



Tutorial de desarrollo de Aplicaciones Web con XUL

Aprende a Desarrollar Aplicaciones web fácil y rápidamente con la Potencialidad que posee XUL (XML User Interface Language).

- *Que es esto de XUL...*
- *Como podemos crear una aplicación web por medio de xul?*

Que Aplicamos al hablar de XUL.

- Desarrollo de aplicaciones de interfaz de usuario en la web.
- Archivos XML, CSS, SVG..
- Mozilla Browser, Mozilla Firefox Browser, Mozilla Thunderbirds.
- Simplicidad en el desarrollo de alguna aplicación de interfaz de usuario.
- Extensiones de Firefox, Temas de Firefox.
- Soporte Multiplataformas.

```
<?xml version="1.0"?
```

```
<swi:tree title="XUL Nested Trees" xmlns="http://www.mozilla.org/1999/xhtml" xmins="http://www.mozilla.org/keymaster/gatekeeper/there.is.only.xul">
```

```
<script>
```

```
function setText(treeID, textID, val)
```

```
var tree = document.getElementById(treeID);  
var selection = tree.contentView.getItemAtIndex(tree.currentIndex);  
var text = selection.firstChild.firstChild.getAttribute("label");  
document.getElementById(textID).value = text;
```

```
</script>
```

```
<description><html:h1>XUL Nested Trees</html:h1></description>
```

```
<hbox flex="1">
```

Para que fue creado XUL.

- XUL fue construido para el desarrollo de Mozilla Browser.
 - Para su facilidad y rapidez.
- Nos permite ahorrar dinero y tiempo en el desarrollo de aplicaciones para multi-plataforma.
- XUL es un lenguaje diseñado para hacer interfaces de usuario portables.
- XUL lleva la ventaja de otros lenguajes XML.
- Principalmente fue para el desarrollo de aplicaciones de interfaz de usuario fuera rápido y expedito.

Lo destacable de XUL.

- XML User Interface Language
 - Lenguaje Multi plataforma.
 - Diseñado para ser portable.
 - Describe aplicaciones de interfaz de usuario.
 - Extensiones de Firefox.
- Ayudo a construir el navegador Mozilla Browser.
 - En el Desarrollo de Aplicaciones de interfaz de usuario.
- Tenemos el desarrollo de Firefox Browser o Thunderbirds
 - Extensiones, Temas (skin).
- XUL
 - Esta hecho en XML lo cual es portable todo lo de XML, lo cual tiene soporte a otras tecnologías.

¿ Donde podemos Ejecutar XUL ?

Entra en juego Mozilla Firefox Browser.

- Nos permite el uso del navegador como estación de trabajo para probar y ejecutar nuestra interfaz de usuario.
- Nos da la posibilidad de poder ocupar Extensiones para el Desarrollo de Aplicaciones en XUL(!).
- Desarrollar en Extensiones!
- Desarrollar en Temas.
- Utilizar el Systema de Chrome.

```
//>  
</script>  
  
<description><html:h1>XUL Nested Trees</html:h1></description>  
  
<hbox flex="1">
```

Dirección Chrome.

- Que es el Chrome (Mozilla System Chrome).
- Es un árbol de directorio donde guardamos nuestra extensión en XUL.
- hace una referencia de un archivo xul.
- Crea un vinculo con el Mozilla System chrome.
- Permite que nuestro navegador pueda encontrar nuestro programa.
- Posee una orientación jerárquica de directorios.
- Creación de enlaces por medio de chrome a nuestro archivo.xul entre otros.

Estructura de directorios XUL.

