

# Flujos programados

Cuatro casos de uso:

- #1: La ejecución diaria a una hora determinada
- #2: La ejecución una vez a la semana a una hora determinada
- #3: La ejecución diaria, de lunes a viernes, a unas determinadas horas
- #4: La ejecución diaria un número máximo de veces

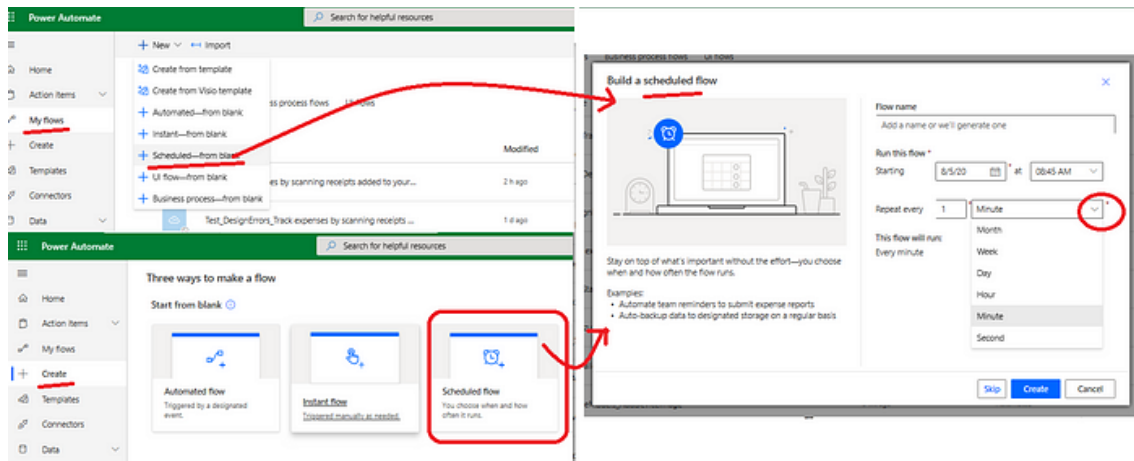
Es probable que ya estés familiarizado con los flujos programados y te estarás preguntando ¿qué pueden enseñarme acerca del trigger *Recurrence* que no sepa ya?. Si ese es tu caso, te reto a resolver los dos últimos casos de uso, que discutiremos en detalle en la segunda entrega de la serie de artículos que inauguramos hoy:

Pero, si tu experiencia con los flujos programados es escasa o nula, en ese caso continúa leyendo el presente artículo. ¡Al lío!

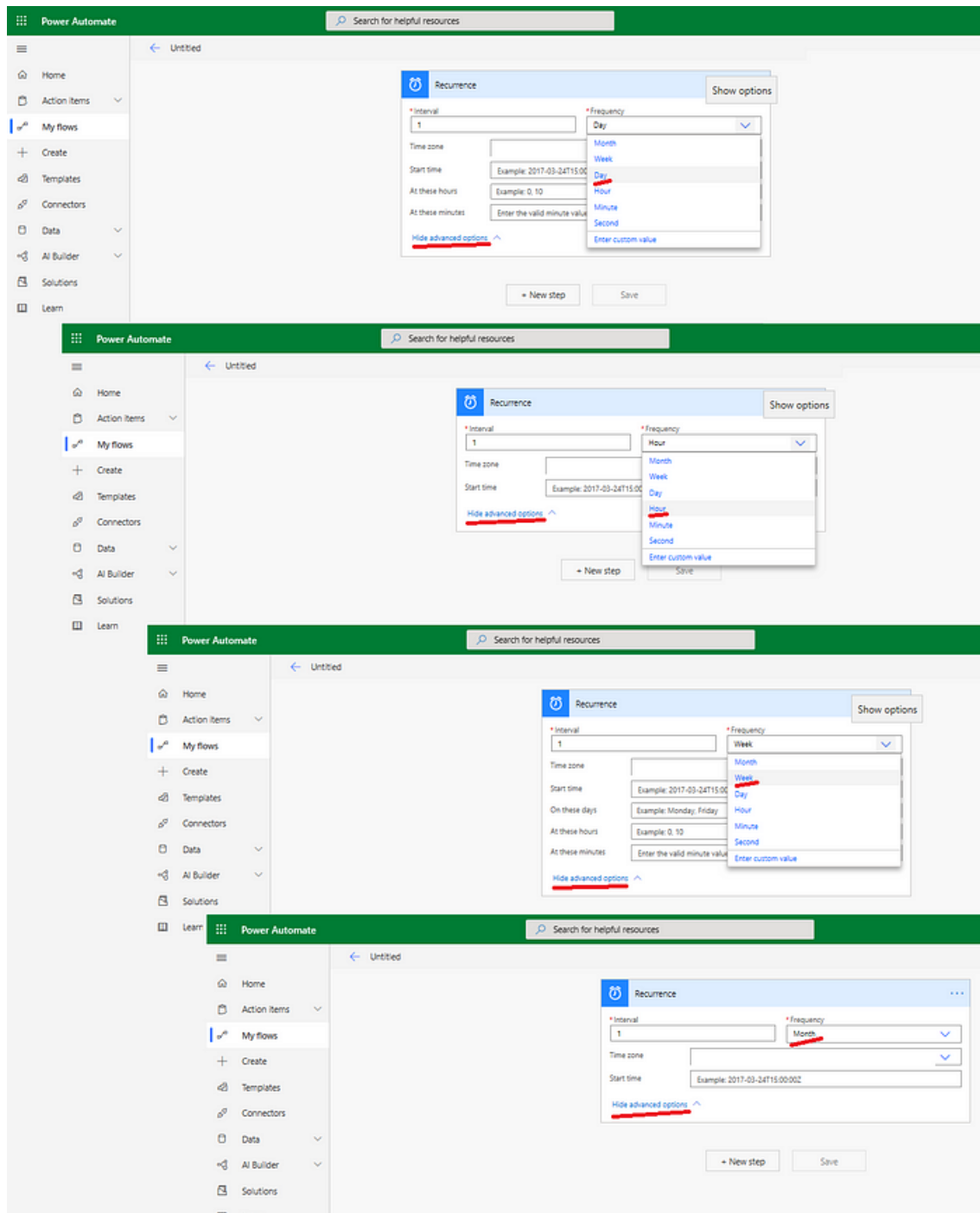
## Las interfaces

A día de hoy, Power Automate ofrece dos interfaces para configurar este trigger:

- El menú contextual al seleccionar el diseño de un nuevo Flow de tipo programado (Schedule):



- La personalización de las entradas *-inputs-* en el editor de Power Automate; siendo necesario pinchar en la opción 'Mostrar opciones avanzadas' *-Show advanced options-* de la esquina inferior izquierda para visualizar la mayor parte de ellas. Al igual que en el caso anterior, dependiendo de la frecuencia que seleccionemos, las entradas disponibles para personalización varían:



## La fecha y hora de inicio

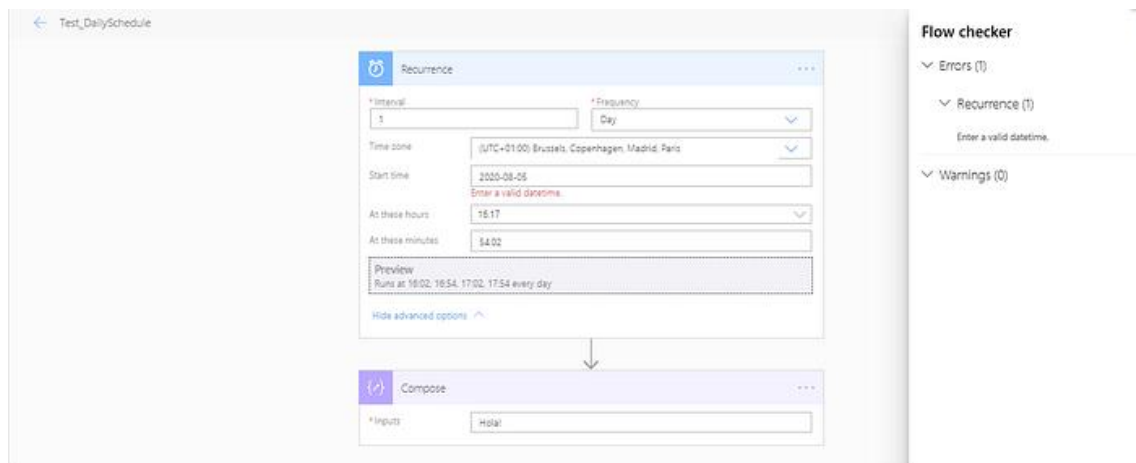
Habréis notado que, no importa la frecuencia que asignemos a nuestro *trigger* de Recurrencia, deberemos personalizar una “*Start time*” u ‘Hora de inicio’ en el formato ISO 8601, y fijar la Zona Horaria local que nos interesa. ¿Que qué es el formato ISO 8601?

De acuerdo con la [Wikipedia](https://es.wikipedia.org/wiki/ISO_8601):

La norma **ISO 8601** *Data elements and interchange formats — Information interchange — Representation of dates and times* (en español, «Elementos de datos y formatos de intercambio — Intercambio de información — Representación de fechas y horas») especifica la notación estándar utilizada para representar instantes, intervalos e intervalos recurrentes de tiempo evitando ambigüedades. Esta notación facilita la migración entre distintas plataformas.

Se sigue el criterio de especificar en orden primeramente los períodos de tiempo más largos y posteriormente los más cortos. Así, para especificar una fecha primero se escribe el año, posteriormente el mes y a continuación el día. Por ejemplo, para especificar la fecha 04 de agosto de 2020, se usa la notación «2020–08–04».

Ojo con seguir al pie de la letra la norma ISO 8601: conforme a la misma, el formato de fecha “2020–08–04” es formalmente válido. Sin embargo, nuestro desencadenador -*trigger*- exige el formato largo: con horas, minutos y segundos con el carácter reservado T (para separar fecha de hora). Si asignamos como valor ‘2020–08–04’ el editor de Power Automate nos informará de un error.

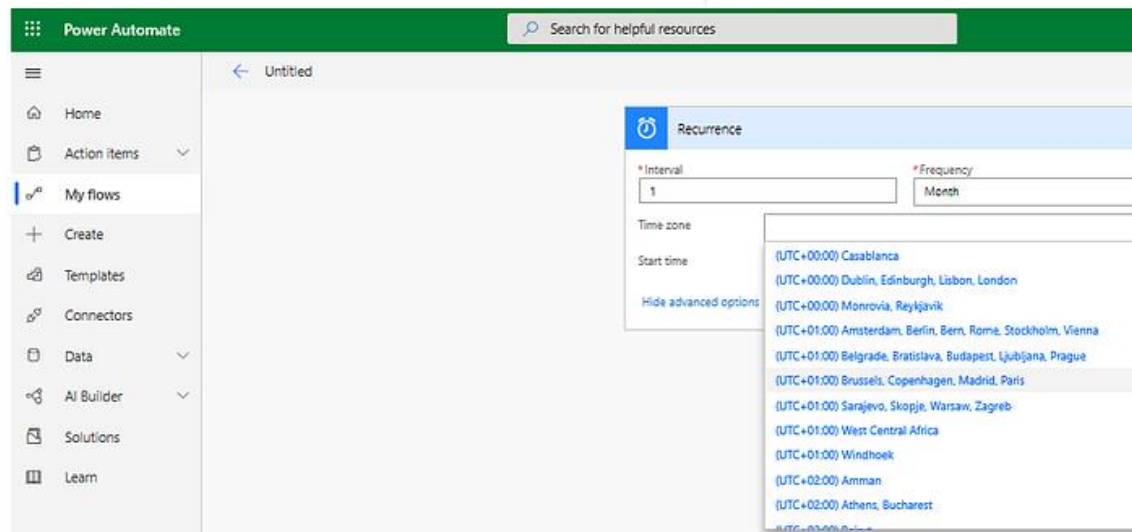
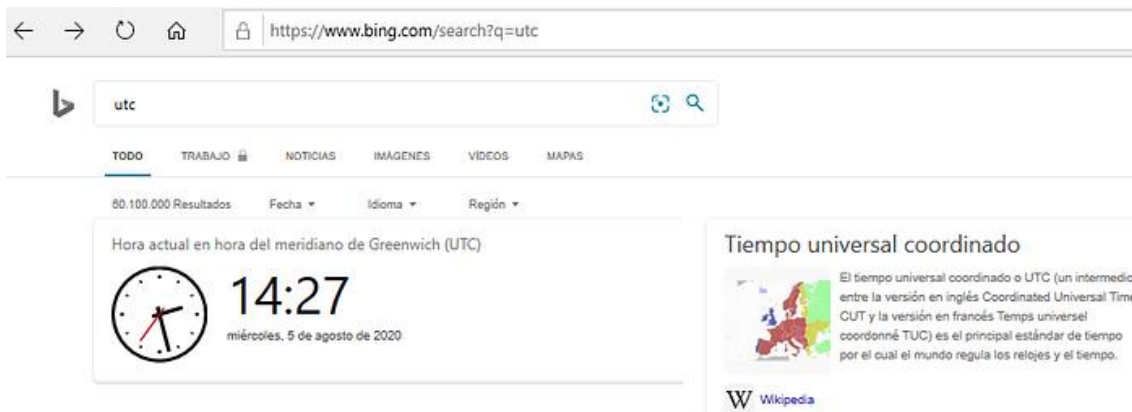


‘Start Time’ deberá personalizarse con un formato ISO 8601 ‘largo’

En el ejemplo que nos facilita el propio editor, el formato largo se completa con el carácter reservado Z (para indicar hora Zulú, o Zona Horaria UTC).

