

INSTRUCCIONES GENERALES

- El desarrollo de este certamen es en equipos de máximo 3 integrantes, **práctico**, correspondiente a una **entrega con presentación**.
- Debe publicar su proyecto en un repositorio publico GitHub cuyo nombre corresponda a la aplicación a desarrollar.
- El plazo máximo de entrega es martes 28 de noviembre a las 14:29 hrs.
- La entrega es exclusivamente vía aula: enlace a GitHub y sitio comprimido en rar.
- Cada grupo es responsable del respaldo de su avance/código del proyecto, de tal forma que estén preparados en caso de fallas o pérdidas de equipos computacionales utilizados para el desarrollo de la aplicación.
- La nota del certamen es asignada en el momento de la presentación. **Si un estudiante no presenta sin motivo justificado sólo obtendrá el 55% de su nota.**

ESPECIFICACIÓN DEL PROYECTO

En este certamen los estudiantes en grupos de máximo tres integrantes deberán construir un sitio web basado en Django Framework que solucione el problema planteado en el caso.

Objetivos

1. Desarrollar aplicaciones que promuevan el desacoplamiento entre sus componentes utilizando un modelo de capas.
2. Desarrollar programas que permitan presentar y modificar información interactuando con bases de datos.
3. Desarrollar programas que realicen llamadas a servicios externos invocando APIs REST.

Descripción del caso

El Departamento de Electrotecnia e Informática requiere una aplicación que permita una mejor gestión del calendario anual de hitos y actividades de las diferentes áreas del departamento, con la intención de permitir un acceso mas eficiente y oportuno a la información.

El calendario está basado en hitos que requieren almacenar la siguiente información: fecha de inicio y término del hito (considerar que existen hitos de un solo día), descripción, tipo y segmento cuyo detalle se presenta en el Anexo 1.

Se requiere que la información del calendario se presente en un sitio web que permita que la comunidad universitaria pueda navegar considerando los siguientes requerimientos:

Requisitos Funcionales:

RF01 El sitio debe contar con una página pública basada en Bootstrap que muestre en un listado los eventos del calendario del año actual, ordenados por fecha. (Ver Anexo 2)

RF02 El sitio debe permitir ocultar los eventos del año que ya no se encuentran vigentes

RF03 El sitio debe permitir filtrar los eventos de acuerdo con su tipo y a su segmento.

RF04 El sitio debe permitir a los usuarios iniciar sesión para obtener una vista personalizada del calendario según su rol. (Ver Anexo 3)

RF05 El sitio debe contar con un sistema de administración (Django Admin) que permita a un super usuario crear usuarios y asignarles un rol. Los roles del sistema están asociados a los Segmentos descritos en el Anexo 1.

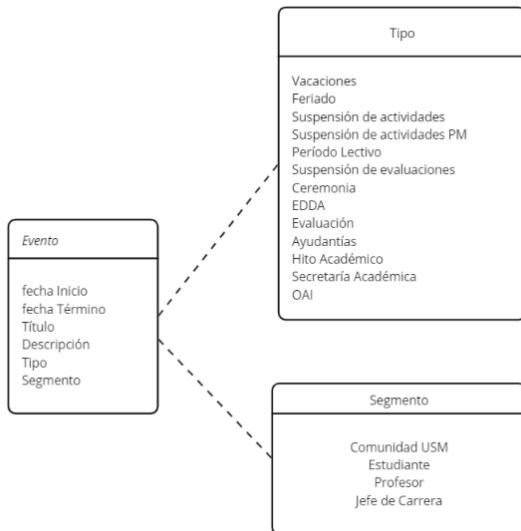
RF06 El sitio debe permitir al super usuario gestionar el calendario de eventos (mediante Django Admin)

RF07 El sitio debe proveer un API REST que entregue la información del calendario. La información debe ser entregada en formato JSON y debe permitir filtrar por año, tipo y segmento.

RF08 El API del sitio debe proveer un acceso para acciones POST, PUT y DELETE solo a usuarios autenticados cuyo rol en el sistema sea developer.

