

**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**

**FACULTAD DE INGENIERÍA**

**Sistema de reporte y planeación de ventas y operaciones**

**INFORME DE ACTIVIDADES PROFESIONALES**

Que para obtener el título de

**Ingeniero en computación**

**P R E S E N T A**

Luis Ricardo Roldán Rivera

**ASESOR(A) DE INFORME**

Ing. Orlando Zaldívar Zamorategui

****

**Ciudad Universitaria, Cd. Mx., 2022**

# Introducción

Es muy común que a lo largo de la vida uno haga planes hacia el futuro, actividades simples que van desde la ropa que me pondré para un evento la siguiente semana, hasta grandes proyectos como construir una casa, el ámbito empresarial no es la excepción, la planeación es un factor muy importante para las organizaciones que persiguen un objetivo, deben considerar una ruta para alcanzar sus metas y estar midiendo a través del tiempo que tanto se desvían de sus objetivos, esto les permite reaccionar de mejor manera y más tempranamente ante eventos que puedan representar un problema.

Uno de los propósitos de una empresa es el de vender la mayor cantidad de productos posible mientras que se mantienen lo más bajo posible los costos de producción, estos 2 aspectos tienen que estar en sincronía, acuerdo y comunicación, si existe compromiso de vender algo que no se puede producir, la empresa puede terminar comprando a alguien más y revender, perdiendo así ganancias o peor aún, no lograr la venta y perder un cliente, por el otro lado, si hay una sobre producción y la demanda no es la suficiente, se tendrá que guardar el producto o incluso, tirar o rematar, este y muchos otros escenarios en los que hay diferencia entre producción y ventas ocasionan que la empresa pierda dinero u oportunidades, tiene que haber un balance entre estas 2 partes que permita en la medida de lo posible, reducir riesgos y mantener un flujo continuo desde las operaciones hasta la venta del producto o servicio.

Existen ciertos sectores en el que este balance es vital, pues los artículos o servicios que venden tienen una fecha de expiración, tal es el caso de las empresas que se dedican a vender artículos perecederos, en estas se debe tener mucho cuidado de que se venda producto un producto fresco y de buena calidad, si después de un tiempo este producto no se vende terminará siendo desechado y junto a él, todo el dinero invertido para producirlo.

Para evitar este y otros problemas se requiere una buena comunicación, la tecnología puede llegar a ser un buen aliado en este sentido, pues nos permite obtener, procesar y mostrar información con una alta eficiencia, además, pueden ahorrarnos tiempo al automatizar la extracción de datos y dándole formato para que su consulta sea clara, rápida y concisa

A continuación, les presento el análisis, diseño e implementación una herramienta de software de la cual fui participe, que captura, procesa y pone a disposición información para la toma de decisiones, abordando lo que actualmente se conoce como Sales & Operations Planning (S&OP).

# Objetivo

## Objetivo general:

Desarrollar una aplicación que permita concentrar la información del departamento de ventas y departamento de operaciones con el propósito de llegar a un acuerdo para balancear la demanda y el suministro de productos para las próximas semanas. Dicha aplicación deberá guardar información sobre las juntas para el consenso de ventas, además de extraer y agrupar información semana a semana de ventas anteriores y llevar un registro de las diferencias entre lo planeado y lo real.

## Objetivos particulares

* Automatizar la extracción diaria de la información de ventas y operaciones desde el ERP de la empresa hacia una nueva base de datos que permita su almacenamiento y procesamiento.
* Agrupar la información por semanas, grupos de productos y bajo ciertos criterios establecidos por las necesidades de la empresa, posteriormente, mostrarla bajo un formato previamente diseñado que sea claro, automatizado y legible reemplazando así la una hoja de calculo que servía como consulta.
* Mostrar dentro de la aplicación una serie de indicadores derivados de su contenido, que permitan saber las desviaciones de la información consensada respecto a la información real.