## La importancia de la Zona Horaria

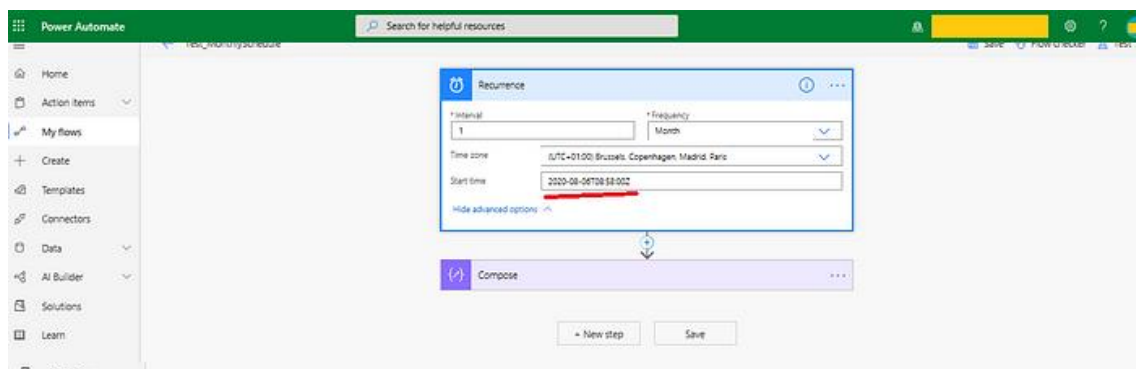
Power Automate se rige por la zona horaria UTC. Para modificar este comportamiento, deberemos seleccionar la zona horaria -*Time Zone*- que nos interese. Notad que, a día de hoy, la opción de personalizar la Zona Horaria no está disponible en el menú contextual de nuestros flujos programados, pero sí en el editor de Power Automate:



Trabajo en Madrid, luego la zona horaria local que me corresponde es (UTC+1). ¿Cuál es la tuya?

## Caso de uso #1. Frecuencia mensual a una hora determinada

Configurar nuestro trigger para su ejecución mensual es quizás el caso de uso más sencillo de abordar, pues tan sólo necesitamos personalizar una “Start time” u ‘Hora de inicio’ en el formato ISO 8601 siguiendo el formato que el editor de Power Automate muestra a modo de ejemplo, y fijar la Zona Horaria local que nos interesa:



De modo que esperamos a que sean las 08.58 AM del 6 de Agosto -hora local en Madrid- y...

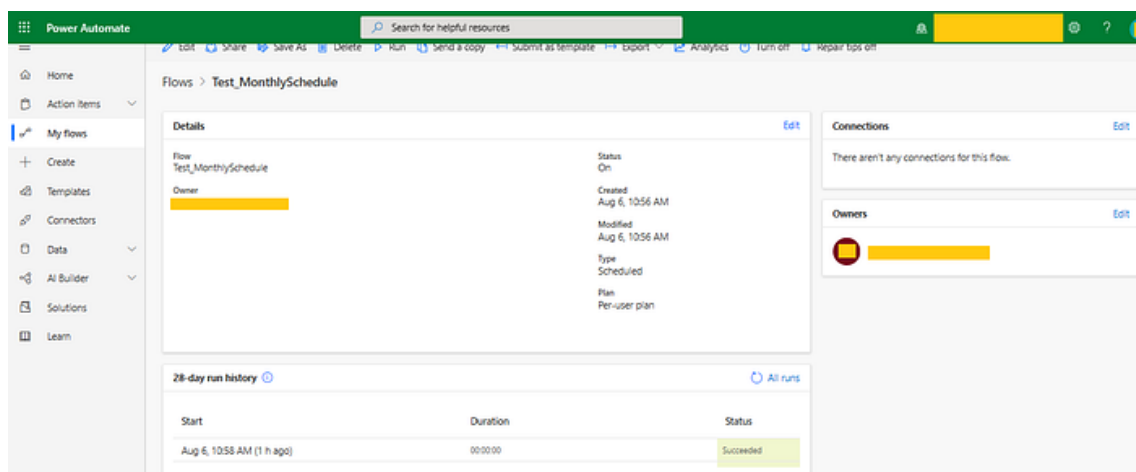
¡No se ejecutó nuestro flujo! Esperamos un par de minutos más... y ¡nada!  
Desconcertados, decidimos esperar un poco más...

¡Ejem! Tras dos horas de intensa espera, ¡se produjo la magia! Pero, ¿por qué dos horas?

Al asignar fecha y hora a la entrada 'Start Time', y acompañarla por el carácter reservado Z conforme al ejemplo, hemos indicado a Power Automate que la fecha y hora están definidas en formato UTC. Dado que en nuestro ejemplo hemos configurado la 'Start time' como...

2020-08-06T08:58:00Z

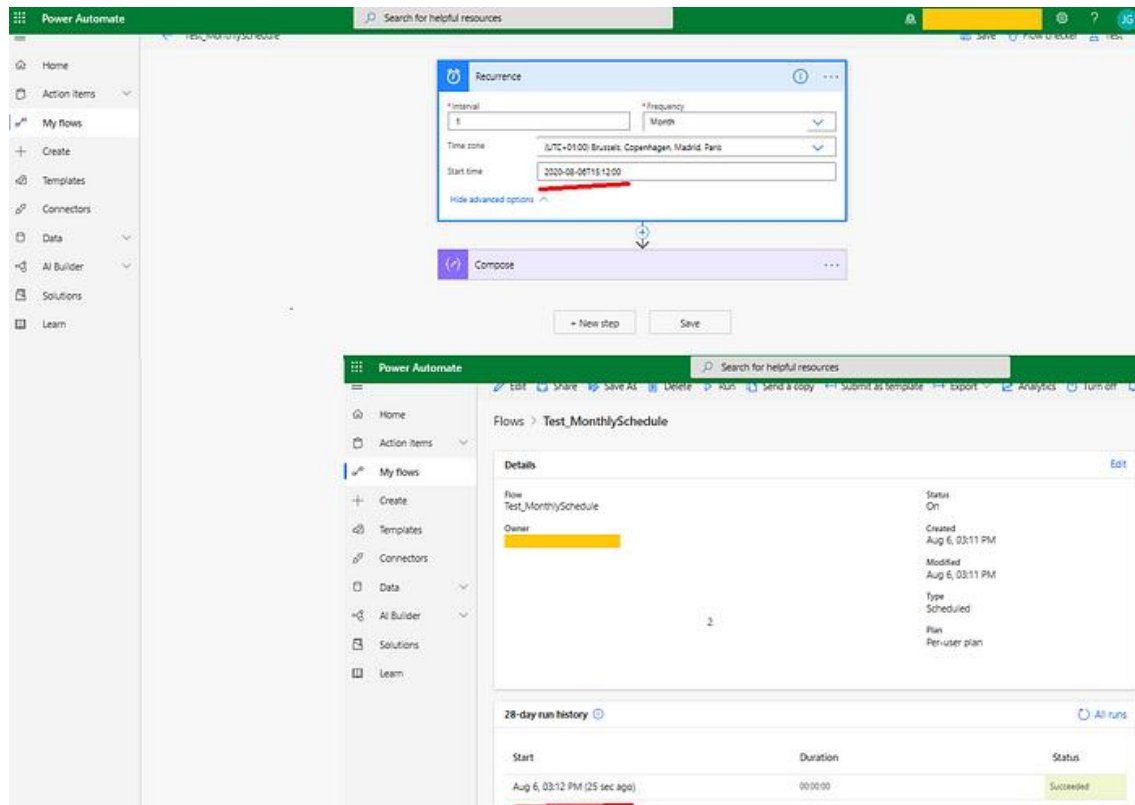
..., nuestra Zona horaria local es (UTC+1) y nos encontramos en el horario de verano (esto es, hay que sumar una hora más de diferencia respecto a UTC). Esto es, en nuestro caso de uso de frecuencia mensual, no importa cuál haya sido la Zona Horaria configurada, nuestro Flow interpreta la 'Hora de Inicio' en formato UTC y se ejecutará el día 6 de cada mes a las 08.58 AM en horario UTC, que en nuestro caso se corresponde con las 10.58 AM hora local.



