

Escuela de Ingeniería

Examen de: Arquitectura de software

Código de materia: 3851/1927

Fecha: 14-8-6

Id Examen: 15701/15759

Hoja 1 de 3

1. Atributos, Escenarios y Tácticas

1.1) (10 puntos) Describa el concepto de **Escenario**. Adicionalmente, clasifique los siguientes **escenarios** en función del **atributo de calidad** al que corresponden indicando para cada uno: *Fuente, Estimulo, Artefacto, Respuesta y Medida*.

- *Un desarrollador desea cambiar el código de UI en tiempo de diseño, la modificación es realizada sin efectos secundarios en un tiempo de tres horas.*
- *Un formato de archivo (no anticipado) es recibido por un proceso del sistema encargado de leer una carpeta del S.O. El proceso audita el error e informa al operador mediante un mensaje. El sistema continúa operando archivos en forma normal.*

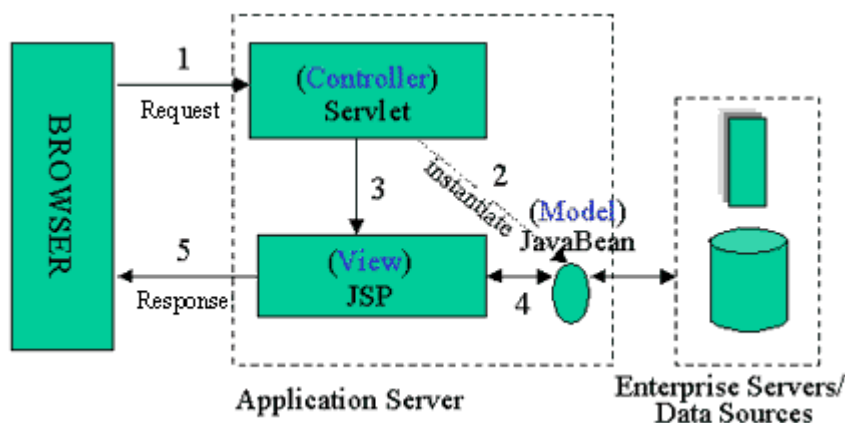
1.2) (15 puntos) Durante el diseño de la arquitectura de una solución el arquitecto del software tomó las siguientes acciones.

- Implementó un sistema en el cual los componentes que ejecutan en los dispositivos clientes (terminales móviles) envían al servidor un mensaje con el código "WRKING" cada 10 segundos.
- Adquirió un producto servidor de base de datos que cuenta con un componente primario que recibe las solicitudes, las procesa e informa a otros componentes redundantes sobre las solicitudes que recibió.
- Introdujo un componente que distribuye los mensajes que recibe el servidor entre distintas instancias del mismo.
- Introdujo copias de determinados datos - en medios rápidos - en los componentes cliente.

Se pide: Identifique (mediante su nombre) los mecanismos de arquitectura utilizados e indique para cada uno ante qué atributo de calidad el arquitecto se estaba enfrentando.

2. Patrones / Estilos

2.1) (10 puntos) Indique exactamente cuales son las **responsabilidades** de cada uno de los componentes del diagrama (Servlet, JSP y JavaBean) si el patrón arquitectónico es **MVC**.



Escuela de Ingeniería

Examen de: Arquitectura de software

Código de materia: 3851/1927

Fecha: 14-8-6

Id Examen: 15701/15759

Hoja 2 de 3

Observación: Nótese que no es necesario conocer la implementación de estos componentes web (Servlets, JSP) para responder la pregunta.

2.2) (15 puntos) Se le encomendó el diseño de un sistema en el cual es necesario contar con un conjunto de componentes que interactúen mediante eventos. Algunos componentes son productores de información y otros consumidores de información. Se deben poder asociar a los consumidores de información --en tiempo de ejecución-- a determinados tipos de eventos. Cuando se produzca un evento de ese tipo deben recibir notificaciones de estos eventos, también en tiempo de ejecución.

Se pide:

Describe el patrón o estilo más apropiado para este tipo de problema.

