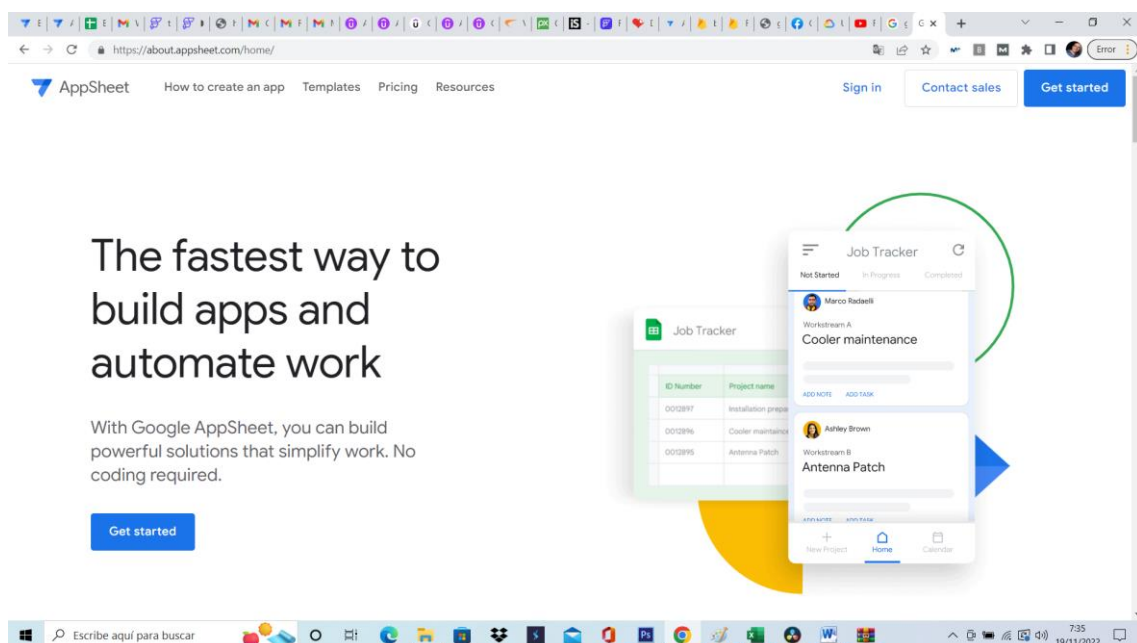


Manual de AppSheet

REGISTRO

Lo primero que debemos hacer para empezar a crear Apps en AppSheet es entrar en la web www.appsheet.com y nos logueamos con la cuenta de google.



CREAR LA BASE DE DATOS, TABLAS Y RELACIONES.

El elemento más importante de la creación de Apps es el diseño de la base de datos y en este debemos tener en cuenta varios factores.

Planificar la base de datos.

Primero tendríamos que hacer un estudio pormenorizado de los procesos y datos que queremos recoger y manejar en nuestra App. En base a este estudio tendremos la suficiente información para saber que tablas y campos necesitaremos.

Además hay que planificar que tipos de datos necesitamos en cada columna o campo de cada tabla y lo que es mas importante, que relaciones tendremos que establecer entre las diferentes tablas de nuestra base de datos. Además

debemos tener claro que tipo de relación habrá entre ellas y que campos se relacionaran entre si.

Decidir que base de datos.

Para crear las tablas tenemos dos opciones, la primera es crear la base de datos en google drive u otra fuente de datos externa como sql y luego conectar con AppSheet. La otra opción es crear una base de datos nativa de AppSheet. Las dos opciones tienen ventajas e inconvenientes.

Hay que tener en cuenta que la database nativa de AppSheet está en fase beta y tiene algunas limitaciones por lo que hay que estar a la espera de que pase a una fase definitiva. Pero bajo mi opinión cuando la database esté totalmente implantada en el sistema esta será la mejor opción aunque hay que ver cada caso por separado y mirar que nos conviene más.

Google Sheets y Google drive

En el caso de que nos decidamos por google drive y google sheets como fuente de datos debemos Ingresar al google Drive, crear una carpeta donde se almacenará todo el contenido de la app, y luego, dentro de esta, hacemos clic en nuevo y clic en Hoja de cálculo de Google.

En la parte superior izquierda hacemos clic en “Hoja de cálculo sin título” y escribimos un nombre que será el que llevará nuestra tabla en la base de datos. Debemos crear tantas tablas como tenga nuestra base de datos, en la primera fila de cada tabla escribiremos los encabezados de las columnas que se van a utilizar para la app. Una vez hechas las tablas es hora de conectarlas con AppSheet.

CREAR NUESTRA PRIMERA APP

Ingresamos, a www.appsheet.com iniciamos sesión con nuestro correo y contraseña, es decir, con nuestras credenciales de google y nos encontramos, si es la primera vez que entramos, con una pantalla de bienvenida que debemos cerrar.

En la pantalla que aparece al cerrar la bienvenida tenemos la interfaz de comienzo del editor de Apps de AppSheet. En esta pagina tenemos por un lado las Apps que hayamos diseñado anteriormente (si hemos hecho alguna) y además las bases de datos nativas que hayamos creado.

Si queremos trabajar con el editor de apps en español debemos traducir la página con la herramienta de traductor de google chrome pero lo que se aconseja es conocer los menús en ingles por si no tenemos a mano el traductor en algún momento. Además en ocasiones al traducir el editor algunos iconos de la interfaz cambian o no se ven bien.

Existe un boton Crear o Create que nos permite crear una nueva app o una nueva base de datos. Podemos crear primero la base de datos y luego conectarla con una nueva App o crear una app en blanco que te creará por defecto una tabla como base de datos. También podemos utilizar una plantilla que existe en AppSheet por defecto.

En el caso que queramos crear una nueva app a partir de datos que ya tenemos ya sea en una base de datos en drive o nativa, luego seleccionamos la opción “Start with your own data”, es decir, comenzar con nuestros propios datos.

Ingresamos un nombre para la app “New App” y en categoría escogemos una (no influye en nada la categoria que elijamos), o elegimos “otro” y clic en “choose your data” . Buscamos en Drive, nuestro archivo que creamos de excel como “Hoja de cálculo en Google” o en base de datos de AppSheet.

Menús editor de Apps vista clasica y actual

En octubre de 2022 los menús de la aplicación web del editor de apps de AppSheet se renovaron, hubo algunos cambios pero existe la posibilidad de seguir utilizando la interfaz anterior, aunque lo correcto sería acostumbrarse a esta nueva porque en algún momento la antigua puede desaparecer.

MENÚ EDITOR VISTA CLÁSICA.

En este caso nos vamos a centrar en el editor de vista clásica, este manual se está actualizando para explicar todos en los dos editores.

AJUSTAR LAS PROPIEDADES DE LA APP

1. Hacemos clic en **Info**, luego clic en **Properties** y desplegamos **App**

Propperties

2. En **Short Name**, Le ponemos un nombre a la APP

3. podemos ingresar una descripción si queremos. en **Short Description**
4. en **Default app folder**, escribimos el nombre de la carpeta que se creó anteriormente en Drive y donde se encuentra la hoja de cálculo de Google.

AJUSTAR LA IMAGEN E ICONO DE LA APP

1. Vamos a la interfaz del usuario **UX**, luego clic en la pestaña **Brand**
2. Podemos seleccionar un tema, un color
3. En **App Logo**, le colocamos un logo la dimensión del logo es de (512*512)
4. En **Style** podemos escoger el estilo del tema tanto para la parte superior o inferior de la App.
5. ACTIVAMOS LA OPCIÓN **Show Logo In Header**, para que se muestre el logo.

AJUSTAR LOS MODOS DE SINCRONIZACIÓN

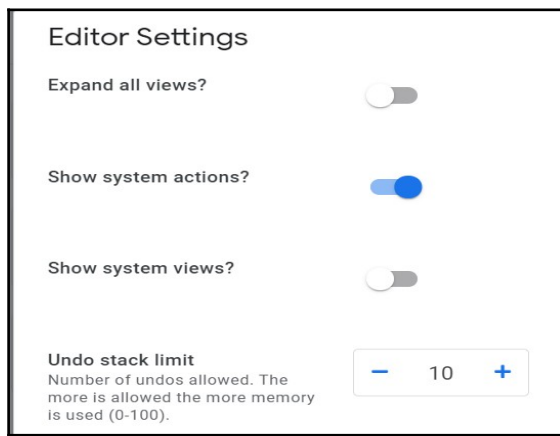
Hay que tener en cuenta que la base de datos está en la nube y la aplicación también y estas se deben sincronizar.

