

# ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

FACULTAD DE INGENIERÍA EN ELECTRICIDAD Y COMPUTACIÓN

DESARROLLO DE APLICACIONES WEB

SEGUNDA EVALUACIÓN – TERMINO 2012-I -- 2012-08-30

Nombre: \_\_\_\_\_ Matrícula \_\_\_\_\_ Paralelo \_\_\_\_\_

## TEMA 1. (50 puntos)

Usted es parte del equipo de desarrollo de una empresa dedicada a la venta de soluciones Web y esta participando en el diseño de una aplicación web para una tienda de venta online. La aplicación consultara la base de datos de la tienda mostrando los catálogos en las páginas web y contara con un carrito de compras que le permitirá al usuario que ingresa al sitio seleccionar y agregar un ítem al carrito. Cuando el usuario haya completado los ítems que desea comprar, el sitio le permitirá realizar la compra en línea.

- a) **(5 pts.)** De que forma implementaría usted el carrito de compra. Justifique su respuesta.

---

---

---

- b) **(5 pts.)** Tomando en consideración que un usuario no logueado puede entrar al sitio y comenzar a llenar su carrito de compras. De que forma usted aseguraría que este logueado al momento de comprar. Justifique su respuesta.

---

---

---

- c) **(5 pts.)** Que haría usted si uno de los requerimientos del cliente es que el usuario que hizo login en la aplicación y cerro el navegador sin dar logout, este logueado la siguiente que ingrese al sitio. Justifique su respuesta.

---

---

---

- d) **(5 pts.)** De que forma registraría el total de visitas que ha tenido el sitio durante toda su historia, tenga en cuenta la persistencia de los datos?

---

---

---

- e) **(5 pts.)** En la sección de la aplicación que corresponde a la administración del sitio es necesario establecer un mecanismo para que las imágenes que son cargadas al sitio no sean tan pesadas; cabe indicar que el sitio siempre debe poner subir la imagen y no detener al usuario. Que mecanismo utilizaría para hacerlo?

---

---

---

- f) **(5 pts.)** Si necesita establecer parámetros de inicialización de la aplicación como: Pagina de inicio, cadenas de conexión a base de datos; cual sería su sugerencia para parte del equipo de diseño de la aplicación:

---

---

---

g) **(5 pts.)** Que tipo de arquitectura sugeriría utilizar para el desarrollo de la aplicación Web. Justifique su respuesta.

---

---

---

h) **(5 pts.)** El arquitecto de software del proyecto desea que implementen MVC. Por qué razón cree usted que el desea implementar este patrón de diseño? Con que tecnologías lo implementaría?

---

---

---

---

---

---

---

i) **(5 pts.)** Cuáles son las ventajas de las arquitecturas multicapas?

---

---

---

j) **(5 pts.)** Explique porque cuando se accede por primera vez a una aplicación web esta demora en iniciar.

---

---

---

---

---

k) **(2 pts. extras)** Si la tecnología a utilizar para páginas Web es JSP y el sistema de pago del bróker de tarjeta de crédito esta en plataforma .NET, cual sería la solución para poder integrar y comunicar ambas plataformas.

---

---

## TEMA 2. (50 puntos)

Usted como parte del grupo de desarrollo de la aplicación Web del Tema 1, usted deberá desarrollar los siguientes requerimientos:

- (20 ptos.)** Servlet MantenimientoCarrito:
  - Considere que este servlet será invocado mediante AJAX para ejecutar cualquiera de sus operaciones mediante POST
  - **(5 ptos.)** Como diseñador de una aplicación usted debe seleccionar que parámetros necesita para ejecutar las operaciones de mantenimiento.
  - **(5 ptos.)** Implementar la funcionalidad de agregar un ítem al carrito
  - **(5 ptos.)** Implementar la funcionalidad de remover un ítem del carrito.
  - **(5 ptos.)** Implementar la funcionalidad de actualizar la cantidad de ítems de un tipo en el carrito.
  - Para la implementación de este servlet considere que tiene disponible un Bean que se encuentra almacenado en sesión y contiene el carrito de compras.
  - **(2ptos. extras)** Modularizar llamando a métodos una clase de lógica de negocio que usted mismo cree.
- (20 ptos.)** Página Carrito.jsp

The screenshot displays a shopping cart interface. The main section, titled 'Items in Your Cart', lists two items: 'iPod touch 8 GB - Black' priced at \$199.00 and 'iPod nano 8GB - Blue - Personalized' priced at \$129.00. Each item has a quantity of 1 and options to 'Remove' or 'Save'. A 'Continue Shopping' button is located at the top right of the items list. Below the items, there is a section for 'Estimate tax and delivery dates' with a 'Ship To:' field and a 'ZIP Code' input. To the right, a 'Cart Summary' box shows a 'Total' of \$328.00, 'Free Shipping', and a 'Check Out Now' button. There are also 'Print' and 'Save' icons in the summary box.

- **(5 ptos.)** El desarrollo de esta pagina debe realizarse en JSP utilizando JSTL  
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>  
<%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt" %>  
<%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn" %>
  - **(10 ptos.)** Mostrar el listado de ítems a partir del bean de sesión que contiene la información del carrito de compras. Tenga en consideración que como parte del equipo existe un diseñador grafico que creara los estilos (clases) que usted utilice.
  - **(2 ptos.)** Generar las llamadas a los respectivos eventos javascript que harán las invocaciones Ajax para remover o actualizar ítems
  - **(3 ptos.)** Cuando el usuario de clic en el botón "Check out Now" debe invocarse al filtro verificaSesion
- (10 ptos.)** Filtro verificaSesion.jsp
    - Este filtro verifica si el usuario esta logueado o no.
    - En caso de estar logueado lo reenvia a la pagina <https://secure1.store.daw.com/ec/checkout>
    - En caso de no estar logueado lo reenvia a la pagina contrario lo reenvia a la pagina [https://secure1.store.daw.com/ec/sign\\_in](https://secure1.store.daw.com/ec/sign_in)
  - (3 ptos. extras)** Que haría usted para evitar que un usuario que no haya ingresado al sitio no pueda entrar directamente al url <https://secure1.store.daw.com/ec/checkout> y lo lleve directamente al url [http://store.daw.com/ec/sorry/session\\_expired](http://store.daw.com/ec/sorry/session_expired)

## Referencias

```
class ItemsBean {
    private ArrayList<Item> items;

    public void add(Item i) { /*...*/ }
    public void remove(Item i) { /*...*/ }
    public void update(Item I, int cantidad) { /*...*/ }

    // METODOS GETTER Y SETTER YA IMPLEMENTADOS
}

class Item {
    private int id;
    private int name;
    private int partNumber;
    private double cost;
    private int count;

    // METODOS GETTER Y SETTER YA IMPLEMENTADOS
}
```

```
@WebFilter(filterName = "Filtro", urlPatterns = {"/patron/*"})
public class Filtro implements Filter {

    public void doFilter(ServletRequest request, ServletResponse response,
        FilterChain chain)
        throws IOException, ServletException {...}

    @Override
    public void init(FilterConfig filterConfig) throws ServletException {...}

    @Override
    public void destroy() {...}
}
```

```
@WebServlet(name = "ServletEjemplo", urlPatterns = {"/ServletEjemplo"})
public class ServletEjemplo extends HttpServlet {

    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {...}

    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {...}

    /**...*/
    @Override
    public String getServletInfo() {...} // </editor-fold>
}
```

<%@page contentType="text/html" pageEncoding="UTF-8"%>