

Instituto Tecnológico y de Estudios Superiores de Monterrey

Actividad 3.2 Análisis de requisitos

Ramírez Álvarez Kenji Omar

A01796851

Pruebas de software y aseguramiento de la calidad (Gpo 10)

Problema 1: Restricciones generales

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.

Referencia: AGOPA (Administración de Obras para Arquitectos). Especificación de requerimientos. Disponible en el sitio del curso [SRS_Example_1].

Identificación de Problemas

- El acceso será para el "personal que el arquitecto designe". No define un proceso dentro del sistema para esta designación. Además, el uso de un password único para todos los usuarios es un error crítico de integridad. No hay forma de auditar quién hizo qué (no hay accountability), lo que viola el atributo de calidad de seguridad.
- El requisito dice que "no es necesario definir diferentes tipos de accesos", pero luego dice que el personal podrá "ejecutar cualquier función". Esto es un error de análisis: en cualquier sistema real, dejar que todos los auxiliares tengan acceso total (como borrar la base de datos completa) pone en riesgo la robustez y la seguridad de la información del proyecto.

- Especificar "C++ orientado a objetos" y "Base de datos Access 2000" son decisiones de diseño, no requisitos funcionales. Obligar al uso de una base de datos de hace más de 20 años (Access 2000) compromete la Factibilidad y el rendimiento a largo plazo.
- La frase "habrá que hacerlo ejecutable en su computadora" es redundante y vaga. No especifica el Sistema Operativo o librerías necesarias para que ese "ejecutable" realmente funcione.

Propuesta de Mejora

El sistema deberá permitir el acceso únicamente a usuarios registrados mediante credenciales individuales (usuario y contraseña), será administrado bajo un usuario con perfil de "Administrador" (Arquitecto) para crear, modificar y dar de baja cuentas de usuario para el personal auxiliar.

Así mismo esta administración de usuarios generará un log para llevar un registro de las modificaciones del sistema por usuario.

El software debe ser entregado como un archivo ejecutable independiente que no requiera la instalación de entornos de desarrollo (IDE) adicionales en la computadora del cliente.

De acuerdo con los principios de compatibilidad, el sistema debe ser compatible con bases de datos relacionales y estas serán exportables a otros sistemas que requieran los usuarios.

Problema 2: Quality Attributes

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.

Referencia: History of Art in 40 Megabytes or Less project. Disponible en el sitio del curso [SRS_Example_6]

Identificación de Problemas

- El texto usa palabras prohibidas por Wiegiers como "easily", "should be" y "fully functional". Decir que algo es fácil de mantener porque es "modular" es una suposición de diseño, no un requisito verificable. Según Earl Beede, esto es "pensamiento de solución" en lugar de definir la necesidad real.
- Existe una contradicción lógica grave: "web-based application on a CD". Si es basada en web, su disponibilidad depende de un servidor y conexión; si está en un CD, es una aplicación local. Además, define la disponibilidad como "quemar un CD", lo cual no tiene nada que ver con la métrica de Availability (tiempo de actividad/uptime).
- Evaluar correctamente si la mejor tecnología para la distribución del sistema aun aplicaría en formato CD o entregar la solución vía web (red local o internet)
- Afirmar que "no habrá problemas de seguridad" es una declaración falsa, todo software tiene vulnerabilidades. Además, descarta la necesidad de seguridad basándose en una suposición subjetiva de que a los estudiantes "no les interesará" el material del profesor. Esto viola el principio de Integridad de los datos.
- Para el principio de reliability, no establece una métrica. Decir que las fallas serán por "causas externas" es una forma de evadir la responsabilidad del software, haciendo que el requisito no sea testeable.

Propuesta de Mejora

Seguridad (Security)

1. AC-SEC-01: El sistema deberá restringir el acceso a la sección "Lesson Plan" mediante autenticación de usuario. Solo los usuarios con rol "Teacher" podrán visualizar o editar este contenido. Para los estudiantes no será necesario un método de autenticación para priorizar el acceso al contenido y evitar el manejo de bases de datos adicionales para la administración tal volumen de usuarios.

Disponibilidad (Availability)

2. La aplicación deberá ser posible entregar por medio de diversas tecnologías para funcionar de manera local o web en función de las restricciones tecnológicas.

Mantenibilidad (Maintainability)

3. El código debe seguir el estándar de documentación y mantener un índice de actualizable que permita añadir nuevos módulos de lecciones sin modificar el núcleo del motor de búsqueda. Este también deberá ser actualizable por medios físicos y vía red.

