

EyeOS

TUTORIAL

MI PRIMER PASO EN eyeOS

JUAN JOSE BONILLA RODRIGUEZ

MARZO 2010

Introducción

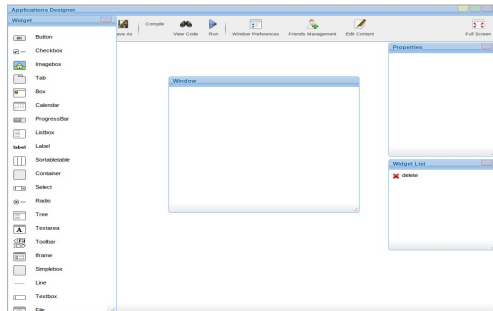
Este pequeño tutorial es el primero de varios que iré desarrollando a efecto de que podamos ir intercambiando experiencias en el desarrollo de aplicativos. Se que parecerá sumamente sencillo pero les aseguro que para lograr hacer funcionar lo que tienen en su mano me llevó como dos semanas. Una para medio entender eyeOS y otra para consultar en la comunidad como llegar a tenerlo funcionando. Doy gracias a los miembros del chat que estuvieron dispuestos a ayudarme en mis consultas y guiarme en el arduo proceso del auto aprendizaje.

El siguiente proyecto en mente ya será algo más dinámico que iré explicando paso a paso como lograr que un botón despliegue un mensaje y así poco a poco iremos aprendiendo el uso de eyeOS en el desarrollo de aplicativos.

Nuevamente gracias por su apoyo y espero poder contar con sus observaciones en eyeos.tutorial@gmail.com. De antemano agradezco todas las observaciones al documento y de igual forma les suplico que si alguien lo traduce me haga llegar la traducción al mismo correo.

Explorador interno en Eyeos

Básicamente es la generación de un espacio en el cual una url puede ser desplegada y puede ser útil al momento de desarrollar un aplicativo web que se desee ejecutar con algún tipo de limitantes para el usuario como es el caso de una intranet que sólo sirva para consulta pero no tenga acceso a la impresión, que fue el caso de nuestra institución y de donde nace la inquietud de investigar sobre eyeOS y evaluar las posibilidades de ejecución.

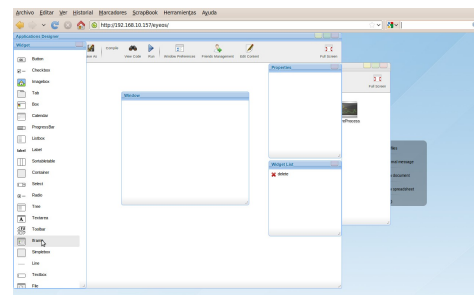


Para el primero de los proyectos utilizaremos el eyedesigner que debería ser similar a figura que se muestra a la par.

Como se aprecia en la figura, al lado izquierdo de la misma encontramos los widget que en pocas palabras vienen a ser los objetos que colocaremos en la ventana que aparece al centro.

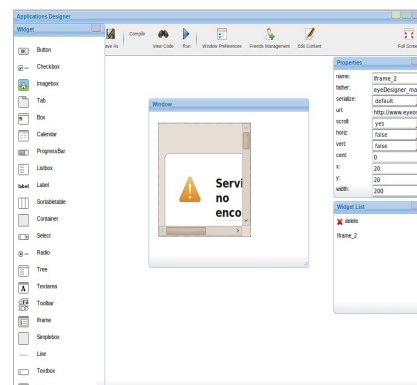
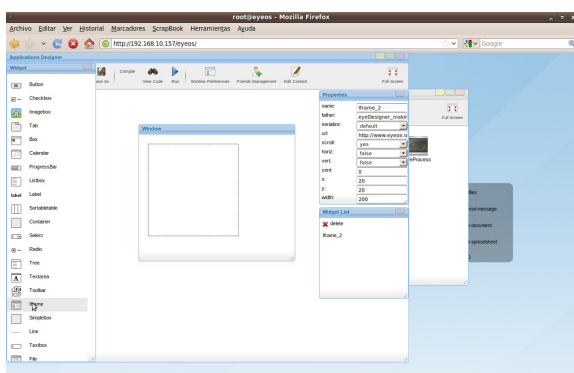
Como segundo paso seleccionamos el widget que deseamos colocar en la ventana y para nuestro ejercicio seleccionamos

el iframe dándole un clic sobre él tal como se ve en la figura siguiente.



