

Plan Maestro Arquitectónico para el Sistema de Gestión de Pedidos D3LTA en Google Sheets

Introducción: Del Formulario al Sistema Funcional

El objetivo principal de este informe es diseñar una arquitectura de gestión de datos integral y escalable dentro de un único libro de cálculo de Google Sheets. Este sistema está diseñado para reflejar y capturar a la perfección toda la gama de información recopilada por el formulario "Página de contratación".¹

La filosofía central de este diseño se aleja estratégicamente de una simple estructura de archivo plano. La complejidad inherente del formulario, en particular sus relaciones de uno a varios (por ejemplo, un único pedido que puede tener múltiples versiones de destino), exige la adopción de un modelo relacional. Este enfoque se implementa utilizando hojas de cálculo separadas que funcionan como tablas interconectadas, vinculadas mediante identificadores únicos.

La arquitectura propuesta ofrece beneficios clave que son fundamentales para la operatividad y el crecimiento del negocio. Primero, garantiza la **integridad de los datos** al minimizar la redundancia y prevenir la entrada de información inconsistente. Segundo, proporciona **escalabilidad**, permitiendo que el sistema se adapte sin esfuerzo a futuros cambios, como la adición de nuevos planes de servicio o funcionalidades. Finalmente, la estructura está concebida para la **analítica**, organizando los datos de una manera que facilita análisis potentes, la generación de informes y la aplicación de inteligencia de negocios directamente dentro del entorno de Google Sheets.

Este documento está dirigido al gerente de negocio o al desarrollador principal responsable de implementar y mantener el sistema de gestión de pedidos. Proporciona un plan maestro completo, desde la estructura de alto nivel hasta las definiciones a nivel de campo individual.

Sección 1: El Modelo Relacional Central - Una Visión General del Sistema

El fundamento de esta arquitectura reside en el concepto de entidad-relación, adaptado al entorno de una hoja de cálculo. Se utilizan **Claves Primarias** (identificadores únicos para cada fila en una hoja) y **Claves Foráneas** (referencias a Claves Primarias en otras hojas) para construir relaciones lógicas entre los datos. Este método transforma un conjunto de hojas aisladas en un sistema de información cohesivo.

La arquitectura del libro de cálculo se puede visualizar a través del siguiente diagrama, que sirve como un mapa de las interconexiones entre las hojas propuestas:

```
[Pedidos] Hoja (El Registro Maestro)
|
| +---< (Relación 1-a-1, vinculada por PedidoID)
|
| +---< (Relación 1-a-muchos, vinculada por PedidoID)
|
| +---< [Activos_Creativos] (Relación 1-a-1, vinculada por PedidoID)
||
| +---< [Pedido_Fuentes] (Relación 1-a-muchos, vinculada por PedidoID)
||
| +---< [Pedido_Colores] (Relación 1-a-muchos, vinculada por PedidoID)
|
| +---< [Adicionales_Impresion] (Relación 1-a-1, vinculada por PedidoID)
```

Un análisis del código del formulario revela una lógica de negocio fundamental que debe ser reflejada en la estructura de datos. El código JavaScript del formulario incluye una función, `validateMainFormButton()`, que mantiene deshabilitado el botón de envío (`open_modal_button`) hasta que se completen ciertos campos clave.¹

Específicamente, el usuario debe rellenar la descripción de la marca (

brand_info), completar al menos una versión de destino (link-entry), y proporcionar instrucciones de diseño si se selecciona el servicio complejo. Esta no es una simple validación para mejorar la experiencia del usuario; es una regla de negocio que define un "Pedido Mínimo Viable". Un simple registro de datos que capture todo indiscriminadamente no reconocería esta regla. El sistema debe, por lo tanto, gestionar el ciclo de vida completo del pedido. Esto conduce a la inclusión de un campo EstadoPedido en la hoja Pedidos, lo que permite una gestión activa del flujo de trabajo (por ejemplo, 'Enviado', 'Diseño en Progreso', 'Esperando Pago', 'Completado'). Esta decisión transforma la hoja de cálculo de un registro pasivo a una herramienta de gestión activa.

El libro de cálculo se organizará en hojas de datos principales (Pedidos, Detalles_Servicio, Destinos, Activos_Creativos, Pedido_Fuentes, Pedido_Colores, Adicionales_Impresion) y hojas de utilidad (Dashboard, Validacion_Datos, LEEME), cada una con un propósito definido que se detallará en las siguientes secciones.

