**1. Rendimiento**

* Respuesta a operaciones comunes (login, registro, consultas) ≤ **3 segundos**.
* Generación de reportes ≤ **10 segundos**
* Soportar al menos **150 usuarios**.

**2. Escalabilidad**

* La arquitectura debe ser **modular**, permitiendo añadir nuevos módulos (ejemplo: torneos internacionales, deportes nuevos).
* La base de datos debe almacenar al menos **5 años de información**.

**3. Usabilidad**

* La interfaz debe ser **intuitiva**, con menús claros e iconografía estándar.
* Fácil interacción con tareas ( login, crear jugador, programar partido) no deben requerir más de **3 pasos**.

**4. Usabilidad móvil**

* El sistema debe ser **responsive** en móviles, tablets y escritorios.
* No debe perder funciones al usarse desde un dispositivo móvil.

**5. Disponibilidad**

* El sistema debe estar disponible (24/7).
* Tiempo máximo de recuperación en caso de fallo: **2 horas**.

**6. Seguridad**

* Contraseñas cifradas con algoritmo de hash seguro (ej. bcrypt).
* Control de roles y permisos.
* Sesiones expiran tras **30 minutos de inactividad**.

**7. Mantenibilidad**

* Debe existir un **manual técnico** y un **manual de usuario**.
* Las actualizaciones deben poder aplicarse sin interrumpir el servicio más de **30 minutos**.

**8. Portabilidad**

* Compatible con navegadores modernos (Chrome, Edge, Firefox, Safari).
* Posibilidad de ejecución en servidores Windows, Linux o en la nube.
* Migración de base de datos entre gestores SQL sin modificaciones críticas.

**9. Confiabilidad**

* El sistema debe manejar errores sin perder datos críticos.
* En caso de fallo, se debe garantizar la **integridad de la información**.
* Los reportes deben mostrar datos consistentes incluso tras reinicios del sistema.

**10. Interoperabilidad**

* Posible integración con servicios externos (ej. Google Login,).
* Exportación de datos a formatos estándar (PDF, Excel).
* Importación de información desde archivos Excel/PDF.

**11. Soporte y documentación**

* El sistema debe contar con **documentación técnica** actualizada.
* Debe existir **guía de usuario final** clara y comprensible.
* Se deben registrar y versionar los cambios en un sistema de control (ej. Git).