| **Información General del Proyecto** | | | | |
| --- | --- | --- | --- | --- |
| **ID. Proyecto** | | BAZZA-ML-001 | **Fecha** | 16/11/2023 |
| **Nombre del Proyecto** | | MESALISTA | | |
| **Director de Proyecto** | | Richard Romero | | |
| **Desarrollador Backend** | | Sergio Galaz | | |
| **Desarrollador Frontend** | | Tyler Miranda | | |
| **Presupuesto** | | $30.000.000 | | |
| **Horas** | | 700 HH | | |

| **Introducción** | | | | | |
| --- | --- | --- | --- | --- | --- |
| **Objetivo General:**  Desarrollar e implementar un sistema web de gestión de pedidos y servicios en restaurantes que optimice la eficiencia operativa, mejore la experiencia del cliente presencial al permitirle realizar pedidos de forma autónoma y en tiempo real, y garantice un flujo de información claro y eficiente entre las mesas, la cocina y el personal, con un enfoque en reducir errores, tiempos de gestión y aumentar la satisfacción del cliente | | | | | |

| **Introducción** | | | | | |
| --- | --- | --- | --- | --- | --- |
| **Objetivos específicos:**   * Diseñar e implementar una aplicación web para dispositivos en mesa que permita a los clientes realizar pedidos de forma autónoma, garantizando un tiempo de respuesta promedio inferior a 2 segundos. * Garantizar el rendimiento del sistema para soportar un mínimo de 30 mesas activas simultáneamente, manteniendo un tiempo de respuesta promedio inferior a 3 segundos bajo condiciones de carga. * Desarrollar una interfaz intuitiva para la cocina que muestre los pedidos en tiempo real, permitiendo a los cocineros gestionar el inicio y fin de la preparación de cada plato con un máximo de 3 clics, reduciendo en un 30% el tiempo promedio de gestión de pedidos. * Implementar un sistema de pedidos digitales que elimine los errores asociados con la toma manual de pedidos, aumentando la precisión de los pedidos en un 25%. * Reducir en un 40% el tiempo que los meseros dedican a tomar pedidos, optimizando su capacidad para atender un 30% más de mesas por turno mediante el uso del sistema de autoservicio digital. * Diseñar un sistema de feedback integrado que permita medir y mejorar la satisfacción del cliente en un 35%, basado en encuestas post-servicio y retroalimentación directa. * Implementar herramientas analíticas para evaluar en tiempo real el desempeño del sistema y garantizar el cumplimiento de los indicadores clave de rendimiento (KPIs) definidos. | | | | | |

| **Casos de uso** | | | | | |
| --- | --- | --- | --- | --- | --- |
| Para garantizar que la arquitectura propuesta cumpla con los objetivos generales y específicos del sistema, es esencial identificar y analizar los casos de uso clave que representan las interacciones principales entre los usuarios y el sistema. Estos casos de uso no solo reflejan los requisitos funcionales, sino que también guían las decisiones arquitectónicas al definir los flujos de información, las dependencias entre componentes y las necesidades de integración. A continuación, se presenta una vista general de los casos de uso **claves**: | | | | | |
| **Proceso de pedido**  El proceso de pedido permite que los clientes realicen sus solicitudes directamente desde un dispositivo en la mesa o utilizando su teléfono móvil al escanear un código QR. Este código genera una sesión única que los clientes pueden compartir con otros miembros de su grupo, permitiendo realizar pedidos de manera conjunta y sincronizada. Este flujo mejora la experiencia del usuario al eliminar la necesidad de interacción inicial con el personal, optimizando el tiempo y facilitando la coordinación en pedidos grupales. | | | | | |
| **Proceso de pago**  El proceso de pago permite que los clientes realicen el pago de sus pedidos de forma autónoma mediante un dispositivo, o soliciten la asistencia de un mesero para completar la transacción. Este flujo asegura la flexibilidad en la experiencia del usuario y permite una gestión eficiente de las órdenes. | | | | | |