Sección 2: La Hoja Pedidos - El Registro Transaccional Central

Esta hoja es la tabla maestra del sistema. Cada fila representa un único pedido de cliente que ha sido confirmado y enviado a través del formulario modal. Contiene los datos de contacto finales, metadatos de envío y, de manera crucial, los componentes de precios calculados que forman el total.

El motor de precios del formulario, impulsado por la función JavaScript calculateCosts(), desglosa el costo final en múltiples componentes: diseño, funcionalidad, adicionales e impresión.¹ Almacenar únicamente el precio total representaría una pérdida significativa de datos valiosos para la inteligencia de negocios. Una estructura de datos óptima debe capturar esta granularidad. Por ejemplo, un gerente de negocio podría necesitar responder preguntas como "¿Cuántos ingresos generamos con el 'Plan Business' el último trimestre?" o "¿Es rentable el adicional de 'Entrega Express'?". Si solo se almacena el total, responder a estas preguntas sería imposible sin volver a ejecutar cálculos complejos sobre los datos brutos, un proceso ineficiente y propenso a errores. Por lo tanto, esta arquitectura dedica columnas específicas en la hoja

Pedidos para almacenar estos componentes de precio individuales, así como el monto del descuento aplicado, para rastrear el impacto de las palabras clave de los vendedores.

Definición de la Tabla: Pedidos

Esta tabla define cada columna, su tipo de dato, una descripción y datos de ejemplo. Los campos clave como la Clave Primaria (PK) se indican para mayor claridad.

Nombre de Columna	Tipo de Dato	Descripción	Ejemplo
PedidoID (PK)	Texto	Identificador único para el pedido, generado secuencialmente.	D3LTA-0001
FechaEnvio	Fecha/Hora	Marca de tiempo automática de cuándo se envió el formulario modal.	2023-10-27 14:30:15
EstadoPedido	Lista Desplegable	El estado actual del pedido en el flujo de trabajo.	Enviado
NombreContacto	Texto	Nombre y apellido del cliente del modal. ¹	Juan Pérez
EmailContacto	Texto (Email)	Correo electrónico del cliente del modal. ¹	juan.perez@email.com
TelefonoContacto	Texto	Número de teléfono del cliente del modal. ¹	11 2345 6789
MetodoContactoPref	Lista Desplegable	Vía de contacto preferida seleccionada. ¹	whatsapp
Moneda	Texto (ARS/USD)	La moneda seleccionada por el usuario para la	ARS

		cotización. ¹	
PalabraClaveVendedor	Texto	La palabra clave ingresada por el usuario. ¹	VENDEDOR01
DescuentoAplicado	Booleano	VERDADERO si la palabra clave fue válida y se aplicó un descuento.	VERDADERO
CostoBaseDiseño_CHF	Número	Costo del nivel de diseño en la moneda base (CHF). ¹	62
CostoVersionesAdic_CHF	Número	Costo de las versiones de destino adicionales en CHF. ¹	9
CostoFuncionalidad_CHF	Número	Costo del plan de URL dinámica en CHF. ¹	95
CostoAdicionales_CHF	Número	Suma de todos los costos de adicionales (express, etc.) en CHF. ¹	13
CostoImpresion_CHF	Número	Costo total del servicio de impresión en CHF. ¹	150.5
MontoDescuento_CHF	Número	Monto del descuento calculado en CHF.	28.95
CostoTotalFinal_CHF	Número	Costo total final en la moneda base (CHF).	260.55
CostoRenovacionAnual_CHF	Número	Costo de renovación anual en CHF (solo para planes dinámicos). ¹	45
CostoTotalFinal_Local	Número (Fórmula)	Costo total final en la moneda	325688

		seleccionada por el usuario.	
CostoRenovacion_Local	Número (Fórmula)	Costo de renovación en la moneda seleccionada por el usuario.	56250
PolíticasAceptadas	Booleano	VERDADERO si el usuario aceptó los términos y condiciones. ¹	VERDADERO

Sección 3: La Hoja Detalles_Servicio - Deconstruyendo la Solicitud Central

El propósito de esta hoja es aislar limpiamente las elecciones fundamentales del servicio de los datos transaccionales de la hoja Pedidos. Esta separación mejora la claridad y la organización, siguiendo el principio de "separación de intereses". Esta hoja mantiene una relación uno a uno con la hoja Pedidos, donde cada fila aquí corresponde a una única fila en la tabla maestra.