La primera ejecución, el 6 de Agosto a las 10:58 AM hora local

Pero entonces ¿qué utilidad tiene la entrada 'Time Zone'? Probemos a incluir como 'Start time' un valor sin el carácter reservado Z:

2020-08-06T015:12:00



¡Si!!! En la captura de pantalla anterior, observamos que la hora de ejecución que muestra la pantalla de Power Automate (que se rige por la zona horaria configurada en Office 365) coincide con la hora fijada en la entrada 'Start time'. En mi caso, 'Time zone' asignado a nuestro trigger, y zona horaria configurada en Office 365 coinciden: UTC+1.

Esto está mucho mejor, ¿no?

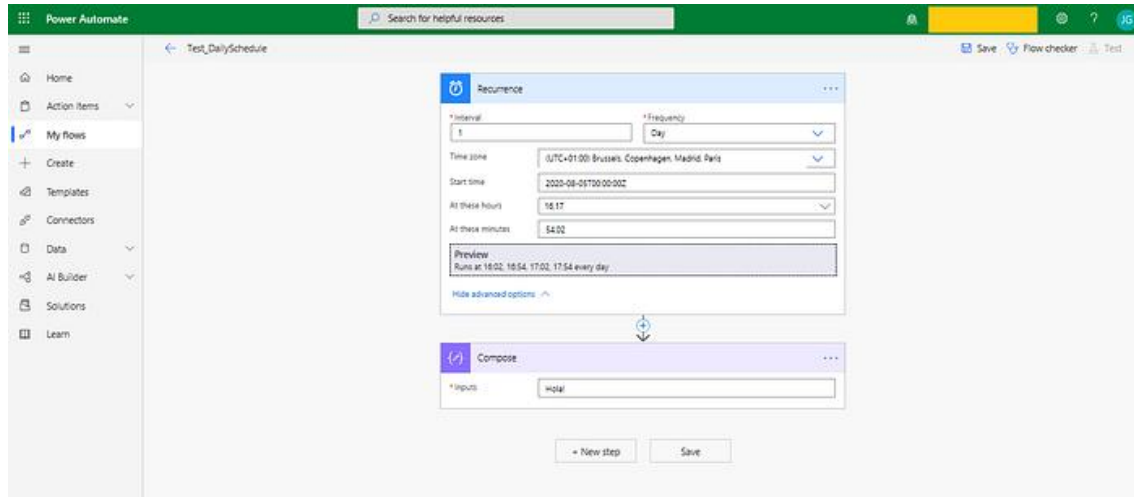
Pero ¿qué podemos deducir de todo ello? Que al añadir una 'Start time' con el carácter reservado 'Z', Power Automate ignora el valor asignado a 'Time Zone'. La razón de implementarlo así la desconocemos. Pero... basta con no incluir ese carácter reservado para ahorrarse una sorpresa inesperada.

¡A por el siguiente reto!

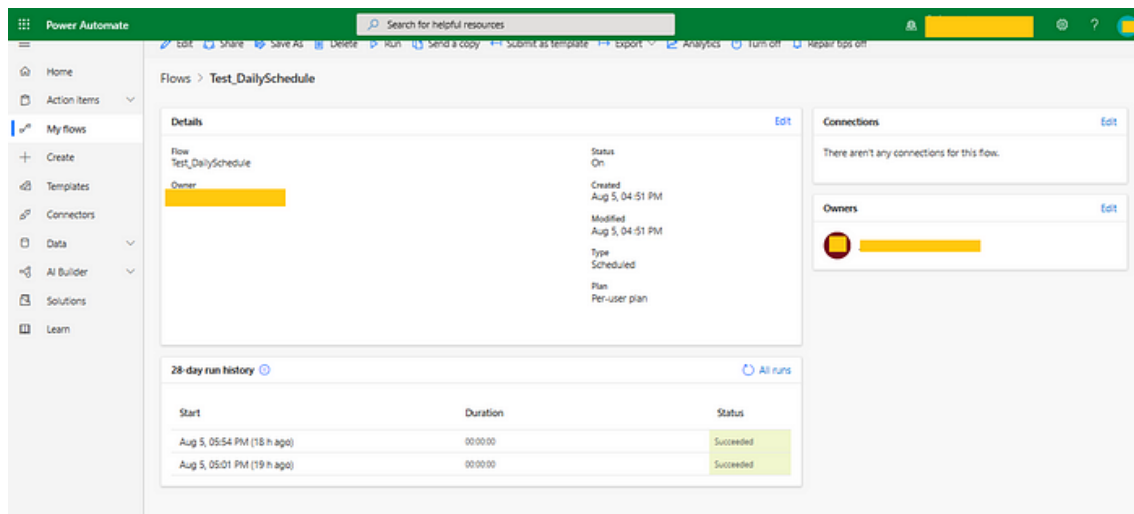
## Caso de uso #2. Frecuencia diaria a una hora determinada

Notad que a diferencia del caso anterior, al seleccionar la frecuencia diaria, el editor de Power Automate nos ofrece dos entradas más a configurar: 'At these hours' y 'At these minutes'. Esto significa que al seleccionar la frecuencia diaria, Microsoft nos ofrece la opción de configurar nuestro desencadenador -trigger- para que se ejecute a determinadas horas y minutos.

Ojo: si ese es el caso, la fecha de inicio continuará siendo la declarada en la entrada 'Start time', pero la hora de inicio se corresponderá con la primera combinación de 'At these hours' y 'At these minutes' que aplique tras la creación del flujo, dejando de aplicar la hora definida en el 'Start time'. Fijaos por ejemplo el resultado de personalizarlo para las combinaciones de un par de horas y un par de minutos:



Creamos nuestro flujo de periodicidad diaria a las 16:51. Esperamos por tanto que las dos primeras ejecuciones se produzcan ese mismo día a las 17.02 y a las 17.54 de la Zona Horaria declarada.... y para ver nuestro flujo ejecutarse a las 16.02 y a las 16.54 deberemos esperar al día siguiente.



