocina a preparar el platillo.
- Cocina. Recibe la orden y es la que prepara el platillo.
- Pedido. Una vez preparado el platillo por la cocina se cambia estatus de lista para entregar.
- Comensal. Recibe su pedido y el administrador cambia estatus a entregado y se cierra ticket.

Diagrama de despliegue

Como se muestra en la Figura 5, un diagrama de implementación es responsable de modelar la arquitectura de tiempo de ejecución del sistema, mostrando la configuración de los elementos de hardware y mostrando cómo los elementos de software y los artefactos interactúan con ellos.

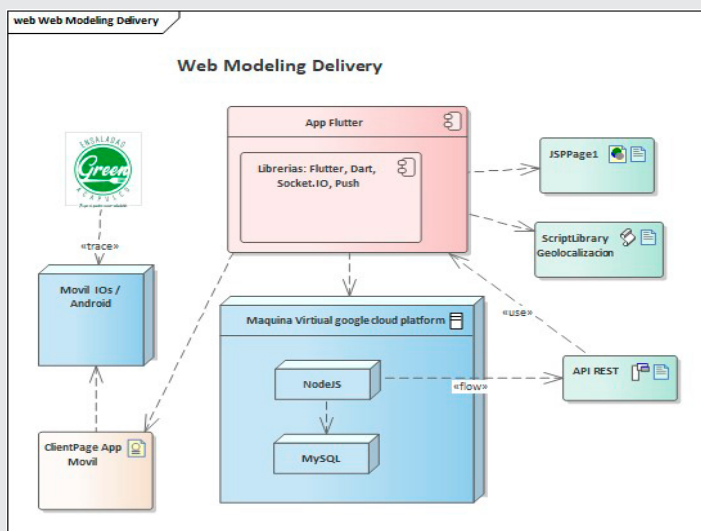


Figura 5. Diagrama de despliegue.
Fuente: Elaboración propia.

RESULTADOS

En esta sección final se analizan los resultados obtenidos en este trabajo de la aplicación móvil, en la cual

se observan las funcionalidades en operación, así como datos relevantes del Algoritmo en funcionamiento.

Administrador del sistema

El módulo está desarrollado para el usuario "Administrador", su vista principal se muestra en la Figura 6.

En la Figura 6 se muestra la vista principal del administrador del restaurante. En la vista de la Figura 6, se tiene la posibilidad de agregar productos, eliminar, editar, además de agregar promociones, horarios de apertura y cierre, así como cortes por fechas.

La vista del administrador está contenida en un menú de su lado izquierdo que posee submenús con diferentes funcionalidades para cada uno de los elementos. En la siguiente página se observan cada una de las vistas.

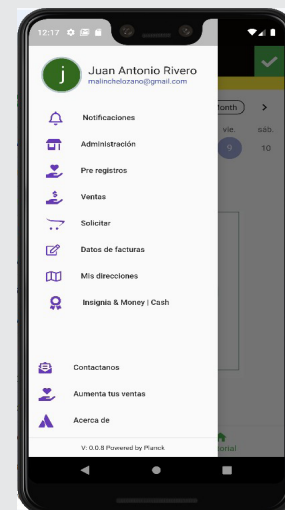


Figure 6. Vista principal del administrador del restaurante.
Fuente: Elaboración propia.

En esta vista Figura 7, se definen los días y horario de apertura y cierre del restaurante.

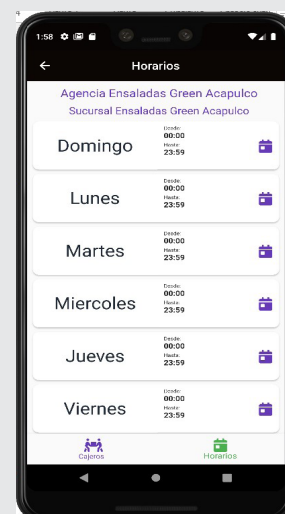


Figura 7. Vista de los horarios de cierre y apertura de la semana.
Fuente: Elaboración propia.

En esta vista Figura 8, se da de alta, baja, editar los productos, así como el precio, y sub productos.



Figura 8. Vista de agregar o modificar productos.
Fuente: Elaboración propia.

En esta vista Figura 9, tenemos el historial de ventas por fechas.



Figura 9. Vista de Historial de Ventas.
Fuente: Elaboración propia.

En la vista Figura 10, se muestra el menú principal para el comensal donde puede explorar los productos ofrecidos por el restaurante.

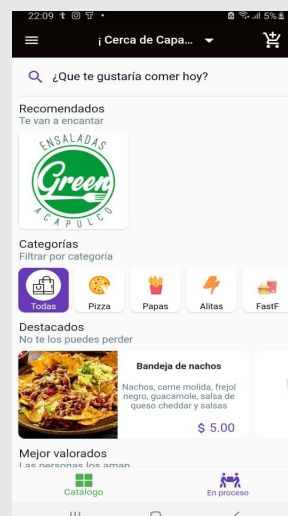


Figura 10. Vista del menú principal del restaurante.
Fuente: Elaboración propia.

