

Sistema de Gestión de Tareas

ARQUITECTURA ATAM

1. Contexto del Proyecto:

Objetivo Principal:

Desarrollar un sistema que permita a los usuarios organizar, administrar y realizar un seguimiento eficiente de sus tareas y proyectos.

Funcionalidades Clave:

- Creación de tareas.
- Edición de tareas.
- Visualización de tareas.
- Eliminación de tareas.

2. Definición de Objetivos de Calidad:

- **Usabilidad:** El sistema debe ser intuitivo y fácil de usar.
- **Eficiencia:** El sistema debe permitir a los usuarios gestionar sus tareas de manera rápida y efectiva.
- **Flexibilidad:** El sistema debe adaptarse a diversos tipos y tamaños de proyectos.
- **Seguridad:** Los datos de las tareas deben estar protegidos y solo accesibles por usuarios autorizados.

3. Identificación de Escenarios:

3.1 Escenario de Usabilidad:

Estímulo: Un usuario intenta crear una nueva tarea.

Respuesta: El sistema permite la creación de la tarea de manera intuitiva y con instrucciones claras.

3.2 Escenario de Eficiencia:

Estímulo: Un usuario intenta visualizar sus tareas pendientes.

Respuesta: El sistema muestra rápidamente la lista de tareas pendientes con opciones para filtrar y ordenar.

3.3 Escenario de Flexibilidad:

Estímulo: Un usuario intenta gestionar tareas de un proyecto grande.

Respuesta: El sistema se adapta para gestionar el volumen de tareas sin pérdida de rendimiento o usabilidad.

3.4 Escenario de Seguridad:

Estímulo: Un usuario no autorizado intenta acceder a las tareas de otro usuario.

Respuesta: El sistema impide el acceso y registra el intento.

4. Creación del Modelo de Arquitectura Inicial:

Desarrollaremos una interfaz de Usuario para realizar acciones sobre las tareas, el Controlador de Tareas procesa estas acciones e interactúa con la Base de Datos para almacenar información.

5. *Análisis Preliminar:*

- Verificar si la arquitectura propuesta cumple con los objetivos de calidad.
- Analizar posibles conflictos entre objetivos de calidad, como la seguridad y la usabilidad.

6. *Identificación de Trade-offs:*

- **Usabilidad vs Eficiencia:** Balancear la simplicidad de la interfaz de usuario con la eficiencia operativa.
- **Flexibilidad vs Rendimiento:** Ajustar la capacidad de adaptación con el mantenimiento del rendimiento.

7. *Iteración y Refinamiento:*

- **Pruebas Continuas:** Realizar pruebas regulares para validar y mejorar los aspectos de calidad definidos.
- **Optimización de Recursos:** Ajustar el uso de recursos para mejorar el rendimiento y la escalabilidad.

8. *Resultados:*

Se desarrolla y se ajusta una arquitectura que cumpla con los objetivos de calidad y una descripción detallada de la arquitectura refinada y optimizada.

ARQUITECTURA QAW

1. *Identificación de Atributos de Calidad:*

Usabilidad:Facilidad de uso y aprendizaje del sistema.

Eficiencia:Rapidez en la gestión de tareas.

Flexibilidad:Adaptabilidad a diversos tipos y tamaños de proyectos.

Seguridad:Protección de datos y control de acceso.

2. *Priorización de Atributos de Calidad:*

Primaria:Seguridad (Proteger la información es crucial).

Secundaria:Usabilidad (Importante para la aceptación del usuario).

Terciaria:Eficiencia y Flexibilidad (Importante para la operación efectiva).

3. *Desarrollo de Escenarios de Calidad:*

Escenario de Seguridad:

- **Estímulo:** Intento de acceso no autorizado.
- **Respuesta:** Bloqueo de acceso y registro del intento.

Escenario de Usabilidad:

- **Estímulo:** Usuario nuevo accede al sistema.
- **Respuesta:** El usuario comprende rápidamente cómo usar el sistema sin necesidad de consultas adicionales.

Escenario de Eficiencia:

- **Estímulo:** Usuario busca visualizar sus tareas.
- **Respuesta:** El sistema muestra rápidamente la información requerida.

Escenario de Flexibilidad:

- **Estímulo:** Se introduce un nuevo tipo de tarea.
- **Respuesta:** El sistema se adapta sin requerir cambios mayores en el código.

4. Definición de métricas:**• Usabilidad:**

- **Tiempo de Aprendizaje:** Tiempo que le toma a un usuario nuevo aprender a usar el sistema hasta un nivel de competencia aceptable.
- **Tasa de Errores de Usuario:** Número de errores que los usuarios cometen al interactuar con el sistema.

• Eficiencia:

- **Tiempo de Respuesta:** Tiempo que tarda el sistema en responder a una solicitud del usuario.
- **Uso de Recursos:** Cantidad de recursos del sistema (por ejemplo, memoria, CPU) que se utilizan durante las operaciones.
- **Throughput:** Cantidad de tareas que el sistema puede procesar en un período de tiempo determinado.

• Flexibilidad:

- **Tiempo y Esfuerzo para Modificar:** Medida del tiempo y esfuerzo requeridos para implementar cambios en el sistema.
- **Facilidad de Añadir Nuevas Funcionalidades:** Nivel de dificultad para añadir nuevas funcionalidades al sistema.

• Seguridad:

- **Tiempo hasta la Primera Fallo de Seguridad:** Tiempo que pasa hasta que se descubre la primera vulnerabilidad de seguridad después de una liberación.
- **Número de Vulnerabilidades Descubiertas:** Número de vulnerabilidades de seguridad descubiertas durante un período de tiempo específico.