En este momento podemos ver que aparece un recuadro en blanco que puede dar un error del tipo web si no hay conexión a internet pero no te preocupes, todo ello indica que las cosas van caminando bien, una vez estés conectado a internet verás que todo opera perfectamente.

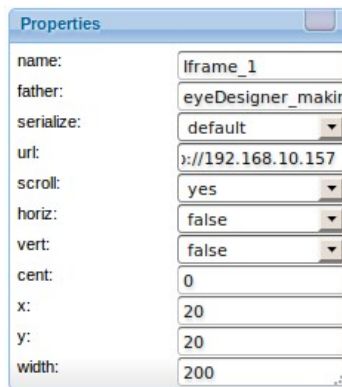
Las siguientes figuras te ilustrarán de manera detallada acerca de lo que acaba de ser explicado en el párrafo anterior.



Para el ambiente en el cual me encuentro desarrollando no cuento con acceso a internet desde el terminal por lo que esa es la razón del error pero cualquier otro que se encuentre conectado a la web no debería tener este problema.

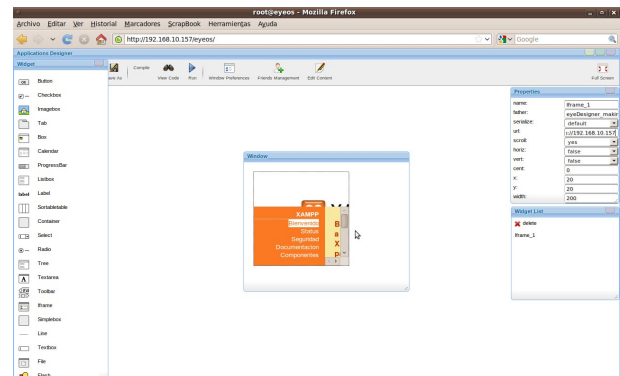
En la parte derecha de la última imagen encontramos dos recuadros en los cuales encontramos información requerida del aplicativo en uso, en el recuadro superior encontramos las propiedades del widget que tengamos seleccionado en la ventana. En el inferior tenemos el listado de los diferentes widgets que hay en la ventana que estamos trabajando, siendo esta en la cual podemos seleccionar el widget a configurar o incluso eliminarlo.

Ahora con todo en pantalla podemos pasar a la etapa de acondicionamiento del widget para que opere de manera agradable y para ello empezaremos por eliminar ese erro que se ve en el iframe y para ello en el recuadro de las propiedades buscamos la propiedad url y en ella colocaremos la dirección a la cual deseamos que el navegador despliegue. Para nuestro caso colocaremos la dirección ip del equipo que tiene el servidor eyeos, que no es el mismo de la laptop, y sería así: <http://192.168.10.157>

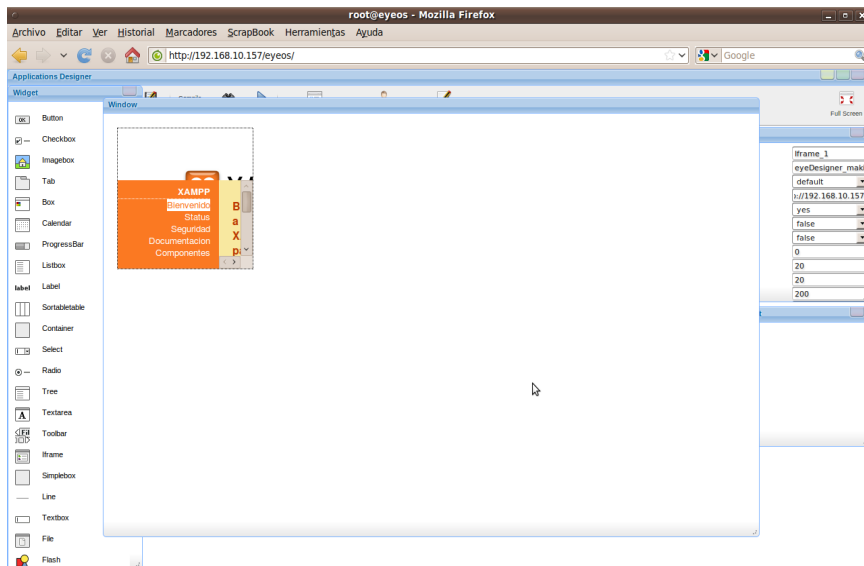


No olvide darle enter a una vez termine de escribir la dirección, mire que yo pasé toda una mañana buscando el error

hasta que apreté el enter y boalá ví el resultado tal y como se muestra en la siguiente figura:



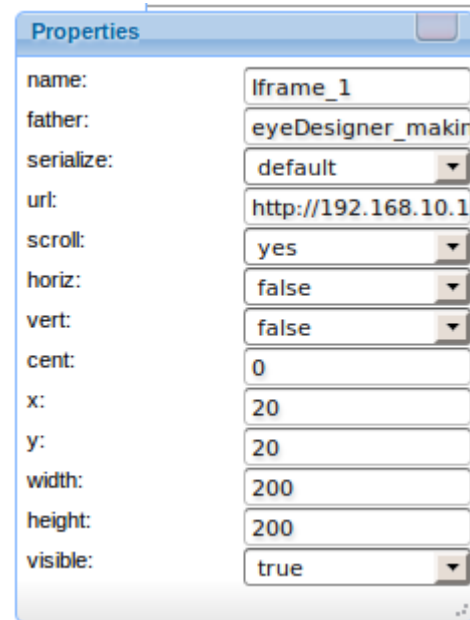
Bien, hasta este momento ya tenemos el pimpollo listo pero hay que cambiarle las dimensiones y posición para que opere a nuestro gusto. Eso lo hacemos siempre en las propiedades del objeto (widget) modificando pero primero debemos modificar el tamaño de la ventana para lo cual deberá acercarse al borde que usted desee modificar hasta que cambie a dos líneas perpendiculares -| y entonces presione click para arrastrar el borde de la misma al tamaño deseado.



En este momento puede apreciarse como el iframe se encuentra en otra posición y como además está en una ventana mucho más grande que la que se veía anteriormente para poder realizar las modificaciones es necesario que se active el recuadro de propiedades del objeto y así revisemos algunas de esas propiedades.

No es el objetivo de este tutorial realizar un detalle del alcance de las propiedades de cada objeto, por el contrario de manera práctica tocaremos las que necesitamos utilizar para poner nuestro script en operación.

- Url: indica la dirección a la cual nos queremos conectar y puede ser tanto una dirección en texto o en ip como se muestra en la imagen.
- x: Indica la posición en la horizontal (contado desde la esquina superior izquierda) que tendrá el objeto que para nuestro caso es un iframe.
- y: Indica la posición en la vertical (contado desde la esquina superior izquierda) que tendrá el objeto que para nuestro caso es un iframe.
- Width: indica el ancho que tendrá el objeto
- height: indica el largo que tendrá el objeto

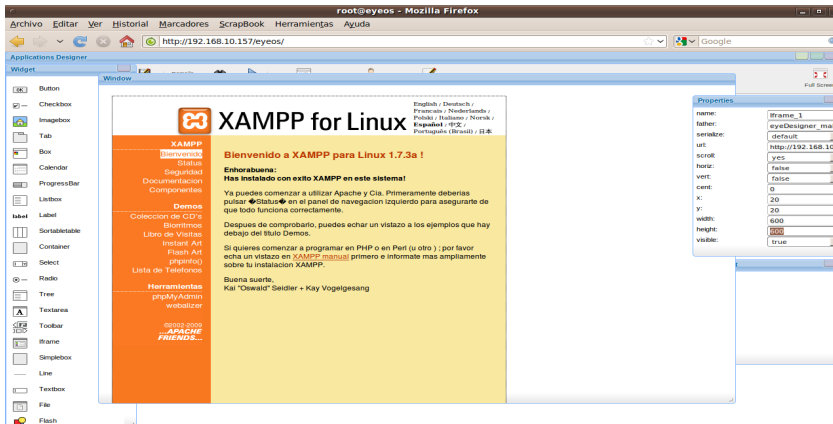


En base a lo anterior podemos decir que Para nuestro ejemplo debemos recordar que el equipo en el que trabajo es uno de prueba y que no estamos conectados a la web sino que a un servidor que nos sirve el sistema por lo que colocaremos la direccion del servidor en cuestión.

Al colocar la siguiente expresión: <http://midireccionip> hago que el navegador busque un archivo index que puede ser con la terminación htm, html, php, o algún otro que se identifique como página web y ese nos mostrará. De no existir este archivo estaremos recibiendo un mensaje de que no existe el archivo en cuestión por parte del servidor de páginas web (Apache, IIS, etc.).