En la vista Figura 11, se muestra la variedad de productos que ofrece el restaurante en el día, y el comensal puede escoger el que más se le antoje o sea de su agrado.

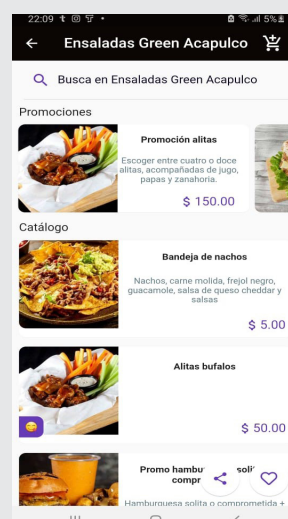
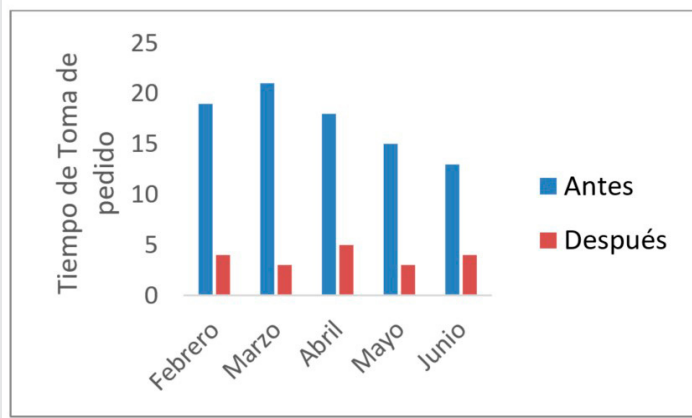


Figura 11. Vista del menú principal del restaurante.
Fuente: Elaboración propia.

Como se puede visualizar en las Figuras anteriores el levantamiento de un pedido es mucho mas facil e interactivo para el usuario en el levantamiento del pedido, ya que tiene el control de levantar el pedido, cancelarlo, modificarlo e incluso hacer una mala calificación ya sea sobre el producto o el restaurante, así como compartir el restaurante o su platillo favorito en las redes sociales como Facebook, Instagram, Whatsaap o incluso vía correo, para así poder promover sus ventas y aumentar su clientela, haciendo una comparativa en la Gráfica 1, se muestra como antes de la aplicación la toma de pedidos se le invierte un tiempo de 13 hasta 15 minutos vía telefónica y este lo plasman de forma manual en un formato que es hecho para esta actividad, en la cual también representó un 20% de errores al tomar el pedido al comensal, después de que se

instaló la aplicación y se puso en marcha, el tiempo de la toma de pedido se redujo a un margen de tiempo de 3 a 5 minutos, también el porcentaje de error a un 0.2%, esto provocó una mayor fluidez en la información del comensal al restaurante.



Gráfica 1. Resultados de tiempo en la toma de pedidos.
Fuente: Elaboración propia.

CONCLUSIONES

Gracias a la transformación digital que se experimentó en el establecimiento aumentó en su productividad, y se minimizaron los errores de recepción de pedidos y entrega.

En la implementación de la plataforma (App) se pudo automatizar los procesos de despacho en el cual beneficio al restaurante y el personal que labora haciendo más eficiente su trabajo y con mayor fluidez en la recepción de la información del consumidor. Con el marco de trabajo Scrum el cliente fue viendo la evolución de su aplicación de forma evolutiva y así entregar un producto eficiente con mínimos errores y satisfacción del cliente, en forma y en el tiempo establecido.

La plataforma (App) de entrega a domicilio, ayudó a mitigar el impacto negativo de la pandemia, además les permitió un crecimiento relativo.

Finalmente, a pesar de este crecimiento relativo en el restaurante, es importante tener en cuenta que la pandemia sí provocó cierres de restaurantes y pérdidas de empleos en otras partes de la comunidad, pero principalmente en aquellos que no tuvieron alguna plataforma de entrega a domicilio.

AGRADECIMIENTOS

Agradezco todo el apoyo, disposición y tiempo de mi profesor, Juan Miguel Hernández Bravo, así como también del CONACYT, quienes fueron de vital importancia y ayuda en el desarrollo de mi trabajo de investigación, también al Instituto Tecnológico de Acapulco, por apoyarme y poner a mi disposición todo el material necesario y sus instalaciones para mi desarrollo profesional e intelectual.

BIBLIOGRAFÍA

[1] R. S. Pressman, *Ingeniería del software, UN ENFOQUE PRÁCTICO*, Mexico: The McGraw-Hill INTERNATIONAL EDITORES, S.A. DE C.V., 2010.

[2] M. Cohn, *User Stories Applied: For Agile Software Development*, US: Pearson Education, Inc, 2004.

[3] A. Menzinsky, G. López and J. Palacio, *Historias de Usuario Ingeniería de requisitos ágil v. 2.0*, Scrum Manager., 2018.

[4] J. Sevilla, "Priorización de requisitos - Técnica MoSCoW.," 22 Marzo 2016. [Online]. Available: <https://www.overti.es/tecnologia/275-priorizacion-de-requisitos-tecnica-moscow>.