



## **Pruebas de Software y aseguramiento de la calidad**

### **Actividad 3.2. Análisis de requisitos**

Alumno: Roberto Perezcano Hernández

Matrícula: A01730502

Fecha: 01 de febrero de 2026

# Contenido

Introducción: .....	3
Problema 1: Restricciones generales .....	3
Contexto:.....	3
Análisis - Retos Identificados .....	3
Propuesta de Redacción.....	3
Problema 2: Quality Attributes .....	4
Contexto:.....	4
Análisis - Retos Identificados .....	4
Propuesta de Redacción.....	4
Problema 3: Cruce de calle .....	5
Contexto:.....	5
Análisis - Retos Identificados .....	5
Propuesta de Redacción.....	5
Problema 4: Confiabilidad.....	6
Contexto:.....	6
Análisis - Retos Identificados .....	6
<b>Falta de Verificabilidad:</b> "Alrededor del 0.1%" es una medida imprecisa. ¿Es un porcentaje de registros, de transacciones o de tiempo?.....	6
<b>Inviabilidad Técnica:</b> La frase "permanentemente disponible" (100% uptime) es prácticamente imposible de alcanzar y extremadamente costosa, lo que lo hace un requisito poco realista. ....	6
<b>Términos Subjetivos:</b> El término "robusto a daños externos" es ambiguo. No define si se refiere a fallos de hardware, ataques cibernéticos o desastres naturales. .....	6
Propuesta de Redacción.....	6
Conclusión: .....	6
Referencias:.....	7

## Introducción:

En este reporte realizaremos un análisis crítico de la calidad de diversos requisitos de software, reconociendo que la precisión en su definición es fundamental para el éxito de cualquier proyecto de desarrollo.

Un requisito mal redactado, ambiguo o incompleto no solo incrementa los costos de corrección en etapas posteriores, sino que compromete la integridad y utilidad del producto final.

## Problema 1: Restricciones generales

### Contexto:

El sistema puede ser usado sólo por personal autorizado, que son aquellas personas que el arquitecto designe como auxiliares en el manejo del sistema. No es necesario definir diferentes usuarios, ni tipos de accesos, ya que todo el personal autorizado podrá tener acceso completo al sistema por medio de un password (único para todos los usuarios), que le permitirá ejecutar cualquier función del sistema.

Otra restricción general es el software con que cuenta la computadora en que será instalado. El sistema se hará en una plataforma de C++, con paradigma orientado a objetos, con una base de datos de ACCESS. El arquitecto tiene Microsoft ACCESS 2000, pero no tiene C++ Builder 5, por lo que habrá que hacerlo ejecutable en su computadora.

### Análisis - Retos Identificados

**Riesgo de Seguridad y Falta de Trazabilidad:** El uso de un "password único" para todos los usuarios impide la auditoría de acciones, es decir, no se sabe quién hizo qué, vulnerando el principio de seguridad básica.

**Ambigüedad en la Designación:** El término "aquellas personas que el arquitecto designe como auxiliares" es subjetivo y no define un proceso técnico o perfil para la creación de cuentas.

**Obsolescencia Tecnológica:** El requisito exige compatibilidad con herramientas obsoletas, en este caso Access 2000, lo cual compromete la mantenibilidad y seguridad a largo plazo.

### Propuesta de Redacción

El sistema deberá implementar un módulo de Control de Acceso basado en Roles (RBAC). Cada usuario designado por el Administrador (Arquitecto) contará con credenciales individuales únicas, usuario y contraseña. El sistema debe ser entregado como un archivo ejecutable autónomo compatible con sistemas operativos Windows 10 o superiores,

utilizando una base de datos relacional que garantice la integridad de los datos sin requerir la instalación previa de entornos de desarrollo en la computadora del cliente.

## Problema 2: Quality Attributes

### Contexto:

**Security:** There will be no security issues embedded into the software. Though teachers may feel their portion of the software should be secured from the students, the teacher portion of the project is merely an outline to help the teacher with the lesson plan and would most likely not be of as much interest as the other components to students.

**Availability:** Since the software will be a web-based application on a CD, it will be readily available to the general public. To obtain a copy of the tool will require the burning of all source files onto a CD.

**Maintainability:** The application should be maintainable because of the modular form. The team designed the application in such a way that more features can easily be added without interfering with the base product.

**Reliability:** Since the software doesn't rely on external devices, the application should be fully functional on any platform with a web browser that has Java Script enabled, audio capabilities, and a Flash plug-in. The only causes of failure in the software might be due to external constraints that the tool has no control over.

### Análisis - Retos Identificados

**Premisa de Seguridad Incorrecta:** Afirmar que "no habrá problemas de seguridad" por un supuesto interés de los alumnos es un error de análisis de riesgos.

