

---

# **Perfil de proyecto: System Request y Requisitos Iniciales**

**Universidad Veracruzana  
Desarrollo de Software**

**Equipo 1**

**Alejo Barrientos Ruben Isai  
Calderón Blas Javier Alberto  
Camarillo Villa Jóse Daniel  
Dominguez García Antonio de Jesús**

## System Request - Sistema para “Italia Pizza”

<b>Patrocinador del proyecto</b>	Pizzería “Italia Pizza”.
<b>Necesidades del negocio</b>	<p>Se requiere la implementación de un sistema de software que agilice los procesos de administración y gestión de la pizzería, permitiendo un mejor control en las operaciones de productos, preparación de alimentos, pedidos y finanzas. Así como también, la incorporación de un módulo de usuarios para contemplar un mejor monitoreo hacia los clientes y empleados.</p>
<b>Funcionalidades del negocio requeridas</b>	<p>Para el personal y clientes de la pizzería “Italia Pizza” quienes necesitan agilizar las operaciones de productos, usuarios y catálogos generales (preparación de alimentos, pedidos y finanzas), el sistema deberá contar con:</p> <ul style="list-style-type: none"> <li>● Administración de usuarios (leer, registrar, editar, eliminar, buscar), así como el inicio de sesión por parte de los empleados de la pizzería.</li> <li>● Administración de productos (leer, registrar, editar, eliminar, buscar), así como la generación de reportes de inventario con los productos disponibles.</li> <li>● Administración de pedidos (leer, registrar, editar) por parte de los clientes.</li> <li>● Administración de preparación de alimentos (leer, editar) de los pedidos, así como la administración de recetas (Ver, registrar, editar, eliminar) de cada platillo que se deberá realizar.</li> <li>● Administración de finanzas abarcando el registro de los cobros de caja y pedidos de domicilio, así como el registro de salidas abarcando los pagos a proveedores, compras de insumos y otros gastos.</li> <li>● Administración del balance diario pudiendo hacer el corte de caja.</li> <li>● Administración de los proveedores de la pizzería (ver, registrar, editar, eliminar) incluyendo la gestión de los pedidos por producto realizados a cada proveedor.</li> </ul>

<b>Valor generado para el negocio</b>	<p>Su funcionamiento está destinado para el manejo y almacenamiento de información relacionada a los productos, personal, clientes y pedidos de la pizzería “Italia Pizza”, así como las finanzas relacionadas a la compra venta de los productos de la pizzería.</p> <p>Este producto de software será desarrollado con tecnologías actuales y debe de estar optimizado de manera en la que evite guardar información no relevante, así como un fácil manejo del sistema por parte del personal.</p> <p>Dentro de las estimaciones que se esperan conseguir como beneficio para la pizzería se incluyen:</p> <ul style="list-style-type: none"> <li>● Optimizar el control y el seguimiento de los pedidos.</li> <li>● Mejorar la administración del inventario con respecto a los productos que se tengan en existencia.</li> <li>● Disminuir el tiempo en el que se llevan a cabo los pedidos tanto en su registro como en su preparación.</li> <li>● Disminuir el tiempo en el que se lleva a cabo la administración financiera.</li> <li>● Obtener un mejor seguimiento de los clientes de la pizzería con respecto a sus pedidos.</li> </ul>
<b>Justificación</b>	<p>El software ofrece una gran oportunidad para el desarrollo y el avance de la pizzería “Italia Pizza”. Esta tecnología permitirá la interacción correspondiente con el actor involucrado, el procesamiento y almacenamiento de datos.</p> <p>En la actualidad, es necesario que la información sea procesada y almacenada de forma efectiva para agilizar los procesos tales como la administración y gestión de la pizzería siendo un punto clave para las operaciones y así lograr un control integral de estas actividades.</p> <p>El diseño y desarrollo del software que sirva de apoyo al proceso de operaciones de la pizzería “Italia Pizza”, cubrirá las necesidades de este, proporcionará efectividad y eficiencia en el manejo del flujo y procesamiento de información que se requieren este tipo de procesos, ejecutando las actividades con el menor esfuerzo humano y en el menor tiempo posible.</p>
<b>Restricciones</b>	<ul style="list-style-type: none"> <li>● El software contará con un usuario-administrador quien será el único capaz de dar registro a un empleado.</li> </ul>

	<ul style="list-style-type: none"> <li>● El software sólo estará disponible siempre y cuando se encuentre instalado para ejecutarlo.</li> <li>● Estará disponible únicamente para sistemas operativos Windows.</li> <li>● El software será implementado con el Servidor de base de datos SQL Server para la capa de datos y con WPF (Windows Presentation Foundation) para la capa de aplicación.</li> <li>● El software será desarrollado bajo un arquitectura cliente-servidor y será de instalación local.</li> </ul>
--	--

## Requerimientos Iniciales

### Requisitos Funcionales

Los siguientes requisitos funcionales fueron escritos bajo la notación EARS (Easy Approach to Requirements Syntax) con el propósito de atacar la falta de formación sobre su redacción, ambigüedad, consistencia y coherencia.

Módulo de Usuarios	
Número	Requisito
RF-01	El Sistema debe registrar a los nuevos usuarios de la pizzería según sean clientes o empleados.
RF-02	Cuando se registre un empleado de la pizzería el Sistema debe guardar un nombre de usuario y contraseña para entrar al sistema de forma autenticada.
RF-03	El Sistema debe actualizar la información de los usuarios registrados (clientes y empleados).
RF-04	El Sistema debe eliminar la información de los usuarios (clientes empleados).
RF-05	Si la cuenta de un empleado tiene una sesión activa, entonces el Sistema no deberá permitir su eliminación.
RF-06	Si la cuenta de un cliente ya ha realizado pedidos, entonces el Sistema no deberá permitir su eliminación.
RF-07	El Sistema debe buscar a los clientes registrados de acuerdo a su nombre, dirección o teléfono.
RF-08	Cuando se busque un cliente por su nombre o dirección, el Sistema debe encontrar la información sin importar si son mayúsculas o minúsculas.