¡Estupendo!

Si examinamos en detalle la captura de pantalla anterior, y en concreto la fecha y hora de las dos ejecuciones registradas en la consola de Power Automate, llegaréis a la misma conclusión que yo: Microsoft interpretó los valores declarados en las entradas 'At these hours' 'At these minutes' como horas correspondientes al 'Time Zone' declarado, en nuestro caso 'UTC+1'.

En esta ocasión, al declarar un valor para la entrada 'Start time', añadí el carácter reservado 'Z', pero a diferencia de lo que vivimos en el *Caso de uso #1*, Power Automate no ha aplicado la zona horaria UTC... ¿La razón?

Si asignáis valores a las entradas 'At these hours' y 'At these minutes', darán preferencia a la definición de 'Time Zone' respecto al valor que hayamos asignado ala entrada 'Start time'; esto es, la presencia/ausencia de la 'Z' no tendrá efecto alguno. Pero, si no declaráis valor alguno para 'At these hours' y 'At these minutes' -que son opcionales-, la presencia/ausencia de la 'Z' determinará comportamientos diferentes, conforme a lo experimentado en el *Caso de uso#1*.

## Caso de uso #3. Frecuencia diaria, de lunes a viernes, a unas determinadas horas

Este tercer caso de uso está tomado de una [consulta a la comunidad de Power Automate](#), donde se planteaba un escenario donde el flujo programado debía ejecutarse de lunes a viernes a distintas horas y distintos minutos:

Hi, I'm quite new to Power Automate and Flows. As a school we just started using Microsoft Teams for our students since this unfortunate situation with the Covid-19.

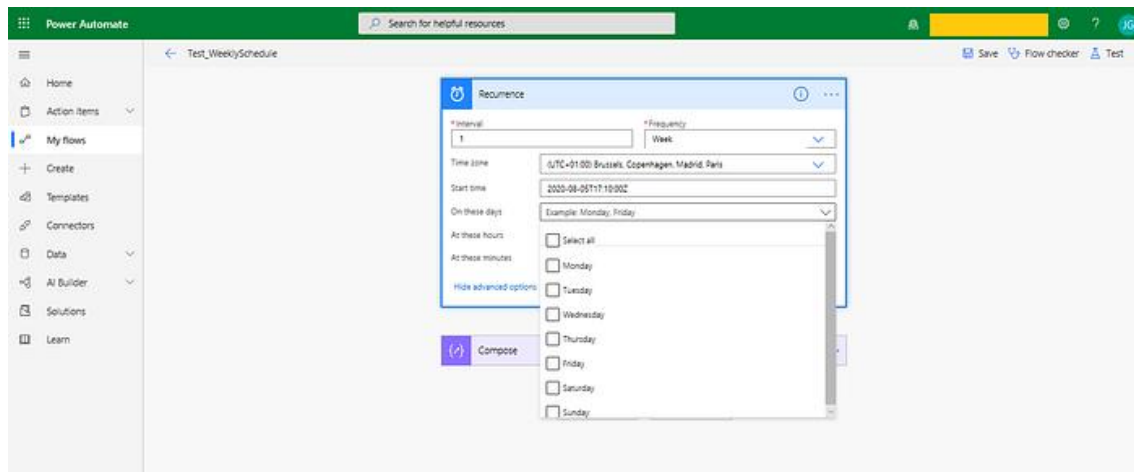
I am trying to send a notification to all users in Teams to simulate a school bell (multiple occurrences a day). I've managed to create the flow that works every working day on one certain time stamp. But I can't figure out how to do multiple a day.

I see you can select multiple hours and multiple minutes, but not a combination of that two.

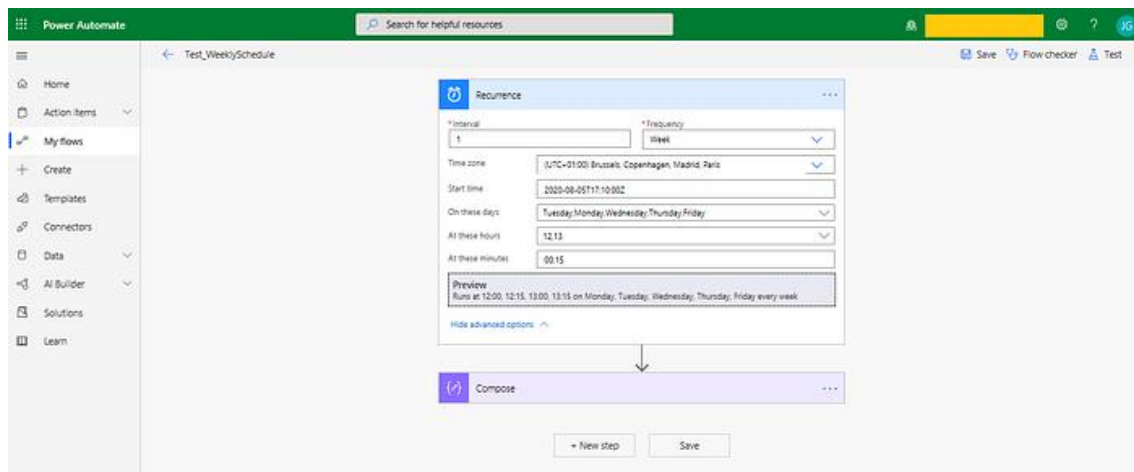
The flow has to start at these 6 time stamps: 9:00, 10:30, 10:45, 12:15, 13:00, 14:45

La primera parte (ejecución diaria, de lunes a viernes) se resuelve en el editor de Power Automate seleccionando la frecuencia semanal -*Weekly*- y desplegando las opciones avanzadas -*Show advanced options*-. para esta frecuencia comprobaremos que se nos ofrece una nueva entrada personalizable desde la que elegir uno o varios días de la semana:





El principal reto, por tanto, no es éste, sino la programación de las horas. Tomemos dos de ellas: las 12:15 y las 13:00. Si en la interfaz del editor de Power Automate asignamos los valores 12 y 13 a la entrada 'At these hours'; y a continuación asignamos los valores 15 y 00 a la entrada 'At these minutes', el resultado sería éste:



Pinta bien, ¿no? Nuestro timbre digital se activará a las 12:15 y a las 13:00.

O no tanto, porque también se activará a las 12:00 y a las 13:15.

Es posible resolver el nuevo problema que tenemos entre manos. Si no se te ocurre cómo, te daré una pista: con la ayuda de un bloque [Condición](#) y de la función WDL [utcNow\(\)](#)

Habréis reparado en que todos los flujos de ejemplo que he documentado constan de un desencadenador *-trigger-* de tipo *Recurrence* y de un único bloque de acción de tipo 'Redactar' *-Compose-*. Puede que os preguntéis qué utilidad tiene semejante combinación en vuestra vida real. ¿La respuesta? Ninguna.