**Contradicción de Disponibilidad:** Define la aplicación como "basada en web" pero distribuida en "CD", lo cual crea confusión sobre el entorno de ejecución real y el despliegue.

**Dependencia de Tecnologías Obsoletas:** El requisito de un "Flash plug-in" hace que el sistema sea no funcional en la actualidad, ya que Flash ha sido descontinuado por completo.

### Propuesta de Redacción

**Seguridad:** El sistema debe restringir el acceso al módulo de planes de lección exclusivamente a usuarios con rol de "Docente" mediante autenticación.

**Disponibilidad:** La aplicación será desplegada en un servidor web institucional, garantizando su acceso durante el ciclo escolar.

**Confiabilidad:** La interfaz debe ser compatible con navegadores modernos (Chrome, Firefox, Edge) utilizando estándares HTML5 y JavaScript, sin requerir complementos de terceros (como Flash).

## Problema 3: Cruce de calle

### Contexto:

Permitir la ubicación de un cruce de calle basándose en la base de datos de red vial y la cartografía de la misma. El sistema deberá ser capaz de ubicar un cruce de calle en la cartografía de red vial y/o planimetría, consultando primero la base de datos de red vial y cruces de calles de manera que el usuario tenga la opción de ubicar el cruce de la siguiente manera:

- Por el inicio del nombre de alguna de las calles que crucen
- Por una parte del nombre de alguna de las calles que crucen

El resultado de los cruces que concuerden con lo solicitado por el usuario deberá ser mostrado en una lista para que seleccionado alguno de estos resultados se tenga la opción de ubicar el cruce en la ventana donde se muestre la cartografía de planimetría y/o red vial.

Este requerimiento debe ser implementado para su operación, a través de un navegador de Internet.

### Análisis - Retos Identificados

**Falta de Especificación de Desempeño:** No se indica el tiempo de respuesta esperado para la búsqueda en la base de datos de red vial.

**Ambigüedad en la Interfaz:** "Mostrar en una lista" es vago; no especifica el número de resultados permitidos o qué datos se deben mostrar para diferenciar cruces similares.

**Requisito Incompleto:** No define el comportamiento del sistema cuando la búsqueda no arroja coincidencias o cuando existen nombres de calles duplicados en diferentes sectores.

### Propuesta de Redacción

El sistema debe permitir al usuario buscar cruces de calles mediante un campo de texto con función de autocompletado. Los resultados deben listarse en menos de 2 segundos, mostrando el nombre de ambas calles y la colonia o sector. Al seleccionar un elemento de la lista, el sistema centrará y resaltará el cruce en el visor cartográfico con un mayor nivel de zoom.

## Problema 4: Confiabilidad

### Contexto:

La información que se maneja en el sistema de difusión es muy importante, por tanto, se requiere un sistema robusto a daños externos para que la pérdida de datos sea alrededor del 0.1%.

El sistema de difusión debe de estar permanentemente disponible por Internet.

### Análisis - Retos Identificados

**Falta de Verificabilidad:** "Alrededor del 0.1%" es una medida imprecisa. ¿Es un porcentaje de registros, de transacciones o de tiempo?.

**Inviabilidad Técnica:** La frase "permanentemente disponible" (100% uptime) es prácticamente imposible de alcanzar y extremadamente costosa, lo que lo hace un requisito poco realista.

**Términos Subjetivos:** El término "robusto a daños externos" es ambiguo. No define si se refiere a fallos de hardware, ataques cibernéticos o desastres naturales.

### Propuesta de Redacción

El sistema de difusión debe garantizar una disponibilidad de servicio del 99.9% (Tier 3) medida mensualmente. Para asegurar la integridad de la información, se debe implementar un esquema de respaldos automatizados cada 24 horas y redundancia de datos (RAID 1 o superior), limitando la pérdida máxima de datos aceptable (RPO) a las últimas 4 horas de operación ante un fallo crítico del servidor.

## Conclusión:

La revisión de los requisitos permite transformar necesidades vagas en especificaciones técnicas accionables.

Como se observó en los casos analizados, la transición de un lenguaje informal a uno estandarizado reduce significativamente la incertidumbre para el equipo de desarrollo y asegura que el aseguramiento de calidad cuente con criterios objetivos para las fases de prueba.

La implementación de estas prácticas de ingeniería de requisitos es, el primer paso para garantizar un producto de software de alta calidad.

## Referencias:

- Weigers, K. (1999). Writing Quality Requirements. Process impact.
- Wiegers, K. E., & Beatty, J. (2013). Software Requirements: Vol. Third edition. Microsoft Press.