# Marco teórico

## Sales & Operations planning

Lo primero que hay que definir es el concepto de Sales and Operations Planning (S&OP por sus siglas en inglés), La APICS (American Production and Inventory Control Society) actualmente conocida como ASCM Association for Supply Chain Management, es una organización internacional sin fines de lucro líder en la fomentación de avances en todo lo que se refiere a cadenas de suministro, la siguiente definición es la traducción personal dada por esta institución

“Un proceso para desarrollar planes tácticos que brinden a la gerencia la capacidad de dirigir estratégicamente sus negocios para lograr ventaja competitiva de manera continua, integrando planes orientados al consumidor para nuevos y existentes productos con la administración de la cadena de suministros. Este proceso reúne todos los planes del negocio (ventas, marketing, desarrollo, manufactura, abastecimiento y finanzas) en un solo e integrado conjunto de planes. Es ejecutado por lo menos una vez al mes y es revisado por administración en su conjunto (familia de productos). El proceso debe conciliar toda la oferta, la demanda, y nuevos planes de productos tanto a nivel de detalle como agregado y vincularse al plan de negocios.”[[1]](#footnote-1)

La anterior definición se refiere a S&OP como un proceso, es decir, una serie de pasos a seguir para alcanzar un objetivo, también habla de puntos clave como plan, conciliación, revisión y ventas que son de interés para este proyecto. Otra definición para S&OP, traducida personalmente, es la siguiente: Una herramienta que unifica diferentes planes de negocio en un solo conjunto de planes, su propósito principal es doble, equilibrar la oferta y la demanda y construir puentes entre el plan de negocio o estratégico y los planes operativos de la firma.[[2]](#footnote-2)

Esta definición describe a S&OP como una herramienta con dos principales propósitos, equilibrar y unificar, personalmente creo que esta definición es más concisa y simple, establece el propósito básico de S&OP y contiene las palabras clave para este proyecto “equilibrar oferta y demanda”. Al final ambas definiciones coinciden en una salida o beneficio de este proceso, el cual es un plan o conjunto de planes unificado.

Otro punto importante a considerar es también el tiempo, S&OP como vimos en su definición, es un plan, tiene un periodo de tiempo establecido y un seguimiento, la misma definición de la ASCM establece un periodo mínimo de revisión mínimo de por lo menos una vez al mes, sin embargo, se refiere a la revisión del plan que tiene como salida el S&OP, como entrada tiene que considerar planes de diferentes sectores de la empresa y cada uno puede tener diferentes horizontes de tiempo dependiendo del rubro y sector de la empresa, estos planes también marcan el paso y forman parte de los diferentes horizontes de tiempo que comprende S&OP, tomando como base las propuestas de Dreyer y Heidi Carin en su artículo Proposals for enhancing tactical planning in grocery retailing with S&OP[[3]](#footnote-3), el plan para S&OP debe contemplar el plan de ventas en un periodo de 6 a 12 meses pero también considerar la demanda variante propia de la venta de comestibles, coincidiendo con la ASCM en una revisión a corto plazo con frecuencia mensual pero ajustable a un plazo menor si las oportunidades o riesgos surgen del lado del área de suministros.

Detallar cómo funciona una cadena de suministros o como se crea un plan de ventas va más allá de mis conocimientos actuales y fuera del alcance de este informe, no obstante, se definen algunos conceptos que fueron comprendidos en el desarrollo y usados frecuentemente en las reuniones de análisis y diseño de la solución, considero que dar una breve descripción de cada concepto es importante para comprenderla mejor.

* Sales Forecast: Se define como una proyección hacia el futuro de la demanda esperada, dadas un conjunto de condiciones ambientales establecidas.[[4]](#footnote-4)
* Stock: El monto total de bienes o de un tipo particular de bienes disponibles en una tienda[[5]](#footnote-5)
* Supply: El reabastecimiento real o planificado de un producto o componente. Las cantidades de reaprovisionamiento se crean en respuesta a una demanda del producto o componente o en previsión de tal demanda.[[6]](#footnote-6)
* Actual Sales o Sales: Numero de artículos vendidos[[7]](#footnote-7)

## Power Platform y servicios

El siguiente contenido tiene fines única y exclusivamente informativos, no representa ningún tipo de comercial o promoción de servicios, a continuación, menciono información básica de la plataforma que se usó para desarrollo del proyecto, considero importante entender cómo esta conformada y los principios básicos de dicha herramienta.

### Power Platform

Power Platform es un conjunto de aplicaciones creadas por Microsoft orientada a crear soluciones empresariales bajo un esquema No Code (Sin código) o Low Code (Poco código), sus aplicaciones son ejecutadas en la nube y se permiten conectar a otros servicios de Microsoft y de terceros. Power Platform está compuesta por los siguientes cinco productos.