1. Clic en **Behavior**, clic en **Offline/Sync**.
2. Activamos la opción **Sync on start** (permite que cada vez que se inicia la aplicación se va a sincronizar, lo que nos asegura que los datos que tendrá la aplicación son actualizados, muy importante cuando la app está en varios usuario, para que estén sincronizados con la App.
3. Nota las tres opciones deben estar activas: **Sync on start**, **Delayed sync** Y **Automatic updates**
4. Nota, la opción **The app can start when offline**, debe esta habilitada cuando la opción **Sync on start** está deshabilitada, pero si habilitamos la opción **Sync on start** entonces se debe deshabilitar la opción **The app can start when offline**

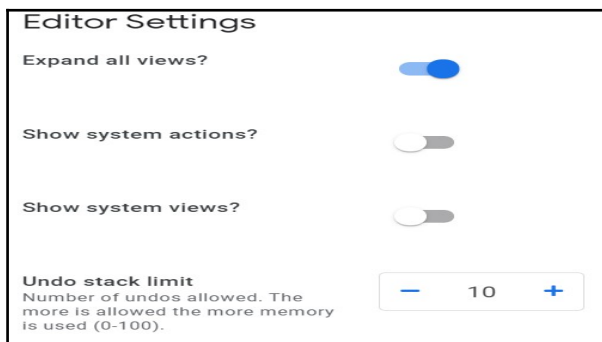
AJUSTAR LA VISTA DEL EDITOR

Nos permite desplegar las diferentes listas, para ganar más tiempo.

1. al lado izquierdo de la ventana, hacemos un clic en **Menaje** y en la pestaña **Author**, hacemos un clic en la parte inferior donde dice **Editor Settings**, donde se muestra lo siguiente:



marcamos la primera opción **Expand all views?**, y desmarcamos la segunda opción que dice **Show system actions?**, clic en **Done**, quedando de la siguiente forma:



AJUSTAR SEGURIDAD (LOGIN)

Determina el comportamiento de nuestra aplicación,

1. A la izquierda de la pantalla hacemos clic en **Security**, y en la pestaña **Require Sign-In**, (es para que el usuario entre a la aplicación y le pida que se loguee con su cuenta de correo) entonces la activamos en el caso de no estar activada.
2. Nota. Si la opción **Allow all signed-in users**, está activada, entonces permitirá que todos los usuarios que estén logueados puedan entrar a la aplicación con el correo
Nota, si queremos crear una lista de usuarios que puedan abrir la aplicación, entonces deshabilitamos **Allow all signed-in users**, y en Users, en la opción **User emails**, podemos añadir manualmente, los correos de los usuarios, y estos serán los que tendrán acceso a la aplicación.

CONFIGURAR LOS AJUSTES DE LAS TABLAS

1. Vamos a **Data** y en la pestaña **Tablet** agregamos las tablas.

2. Escogemos las opciones de actualizar, editar, eliminar o solo lectura. en la opción solo lectura no va a permitir adicionar datos y la vista se muestra en blanco.

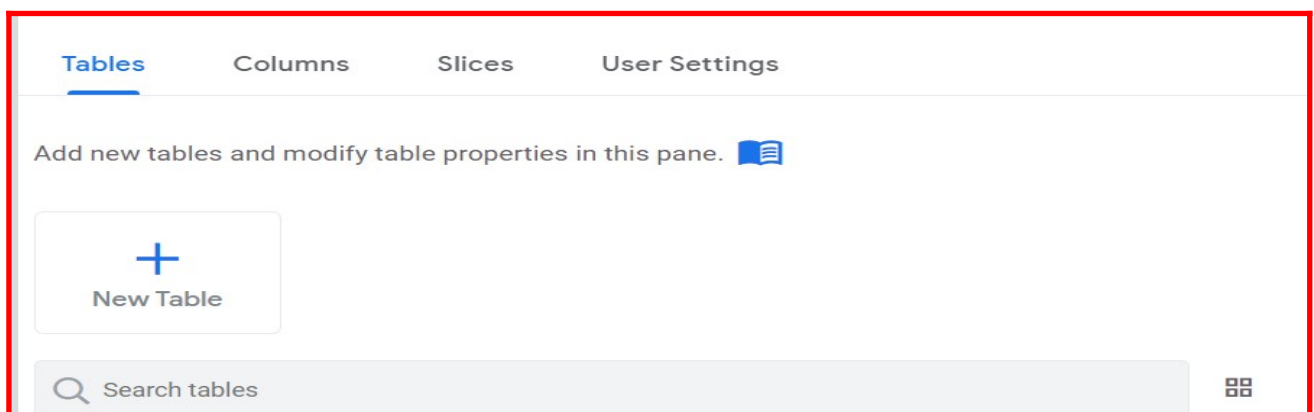
COPIAR LOS LINK DE LA APP

1. Clic en **Users** y clic en la pestaña **Link**
2. En **Install Link** se encuentra el Link para instalar la APP en el celular
3. En **Browser Link** se encuentra el link es para ver la App en cualquier navegador o browser

