

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL
FACULTAD DE INGENIERÍA EN ELECTRICIDAD Y COMPUTACIÓN
DESARROLLO DE APLICACIONES WEB
TERCERA EVALUACIÓN – TERMINO 2012-II -- 2013-02-14

Nombre: _____ Matrícula _____ Paralelo _____

Usted es parte de una consultora que se encarga de realizar un Aplicación Web para la M.I. Municipalidad de Guayaquil. Dicha aplicación consiste en exponer algunos de los servicios a través del internet para que los ciudadanos del Cantón puedan utilizarlos de forma más rápida.

Para realizar alguna consulta en el sistema, los ciudadanos (usuarios) necesitan ingresar su código predial el cual es un numero compuesto por 7 secciones: Sector, Manzana, Lote, División, PHV, PHH y Numero (Siempre debe ser 1)

(10 puntos) Cree el formulario **index.jsp** que solicite el código predial y muestre la imagen del código de seguridad (**CodigoSeguridad.jsp**), el mostrado en la imagen se encuentra almacenado en la variable de sesión "CodigoSeguridad". Cuando se dé clic en el botón "Consultar" se deberá invocar al Servlet Ingreso que se encuentra mapeado en la raíz del sitio. Cabe señalar que el archivo CodigoSeguridad.jsp tiene como context-type "image/gif" y no text/html.

Sector	Manzana	Lote	División	Phv	Phh	Número
0 -	0 -	0 -	0 -	0 -	0 -	1

"Por favor digite el código de seguridad **34668**, antes de continuar"



Página 1 - Index.jsp

```
public class Predio implements java.io.Serializable {
    private int idpredio;
    private Propietario propietario;
    private short sector, manzana, lote, division, phh, phv;
    private short estado;
    private BigDecimal area;
    private Set actividades = new HashSet(0);

    public Predio() {
    }

    public Predio(int idpredio, short sector, short manzana, short lote, short
    division, short phh, short phv, short estado, BigDecimal area) { /*...*/ }

    public Predio(int idpredio, Propietario propietario, short sector, short
    manzana, short lote, short division, short phh, short phv, short estado,
    BigDecimal area, Set actividades){ /*...*/ }

    /** getters y setters */
}
```

(15 puntos) Cree el servlet **"ingreso.java"** que sea invocado cuando el usuario presionó el botón consultar. El servlet deberá validar el código de seguridad ingresado contra el código que se encuentra en la variable de sesión. Si el código es correcto cargue en un bean de sesión la información del predio y redirija el requerimiento a la página **SUS0001.jsp**. Caso contrario regréselo a **index.jsp**. Asuma que cuenta con la clase PredioBusiness Logic la cual expone el método "public Predio cargarPredio(String codigoPredial)", el mismo que devuelve un objeto de tipo Predio con la información de su propietario y un Set de Actividades que puede desarrollar. Adicionalmente, cree en el archivo web.xml la configuración del servlet ingreso.

(10 puntos) El cliente quiere que el usuario de la aplicación Web no pueda entrar a la página **SUS0001.jsp** a menos que este haya pasado por la página **index.jsp**. Implemente el mecanismo necesario y configúrelo en el archivo web.xml.



(15 puntos) Usando JSTL implemente la página **SUS0001.jsp** en la cual se presentara la información del predio leyendo el bean de sesión que contiene la información del mismo. Debera mostrar los datos del propietario del predio asi como el listado de actividades asociadas al mismo.

Considere para su implementación la función javascript **"mostrarRequerimientos (idActividad, idPredio)"**, la cual es invocada cuando el usuario de clic en una actividad. Adicionalmente, considere la inclusión del archivo **cabecera.jsp** el cual genera el publicidad y menú que se encuentran en la parte superior de la página.

Incluya la referencia al archivo daw.js, el mismo que contiene las funciones java script. Asuma que cuenta con el archivo daw.css que le da estilo a su pagina.

Listado de Requerimientos para 'Academia de Idiomas - Lenguas'	
Tipo	Requerimiento
Condición Edificación	Edificación combinada con otras actividades
Emissiones	NO Requiere Pronunciamento de Medio Ambiente
Parqueos	Para edificaciones nuevas y edificaciones existentes que cambien de uso residencial a uso comercial será lo determinado en el Registro de Construcción
	Edificaciones existentes será de acuerdo a lo que determine la DUAR en función de la ubicación.

(10 puntos) Implemente la función javascript **mostrarRequerimientos**, la misma que recibe por parámetro el identificador de una actividad y el identificador de un predio. La función realiza por medio de AJAX una llamada al servicio **"ConsultarActividades.aspx"**, el cual recibe los parámetros idActividad y idPredio por método POST. Adicionalmente, invoque a la función javascript **procesarRequerimientos**.

(5 puntos) Con los siguientes datos genere un ejemplo de un archivo XML que cumpla con el XSD:

- Predio: 90-100-10-0-0-0-1
- Actividad: Academia de idiomas – Lenguas
- Requerimientos
 - Condición Edificación: Edificación combinada con otras actividades.
 - Emissiones: NO Requiere Pronunciamento de Medio Ambiente
 - Parqueos: Para edificaciones nuevas y edificaciones existentes que cambien de uso residencial a uso comercial será lo determinado en el Registro de Construcción.

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema>
  <xs:element name="requerimientos">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="actividad" />
        <xs:element name="predio" />
        <xs:element maxOccurs="unbounded" name="requerimiento">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="tipo" />
              <xs:element name="descripcion" />
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

(10 puntos) Procese la respuesta del servicio y muéstrela en el div **"divRequerimientos"** el cual debe ser mostrado luego de cargar la información. Cabe mencionar que la respuesta del servicio es un archivo xml que cumple con el archivo XSD.

(5 puntos) Durante la fase de pruebas el cliente nota que cuando nadie está en el sitio y entra a la aplicación la página demora en cargar y le pide una explicación.

(5 puntos) ¿Qué sucede si usted se en el lugar de guardar la variable que almacena la información del predio en una variable de sesión la almacena en una variable de aplicación? Detalle su respuesta.

(5 pts.) En el uso de CSS, bajo qué condiciones o requerimientos usaría display:none en lugar de visibility:hidden.

(5 pts.) ¿Qué es un postback?

(5 pts.) ¿Por qué utilizar AJAX?

Referencias

```
@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"%>

```
public class Actividad implements java.io.Serializable {
    private short codigo;
    private String descripcion;
    private Set predios = new HashSet(0);

    public Actividad(){/*...*/ }
    public Actividad(short codigo, String descripcion) {/*...*/ }
    public Actividad(short codigo, String descripcion, Set predios) {/*...*/ }
    /*** getters y setters ***/
}

public class Propietario implements java.io.Serializable {
    private int idpropietario;
    private String nombre, identificacion, direccion;
    private Set predios = new HashSet(0);

    public Propietario(){/*...*/ }

    public Propietario(int idpropietario) {/*...*/ }
    public Propietario(int idpropietario, String nombre, String identificacion,
        String direccion, Set predios) {/*...*/ }
    /*** getters y setters ***/
}
```