#### Power Apps

Power Apps proporciona un entorno de desarrollo de código bajo rápido para crear aplicaciones personalizadas para las necesidades comerciales. Tiene servicios, conectores y una plataforma de aplicaciones y servicios de datos escalables (Microsoft Dataverse) para permitir una integración e interacción sencillas con los datos existentes. Power Apps permite la creación de aplicaciones web y móviles que se ejecutan en todos los dispositivos.[[8]](#footnote-8)

#### Power Automate

Power Automate permite a los usuarios crear flujos de trabajo automatizados entre aplicaciones y servicios. Ayuda a automatizar los procesos comerciales repetitivos, como la comunicación, la recopilación de datos y la aprobación de decisiones.[[9]](#footnote-9)

#### Power BI

Power BI (Business Intelligence) es un servicio de análisis de negocios que brinda herramientas para analizar datos. Puede compartir esos conocimientos a través de visualizaciones de datos que componen informes y paneles para permitir decisiones rápidas e informadas.[[10]](#footnote-10)

#### Power Virtual Agents

Power Virtual Agents permite la creación chatbots utilizando una interfaz gráfica guiada sin código, permite construir y mantener conversacionales. Los usuarios pueden permitir que los chatbots realicen una acción simplemente llamando a un flujo de Power Automate. Los flujos ayudan a los usuarios a automatizar actividades o llamar a sistemas back end.[[11]](#footnote-11)

#### Power Pages

Microsoft Power Pages es una plataforma de alojamiento y desarrollo para crear sitios web comerciales modernos, seguros y centrados en datos. Power Pages permite diseñar, configurar y publicar rápidamente sitios web para clientes, socios y comunidades.[[12]](#footnote-12)

## Leguaje de programación, Power Fx

A pesar de que Power Platform esta orientada a soluciones No Code o Low Code, esta hace uso de un lenguaje de programación común en sus cinco productos principales, su nombre es Power Fx. Es un lenguaje de alto nivel de propósito general, fuertemente tipado, declarativo y funcional, esta última característica es la que define su uso, es muy similar a lo que se conoce como una formula de Excel, cuando tu quieres establecer un valor a componente, ya sea a una gráfica en Power BI, un valor de un flujo en Power Automate o el color de un botón en Power Apps lo haces por medio de una función o valor de una constante, Power Fx contiene por defecto gran cantidad de funciones como filtros, búsquedas, manejo de tablas, variables de diferentes tipos, ciclos de iteración, delegaciones, etc. Todas estas funciones precargadas ofrecen herramientas de fácil y rápida implementación, por un lado, permiten agilizar los tiempos de desarrollo y conocimientos necesarios para programar, por el otro, son poco personalizables y hacen la plataforma un poco rígida al no tener la flexibilidad de cambiar el código de esas funciones o al momento de crear nuevas.

Power Fx sigue la filosofía de Power Platform, ofrecer una herramienta que pueda ser usada por gente sin conocimientos de programación, cumple su función en casos simples como crear galerías, gráficas o flujos simples, sin embargo, si se quieren desarrollar soluciones ya mas complejas, con una lógica que sigue largos procesos, es necesario tener conocimientos de programación, personalmente, en los proyectos que he trabajado usando esta plataforma he aplicado conceptos como bases de datos, formatos JSON, llamadas a APIS, servicios de mapas y navegación, UI y UX, modelos de datos, diagramas de flujo, tipos de dato, inferencia de tipos de dato, expresiones regulares, funciones booleanas, etc. Todos estos conceptos han sido necesarios para lograr el desarrollo de aplicaciones, sin los cuales serían muy difícil lograr los objetivos, así que si, puedes crear aplicaciones sin tener conocimientos o una carrera en Ingeniería en Computación, pero las soluciones al integrarse y estar inmersas en otros servicios de cómputo, requieren conocimientos un poco más profundos para su desarrollo, desde los servidores hasta las paginas o aplicaciones en pantalla.

