**Ejercicio de Aplicación V&B y UML**

**Mecánica: realice los ejercicios individualmente y luego compártalos con sus compañeros de grupo y lleguen a un consenso de solución o respuesta. En algunos ejercicios se solicita que intercambien la solución acordada con otros grupos.**

1. La portabilidad de un sistema se define como la capacidad del mismo de soportar cambios en su plataforma de ejecución.

Al describir la arquitectura de un sistema con importantes requerimientos de portabilidad, ¿En qué vista debería hacer énfasis el arquitecto?

1. Vista de capas lógicas (módulos)
2. Vista de capas físicas (componentes y conectores)
3. Vista de descomposición (módulos)
4. Vista de despliegue (asignación)
5. El diagrama que se muestra a continuación representa la vista de descomposición (módulos) del sistema de gestión de recursos de una determinada empresa:



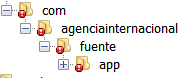
Una primera revisión del documento de arquitectura revela que esta vista de descomposición resulta demasiado extensa y compleja de entender para los distintos accionistas debido a que se incluyen demasiados elementos en la misma.

¿Qué debería hacer para comunicar la descomposición en módulos del sistema de forma efectiva?

1. Omitir todos los elementos (módulos) que se encuentran en el último nivel de la jerarquía. Por ejemplo: *uy.edu.ort.erp.facturacion.ui.web*.
2. Crear múltiples vistas de descomposición. Por ejemplo: “vista de descomposición del sistema”, “vista de descomposición del módulo de facturación”, “vista de descomposición del módulo de clientes”, etc.
3. Modificar el diseño del sistema “juntando” distintos paquetes o eliminando sub-paquetes.
4. Utilizar otra notación distinta a UML. Por ejemplo, notación informal.

**Ejercicio 1**

Dados los siguientes fragmentos de un proyecto en un IDE, transfórmelo en un diagrama de la vista de módulos utilizando la notación UML y el conector de nesting.



**Ejercicio 2**

Asumiendo que se está diseñando un sistema para la biblioteca. Dados los siguientes componentes

**2.1 Diagrama por separado cada uno de los componentes mostrando las interfaces provistas y requeridas**

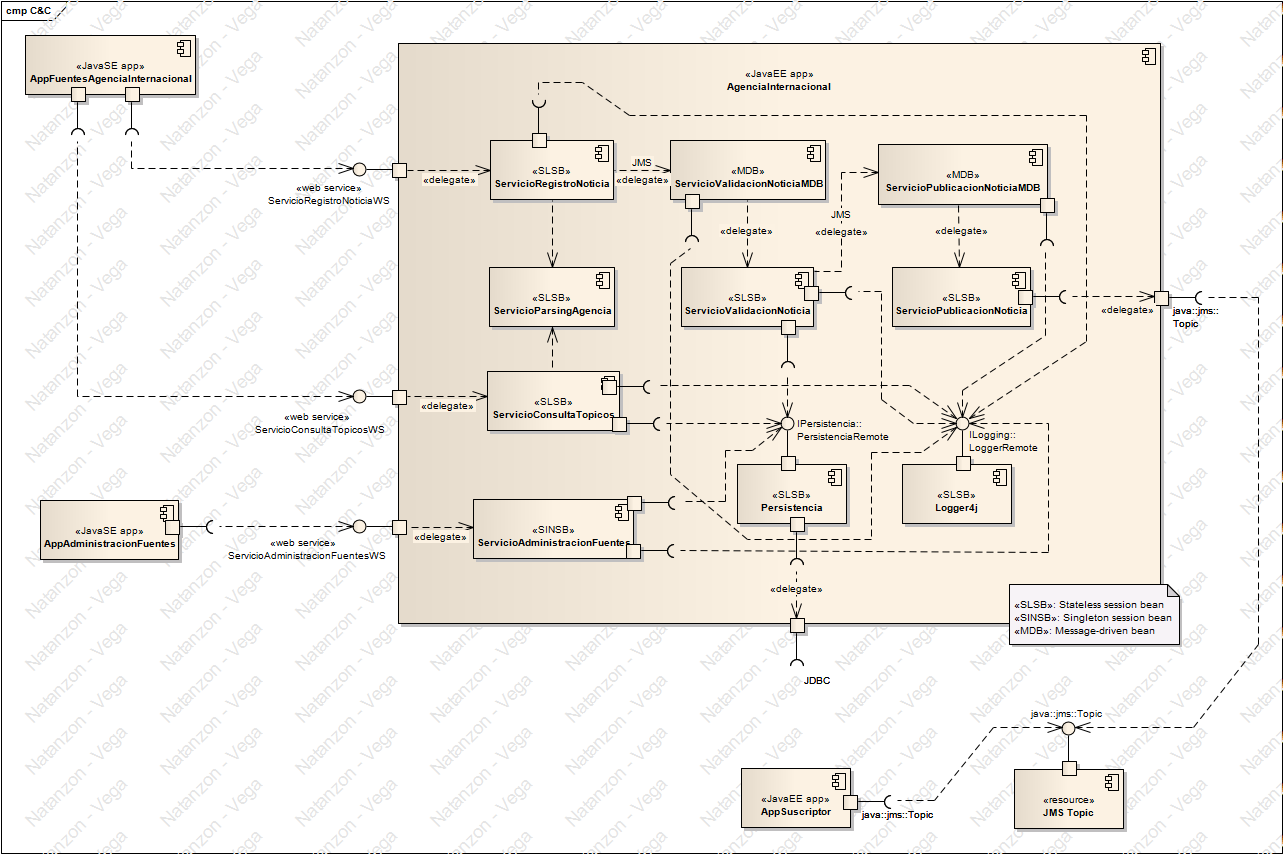
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Componente** | **Interfaz Provista** | **Descripción** | **Interfaz Requerida** |
| 1 | BuscadorCatalogo | IBuscarCat | Provee operaciones para buscar por criterios en el catálogo de publicaciones | IPubData |
| 2 | GestionUsuarios | IUsuarios | Provee operaciones para mantener los datos de los usuarios de la biblioteca | IUsuariosData |
| 3 | GestionPublicaciones | IPublicacion | Provee operaciones para mantener los datos de las publicaciones de la biblioteca | IPubData |
| 4 | Prestamos | IPrestamos | Provee operaciones para mantener los datos de los préstamos y devoluciones | IPrestamosData |
| 5 | AccesoDatos | IPubData | Provee operaciones de acceso a datos para las publicaciones |  |
| IUsuariosData | Provee operaciones de acceso a datos para Usuarios |  |
| IPrestamosData | Provee operaciones de acceso a los datos de los préstamos y devoluciones |  |
| 6 | AppBiblioUI |  |  | IBuscarCat |
|  |  |  | IUsuarios |
|  |  |  | IPublicacion |
|  |  |  | IPrestamos |

