



Proyecto Final
Facultad de Ciencias de la Computación - BUAP
Desarrollo de Aplicaciones Móviles – Otoño 2025
M.C. Luis Yael Méndez Sánchez ©

Objetivo: Desarrollar un sistema completo basado en Angular (frontend) y Django (backend) que permita realizar **CRUD de Eventos Académicos** realizados dentro de la Facultad de Computación, implementando validaciones estrictas, servicios HTTP, autenticación basada en roles y visualización mediante tablas dinámicas y gráficas. La aplicación será **tipo web-móvil**, es decir, **100% responsiva**.

Forma de trabajo: El proyecto es individual (la revisión se realiza a cada alumno), sin embargo, eso no impide ayudarse entre ustedes (sus equipos de trabajo).

Descripción:

1. Utilizarán el mismo proyecto web que hemos desarrollado en clase. Una vez hecho login, habilitarán en el menú lateral (sidebar) la opción de Registro de Eventos Académicos, pueden colocarla donde ustedes elijan y deben usar un **dropdown**.
 2. Cuando inicien sesión y entren a la pantalla del “home”, ya muestra la lista de usuarios (administradores, maestros y alumnos). Una vez que habiliten la opción de registrar eventos en el menú, deberá mandarlos a una nueva vista donde diseñarán su formulario de registro correspondiente.
 3. El formulario de registro contará con los siguientes campos:
 - Nombre del evento
 - Tipo de evento
 - Fecha de realización
 - Horario (Hora inicio y hora final)
 - Lugar
 - Público objetivo
 - Programa educativo (solo aparecerá si el público objetivo seleccionado es “Estudiantes”).
 - Responsable del evento
 - Descripción breve
 - Cupo máximo de asistentes
1. El formulario deberá ser validado correctamente, es decir, **todos los campos son obligatorios**.

A continuación, se describen a detalle todas las validaciones correspondientes a los campos para el formulario de materias:

2. Nombre del evento: Obligatorio. Solo se permiten letras, números y espacios. No se permiten caracteres especiales.

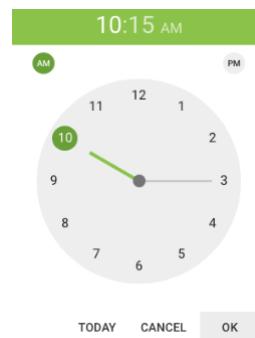
3. Tipo de evento:

Será un campo select que mostrará las opciones:

- Conferencia
- Taller
- Seminario
- Concurso

4. Fecha de realización: Solo caracteres propios que usan las fechas con formato “dd/MM/aaaa”. Se deberá utilizar un DatePicker. No se podrán seleccionar fechas anteriores al día actual.

5. Horario (Hora de Inicio y Hora de Finalización): La hora de inicio debe ser menor que la hora de finalización. Se debe implementar un DateTimePicker, el cual es un componente que da un reloj para poder elegir la hora y los minutos. Ejemplo:



Pueden utilizar dos campos: uno para la “hora inicial” y en otro input de tiempo pedir la “hora final” de la materia.

Revisar los siguiente links para entender los ejemplos:

<https://beenote.github.io/angular-material-datetimepicker/>

<https://material.angular.dev/components/timepicker/overview>

6. Lugar: Solo debe admitir caracteres alfanuméricos y espacios al escribir en el input.

7. Público objetivo:

Se mostrará una lista de checkbox con las opciones:

- Estudiantes
- Profesores

- Público general

8. Programa educativo: Se requiere que despliegue un “Mat-select” de angular material, el cual desplegará las 3 opciones de carreras que tiene la FCC:

- 1) Ingeniería en Ciencias de la Computación
- 2) Licenciatura en Ciencias de la Computación
- 3) Ingeniería en Tecnologías de la Información

Ejemplo:

Usando mat-select de angular material

The image shows two views of a 'mat-select' component. The top view is a standard input field with a placeholder 'Programa educativo:' and a dropdown arrow. Below it is a button labeled 'Seleccionar'. The bottom view shows the expanded dropdown menu with three items: 'Ingeniería en Ciencias de la Computación...', 'Licenciatura en Ciencias de la Comp...', and 'Ingeniería en Tecnologías de la Infor...'. The menu has a light gray background and a white header bar.

9. Responsable del evento: Se debe implementar un **mat-select** donde listará todos los maestros y administradores que existen registrados en la Base de Datos para seleccionar alguno de ellos. Por ende, se debe realizar un servicio para consultar toda la lista de maestros o bien toda la lista de administradores y agregarla al select (el servicio que devuelve la lista de maestros y administradores ya lo hicimos en clase).

