

PROGRAMA DE ESTUDIOS

DESARROLLOS DE SISTEMAS DE INFORMACIÓN

**COMUNITY MANAGER Y
MARKETING DIGITAL**

Tema

**CREACIÓN DE APLICACIONES
DINÁMICAS CON JQUERY**

Creación de Aplicaciones Dinámicas con jQuery

Objetivo:

Aplicar jQuery para desarrollar aplicaciones dinámicas mediante la manipulación del DOM, la gestión de eventos y el uso de AJAX para interactuar con datos en tiempo real.

PARTE CONCEPTUAL

Pregunta detonadora:

¿Cómo podemos mejorar la interactividad de una aplicación web sin recargar la página?

Repaso de funcionalidades clave de jQuery para apps dinámicas:

Manipulación del DOM:

- `.append()`: Inserta contenido al final de un elemento.
- `.prepend()`: Inserta contenido al principio.
- `.remove()`: Elimina elementos del DOM.

Manejo de eventos:

- `.on("click", handler)`: Asigna un evento a uno o varios elementos.
- `.keyup()`: Se activa al presionar una tecla.
- `.submit()`: Detecta el envío de un formulario.

AJAX con jQuery:

- `.get(url, callback)`: Obtener datos de una fuente externa.
- `.post(url, data, callback)`: Enviar datos al servidor.



PARTE DEMOSTRATIVA (20 minutos)



Ejemplo guiado: Lista de tareas dinámica

HTML base:

html

```
<input type="text" id="nueva-tarea" placeholder="Escribe una tarea">
<button id="agregar">Agregar</button>
<ul id="lista-tareas"></ul>
```

JS con jQuery:

javascript

```
$("#agregar").click(function() {
  const tarea = $("#nueva-tarea").val();
  if (tarea.trim() !== "") {
    $("#lista-tareas").append(
      `<li>${tarea} <button class="eliminar">Eliminar</button></li>`
    );
    $("#nueva-tarea").val("");
  }
});

$(document).on("click", ".eliminar", function() {
  $(this).parent().fadeOut(300, function() { $(this).remove(); });
});
```

Bonus: Efectos visuales

javascript

```
$("#li").hover(  
  function() { $(this).css("background-color", "#eee"); },  
  function() { $(this).css("background-color", "white"); }  
);
```

 **ACTIVIDAD PRÁCTICA INDIVIDUAL (25 minutos)** **Desafío:**

Crea una miniaplicación que cumpla con los siguientes requerimientos:

- **Tabla dinámica:**
 - Un formulario para agregar elementos (nombre, edad, etc.).
 - Botones para eliminar filas de la tabla.
- **Filtro en tiempo real:**
 - Un input que permita filtrar los elementos por nombre.
- **Consulta AJAX:**
 - Realiza una petición a una API pública y muestra los resultados en la interfaz.

 **Sugerencias técnicas:****HTML:**

html

```
<input type="text" id="filtro" placeholder="Buscar...">  
<table>  
  <thead><tr><th>Nombre</th><th>Acciones</th></tr></thead>  
  <tbody id="tabla-datos"></tbody>
```

</table>

JS:

javascript

```
$("#filtro").on("keyup", function() {  
    const valor = $(this).val().toLowerCase();  
    $("#tabla-datos tr").filter(function() {  
        $(this).toggle($(this).text().toLowerCase().includes(valor));  
    });  
});  
  
$.get("https://jsonplaceholder.typicode.com/users", function(data) {  
    data.forEach(user => {  
        $("#tabla-datos").append(  
            `|<td>${user.name}</td><td><button class="borrar">X</button></td></tr>`  
        );  
    });  
});  
  
$(document).on("click", ".borrar", function() {  
    $(this).closest("tr").remove();  
});

```

CIERRE DE SESIÓN (15 minutos)

Reflexión:

¿Cómo contribuye jQuery a mejorar la experiencia del usuario en aplicaciones web dinámicas?

Posibles respuestas:

- Evita recargas innecesarias.
- Permite manipular la interfaz en tiempo real.
- Facilita la interacción con APIs externas.

Revisión:

- ¿Se agregan y eliminan elementos correctamente?
- ¿Funciona el filtro de búsqueda en tiempo real?
- ¿La API responde y muestra resultados?

Tarea:

Crear una aplicación web con jQuery que incluya:

- Un formulario para agregar elementos a una lista o tabla.
- Botones para eliminar elementos dinámicamente.
- Una consulta AJAX que muestre datos de una API pública (por ejemplo, posts o usuarios).



INSTITUTO
KHIPU