- Para tener nuestro directorio CHROME Estandar, primero debemos crear un directorio (De nuestro proyecto) con el nombre de nuestro programa, luego crear un subdirectorio llamado chrome.
- Dentro de ese subdirectorio chrome, podemos tener otros subdirectorios los cuales pueden ser.
 - Content
 - Skin
 - Locale

```
</script>  
<![CDATA[
```

```
function setText(treeID, textID, val)
```

```
var tree = document.getElementById(treeID);
```

```
selection = tree.contentView.getItemAtIndex(tree.currentIndex);
```

```
var text = selection.firstChild.firstChild.getAttribute("label");
```

```
document.getElementById(textID).value = text;
```

```
]
```

```
]]>
```

```
</script>
```

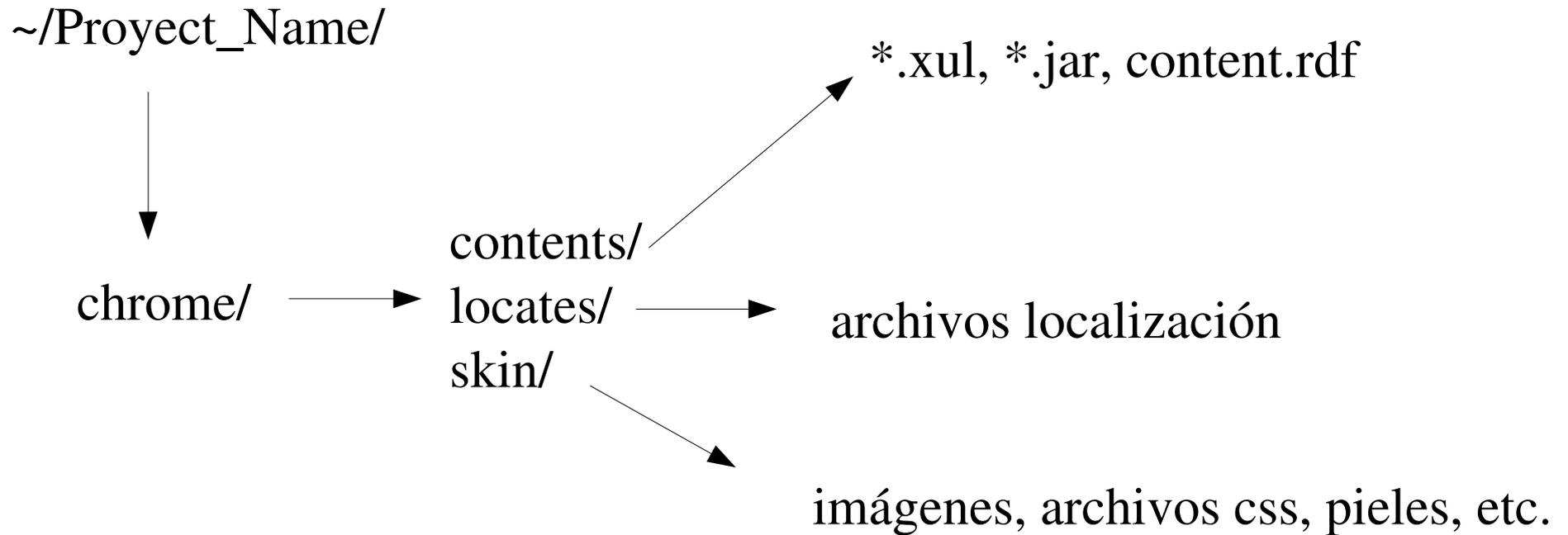
```
<description><html:h1>XUL Nested Trees</html:h1></description>
```

```
<hbox flex="1">
```

Descripción de directorios XUL.

- Cada subdirectorio necesita una referencia la cual la damos con un archivo llamado content.rdf.
- Subdirectorio Content.
 - Contiene los Script de la ventana.
 - Los Archivos con Extension .XUL
 - Archivos con Estención JS o JavaScript.
- Subdirectorio Skin.
 - Contiene las imágenes que se utilizan
 - Algún archivo de Temas específico.
- Subdirectorio Locate.
 - Los archivos de localización para el Idioma.

Rama de directorios en XUL.



Descripción básica de un RDF.

