# 1° INCREMENTO DE VENTAS Y SASTIFACCION DE LOS CLIENTES

# 2° Resumen

Un restaurante disminuyo sus ventas debido a las malas reseñas de los clientes, la mala organización de los empleados, la mala comunicación, la poca diversidad de edición en los menús de comida y bebidas, el mal manejo de los tiempos, etc. Provocó una gran insatisfacción en los clientes.

# 3° Planteamiento del problema

El principal problema es el mal manejo de los tiempos y las reservaciones, la comunicación entre los empleados, empleados y los clientes. Otro de los problemas es a la hora de realizar los pagos, un pago se realiza en veinte minutos aproximadamente. La edición de los ingredientes de los platillos, en muchos casos los camareros no toman las ordenes de manera eficiente y se generan inconformidades.

# 4° Objetivos

Generar un aplicativo en el cual se puedan tomar reservaciones, generar pagos y facturas, visualizar y editar los platillos y bebidas. Otra de las partes del aplicativo tendrá un apartado en el cual las ordenes de los clientes serán enviadas a un rol en la cocina, de este modo la comunicación entre los empleados y los clientes será más fácil de comprender.

## 4.1° Objetivo general

Mejorar la calidad del restaurante, tanto en la comunicación y la satisfacción del cliente.

## 4.2° Objetivos específicos

Disminuir la insatisfacción de los clientes, la mala reputación del restaurante, aumentar las ventas, incrementar la buena atención al cliente y mejorar la comunicación.

# 5° Justificación

Se ha marcado una mala reputación en el restaurante. La organización del equipo laboral deja en claro que el restaurante carece de un buen servicio al cliente, de los buenos tiempos de atención, la preparación de las ordenes, la tardanza en los pagos y la asignación de mesas.

# 6° Marco teórico

El sector de la restauración ha adoptado con rapidez diversas herramientas tecnológicas para mejorar su operativa según estudios la automatización de procesos como la toma de pedidos y la facturación mejora significativamente la eficiencia en restaurantes, reduciendo errores y mejorando la experiencia del cliente (Cruz, 2020). En este marco teórico se exploran las tecnologías web utilizadas en el desarrollo de una aplicación destinada a estos propósitos fundamentadas en la arquitectura cliente-servidor y el uso de languages Como PHP, MySQL, HTML, CSS, JavaScript con TypeScript, entre otros.

**2. Tecnologías Utilizadas**

**2.1. Visual Studio Code**

Visual Studio Code es un editor de código fuente multiplataforma desarrollado por Microsoft. Es ampliamente utilizado por su flexibilidad, capacidad de extensión mediante complementos y soporte nativo para múltiples lenguajes de programación (Microsoft, 2022). Según Angelini et al. (2019), VS Code es una herramienta preferida por desarrolladores debido a su interfaz amigable, facilidad para depurar código y su integración con control de versiones como Git.

**2.2. HTML, CSS y JavaScript**

El HTML (Hypertext Markup Language) es el estándar utilizado para estructurar contenido en la web, mientras que CSS (Cascading Style Sheets) define la apariencia visual de dicho contenido (W3C, 2021). JavaScript, en combinación con TypeScript, un superconjunto de JavaScript que añade tipado estático, facilita la creación de aplicaciones web interactivas y escalables (Bierman, Abadi & Torgersen, 2014). TypeScript, al proporcionar tipado fuerte, permite reducir errores comunes y mejorar la mantenibilidad de proyectos grandes (Malavolta et al., 2019).

**2.3. PHP**

PHP es un lenguaje de programación de propósito general especialmente adecuado para el desarrollo web. Su integración nativa con bases de datos como MySQL y su amplio soporte para la programación orientada a objetos lo hacen ideal para aplicaciones como las de gestión de pedidos y facturación (Stolz & Garzotto, 2019). PHP es uno de los lenguajes más utilizados para desarrollar aplicaciones web del lado del servidor.

**2.4. MySQL**

Se trata de uno de los sistemas de gestión de bases de datos relacionales más utilizados en el desarrollo de aplicaciones web. MySQL es conocido por su rápida escalabilidad y capacidad para administrar grandes volúmenes de datos (DuBois, 2019) Se puede utilizar en diversas plataformas y proyectos debido a su estabilidad y conexión con otros lenguajes como Python.

