

Instituto Tecnológico de Costa Rica  
Administración de Tecnologías de Información  
Lenguajes de Programación  
Semestre 2, 2015  
Profesor: Ing. Isaac Alpízar Chacón, M.Sc.

## Tarea Programada #3 **Academic Notebook Web App (ANWA)**

### 1. Motivación

El uso de servicios de terceros ha tenido un gran auge para hacer las aplicaciones web más dinámicas y más ricas en contenido. Google ofrece una variedad de contenido a través de distintos API's para incluir su contenido en distintos servicios web.

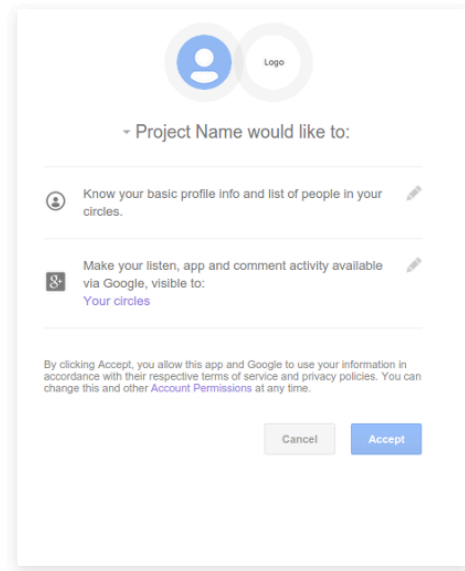
Usted ha sido designado para crear una aplicación web 100% desarrollada en HTML y JavaScript para manejar una agenda de trabajos académicos, para optimizar su tiempo y poder salir adelante con todas las asignaciones del semestre. La aplicación usará los servicios de Google para almacenar eventos en Google Calendar y para almacenar información y archivos en Google Drive.

### 2. Contexto y las funcionalidades esperadas

El sistema tendrá una página principal donde se podrá acceder a las siguientes funcionalidades:

#### 2.1 Ingreso y autenticación

La aplicación web usa y almacena información usando la cuenta de Google del usuario, para poder acceder a los servicios de Calendar y Drive, por lo que al ingresar a la página de ANWA lo primero que se deberá desplegar es la ventana para que el usuario autorice su cuenta de google utilizando el protocolo OAuth 2.0 (ver <https://developers.google.com/identity/protocols/OAuth2?hl=en>) y así poder sincronizar y acceder a la información. Por ejemplo, deberá salir una ventana como la siguiente:



Una vez autorizada la aplicación web, se podrá guardar/leer/acceder/subir información utilizando la cuenta de google del usuario.

## 2.1 Agregar evento:

El usuario podrá agregar nuevos eventos a la aplicación. Por cada evento se debe especificar (de igual manera que en Google Calendar):

- Título/resumen.
- Inicio y fin (con o sin hora).
- Lugar (especificar el valor es opcional).
- Descripción (especificar el valor es opcional).
- Recordatorio (especificar el valor es opcional).
- Invitados (mediante correo electrónico) (especificar el valor es opcional).
- Etiquetas (ver punto 2.3).
- Archivos adjuntos (ver punto 2.4) (especificar el valor es opcional).

Los eventos se deben crear siempre en el calendario principal del usuario en Google Calendar. Al crear un evento, este se crea en el calendario de Google Calendar del usuario, y también queda en la aplicación web (ANWA).

## 2.2 Visualizar eventos:

Una sección de la aplicación web será para visualizar los eventos. El usuario seleccionará alguna de las siguientes opciones de visualización:

- Día actual, se visualizan los eventos para el día de hoy.
- 7 días, se visualizan los eventos del día actual y de los siguientes 6 días.

- Mes: Se visualizan todos los eventos para el mes actual.

Además para cada una de las visualizaciones anteriores se pueden aplicar alguno de los siguientes filtros:

- Por etiqueta: El usuario selecciona una de las etiquetas disponibles, y solo se muestran los evento con dicha etiqueta.
- Por lugar: El usuario introduce un texto que representa el lugar donde se llevará a cabo el evento, y solo se muestran los eventos en dicho lugar.
- Por búsqueda: El usuario introduce un texto y se muestran los eventos cuya descripción hace coincidencia con el texto de búsqueda.

