

## Arquitectura

### 1. Cumplimiento de Requisitos

- ☐ La arquitectura aborda todos los requisitos funcionales especificados.
- ☐ El rendimiento, escalabilidad, seguridad y mantenibilidad están considerados.
- ☐ Cada requisito está mapeado a un componente arquitectónico claro.

En la siguiente tabla se le asigna a cada requisito uno o más componentes, para verificar que a cada requisito le corresponda un componente.

Requisito	Componente

### 2. Claridad y Comprensibilidad

- ☐ El documento utiliza lenguaje comprensible y evita tecnicismos innecesarios.
- ☐ Las secciones del documento no tienen contradicciones y son consistentes entre sí.

En la siguiente tabla se verifica que los diagramas sean claros.

Diagrama	
	<ul style="list-style-type: none"><li><input type="checkbox"/> Los diagramas están bien etiquetados</li><li><input type="checkbox"/> Los diagramas explican claramente las interacciones del sistema.</li></ul>

	<input type="checkbox"/> Los diagramas están bien etiquetados <input type="checkbox"/> Los diagramas explican claramente las interacciones del sistem
	<input type="checkbox"/> Los diagramas están bien etiquetados <input type="checkbox"/> Los diagramas explican claramente las interacciones del sistem
	<input type="checkbox"/> Los diagramas están bien etiquetados <input type="checkbox"/> Los diagramas explican claramente las interacciones del sistem
	<input type="checkbox"/> Los diagramas están bien etiquetados <input type="checkbox"/> Los diagramas explican claramente las interacciones del sistem
	<input type="checkbox"/> Los diagramas están bien etiquetados <input type="checkbox"/> Los diagramas explican claramente las interacciones del sistem
	<input type="checkbox"/> Los diagramas están bien etiquetados <input type="checkbox"/> Los diagramas explican claramente las interacciones del sistem

En la siguiente tabla se verifica que cada módulo tenga un nombramiento correcto.

Módulo	
--------	--

	<input type="checkbox"/> El nombramiento de los módulos representa claramente lo que son los módulos en términos del problema. <input type="checkbox"/> El nombramiento de los módulos no revela el funcionamiento interno de estos.
	<input type="checkbox"/> El nombramiento de los módulos representa claramente lo que son los módulos en términos del problema. <input type="checkbox"/> El nombramiento de los módulos no revela el funcionamiento interno de estos.
	<input type="checkbox"/> El nombramiento de los módulos representa claramente lo que son los módulos en términos del problema. <input type="checkbox"/> El nombramiento de los módulos no revela el funcionamiento interno de estos.
	<input type="checkbox"/> El nombramiento de los módulos representa claramente lo que son los módulos en términos del problema. <input type="checkbox"/> El nombramiento de los módulos no revela el funcionamiento interno de estos.
	<input type="checkbox"/> El nombramiento de los módulos representa claramente lo que son los módulos en términos del problema. <input type="checkbox"/> El nombramiento de los módulos no revela el funcionamiento interno de estos.
	<input type="checkbox"/> El nombramiento de los módulos representa claramente lo que son los módulos en términos del problema.

	<input type="checkbox"/> El nombramiento de los módulos no revela el funcionamiento interno de estos.
--	---

### 3. Estructura del Sistema

En la siguiente tabla se verifica por módulo que la estructura del sistema sea correcta.

Módulo	
	<input type="checkbox"/> Representa un ADT. <input type="checkbox"/> Tienen una función clara. <input type="checkbox"/> Las interfaces entre componentes están claramente especificadas, incluyendo protocolos y dependencias. <input type="checkbox"/> No tiene dependencias innecesarias.
	<input type="checkbox"/> Representa un ADT. <input type="checkbox"/> Tienen una función clara. <input type="checkbox"/> Las interfaces entre componentes están claramente especificadas, incluyendo protocolos y dependencias. <input type="checkbox"/> No tiene dependencias innecesarias.
	<input type="checkbox"/> Representa un ADT. <input type="checkbox"/> Tienen una función clara. <input type="checkbox"/> Las interfaces entre componentes están claramente especificadas, incluyendo protocolos y dependencias. <input type="checkbox"/> No tiene dependencias innecesarias.
	<input type="checkbox"/> Representa un ADT. <input type="checkbox"/> Tienen una función clara.

	<input type="checkbox"/> Las interfaces entre componentes están claramente especificadas, incluyendo protocolos y dependencias. <input type="checkbox"/> No tiene dependencias innecesarias.
	<input type="checkbox"/> Representa un ADT. <input type="checkbox"/> Tienen una función clara. <input type="checkbox"/> Las interfaces entre componentes están claramente especificadas, incluyendo protocolos y dependencias. <input type="checkbox"/> No tiene dependencias innecesarias.
	<input type="checkbox"/> Representa un ADT. <input type="checkbox"/> Tienen una función clara. <input type="checkbox"/> Las interfaces entre componentes están claramente especificadas, incluyendo protocolos y dependencias. <input type="checkbox"/> No tiene dependencias innecesarias.
	<input type="checkbox"/> Representa un ADT. <input type="checkbox"/> Tienen una función clara. <input type="checkbox"/> Las interfaces entre componentes están claramente especificadas, incluyendo protocolos y dependencias. <input type="checkbox"/> No tiene dependencias innecesarias.

#### 4. Cumplimiento de Estándares

- ☐ Se cumple con el estándar IEEE 1471 ISO/IEC 42010 .

#### 5. Escalabilidad y Flexibilidad

- ☐ Se usa un patrón de diseño de componentes

#### 6. Rendimiento y Eficiencia

- ☐ El diseño cumple con los requisitos de rendimiento.
- ☐ La arquitectura maneja eficientemente diferentes cargas de trabajo.

## **7. Seguridad**

- ☐ La arquitectura contempla medidas adecuadas para proteger contra amenazas y vulnerabilidades.
- ☐ Se han especificado claramente cómo se gestionan accesos y permisos de usuarios.
- ☐ La arquitectura incluye estrategias para mitigar riesgos de seguridad y responder ante incidentes.

## **8. Mantenibilidad y Evolución**

- ☐ La arquitectura está diseñada para facilitar el mantenimiento y corrección de errores.
- ☐ Existen procesos para manejar cambios en la arquitectura sin comprometer la estabilidad del sistema.
- ☐ Toda la documentación necesaria está completa y es accesible para futuros mantenedores.

## **9. Compatibilidad y Portabilidad**

- ☐ El sistema debe de ser compatible con distintos dispositivos con navegador web.

## **10. Consideraciones de Despliegue**

- ☐ La arquitectura incluye una estrategia clara para el despliegue del sistema en producción y preproducción.
- ☐ Se han considerado mecanismos de monitorización y recuperación ante fallos.