En esta oportunidad quiero explicar varios conceptios mientras desarrollo una aplicación para la creacion de encuestas o examenes en linea. En cada tutorial separado explicare mas detalladamente cada uno de los puntos de esta entrada, ¿Que vamos a aprender mientras creamos esta aplicacion? Muy interesante, voy a enseñarles todo lo que yo utilizo a la hora de crear una pagina web:

A la hora de crear la pagina vamos a tener las siguientes premisas funcdamentales:

* **Pensar todo el tiempo en el usuario:** Vamos a evitar cualquier actividad extra para el usuario, crear ayudas en todo momento, prevenir en lugar de corregir, darle velocidad a la pagina, interfaz mimalista, etc.
* **La simplicidad no tiene nada que ver con el tamaño:** Tu puedes tener la pagina mas grande del mundo pero que siga siendo muy simple desde todo punto de vista (arquitectura, organizacion, usabilidad, etc.)
* **Estandarizar los procesos:** Hay actividades que son muy repetitivas, hay que tratar de estandarizarlas o incluso tratar de automatizarlas para que cada vez se hagan mas rapido y sin errores.
* **Patrones de diseño:** Todos los problemas que se nos puedan presentar a la hora de crear una pagina web ¡ya se le presentaron a otra persona!, busca en interner, utiliza los patrones de diseño, tal vez trabajemos mas al principio pero luego estaremos volando y sin equivocaciones.

1. **¿Cómo diseñar una pagina con maquetación CSS?**

Primero vamos a maquetar la pagina en contenedores <div> para utilizar estandares de CSS, asi vamos a poder diseñar una pagina pensando en la informacion y el contenido y luego le daremos un diseño creando hojas de estilos que tendran classes que se les aplicaran a los contenedores que definimos, algo asi:

Fijense que la pagina web html solo contiene contenedores de informacion que no tienen ningun diseño (la cabecera de la pagina, el contenido y el pie de la pagina), cuando se le aplica la capa de la derecha (la hoja de estilos) los contenedores cambian de color, tipo de letra, color de letra, sangrias para el texto, etc. Cuando nosotros queramos podemos cambiar la hoja de estilos y nuestra aplicacion se vera muy diferente pero seguira teniendo la misma estructura e informacion.

1. **¿Cómo utilizar plantillas para poder compartir la misma cabecera, menu y pie en todas las paginas sin tener que copiar y pegar el codigo?**

La idea es crear una archivo “header.php” que sera la cabecera de las paginas y lo importaremos cada vez que queramos renderizar o mostrar una pagina en expecifico, de esta manera si realizamos un cambio en esta cabecera no tendremos que ir pagina por pagina realizando el cambio en todas las cabeceras sino que cambiariamos el archivo header.php y el cambio se reflejaria en todas las paginas. Lo mismo ocurriria en el caso del menu (menu.php) y el pie de la pagina (footer.php).

1. **¿Cómo evitar recargar la pagina tan amenudo?**

Con las nuevas tecnologias es posible recargar contenedores especificos de las paginas web en vez de recargar toda la pagina, de esta manera aumentamos mucha facilidad de uso y tiempo de espera. Ademas recargamos menos al servidor ya que realizamos menos peticiones y por ultimo vamos a ver como termina siendo incuso ¡Mucho mas facil que antes! Trabajar de esta manera. Nosotros utilizaremos el framework de javascript llamado JQuery para lograr estas funcionalidades.

1. **¿Como estructurar nuestra pagina de una manera sencilla y organizada para que siga siendo simple a pesar de que ser bastante grande?**

Luego de buscar y probar la gran mayoria de los frameworks de desarrollo yo decidi crear mi propia arquitectura para mis paginas web, siguiendo algunes estandares y patrones de diseño pero sin volverla tan compleja y abstracta como me parece que ocurre en el caso de frameworks existentes como symfony.

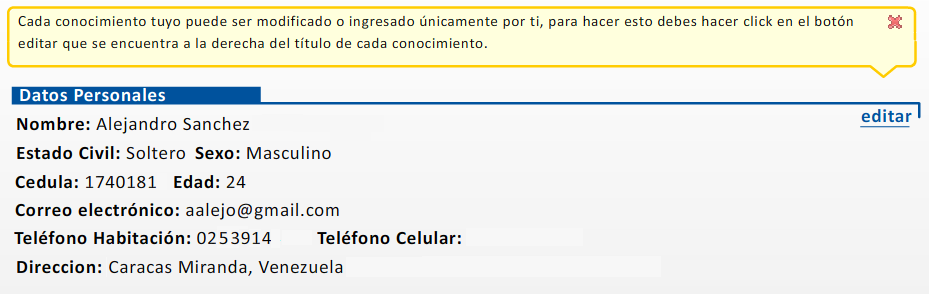
1. **¿Como dividir en capas toda la pagina web para lograr la independencia de cada una de ellas?**

En la arquitectura que diseñe la pagina web se divide en 6 capas donde cada una de ellas es completamente independiente, es decir, si quieres cambiar de base de datos solo tienes que cambiar la capa de comandos, si quieres cambiar el diseño de la pagina, solo tienes que cambiar la capa de diseño, etc.