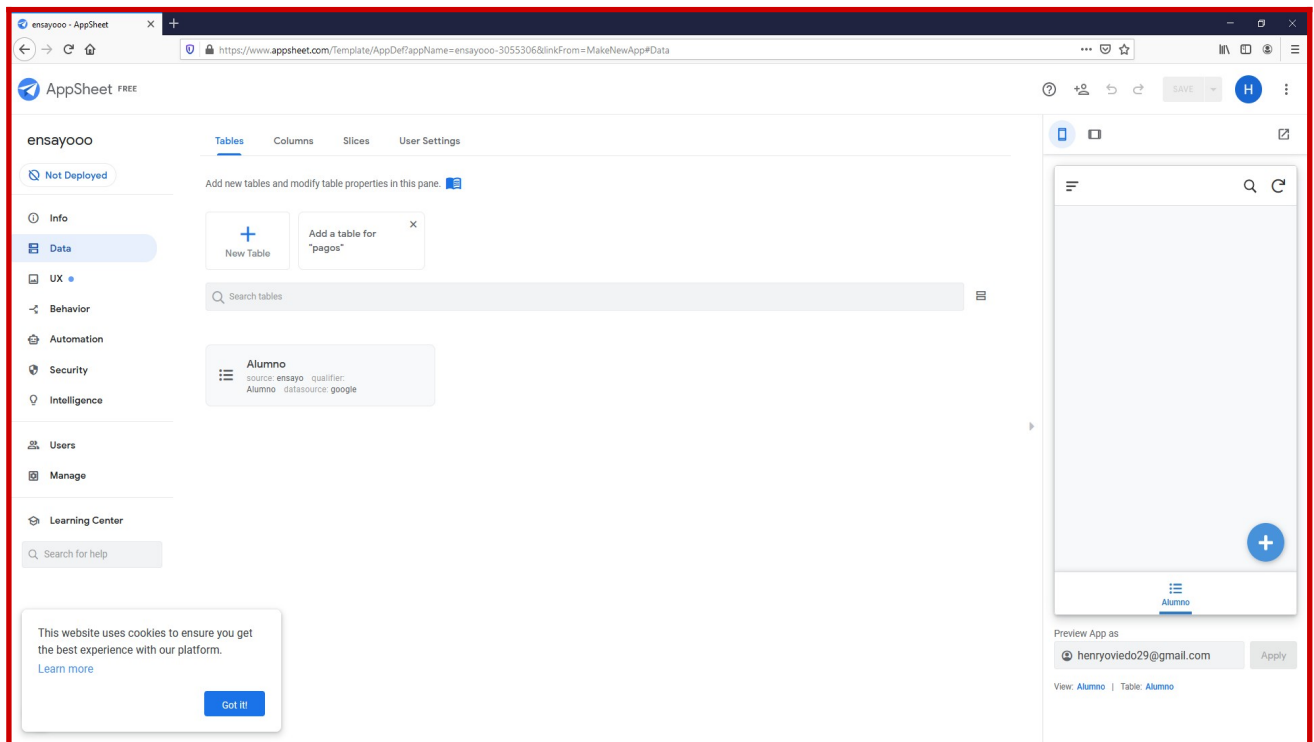
OPCIÓN DATA

TABLES

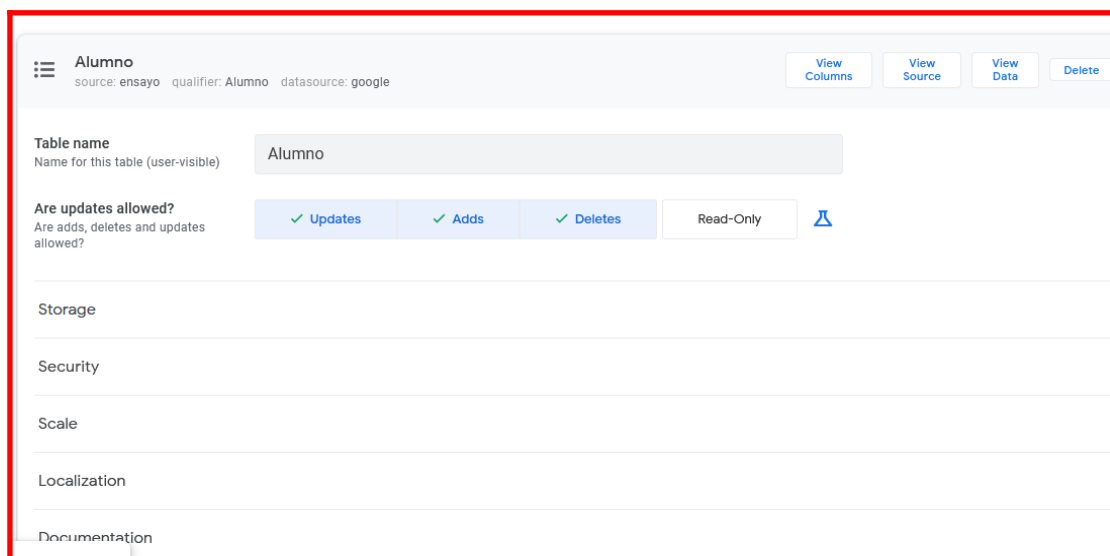
Las tablas u hojas de un Libro, se les recomienda que siempre en su primera columna debe tener un campo, que lleve el ID o clave principal de la tabla, este campo debe ser único y no debe admitir duplicado. No deben haber en una misma tabla columnas con el mismo nombre.



-Estando ubicado en “Data” (Parte izquierda de la ventana) adicionamos o agregamos las tablas (Hojas de excel), que se mostrarán en la App. y observamos a nuestra derecha como está quedando nuestra App, vista desde el celular. Por ejemplo si nuestra hoja de cálculo tuviera dos hojas una que se llamara “Alumnos” y otra que se llamara “Pagos” se vería así:



- haciendo un clic en una de las tablas por ejemplo en la tabla “Alumno”, podemos, seleccionar si esta tabla se le puede (actualizar, adicionar, borrar o si es de solo lectura) por lo general está activo Updates, Adds y Deletes



NOTA: si queremos que desaparezca el botón de más “+”, en la aplicación, es porque no queremos que se permita adicionar datos y se logra desactivando la opción “Adds”

STORAGE

Se le llama almacenamiento, en él, podemos ver:

Source Path : el nombre del libro

Table Name: Cual es la hoja que hemos seleccionado

Worksheet Name/Qualifier: Cual es la hoja que hemos seleccionado, de donde provienen nuestros datos, que en este caso es de Google,

Source ID: el ID de la ruta

Store for image and file: Las imágenes se están guardando por defecto

SECURITY

Se ven diferentes opciones que restringen la información en base a criterios, que nosotros especifiquemos mediante una condición. esta condición se escribiría qui

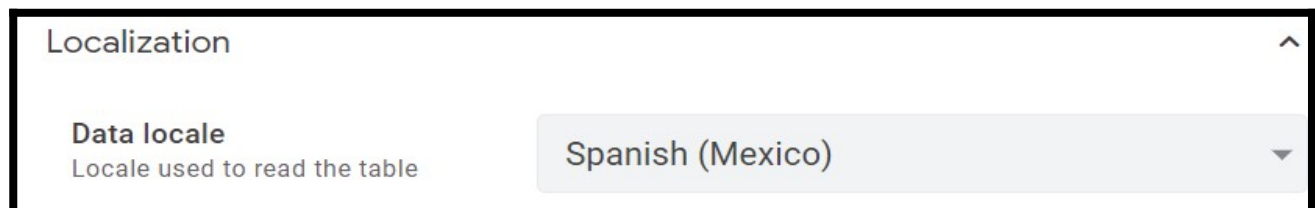


por ejemplo si quisiera ver solo las inspecciones que se han hecho hoy. Entonces tendríamos que decir que la fecha es igual a hoy.

Scale

Son opciones avanzadas para parte de planes de pago

Localization



Muestra cual es el idioma o localización que ha determinado, en el que está hecha la tabla. esto es importante, porque nuestros dispositivos y la ubicación de la aplicación deben coincidir para que no haya inconsistencia en formatos de números y datos.

Documentation

Documentation

Descriptive comment
Comments help you and your collaborators better record and understand the structure of the app

No lo habilita Apsheet, para que podamos escribir, cualquier anotación, importante si trabajamos en equipo, para dejar explicado cualquier decisión que hayamos tomado y que hayamos configurado, para que las otras personas del equipo entiendan que hemos hecho y porque lo hemos hecho.

COLUMNS

Muestra los campos de la tabla que se seleccione (los campos son los mismos títulos de columna de una determinada hoja) . Aquí podemos cambiar el tipo de dato del campo y asignar diferentes propiedades. haciendo un clic sobre el lápiz de la columna “Name” podemos hacer otras modificaciones e ingresar códigos al tipo de dato de la columna.

Tables

Columns

Slices

User Settings

☰

Alumno

5 columns: Codigo Codigo

View Table

Add Virtual Column

Regenerate Structure

	NAME	TYPE	KEY?	LABEL?
1	_RowNumber	Number	☐	☐
2	Codigo	Text	☒	☒
3	Nombre	Text	☐	☐
4	Apellido	Text	☐	☐

NAME

Corresponde, al nombre de cabecera de columnas de nuestra hoja de datos. pero si queremos que la columna se muestre con otro nombre en la vista de la aplicación, entonces podemos cambiarle el nombre pero en la opción “Display Name”

TYPE

El tipo de columna, por ejemplo alguno más usados tenemos:

Algunos Ejemplos de Tipos de Columnas	
Address	para direcciones de domicilios o residencias. esta dirección se podría abrir en Google Map, al hacer clic en ella.
Date	Para fechas
Date Time	para fecha y hora
Text	una sola línea de texto. también se usa para columnas de ID y para nombres
Long Text	Una o más líneas de texto
Number	Un número entero
Decimal	número real o con decimales
Percent:	Porcentaje
Price	valor o cantidad monetaria
Duration	Un periodo de tiempo expresado en horas, minutos y segundos
Change counter	Muestra cuántas veces se ha editado una entrada
Change Location	Se completará automáticamente con la ubicación GPS actual. muestra donde se realizó el cambio
Change Timestamp	muestra cuándo se editó por última vez una entrada.
Name	Para nombres de personas o cosas
Email	Permite escribir un Email y a demás permite validarlo si es correcto
Enum	Permite crear una lista de opciones, por ejemplo si queremos que escojan una carrera de varias que se ofrecen. Aquí podemos hacer que aparezcan las diferentes carreras para que el usuario escoja una. pueden aparecer en forma de botón donde debe hacer un clic.
Yes/No	Campos tipo pregunta, llevan al final de la pregunta el signo de interrogación
Signature	Para firmas, o cuando se requiera dibujar trazos
Ref	campo de tipo referencia, hace referencia a los datos de otra tabla. Da la propiedad relacional a una columna.
Enum	Permite establecer varias opciones para escoger. por ejemplo para escoger entre las opciones Si, No, No lo Se
Enum List	Para escoger múltiples opciones

Type de campo Enum para escoger varias opciones

PROPIEDADES DEL CAMPO DATE

una vez estando en “Columns” y después de escoger el campo “**Date**”, hacemos un clic en el lápiz que aparece antes del nombre del campo, y se abre una ventana. bajamos y hacemos un clic en “**Auto Compute**” luego en “**Initial Value**” hacemos un clic en la casilla donde se muestra el signo (=) nos aseguramos de que esté seleccionada la pestaña “**Time**” y podemos proceder hacer los siguientes cambios:

Que se muestre la fecha actual: Insertamos **Today()**

Que se muestre la hora actual: Insertamos **Time Now()**

Que se muestre la fecha y la hora: Insertamos **Now()**

OPERACIÓN DEL CAMPO REF

Por ejemplo si tenemos dos tablas, una llamada **Alumnos** y otra llamada **Programas**.

La tabla principal será la tabla “**Alumnos**” pero la primer tabla que se debe llenar será la tabla “**Programas**”

Cuando abrimos el formulario que contiene la tabla “Alumno” , y teniendo en cuenta que el ID se rellena automáticamente y que no se muestra, entonces escribimos los nombres del alumno, y para el programa simplemente al hacer un clic en la casilla programa se mostrarán los 4 programas, que se encuentran en la tabla “Programas” donde debemos escoger uno.