Modificando la propiedad width y la propiedad de 200 a 600 triplicaremos las dimensiones del objeto como puede apreciarse en la figura pero por favor no solo escriba el número, por favor, presione el enter o de lo contrario no verá cambios (yo pasé por un gran dolor de cabeza hasta que lo hice).

De verdad se los digo sino haga la prueba, escriba 600 y presione el tab y no verá cambios, después retorne a las propiedades width y height y solo presione enter y verá una imagen similar a la siguiente.



Muy bien, ya está hecho el diseño, no podemos hablar de que el aplicativo esté listo.

En estos momentos vemos algo operativo pero que no opera por sí sólo por lo que les digo que estamos a la mitad del proyecto.

Ahora procedemos a guardarlo para evitar cualquiera de esos accidentes tan conocidos en el ámbito informático que sin duda

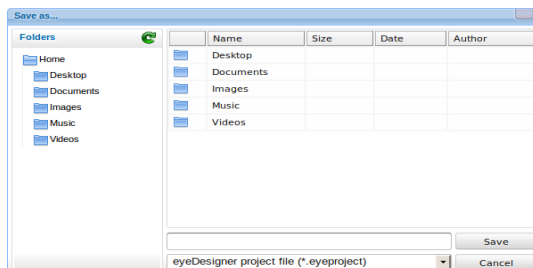
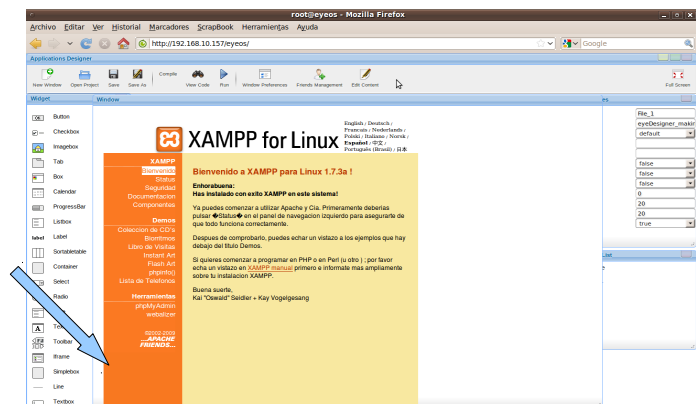
nos “alegran la vida” cuando sucede (hablé de la pérdida de todo el trabajo).

Para almacenarlo necesitaremos bajar un poco la ventana y disminuir el tamaño del listado de widgets para poder tener a la vista las opciones del menú como se aprecia en la siguiente figura.

Recuerde que para disminuir el listado de widget es mejor si lo trata de hacer por la parte de arriba que si lo hace por los laterales ya que puede empezar a activar objetos sin darse cuenta (ya me pasó... :-).

Como es de imaginarse,

para almacenar el trabajo se presiona el icono que dice save y con eso tenemos un recuadro en el cual nos pide que le indiquemos en donde se guardará. Esto funciona de manera similar a como opera cada una de las opciones de guardar en los sistemas y puede apreciarse en la imagen que se muestra a continuación.

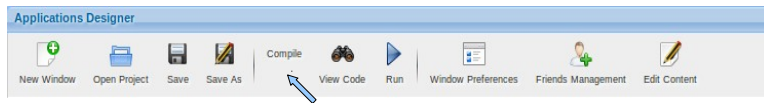


Recordemos que lo que se ha grabado es un proyecto, no es un programa que pueda ejecutarse fuera de la herramienta de diseño, para que ello pueda realizarse es necesario que se compile el proyecto, cosa

que puede realizarse sin salvar el proyecto pero que considero que no es una buena práctica pues un descuido y tendremos todo nuestro trabajo tirado.

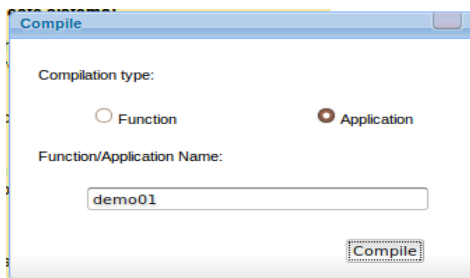
Una vez el proyecto está almacenado no puede decirse que se encuentre operativo ya que en este momento es cuando tenemos que llevar todo nuestro trabajo al servidor y proceder a prepararlo para que esté disponible a los usuarios por ello lo siguiente a realizar es la compilación del proyecto.