Definición de la Tabla: Detalles_Servicio

Nombre de Columna	Tipo de Dato	Descripción	Ejemplo
DetalleID (PK)	Texto	Identificador único para esta entrada de detalle.	DET-0001
PedidoID (FK)	Texto	Vincula esta entrada al PedidoID en la hoja Pedidos.	D3LTA-0001
ComplejidadDiseño	Lista Desplegable	Nivel de diseño elegido (basic,	standard

		standard, premium). ¹	
PlanURLDinamica	Lista Desplegable	Plan de funcionalidad elegido (none, starter, business, scale). ¹	business
ServicioDiseñoExt	Lista Desplegable	Servicio de diseño gráfico adicional (none, integration, complex). ¹	none
InstruccionesDiseñoExt	Texto Largo	Instrucciones del campo design_complex_instructions. ¹	Queremos un afiche A4 con temática vintage...
DescripcionEstiloMarca	Texto Largo	Descripción del campo brand_info. ¹	Somos una empresa de café de especialidad...

Sección 4: La Hoja Destinos - Gestionando Múltiples Versiones con Precisión

Esta es la hoja más crítica para manejar la complejidad del formulario, ya que resuelve la relación de uno a varios donde un solo pedido puede tener múltiples versiones de destino. Cada fila en esta hoja representa una "Versión de Destino" (link-entry) del formulario.¹

El diseño de esta hoja aborda un desafío común en la estructuración de datos conocido como "asociación polimórfica". El selector "Tipo de Destino" (link-type-selector) en el formulario altera dinámicamente los campos de entrada requeridos.¹ Por ejemplo, seleccionar

vcard muestra campos para nombre, puesto y empresa, mientras que seleccionar wifi muestra campos para SSID y contraseña. Un enfoque de base de datos estrictamente normalizado requeriría múltiples tablas separadas (Datos_vCard, Datos_WiFi, etc.), lo cual es impráctico y excesivamente complejo de gestionar y consultar en Google

Sheets. La solución pragmática es emplear una única tabla "ancha" con un conjunto unificado de columnas para albergar todos los tipos de datos posibles. Para una fila dada, solo se rellenarán las columnas relevantes para su TipoDestino; el resto permanecerá nulo. Esta es una decisión consciente de desnormalización que prioriza la facilidad de uso, el filtrado y la comprensión dentro de las limitaciones de la plataforma.

Definición de la Tabla: Destinos

Nombre de Columna	Tipo de Dato	Descripción	Ejemplo
DestinoID (PK)	Texto	Identificador único para esta versión de destino específica.	DST-0001-1
PedidoID (FK)	Texto	Vincula al pedido principal en la hoja Pedidos.	D3LTA-0001
NumVersion	Número	Número secuencial de la versión dentro del pedido.	1
TituloDestino	Texto	Título proporcionado en link-title-input. ¹	Sitio Web Principal
TipoDestino	Lista Desplegable	Tipo seleccionado (url_static, vcard, wifi, etc.). ¹	url_static
EsURLDinamica	Booleano	VERDADERO si se marcó link-dynamic-url-toggle. ¹	FALSO
Valor_URL	Texto (URL)	Para el tipo url_static.	https://miempresa.com
Valor_Texto	Texto Largo	Para el tipo text.	Promoción especial de otoño.

Valor_LinkPago	Texto	Para el tipo payment.	miempresa.mp
vCard_Nombre	Texto	Para el tipo vcard.	Ana García
vCard_Puesto	Texto	Para el tipo vcard.	Gerente de Ventas
vCard_Empresa	Texto	Para el tipo vcard.	Mi Empresa S.A.
vCard_Telefono	Texto	Para el tipo vcard.	+54911...
vCard_Email	Texto (Email)	Para el tipo vcard.	ana.garcia@miempres a.com
vCard_SitioWeb	Texto (URL)	Para el tipo vcard.	https://miempresa.co m
WiFi_SSID	Texto	Para el tipo wifi.	RedDeInvitados
WiFi_Contraseña	Texto	Para el tipo wifi.	claveSegura123
Geo_Latitud	Texto	Para el tipo geo.	-34.6037
Geo_Longitud	Texto	Para el tipo geo.	-58.3816
NotasEspecificas	Texto Largo	Del campo link-notes-input. ¹	Usar para la campaña de marketing de octubre.
TieneTextoExtra	Booleano	Del checkbox link-extra-text-toggle . ¹	VERDADERO
ContenidoTextoExtra	Texto	Del campo link-extra-text-input. ¹	¡Escaneame!