Tabla Alumnos

ID Alumno	Nombres	Programa	Fecha de Nacimiento
a01	Diego campos		
a02	Luis Duque		
a03	Maria Prieto		
a04	Pedro Coral		

Tabla Programas

ID Programa	Nombre del Programa	Modalidad
p01	Sistemas	Presencial
p02	Pedagogía	Presencial
p03	Inglés	Virtual
p04	Contabilidad	Virtual

En la tabla “**Alumnos**” y en el campo “**Programa**” en “**Type**” seleccionamos “**Ref**” para este tipo de campo. hacemos un clic sobre el lápiz (al lado izquierdo) y observamos que en “**Column Name**” se selecciona el nombre de la columna de la cual se quiere seleccionar el programa; en este caso es la columna “**Programa**” de la tabla “**Alumnos**” y en “**Source Table**” se escoge la tabla que contiene la columna donde se quiere seleccionar el programa; en este caso es la tabla u hoja “**Programas**”

FORMULA

Aquí se definen qué valores serán mostrados del respectivo campo, es decir podemos hacer algún tipo de filtro. por ejemplo si queremos que de la columna programa únicamente me muestre los estudiantes del programa de sistemas.

SHOW

Determinan los campos que se muestran en el formulario.

INITIAL VALUE

Aquí, podemos condicionar para que un campo tome un valor inicial por defecto.

HACER QUE UN CAMPO ID SE AUTO RELLENE EN EL FORMULARIO

En “Initial Value” del campo ID, hacemos un clic y en la nueva ventana que aparece escribimos “UNIQUEID()” Hay que tener en cuenta, que este campo, debe ser tipo Texto, para que este campo no se muestre en el formulario, desactivamos la opción “Show”.

EDITABLE

Si desactivamos esta opción, este campo en el formulario no se podrá editar.

REQUIRE

Si está activado, obliga a llenar este campo en el formulario.

DISPLAY NAME

Aquí, podemos cambiar el nombre del campo, si queremos que este nombre se muestre en el formulario.

DESCRIPTION

PODEMOS DESCRIBIR, QUÉ PROPIEDADES LE ESTAMOS DANDO A ESTA COLUMNA Y PORQUE.

SEARCH

Permite hacer una búsqueda por un determinado campo si se encuentra activada.

SCAN

Sirven para detectar códigos de barra y códigos QR

NFC

Tecnología que permite que al acercar el celular, podemos tener los datos.

PII

Se utiliza para datos sensibles, que requieren máxima protección.

SLICES

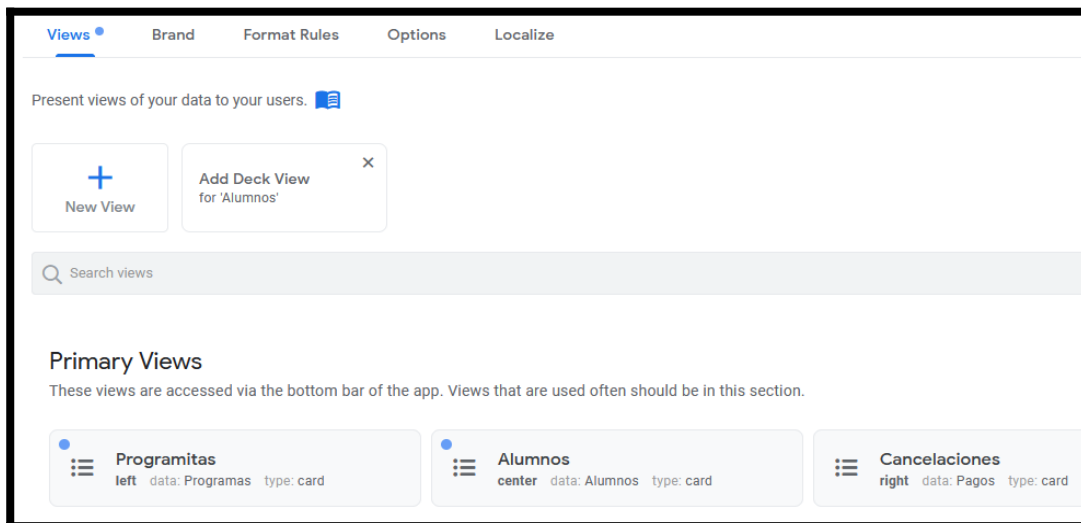
Se utiliza para crear filtros o recortes de una tabla.

CREAR UN FILTRO QUE MUESTRE LOS PAGOS DE LA FECHA ACTUAL

1. Clic en **Data** y Clic en la pestaña **Slices**,
2. Clic en **New Slice**, en **Slice Name**, ingresamos un nombre para el filtro y en **Source Table** seleccionamos la tabla que se quiere filtrar
3. En **Row filter condition**, escribimos la expresión o código del filtro, en nuestro caso escribimos: `= [Fecha] = TODAY()`
4. En **Slices Columns**, escogemos qué columnas se quieren ver
5. en **Slice Actions**, podemos escoger si queremos que en el slice podamos borrar o editar; o de lo contrario lo dejamos **Auto**
6. En Update Mode, podemos restringir en la aplicación que solamente se puedan, actualizar, borrar, ver o que sea de solo lectura.
7. ahora hacemos un clic en la interfaz del usuario que es la opción **UX**, y en la pestaña **View**, hacemos clic en **New View**, en **View Name**, le damos un nombre, que será el que se muestra en la aplicación en la parte inferior.
8. En For This Data, seleccionamos el nombre del filtro que se creó previamente.
9. En **view type** escogemos la apariencia ejemplo **Desk**
10. En **position** escogemos una posición para el nombre de la nueva vista

OPCIÓN UX

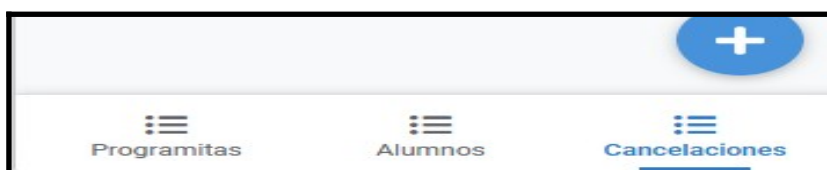
Permite modificar la apariencia o vistas de la App



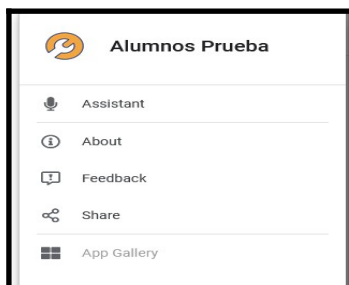
PESTAÑA VIEWS

Hay varios tipos de menu:

Primary Views : encontramos las vistas que se muestran en la parte inferior de nuestra aplicación.

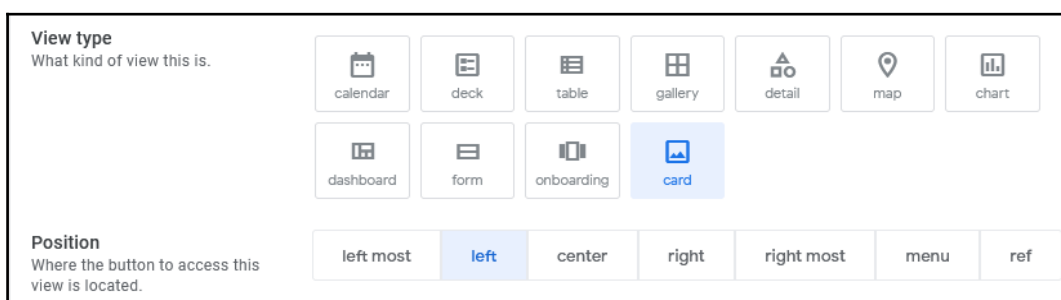


Menu Views : encontramos las vistas que encontraremos en el menú hamburguesa de la app



PRIMARY VIEWS

Views Type



Los 5 primeros botones de la parte inferior (**left most - left - center y right most**) corresponden, a los menú de la App de la parte inferior. el botón **menú**, permite que el menú que se encuentre seleccionado, de la APP (los que están en la parte inferior de la APP), haciendo clic en el botón **menú** lo mueva al menú hamburguesa. El botón **Ref** permite, que si está seleccionado uno de los menú que se encuentran en la parte inferior de la App este lo ocultará

LOS TIPOS DE VISTA EN LA APP

Nota: dependiendo de la vista que se seleccione, inmediatamente, aparecerán en en la parte inferior de las vistas, todas las propiedades que se le puedan modificar a la vista seleccionada.

hay que aclarar que tenemos tres opciones, que también se muestran en la parte inferior pero que serán, comunes para todas las vistas como son: **Display - Behavior y Documentation**

deck : Permite añadir imágenes, y hasta un segundo título en la APP y nos permite, ver los botones de acceso directo como editar, eliminar.

table: Muestra el formulario dispuesto en tabla.

gallery: Muestra la imágenes mucho más grandes con un título

detail : muestra solo una imagen y se puede navegar de izquierda a derecha

map : nos muestra un mapa dependiendo de la ubicación y dirección que se tenga en el formulario

chart : muestra un gráfico de la columna que se haya escogido. una vez que se ha hecho clic en chart, se muestran varias opciones y escogemos la opción **Chart Columns** y seleccionamos la columna, para que en la App se muestra el gráfico

dashboard : Muy importante. porque permite ver detalles de casi todas las tablas, por medio de módulos, esta opción es viendo la App desde el navegador, no desde el celular. adicionando las opciones en **View entries**

AGRUPAR O CATEGORIZAR LOS DATOS

1. Estando en la vista “**DESK**” en la opción **Group By** hacemos clic en **Add** y escogemos por cual campo se quiere agrupar.