## SQL

Las aplicaciones de software giran alrededor de los datos y procesos, para este caso los procesos los automatizamos con usamos Power Platform y para los datos, una base de datos SQL. Una base de datos es un conjunto estructurado de datos que representa entidades y sus interrelaciones. La representación será única e integrada, a pesar de que debe permitir utilizaciones varias y simultáneas.[[13]](#footnote-13) El diseño de la aplicación a desarrollar debe permitir organizar la información y extraerla, una base de datos o Sistema Gestor de Bases de Datos (SGBD) nos permite hacerlo de forma rápida y eficiente, además de que ofrece las siguientes ventajas[[14]](#footnote-14):

* Consultas no predefinidas y complejas
* Flexibilidad e independencia, La complejidad de las BD y la necesidad de irlas adaptando a la evolución del Sistema de Información hacen que un objetivo básico de SGBD sea dar flexibilidad a los cambios. Interesa obtener la máxima independencia posible entre los datos y los procesos usuarios para que se pueda llevar a cabo todo tipo de cambios tecnológicos.
* Evitar problemas de la redundancia, la propia base de datos se encarga de mantener actualizados todos los lugares que tengan redundancia de datos
* Integridad de los datos, cuando el SGBD detecte que un programa quiere hacer una operación que va contra las reglas establecidas al definir la BD, no se lo deberá permitir, y le tendrá que devolver un estado de error.
* Concurrencia de usuarios, un objetivo fundamental de los SGBD es permitir que varios usuarios puedan acceder concurrentemente a la misma base de datos.
* Seguridad, los SGBD permiten definir autorizaciones o derechos de acceso, almacenar la información con una codificación secreta y otras mediadas de seguridad, que protegen la información almacenada.

Por estar razones y algunas otras que se comentarán más adelante en el desarrollo de la solución, se optó por una base de datos SQL como medio de almacenamiento, en específico, una base de datos en Azure SQL. Azure SQL es una familia de productos administrados, seguros e inteligentes que usan el motor de base de datos de SQL Server de la nube de Azure.[[15]](#footnote-15) Dicho de una forma más simple, es un servicio de Microsoft de bases de datos que vive en la nube basado en SQL Server (Un gestor de bases de datos de Microsoft).

## Blob Storage

Para el almacenamiento de datos no estructurados, por ejemplo, imágenes, video, música o documentos, se utilizó también un servicio de Microsoft dedicado conocido como Blob Storage, el servició nos permitió guardar iconos e imágenes usados en la aplicación, accedimos a ellos por medio de la URL pública que genera la plataforma. Hacer uso de este servicio tiene ventajas sobre usar el almacenamiento para media que viene incluido en la aplicación, lo retomaremos más adelante en el desarrollo.

## ERP

## Metodología

En la ingeniería del desarrollo de software existen etapas definidas que se tienen que seguir para la construcción de los sistemas, estas son análisis, diseño, desarrollo, pruebas e implementación, hay algunos autores que manejan algunas etapas más como el levantamiento de requerimientos o el mantenimiento del software ya en operación, pero se pueden tomar dentro de las etapas de análisis e implementación respectivamente. Cada etapa tiene sus características, lo que se busca es construir software de manera organizada, analizar la situación o problema a resolver, diseñar una solución que satisfaga las necesidades del cliente, desarrollar dicha solución, verificar que no tenga errores y asegurar que cumpla con lo acordado para finalmente ponerlo en producción y monitorear su desempeño. Las veces que se pasa por cada etapa o el enfoque que se le da a cada una varía según la metodología aplicada, pero el núcleo de las diferentes metodologías es el mismo, estas cinco etapas son la base de varias de las metodologías de desarrollo de software.

En este proyecto se usó la metodología de desarrollo de software conocida como metodología en espiral, modelo en espiral o desarrollo incremental, este se basa en la idea de diseñar una implementación inicial, exponer ésta al comentario del usuario, y luego desarrollarla en sus diversas versiones hasta producir un sistema adecuado. Las actividades de especificación, desarrollo y validación están entrelazadas en vez de separadas, con rápida retroalimentación a través de las actividades.[[16]](#footnote-16) La base de esta metodología es primeramente construir un prototipo funcional básico, validarlo, y una vez que sea aprobado continuar sobre este prototipo para seguir agregando características y funcionalidades hasta cumplir con todos los requerimientos.

Esta forma de trabajo es muy usada en ámbitos empresariales, pues tiene la ventaja de presentar avances en cortos periodos de tiempo y recibir validación del usuario de manera temprana, no siempre es posible aplicar esta metodología a los proyectos, hay ocasiones en las que el proyecto se tiene que realizar de la forma tradicional y secuencial debido a la complejidad o el ambiente de desarrollo, frecuencia de cambios, entre otros. En este caso al tener el desarrollo dividido en categorías de información y como cada categoría es independiente entre sí, se pudo aplicar la metodología de desarrollo en espiral, también algo que vale la pena mencionar es que el cliente estaba a disposición para consultas y validaciones, por lo que el modelo se ajustó muy bien al desarrollo.

