

## Valor del Caso de uso

Pasar a tener clases 100% virtual causó que todas las actividades que teníamos que llevar a cabo en la facultad, ahora es todo un “entregable” con fecha de vencimiento.

Y queda en nosotros tener en nuestras agendas, en nuestra memoria, todas esas obligaciones que tenemos para la semana que viene o incluso para dentro de 1 mes. Campus en cierta forma nos ayuda con esto enviándonos todos los días un correo con todo lo nuevo que ocurrió en mis materias durante ese día: ya sea un contenido que nos subieron, una actividad que debo entregar, una actividad que venció, etc.

El problema con el que me encontré es que me di cuenta que estaba agregando una cantidad enorme de eventos en el calendario personal de mi cuenta del ITBA. Cada vez que para alguna materia debía entregar o realizar algo, me anotaba en mi calendario la fecha límite para hacerlo. Con esta integración entre Gmail y Google Calendar, automaticé esto que me llevaba aproximadamente 30 segundos por cada evento que quería agendar.

## Explicación del caso de uso

Para poder automatizar lo comentado, todos los correos que me llegan al inbox de [hpeco@itba.edu.ar](mailto:hpeco@itba.edu.ar) con el asunto **Daily Notifications** y que viene de la casilla [campus\\_do\\_not\\_reply@itba.edu.ar](mailto:campus_do_not_reply@itba.edu.ar) lo redirijo a [activitytestitbav2@robot.zapier.com](mailto:activitytestitbav2@robot.zapier.com). La ventaja de hacer esto, es que la forma en la que me “dejaba” el cuerpo del mail para analizarlo con Python era mejor que si lo sacaba desde Gmail directamente.

Una vez que tomo el cuerpo del correo, el código de Python se puede dividir en 5 partes:

- Importación de librerías y declaración de variables
- Un loop en el que:
  - guardo el tipo de actividad
  - guardo el nombre de la actividad
  - guardo la fecha de vencimiento de la actividad
  - guardo la materia de la actividad

La explicación del código se encuentra en el [video](#).

El **output** de este código es un diccionario donde las *keys* contienen una lista de valores. La cantidad de valores dentro de cada lista variará según la cantidad de eventos que se necesite agregar:

- `activityType`: listas de cada tipo de actividad para cada evento
- `activityName`: listas de cada nombre de actividad para cada evento
- `dueDate`: listas de cada fecha de vencimiento para cada evento
- `course`: listas de cada nombre de la materia para cada evento

Este output va a ser **input** para el Excel en el que hay 5 columnas que van a estar mapeadas con la salida del código Python de la siguiente manera:

- Tipo Actividad = activityType
- Materia = Course
- Descripción = activityName
- Empieza = dueDate - 1h
- Termina = dueDate

Por cada elemento en la lista de las *keys* del diccionario, es decir por cada evento que sea necesario crear, se registrará una nueva fila en la hoja de excel.

Este nuevo registro es un trigger para la segunda API entre Google Sheets y Google Calendar. Cada vez que se registre una nueva fila, las columnas utilizadas en dicha fila serán los campos que se utilizaran para crear el evento en el calendario.

### Potencial roadmap de la solución

En principio, buscaría la forma de no hacerlo en 2 APIs por separado y que todo suceda en una.

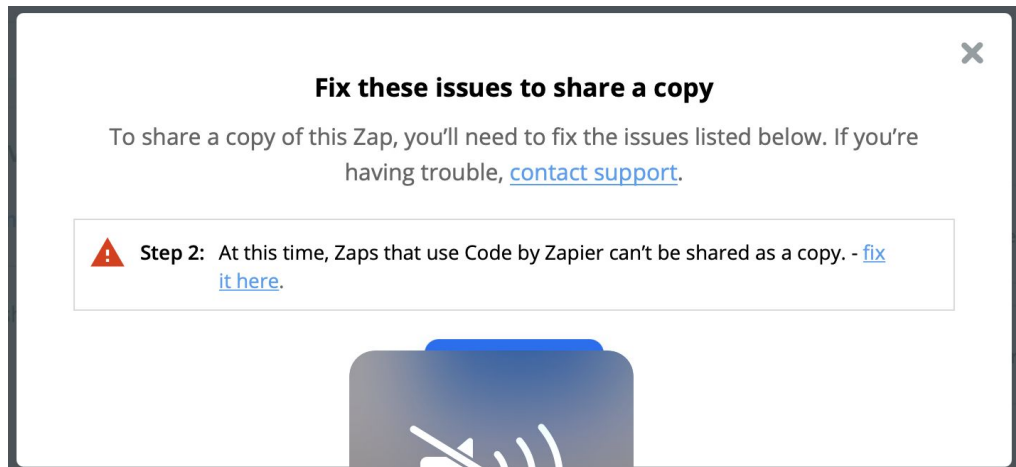
También para no llenar de mails automáticos la casilla de mi correo de la facultad, podría configurar que una vez que se redirigió el correo a la otra casilla, eliminar el mismo de mi cuenta del ITBA.

Como feature adicional, podría por ejemplo calcular el tiempo que falta para el vencimiento de dicha actividad y crear el evento con un determinado color para poder visualizar la urgencia o no de dicha actividad.

## Links de interés

### Zapier - API Gmail - Google Sheet:

No me deja compartirla porque contiene código. Adjunto el código aparte en un archivo .py



### Zapier API Excel - Google Calendar:

<https://zapier.com/shared/58c0a5271e174e37f7b5014f65529efc0e720d92>

### Video Explicativo paso a paso:

<https://youtu.be/DmqTnFuFpic>

### Link GitHub

<https://github.com/bpecoITBA/API-ITBA-Calendar-Events>