MOSTRAR ALGUNOS BOTONES DE ACCION

2. Estando en la vista “**DESK**” en la opción **Show Action Bar**, la activamos, luego en **Actions** podemos seleccionar los botones de acción que queremos mostrar. por ejemplo como el de eliminar editar y otros.

CONTAR LOS DATOS AGRUPADOS

1. Estando en la vista “**DESK**” en la opción **Group aggregate** escogemos **COUNT**

ESCOGER LOS NOMBRES DE CAMPOS QUE SE QUIEREN MOSTRAR

1. Estando en la vista “**DESK**” en la opción **Primary header** y en la opción **Secondary header**, escogemos los campos a mostrar.

APLICAR EVENTO CUANDO DESLIZAMOS A LA DERECHA O IZQUIERDA DE LA PANTALLA DEL CELULAR

1. Estando en la vista “**DESK**” en la opción **Behavior**, luego en **Event Actions** y en **Row Swiped Left (beta)** y **Row Swiped Right (beta)** escogemos lo que quiera que pase al suceder este evento, por ejemplo hacer una llamada o editar un dato.

CAMBIARLE LA IMAGEN O LOGO A UN MENÚ DE LA APP

3. Seleccionamos el menú de la APP
4. Seleccionamos la vista
5. en la propiedad **Display** y en **Icon**, escogemos la imagen.

CAMBIARLE EL NOMBRE DEL MENÚ DE LA APP

6. Seleccionamos el menú de la APP
7. Seleccionamos la vista
8. Seleccionamos la propiedad **Display**
9. En **Display Name** escribimos el nuevo nombre entre comillas

APARTADO BEHAVIOR

Permite elegir un comportamiento de nuestra vista.

VISTA DETAIL VIEWS DE LA OPCIÓN UX

EJEMPLO DE CAMBIOS EN LA VISTA DETAIL

1. estando en **UX** y en la pestaña **View**, copiamos o creamos una nueva vista. En **View Name** asignamos un nombre para la vista. y en **For this data**, seleccionamos la tabla de la cual se creará la vista.
2. En **View Type** escogemos el tipo de **vista Detail**, y en **Posición** hacemos clic en **Ref**, para que esta vista se abra desde otra vista
3. en **View Option** en **Main Image**, seleccionamos la imagen o foto de la tabla.
4. En **Header Column**, podemos escoger el nombre para que se muestre sobre la imagen.
5. En **Sort By** podemos escoger el orden en que se van a desplazar los registros a los lados. por ejemplo podemos colocar **Nombre** y **ascendente**
6. en **Column order**, podemos escoger que columnas queremos mostrar y en qué orden. Aquí simplemente las añadimos
7. En Display mode podemos escoger un modo: automático, normal, centrado, sin cabecera o lado a lado. Normalmente lo mostramos **Centered**, centrado.
8. En **Image style**, permite cambiar el estilo como se ve la imagen
9. Para activar o desactivar el desplazamiento de la imagen de derecha a izquierda o viceversa modificamos **Slideshow mode**, por lo general lo dejamos activo.
10. En Icon, podemos aplicar un icono para la vista, pero si está activado **Ref**, no justifica colocar un icono ya que esta vista no se podrá ver.
11. En **Display Name** podemos colocar un nombre o una fórmula que le haga un cambio
12. En **Show if**, podemos utilizar una condición para hacer que se muestre o no se muestre esta vista.

13. La opción **Quick edit columns**, nos permite, que si hay un campo que permita escoger entre varias opciones, ejemplo un campo enum. no permitirá mostrar las opciones para poder editar o escoger otra opción.

NOTA: Esta **vista Detail View**, como está seleccionada en **Ref**, no se muestra en el menú inferior, solo se muestra cuando, al abrir la vista principal que está en el tipo de vista **Desk** y que contiene la misma tabla de la vista **Detail**. y al hacer clic en uno de los alumnos, entonces se puede observar esta vista en la App.

VISTA DECK VIEWS DE LA OPCIÓN UX

Group by: permite utilizar agrupadores, por ejemplo en el caso de los alumnos los podemos agrupar de acuerdo al programa o curso que están realizando.

Group aggregate: Al seleccionar **Count** nos permite contar los alumnos por cada uno de los programas en que se agruparon.

Main image: permite escoger la imagen principal que se quiere mostrar en esta vista en nuestro caso es la foto del alumno.

Primary header: Permite escoger un primer campo a mostrar en nuestro caso será el nombre del estudiante, se mostrará después de la foto a lado derecho

Secondary header: Permite escoger un segundo campo a mostrar, en nuestro caso será el documento del estudiante. que se mostrará debajo del nombre del estudiante

Summary column: Permite escoger un tercer campo a mostrar en nuestro caso será el celular que se mostrará a la derecha de la App.

Nested table column: es una tabla adicional que nos permite mostrar otros datos.

Image shape: Cambiamos la forma de la imagen que se muestra, puede ser cuadrada, redonda, o de gran tamaño.

Show action bar: nos permite escoger qué botones queremos mostrar, al interior de la App



Actions: Aquí adicionamos únicamente los botones que se quieren mostrar

En **Display** en **Icon**: agregamos un icono para la App.

Display name: podemos cambiar el nombre bajo una fórmula o condición

Show if: para mostrar u ocultar esta vista bajo una condición

Behavior / Event Actions: Aquí encontramos tres eventos para la vista que son:

Row Selected: Cuando hacemos un clic sobre la foto del alumno, nos abre la vista Detail, pudiendo observar los campos que se configuraron en esa vista.

Row Swiped Left (beta): Que cuando deslicemos a la izquierda de la pantalla del celular, le podamos dar una acción ejemplo que haga una llamada

Row Swiped Right (beta): Que cuando deslicemos a la derecha de la pantalla del celular, le podamos dar una acción ejemplo que se pueda editar los campos.

PESTAÑA BRAND

Aquí encontramos información relativa a la parametrización, del logo de la imagen de lanzamiento, como se debe ver nuestra barra inferior, superior.

Theme: cambia el fondo de la App fondo blanco o fondo negro

Primary Color: cambia el color del texto y de la imagen del menú de la App de la parte inferior.

App Logo : cambia el logo de la aplicación

Launch Image : permite poner una imagen de fondo que se muestra al inicio o memento de abrir o sincronizar la APP.

Header y Footer :

Show view name in header: _ (muestra el nombre de la tabla en la parte superior).

Show logo in header: _ (muestra el logo en el encabezado de la APP)

Hide menu and search buttons: _ (oculta el botón de búsqueda, que se muestra en la parte superior de la APP)

Style: permite colocarle un color de fondo al área del encabezado y al área donde se muestran los menús de la app en la parte inferior.

PESTAÑA FORMAT RULES

Regla de formato, por ejemplo, si queremos que la columna donde se muestran los alumnos, no la represente de un color especial.

PESTAÑA OPTIONS

Permite establecer una serie de opciones de visualización

PESTAÑA LOCALIZE

si tenemos nuestra aplicación para diferentes países en “**Share**” podemos definir que nombre adoptará la Aplicación en función del país en que esté

-

OPCIÓN BEHAVIOR

Automatización y comportamientos. Aquí definimos , cómo se comporta nuestra aplicación. Hay botones que se crean automáticamente en la aplicación, como el botón eliminar, editar, envío de mensajes y otros.

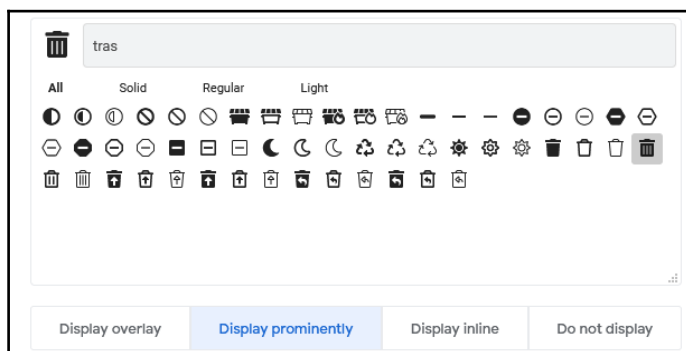
PESTAÑA ACTIONS

Show System Actions

si hacemos clic en **“Show System Actions”** nos muestra un listado de las acciones que se han creado automáticamente en la APP, con inteligencia virtual para cada una de las tablas.