# Definición del problema o contexto de la participación

Para que mi participación y los resultados obtenidos se entiendan mejor, a continuación, menciono algunos puntos de la empresa donde participé, la empresa para la cual se hizo el desarrollo y las demás entidades involucradas en este proyecto.

* Mi principal función dentro de la empresa es el desarrollo de aplicaciones con el uso de la plataforma de Microsoft Power Apps.
* Yo junto con dos de mis compañeros fuimos los principales responsables del desarrollo de la aplicación, uno de mis compañeros tiene el rol de Project Manager/Analista de negocio y el otro de desarrollador.
* Si bien participó un compañero de trabajo que también tenía el rol de desarrollador, él estaba enfocado en otra aplicación para el mismo cliente, su participación en esta aplicación se limitó al análisis y diseño. En el caso de mi manager, su participación también se dio durante las etapas de análisis y diseño, pero con una visión más enfocada a procesos de negocio, él tiene una formación en Administración.
* La aplicación forma parte de otras aplicaciones pensadas y diseñadas para soportar procesos de la empresa del cliente.
* Para el servicio ofrecido por nuestra empresa se creó una arquitectura con servicios de Microsoft que contempla dos bases de datos, conexiones a otras bases, almacenamiento por blob storage, entre otros. Tal arquitectura ya estaba planteada cuando me integré al proyecto y fue realizada por uno de mis supervisores.
* Hay una empresa con la cual se estuvo trabajando, es una consultoría de negocios, esta fue la que vendió al cliente las aplicaciones (entre otras cosas), ellos se dedican a analizar procesos y proponer nuevos para obtener mejores resultados en tiempos y costos, dicha empresa fue la responsable de darnos el requerimiento y las historias de usuario.
* El cliente para el cual se desarrolló la aplicación es una empresa dedicada a la producción y comercialización de vegetales, es una empresa de mediano tamaño establecida en Estados Unidos con más de 40 años de existencia y más de 10 años de operaciones en México.

Para dar más detalle de mi participación, en el periodo en el cual desarrollé la aplicación no tenía mucha experiencia laboral dentro del desarrollo de software, tenía un poco experiencia desarrollando aplicaciones en la misma empresa para clientes previos, en concreto 2 aplicaciones en un periodo de alrededor de 4 meses, así mismo aún seguía (y sigo) aprendiendo acerca de la herramienta Power Apps, bases de datos, metodologías, procesos, etc.

Es importante mencionar que yo desconocía totalmente el concepto de Sales & Operations Planning, fue un reto para mi aprender de ello ya que no estoy muy familiarizado con los términos y procesos de negocio, además no hay como tal una receta o camino claro para implementar el concepto, Sales & Operations Planning es más una guía que un instructivo o receta, según sea la empresa y situación específica es la forma implementarlo.

Existen ya en el mercado softwares de S&OP como streamline, Aspen Technology, Oracle o SAP entre otros, la mayoría de ellos manejan una estructura muy similar, un calendario de ventas programadas y un reporte de lo vendido y suministrado, si bien los softwares comerciales por lo general son más completos y ofrecen más características, comprarlo no era opción viable para este desarrollo por 4 razones:

* Se tenía que extraer información de un ERP poco conocido y con dificultades de acceso
* El costo era muy elevado para el cliente
* La función que se buscaba era más simple que las ofrecidas por softwares comerciales
* Se buscaba centralizar la información en un mismo lugar (base de datos)

Estos puntos se detallarán más a fondo en los siguientes apartados.

# Participación profesional

Antes de describir el proceso de construcción, hay que considerar el hecho de que en la empresa se tiene también una metodología de trabajo, anteriormente se describió la metodología de desarrollo, esta nos dice la manera en la que se va a construir el software, sin embargo, antes de eso hay que revisar la administración del proyecto, este proceso de planear y ejecutar esta un nivel más arriba de la construcción, pues considera tiempos, alcances y recursos en base al análisis del problema y capacidad de la empresa, en otras palabras, la metodología de desarrollo nos dice el cómo se va a construir y la administración de proyectos nos dice quien, cómo y cuándo.