En mi arquitectura yo planteo las siguientes capas:

1. Diseño: Renderiza los datos, les da colores, tamaño, etc. Puras hojas de estilo.
2. Vista: La estructura de los datos, contiene la informacion y no contienen practicamente ninguna logica.
3. Cliente: Creacion de componentes DatePicker, Slider, etc. Validaciones, Evitar recargo de la pagina, ayuda de usuario, etc.
4. Logica: Transacciones (crudManager), Plantillas (templateManager), Permisos de usuario, Validaciones, etc.
5. Comando: Enlace entre el modelo y la logica para independizar a la pagina de la escructura de los datos.
6. Modelo: Puede ser una BD, unos web services, o cualquier otra forma de obtener y guardar datos.
7. **¿Como crear una ayuda de usuario sobre la pagina?**

El problema con las ayudas es que siempre son una seccion aparte de la aplicacion y el usuario nisiquiera sabe como encontrar la ayuda para lo que esta buscando. Otra de las ventajas que te da estructurar tu pagina en contenedores <div> es que tienes la posibilidad de mostrarlos y ocultarlos facilmente. De esta manera si nosotros consideramos que la pagina actual que estamos creando debemos colocar un espacio de ayuda entonces podemos crear un contenedor con esa informacion y en cualquier momento el usuario podra ocultarlo y continuar como si nada. Por ejemplo:



1. **¿Cómo prevenir en lugar de corregir?**

La parte mas importante a la hora de crear una web es como se plantea la interaccion del usuario con la aplicacion, la mayoria de las paginas utilizan los componentes estandares de formulario que tiene html para lograr esta interaccion, pero lamentablemente estos componentes son muy limitados y en un gran porcentaje de las veces dificultan el uso de la pagina. Para eso en la arquitectura que estamos planteando hay un espacio dedicado a la creacion o importacion de nuevos componentes javascript. ¿Porque introducir una fecha en un TextInputo en una seria de ComboBoxs? Lo mas natural seria un calendario, ¿no les parece? Existen una serie de componentes como NumericStepper, Slider, DatePicker, etc. Que hacen que la experiencia de usuario se vuelva mucho mas simple y practica y ademas evitan muchisimas validaciones hacia nosotros porque no dejan espacio para los errores. Y si ademas colocas tremenda ayuda, ¡es excelente!

1. **¿Como hacer una página segura?**

Lo primero que debemos tomar en cuenta a la hora de hacer una pagina segura es que la capa de javascript no puede ser tomada en cuenta para la seguridad en casi nada, ¿Porqué? Porque puede ser desactivada en cualquier momento por los hackers, incluso los navegadores mas viejos insiquiera soportan javascript. ¿Que quiere decir esto? Pues muy facil, las validaciones de datos que hacemos en javascript tambien las debemos hacer en PHP o en cualquier lenguaje que estemos usando del lado del servidor. Pero OJO: Igual debemos seguir haciendo las validaciones del lado del cliente para no recargar al servidor y para seguir con nuestra premisa de PREVENIR en lugar de CORREGIR.

Para hacer nuestra arquitectura segura yo decidi dividir toda la interaccion con los datos en transacciones, si se fijan en la capa logica existe el crudManager y el infoManager. El primero de estos se encarga de eliminar, modificar y insertar los datos. El segundo se encarga solo de consultaros, por lo tanto requieren permisos de usuario diferentes para ser utilizados. Los mismo ocurre con el templateManager que se encarga de armar las vistas que el usuario va a visualizar, si el usuario no tiene permisos para visualizarlas entonces lo lleva a una pagina de error.

¿Como se cargan los permisos? En variables de sesion, en la arquitectura que vamos a utilizar ya existe un modulo de autenticacion que solo debemos realizarle pequeños cambios para adaptarlo a nuestra pagina, al momento de loguearte te otoga permisos de usuario que luego tu controlas para acceder a las vistas y transacciones desde el crudManager, infoManager y templateManager.

