**Plan de Calidad para el Proyecto de Desarrollo de una plataforma tecnológica para mejorar el acceso a tutores de calidad en América: Un enfoque para estudiantes de primaria y secundaria en Colombia**

Objetivo de Calidad:

Contribuir a mejorar el acceso a tutores de calidad para niños de primaria y secundaria en América Latina, reduciendo las barreras que impiden encontrar tutores calificados, asequibles y seguros.

Actividades de Planificación de la Calidad:

1. **Definición de Requisitos de Calidad**:
   * **Funcionalidad**:
     1. El sistema deberá de permitir un registro.
     2. El sistema deberá tener un control de acceso.
     3. El sistema deberá permitir una asignación.
     4. El sistema deberá permitir hacer un seguimiento.
     5. El sistema deberá permitir asignación de actividades.
     6. El sistema deberá permitir acceso a contenido interactivo.
   * **Rendimiento**:
     1. El sistema debe tener tiempos de respuesta rápidos al realizar operaciones de gestión de turnos, incluso cuando se manejen grandes volúmenes de datos.
     2. Debe ser capaz de manejar simultáneamente múltiples sesiones de usuarios sin degradación significativa del rendimiento.
   * **Usabilidad**:
     1. La interfaz de usuario debe ser intuitiva y fácil de usar, permitiendo a los usuarios navegar fácilmente por el sistema y realizar sus tareas sin dificultad.
     2. Debe proporcionar herramientas y funcionalidades que faciliten la programación y gestión de turnos, como calendarios interactivos y notificaciones automáticas.
   * **Seguridad**:
     1. El sistema debe proteger la información confidencial de los empleados, como datos personales y detalles de los turnos, mediante mecanismos de autenticación y autorización.
     2. Debe cumplir con los estándares de seguridad de la industria para garantizar la integridad y confidencialidad de los datos.
   * **Mantenibilidad**:
     1. El código del sistema debe estar bien estructurado y documentado, permitiendo futuras modificaciones y actualizaciones sin causar impactos negativos en el funcionamiento del sistema.
     2. Debe ser fácil de mantener y administrar, con herramientas que faciliten la identificación y corrección de errores.
   * **Compatibilidad**:
     1. El sistema debe ser compatible con diferentes dispositivos y navegadores web, permitiendo a los usuarios acceder y utilizar el sistema desde cualquier lugar y dispositivo.
     2. Debe integrarse con otros sistemas de la organización, como sistemas de recursos humanos y sistemas de planificación de recursos empresariales (ERP).
2. **Establecimiento de Estándares de Calidad**:
   * **Codificación**:
     1. Se utilizarán convenciones de nomenclatura claras y consistentes para variables, funciones y archivos.
     2. Se seguirán las mejores prácticas de codificación, como la modularización y la reutilización de código.
     3. Se evitarán las prácticas de codificación problemáticas, como la duplicación de código y la utilización de variables globales.
   * **Diseño de la Interfaz de Usuario**:
     1. La interfaz de usuario será intuitiva y fácil de usar, siguiendo principios de diseño centrados en el usuario.
     2. Se utilizarán estándares de diseño de interfaz de usuario reconocidos, como Material Design o iOS Human Interface Guidelines, para garantizar la coherencia y la familiaridad para los usuarios.
   * **Pruebas**:
     1. Se llevarán a cabo pruebas exhaustivas en todas las etapas del desarrollo, incluyendo pruebas unitarias, de integración y de aceptación.
     2. Se establecerán criterios de cobertura de pruebas para garantizar que se prueben todas las funcionalidades críticas del sistema.
   * **Seguridad**:
     1. Se implementarán medidas de seguridad adecuadas, como autenticación de usuarios, cifrado de datos y protección contra vulnerabilidades conocidas.
     2. Se realizarán pruebas de seguridad regulares para identificar y corregir posibles vulnerabilidades en el sistema.
   * **Documentación**:
     1. Se mantendrá una documentación detallada y actualizada del sistema, incluyendo especificaciones de diseño, guías de usuario y manuales técnicos.
     2. La documentación será accesible y fácil de entender para los desarrolladores, usuarios finales y otros miembros del equipo.
   * **Mantenibilidad**:
     1. El código del sistema se estructurará de manera clara y modular, facilitando su comprensión y modificación por parte de otros desarrolladores.
     2. Se establecerán procedimientos de gestión de cambios y versiones para garantizar la trazabilidad y la integridad del código.
   * **Compatibilidad**:
     1. El sistema será compatible con una variedad de navegadores web y dispositivos, asegurando una experiencia consistente para todos los usuarios.
     2. Se realizarán pruebas de compatibilidad para verificar el rendimiento del sistema en diferentes entornos y configuraciones.
3. **Planificación de Pruebas**:
4. **Pruebas Unitarias**:
   * Verificar que cada componente del sistema funcione correctamente de manera individual.
   * Validar que los módulos de programación de turnos, gestión de permisos y registro de empleados funcionen según lo previsto.
5. **Pruebas de Integración**:
   * Probar la interacción entre los diferentes componentes del sistema para garantizar su correcta integración.
   * Verificar la comunicación y el intercambio de datos entre los módulos de gestión de turnos, registro de empleados y gestión de permisos.
6. **Pruebas Funcionales**:
   * Validar que el sistema cumpla con todos los requisitos funcionales especificados en la documentación.
   * Realizar pruebas de extremo a extremo para simular escenarios de uso real, como la creación de turnos, la asignación de permisos y la gestión de días feriados.
7. **Pruebas de Rendimiento**:
   * Evaluar el rendimiento del sistema bajo diferentes cargas de trabajo y volúmenes de datos.
   * Verificar los tiempos de respuesta del sistema al realizar operaciones de gestión de turnos, especialmente durante horas pico de actividad.
8. **Pruebas de Usabilidad**:
   * Evaluar la facilidad de uso y la experiencia del usuario del sistema.
   * Realizar pruebas de usabilidad con usuarios reales para identificar posibles problemas de navegación, diseño de interfaz y flujo de trabajo.
9. **Pruebas de Seguridad**:
   * Evaluar la seguridad del sistema mediante pruebas de penetración y análisis de vulnerabilidades.
   * Verificar la resistencia del sistema a posibles ataques maliciosos y la protección de datos sensibles de los empleados.