CAMBIAR DE APARIENCIA ALGUNOS BOTONES DE ACCIÓN

- Seleccionamos el menú correspondiente a la tabla en la parte inferior de la APP.
- En el área central hacemos clic sobre la tabla que se le quiere modificar
- clic sobre el botón de acción (**Eliminar, Editar, mensajes**)
- haciendo clic en **Appearance**, se muestran las opciones a escoger.



Ocultar el Botón de acción

Hacemos un clic en **“Do not display”**

Mostrar los Botones de acción que se generan automáticamente

Hacemos un clic en **“Display prominently”**

Cambiar el Comportamiento del Botón de Acción con Fórmula

-Hacemos un clic en Behavior

-Hacemos clic en **“Only if this condition is true”** y después del signo igual (=) con un clic escribimos la fórmula.

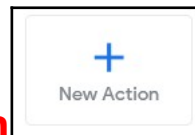
Que la APP Solicite confirmación, en un Botón de Acción

Activamos la opción **“Needs confirmation?”**

Cambiar el mensaje que sale al hacer un clic en el botón de acción

En Confirmation Message escribimos el texto que se quiere que aparezca, cuando se haga un clic en el botón de acción.

Generar o Crear una Nueva Acción



Ejemplo Botón de acción, que permite cambiar el valor del dato de una fila de determinada columna

Si de la tabla **Programas**, queremos dos botones de acción uno llamado **“Con descuento”** y otro llamado **“Descuento Negado”**

Cuando se haga un clic en el botón de acción **“Con descuento”** entonces una vez confirmado; la columna **Descuento** de la tabla **Programas**, cambiará al valor a **“Aprobado”**.

Pero si hacemos un clic en el botón de acción **“Descuento Negado”** entonces una vez confirmado; la columna **Descuento** de la tabla **Programas**, cambiará al valor a **“Negado”**

Pasos: para crear el botón de acción “Con descuento”

- 1. Hacemos un clic en el Botón “**New Action**”
- 2. Se muestra una nueva ventana,
- 3. En “**Action Name**”, escribimos el nombre para el botón de acción “
- 4. En la opción “**For a record of this table**” escogemos la tabla
- 5. En la opción “**Do this**” definimos qué acción se quiere que realice. Aquí se encuentran 3 grupos:

El grupo de las **APP**, permiten crear comportamientos dentro de las APP. que al tocar un botón nos lleve a otra parte de la APP.

El grupo de **DATA**, permite hacer modificaciones de los datos que tenemos actualmente.

El grupo de las **EXTERNAL**, nos lleva a partes que están fuera de la aplicación

GROUPED, en esta podemos encadenar varias acciones.

Un ejemplo, si de la tabla **Programas**, queremos dos botones de acción uno llamado “**Con descuento**” y otro llamado “**Descuento Negado**”

Cuando se haga un clic en el botón de acción “**Con descuento**” entonces una vez confirmado; la columna **Descuento** de la tabla **Programas**, cambiará al valor a “Aprobado”.

Pero si hacemos un clic en el botón de acción “**Descuento Negado**” entonces una vez confirmado; la columna **Descuento** de la tabla **Programas**, cambiará al valor a “**Negado**”

Descuentos
effect: Data: set the values of some columns in this row
Copy
Delete

Action name
A unique name for this action

Descuentos

For a record of this table
This action applies to rows of which table?

Programas

[View Definition](#)

Do this
The type of action to perform

Data: set the values of some columns in this row

Set these columns
To the constant or expression values defined

Descuento

= "Aprobado"

Add

Appearance

Display name
The name shown for this action in the app. Leave this empty to just use the action name. Or give it a text value (double quoted) or a formula.

= "Con Descuento"

T

- 6. En “ **Set these columns**” escogemos la fila a la cual se le quiere cambiar los valores y al frente escribimos = **”True”**
- 7. En **Appearance**, escribimos un nombre para el botón de acción
- 8. En **Action icon** seleccionamos un icono para el botón de acción
- 9. En **Behavior** en **Needs confirmation** escogemos si necesita confirmación, activamos y en **Confirmation Message**, escribimos un mensaje para la confirmación.

Behavior

Only if this condition is true
This action is available only for rows of the table that satisfy this condition

= true

Needs confirmation?
Does the user need to confirm the action before it is applied?

☒

Confirmation Message
What confirmation message should the user see?

= ¿Esta seguro de conceder el descuento?

T

Pasos: para crear el botón de acción “Descuento negado”

-1. Como ya se creo un botón de acción, similar de la misma tabla y de la misma columna, entonces hacemos una copia del botón de acción “con descuento” y realizamos algunos cambios mínimos así:

 Sin Descuentos

effect: Data: set the values of some columns in this row

CopyDelete↕

Action name

A unique name for this action

Sin Descuentos

For a record of this table

This action applies to rows of which table?

Programas ▾

[View Definition](#)

Do this

The type of action to perform

Data: set the values of some columns in this row ▾

Set these columns

To the constant or expression values defined

⋮

Descuento ▾

= "Negado"



Add

Appearance

⌵

Display name

The name shown for this action in the app. Leave this empty to just use the action name. Or give it a text value (double quoted) or a formula.

= "Descuento Negado"



Action icon

An icon for this action

 hand-point-down

Prominence Some actions are used often and should be prominently displayed.	Display overlay	Display prominently	Display inline	Do not display
---	-----------------	----------------------------	----------------	----------------

Behavior

Only if this condition is true
 This action is available only for rows of the table that satisfy this condition

= True 

Needs confirmation?
 Does the user need to confirm the action before it is applied?

☒

Confirmation Message
 What confirmation message should the user see?

= ¿Esta seguro de quitar el descuento?  T 

PESTAÑA WORK FLOW

Flujo de Trabajo, permite añadir distintas reglas que desencadenan tipos de acciones.

Ejemplo vamos a crear un **Work Flow**, que permite que cuando se agregue un nuevo registro en la tabla **Alumnos**, y se escoja el **programa** de Electricidad, nos envíe a un correo que escojamos, la notificación de este alumno del programa de electricidad.

PASOS PARA UNA NOTIFICACIÓN AL CORREO WORK FLOW

1. Al lado izquierdo de la pantalla, hacemos un clic en **Behavior** y clic en la columna **Work Flow**.
2. clic en **New Workflow Rule**
3. en **Rule name** ingresamos un nombre para la regla
4. hacemos un clic en **When this happens...** y en **Target Data** escogemos la tabla
5. En **Update event** escogemos Update event escogemos **ADDS-ONLY** (quiere decir sólo cuando se añaden nuevos registros)

Actions Workflow Reports Offline/Sync

+ New Workflow Rule

Add email automations for records in 'Alumnos' X

Add email automations for records in 'Programas' X

Add email automations for records in 'Pagos' X

Search workflow rules

Alumnos

Estudiantes de Electricidad

tasks: "Nueva notificación programa de electricidad":Send an email

Test Log Copy Delete Disable

Rule name

Unique name for this workflow rule.

Estudiantes de Electricidad

When this happens...

Target data

Changes to which table should trigger this rule?

Alumnos

[View Definition](#)

Update event

What sort of data changes should trigger this rule?

ADDS_ONLY

6. Hacemos un clic en **If this is true...** y en **Condition** ingresamos la fórmula (hacemos un clic, luego clic en **Columns**, buscamos la columna que en este caso es **programas** y al frente en **Insert** hacemos un clic y completamos la fórmula así: **[Programa]="Electricidad"**) y clic en **salvar**.
7. En **Run these tasks...** hacemos un clic en **"Nueva notificación"**
8. Observamos que este marcado el botón **Send an email** y en la opción **To** escribimos nuestro email

Run these tasks...

✉ 1: "Nueva notificación programa de electricidad"

Delete Task



Workflow Task Name

Name of the task to invoke

"Nueva notificación programa de electricidad"

Task category

Category of task



Send an
email



Send a
notification



Send an
SMS



Change
data

HTTP

Call a
webhook



Create a
new file

Via channel

The messaging channel to use.

System Default



To

Send email to these email addresses. (Expressions that yield email addresses are supported.)



= "henryoviedo29@gmail.com"



Add

Use default content?

If enabled, the task constructs a sensible default message. If disabled, you should define the content explicitly.



OPCIÓN USERS

Users Broadcast Notifications Links

Share app

Add entire domain? ☐

User emails
Example: "kim@mycompany.com"

Add

Users

User emails

Aquí podemos enviar una invitación para que un amigo o un usuario o un compañero de trabajo pueda instalar en su celular, o verlo desde el navegador la aplicación que hemos creado. Les va llegar un email, a través del cual van hacer un clic en un botón, que les llevará desde su celular a instalar APPSHEET, y posteriormente instalar la aplicación o si es desde el navegador entrar a la vista de la aplicación en el navegador.