Recordad que Power Automate es el arte de combinar triggers y bloques de acción, y en la vida real hay mil y un casos de uso infinitamente más prácticos. Si tenéis curiosidad por ver en acción un flujo programado resolviendo un problema real, os aconsejo que no os perdáis uno de los artículos más populares de este blog:

# Caso de uso #3. Frecuencia diaria, de lunes a viernes, a unas determinadas horas

Nuestro caso de uso está tomado de una [consulta a la comunidad de Power Automate](#), donde se planteaba un escenario donde el flujo programado debía ejecutarse de lunes a viernes a distintas horas y distintos minutos:

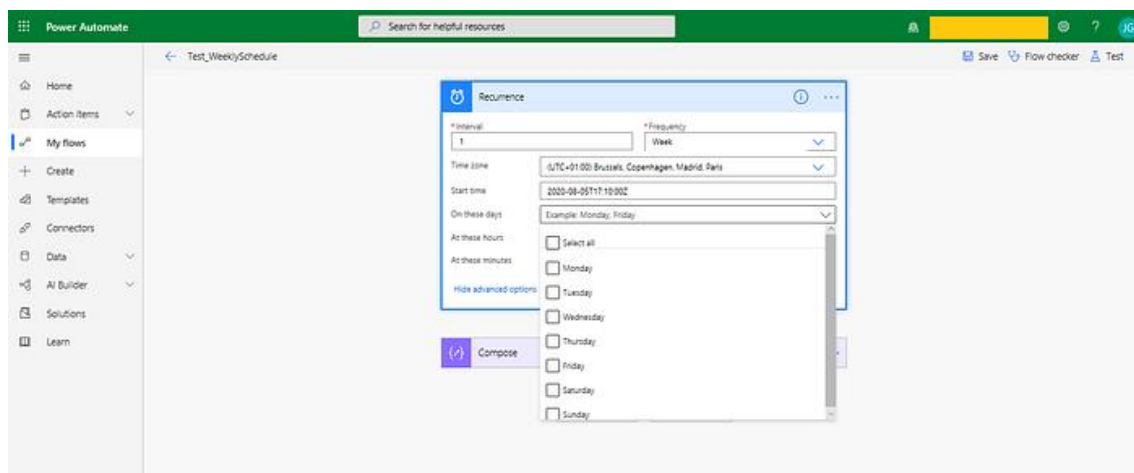
Hi, I'm quite new to Power Automate and Flows. As a school we just started using Microsoft Teams for our students since this unfortunate situation with the Covid-19.

I am trying to send a notification to all users in Teams to simulate a school bell (multiple occurrences a day). I've managed to create the flow that works every working day on one certain time stamp. But I can't figure out how to do multiple a day.

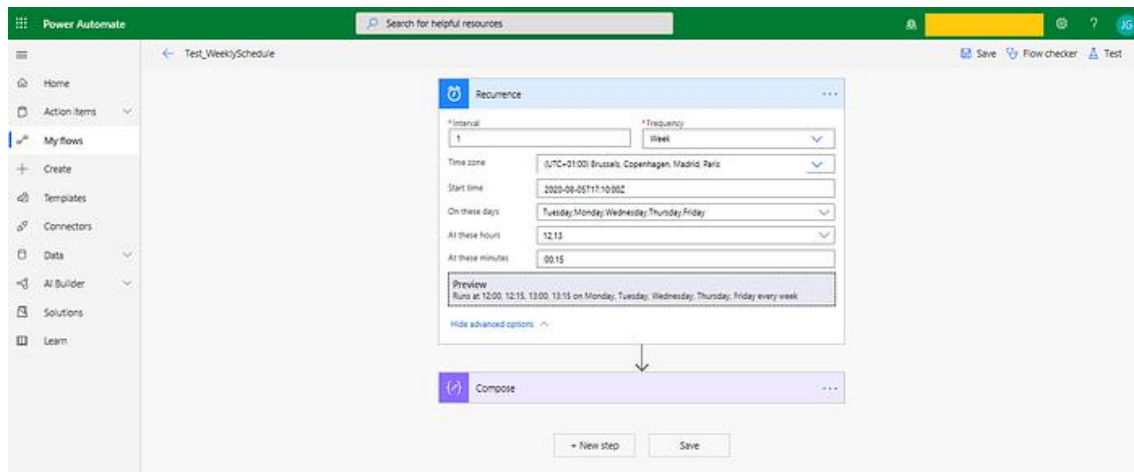
I see you can select multiple hours and multiple minutes, but not a combination of that two.

The flow has to start at these 6 time stamps: 9:00, 10:30, 10:45, 12:15, 13:00, 14:45

La primera parte (ejecución diaria, de lunes a viernes) se resuelve en el editor de Power Automate seleccionando la frecuencia semanal -Weekly- y desplegando las opciones avanzadas -*Show advanced options*-: para esta frecuencia comprobaremos que se nos ofrece una nueva entrada personalizable desde la que elegir uno o varios días de la semana:



El principal reto, por tanto, no es éste, sino la programación de las horas. Tomemos dos de ellas: las 12:15 y las 13:00. Si en la interfaz del editor de Power Automate asignamos los valores 12 y 13 a la entrada 'At these hours' ; y a continuación asignamos los valores 15 y 00 a la entrada 'At these minutes', el resultado sería éste:



Malas noticias, porque nuestro timbre digital se activará a las 12:15 y a las 13:00... pero también se activará a las 12:00 y a las 13:15.

