**Guía Técnica de Funcionamiento - Automatización de Notificaciones en Google Sheets (Versión 1.0)**

**1. Introducción**

El presente documento describe el funcionamiento del sistema de notificación automática de tareas completadas en Google Sheets. Se detallan los aspectos técnicos del código, su implementación, la estructura de la hoja de cálculo y el proceso de envío de notificaciones mediante correo electrónico. Además, se incluyen instrucciones para la configuración del activador, el cual permite la ejecución periódica del script de forma automatizada.

Este sistema está diseñado para optimizar la gestión de tareas y reducir la intervención manual en la supervisión de pedidos o asignaciones dentro de un entorno de trabajo basado en hojas de cálculo colaborativas.

**2. Objetivo del Sistema**

El objetivo principal del sistema es **automatizar la detección de tareas completadas en Google Sheets** y **enviar notificaciones por correo electrónico a los responsables asignados**. Una vez identificada la finalización de una tarea, el sistema marca la tarea como "NOTIFICADO" en la hoja de cálculo, evitando el envío de notificaciones duplicadas y garantizando un control eficiente sobre los procesos de seguimiento.

**3. Estructura y Funcionamiento del Código**

El sistema está basado en una función principal denominada verificarTareasCompletas(), la cual se encarga de:

1. **Recorrer todas las hojas** dentro del archivo de Google Sheets.
2. **Identificar tareas en curso** mediante palabras clave en la hoja de cálculo.
3. **Detectar si todas las cantidades pendientes han sido entregadas**.
4. **Enviar notificaciones por correo electrónico** al responsable de la hoja correspondiente.
5. **Actualizar la hoja de cálculo** marcando las tareas notificadas.

El código ha sido estructurado de forma modular para garantizar la escalabilidad y mantenimiento del sistema.

**4. Explicación del Código**

**4.1. Inicialización del Script**

javascript

CopiarEditar

function verificarTareasCompletas() {

var libro = SpreadsheetApp.getActiveSpreadsheet();

var hojas = libro.getSheets();

var emailPorHoja = {

"JT": "jotoapanta@telconet.ec",

"VP": "eslay777j@gmail.com"

};

* Se obtiene la instancia del archivo de Google Sheets activo.
* Se recopilan todas las hojas disponibles en el documento.
* Se establece un diccionario emailPorHoja, que contiene la asignación de correos electrónicos a cada hoja específica.

**4.2. Procesamiento de Hojas y Validación**

javascript

CopiarEditar

for (var h = 0; h < hojas.length; h++) {

var hoja = hojas[h];

var nombreHoja = hoja.getName();

if (!(nombreHoja in emailPorHoja)) {

Logger.log(" La hoja '" + nombreHoja + "' no tiene un correo asignado. Se omite.");

continue;

}

var emailDestino = emailPorHoja[nombreHoja];

Logger.log(" Procesando la hoja: " + nombreHoja + " - Enviando correos a: " + emailDestino);

* Se recorre cada hoja disponible en el archivo de Google Sheets.
* Se verifica si la hoja tiene un correo electrónico asignado.
* Si la hoja no está en el diccionario emailPorHoja, se omite su procesamiento.

**4.3. Identificación de Tareas Pendientes**

javascript

CopiarEditar

var datos = hoja.getDataRange().getValues();

var ultimaFila = datos.length;

var pendientesIndices = [];

var tareasCompletas = [];

for (var i = 0; i < ultimaFila; i++) {

var celda = datos[i][3]; // Columna D

if (celda && celda.toString().trim() === "PENDIENTE ENTREGA") {

pendientesIndices.push(i);

} else if (celda && celda.toString().trim() === "DELIMITADOR") {

pendientesIndices.push(i);

}

}

* Se obtienen todos los datos disponibles en la hoja.
* Se recorren las filas en busca de palabras clave **"PENDIENTE ENTREGA"** y **"DELIMITADOR"**, las cuales delimitan el inicio y el fin de un conjunto de tareas.

**4.4. Verificación del Estado de las Tareas**

javascript

CopiarEditar

for (var j = 0; j < pendientesIndices.length - 1; j += 2) {

var inicio = pendientesIndices[j];

var fin = pendientesIndices[j + 1];

var tareaFila = buscarTareaFila(datos, inicio);

if (tareaFila < 0) continue;

var tarea = datos[tareaFila] ? datos[tareaFila][1] : null; // Columna B

var estado = datos[tareaFila] ? datos[tareaFila][7] : null; // Columna H

* Se evalúa cada grupo de tareas delimitado por "PENDIENTE ENTREGA" y "DELIMITADOR".
* Se obtiene el número de tarea desde la columna B y su estado de notificación desde la columna H.

**4.5. Envío de Notificación y Actualización de Estado**

javascript

CopiarEditar

if (completado) {

tareasCompletas.push({ tarea: tarea, fila: tareaFila });

}

if (tareasCompletas.length > 0) {

var mensaje = "Las siguientes tareas han sido completadas en " + nombreHoja + " y deben ser reasignadas:\n\n";

var filasActualizar = [];

tareasCompletas.forEach(function (t) {

mensaje += t.tarea + "\n";

filasActualizar.push(t.fila);

});

MailApp.sendEmail({

to: emailDestino,

subject: " Tareas Completadas - Reasignación Necesaria (" + nombreHoja + ")",

body: mensaje

});

filasActualizar.forEach(function (fila) {

hoja.getRange(fila + 1, 8).setValue("NOTIFICADO");

});

}

* Se envía un correo de notificación al responsable de la hoja.
* Se actualiza la columna H con la palabra **"NOTIFICADO"** para evitar que la misma tarea sea notificada nuevamente.

**5. Configuración del Activador del Script**

Para garantizar la ejecución automática del script, se debe configurar un **activador basado en tiempo** en Google Apps Script.

**5.1. Procedimiento para Configurar el Activador**

1. Acceder a Google Sheets y abrir **Extensiones > Apps Script**.
2. Ir a la pestaña **Desencadenadores** (ícono de un reloj).
3. Hacer clic en **Añadir desencadenador**.
4. Configurar los siguientes valores:
   * **Función a ejecutar:** verificarTareasCompletas
   * **Seleccionar evento de activación:** "Basado en tiempo"
   * **Elegir tipo de tiempo:** "Minutero"
   * **Frecuencia:** "Cada 1 minuto"
5. Guardar los cambios y verificar que el activador esté activo.

**Nota:** La frecuencia del activador puede ajustarse según las necesidades del usuario.

**6. Consideraciones Finales y Mantenimiento**

* Se recomienda realizar pruebas periódicas del sistema para verificar su correcto funcionamiento.
* La asignación de correos en el diccionario emailPorHoja debe mantenerse actualizada.
* Se debe evitar copiar y pegar datos con la palabra **"NOTIFICADO"** en la columna H, ya que esto podría interferir con la detección de nuevas tareas.
* En caso de no recibir correos, se deben revisar los registros de ejecución en Google Apps Script.

**7. Conclusión**

El sistema de automatización de notificaciones en Google Sheets proporciona una solución eficiente para la gestión de tareas y pedidos, eliminando la necesidad de supervisión manual constante. Su diseño modular permite futuras mejoras y adaptaciones según las necesidades del usuario. La implementación del activador garantiza que las tareas sean evaluadas de manera continua, asegurando una gestión fluida y optimizada.