- Resource Description Framework.
 - Descripción de los paquetes de nuestro programa por medio de algún content.rdf
 - esto esta basado en XML.
- Ejemplo install.rdf del chrome del programa en xul.
 - describe las características del programa (versiones que acepta, nombre de la aplicación, directorio chrome donde se ubica, etc)
- Nos permite hacer localizaciones(locate).
- veamos un ejemplo ----

```
<?xml version="1.0" ?>
```

```
<?xml-stylesheet href="chrome://global/skin" type="text/css" ?>
```

```
<window title="XUL Nested Trees"
```

```
xmlns="http://www.mozilla.org/keymaster/gatekeeper/there.is.only.xul">
```

```
</window>
```

```
</script>
```

```
</script>
```

```
var selection = tree.contentView.getAttributeIndex(tree.currentIndex)
```

```
document.getElementById(textID).value = text;
```

```
</script>
```

```
</script>
```

```
<h1>XUL Nested Trees</html.h1></description>
```

```
<hbox flex="1">
```

Ejemplo del content.rdf

- content.rdf
- Syntax básica
- Ejemplo..

```
<?xml version="1.0"?>
<?xml-stylesheet href="chrome://global/skin" type="text/css"?>
<window title="XUL Nested Trees"
  xmlns:html="http://www.w3.org/1999/xhtml"
  xmlns:rdf="http://www.mozilla.org/keymaster/gatekeeper/there.is.only.xul">
  <script>
    function setText(treeID, textID, val)
    {
      var tree = document.getElementById(treeID);
      var selection = tree.contentView.getItemAtIndex(tree.currentIndex);
      var text = selection.firstChild.firstChild.getAttribute("label");
      document.getElementById(textID).value = text;
    }
  </script>
  <description><html:h1>XUL Nested Trees</html:h1></description>
  <hbox flex="1">
```

Ejemplo de install.rdf

- install.rdf
- Características del install.rdf
- Syntax..
- Ejemplo..

```
<?xml version="1.0"?>
<?xml-stylesheet href="chrome://global/skin" type="text/css"?>
<window title="XUL Nested Trees"
  xmlns:html="http://www.w3.org/1999/xhtml"
  xmlns:chrome="http://www.mozilla.org/keymaster/gatekeeper/there.is.only.xul">
  <script>
    [CDATA[
      function setText(treeID, textID, val)
      {
        tree = document.getElementById(treeID);
        var selection = tree.contentView.getItemAtIndex(tree.currentIndex);
        var text = selection.firstChild.firstChild.getAttribute("label");
        document.getElementById(textID).value = text;
      }
    ]>
  </script>
  <description><html:h1>XUL Nested Trees</html:h1></description>
  <hbox flex="1">
```

Herramientas para Desarrollar en XUL

- Primero que nada necesitamos un editor de texto: esto lo dejamos a tu decisión para no evitar problemas.
- Vamos a utilizar la Extensión “Extension Developer Extension”.
 - Editor de javascript en RealTime.
 - Editor XUL en RealTime.
 - Launch Extension Builder.
 - Reload the CHROME.
- XUL : Referencias para los Programadores (doc).
- XUL referencias de los elementos.
- JavaScript para controlar los eventos por medio de funciones.

Uso de la herramienta Zip.

- Nos permitirá comprimir y descomprimir nuestro paquete creado.
- También ayuda a agrupar los directorios dentro de chrome en el paquete que vamos a crear *.xpi
- Herramienta fundamental para el trabajo o desarrollo de xul.
- veamos un ejemplo...

```
<?xml version="1.0"?>
<?xml-stylesheet href="chrome://global/skin" type="text/css"?>
<window title="XUL Nested Trees"
  xmlns="http://www.mozilla.org/keymaster/gatekeeper/there.is.only.xul">
  <||CDATA[
function setTexttreeID(textID,val)
{
  var tree = document.getElementById("tree");
  var selection = tree.contentView.getItemAtIndex(tree.currentIndex);
  var text = selection.firstChild.firstChild.getAttribute("label");
  document.getElementById(textID).value = text;
}
  ]>
</script>
<description><html:h1>XUL Nested Trees</html:h1></description>
<hbox flex="1">
```