## utcNow() al rescate

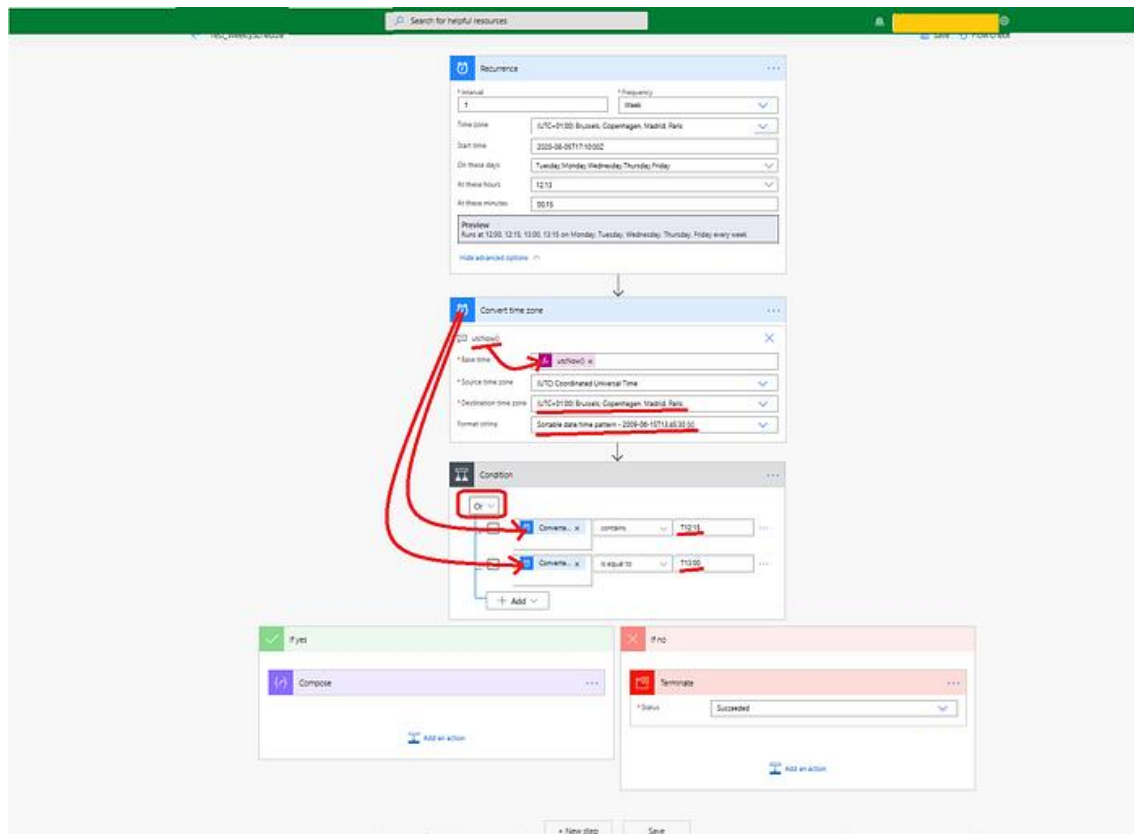
[utcNow\(\)](#) es una de las funciones más útiles en Power Automate. Permite que nuestros flujos obtengan la fecha y hora a la que esta función es invocada, lo que nos abre un mundo de posibilidades, entre ellas:

- Comparar la fecha de ejecución con la disponible en alguna fuente de datos (entradas de una lista de Sharepoint, filas de una tabla Excel, tareas en Planner, correos en Outlook, etc)
- Realizar cálculos a partir de la fecha de ejecución (los últimos quince días, los próximos 7 días) con los que filtrar fuentes de datos
- Definir Timestamps únicos, basados en fecha, hora, minuto, segundo y incluso milésimas de segundo
- ...

En el caso de uso que nos ocupa, nos interesa verificar la hora a la que se ejecuta nuestro flujo, y compararla con el listado de aquellas que deben activar el timbre, recordando que:

[utcNow\(\)](#) nos devuelve la fecha y hora en zona horaria UTC

Esto es, debemos traducir la salida de *utcNow()* a nuestra hora local. La magia correrá en esta ocasión a cargo del bloque de acción “[Convert time zone](#)” o de su correspondiente función WDL [convertTimeZone\(\)](#)



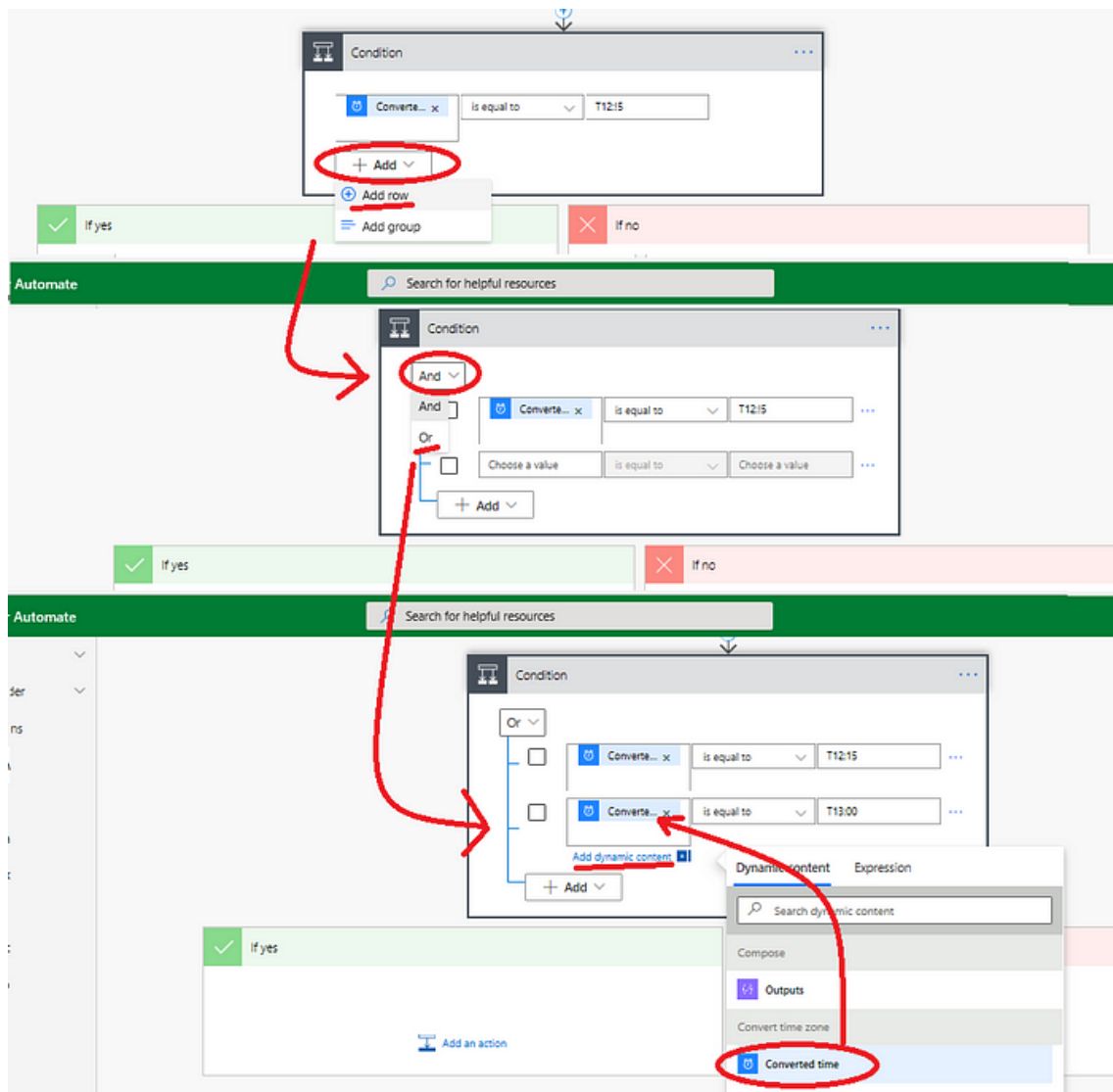
Dado que el ‘*Format string*’ que hemos escogido como salida de nuestro bloque de acción ‘*Convert time zone*’ ha sido el ‘*Sortable date/time pattern*’, continuamos trabajando en formato ISO 8601. Esto significa que habrá un carácter reservado ‘T’ a modo de separador entre la fecha y la hora. Nos apoyaremos en esta particularidad para verificar que la hora y minuto de ejecución coinciden con las que realmente nos interesan, e ignorar el resto.