10. Descripción breve: Campo de tipo textarea, con un límite máximo de 300 caracteres. Solo se permiten letras, números y signos de puntuación básicos.

11. Cupo máximo de asistentes: Solo se permitirá el ingreso de números enteros positivos, limitado a 3 dígitos.

12. El registro del evento académico debe hacerse por medio de un servicio el cual conectará con el backend para guardar la información a la base de datos, es decir, utilizarán las diferentes peticiones HTTP a lo largo del desarrollo de este CRUD.

13. Dentro de la pantalla del home, en el menú lateral, habilitarán una pestaña llamada **Eventos Académicos**. Al darle click, redireccionará a una nueva vista que mostrará en una tabla todos los eventos registrados en la base de datos. Para la tabla pueden usar el mat-table de Angular Material con la **paginación, filtering y sorting** (estos dos últimos tendrán que usar el campo

nombre para la búsqueda y el ordenamiento), además, es obligatorio usar inyección de componentes como en la clase.

14. La tabla mostrará cada uno de los datos registrados del evento académico, así como dos botones, uno para “Editar” y otro para “Eliminar”.

15. Editar: Cuando el **administrador** dé clic en editar, deberá abrirse y reutilizarse la vista del formulario de registro. Todos los campos deberán aparecer llenos, permitiendo modificar solo los que desee. Al actualizar, deberá mostrarse un **modal de advertencia**, como el siguiente ejemplo:



16. Eliminar: Se habilitará la opción de eliminar (**solo para el administrador**) dando click al botón correspondiente. Se abrirá un **modal de confirmación**, como el visto en clase, que presentará dos botones: **Cancelar** y **Eliminar**.

17. De esta manera se completaría el CRUD para la administración de los eventos académicos, cada una de las funciones deberán ejecutarse correctamente, se revisará el frontend, backend y la base de datos.

18. Importante: El único que tiene permisos para registrar eventos es el usuario **Administrador**, los alumnos y maestros no podrán realizar este tipo de registro, tienen que agregar la validación correspondiente para que cuando inicie sesión alguno de estos dos usuarios no les aparezca la opción de registrar eventos.

19. Todos podrán ver la lista de eventos, pero **ojo**, si eres el **maestro** al entrar a la lista de eventos **no** le tienen que aparecer en la tabla las opciones de **editar ni eliminar y tendrán que ver únicamente los eventos propios para maestros y los que sean de público general**.

20. Si eres el **alumno** al entrar a la lista de eventos **no** le tienen que aparecer en la tabla las opciones de **editar ni eliminar y tendrá que ver únicamente los eventos que sean propios para estudiantes y los de público general**.

- 21.** Verificar bien todos los **permisos de autorización** que tienen los usuarios (administradores, maestros y alumnos) se evaluará la lógica de esos permisos.
- 22.** En la pestaña de gráficas, ahora tendrán que investigar cómo se le pasan los datos dinámicamente al array (data), esto porque se requiere que las gráficas (histograma, barras, circular y dona) mostrarán solamente el total de usuarios registrados (administradores, maestros y alumnos), al entrar a esa vista se verá reflejada gráficamente la información.

Entregables:

1. Entregar su reporte del proyecto explicando las funcionalidades más importantes de la aplicación web, es decir, agregar código de las funciones relevantes, así como capturas de pantalla, el formato será en PDF.
2. Código fuente en un repositorio de GitHub, es decir, en la asignación de Teams compartirán el link del repositorio de su proyecto.
3. **Desplegar la aplicación web en un hosting gratuito:**
 - Para el frontend pueden usar Vercel (<https://vercel.com/guides/deploying-angular-with-vercel>).
 - Para el backend pueden usar Render (<https://render.com/docs/deploy-django>).
 - **Por lo tanto en el entregable deberán angular su URL correspondiente a su aplicación web.**

Revisión del proyecto:

1. Será 100% presencial, sin excepciones y solo en el horario establecido.
2. Se revisará la aplicación alojada en el hosting NO de manera local.
3. Se realizarán las preguntas correspondientes acerca del desarrollo del proyecto, así como demostrar los conocimientos en Angular y Django.

NOTA IMPORTANTE: Cualquier actitud no ética (plagio, falsedad, códigos desarrollados por IA) se penalizarán con la respectiva invalidación del proyecto y su evaluación.