LINK

Permite compartir un Link por email o whatsapp, nota si se envia el link, pero no hemos dada de alta al usuario no podremos entrar

Install Link

Permite a los usuarios de dispositivos móviles instalar la aplicación en su dispositivo.

Browser Link

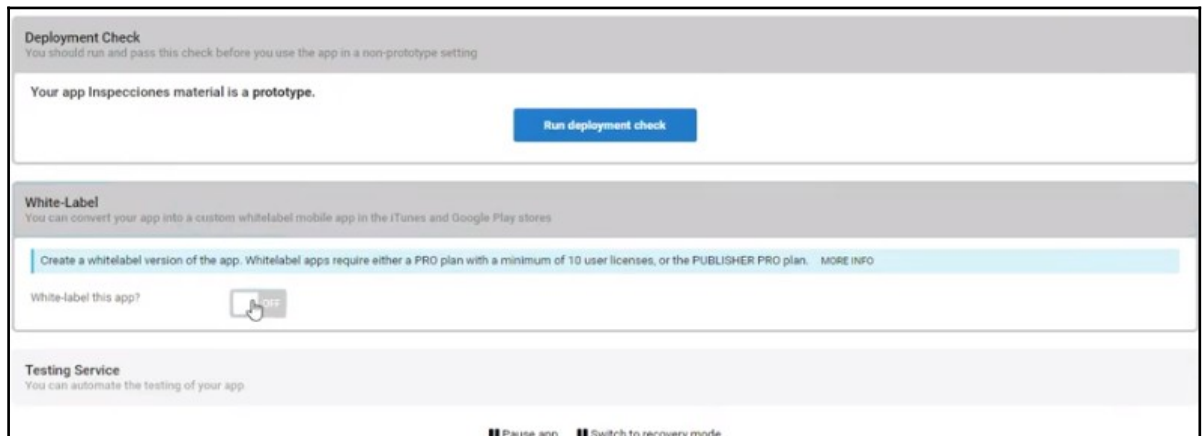
este link, permite ejecutar la aplicación directamente desde el navegador.

Not Deployed

Cuando ya está completamente terminada nuestra aplicación. entramos aquí, y y completamos la información que haga falta. aparecen unas advertencias, y el error que aparece es que no tenemos un plan de pago, pero aun así Apsheet permite. Para lanzar la aplicación hacemos un clic en el botón **“Move app to deployed state despite errors”**

PUBLICAR LA APP EN EL PLAYSTORE

Se debe crear una cuenta en el publisher de android o de IOS, y seguir el procedimiento normal que se hacen para publicar aplicaciones, en el caso de publicar en el Playstore habría que activar la opción **White label this app**



2. Ver vídeos canal YT de AppSheet

6. Estudiar las Sample Apps

curso de cero a experto

<https://appsheet.es>

https://www.youtube.com/watch?v=h9WL3y-loSo&list=PLoWb c9-O43FFfo-9IcAhauL_yWAbNINig&index=11

PASAR LOS TEXTOS QUE SE MUESTRAN EN INGLES A ESPAÑOL

1. Abierta la aplicación hacemos un clic en la interfaz del usuario **UX**, luego hacemos un clic en la última pestaña **Localize**.
2. Aquí podemos personalizar los textos de nuestra App. también podemos usar una fórmula, para cambiar de idioma, cuando el usuario lo requiera. Pero en este caso simplemente escribimos las palabras en español y listo. Nota. Se debe tener en cuenta, que hay algunas palabras reservadas para Appsheet. ejemplo "Ver".

<https://www.facebook.com/HLApps.info/>

Crear un Dashboard Velocímetro/Gauge

CREANDO UNA PORTADA O TABLA DE INICIO

https://www.youtube.com/watch?v=cG_K3-1CcKY

1. En la hoja de cálculo de Google, creamos una nueva hoja, en mi ejemplo llamaré **Inicio**. Le creamos dos columnas una **ID** y la otra **Nombre**. En la columna **ID** escribimos **1** y en la columna **Nombre** escribimos el texto de bienvenida que se mostrará en la portada. por ejemplo "Politécnico Tolimense"
2. en la App, agregamos la nueva tabla, haciendo clic en **Data**, pestaña **Table** y clic en **New Table**. la agregamos y la dejamos en modo de solo lectura **Read-Only**
3. luego en **Column** nos aseguramos que **Nombre** sea texto.
4. Creamos una nueva vista haciendo clic en **UX** y en la pestaña **Views** agregamos la vista para la tabla **Inicio**.
5. En **View Name**, escribimos un nombre para la vista, en **For This Data** seleccionamos la tabla inicio.
6. En **View Type**, seleccionamos **Detail** y en **Position** escogemos **Center**
7. en **Header Column** seleccionamos el campo **Nombre**, y en **Column Order** también escogemos el campo **Nombre**.
8. Estando en la interfaz del usuario **UX**, y en la pestaña **Options** y en **Starting View** seleccionamos la tabla Inicio.

USAR UNA COLUMNA VIRTUAL PARA CREAR UN FILTRO QUE PERMITA CONTAR REGISTROS

1. Clic en **Data** luego en **Column** y escogemos la tabla, en nuestro caso escogemos la tabla **Inicio**
2. Clic en **Add virtual Column**, en **Column Name** escribimos **Total Alumnos**, luego hacemos clic en **App Formula** e ingresamos la siguiente fórmula:

para contar todos los Registros: **COUNT(Alumnos[ID Alumno])**

Donde **Alumnos** es el nombre de la tabla donde están los alumnos

Donde **ID Alumno** es la clave o Id de la tabla Alumnos

Para contar únicamente los alumnos de sistemas: **COUNT(FILTER(Alumnos, [Programa]= "bbc4d89a"))**

Donde **Alumnos** es el nombre de la tabla donde están los alumnos

Donde **Programa** es la columna de la tabla alumnos que contiene los programas.

Donde **bbc4d89a** es la referencia de los programas de Ingles, porque en esta tabla se usó una referencia para llamar los programas de otra tabla.

Nota. si queremos mostrar las dos columnas virtuales que hemos creado hacemos lo siguiente: vamos a **UX** y en **View** escogemos la tabla **Inicio** y en la opción **Column order** agregamos las dos columnas virtuales, para que se visualicen en la app.

AÑADIR UN GRAFICO A LA PORTADA O TABLA INICIO

para esto usamos la url de una página como es <https://quickchart.io/> que dispone de una variedad de gráficos para usar.

El gráfico que tomaremos de Ejemplo. es **Radial Gauge / "Meter" Charts**

por lo tanto en el código que se muestra en la web correspondiente a este gráfico, copiamos el código donde inicia la url.

- creamos una nueva columna virtual en nuestra tabla Inicio, le damos el nombre de indicadores y en App Fórmula, pegamos la línea de código de la url que se había copiado.

"https://quickchart.io/chart?c={type:'radialGauge',data:{datasets:[{data:[70],backgroundColor:'green'}]}}"

Expression Assistant

App Formula for column Indicadores (Unknown)

"https://quickchart.io/chart?c={type:'radialGauge',data:{datasets:[{data:[70],backgroundColor:'green'}]}}"

✓

"https://quickchart.io/chart?c={type:'radialGauge',data:{datasets:[{data:[70],backgroundColor:'green'}]}}"

Yes/No

Math

Text

Time

Columns

Lists

Deep Links

Other

Modificamos la fórmula y reemplazamos **[70]**

Expression Assistant

App Formula for column Indicadores (Text)

"https://quickchart.io/chart?c={type:'radialGauge',data:{datasets:[{data:["& TEXT(100* [Total Alumnado] / [Total alumnos de Ingles]) &"] , backgroundColor:'green'}]}}"

The concatenation of all these text values (

...."https://quickchart.io/chart?c={type:'radialGauge',data:{datasets:[{data:["

✓TEXT(

.....((100 * The value of column 'Total Alumnado' / The value of column 'Total alumnos de Ingles')

...."] , backgroundColor:'green'}]}}"

en nuestra aplicación se verá de esta forma

☰

🗑️ ↺ ⋮

Mi Politecnico

Total Alumnado

6

Total alumnos de Ingles

3

Indicadores

https://quickchart.io/chart?c={type:'radialGauge',data:{datasets:[{data:[200] , backgroundColor:'green'}]}}

Ahora clic en Data, pestaña Column, escogemos la tabla Inicio

	NAME	TYPE	KEY?	LABEL?	FORMULA	SHOW?
1	_RowNumber	Number	☐	☐	=	☐
2	ID	Number	☒	☒	=	☒
3	Nombre	Text	☐	☐	=	☒
4	Total Alumnado	Number	☐	☐	= COUNT(Alumnos[ID Alum	☒
5	Total alumnos de Ingles	Number	☐	☐	= COUNT(FILTER(Alumnos,	☒
6	Indicadores	Text	☐	☐	= "https://quickchart.io/cha	☒

Observamos que la columna virtual indicadores esta **texto**, entonces cambiamos el tipo texto por **Image** y el resultado es que se mostrará el gráfico en nuestra App. así:



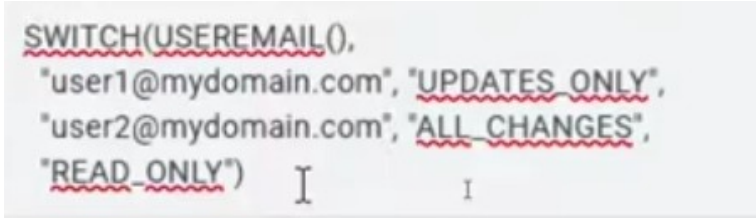
FORMULAS EN APPSHEET

Sumar los diferentes pagos, que realiza un alumno:

En la tabla pagos creamos una nueva columna virtual y en la casilla de fórmulas escribimos:

SUM(SELECT(Pagos[Valor],[Nombre]=[_THISROW].[Nombre]))

Traducción: Suma, siempre y cuando seleccione de la tabla Pagos, en la columna Valor, y en la columna Nombre, todos los pagos que sean iguales o corresponden a este mismo nombre.

