***Informe de Avance del Proyecto Software de Comandas, Mes Setiembre***

**Resumen Ejecutivo**

El proyecto de software de comandas ha logrado avances significativos hasta la fecha, comenzando con la definición del problema central del restaurante, donde se identificaron ineficiencias en la gestión de comandas y la comunicación entre el personal. Se llevó a cabo un diagnóstico detallado de los procesos involucrados, estableciendo una base sólida para el desarrollo del proyecto. Además, se diseñó un cronograma estructurado que guía la presentación de avances y permite planificar los hitos clave. A partir de este cronograma, se determinó el presupuesto necesario y se realizó un análisis de costo-beneficio para evaluar la viabilidad del proyecto. Hasta el momento, el proyecto se encuentra alineado con los objetivos establecidos y dentro del presupuesto planificado.

**Estado del cronograma**

El cronograma del proyecto de software de comandas está diseñado para completarse en un plazo establecido con fases definidas que incluyen: planificación, gestión de requisitos, seguimiento y control del proyecto, análisis, diseño, gestión de calidad de procesos y productos, gestión de la configuración del proyecto y gestión y acuerdos con los proveedores. Actualmente, el estado del cronograma refleja un progreso del 100% (proporción alcanzada hasta la fecha) con base en las actividades planificadas.

**Estado del presupuesto**

El presupuesto total del proyecto de software de comandas asciende a $50,650, dividido en tres componentes principales que cubren los recursos esenciales para su desarrollo y finalización. A continuación, se detallan los costos distribuidos por categoría:

1. Recursos de Personal:
   * Representa un costo de $20,700, destinado a la contratación y asignación del equipo de trabajo, incluyendo desarrolladores, analistas y personal de soporte técnico.
   * Este componente abarca los honorarios por las horas de trabajo, reuniones y revisiones realizadas durante el ciclo del proyecto.
2. Hardware:
   * Se asignaron $23,600 para la adquisición de equipos y dispositivos necesarios para el desarrollo, pruebas y puesta en marcha del sistema.
   * Este rubro incluye servidores, estaciones de trabajo y otros equipos requeridos para asegurar la calidad del producto final.
3. Software y Licencias:
   * Un monto de $6,350 fue destinado a la adquisición de licencias de software, herramientas de desarrollo y plataformas necesarias para la implementación del proyecto.
   * Este costo permitió el uso de entornos de programación y software de gestión que facilitaron las tareas durante todo el desarrollo.

**Recursos**

Durante el desarrollo del proyecto de software de comandas, se utilizaron recursos clave como personal especializado, infraestructura tecnológica y herramientas de software, todos ajustados al plan inicial sin variaciones significativas. Los recursos necesarios para completar el proyecto incluye el mantenimiento técnico del sistema y la capacitación de los usuarios finales. No se identificaron limitaciones relevantes en la disponibilidad de recursos durante el desarrollo, aunque a futuro podría ser necesario planificar adecuadamente el soporte técnico para garantizar la continuidad del sistema.

**Riesgos**

Durante el desarrollo del proyecto de software de comandas, se identificaron varios riesgos que podrían haber afectado el cumplimiento de los objetivos. Entre los principales riesgos se encuentra la renuncia de un colaborador, lo que podría generar retrasos en las entregas y la producción debido a la rotación de personal. También se identificó el riesgo de fallos en la base de datos, lo que podría resultar en la pérdida de información crítica y retrasos en la resolución de errores. Además, los cambios en los requisitos del cliente podrían incrementar tanto los tiempos como los costos del proyecto. La falta de pruebas adecuadas fue otro riesgo importante, ya que errores no detectados en etapas tempranas pueden ocasionar retrabajos y demoras en la implementación. Por otro lado, la baja conectividad entre dispositivos podría derivar en fallos en la funcionalidad y afectar directamente a los usuarios. También se consideraron los retrasos en la entrega de hardware o software por parte de proveedores, lo que podría causar atrasos en la integración del sistema. Finalmente, la ausencia de monitoreo constante del proyecto podría generar desviaciones en los cronogramas y presupuestos. Para mitigar estos riesgos, se implementaron estrategias de control y seguimiento, así como mecanismos de respuesta rápida para minimizar su impacto potencial.