Diagrama su topología y describa las responsabilidades de los componentes.

3. Principios de Diseño de Componentes

3.1) (10 puntos) Defina el principio de diseño “**Dependencias a cíclicas**”.

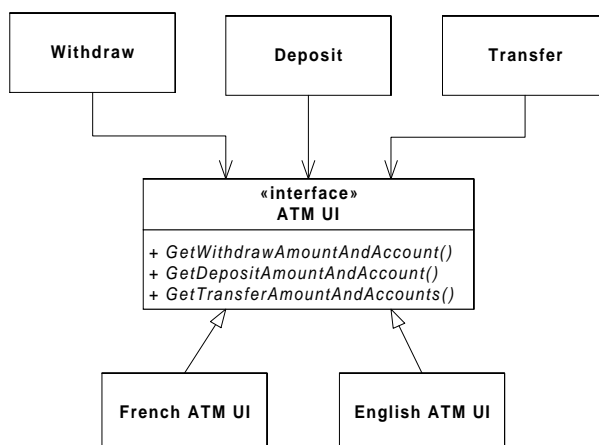
Se pide:

- Explique que tipos de problemas de diseño intenta resolver este principio.
- Explique detalladamente cómo se pueden eliminar los ciclos.

3.2) (10 puntos) El principio de segregación de interfaces se relaciona con las interfaces inapropiadas o “gordas”.

Se pide:

- Si las operaciones y la UI se empaquetaran en componentes independientes **Operación** (*Withdraw*, *Deposit*, *Transfer*) y **UI** (*ATMUI*, *FrenchATMUI*, *EnglishATMUI*), muestre un diagrama de UML completo que permita identificar las **interfaces provistas / requeridas** entre los componentes.
- Aplique el **principio de segregación** para la interface ATMUI desde la perspectiva de las entidades que la emplean.



Escuela de Ingeniería

Examen de: Arquitectura de software

Código de materia: 3851/1927

Fecha: 14-8-6

Id Examen: 15701/15759

Hoja 3 de 3

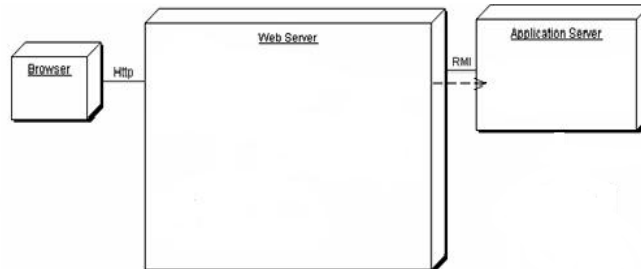
4. Enterprise Java Beans (20)

Dado el siguiente SessionBean se pide:

- 5.1) Bosqueje el código de las **interfaces home, remote** y anote el código necesario para invocar los métodos del Bean desde una clase Cliente.
- 5.2) Anote un **diagrama de clases** que considere todas las entidades de las partes anteriores.
- 5.3) Suponiendo que **SolicitudEmpleadoBean** esta distribuido en un nodo diferente al de los clientes que lo emplean ¿Cómo podría optimizar las llamadas a los métodos este Bean?
- 5.4) ¿En que nodo tiene sentido realizar el deployment del Componente (*Browser, WebServer, Application Server*)?

```
import javax.ejb.*;
public class SolicitudEmpleadoBean implements SessionBean {

    public String getNombre() {
        //.....
    }
    public String getApellido() {
        //.....
    }
    public int getNroEmpleado(int nro) {
        //.....
    }
    public void ejbActivate() {}
    public void ejbPassivate() {}
    public void ejbRemove() {}
    public void ejbCreate() {}
}
```



5. .NET Remoting (10)

- 5.1) Indique la diferencia entre **Marshal-by-Value**, y **Marshal-by-Reference** en .Net Remoting.
- 5.2) Mencione los posibles **canales** de comunicación que se pueden emplear entre los clientes y el servidor en .Net Remoting.

Duración:	3 horas
Con material:	No
Puntaje máximo:	100 puntos