**Plan de Calidad de Software**

|  |  |
| --- | --- |
| **Nombre del Proyecto:** | Zuluaga y Soto S.A.S |
| **Patrocinador:** | Jael López Gómez |
| **Empresa:** | Zuluaga y Soto S.A.S |
| **Dirección / Área:** | Administración |
| **Project Manager:** | Dayanna Loaiza |
| **QA Manager / Líder de Calidad:** | David Medina |

|  |
| --- |
| **Propósito del documento** |
| Mejorar la eficiencia y calidad de servicios de los servicios de comercialización y distribución de productos de consumo masivos, también adoptar prácticas ágiles, para gestionar de manera mas efectiva los proyectos relacionados con la comercialización y distribución de productos, implementar un ambiente de innovación y adaptabilidad, buscando constantemente nuevas formas de mejorar nuestros procesos y servicios, también obtener una mejora continua de la experiencia del cliente en todos los aspectos de la comercialización y distribución. |

|  |
| --- |
| **Estrategia de Pruebas** |
| Esta sección describe los tipos de pruebas que se realizarán en cada etapa del proyecto.  Por tipo de ejecución:   * Automatizado * Manuales   Pruebas funcionales:  • Gestión de inventario: el sistema debe permitir la actualización en tiempo real del inventario de productos, incluidos los bienes entrantes y salientes, así como la gestión de niveles mínimos y máximos de inventario.  • Realizar pedidos: los clientes deberían poder realizar un seguimiento del estado de sus pedidos en línea, desde la confirmación de la compra hasta la entrega final, con detalles de la fecha de entrega estimada y cualquier actualización relevante.  • Personalización de ofertas. El sistema debería poder analizar el historial de compras de los clientes y ofrecer recomendaciones de productos, promociones y descuentos personalizados adaptados a sus preferencias y comportamiento de compra.  • Gestión de promociones: debería poder configurar y gestionar promociones especiales, como descuentos por volumen, ofertas de temporada o cupones de descuento, con la opción de aplicar automáticamente promociones activas durante el proceso de compra.  • Análisis de datos de ventas. El sistema debe proporcionar herramientas de análisis e informes que permitan a la empresa monitorear de cerca el desempeño de las ventas, identificar tendencias, analizar el comportamiento del cliente y tomar decisiones informadas sobre estrategias de marketing y gestión de inventario.  Pruebas no funcionales:  • Seguridad de los datos: El sistema debe garantizar la seguridad y confidencialidad de la información del cliente, incluidos los datos personales y las transacciones financieras, mediante la implementación de medidas de cifrado, autenticación y protección contra el acceso no autorizado.  • Disponibilidad del sistema. El sistema debe estar disponible las 24 horas del día, los 7 días de la semana, con un tiempo de inactividad mínimo planificado para mantenimiento y actualizaciones para garantizar la continuidad del servicio y la satisfacción del cliente.  • Rendimiento y escalabilidad. El sistema debe ser capaz de manejar picos de tráfico y volúmenes de transacciones durante períodos de alta demanda, manteniendo un rendimiento óptimo y capacidad de respuesta del sitio web sin comprometer la calidad del servicio.  • Compatibilidad y facilidad de uso. El sistema debe ser compatible con una amplia gama de dispositivos y navegadores web, garantizando una experiencia de usuario consistente y fácil de usar para clientes de diferentes perfiles y niveles de habilidad tecnológica. |
| **Metodología** |
| En esta sección se describe la metodología a utilizar durante el ciclo de vida del proyecto.  • Product Backlog: Cree una cartera de productos que contenga una lista priorizada de todas las características, funcionalidades y mejoras que desea implementar en el proyecto. Este trabajo pendiente es propiedad del propietario del producto, que representa los intereses de los clientes y las partes interesadas.• Sprints cortos y regulares: divida el proyecto en iteraciones cortas y regulares, que normalmente duran de 1 a 4 semanas. Al final de cada sprint, el equipo entrega un producto potencialmente entregable.• Reuniones diarias: celebre reuniones diarias breves, conocidas como reuniones diarias, donde cada miembro del equipo comparte el progreso realizado, los obstáculos encontrados y los planes para el día.• Reunión de revisión del Sprint. Al final de cada sprint, el equipo lleva a cabo una reunión de revisión del sprint en la que presentan el trabajo completado al propietario del producto y a otras partes interesadas, reciben comentarios y ajustan el trabajo pendiente del producto según sea necesario.• Reunión Retrospectiva del Sprint. Después de la reunión de revisión del sprint, el equipo lleva a cabo una reunión retrospectiva del sprint para evaluar su desempeño, identificar áreas de mejora y planificar acciones para el próximo sprint.• Iteración y mejora continua: Scrum fomenta la iteración y la mejora continua, permitiendo al equipo adaptarse rápidamente a los cambios en los requisitos del cliente y el entorno del proyecto.  En caso de utilizar una metodología Ágil, se deberá describir el proceso y las actividades a realizar por el equipo de QA en cada iteración o Sprint. |
