

# Ejercicio 1

## HTML5 y CSS

**Prof: José Urzua**

Auxiliar: Pablo Pizarro, Gabriel Iturra

## Actividad

Supongamos que ud. tiene una empresa en la que debe subir archivos muy pesados a un servidor tipo Dropbox, pero éste no le permite dado el tamaño. Es por ello que piensa implementar su propio gestor de archivos con diferentes opciones para agilizar el desarrollo de su empresa (**Hackbox**). Después de contactarse con clientes y desarrolladores las siguientes ideas surgieron en una primera iteración:

- Al subir un archivo éste debe ser asignado con un nombre (Obligatorio). De máximo 100 caracteres y mínimo 10.
- Debe tener un botón para poder escoger el archivo (Obligatorio).
- Cada archivo debe tener diferentes niveles de “privacidad”, si es público el archivo aparecerá en la lista principal del servidor. Si es privado, sólo las personas con el link correcto podrán acceder.
- Un archivo puede ser subido con una contraseña (Opcional), sólo aquellos que posean dicha contraseña pueden descargar el contenido.
- Además, el archivo se puede autodestruir después de cierto tiempo (Opcional). Este será un campo en segundos, si es distinto a cero entonces el servidor lo borraría.
- Se puede añadir también un texto para poder realizar comentarios. (Opcional), máximo 1000 caracteres.
- Al final de la página debe existir un botón para subir el archivo.

Para ello debemos implementar un documento HTML5 y encoding UTF 8 con los siguientes elementos, en orden:

- Título de la página y tag (H1) o similar para desplegar el título de la aplicación (Hackbox).
- Formulario que ejecute una invocación POST y action, por ahora vacío.
- Dentro del formulario se deben agregar los siguientes elementos:
  - Texto: “Nombre”. Entrada de texto de tamaño 10 y con 100 caracteres como máximo (id=nombre). Obligatorio. Class=text-field.

- Texto: “Archivo”. Botón para escoger archivo. Input type-file. (id=archivo). Obligatorio.
- Texto: “Privacidad”. Selección múltiple (id=privacidad) donde el usuario puede escoger cualquiera de los siguientes: Público (valor 1), Privado (valor 2). Obligatorio.
- Texto: “Contraseña”. Entrada de texto tipo password de tamaño 10 y con 10 caracteres como máximo (id=password). Opcional.
- Texto: “Autodestruir”. Entrada de texto tipo numérico de tamaño 10, valor mínimo 1, máximo 31536000 (id=autodestruir). Opcional.
- Texto: “Comentario”. Caja de texto libre multilíneas, de ancho 40 y 10 líneas de alto (id=comentario). Agregue un placeholder “ingrese sus comentarios acá”.
- Botón para enviar con el nombre “Upload”.
- La vista debe ser validada en la **w3**.
- Puede usar el tag <br/> para agregar saltos de línea.
- Todo el formulario debe estar alineado al centro. Agregue incluya su formulario dentro de un tag div, luego utilice CSS para centrar dicho elemento. Utilice los atributos “width”, “margin”, “padding”.
- Todo el css necesario puede estar en el mismo archivo o en uno separado. No use *inline styles*.

## Ejercicio

Implementar toda la página web, teniendo cuidado con el nombre de los campos y que sea visualizable en navegadores modernos como Chrome, Opera, Firefox, etc.

Responder además las siguientes preguntas de manera breve. Puede hacer un LaTeX o un documento tipo word. Enviar como PDF adjunto a los archivos de su tarea.

- ¿Qué dificultades presenta implementar una aplicación de este estilo en términos de seguridad?
- ¿Cuál es la diferencia entre HTML y CSS?
- ¿Cuál es la diferencia entre HTML y HTML5?
- ¿Todos los tags en HTML son del estilo <tag>, <\tag>?