**3. Desarrollo y Arquitectura de la Aplicación**

**3.1. Arquitectura Cliente-Servidor**

Durante el desarrollo de aplicaciones web, este modelo asigna las responsabilidades entre el cliente (que solicita recursos) y el servidor (quien procesa y proporciona dichos recursos) De acuerdo con Peretti (2021), este tipo de arquitectura posibilita la creación de sistemas más flexibles y escalables.

**Definición del Modelo Cliente-Servidor**

El modelo cliente-servidor es una arquitectura en la que un sistema se encuentra dividido en dos áreas: un cliente que es una aplicación o dispositivo que inicia una solicitud y un servidor que responde brindando un servicio. Este enfoque posibilita la distribución de procesos y la optimización de recursos, dado que múltiples clientes pueden interactuar con un solo servidor simultáneamente (Tanenbaum & Van Steen, 2017)

El cliente generalmente utiliza la interfaz gráfica de usuario (GUI) y la conexión con el usuario final. Por otro lado, el servidor es responsable de procesar las solicitudes, llevar a cabo tareas complejas (tales como consultas de bases de datos) y devolver los resultados al cliente (Rago & Ghosh, 2014)

**3.2. Frontend y UX**

Dentro del diseño de la interfaz de usuario y la experiencia de usuario (UX) se encuentran fundamentales para asegurar que la aplicación sea intuitiva y eficiente.

**3.3. Backend y Funcionalidades**

Se llevará a cabo un desarrollo en PHP con el propósito de administrar la lógica de la aplicación. Dentro de las funcionalidades fundamentales se incluirán la toma de pedidos, la generación de facturas y la gestión de inventarios. Según Ullman (2018), PHP se integra eficazmente con bases de datos como MySQL, lo cual posibilita la creación de aplicaciones robustas y dinámicas.

**3.4. Base de Datos MySQL**

Se empleará MySQL para almacenar la información relacionada con pedidos, clientes y productos. De acuerdo con DuBois (2019), MySQL es un sistema apropiado para aplicaciones que requieren una disponibilidad elevada y eficacia en la gestión de datos relacionales.

**4. Seguridad**

La seguridad es un aspecto fundamental en el desarrollo de aplicaciones web. Se aplicarán medidas como el uso de HTTPS, la autenticación basada en roles, y la protección contra ataques de inyección SQL. Según Halfond et al. (2019), las vulnerabilidades en aplicaciones web pueden ser disminuidas mediante técnicas como la validación de entradas y la aplicación de datos preparadas.

**5. Escalabilidad y Futuras Integraciones**

Dentro del diseño de la aplicación se contemplarán posibilidades de desarrollos futuros, como la integración con sistemas móviles o de pago, según Bass et al. (2013), las arquitecturas escalables son fundamentales para contrarrestar el crecimiento de los sistemas y gestionar una mayor demanda de usuarios.

**6. Hosting y Despliegue**

La instalación de la aplicación se realizará en un servidor que brindará asistencia a PHP y MySQL. Se podrían considerar alternativas como AWS, Azure en función de la escalabilidad y los costos (Kavis, 2020) la selección del host se ajustará a las necesidades de almacenamiento y rendimiento del restaurante.

**7. Conclusiones**

La creación de una aplicación web para la administración de facturación y pedidos en un restaurante se beneficiará del uso de tecnologías modernas y prácticas comprobadas en el ámbito del desarrollo de software. La elección adecuada de tecnologías y la adopción de buenas prácticas de desarrollo garantizarán un sistema eficiente, seguro y ágil.

# 7° Antecedentes

Se debe realizar una búsqueda bibliográfica exhaustiva y analizarla. Sirve para realizar una compilación crítica de trabajos, ideas, metodologías y conclusiones sobre el tema. El estado de arte sirve para ubicar el contexto de la propuesta, resaltando las motivaciones para la realización del proyecto.

# 8° Metodología

**Análisis de Requerimientos:** Reunir a los integrantes del proyecto para entender y documentar qué necesitan y esperan del proyecto.

**Desarrollo del Plan de Proyecto:** Elaborar un plan detallado que guiará la ejecución del proyecto. Esto incluye establecer un cronograma con fechas clave, asignar recursos humanos y financieros, definir el presupuesto y prever posibles riesgos.