| **Diagrama de secuencias** | | | | | |
| --- | --- | --- | --- | --- | --- |
| Los diagramas de secuencias son herramientas esenciales para representar el flujo de mensajes e interacciones entre los actores y componentes del sistema.. Estos diagramas nos permiten visualizar cómo se desarrollan los procesos clave dentro de Mesalista, ayudando a comprender las dependencias y la lógica detrás de cada acción. A continuación, se presentan los diagramas de secuencias para los procesos de pago y pedido, destacando las interacciones entre los actores y el sistema, de los procesos **claves**. | | | | | |
| **Diagrama de secuencias - Pedido**  El diagrama de secuencias del proceso de pedido ilustra las interacciones que ocurren cuando un cliente inicia una sesión de pedido escaneando un código QR desde un dispositivo en la mesa o propiamente del dispositivo en la mesa. Destaca cómo el sistema permite que otros usuarios se unan a la sesión compartida y coordinen pedidos de manera colaborativa, asegurando que todos los pedidos se registren correctamente en el sistema. | | | | | |
| **Diagrama de secuencias - Pago**  El diagrama de secuencias del proceso de pago describe cómo los clientes y los meseros interactúan con el sistema para completar una transacción. Incluye tanto el flujo de pago autónomo desde un dispositivo como el flujo asistido, mostrando el intercambio de mensajes necesario para cerrar la orden y garantizar la sincronización entre los componentes del sistema. | | | | | |

| **Diagrama de componentes** | | | | | |
| --- | --- | --- | --- | --- | --- |
| El diagrama de componentes del sistema representa la interacción entre los servicios del frontend, backend, paquetes compartidos y servicios externos. En este sistema, el frontend se divide en una interfaz de cliente y una de administración, ambas comunicándose con un backend que gestiona la lógica de negocio, rutas públicas y privadas, y la capa de acceso a datos. Los paquetes compartidos y los servicios externos, como bases de datos, almacenamiento en la nube y caché, complementan el sistema proporcionando funcionalidades esenciales que aseguran la eficiencia y escalabilidad de la solución. | | | | | |
|  | | | | | |

| **Diagrama de arquitectura** | | | | | |
| --- | --- | --- | --- | --- | --- |
| El siguiente diagrama representa la arquitectura del sistema, diseñada para garantizar alta disponibilidad, escalabilidad y eficiencia. La solución está implementada sobre Google Cloud Platform (GCP) utilizando servicios como Cloud Run para la ejecución de aplicaciones, conectores VPC para integraciones seguras y un flujo CI/CD basado en GitHub. La capa de cliente incluye navegadores web y dispositivos móviles, mientras que los servicios externos, como bases de datos, caché, almacenamiento y correo, complementan las capacidades del sistema. Esta arquitectura permite un flujo de trabajo optimizado, integrando observabilidad mediante Cloud Logging y manteniendo una gestión robusta de datos y servicios. | | | | | |
|  | | | | | |

| **Conclusión** | | | | | |
| --- | --- | --- | --- | --- | --- |
| El diseño y desarrollo de este sistema ha seguido un enfoque metódico, abordando cada componente clave desde la definición inicial hasta la implementación de la arquitectura final. A través de diagramas claros y detallados, como los de casos de uso, secuencias y componentes, hemos podido visualizar y documentar cada aspecto del flujo del sistema, desde el proceso de pedidos y pagos hasta la interacción con servicios externos.  La arquitectura final presentada reafirma la importancia de construir sistemas modernos y escalables, aprovechando las capacidades de Google Cloud Platform. El uso de Cloud Run, conectores VPC y un flujo CI/CD con GitHub asegura una infraestructura flexible, segura y altamente automatizada. Además, la integración de servicios externos como Neon Tech DB, Upstash Redis y Cloudinary optimiza el rendimiento, mientras que herramientas como Cloud Logging permiten una observabilidad robusta para la gestión y monitoreo continuo.  Este diseño no solo satisface los requerimientos funcionales, como la gestión de pedidos y pagos de forma eficiente, sino que también asegura que el sistema pueda escalar y adaptarse a futuros desafíos. En conjunto, la solución refleja una combinación de simplicidad, eficacia y visión estratégica, posicionándose como una arquitectura sólida y confiable para ofrecer una experiencia excepcional a los usuarios finales. | | | | | |