| **Gestión de Casos y Escenarios de Pruebas** |
| * Definición de los requisitos del proyecto: el equipo del proyecto colaborará para definir los requisitos funcionales y no funcionales del proyecto, en función de las necesidades del cliente y las expectativas de las partes interesadas. Esto incluye la identificación de características clave, funcionalidad esperada y limitaciones del sistema. * Casos de prueba: basándose en los requisitos especificados, el equipo creará una lista de casos de prueba que cubran todos los escenarios de uso posibles del sistema. Cada caso de prueba debe ser claro, específico y medible, con pasos detallados para reproducir el escenario y verificar el correcto funcionamiento. * Criterios de Aceptación: Para cada requisito, el equipo establecerá criterios de aceptación que determinarán cuándo el requisito se considera cumplido satisfactoriamente. Estos criterios servirán como referencia durante las pruebas para determinar si el sistema cumple con las expectativas del cliente y las partes interesadas. * Escenarios a incluir: Los escenarios de prueba cubrirán una variedad de situaciones que pueden surgir durante el uso del sistema, incluidos casos normales, casos extremos y casos de error. Se deben considerar diferentes combinaciones de insumos y condiciones para garantizar una cobertura completa de los requisitos. . |
| **Organización para la Calidad del Proyecto** |
| En esta sección se incluye la estructura del equipo de QA. También se debe incluir los roles, funciones y responsabilidades de cada miembro del equipo. |
| **Cronograma** |
| En esta sección se debe detalla el cronograma de actividades de QA, la utilización de recursos y los hitos importantes del proyecto.  Related image |
| **Herramientas** |
| Registro de Requisitos y Criterios de Aceptación:   * Microsoft Excel * Jira * Trello * Microsoft Word |
| **Plantillas y Formatos** |
| 1. Plantilla de caso de prueba: Esta plantilla debe incluir campos para describir el caso de prueba, los pasos de la prueba, los datos de entrada, los resultados esperados, el estado de la prueba y otros.  2. Plantilla de escenario de prueba: Esta plantilla puede contener campos para describir varios escenarios de prueba, incluidos datos de entrada, acciones del usuario, resultados esperados y observaciones.  3. Patrón de prueba de aceptación del usuario (UAT): Esta plantilla puede incluir campos para registrar los requisitos que se van a probar, los criterios de aceptación, los resultados de las pruebas y cualquier observación o comentario relevante.  4. Informe de prueba: Este formato puede incluir secciones para resumir los resultados de las pruebas realizadas, incluidos los hallazgos, los defectos encontrados, la cobertura de las pruebas, las métricas de calidad y las recomendaciones de mejora.  5. Formato de seguimiento de defectos: Este formato puede incluir campos para registrar detalles de cada defecto encontrado, incluida la descripción, prioridad, estado, asignación, fecha de descubrimiento, fecha de resolución y otros. |
| **Infraestructura** |
| Aquí mencionamos cada uno de los componentes necesarios para implementar el plan de calidad, entre los que podemos mencionar:  • Servidor para entorno de pruebas: un servidor dedicado o máquina virtual configurada específicamente para realizar pruebas de software.  • Conectividad: acceso a Internet de alta velocidad y red local para facilitar la comunicación y colaboración entre miembros calificados del equipo.  • Computadora: Una computadora de escritorio o portátil con capacidad suficiente para ejecutar herramientas de prueba y software de gestión de calidad.  • Tablet: Dispositivo móvil tipo tableta para realizar pruebas de compatibilidad y usabilidad en diferentes plataformas.  • Teléfono/Smartphone: Dispositivo móvil para probar aplicaciones móviles en diferentes sistemas operativos y versiones.  • Otros dispositivos: otro hardware necesario para realizar una prueba específica, como impresoras, escáneres, lectores de códigos de barras, etc.  • Software: herramientas de gestión de pruebas, automatización de pruebas, seguimiento de defectos y análisis de calidad del software, según los requisitos del proyecto.  • Oficina/Espacio Físico para el Equipo: Espacio físico adecuado para acomodar al equipo de calidad, con mesas, sillas, equipos de cómputo y otra infraestructura necesaria para realizar el trabajo de manera efectiva. |