Herramientas y Recursos:

* Herramientas de automatización de pruebas para agilizar el proceso de prueba y aumentar la cobertura de pruebas.
* Equipos de prueba y entornos de prueba dedicados para realizar pruebas exhaustivas sin afectar el entorno de producción.
* Personal de prueba capacitado y con experiencia para ejecutar y supervisar las actividades de prueba de manera efectiva.

Criterios de Aceptación:

* Todas las pruebas unitarias y de integración deben pasar exitosamente sin errores críticos.
* El sistema debe cumplir con al menos el 90% de los casos de prueba funcional.
* El rendimiento del sistema debe cumplir con los objetivos establecidos para los tiempos de respuesta y la capacidad de carga.

Plan de Ejecución:

* El plan de pruebas se ejecutará durante un período de dos semanas, con sesiones diarias de prueba y reuniones de revisión de progreso.
* Se asignarán recursos adicionales según sea necesario para abordar problemas identificados durante las pruebas.

1. **Asignación de Recursos de Calidad**:
   * Designar un equipo de calidad responsable de ejecutar y supervisar las actividades de aseguramiento de la calidad, incluyendo pruebas, revisión de código y seguimiento de defectos.
   * Proporcionar recursos adecuados, como herramientas de pruebas y equipos de prueba, para garantizar la efectividad de las actividades de calidad.
2. **Definición de Procesos de Gestión de Cambios**:
   * Establecer procesos para gestionar cambios en el software, incluyendo solicitudes de cambio, evaluación de impacto y seguimiento de versiones.
   * Garantizar la integridad y trazabilidad del software a lo largo de su ciclo de vida.
3. **Planificación de Revisiones y Auditorías**:
   * Programar revisiones periódicas del código, la documentación y los artefactos del proyecto para identificar posibles problemas y áreas de mejora.
   * Realizar auditorías de calidad para verificar el cumplimiento de los estándares y procedimientos definidos.

Responsabilidades:

* El equipo de desarrollo es responsable de seguir los estándares de calidad establecidos, producir código de alta calidad y realizar pruebas exhaustivas.
* El equipo de calidad es responsable de supervisar y ejecutar las actividades de aseguramiento de la calidad, incluyendo pruebas, revisiones y auditorías.

Recursos y Cronograma:

* Se asignará un porcentaje de tiempo y recursos del proyecto específicamente para actividades de calidad, como pruebas y revisiones.
* Las actividades de calidad se integrarán en el cronograma general del proyecto para garantizar su inclusión y seguimiento adecuados.

Seguimiento y Mejora Continua:

* Se realizarán reuniones regulares para revisar el progreso del proyecto en términos de calidad y identificar áreas de mejora.
* Se llevará a cabo una evaluación final de la calidad del sistema antes de la entrega, utilizando las métricas y criterios definidos previamente.