Para compilar un proyecto basta con presionar el botón compile (que se muestra en la siguiente figura):



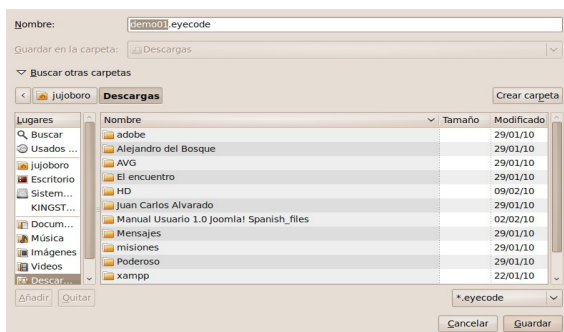
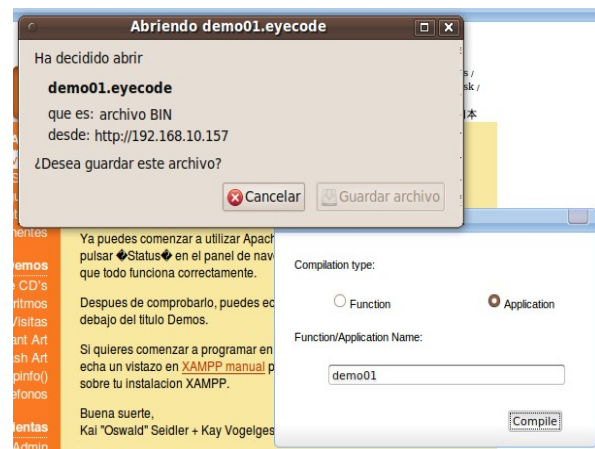
Al presionar el botón compile el sistema realmente ejecuta una descarga web desde el servidor hacia su computadora pero

primero debe indicarse si se compilará como función o como aplicación. Para nuestro ejemplo lo haremos como una aplicación así como también le daremos un nombre y para nuestro caso la llamaremos demo01 como se muestra en la siguiente figura:



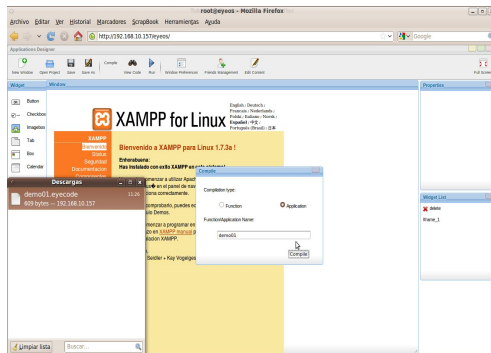
Al seleccionarse el tipo de compilación y haberle asignado el nombre se presiona el botón compile iniciará una descarga tipo web que puede variar del mostrado debido al sistema

operativo o al navegador. Para nuestro caso que el cliente es una laptop linux ubuntu Karmic Koala (9.10) tenemos la siguiente secuencia:



A lo cual presionamos el botón Guardar archivo y aparece un cuadro en el cual se nos pide que indiquemos en que lugar se va a guardar el archivo

Para nuestro ejemplo y por orden de trabajo he preparado una carpeta para almacenar los compilados, que realmente son scripts php los que nos entrega el diseñador de eyeos así que procedemos a indicar la el lugar donde queremos que sean depositados nuestro archivo para finalizar con un mensaje de descarga finalizada como se ve en la siguiente figura.



Una vez finalizada la descarga iremos a ver el archivo que nos ha entregado el diseñador de eyeos pero antes procedamos a cerrar eyedesigner.

Al revisar la carpeta en la cual está el producto de la compilación encontramos un sólo archivo con terminación eyecode que se llama demo01.eyecode y que realmente es un archivo php, este archivo debemos prepararlo para operar en el servidor eyeos y para ello ejecutamos los siguientes pasos:

1. Lo copiamos y le cambiamos el nombre a “app.eyecode” (sin comillas)
2. creamos en el servidor una carpeta con el nombre original del aplicativo demo01 que deberá estar en la siguiente dirección:
3. `../eyeosxxxxxxxxxx/apps/demo01`
4. en la carpeta que acabamos de crear copiamos el archivo con nombre app.eyecode que acabamos de crear. Este es el aplicativo que se ejecutará pero aún no es visible desde las estaciones de trabajo por eyeOS.
5. Para que el aplicativo sea visible por eyeOS se debe agregar un archivo más y se un archivo que debe tener el nombre info.xml y que para no hacer difícil su creación podemos copiarlo de la carpeta `../eyeosxxxxxxxxxx/apps/eyeApps`.
Podríamos haberlo copiado de cualquier otra carpeta pues este le da a eyeOS la información necesaria para que sea desplegado en el ambiente.
6. En estos momentos no es visible el aplicativo ya que es necesario ajustar los datos del mismo en el archivo info.xml a efecto de que el sistema pueda identificarlo como un aplicativo por lo que dedicaremos un poco de tiempo a este archivo.

