Informe de Selección de Metodología de Desarrollo de Software

Proyecto: Sistema de Gestión de Pedidos para Restaurantes  
Elaborado por: [Tu Nombre]  
Fecha: [Fecha de entrega]

# Introducción

El presente documento tiene como objetivo describir y justificar la metodología de desarrollo de software seleccionada para el proyecto **Sistema de Gestión de Pedidos para Restaurantes**. El sistema propuesto busca optimizar el proceso de toma de pedidos, gestión de cocina, control de caja y generación de reportes administrativos en restaurantes de tamaño pequeño y mediano.

La correcta elección de la metodología es un factor crítico de éxito, pues define la forma en que se gestionarán los requisitos, la comunicación con el cliente, la frecuencia de entregas y el control de calidad durante todo el ciclo de vida del software.

# Contexto y Características del Proyecto

El proyecto se centra en el diseño y desarrollo de un sistema multiplataforma accesible desde dispositivos móviles y tabletas, que permita:

* La toma de pedidos digital por parte de los meseros.
* La visualización de pedidos en cocina para priorizar su preparación.
* La facturación y cobro rápido en caja.
* La generación de reportes de ventas e indicadores de gestión para el administrador.

Se trata de un sistema que debe ser **fácil de usar**, **rápido**, **confiable** y que pueda **escalar** en el futuro para integrar funcionalidades adicionales como reservas de mesas, pedidos a domicilio y gestión de clientes frecuentes.

## ****Filtros para la Selección de la Metodología****

Para elegir la metodología de desarrollo más adecuada, se tuvieron en cuenta los siguientes **criterios de selección**:

### ****1. Tamaño del Proyecto****

El alcance del proyecto es de tamaño **pequeño a mediano**, ya que se implementará para un solo restaurante inicialmente, aunque con posibilidad de expansión futura a varias sucursales. Esto implica que se requiere una metodología que permita entregar resultados funcionales en ciclos cortos, priorizando los requisitos más importantes.

**Justificación:** Metodologías ágiles, como Scrum, permiten manejar proyectos de este tamaño de manera eficiente, dividiendo el trabajo en sprints de 2-4 semanas, lo que garantiza avances visibles para el cliente.

### ****2. Complejidad Técnica y Estado de la Tecnología****

El sistema será desarrollado con tecnologías web modernas y debe ser accesible desde navegadores y dispositivos móviles. No se trata de un proyecto de investigación de alta incertidumbre técnica, sino de un sistema con componentes bien conocidos (frontend, backend, base de datos).

**Justificación:** Esto permite adoptar un enfoque ágil sin grandes riesgos técnicos, aprovechando buenas prácticas de desarrollo incremental y pruebas continuas.

### ****3. Periodicidad de Retroalimentación con el Cliente****

El cliente requiere **revisiones frecuentes** para validar que el sistema cumple sus expectativas, especialmente en módulos sensibles como pedidos y facturación.

**Justificación:** Scrum facilita reuniones periódicas (revisiones de sprint y retrospectivas), donde el cliente puede ver avances tangibles, aportar retroalimentación y ajustar prioridades si es necesario.

### ****4. Flexibilidad y Cambios en los Requisitos****

Durante el levantamiento de requisitos se identificó que podrían surgir **cambios o nuevas necesidades**, por ejemplo, integrar pedidos a domicilio o estadísticas más avanzadas en el futuro.

**Justificación:** Metodologías ágiles permiten priorizar y replanificar de forma continua, adaptando el alcance de acuerdo con la evolución de las necesidades del negocio.

### ****5. Recursos Disponibles y Colaboración del Equipo****

El equipo de desarrollo es pequeño y multidisciplinario (desarrollador, diseñador, analista), lo que facilita la colaboración en ciclos cortos. Además, el cliente está disponible para reuniones periódicas.

**Justificación:** Un enfoque ágil permite aprovechar al máximo esta cercanía y colaboración para evitar retrabajos y garantizar que el producto sea el que realmente necesita el negocio.

### ****6. Gestión de Riesgos y Calidad****

El sistema debe garantizar la confiabilidad de la información y evitar errores en pedidos y cobros.

**Justificación:** Scrum, al ser iterativo e incremental, permite realizar pruebas en cada sprint, identificando fallos de manera temprana y reduciendo el riesgo de fallas en producción.

## Metodología Seleccionada: Scrum

Scrum es un marco de trabajo ágil basado en iteraciones llamadas **sprints,** en las que se entregan incrementos funcionales del producto. Esta metodología fomenta la colaboración continua entre el equipo de desarrollo y el cliente, asegurando que las funcionalidades se construyan conforme a las prioridades de negocio.

Tras analizar los criterios anteriores, se selecciona **Scrum** como la metodología de desarrollo de software para este proyecto, ya que:

* Permite trabajar en **iteraciones cortas** (sprints).
* Involucra activamente al **cliente** en revisiones periódicas.
* Facilita la **priorización de requisitos** y la adaptación a cambios.
* Promueve la **entrega continua de valor** y la mejora constante.

# Conclusiones

La selección de Scrum como metodología de desarrollo es la más adecuada para el Sistema de Gestión de Pedidos para Restaurantes, ya que permite entregar resultados de forma incremental, involucrar al cliente en el proceso, reaccionar ante cambios de requisitos y asegurar la calidad del producto mediante revisiones constantes.

## ****Bibliografía****

* Sommerville, I. (2011). Ingeniería de Software (9a ed.). Pearson.
* Pressman, R. S. (2015). Ingeniería de Software: Un enfoque práctico (7a ed.). McGraw-Hill.
* Schwaber, K., & Sutherland, J. (2020). La Guía de Scrum: La Guía Definitiva de Scrum: Las Reglas del Juego. Recuperado de https://scrumguides.org/
* IEEE Std 830-1998. IEEE Recommended Practice for Software Requirements Specifications.