

Preguntas de Diseño Arquitectónico (70 puntos)

Pregunta 1

(5 puntos) Explique detalladamente la definición de Arquitectura de Software propuesta por Len Bass vista en el curso. (Nota: Por “detalladamente” se entiende que desarrolle las partes clave de la definición).

Ejercicio 2

Una empresa del rubro de ventas de artículos por internet, debido al inesperado crecimiento que ha tenido en los últimos años, está comenzando a tener dificultades con su sitio web.

Dentro de los principales problemas que ha identificado se encuentran:

* El sitio deja de prestar servicios frecuentemente debido a fallas. El tiempo medio entre fallas es de 2 horas por mes y el tiempo medio de reparación de fallas es de 20 minutos. En general las fallas se deben a defectos en el código de las aplicaciones (lógica del negocio). Otro problema asociado a esto es que los operadores del sistema se enteran de la falla debido a los llamados o mails de los usuarios.

* El tiempo de espera por parte de los usuarios al realizar las compras es muy alto, en particular los viernes que son días de oferta lo cual ocasiona que los usuarios abandonen el sitio dejando compras sin completar. Incluso es frecuente que a pesar de que el sistema es funcionando el usuario reciba el mensaje de ERROR 503 SERVICE UNAVAILABLE cuando se superan los 3 segundos luego de realizar una acción de compra, lo cual es inaceptable. El tiempo promedio deseado para cualquier transacción de compra no deben superar el segundo en condiciones normales de operación y de 2 segundos en estado sobrecargado (situación que debe notificarse al usuario).

* Cada vez que es necesario arreglar el software debido a una falla es necesario corregir el código alojado en distintos paquetes a pesar de que solamente se está cambiando una funcionalidad. Por otra parte cada vez que se realiza un cambio es necesario recompilar y desplegar varios componentes.

El sistema de software tiene desplegados todos los componentes de la arquitectura en un solo procesador en el cual se alojan: el contenedor web con la lógica de presentación, el contenedor de EJB con la lógica del negocio y de acceso a datos y el manejador de base de datos (RDMS).

a) (4 puntos) Calcule α (%) en función de la formula que se presenta en el libro de texto.

b) (9 puntos) Identifique los atributos y/o subatributos de calidad que se están viendo afectados. Explique por qué los seleccionó.

c) (12 puntos) Para uno de los atributos identificados escriba al menos dos escenarios que describan el comportamiento esperado por el cliente. El formato debe incluir los elementos que considere necesarios (fuente, estímulo, artefacto, ambiente, respuesta y medida).

d) Para resolver los problemas que tiene el sistema actualmente

1. (12 puntos) Identifique y describa el estilo de arquitectura que recomendaría. Justifique su respuesta

2. (16 puntos) **Identifique y explique brevemente** las principales tácticas que recomendaría para resolver los problemas antes enumerados agrupándolas por atributo de calidad y enfoque táctico. Se deben describir al menos **4 tácticas**, cubriendo **todos los atributos de calidad identificados** y detallar solo **una** táctica por enfoque táctico.

Ejercicio 3

(12 puntos) Describa en forma concisa el patrón de arquitectura Model View Controller (MVC), incluyendo la siguiente información: objetivo, topología, comportamiento y qué beneficios provee.

Preguntas de Tecnología (30 puntos)

NOTA IMPORTANTE: Responde en hoja aparte

Pregunta 1 – Ejercicios JEE-EJB-JSF

Una empresa de Espectáculos lo contrata como Arquitecto de Software para realizar una aplicación Web para la compra y reserva de entradas para sus socios.

RF1-Login

El sistema debe permitir la autenticación de los usuarios para el ingreso al sistema.

RF2-Reserva de entradas

El sistema debe permitir la reserva de entrada para un espectáculo.
El usuario ingresa el número de socio, nombre del socio y selecciona el espectáculo que desea ver y realiza la reserva. El sistema la asigna un número de reserva.
La reserva caduca dos días antes del estreno del espectáculo

RF3-Compra de entradas

El sistema debe permitir la compra de entrada para un espectáculo.
El usuario ingresa el número de socio, nombre del socio y selecciona el espectáculo que desea ver y realiza la compra. El sistema la asigna un número de compra.

RF4-Cancelacion de reserva

El sistema debe permitir la cancelación de la reserva de la entrada.
El usuario ingresa el número de socio y el número de reserva y realiza la cancelación.

RF5-Ver reservas de socio

El sistema debe permitir el listado de las reservas de entradas de un socio para su cancelación o compra.