Sección 5: Activos_Creativos y Hojas Relacionadas - Catalogando Materiales del Cliente

El propósito de esta sección es rastrear todos los archivos y elecciones creativas proporcionadas por el usuario. Dado que el número de fuentes y colores es variable según el plan de servicio seleccionado, este aspecto se gestionará a través de tres

hojas relacionadas para mantener una estructura de datos limpia y escalable.

El formulario permite una cantidad variable de selecciones de fuentes y colores, controlada por las funciones `updateFontSelectorUI` y `updateColorPickerUI`.¹ Un error de diseño común sería crear columnas como

`Color1`, `Color2`, `Color3`. Esta aproximación es rígida y difícil de analizar. Si un nuevo plan de servicio permitiera más colores, se requeriría una modificación de la estructura de la tabla. Además, responder a una pregunta como "¿Cuál es nuestro color de marca más solicitado?" implicaría consultar y agregar datos de múltiples columnas. La solución relacional correcta es normalizar estos datos, otorgando a cada selección de fuente y color su propia fila en una tabla dedicada. Una orden premium con siete colores generaría siete filas en la tabla `Pedido_Colores`, todas vinculadas al mismo `PedidoID`. Esta estructura es infinitamente escalable y simplifica enormemente el análisis.

5.1. Definición de la Tabla: `Activos_Creativos`

Esta hoja rastrea los activos basados en archivos.

Nombre de Columna	Tipo de Dato	Descripción	Ejemplo
<code>ActivoID (PK)</code>	Texto	ID único para el registro del activo.	ACT-0001
<code>PedidoID (FK)</code>	Texto	Vincula al pedido principal.	D3LTA-0001
<code>TipoActivo</code>	Lista Desplegable	Tipo de archivo (Logo, <code>ImagenReferenciaColor</code> , <code>ArchivoIntegracionDiseño</code>).	Logo
<code>NombreArchivoOriginal</code>	Texto	Nombre del archivo subido (<code>logo_upload</code> , <code>color_ref_image</code> , etc.). ¹	logo_final_v2.svg

5.2. Definición de la Tabla: Pedido_Fuentes

Esta hoja normaliza las selecciones de tipografías.

Nombre de Columna	Tipo de Dato	Descripción	Ejemplo
FuenteEntradaID (PK)	Texto	ID único para la entrada de la fuente.	FNT-0001-1
PedidoID (FK)	Texto	Vincula al pedido principal.	D3LTA-0001
NombreFuente	Texto	Nombre de la fuente seleccionada de la lista. ¹	Montserrat

5.3. Definición de la Tabla: Pedido_Colores

Esta hoja normaliza las selecciones de colores manuales.

Nombre de Columna	Tipo de Dato	Descripción	Ejemplo
ColorEntradaID (PK)	Texto	ID único para la entrada del color.	CLR-0001-1
PedidoID (FK)	Texto	Vincula al pedido principal.	D3LTA-0001
ValorHex	Texto	Código hexadecimal del color seleccionado. ¹	#F7DF4E

Sección 6: La Hoja Adicionales_Impresion - Consolidando Servicios Opcionales

El objetivo de esta hoja es capturar de forma limpia todos los servicios adicionales opcionales (que funcionan como booleanos) y los parámetros específicos para el servicio de impresión. Esta separación simplifica las hojas Pedidos y Detalles_Servicio y hace que el análisis relacionado con los adicionales sea directo y sencillo. Mantiene una relación uno a uno con la hoja Pedidos.

Definición de la Tabla: Adicionales_Impresion

Nombre de Columna	Tipo de Dato	Descripción	Ejemplo
AdicionalID (PK)	Texto	Identificador único para este registro.	ADD-0001
PedidoID (FK)	Texto	Vincula al pedido principal.	D3LTA-0001
TieneEntregaExpress	Booleano	VERDADERO si se marcó express_delivery. ¹	FALSO
TieneCertGigantografia	Booleano	VERDADERO si se marcó large_format. ¹	FALSO
TieneServicioImpresion	Booleano	VERDADERO si se marcó print_stickers. ¹	VERDADERO
CantStickersPorVersion	Número	Cantidad del campo sticker_quantity. ¹	100
TamañoSticker	Lista Desplegable	Tamaño seleccionado (5x5, 8x8, 10x10). ¹	8x8
TieneCorteIndividual	Booleano	VERDADERO si se marcó cut_stickers. ¹	VERDADERO
FormaCorteSticker	Lista Desplegable	Forma de corte seleccionada (circular, cuadrado,	circular

		etc.). ¹	
--	--	---------------------	--

Sección 7: Hojas de Utilidad y Configuración - Convirtiendo el Libro en una Aplicación Útil

