

# Entornos de Desarrollo

## Bloque 1

### Tema 4: Especificación de los requisitos de una aplicación

---

Solución de los ejercicios:

**1.4.2.1.** Responde a las siguientes afirmaciones indicando si son verdaderas o falsas, y razonando la respuesta:

- Las entrevistas son la forma más recomendable de obtener información para un proyecto software, en general.

Se puede considerar **falsa** (porque depende del proyecto en sí) o **verdadera** (porque es la forma más completa de conocer las necesidades del usuario en algunos proyectos).

- Mediante JAD se pretende obtener de forma rápida una especificación de requisitos para proyectos donde intervienen diferentes departamentos o secciones de la empresa cliente.

Como en la afirmación anterior, podríamos considerarla **verdadera** (porque es un método adecuado cuando hay más de un departamento del cliente participando en el proyecto) y **falsa** (porque no es rápido ya que las reuniones pueden durar varios días)

- Los cuestionarios son útiles para recabar información del cliente cuando sus componentes están dispersos geográficamente.

#### Verdadero

- Mediante únicamente la observación *in situ* o el estudio de documentación se podría obtener una ERS muy completa.

**Falso.** Si no obtenemos información de la gente que trabaja en la empresa, difícilmente podremos obtener requerimientos completos y útiles.

**1.4.3.1.** Lee el siguiente enunciado sobre el desarrollo de un producto software, e identifica los requisitos de sistema, funcionales y no funcionales:

*Un blog tiene tres tipos de usuarios: administradores, editores y visitantes. Cualquiera de ellos debe loguearse para entrar a la aplicación. Los administradores pueden dar de alta a otros usuarios, los editores pueden publicar posts, y los visitantes pueden comentar los posts de los editores. Se espera que el blog tenga una gran afluencia, por lo que la disponibilidad del servicio debe de ser alta. También se necesitarán tener varios equipos servidores para balancear la carga y repartirse los accesos de los clientes, evitando la saturación. Por otra parte, al no contar con editores demasiado experimentados, se desea que la interfaz de usuario para que éstos puedan*

*publicar los posts sea lo más sencilla posible. Toda la información se almacenará en una base de datos MySQL, con copias de respaldo diarias. Las contraseñas de los usuarios estarán encriptadas para evitar que se puedan descifrar.*

**Solución:**

Requerimientos		
Sistema	Funcionales	No funcionales
<ul style="list-style-type: none"> <li>Varios servidores con balanceo de carga</li> <li>Conexión de Internet</li> <li>Bases de datos MySQL</li> </ul>	<ul style="list-style-type: none"> <li>Tres roles de usuario (Administradores, editores y visitantes)</li> <li>Los administradores pueden registrar otros usuarios</li> <li>Los editores pueden publicar posts</li> <li>Los visitantes pueden comentar posts</li> <li>Login de usuario</li> <li>Almacenar cada post y cada comentario en la base de datos.</li> <li>Realizar copias de seguridad de la base de datos.</li> <li>Las contraseñas deben estar encriptadas.</li> </ul>	<ul style="list-style-type: none"> <li>Alta disponibilidad</li> <li>Fácil de usar</li> </ul>

**1.4.3.2.** A partir de la siguiente propuesta de software a desarrollar, identifica los requisitos de sistema, funcionales y no funcionales.

*Un centro cultural de un municipio se dedica al préstamo de dos tipos de materiales de préstamo: discos y libros. Necesitamos una aplicación que nos permita dar de alta materiales de ambos tipos. Por otra parte, a este centro cultural acuden clientes, que se identificarán en la aplicación con un login y password. Después, deben poder buscar los materiales por tipo (disco o libro) y/o título del material. Una vez seleccionado el material, deben poderlo pedir, o reservar si no está disponible. Los clientes pueden tener hasta 5 materiales en préstamo simultáneamente. Al ser un municipio pequeño, no se espera que el sistema esté muy solicitado, y basta con un simple equipo y una pequeña base de datos para almacenar la información. En el caso de que un cliente no recuerde su contraseña, se le volverá a enviar al correo que indicó al darse de alta en el sistema.*

**Solución:**

Requerimientos		
Sistema	Funcionales	No funcionales
<ul style="list-style-type: none"> <li>Un ordenador sencillo es suficiente</li> <li>Una base de datos pequeña</li> </ul>	<ul style="list-style-type: none"> <li>Los usuarios deben identificarse</li> <li>Los usuarios deben poder recuperar su contraseña por email</li> <li>Añadir nuevos libros y discos a la colección</li> <li>Buscar objetos por tipo y/o por título</li> <li>Comprobar la disponibilidad de objetos</li> <li>Permitir 5 objetos por usuario simultáneamente</li> </ul>	<ul style="list-style-type: none"> <li>Se pueden encriptar las contraseñas para incrementar la seguridad del sistema. En este caso, las contraseñas no se deberían recuperar sino ser renovadas.</li> <li>El sistema debería ser fácil de usar (no está especificado en los requerimientos pero teniendo en cuenta que va a ser utilizado por personas de diferentes tipos, sería conveniente)</li> </ul>

**1.4.3.3.** Indica de qué tipo es cada uno de los siguientes requisitos (de sistema, funcional o no funcional):

*Solution:*

- La aplicación debe guardar los datos antes de cerrarse

Funcional

- El servidor de base de datos debe estar separado del de la aplicación

Sistema

- Si no hay conexión con el servidor remoto, se deberá mostrar un mensaje de error y cerrar la aplicación

Funcional

- El tiempo de respuesta deseable no debe exceder los 10 milisegundos

No funcional

- Se debe reservar el suficiente ancho de banda para enviar el vídeo en streaming en tiempo real

Sistema (Conexión de internet) y No funcional (suficiente ancho de banda)

- Se deben verificar que los datos introducidos por el usuario (nombre, DNI y teléfono) son correctos

Funcional