Se desea implementar la funcionalidad de reserva de entradas. Para esto se han construido los siguientes elementos:

Página reserva.jsp

```
<f:view>
  <html>
    <head>
      <meta http-equiv="Content-Type" content="text/html; charset=UTF-8"/>
      <title>JSP Page</title>
    </head>
    <body>
      <h1><h:outputText value="Realizar Reserva"/></h1>
      <h:panelGrid columns="2">
        <h:outputText value="Nro Socio"/>
        <h:inputText value="#{reservaMB.numeroSocio}" />
        <h:outputText value="Nombre Socio"/>
        <h:inputSecret value="#{reservaMB.nombreSocio}"/>
      </h:panelGrid>
      <h:dataTable id="espectaculosList" value="#{reservaMB.espectaculos}"
        var="espectaculo" rows="10" width="50%">
        <h:column>
          <f:facet name="header">
            <h:outputText value="Nombre Espectaculo"/>
          </f:facet>
          <h:outputText value="#{espectaculo.numeroEspectaculo}" />
        </h:column>

        <h:column>
          <f:facet name="header">
            <h:outputText value="Nombre Espectaculo"/>
          </f:facet>
          <h:outputText value="#{espectaculo.nombreEspectaculo}" />
        </h:column>

        <h:column>
          <f:facet name="header">
            <h:outputText value="dia"/>
          </f:facet>
          <h:outputText value="#{espectaculo.diaEspectaculo}" />
        </h:column>

        <h:column>
          <f:facet name="header">
            <h:outputText value="Lugar"/>
          </f:facet>
          <h:outputText value="#{espectaculo.lugarEspectaculo}" />
        </h:column>

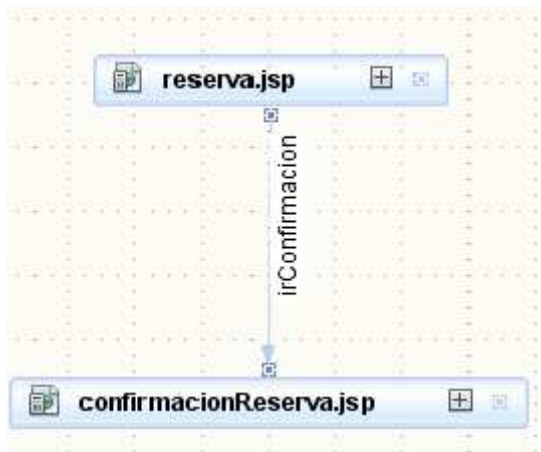
        <h:column>
          <f:facet name="header">
            <h:outputText value="horario"/>
          </f:facet>
          <h:outputText value="#{espectaculo.horarioEspectaculo}" />
        </h:column>

        <h:column>
          <f:facet name="header">
            <h:outputText value="costo"/>
          </f:facet>
```

```
</f:facet>
<h:outputText value="#{espectaculo.costoSpectaculo}"/>
</h:column>
</h:dataTable>
<h:commandButton value="Aceptar" action="#{reservaMB.aceptar}"/>

</body></html></f:view>
```

Navegación de la página reserva.jsp:



EJB con la lógica de negocio:

```
public interface IReserva{
    public void  realizarReserva (Reserva reserva);
}
public class ReservaSessionBean {
    public void realizarReserva (Reserva reserva);{
        // Logica de negocio
    }
}
```

Entidad Socio:

```
public class Socio{
    private int numeroSocio;
    private String nombreSocio;
    private String direccionSocio;
    private String teléfonoSocio;

    //getters y setters
}
```

Entidad Espectáculo:

```
public class Espectaculo{
    private int numeroEspectaculo;
    private String nombreEspectaculo;
    private Date diaEspectaculo;
    private String lugarEspectaculo;
    private String costoEspectaculo;
    //getters y setters
}
```

Entidad Reserva

```
public class Reserva{
    private Socio socioReserva;
    private Espectaculo espectaculoReserva;
    //getters y setters
}
```

Se pide:

1- (6 puntos) ¿Qué tipo/s de contenedor/es participa en la implementación del código anterior? Y describa algunos servicios que ofrece/en?

2-(6 puntos) Escriba la clase del Managed Bean (propiedades, métodos, etc.) asociado a la página. **Nota: No es necesario especificar el DataTable para el manejo del dataTable.**

3-(4 puntos) Describa los tags para la definición del Managed Bean en el archivo faces-config.xml.

4-(4 puntos) Describa los tags para la definición de la navegación en el faces-config.xml.

5-(6 puntos) Agregar las anotaciones necesarias para que los servicios que expone el SessionBean sean consumidos por el Managed Bean

Pregunta 2 – JMS y MDB (4 puntos)

1) Indique cuales de las siguientes opciones representan el modelo de programación de JMS:

- a) Connection Factory(crea)-> Connections(crea)-> Sessions(crea)-> Message Producers(envia)->Destinations
- b) Connection Factory (crea)-> Sessions(crea)-> Message Producers(envia)->Destinations
- c) Connection Factory (crea)-> Sessions(crea)-> Message Consumers (recibe)->Destinations
- d) Connection Factory (crea)->Connections(crea)->Sessions(crea)-> Message Consumers (recibe)->Destinations
- e) Connections(crea)-> Sessions(crea)-> Message Producers(envia)-> Destinations

Ingeniería

Examen de: Arquitectura de Software

Código de materia: 3851

Fecha: 15/02/12

Hoja 7 de 2

Posibles respuestas (solo 1 opción es correcta):

1-Opción A

2-Opción B

3-Opción C

4-Opción D

5-Opción E

6-Opción A y B

7-Opción A y C

8-Opción A y D

NOTA: se asume que el objeto Message esta creado

Duración: 3 horas

Con material:No

Puntaje máximo: 100 puntos