

```
<!DOCTYPE html>

<html lang="es">

<head>

  <meta charset="UTF-8" />

  <meta name="viewport" content="width=device-width, initial-scale=1.0" />

  <title>Agenda con Recordatorios</title>

  <script src="https://cdn.tailwindcss.com"></script>

  <script src="https://cdn.jsdelivr.net/npm/sweetalert2@11"></script>

  <style>

    body {

      font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;

    }

    .fade-in {

      animation: fadeIn 0.5s ease-in-out;

    }

    @keyframes fadeIn {

      from { opacity: 0; transform: translateY(10px); }

      to { opacity: 1; transform: translateY(0); }

    }

  </style>

</head>

<body class="bg-gray-100 min-h-screen flex flex-col items-center p-4">

  <!-- Imagen atractiva -->

  

  <!-- Formulario para añadir eventos -->
```

```
<div class="bg-white p-6 rounded-xl shadow-md w-full max-w-md fade-in">

  <h2 class="text-xl font-bold mb-4 text-center">Agregar Evento</h2>

  <input id="title" type="text" placeholder="Título del evento" class="w-full p-2 mb-2 border rounded" />

  <input id="time" type="time" class="w-full p-2 mb-4 border rounded" />

  <button onclick="addEvent()" class="w-full bg-blue-600 text-white p-2 rounded hover:bg-blue-700 transition">Agregar</button>

</div>
```

```
<!-- Dropdown para eventos del día -->

<div class="mt-6 w-full max-w-md fade-in">

  <details class="bg-white p-4 rounded-xl shadow-md">

    <summary class="cursor-pointer font-semibold text-gray-700">Ver eventos del día</summary>

    <ul id="eventList" class="mt-2 space-y-3"></ul>

  </details>

</div>
```

```
<script>

  let events = JSON.parse(localStorage.getItem('agendaEvents')) || [];

  // Guardar y actualizar en LocalStorage

  function saveEvents() {
    localStorage.setItem('agendaEvents', JSON.stringify(events));
  }
```

```
  // Agregar evento

  function addEvent() {

    const title = document.getElementById('title').value.trim();

    const time = document.getElementById('time').value;
```

```
if (!title || !time) {  
  Swal.fire('Error', 'Por favor, completa todos los campos.', 'error');  
  return;  
}
```

```
const event = { id: Date.now(), title, time, completed: false };  
events.push(event);  
saveEvents();  
renderEvents();  
document.getElementById('title').value = "";  
document.getElementById('time').value = "";  
Swal.fire('Evento agregado', "", 'success');  
}
```

```
// Renderizar eventos
```

```
function renderEvents() {  
  const list = document.getElementById('eventList');  
  list.innerHTML = "";  
  const now = new Date();  
  
  events.forEach(event => {  
    const li = document.createElement('li');  
    li.className = 'bg-gray-50 p-3 rounded-lg border flex justify-between items-center';  
  
    const content = document.createElement('div');  
    content.innerHTML =  
    `<strong>${event.title}</strong><br><small>${event.time}</small>`;
```

```
if (event.completed) {
  content.classList.add('line-through', 'text-gray-400');
}

const btns = document.createElement('div');
btns.className = 'space-x-2';

// Botón posponer
const postponeBtn = document.createElement('button');
postponeBtn.innerText = '🕒 Posponer';
postponeBtn.className = 'text-sm bg-yellow-400 text-white px-2 py-1 rounded';
postponeBtn.onclick = () => postponeEvent(event.id);

// Botón completar
const completeBtn = document.createElement('button');
completeBtn.innerText = '✅ Completar';
completeBtn.className = 'text-sm bg-green-500 text-white px-2 py-1 rounded';
completeBtn.onclick = () => completeEvent(event.id);

btns.appendChild(postponeBtn);
btns.appendChild(completeBtn);

li.appendChild(content);
li.appendChild(btns);
list.appendChild(li);
});
}

// Posponer evento
```

```
function postponeEvent(id) {  
  const event = events.find(e => e.id === id);  
  if (event) {  
    const [hour, minute] = event.time.split(':');  
    const newDate = new Date();  
    newDate.setHours(parseInt(hour), parseInt(minute) + 10);  
  
    event.time = newDate.toTimeString().slice(0, 5);  
    saveEvents();  
    renderEvents();  
    Swal.fire('Evento pospuesto 10 minutos.', '', 'info');  
  }  
}
```

// Completar evento

```
function completeEvent(id) {  
  Swal.fire({  
    title: '¿Marcar como completado?',  
    icon: 'question',  
    showCancelButton: true,  
    confirmButtonText: 'Sí, completar',  
    cancelButtonText: 'Cancelar'  
  }).then(result => {  
    if (result.isConfirmed) {  
      const event = events.find(e => e.id === id);  
      if (event) {  
        event.completed = true;  
        saveEvents();  
        renderEvents();  
      }  
    }  
  });  
}
```

```
    Swal.fire('¡Buen trabajo!', 'Actividad marcada como completada.', 'success');  
  }  
}  
});  
}
```

```
// Revisar recordatorios cada 30s
```

```
setInterval(() => {  
  const now = new Date();  
  const currentHour = now.getHours();  
  const currentMinute = now.getMinutes();  
  
  events.forEach(event => {  
    const [hour, minute] = event.time.split(':').map(Number);  
    const reminderTime = new Date();  
    reminderTime.setHours(hour, minute - 10, 0, 0);  
  
    if (!event.completed && Math.abs(reminderTime - now) < 30000) {  
      Swal.fire({  
        title: 'Recordatorio',  
        text: `Tu evento "${event.title}" es en 10 minutos.`,  
        icon: 'info',  
        confirmButtonText: 'Entendido'  
      });  
    }  
  });  
}, 30000);
```

```
// Inicializar
```

```
renderEvents();
```

```
</script>
```

```
</body>
```

```
</html>
```

FILIBERTO RODRIGUEZ OCHOA