**Problemas**

Durante el desarrollo del proyecto de software de comandas, surgieron diversos problemas significativos que fueron abordados de manera efectiva. Entre ellos se incluyó un error en la asignación de mesas, que fue resuelto mediante ajustes en la lógica del algoritmo. También se presentó duplicidad de pedidos en la barra, lo que se soluciona implementando un sistema de verificación antes de enviarlos. La lentitud en la carga de datos de cocina fue atendida mediante la optimización de las consultas a la base de datos. Además, se identificó un error en la visualización de los estados de los pedidos, corregido ajustando la interfaz de usuario. Otro desafío importante fue la falta de sincronización entre dispositivos, que se resolvió implementando un mecanismo de sincronización en tiempo real. Por último, se abordó la pérdida de pedidos no enviados mediante la creación de un sistema de recuperación de pedidos pendientes. Aunque todos estos problemas fueron solucionados, en el futuro podrían surgir nuevos inconvenientes menores que serán gestionados a través de un plan de mantenimiento técnico y mejoras continuas en el sistema.

**Plan de Acción**

1. Capacitación de usuarios finales:
   * Específico: Implementar sesiones de formación práctica para que los usuarios finales (mozos, cajeros, gerentes) comprendan el uso del sistema de comandas.
   * Medible: Realizar un mínimo de tres sesiones de capacitación con todos los roles clave.
   * Alcanzable: Utilizar al equipo de desarrollo para liderar las capacitaciones con ejemplos prácticos.
   * Relevante: Reducir errores operativos y garantizar la correcta implementación del sistema en el restaurante.
   * Tiempo: Completar las sesiones dentro del primer mes posterior a la entrega del sistema.
2. Mantenimiento técnico preventivo:
   * Específico: Establecer un plan de mantenimiento mensual para revisar y corregir posibles fallas técnicas en el sistema.
   * Medible: Realizar al menos una revisión técnica mensual durante los primeros seis meses.
   * Alcanzable: Asignar personal especializado para ejecutar estas revisiones.
   * Relevante: Prevenir interrupciones en el servicio y garantizar la estabilidad del sistema.
   * Tiempo: Iniciar el mantenimiento técnico inmediatamente después de la implementación.
3. Monitoreo y gestión de riesgos:
   * Específico: Implementar un sistema de monitoreo continuo para identificar y mitigar riesgos operativos y técnicos.
   * Medible: Documentar y evaluar al menos un informe mensual sobre posibles riesgos y su resolución.
   * Alcanzable: Utilizar herramientas de gestión de proyectos para el monitoreo.
   * Relevante: Garantizar la continuidad y eficiencia del sistema.
   * Tiempo: Realizar el monitoreo durante el primer año posterior a la entrega.
4. Optimización de la conectividad entre dispositivos:
   * Específico: Mejorar el sistema de sincronización en tiempo real entre dispositivos.
   * Medible: Lograr una reducción del 90% en problemas de conectividad reportados por los usuarios.
   * Alcanzable: Aplicar ajustes técnicos utilizando los recursos ya disponibles.
   * Relevante: Asegurar una experiencia fluida para los usuarios del sistema.
   * Tiempo: Finalizar las optimizaciones en un plazo máximo de dos semanas tras la identificación del problema.
5. Gestión de requisitos adicionales del cliente:
   * Específico: Crear un protocolo para evaluar y priorizar solicitudes adicionales de funcionalidades.
   * Medible: Revisar y clasificar cada nueva solicitud en un plazo de 5 días hábiles.
   * Alcanzable: Utilizar al equipo de desarrollo para implementar cambios viables según la prioridad.
   * Relevante: Adaptar el sistema a las necesidades del cliente sin afectar su operación actual.
   * Tiempo: Implementar el protocolo dentro del primer mes tras la entrega del sistema.

Este plan asegura que los problemas y riesgos identificados sean gestionados de manera eficiente, manteniendo la funcionalidad y la satisfacción del cliente.