A continuación, se redactan las etapas por las que atravesó el proyecto siguiendo la metodología planteada, pero primero, un análisis del requerimiento de manera general, algo que se conoce como una vista a 30 000 pies de altura a partir de la cual se asignan tiempos, recursos y se delimitan alcances, se le dice así porque el objetivo de la reunión es ver a nivel macro todo lo que hay que implementar o desarrollar, no nos metemos tanto en detalles y priorizamos el establecimiento de los objetivos, también se da un contexto de nuestra participación, el modo de trabajo tanto de consultoría como de tecnología y se aclaran dudas acerca del proceso o información del cliente.

## Vista a 30 000 pies de altura

Para construir o desarrollar es importante saber qué es lo que se quiere construir, saber cuál es el objetivo general es de vital importancia, para eso, mi equipo (Manager, Supervisor y Desarrollador) y yo pedimos una plática al consultor en donde se nos dijera cual es el objetivo de la aplicación, esto nos sirve para dimensionar el alcance y empezar a estimar tiempos, además, permite que todo el equipo este enterado de cuál es la meta y en que parte colabora, es importante que en esta reunión participen todos los involucrados del proyecto para alinear los objetivos desde un principio.

Otro tema que se tocó en esta reunión fue una explicación muy rápida de lo que es S&OP, el consultor nos comentó que se buscaba una conciliación entre producción y ventas, para lograrlo, implementó un spreadsheet o formato en Excel en donde colocaba información de ambos departamentos, implementarlo así es algo común entre la bibliografía[[17]](#footnote-17) e incluso hay estudios que dicen que una mínima o básica implementación de S&OP puede hacer grandes diferencias.[[18]](#footnote-18)

Application, table

Description automatically generated

Ilustración - Formato original de S&OP por parte de consultoría

Al final de la reunión se nos comentó que el objetivo era automatizar la extracción de información que contenía ese formato y al mismo tiempo implementarlo en una aplicación para que se guardara el consenso de S&OP (consensus forecast en la ilustración 1).

## Alcances

A lo largo del documento se habla de categorías de información, para entender mejor esto hay que revisar la ilustración 1, acorde al objetivo e implementación de S&OP, la planeación de cada departamento o área debe ponerse sobre la mesa y someterse a discusión, pues bien, a la información generada por cada área de la empresa se le asignó el nombre de categoría, cada una de ellas es capturada de manera independiente, después de unas pocas reuniones con el consultor se determinó que no todas las categorías eran posibles de desarrollar debido a que no existía fuente de información y la captura quedaba fuera de alcance, al final, las categorías y subcategorías a desarrollar fueron las siguientes:

|  |  |  |  |
| --- | --- | --- | --- |
| Categoría | Subcategoría | Descripción | Fuente de datos |
| Forecast | Consensus Forecast | Numero de cajas/unidades a vender acordadas entre producción y ventas | Es un dato que se discute durante la reunión semanal de S&OP, la aplicación debe guardar ese número. |
| Demand Forecast | Open | Cantidad proyectada de ventas que serán vendidas a precio de mercado | Antes de terminar el año se hace una planeación anual o semestral para el siguiente año, dicha información se guardaba en el archivo original de Excel de S&OP. |
| Fixed | Cantidad proyectada de ventas que serán vendidas a precio fijo |
| Hi – Low | Cantidad proyectada de ventas que serán vendidas a precio fijo o precio de mercado según quede fuera de un rango acordado. |
| Production Forecast | Ranch | Cantidad proyectada de producción por parde de los ranchos de la empresa | De la misma forma que Demand Forecast, se hace una planeación anual o semestral de lo que se va a cosechar en cada rancho y es guardada en el archivo original de Excel de S&OP. |
| Program Growers | Cantidad proyectada de producción por parde de los ranchos bajo acuerdo o contrato |
| Non Program Growers | Cantidad proyectada de producción por parde de los ranchos fuera de la empresa y que no están bajo acuerdo o contrato |
| Actual Sales | Open | Cantidad real de producto vendido a precio de mercado | ERP de la empresa |
| Fixed | Cantidad real de producto vendido a precio fijo |
| Hi - Low | Cantidad real de producto vendido a precio fijo o precio de mercado según quede fuera de un rango acordado. |
| Actual Supply | Total | Cantidad total de entradas a inventario | ERP de la empresa |
| Referent Information | Original Sales Budget | *Nunca se recibió definición para este apartado* | - |
| Last Year Sales | Ventas del año pasado para el mismo periodo de tiempo | ERP de la empresa |
| Delta Actual Year/Last Year | Diferencia en % entre las ventas del año pasado y las actuales para el mismo periodo de tiempo | ERP de la empresa |
| Stock | On Hand | Cantidad de producto en almacén | ERP de la empresa |
| In Transit | Cantidad de producto en camino a almacén | ERP de la empresa |