Notad que nuestro Flow seguirá ejecutándose a las 12:00, las 12:15, las 13:00 y las 13:15. Sin embargo, su nuevo diseño evaluará la hora local de ejecución; tomará la rama de la izquierda (condición verdadera -*If yes*-) cuando la hora en cuestión se corresponda con las 12:15 (esto es, la cadena de texto incluya la subcadena ‘T12:15’) o las 13:00 (esto es, la cadena de texto incluya la subcadena ‘T13:00’), y tomará la rama de la derecha (condición falsa -*If no*-) cuando la hora en cuestión sea otra. Nuestro flow se ejecutará todos los días a las 12:15 y a las 13:00, pero también a las 12:00 o las 13:15. Evitaremos estas dos últimas combinaciones con este truco.

Bastará pues con:

- añadir un bloque de acción de tipo *Terminate* a nuestra rama de la derecha (condición falsa -*If no*-)
- añadir la lógica que deseamos ejecutar en la rama de la izquierda (condición verdadera -*If yes*-). En nuestro ejemplo es un simple bloque de acción ‘Redactar’ -Compose-, pero en la vida real hay mil y un casos de uso infinitamente más prácticos, como el analizado en una de las primeras entradas de nuestro blog: [“Power automate y el desafío de las columnas con formato fecha en Excel”](#)
- representar cada una de las horas que nos interesa con su propia regla de condición (condition rule) relacionando todas ellas con el operador lógico ‘O’

(Or). En el ejemplo sólo hemos incorporado dos de ellas, pero en la consulta a la comunidad de Power Automate se mencionaban hasta seis: 9:00, 10:30, 10:45, 12:15, 13:00, 14:45.



¡A por el siguiente reto!

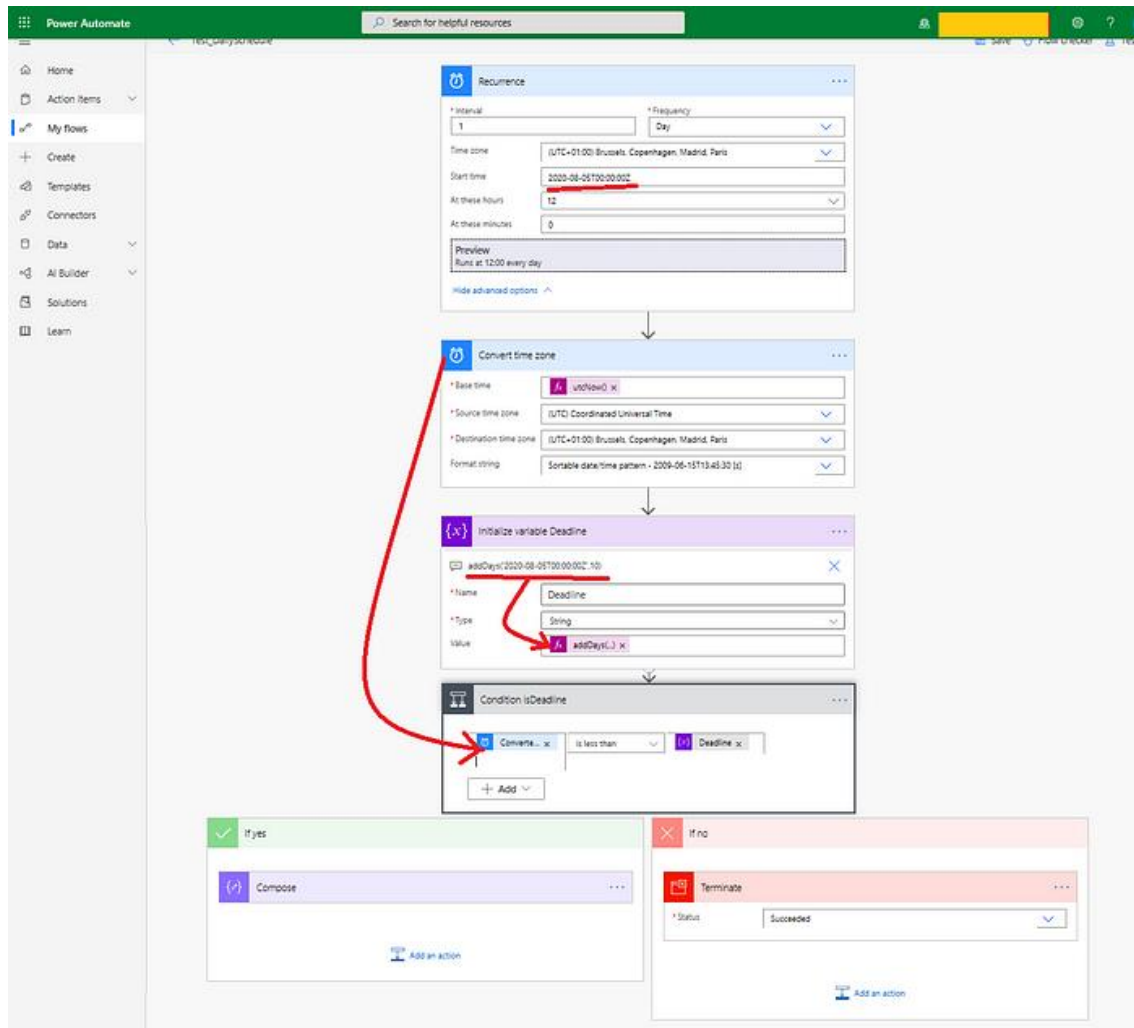
## Caso de uso#4. Ejecución diaria, un número máximo de veces

En la [comunidad de Power Automate](#) encontramos una nueva consulta relacionada con los flujos programados:

Let's say I have a flow that runs every day. I would like it to run 10 times and then stop.

¿Pista? Nada que no logremos resolver ahora que ya estamos familiarizados con el truco de las condiciones y `utcNow()` ¿Cómo?

Añadiendo diez días a la fecha de la primera ejecución de nuestro Flujo. La magia correrá esta vez por cuenta de la función WDL [addDays\(\)](#)



Esto significará que nuestro Flow se ejecutará todos los días, a mediodía, comenzando a partir de la fecha de inicio marcada (en nuestro ejemplo el 5 de agosto de 2020). La evaluación de la regla condicional tomará la rama de la izquierda (condición verdadera - *Yes branch*-) en tanto la condición se cumpla, esto es, la fecha de ejecución esté comprendida entre los nueve días posteriores a la fecha de inicio. Y tomará la rama de la derecha (condición falsa -*No branch*-) cuando la fecha de ejecución sobrepase ese umbral.

¡Prueba superada!