Módulo de Productos	
Número	Requisito
RF-09	El Sistema debe registrar un nuevo producto.
RF-10	El Sistema debe actualizar la información de los productos registrados a excepción de su código.
RF-11	El Sistema debe eliminar la información de los productos.
RF-12	Si el producto ya se ha utilizado en algún pedido, entonces el Sistema no deberá permitir su eliminación.
RF-13	El Sistema debe buscar los productos registrados de acuerdo a su nombre o código.
RF-14	Cuando se busque un producto por su nombre, el Sistema debe encontrar la información sin importar si son mayúsculas o minúsculas o sólo una parte del mismo.
RF-15	El Sistema debe generar reportes en archivos PDF del inventario de los productos que se tengan en existencia (disponibles).
RF-16	El Sistema debe permitir comparar la cantidad de cada uno de los productos que se tienen en existencia con la cantidad que se tiene registrada en el inventario.
RF-17	Cuando se compare la cantidad de cada uno de los productos que se tienen en existencia con la cantidad que se tiene registrada en el inventario, el Sistema debe verificar si existen faltantes, sobrante o si la cantidad está correcta.

Módulo de Pedidos	
Número	Requisito
RF-18	El Sistema debe permitir registrar un nuevo pedido solicitado por un cliente.
RF-19	Mientras se esté registrando un pedido, el Sistema debe permitir agregar más de un producto del mismo tipo o de otros.
RF-20	El Sistema debe calcular el total a pagar a partir de los productos y cantidad solicitadas en el pedido.
RF-21	El Sistema debe permitir únicamente actualizar la cantidad de productos y estatus de un pedido.
RF-22	El Sistema debe buscar pedidos por usuario, fecha o estatus.

Módulo de Cocina	
Número	Requisito
RF-23	El Sistema debe mostrar una lista de todos los pedidos a preparar.
RF-24	El Sistema debe administrar (leer, registrar, actualizar y eliminar) las recetas de cada platillo a realizar.
RF-25	El Sistema debe permitir marcar en la lista los pedidos que ya han sido preparados.

Módulo de Finanzas	
Número	Requisito
RF-26	El Sistema debe registrar las entradas (cobros de caja y pedidos a domicilio).
RF-27	El Sistema debe registrar las salidas (pagos a proveedores, compras de insumos y gastos varios).
RF-28	El Sistema debe realizar un balance diario (corte diario de caja).
RF-29	El Sistema debe administrar (leer, registrar, actualizar y eliminar) a los proveedores de la pizzería.
RF-30	El Sistema debe administrar (leer, registrar, actualizar y eliminar) los pedidos de productos.

## Requisitos No Funcionales

Atributo de Calidad	Definición	Descripción
Disponibilidad	Se refiere a la capacidad de un sistema para enmascarar o reparar fallos de manera que el período de interrupción del servicio acumulado no supere un valor requerido en un intervalo de tiempo especificado.	<p>DISP-01: El Sistema debe estar disponible el 99% del tiempo durante los próximos 3 meses posteriores a su despliegue.</p> <p>DISP-02: El Sistema debe estar disponible tras 1 hora en caso de presentar una falla.</p> <p>DISP-03: El Sistema no debe presentar más de 5 fallas graves por semana.</p>

<b>Seguridad</b>	La seguridad es una medida de la capacidad del sistema para proteger los datos y la información del acceso no autorizado mientras se proporciona acceso a personas y sistemas que están autorizados.	SEG-01: El Sistema debe contar con autenticación a través de un usuario y contraseña para poder ingresar.  SEG-02: El Sistema debe cifrar los datos sensibles que sean almacenados.
<b>Usabilidad</b>	Grado en el que software protege la información y los datos para que las personas u otros productos o sistemas tengan el grado de acceso a los datos adecuado a sus tipos y niveles de autorización.	USAB-01: El Sistema debe aumentar la confianza y satisfacción de los usuarios.  USAB-02: El RumMichy debe minimizar el impacto de los errores en las actividades del usuario.
<b>Mantenibilidad</b>	Representa el grado de efectividad y eficiencia al que los desarrolladores pueden modificar el software para mejorarlo, corregirlo o adaptarlo a cambios en el entorno y/o requisitos.	MANT-01: El Sistema en su versión final de código deberá estar documentado.  MANT-02: El Sistema debe implementar patrones arquitectónicos y de diseño.
<b>Robustez</b>	Capacidad para manejar condiciones de error mientras se ejecuta si la conexión a Internet se cae o si hay un corte de energía o fallo de hardware.	ROB-01: El Sistema debe mantener la información introducida por el usuario en los campos, en caso de haber estado llenando algún formulario del sistema, tras un cierre inesperado por fallos del dispositivo de cómputo o conexión a internet.  ROB-02: El porcentaje de eventos del Sistema que provoquen fallas en el sistema debe de ser menor al 10%.

## **Referencias**

Bass, L., Clements, P., & Kazman, R. (2003). *Software Architecture in Practice*. Amsterdam University Press.

Gomaa, H. (2011). *Software Modeling and Design: Uml, Use Cases, Patterns, and Software Architectures* (Illustrated ed.). Cambridge University Press.