Características de los Elementos de XUL

```
<?xml version="1.0"?>
<?xml-stylesheet href="chrome://global/skin" type="text/css"?>
```

- Toda cabecera de un archivo con extensión .xul necesita los namespace para comenzar, según en xul necesitamos la mismas cabecera de un archivo XML.
- Todos los widget que se encuentren en el tab de <window> los cuales Heredan Gran parte de los atributos jerarquizados de los widget superiores.
- La mayoría de los widget terminan con el tab />
- Vamos a ver un archivo con extensión .xul..

```
</CDATA>
<script>
var selection = tree.contentView.getItemAtIndex(tree.currentIndex);
selection.firstChild.getAttribute("label");
document.getElementById(textID).value = text;
</script>
<description><html:h1>XUL Nested Trees</html:h1></description>
<hbox flex="1">
```

Como trabajar con XUL

- Que tipo de Interfases se usuario se pueden crear con XUL ?
- Características nuevas de las interfases de usuarios mas nuevas.
- Widget Tools.
- Algunos elementos que se pueden crear con XUL
 - TextBox.
 - CheckBox.
 - Slippers.
 - Toolbars.
 - Menu bar.
 - Tabs...

Estructura de una ventana básica en XUL

- `<?xml version="1.0"?>`
`<?xml-stylesheet href="chrome://global/skin/" type="text/css"?>`
`<window id="_Identificador"`
`xmlns="`
`http://www.mozilla.org/keymaster/gatekeeper/there.is.only.xul">`
- `</window>`
- Para comprobar, necesitamos salvar un archivo con la extensión .xul
- Abrir el archivo con Firefox browser.
- Comprobando con la herramienta de extensión.

Algunos Widget de XUL

Cada widget posee atributos o propiedades.

- Layout
- Button
- Grids
- Imágenes
- Labels
- Lists
- Menubar
- Tabs
- Popus
- Progressmeters
- Radiobuttons
- Scrollings
- Splitters
- Textboxes
- Tree view Simples, Nested

Ejemplo de atributos de un widget.

Conozcamos para que sirven los atributos que se añaden a los widget.

- VBox (id, flex..)

- Button

- Textboxes

```
<?xml version="1.0"?>
<?xml-stylesheet href="chrome://global/skin" type="text/css"?>
<window title="XUL Nested Trees"
  xmlns:html="http://www.w3.org/1999/xhtml"
  xmlns="http://www.mozilla.org/keymaster/gatekeeper/there.is.only.xul">
  <script>
    function setText(treeID, textID, val)
    {
      var tree = document.getElementById(treeID);
      var selection = tree.contentView.getItemAtIndex(tree.currentIndex);
      var text = selection.firstChild.firstChild.getAttribute("label");
      document.getElementById(textID).value = text;
    }
  </script>
  <description><html:h1>XUL Nested Trees</html:h1></description>
  <hbox flex="1">
```

Agrupando nuestros Widget.

Primero necesitamos orientación a nuestros widgets, la cual necesitamos en los siguientes widget.

- Hbox :
- VBox :
- groupbox
- vamos a ver un ejemplo...

```
<?xml version="1.0"?>
<description><html:head><title>XUL Nested Trees</title></html:head>
<html:html xmlns:html="http://www.w3.org/1999/xhtml"
  xmlns="http://www.mozilla.org/keymaster/gatekeeper/there.is.only.xul">
</script>
<![CDATA[
function setText(treeID, textID, val)
{
  var tree = document.getElementById(treeID);
  var selection = tree.contentView.getItemAtIndex(tree.currentIndex);
  var text = selection.firstChild.firstChild.getAttribute("label");
  document.getElementById(textID).value = text;
}
]]>
</script>
</description><html:h1>XUL Nested Trees</html:h1></description>
< hbox flex="1">
```

