|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Checklist: Especificación de Arquitectura de Software** | | | | |
| 1. **Identificación del proyecto y del producto** | | | | |
| **Proyecto:** | | | | |
| **Producto:** | | | | |
| 1. **Verificado por:** | | | | |
| **Nombre y firma** | **Rol:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | | | |
| **e-mail:** |
| 1. **Checklist** | | | | |
|  | | **Sí** | **No** | **N/A** |
| **Adherencia** | |  |  |  |
| * El documento se adhiere a los estándares establecidos | |  |  |  |
| * El diseño fue desarrollado de acuerdo a las metodologías y técnicas predefinidas | |  |  |  |
| **Claridad** | |  |  |  |
| * El diseño representa claramente la arquitectura:   Vista de casos de uso  Vista lógica  Vista de implementación  Vista de despliegue  Vista de datos | |  |  |  |
| * Las vistas representan claramente flujos de datos, nomenclatura, nombres de forma clara | |  |  |  |
| * La terminología utilizada es consistente con la empleada por los desarrolladores | |  |  |  |
| **Completitud** | |  |  |  |
| * Se encuentran claros el propósito y alcance del documento de especificación de arquitectura de software | |  |  |  |
| * La vista de casos de uso representa las funciones que deben ser provistas por el software | |  |  |  |
| * La vista de diseño detallado contiene los módulos cubre completamente la funcionalidad de los requerimientos del software | |  |  |  |
| * En el diseño detallado se representa de manera correcta la interacción que se tiene entre los módulos así como mensajes de entrada, salida. | |  |  |  |
| * El diagrama de despliegue muestra de manera completa todos los elementos que se ocupan para montar el sistema de información en ambiente de operación. | |  |  |  |
| * El diagrama entidad relación representa todas las entidades, atributos, relaciones necesarias para el buen funcionamiento del sistema. | |  |  |  |
| * Se especifica un diccionario de datos con el nombre de los atributos y dominio correspondiente. | |  |  |  |
| **Consistencia** | |  |  |  |
| * Se utilizan consistentemente los nombres de los elementos de datos, procedimientos y funciones a lo largo de la descripción y representación del diseño | |  |  |  |
| * El diseño representa el hardware, el software y el entorno del sistema | |  |  |  |
| * El diseño es consistente con la especificación de requerimientos | |  |  |  |
| **Facilidad de pruebas** | |  |  |  |
| * Es factible probar, demostrar o analizar los diseños contenidos en la especificación de arquitectura de software | |  |  |  |
| * Es posible integrar y probar el software generado a partir de este diseño | |  |  |  |
| **Factibildad** | |  |  |  |
| * El diseño es factible según la calendarización, el presupuesto y la tecnología disponibles | |  |  |  |
| **Mantenibilidad** | |  |  |  |
| * El diseño es modular | |  |  |  |
| * Los módulos tienen alta cohesión y bajo acoplamiento | |  |  |  |
| **Trazabilidad** | |  |  |  |
| * Es posible trazar el diseño con los requerimientos | |  |  |  |

|  |  |
| --- | --- |
| **Resultados de la verificación:** | **Acciones a realizar:** |
|  |  |