	Pagos: Corresponde a la tabla Pagos Valor: corresponde a la columna o campo Valor de la tabla pagos (donde se encuentran los pagos realizados) Nombre: corresponde a la columna o campo Nombre de la tabla Pagos (aquí se encuentran los nombres de los alumnos que realizaron los pagos:
Buscar en cascada o aplicar varios filtros en una vista. https://www.youtube.com/watch?v=qjfojNfilWU Permite realizar varias búsquedas, de un determinado registro o alumno, pudiendo seleccionar los campos que queremos ver Dependent Dropdwn filters. Formula para Administración, de cuentas de usuarios. Permite colocar restricciones a usuarios que no sean administrador, para el manejo de la App. Debemos ir a Users y en User emails, agregamos los correos de los usuarios, que manejaran nuestra App. Les llegara a sus correos el Link de nuestra App. Pero antes debemos colocar las restricciones, para cada usuario. En la parte inferior de la ventana	<pre>SELECT(Pagos[Fecha], AND(OR(ISBLANK([_THISROW].[Nombre]),[Nombre]=[_THISROW].[Nombre]), OR(ISBLANK([_THISROW].[Recibo]),[Recibo]=[_THISROW].[Recibo]), OR(ISBLANK([_THISROW].[Valor]),[Valor]=[_THISROW].[Valor])))</pre> Previamente se debió crear una hoja llamada por ejemplo Filtro, en la hoja de Google. Y en la hoja filtro se crean las columnas que se quieren filtrar. Es necesario un campo que haga referencia a la tabla Pagos en nuestro ejemplo a ese campo lo llamaremos Llave(que es donde queremos filtrar) Estando en nuestra APP, en Tablas creamos una nueva Tabla para la búsqueda o filtro. En Source Id escogemos el nombre de la Hoja de Google, y en Worksheet Name/Qualifier escogemos la columna Filtro. Editamos la columna a filtrar y en la opción Data Validity y en Valid If validamos si se cumple con una condición y escribimos la formula. Nota: Esta misma formula la escribimos en cada una de las columnas a filtrar. Únicamente cambiamos, el nombre a mostrar de la columna filtrada  Pero si tenemos mucho usuarios, seria molesto hacerlo usuario por usuario mejor lo hacemos teniendo en cuenta su rol. Así:

usuario, se crean el rol del usuario.

Estos son los valores que puedes poner y sus efectos:

"READ_ONLY" Solo leer
"UPDATES_ONLY" Solo actualizar no crear
"ADDS_ONLY" Solo agregar, no modificar
"ADDS_AND_UPDATES" Agregar y actualizar, no borrar

"DELETES_ONLY" Solo borrar
"UPDATES_AND_DELETES" Actualizar y borrar, no agregar

"ADDS_AND_DELETES" Agregar y borrar, no actualizar

"ALL_CHANGES" Todos los cambios Como USERROLE() solo tiene dos posibilidades por default (Creo que se puede ampliar) normalmente al "Admin" le darías todos los cambios "ALL_CHANGES" y a "User" el que creas mas adecuado.

```
SWITCH(USERROLE(),  
"User", "UPDATES_ONLY",  
"Admin", "ALL_CHANGES",  
"READ_ONLY")|
```

Data/Tables escogemos y en **Are updates allowed?** Hacemos clic en el símbolo de formula y escribimos la formula

Para restringir entramos a datos y depende la tabla que escojamos podemos crear la restricción. Podemos restringir, lo que queramos de la APP. Para cada usuario de forma independiente

FORMULA QUE MULTIPLICA EL VALOR DE UN CAMPO POR EL VALOR DE OTRO CAMPO EN UNA MISMA TABLA. Y SE NUESTRA EL RESULTADO EN LA VISTA.

Imaginemos las siguientes tablas: **Productos** y **Ventas**

Productos						
7 columns: ID ID						
	NAME	TYPE	KEY?	LABEL?	FORMULA	SH
1	_RowNumber	Number	☐	☐	=	
2	ID	Text	☒	☒	=	
3	Nombre	Text	☐	☐	=	
4	Descripcion	Text	☐	☐	=	
5	Unidad	Enum	☐	☐	=	
6	Precio	Decimal	☐	☐	=	
7	Related Ventas	List	☐	☐	= REF_ROWS("Ventas", "Pro	

Ventas					
6 columns: ID ID					
NAME	TYPE	KEY?	LABEL?	FORMULA	
1 _RowNumber	Number	☐	☐	=	
2 ID	Text	☒	☒	=	
3 Fecha	Date	☐	☐	=	
4 Producto	Ref	☐	☐	=	
5 Cantidad	Number	☐	☐	=	
6 Precio Total	Decimal	☐	☐	=	

Observemos que la columna Producto de la tabla Ventas es una referencia de la tabla Productos. Es decir que una vez ingresado los datos en la tabla Productos, podemos hacer que desde la tabla ventas, en el campo Producto de esta tabla se llamen los datos de la tabla Producto.

Se requiere que la columna Precio Total, se muestre calculado automaticamente.

Entonces en **Data/Columns** de la tabla ventas, editamos el Precio Total y en la Opción **Auto Compute / App formula**, escribimos la siguiente formula:

[Cantidad]*[Producto].[Precio]

Cuando registramos una venta y ponemos la cantidad y el precio automaticamente nos muestra el valor total, el cual no se puede editar.

ID*

11d1f511

Fecha

07/11/2020

☐

Producto

Cantidad

0

Precio Total

0.00

ID*

11d1f511

Fecha

07/11/2020

Producto

Azucar

Cantidad

1

Precio Total

10.00

Aquí iríamos agregar una nueva venta y observe que el precio total esta en cero

Cuando agregamos la cantidad, que en este caso es 1, automáticamente se muestra el valor total que es 10

LEER LA INFORMACIÓN DE OTRA TABLA A TRAVES DE UN CAMPO REF

MOSTRAR EL DETALLE DE UNA COLUMNA EN LA APP

Si, queremos mostrar el detalle de un producto (su precio y su unidad), al momento de hacer la venta. Hacemos.

1. En **Data/Column** escogemos la tabla **Ventas**, y agregamos una columna virtual
2. En **Column Name** escribimos, Detalle del Producto y en **App Formula** escribimos la siguiente formula:

Nota:

“precio: \$“ & [Producto].[Precio] & “por “ & [producto].[Unidad]

Al agregar una venta, observemos que al final se muestra la información de la columna virtual

Detalle del Producto
Precio: 10 por Kilos

ID *	52fc5783
Fecha	07/11/2020
Producto	Azucar
Cantidad	0
Precio Total	0.00
Detalle del Producto	Precio: 10 por Kilos

Para mover la información del detalle del producto, justo despues del producto. Hacemos:

Acudimos a las vistas, en la parte inferior de la App, se muestra una vista auto generada, “ventas_Form”, hacemos clic en ella estas son vistas de sistemas que estan ocultas y para verlas, en **UX/Views** en la p´rte inferior hacemos un clic en **Hide system Views**, haciendo clic en la tabla **Ventas_Form** procedemos hacer una copia; le cambiamos el nombre a la vista y en la opción **Column Order** cambiamos el orden de las columnas.

Base de datos AppSheet Lección 6: Papelería

Micra parte 4 Mensaje de WhatsApp

<https://www.youtube.com/watch?v=ihWEdu8Cd28>

video youtube

AppSheet Tutorial 2021 - 05. Relaciones entre tablas

<https://translate.google.es/translate?hl=es&sl=en&tl=es&u=https%3A%2F%2Fhelp.appsheet.com%2Fen%2Farticles%2F1013271-column-types-diving-deeper&prev=search>