Eventos en los elementos de XUL

- Como podemos agregar eventos a nuestra ventana en XUL.
- Entra en Juego
 - JavaScript.
- Vamos a ver ejemplos básico de script en un archivo xul.

```
<?xml version="1.0"?>
<?xml-stylesheet href="chrome://global/skin" type="text/css"?>
<window title="XUL Nested Trees"
  xmlns:html="http://www.w3.org/1999/xhtml"
  xmlns="http://www.mozilla.org/keymaster/gatekeeper/there.is.only.xul">
  <script>
    <![CDATA[
function setText(treeid, textID, val)
{
  var tree = document.getElementById(treeid);
  var selection = tree.contentView.getItemAtIndex(tree.currentIndex);
  var text = selection.firstChild.firstChild.getAttribute("label");
  document.getElementById(textID).value = text;
}
    ]]>
  </script>
  <description><html:h1>XUL Nested Trees</html:h1></description>
  <hbox flex="1">
```

Empaquetar nuestro ejemplo. [xpi]

- Vamos a una terminal.
- Necesitamos crear un script completamente básico en este caso en linux dentro del directorio de nuestro ejemplo.
- Empaquetando los directorios junto con los archivos.
- Primero partimos empaquetando el contents, skin, locales, etc en un .jar
- Luego empaquetamos el chrome junto con los archivos install.rdf en un .xpi + el .jar
- Simplemente eso se puede hacer por medio de un básico script

Instalación del Ejemplo

- vamos a instalar nuestro ejemplo en el navegador Mozilla Firefox Browser.
- ubicamos nuestro ejemplo XPI generado.
- Lo abrimos con nuestro navegador firefox.
- ha disfrutar.

```
<?xml version="1.0"?>
<?xml-stylesheet href="chrome://global/skin" type="text/css"?>
<window title="XUL Nested Trees"
  http://www.w3.org/1999/xhtml"
  xmlns="http://www.mozilla.org/keymaster/gatekeeper/there.is.only.xul">
  <|CDATA[
  function getTreeItemById(treeID, var)
  {
    document.getElementById(treeID);
    var selection = tree.contentView.getItemAtIndex(tree.currentIndex);
    var text = selection.firstChild.firstChild.getAttribute("label");
    document.getElementById(textID).value = text;
  }
  </script>
  <description><html:h1>XUL Nested Trees</html:h1></description>
  <hbox flex="1">
```

Buscando una extensión en la web.

- Busquemos en barra de búsqueda de firefox
- Con el nombre de : Amazon webservice firefox extension.
- Extension Room (1 url.)
- Sitio web donde encontrar extensiones, temas, etc.
- Mozilla Update.
- MozDev.

<https://addons.mozilla.org/>

Atención con el soporte para la versión de tu navegador.

En el empeño y desarrollo puedes conseguirlo como lo siguiente.

- Vamos a ver el claro ejemplo de Amazon que esta muy bueno.
- Tienes ventajas de poder conectar a un webservice,

```
<?xml version="1.0"?>
<?xml-stylesheet href="chrome://global/skin" type="text/css"?>

<window title="XUL Nested Trees"
  xmlns="http://www.mozilla.org/keymaster/gatekeeper/there-is-an-xul">
  <script>
    <![CDATA[
      function test() {
        var tree = document.getElementById(treeID);
        var selection = tree.contentView.getItemAtIndex(tree.currentIndex);
        var text = selection.firstChild.firstChild.getAttribute("label");
        document.getElementById(textID).value = text;
      }
    ]>
  </script>

  <description><html:h1>XUL Nested Trees</html:h1></description>

  <hbox flex="1">
```


Dudas, Consultas, Gracias..

Referencias.

Extension Developers extension

<http://ted.mielczarek.org/code/mozilla/extensiondev/>

Ejemplos de Widget.

<http://www.hevanet.com/acorbin/xul/top.xul>

XUL programming refrence

<http://www.mozilla.org/xpfe/xulref/>

XUL planet

<http://xulplanet.com/>

Wiki firefox.cl

<http://wiki.firefox.cl/>

Leonardo Olmos Fernández (bash@firefox.cl)