Confiabilidad (Reliability)

4. El sistema debe funcionar sin errores críticos durante sesiones de uso continuo de 6 horas en navegadores compatibles y modernos.

Problema 3: Cruce de calle

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.

Identificación de Problemas

- El requisito repite varias veces los términos "cartografía", "red vial" y "planimetría" en párrafos largos. Según Wiegers, un requisito de calidad debe ser conciso. La redacción actual dificulta identificar rápidamente la acción principal del sistema.
- El texto dice que al seleccionar un resultado, se tendrá "la opción de ubicar el cruce". No aclara qué significa "ubicar": ¿El mapa se centra automáticamente? ¿Se coloca un marcador/pin? ¿Se resalta la calle? Esta falta de detalle afecta la Verificabilidad, eso dificulta saber cómo testear el sistema.
- Este requerimiento no especifica qué sucede si no hay coincidencias o si existen múltiples cruces con nombres idénticos.

- La base de datos de cartografía y la de red vial no están sincronizadas o en que manera se requieren unificar.
- No define si la búsqueda por "parte del nombre" debe ser sensible a mayúsculas, acentos o si debe ignorar artículos (ej. "Reforma" vs. "Paseo de la Reforma").

Propuesta de Mejora

Requisitos Funcionales (Búsqueda y Ubicación):

1. El sistema deberá permitir la búsqueda de cruces de calles mediante dos criterios: coincidencia inicial del nombre o coincidencia parcial de cualquiera de las vías integrantes.
2. El sistema deberá mostrar los resultados que coincidan en una lista que incluya el nombre oficial de ambas calles y la zona/colonia correspondiente.
3. Al seleccionar un elemento de la lista, el sistema deberá centrar la vista de la cartografía en las coordenadas del cruce y marcarlo con un icono distintivo.
4. Si la búsqueda no genera resultados, el sistema deberá mostrar un mensaje informativo sugiriendo revisar la ortografía o el nombre de las calles.

Restricciones de Implementación:

1. La funcionalidad de consulta y visualización cartográfica debe ser accesible y operativa a través de navegadores web estándar (HTML5).
2. Los resultados de las búsquedas deberán dar respuesta en 2 segundos.
3. La búsqueda comienza automáticamente después de 3 caracteres.
4. La carga de información de la cartografía será mediante una api para utilizarla de manera actualizada.

Problema 4: Confiabilidad

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.

Identificación de Problemas

- El término "alrededor del 0.1%" es inaceptable en un requisito de calidad según Wieggers. No es clara. ¿Es el 0.1% de los registros, del tamaño de los archivos, o de las transacciones diarias? Sin una definición precisa no se puede validar el cumplimiento.
- Exigir que un sistema esté "permanentemente disponible" implica un 100% de disponibilidad y en la práctica el 100% de disponibilidad es teóricamente imposible y extremadamente costoso. Incluso los niveles de "cinco nueves" (99.999%) requieren redundancias masivas. Este requisito no es Factible
- ¿A qué se refiere con daños externos? ¿Ataques DDoS, desastres naturales, fallos en el suministro eléctrico o errores del ISP? Sin definir el ámbito y el contexto los desarrolladores no saben si deben implementar redundancia de servidores, firewalls avanzados o protocolos de recuperación ante desastres
- El requisito mezcla la Integridad (no perder datos) con la Disponibilidad (estar en línea) y la Robustez (resistir fallos). Según Wieggers, estos deben tratarse como atributos separados porque tienen diferentes tradeoffs.

Propuesta de Mejora

1. Disponibilidad

Porcentaje de tiempo que el sistema de difusión es accesible vía HTTP/HTTPS durante un mes calendario.

- Mínimo (Must): 99.9%.
- Objetivo (Plan): 99.95%.

Medidor: Reporte mensual del software de monitoreo de red

2. Integridad de Datos

El sistema debe garantizar que, ante fallos críticos del servidor o pérdida de conectividad, los datos de difusión se preserven sin corrupción.

La tasa de pérdida de paquetes o corrupción de archivos debe ser menor al 0.1% del volumen total de datos procesados por sesión.

Implementación de respaldos automáticos cada 24 horas y protocolos de verificación de redundancia.

Bibliografía

- **Wiegers, K. E. (1999).** *Writing quality requirements*. Process Impact.
<https://www.processimpact.com/articles/qualreqs.pdf>
- **Wiegers, K. E., & Beatty, J. (2013).** *Software requirements* (3.^a ed.). Microsoft Press.
- **Construx Software. (2023, 26 de julio).** *Four reasons why requirements are hard | Earl Beede* [Video]. YouTube. <https://www.youtube.com/watch?v=aocJgcycWME>