1. **¿Como realizar los crud sin recargar la pagina?**

Es demasiado comun en una pagina web querer insertar, modificar, eliminar y consultar valores de una entidad en especifico (Persona, Paciente, Profesor, etc.). Con unos pocos conceptos es posible estandarizar estos procedimientos para hacerlos mas faciles de programar y ademas se logra a la misma vez evitar recargar la pagina tan amenudo. Si estamos utilizando la arquitectura COCOAS (De la que he hablado durante toda esta entrada) solo debemos crear dos contenedores, uno para visualizar la informacion y otro para modificarla, por ejemplo: En el siguiente ejemplo se tiene una lista de anuncios publicada por un condiminio, y se quiere modificar un anuncio en particular:

Este seria el codigo del contenedor para visualizar los datos:

<div id="showPost1" class="post">  
 <h1 class="title"><a href="#">Racionamiento de agua</h1>  
 <p>Los dias lunes y martes no entrará agua al edficicio de la calle, por lo tanto se racionará el agua del tanque en el siguiente horario: Lunes: de 8am a 12pm - de 2pm a 8pm. | Martes: de 8am a 12pm - de 2pm a 8pm.</p>  
 <p style="text-align:right;">  
 <a id="detallesLink" class="editLink" name="Post1>" href="#">editar</a>   
 <a class="deleteLink" name="Post" href="crudManager.php?action=delete\_anuncio&codigo=1">Eliminar</a>  
 </p>  
</div>

Y este seria el contenedor para modificar los datos:

<div id="editPost1" class="post" style="display:none;">  
 <form action="crudManager.php" method="post" id="anuncioForm">  
 <input name="action" type="hidden" value="edit\_anuncio" />  
 <input name="codigo" type="hidden" value="1" />  
 <h1 class="title">Titulo: <input name="titulo" type="text" value=” Racionamiento de agua” /></h1>  
 <textarea name="mensaje" cols="" rows="" style="width:500px; height:100px;"> Los dias lunes y martes no entrará agua al edficicio de la calle, por lo tanto se racionará el agua del tanque en el siguiente horario: Lunes: de 8am a 12pm - de 2pm a 8pm. | Martes: de 8am a 12pm - de 2pm a 8pm.</textarea>  
 <p style="text-align:right;"><input name="" type="submit" value="Guardar" /></p>  
 </form>  
</div>

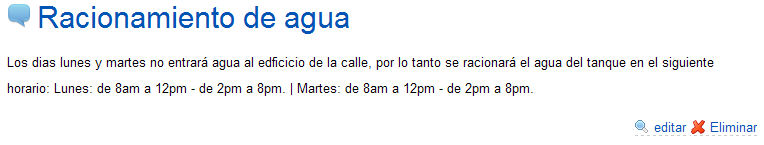
He marcado de colores las zonas que a son importantes en ambos contenedores, si estamos trabajando con COCOAS, debemos decirle cual contenedor es el que se utiliza para mostrar la informacion y cual contenedor es el que se utiliza para editar la informacion, esto lo hacemos con el campo ID de cada uno de los contenedores, colocando la palabra “show” o “edit” (segun el caso) seguida de el nombre de la entidad que estamos modificando en este caso “Post” seguida de el codigo de la entidad especifica que estamos modificando (can caso de que estemos en una lista de items de la misma entidad).

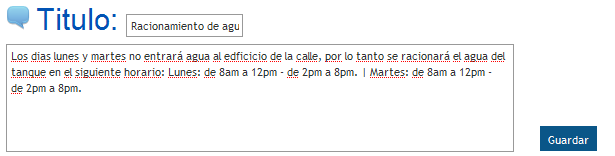
Siempre el contenedor que se creo para modificar la entidad debe tener por defecto en sus estilos la propiedad **display:none** para que la pagina se desempeñe mucho mejor, esta propiedad le dice al contenedor que por defecto el debe estar oculto.

Luego debemos irnos al vinculo que hemos definido que se le debe hacer click para modificar la entidad (ocultar el contenedor de visualizar y mostrar el contenedor de modificar) y colocarle en su propiedad class el valor “editLink”, y en su propiedad name el nombre de la entidad y codigo que este vinculo modifica (en este caso “Post1”) de esta manera COCOAS sabra lo que tiene que hacer.

Por ultimo debemos irnos al vinculo que hemos definido que se le debe hacer click para eliminar la entidad y colocarle en su propiedad class el valor “deleteLink” y de esta manera COCOAS sabra lo que tiene que hacer.

Al final los dos contenedores se verian de la siguinte manera, por supuesto que no se verian al mismo momento, sino que al presionar “editar” se ocultaria el primer contenedor y se mostraria el segundo contenedor:





Bueno en las siguientes entradas explicare mucho mejor cada uno de los puntos de los que he hablado en esta oportunidad ya que todavia nos queda muuuuuucho de que hablar y buenas practicas que seguir para conseguir mucha rapidez, de todas maneras para los que saben leer ingles aqui les dejo un pequeño documento donde explica un poco mas detallada la arquitectura COCOAS.