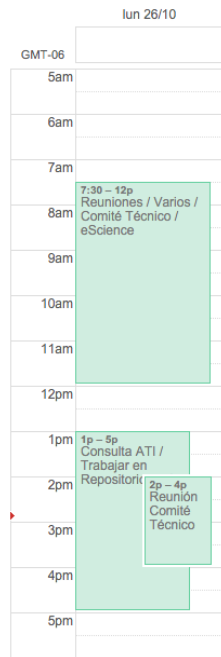
Por cada día, además, se debe incluir la información del clima. La información del clima se deberá presentar de manera gráfica, por ejemplo:

<b>27.10.</b>	<b>28.10.</b>	<b>29.10.</b>	<b>30.10.</b>	<b>31.10.</b>	<b>1.11.</b>	<b>2.11.</b>	<b>3.11.</b>	<b>4.11.</b>
Tue 	Wed 	Thu 	Fri 	Sat 	Sun 	Mon 	Tue 	Wed 
<b>17°C</b>	<b>15/13°C</b>	<b>15/10°C</b>	<b>17/10°C</b>	<b>17/8°C</b>	<b>18/8°C</b>	<b>16/7°C</b>	<b>16/9°C</b>	<b>18/9°C</b>

Se incluirá un ícono para el clima, la fecha, así como la temperatura mínima y máxima. Para obtenerlo se usará el campo de **lugar** de los eventos para ese día. Si existen más de un lugar diferente para todos los eventos del día, entonces se debe mostrar el clima en cada uno de los lugares diferentes.

En la visualización, se muestra el título/resumen del evento, y el usuario puede hacer click en cualquier evento para que despliegue toda la información asociada a él (ver punto 2.1). Por ejemplo, el usuario podría descargar los archivos adjuntos asociados al evento, o ver con más detalle la descripción del mismo.

**Nota 1:** Se deben visualizar siempre los eventos para un mismo día por medio de la hora del evento, como lo hace Google Calendar, por ejemplo:



**Nota 2:** SOLO se visualizan los eventos que han sido creados a través de la aplicación web (ANWA), no se visualizan los eventos creados directamente en el calendario del usuario de Google Calendar.

## 2.3 Etiquetas:

Se podrán crear etiquetas para poder marcar los eventos según distintas categorías. Cada etiqueta consiste de un color y de un nombre. Se pueden crear/editar/borrar etiquetas en cualquier momento. Los colores de las etiquetas deben ir asociados a los colores de los eventos de Google Calendar. En Google Calendar se pueden asociar los siguientes colores a un evento:

Color del evento           

Entonces, por ejemplo, el usuario en la aplicación web (ANWA) podría crear una etiqueta con el color azul y el texto “Lenguajes de Programación”. Cuando el usuario cree un evento, y le asigne la etiqueta “Lenguajes de Programación” el evento en Google Calendar se crea con el color azul.

## 2.4 Archivos adjuntos:

Se puede agregar cualquier cantidad de archivos adjuntos a un evento. Para agregar un archivo adjunto el procedimiento será el siguiente:

1. Deberá haber un botón que permita buscar el archivo que se quiere subir, y despliegue una ventana para seleccionarlo de forma local en la computadora del usuario.
2. El archivo se sube al Google Drive del usuario.
3. El archivo se adjunta al evento que se está creando en el calendario.

Todos los archivos que se adjunten a los eventos deben agregarse al google Drive (puede ser en una carpeta que se cree específicamente para la aplicación, o en la raíz). Y si se accede al Drive directamente los archivos deben estar disponibles.

## 2.6 Exportar:

Debe existir una opción para que el usuario pueda exportar todos los eventos creados a un archivo con formato JSON (.json). También se podrán exportar los resultados de los eventos según la búsqueda del punto 2.2. (no es realizar una nueva búsqueda, sino que cuando se visualizan los eventos, estos se podrán exportar). El formato del JSON es definido por los estudiantes, pero debe ser entendible con solo leerlo.

## 2.7 Otros:

- La aplicación web (ANWA) deberá crear una proyecto de google utilizando el Google Developers Console (<https://console.developers.google.com/>), para poder acceder a los APIs: Drive API y Calendar API. La aplicación deberá tener un nombre significativo y una imagen apropiada.
- Si necesitan almacenar información extra para el correcto funcionamiento de la aplicación web pueden almacenarla en el Google Drive del usuario, en una carpeta que la misma aplicación puede crear para tal efecto.

## 3. Aspectos técnicos

El sistema deberá desarrollarse como una aplicación 100% del lado del cliente, usando únicamente HTML + JavaScript + CSS (si se desea). No se puede utilizar ningún otro lenguaje de back-end como php, jsp, jnode, o .Net.

El sistema debe estar alojado en algún servidor gratuito en Internet, por lo que el proyecto se revisará a través de la **dirección web especificada en la documentación**. No se revisará de forma local ningún proyecto.