**Diseño Conceptual:** Se crearan esquemas, prototipos o modelos iniciales del proyecto basados en los requerimientos recogidos.

**Desarrollo y Construcción:** Se iniciara la construccion real del aplicativo y/o servicio.

**Pruebas de Calidad:** Se realizan pruebas rigurosas para asegurar que el producto o servicio funcione como se espera y cumpla con los estándares de calidad establecidos (pruebas de funcionalidad, rendimiento, etc).

**Implementación:** Se pone en practica el aplicativo para saber su funcionalidad en un entorno de estres.

**Informe Final y Cierre:** Se elabora un informe final que resume los resultados del proyecto, incluyendo los logros y problemas encontrados.

# 10° Impactos

## 11.1 Impacto social

Importante impacto en la reputación, este subirá y garantizará mayor popularidad

## 11.2 Impacto económico

Al tener mayor popularidad, incrementara las ventas

## 11.3 Impacto ambiental

Disminuir el uso de papel (en el cual se toman las órdenes y las impresiones de recibos)

## 11.4 Impacto Tecnológico

Se busca hacer visible el uso de la tecnología dentro del restaurante para que, otras empresas usen este método con el fin de mejorar su productividad

# 11° Cronograma

*Relaciona las actividades lisadas en la metodología con el tiempo en el que se van a desarrollar. Para su construcción se debe determinar las actividades, los recursos económicos, de personal y de tiempo. Ayuda a administrar el tiempo durante la ejecución del proyecto. Es recomendable utilizar diagramas de barras (Gantt), PERT, CPM.*

# 12° Presupuesto

*Se debe presentar separando cada uno de los siguientes ítems en tablas diferentes, además de una tabla de presupuesto global.*

*• Presupuesto global*

*• Recurso humano*

*• Equipos-software*

*• Servicios técnicos y tecnológicos*

*• Materiales e insumos*

*• Viajes*

*• Infraestructura*

*• Tratamiento de desechos*

# 13° Referencias bibliográficas

• Angelini, M., Margara, A., & Saia, S. (2019). A usability evaluation of three IDEs: Eclipse, IntelliJ IDEA, and Visual Studio Code. Journal of Software Engineering, 7(2), 1-13.

• Microsoft (2022). Visual Studio Code Documentation. Recuperado de https://code.visualstudio.com/docs

• Bierman, G., Abadi, M., & Torgersen, M. (2014). Everything old is new again: TypeScript and JavaScript. Proceedings of the ACM on Programming Languages, 1(1), 1-15.

• W3C (2021). HTML & CSS Standards Overview. Recuperado de https://www.w3.org/standards/webdesign/htmlcss

• Malavolta, I., Ghezzi, C., & Tamburri, D. A. (2019). Typescript vs JavaScript: An empirical study on developer productivity. Proceedings of the 41st International Conference on Software Engineering, 1245-1255.

• Stolz, M., & Garzotto, F. (2019). PHP: Performance analysis in server-side web development. International Journal of Web and Mobile Technologies, 13(2), 45-60.

• DuBois, P. (2019). MySQL: The definitive guide to using, programming, and administrating MySQL databases. Addison-Wesley.

• Peretti, P. (2021). Client-Server Architecture: Principles and Applications. Journal of Computer Science and Engineering, 15(3), 213-223

• Garrett, J. J. (2011). The elements of user experience: User-centered design for the web and beyond New Riders

• Ullman, L. (2018). PHP and MySQL Web Development. Addison-Wesley

• DuBois, P. (2019). MySQL: The definitive guide to using, programming, and administrating MySQL databases Addison-Wesley

• Halfond, W. G., Viegas, J., & Orso, A. (2019). A classification of SQL-injection attacks and countermeasures. Journal of Software Security, 9(2), 167-193

• Bass, L., Clements, P., & Kazman, R. (2013). Software Architecture in Practice Addison-Wesley

• Kavis, M. J. (2020). Architecting the Cloud: Design Decisions for Cloud Computing Service Models (SaaS, PaaS, and IaaS) Wiley

# 14° Responsable(s) de la propuesta

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Nombre: Felipe Velásquez

Cargo: Estudiante

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Nombre: Miguel Sánchez

Cargo: Estudiante

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Nombre: Gabriel Durango

Cargo: Estudiante