

## Laboratorio Desarrollo de aplicaciones GeneXus 2011

Una empresa multinacional de alquiler de autos decide ofrecer sus servicios de renta y consultas a través de un sistema Web. Se busca que cualquier cliente registrado pueda consultar la flota de automóviles que ofrece la empresa, y de quererlo realizar la reserva del coche. Entonces se quiere un sistema que requiera autenticación, en caso de que el cliente no esté registrado se le dará la opción de hacerlo. Si el usuario no se autentica entonces no podrá acceder a ninguna funcionalidad del sistema y este lo redirigirá automáticamente a la pantalla de login.

### *Cientes:*

Al registrarse, el usuario deberá elegir un Nick el cual debe ser único. Deberá empezar con una letra, no podrá contener espacios y solo puede estar formado por letras, números y guión bajo. Además deberá ingresar obligatoriamente nombres, apellidos, país, ciudad, dirección e email de contacto. También ingresa la contraseña y una casilla extra para repetir la contraseña así se puede verificar que fue escrita correctamente. Al ingresar el País mediante un ComboBox automáticamente se muestran solo las ciudades de ese país en el comboBox de ciudades.

Luego de realizado el login se muestra una pantalla en la que se muestra algún promoción de la empresa. La promoción consta de algún modelo de la flota y una descripción de Marketing. En todo momento debe visualizarse el usuario conectado y el rol (por ejemplo en la Master Page).

### *Vehículos:*

Los vehículos serán rentados por la empresa a los clientes. Se cuenta con varios vehículos de cada modelo. Cada Modelo está identificado por la marca y el nombre del modelo (p.e Ford, Escort). Deberá existir una pantalla para administrar las marcas. De cada modelo se conoce una foto, una descripción breve del modelo, cilindrada, tipo de motor (gasoil o nafta), el costo de alquiler por día, si tiene dirección hidráulica, si tiene aire acondicionado y capacidad. También se desea visualizar la cantidad de vehículos en la flota que son de ese modelo,

Un vehículo está identificado por la matrícula y se conoce su modelo. Además es necesario registrar la cantidad de kilómetros que tiene, color, si está en renta o no, y si está en el taller o no.

### *Renta:*

Cuando un cliente desea rentar un auto realiza una búsqueda de modelos de autos. Debe poder realizar búsquedas pudiendo filtrar por al menos marca, modelo, capacidad y precio. De los modelos que cumplan las condiciones de búsqueda se debe mostrar foto del modelo, nombre del modelo, marca, una descripción breve del modelo, cilindrada, tipo de motor (gasoil o nafta), el costo de alquiler por día, si tiene

dirección hidráulica, si tiene aire acondicionado y capacidad. Para esto utilizar una *Free Style grid*. Junto a cada modelo debe aparecer la opción de rentar.

Al seleccionar la opción se redirige a una pantalla en la cual se muestran otra vez los datos del modelo seleccionado y se pide al usuario que ingrese la fecha desde la cual quiere alquilar y la cantidad de días. A través de un botón se muestra una grilla con los vehículos disponibles del modelo seleccionado y el costo total de la renta. El usuario elige un vehículo y se emite un listado pdf que funciona como comprobante en el que se muestra los datos del cliente, la fecha de la reserva, la cantidad de días, el total a pagar y los datos del vehículo.

Al momento de confirmar se ingresa la reserva del vehículo. Una reserva está identificada por el vehículo, y la fecha de la reserva (no es posible reservar un vehículo más de una vez en un día). La reserva tiene además la cantidad de días que se alquiló, el kilometraje que se hizo durante el alquiler, el cliente que hizo la reserva y el estado (abierta, cerrada). Al ingresar la reserva, esta queda abierta

Luego, cuando el cliente retorna con el vehículo un usuario administrador ingresa los kilómetros realizados y marca la reserva como cerrada.

## Objetivo:

Implementar la seguridad requerida para los clientes. Esto incluye mostrar solo las opciones de menú que corresponden.

Implementar para el cliente las funcionalidades de:

- Buscar Modelo
- Rentar auto
- Ver las reservas realizadas.

Implementar para el administrador las funcionalidades de:

- Reporte de estado de vehículos (permitir filtros por modelo)
- Ver reservas pendientes
- Reporte de historial de reservas de un vehículo.
- Reporte de reservas por rango de fechas. Agrupar por fecha los totales de las reservas de cada modelo (solo los modelos que efectivamente tuvieron reservas en el rango de fecha)

## Consideraciones:

1) Instalar el User control MultiLevelMenu que viene en el archivo MultiLevelMenu.zip:

<http://wiki.gxtechnical.com/commwiki/servlet/hwiki?Default+Installation+Instructions+for+User+Controls>,

2) Luego, y solo luego de instalar el *user control*, cargar el *XPZ laboratorioGX2011.xpz* que contiene la *masterPage* y el tema a utilizar para el laboratorio.

Deberá modificarse el *data provider* *dataProviderMLM* para cargar las opciones del menú de la aplicación y establecer como *MasterPage* y tema por defecto los del XPZ (el tema es K2b3).



3) Para el manejo de la seguridad utilizar el tipo de dato *WebSession*. Este tipo de dato permite almacenar pares {etiqueta, valor} a lo largo de la sesión del usuario. Por ejemplo para la seguridad se puede utilizar la etiqueta 'userid' para mantener el id del usuario, entonces al momento de hacer el login guardamos este valor utilizando el método *set* del tipo de dato *webSession*:

```
&Session.set('userid', <NombreDeUsuario>)
```

Luego en el evento *Start* de la *masterPage* verificamos que este valor exista de lo contrario redirigimos al usuario a, por ejemplo, la pantalla de login:

```
&Usuario = &Session.get('userid')
```

```
If &usuario.isEmpty()  
    Login.call()
```

<http://wiki.gxtechnical.com/commwiki/servlet/hwiki?WebSession>,

## Inscripción:

El laboratorio se realizará en grupos de hasta 3 personas. Enviar correo a [ernestolopezmoreira@gmail.com](mailto:ernestolopezmoreira@gmail.com) con los nombres de los integrantes antes del 30 de octubre.

## **Fechas de entrega y evaluación:**

Enviar a [ernestolopezmoreira@gmail.com](mailto:ernestolopezmoreira@gmail.com) un archivo comprimido de nombre *labGXGr\_XX.zip* (donde XX es el número de grupo), conteniendo un XPZ de nombre *labGXGrXX.XPZ* con la solución del laboratorio y un documento de nombre *labGXGrXX.doc* con una descripción breve de la solución implementada. No está permitido intercambiar objetos GeneXus entre los grupos y aquellos casos en los que se detecten objetos copiados serán pasibles de sanciones.

La entrega debe realizarse hasta las 23:59 del día 23 de noviembre.

## **Referencias:**

*Search de GeneXus:* <http://www.gxtechnical.com/gxsearch>

*Wiki de GeneXus:* <http://www.gxtechnical.com/wiki/>

*Grupo google del curso:* <http://groups.google.com.uy/group/desarrollo-de-aplicaciones-con-genexus-tecnologo/>