ANEXOS

Anexo 1: Estructura de un Evento



NOTAS:

1. Un evento puede estar asociado a más de un segmento y un segmento puede tener más de un evento.
2. Sólo los usuarios de los segmentos Profesor y Jefe de Carrera pueden iniciar sesión en el sistema para acceder a la vista personalizada (RF04)

Anexo 2: Layout Sitio Público

Logo

Calendario de Actividades

<año>

Segmento

Tipo

Actividades				
Hito	Fecha Inicio	Fecha Término	Tipo	Segmento

[Footer]



#004B85



#D60019



#008452



#F7AE00



#000000

NOTAS:

El sitio debe estar basado en la paleta de colores adjunta.

Logos, imágenes y otros recursos pueden ser obtenidos desde:

<https://comunicaciones.usm.cl/kit-digital-usm/>

Fuente:

Roboto
 Condensed

QWERTYUIOPASDFGÇHJKLÑZXCVBNM
 qwertyuiopasdfgçhklñzxcvbnm
 1234567890
 !".\$%&/()=?¿`+´ç,-^*"

Anexo 3: Layout Sitio por Segmento

Logo

Calendario de Actividades

<año>

Próxim

os

Eventos

Segmento

23

marzo

>Lorem ipsum dolor sit amet
Consectetur adipiscing elit. Donec
sodales facilisis euismod.
Suspendisse metus lacus, tincidunt
in nulla vel, molestie mollis enim. Ut
vulputate in turpis ut efficitur.

23

marzo

>Lorem ipsum dolor sit amet
Consectetur adipiscing elit. Donec
sodales facilisis euismod.
Suspendisse metus lacus, tincidunt
in nulla vel, molestie mollis enim. Ut
vulputate in turpis ut efficitur.

23

marzo

>Lorem ipsum dolor sit amet
Consectetur adipiscing elit. Donec
sodales facilisis euismod.
Suspendisse metus lacus, tincidunt
in nulla vel, molestie mollis enim. Ut
vulputate in turpis ut efficitur.

Segmento

Tipo

Actividades				
Hito	Fecha Inicio	Fecha Término	Tipo	Segmento

[Footer]

NOTAS:

La sección de eventos destacadas debe mostrar los próximos tres eventos del segmento al que pertenece el usuario autenticado.

La lista de eventos se mantiene como en la versión pública.

EVALUACIÓN

Restricciones

1. Aplicaciones que no compilen tendrán como nota máxima un 20.
2. Archivos con código que están en el proyecto pero que no son utilizados por la aplicación no tienen puntaje.
3. El repositorio Github del equipo debe registrar contribución de dificultad/esfuerzo equivalente de cada integrante. Repositorios con un único commit o commit de solo algunos integrantes del equipo tendrán un descuento de 20 puntos.
4. El puntaje por preguntas al estudiante es individual.
5. El estudiante o grupo obtiene nota cero si:
 - a. Copia código. La copia no está permitida, ya sea facilitando código a otros grupos, copiando código de otros grupos o copiando aplicaciones (o partes de ella) desde internet u otro tipo de terceras personas.
 - b. No entrega aplicación en el plazo establecido.

Pauta De Corrección

	ASPECTO A EVALUAR	PUNTAJE
1.	Construye un proyecto Django, organizando los archivos respetando el patrón MVT y los principios de desacoplamiento utilizando apps para cada módulo del sistema.	5
2.	Genera Modelos que cumple con las necesidades de la organización	10
3.	Realiza las migraciones correspondientes para traspasar el set de modelos a la base de datos	5
4.	Construye las Vistas necesarias para cumplir con los requerimientos del cliente	10
5.	Construye una app basada en Django REST Framework que permita desplegar la información solicitada a otros sistemas externos	20
6.	El equipo cuenta con un repositorio GIT para el control de versiones del sistema, demostrando trabajo colaborativo mediante commits, branch o merge. El aporte al repositorio de cada uno de los integrantes del equipo es balanceado y se aprecia claramente las tareas realizadas por cada miembro del equipo. (El puntaje de éste item se distribuirá en proporción al aporte de cada integrante del equipo al Repositorio GIT)	20
7.	Responde a las preguntas realizadas en la presentación demostrando conocimiento respecto a las tecnologías utilizadas, las características del patrón de arquitectura o el detalle de implementación de alguno de los componentes en los que trabajó	30
	PUNTAJE TOTAL	100