Estas hojas proporcionan la infraestructura y las herramientas orientadas al usuario que transforman el conjunto de hojas de datos en una aplicación de negocio funcional y robusta.

Hoja Dashboard

Se propone una hoja de resumen de solo lectura para visualizar el estado del negocio de un vistazo. Incluiría indicadores clave de rendimiento (KPIs) y gráficos dinámicos.

- **KPIs:** Total de Pedidos, Ingresos Totales (por moneda), Valor Promedio por Pedido.
- **Gráficos:** Gráfico circular de selecciones de ComplejidadDiseño, gráfico de barras de ingresos por PlanURLDinamica, gráfico circular de popularidad de Adicionales.
- Ejemplo de Consulta para un KPI: Para mostrar un resumen de pedidos por estado, se puede usar la siguiente fórmula en una celda:
=QUERY(Pedidos!A:Z; "SELECT C, COUNT(C) WHERE C IS NOT NULL GROUP BY C LABEL COUNT(C) 'Total de Pedidos'")

Hoja Validacion_Datos

Una hoja oculta que actúa como la única fuente de verdad para todos los menús desplegables utilizados en las hojas de entrada de datos.

- **Listas a Incluir:** EstadoPedido, ComplejidadDiseño, PlanURLDinamica, ServicioDiseñoExt, TipoDestino, TipoActivo, TamañoSticker, FormaCorteSticker.

- **Beneficio:** Este enfoque centralizado garantiza la consistencia de los datos y facilita la actualización de las opciones en un único lugar, propagando los cambios a todo el sistema.

Hoja LEEME

Un manual de instrucciones crucial y orientado al usuario, que será la primera hoja visible del libro.

- **Contenido:** Una explicación del propósito de cada hoja, una descripción de la convención de codificación por colores (por ejemplo, PK en amarillo, FK en azul, entrada de usuario en blanco) e instrucciones claras sobre cómo agregar un nuevo pedido manualmente. El proceso manual implicaría crear una nueva fila en Pedidos y luego agregar las filas correspondientes en las hojas relacionadas utilizando el mismo PedidoID.

Conclusión y Recomendaciones Estratégicas

Este informe ha detallado una arquitectura de datos que adopta un modelo relacional, maneja pragmáticamente datos polimórficos, normaliza datos de longitud variable y captura componentes de precios granulares para análisis futuros. El resultado es un sistema robusto, escalable y analíticamente potente dentro de Google Sheets.

Para la implementación, se formulan las siguientes recomendaciones estratégicas:

- **Integridad de los Datos:** Se recomienda encarecidamente utilizar la función "Proteger Rangos" de Google Sheets en todas las columnas de Clave Primaria, Clave Foránea y aquellas que contienen fórmulas. Esto evitará la modificación accidental por parte de los usuarios y protegerá la integridad estructural del sistema.
- **Codificación por Colores:** Implementar un esquema de color visual, como se describe en la hoja LEEME, para distinguir entre tipos de claves, celdas con fórmulas y campos de entrada manual. Esto mejora drásticamente la usabilidad y reduce los errores de entrada.
- **Preparación para el Futuro y Automatización:** El siguiente paso lógico para

mejorar la eficiencia es el uso de **Google Apps Script**. Se podría desarrollar un script que, al agregar una nueva fila en la hoja Pedidos, genere automáticamente un PedidoID único y cree las filas vinculadas correspondientes en todas las hojas relacionadas. Esto reduciría drásticamente el esfuerzo manual y el riesgo de error humano. Además, esta arquitectura normalizada está diseñada para ser fácilmente exportable a un sistema de base de datos más robusto (como PostgreSQL o MySQL) en el futuro, si el negocio escala más allá de lo que Google Sheets puede manejar cómodamente. La estructura propuesta hará que este proceso de migración sea significativamente más fluido.

Fuentes citadas

1. Página de contratación.txt