En este archivo aparecen varias etiquetas y hablaremos de ellas a fin de que podamos configurar nuestro aplicativo de la mejor manera posible y así cuando sea presentado en eyeOS este aparezca donde exactamente queremos y no en otro lado.

- `<name>` Define el nombre con el cual se conoce el aplicativo y para nuestro caso le pondremos demo01
- `<category>` Define la categoría a la cual pertenece el aplicativo, esto es la posición en la cual aparece cuando activamos el ícono applications y aparecen varias como son:

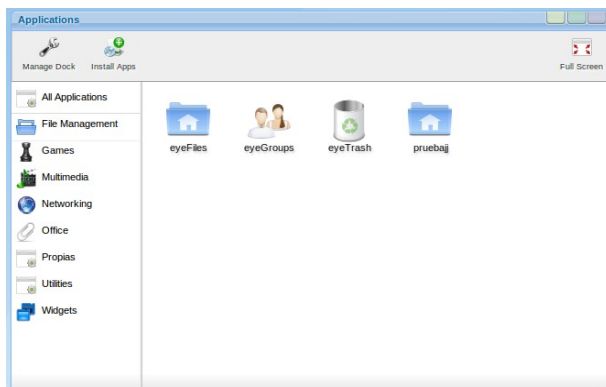
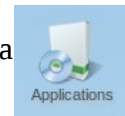
- File Management
- Games
- Multimedia
- Networking
- Office
- Utilities
- Widgets

A efecto de clasificar nuestros aplicativos le pondremos una categoría totalmente diferente a la existente por lo que para nuestro caso le llamaremos **Propias**, lo cual nos desplegará en el listado de Applications una nueva categoría con ese nombre como se aprecia en la imagen mostrada a continuación.

- <version> Nos sirve para identificar que versión de aplicativo estamos desarrollando
- <description> Permite incorporar una breve descripción de la utilidad del sistema por lo que le colocaremos algo como esto: Primer demo de Eyeos.
- <author> Colocamos nuestro nombre
- <license> no la tocaremos
- <type> no la tocaremos
- <icon> Define el ícono con el cual se identifica el aplicativo.

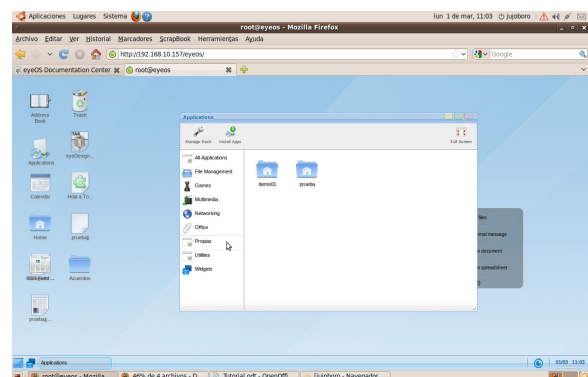
Una vez se han realizado todas las modificaciones y las grabamos procedemos a verificar la operatividad del aplicativo en eyeOS.

7. Para verificar la operatividad de demo01 activamos la opción applications para obtener una pantalla similar a la siguiente figura:



Como se observa aparece una nueva categoría que es propias y es en esa categoría en la cual podremos encontrar la aplicación que acabamos de seleccionar que debería ser presentada como se muestra a continuación.

En la figura mostrada se aprecian dos scripts, demo01 y prueba, cada uno son aplicativos desarrollados por su servidor y que han permitido incursionar en el mundo de eyeOS. Para verificar que el aplicativo desarrollado opera basta con dar un clic en el mismo y obtenemos:





En este momento podemos apreciar en la imagen mostrada el navegador que ha sido dirigido por eyeos de manera estática a una dirección ip donde podrá realizar tareas que el aplicativo invocado le permita ejecutar.