El sistema debe poder visualizarse correctamente utilizando el navegador web Chrome o Firefox.

Solamente se pueden utilizar las siguientes librerías externas a JavaScript: Calendar API, Drive API, + uno API de su elección para el clima y alguna librería que pueden usar para el manejo (lectura/escritura) de archivos. No se puede utilizar JQuery.

**Es obligatorio diseñar e implementar el programa bajo el paradigma de orientación a objetos.**

## 4. Documentación

La documentación es un aspecto de gran importancia en el desarrollo de programas, especialmente en tareas relacionadas con el mantenimiento de los mismos.

Para la documentación interna, deberán incluir comentarios descriptivos para cada función o predicado, que permita entender con claridad el propósito del código.

La documentación externa deberá incluir:

1. Portada.
2. Manual de usuario: Instrucciones de uso y dirección web donde esta hospedada la página.
3. Descripción del problema.
4. Diseño del programa: decisiones de diseño, algoritmos usados, etc.
5. Librerías usadas.
6. Análisis de resultados: objetivos alcanzados, objetivos no alcanzados, y razones por las cuales no se alcanzaron los objetivos (en caso de haberlos).

## 5. Evaluación

1. Documentación interna 3%.
2. Documentación externa 12%.
3. Ingreso y autenticación 5%.
4. Agregar evento 15%.
5. Visualizar eventos 20%.
6. Manejo de etiquetas 10%.
7. Manejo de archivos adjuntos 15%.
8. Exportar/importar eventos (JSON) 10%.
9. Revisión de tarea 10%.
10. Puntos extra: Usar el Google Maps Api para mostrar un mapa con el lugar donde se va a llevar a cabo un evento 8%.

## 6. Aspectos administrativos

1. Debe crear un archivo .zip ("TP3.zip") que contenga únicamente un archivo **info.txt** y 2 carpetas llamadas **documentacion** y **solucion\_computacional**, en la primera deberá incluir el documento de *word* o *open office* (no pdf) solicitado y en la segunda los archivos y/o carpetas necesarias para la implementación de esta tarea. El archivo **info.txt** debe contener la siguiente información (cualidades):
  - a. Nombre del curso
  - b. Número de semestre y año lectivo
  - c. Nombre del Estudiante x3
  - d. Número de carnet x3
  - e. Número de tarea programada
  - f. Fecha de entrega
  - g. Estatus de la entrega (definido por el responsable de la implementación de la tarea): [Deplorable|Regular|Buena|MuyBuena|Excelente|Superior]

2. Las revisiones de la tarea programada se realizarán el día **lunes 23 de noviembre**. Ese día deberá presentarse a la hora asignada para la revisión de la tarea. Las citas de revisión se darán de forma aleatoria y se darán a conocer oportunamente. Cada grupo deberá traer una computadora portátil para la revisión de ser posible. Antes de la revisión, deberá subir el archivo antes mencionado al TEC Digital en el curso de LENGUAJES DE PROGRAMACION GR 01, en la asignación llamada "Tarea Programada 3" debajo del rubro de "Proyectos programados".
3. Para la revisión se deberá traer un archivo .txt (PrimerNombreMiembro1.PrimerNombreMiembro2. PrimerNombreMiembro3.txt) que contenga todo el texto de la solución de todos los archivos del proyecto. Este archivo puede ser revisado en el sistema de Control de Plagio del TEC Digital. Todo el código de cada proyecto debe ser 100% original, y no se va a tolerar el plagio.
4. Se habilitará un foro de consultas sobre el proyecto en el TEC Digital. Por favor, plantee sus dudas o consultas por esa vía, NO se resolverán dudas por correo electrónico.
5. Las tareas deberán ser revisadas con el profesor o el asistente. Todos los miembros del grupo deberán participar de la revisión, ya que de lo contrario no se les asignará el puntaje correspondiente. La nota de la revisión es individual, el resto de la nota es grupal.
6. La tarea se hará en grupos de 2-3 personas, solamente.
7. La tarea vale un 15% de la nota del curso.
8. Antes de la entrega del proyecto se pasará en clase una lista donde deben anotar los miembros de su equipo de trabajo. Opcionalmente, pueden mandar un correo electrónico con el asunto "[Lenguajes 215] TP3:GRUPO" e indicar los nombres de las personas que conforman el grupo de trabajo para esta tarea programada. Esta información es necesaria para crear los grupos en el TEC Digital. Si no se anotan en la lista, o no envían el correo, se calificará la tarea de forma individual al miembro del equipo que la suba.