4.2 – Asumiendo que se va a diseñar un componente BiblioLogicaNegocio que es utilizado por AppBiblioUI y que utiliza al componente AccesoDatos realice un diagrama que muestre la relación entre los mismos. BiblioLogicaNegocio provee las interfaces de los componentes 1,2,3,4 y requiere las de 5.

4.3 – Realice un diagrama mostrando la estructura interna de BiblioLogicaNegocio. Agregue los puertos y conectores de delegación necesarios.

**Ejercicio 3**

Dado el siguiente diagrama en la notación UML.

****

**Justifique cada una de sus respuestas**

1.- ¿A qué vista del modelo Views And Beyond pertenece?

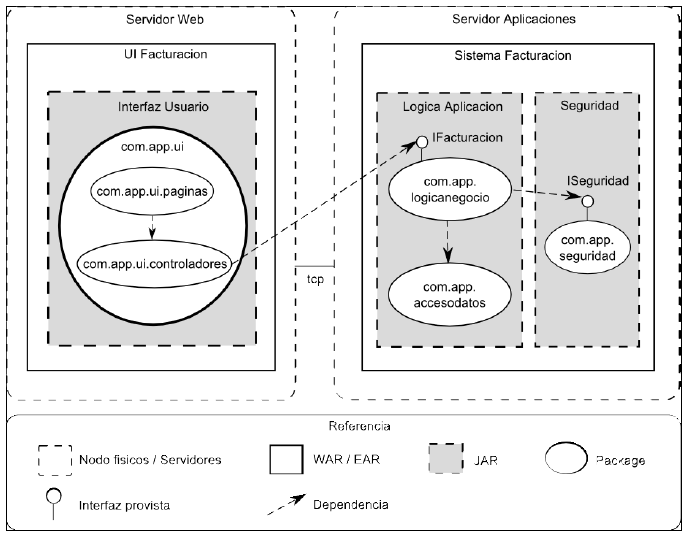
1. Vista de Módulos
2. Vista de Asignación
3. Vista de Componentes y conectores
4. Vista de Dependencias

2.- ¿para mostrar qué aspectos del sistema se utiliza este tipo de diagrama?

3.- Identifique los principales tipos de componentes y conectores

**Ejercicio 4**

El siguiente diagrama representa, utilizando notación informal, los elementos fundamentales de un determinado sistema de software:



Asumiendo que va a convertir la descripción anterior a una nueva en el enfoque de V&B y utilizando UML como notación. Realice los diagramas correspondientes con el grado de detalle necesario para que un tercero los comprenda.