| **Estándares** |
| • ISO 9001: Sistema de Gestión de Calidad (QMS): Norma internacional que especifica los requisitos para un sistema de gestión de calidad eficaz en una organización.  • ISO/IEC 25000: Ingeniería de Calidad de Software y Sistemas (SQuaRE): Un marco de referencia que proporciona una estructura para evaluar la calidad del software y los sistemas de información.  • CMMI (Integración del modelo de madurez de capacidad): modelo que describe las mejores prácticas para mejorar los procesos de desarrollo y mantenimiento del sistema.  • Six Sigma: Metodología que se enfoca en mejorar la calidad reduciendo la variabilidad en los procesos. |
| **Métricas** |
| En esta sección se define las métricas que se utilizarán para cuantificar los defectos del software, nivel de satisfacción, la eficiencia y los procesos empleados en el aseguramiento y control de la calidad.   * Número de Defectos Encontrados en las Pruebas: Durante el proceso de pruebas del sistema de gestión de inventarios de Zuluaga y Soto S.A.S., se identificaron un total de 35 defectos, que incluyeron problemas de funcionalidad, errores de cálculo y problemas de interfaz de usuario. * Número de defectos descubiertos por los usuarios finales: en el primer mes después del lanzamiento del nuevo sistema, los usuarios finales informaron un total de 15 defectos, incluidos errores de informes, problemas de rendimiento y dificultades de navegación. * Encuesta de Satisfacción del Usuario Final: Encuesta de satisfacción enviada a los clientes de Zuluaga y Soto S.A.S. para evaluar su experiencia con el nuevo sistema. El 85% de los encuestados expresó satisfacción con la facilidad de uso y funcionalidad del sistema. * Cobertura, Diseño y Requisitos del Código: Se realizó un análisis de cobertura de pruebas y se determinó que se había probado el 95% del código, el 90% de los requisitos funcionales y el 85% del diseño del sistema de gestión de inventarios. * Gravedad y prioridad de los defectos: los defectos descubiertos durante las pruebas se clasifican según su gravedad y prioridad. Por ejemplo, los defectos que afectan la función principal del sistema se clasifican como de alta gravedad y alta prioridad, mientras que los problemas cosméticos menores pueden tener baja gravedad y baja prioridad. * Distribución de Defectos: Se identificó que la mayoría de los defectos descubiertos durante las pruebas estaban relacionados con el módulo de gestión de pedidos, representando el 40% del total, seguido del módulo de facturación con un 30% y el módulo de reportes con un 20%. * Edad y Densidad del Defecto: Se ha calculado que el tiempo medio para resolver un defecto desde su detección hasta su corrección es de 2 días. Además, se determina que la densidad de defectos es 1 defecto por cada 1000 líneas de código. * Número de llamadas al servicio de asistencia técnica debido a defectos de software: durante el primer mes después del lanzamiento |
| **Presupuesto** |
| * Personal: Se deben calcular los costos de personal necesarios para llevar a cabo las actividades de aseguramiento de la calidad, incluyendo salarios, beneficios y posibles costos de capacitación. * Herramientas y software: se debe estimar el costo de comprar o licenciar las herramientas y el software necesarios para realizar pruebas, gestionar defectos y analizar la calidad del software. * Infraestructura: Se deben estimar los costos asociados con la compra, configuración y mantenimiento de la infraestructura necesaria para realizar pruebas, incluidos servidores, equipos informáticos, dispositivos móviles y otros dispositivos. * Capacitación: Se debe tener en cuenta el costo de capacitar al personal en el uso de herramientas y métodos de garantía de calidad, así como las mejores prácticas y estándares relevantes. * Costos generales: costos generales asociados con el proyecto de garantía de calidad, tales como: Deben incluirse el alquiler del lugar de trabajo, el material de oficina, los viajes y otros costes indirectos. * Contingencias: Se deberá reservar un porcentaje del presupuesto para contingencias e imprevistos que puedan surgir durante la ejecución del proyecto. |