Tabla 1 – Categorías de S&OP

Adicional a la extracción de información se planteó el desarrollo de unos indicadores, estos indicadores estaban en su mayoría definidos a partir de la información propia de la aplicación y algunos otros mostraban datos de fuentes externas, en general no eran muy complejos de desarrollar y se acordó incluirlos dentro de la aplicación en la misma pantalla del layout. Un aspecto importante que se revisó es que PowerApps no esta diseñado para la visualización y/o desarrollo de indicadores, Power BI sí, solo que como los indicadores a desarrollar eran simples y mis compañeros de trabajo encargados de desarrollar indicadores estaban saturado, decidí que la mejor opción era implementar los cálculos en la base de datos con ayuda de las funciones de SQL Server y vistas, para luego mostrar los resultados en la aplicación, más tarde se detallará esta parte.

Cuando entendimos que era S&OP (de manera general) y el formato desarrollado por parte del consultor, se le pidió al consultor que llenara junto con el usuario principal un formato con las historias de usuario, esto para que pudiéramos tener la lista de requerimientos y delimitar alcances. Le explicamos la manera en la que se tenía que llenar y la necesidad de describir roles, al principio las historias de usuario se redactaron de manera muy poco especifica, es decir, no explicaban a detalle las funcionalidades que deberían de tener y parecían hablar más de beneficios generales, se habló con el consultor y se le mostró un ejemplo para que las redactara con un nivel más de detalle, sin embargo, la versión 2 también presentaba las mismas deficiencias y nos hablaba de una aplicación que no podía ser construida en el tiempo establecido, los alcances hablaban de una aplicación muy completa, con simulaciones, alertas, manejo de versiones, análisis estadístico, rastreabilidad del producto, etc. No es que no se pudiera construir, pero el tiempo que se tenía asignado para esta aplicación era muy corto por lo que se optó construir lo básico, extracción de información y guardado del dato del consenso. A continuación, se muestran las historias de usuario provistas por el consultor.

**Tabla 1. Historias de usuario de S&OP**

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Rol** | **Quiero / Me gustaría** | **Para que pueda** |
| 1 | Sales Manager | Quiero saber si la demanda supera la capacidad de abastecimiento de la empresa | para poder buscar más fuentes de abastecimiento en tiempo y forma para así poder disminuir las ventas perdidas y maximizar las ganancias de la empresa |
| 2 | Procurement Manager | Quiero saber si la demanda supera la capacidad de abastecimiento de la empresa | para poder buscar más fuentes de abastecimiento en tiempo y forma para así poder disminuir las ventas perdidas y maximizar las ganancias de la empresa |
| 3 | Sales Support | Hacer análisis estadístico y colaboración del área comercial | para poder planificar la demanda y así poder cumplir mi rol como soporte a las ventas y poder hacer un forecast semanal para las siguientes 52 semanas |
| 4 | SCM Planner | Hacer análisis de capacidad con el fin de buscar el mayor cumplimiento al plan de ventas | para poder planificar el suministro y así poder cumplir mi rol como soporte a la producción y poder hacer un forecast semanal para las siguientes 52 semanas |
| 5 | SCM Planner | definir el mejor plan basado en la estrategia de la empresa | para llegar a un acuerdo/consenso entre ventas y operaciones |
| 6 | SCM Planner | quiere conocer las restricciones y ajustes que requiere | para llegar a un acuerdo/consenso entre ventas y operaciones y ejecutar el abasto contra un único forecast de la empresa |
| 7 | Sales Support | quiere conocer las restricciones y ajustes que requiere | para llegar a un acuerdo/consenso entre ventas y operaciones y ejecutar las ventas contra un único forecast de la empresa |
| 8 | SCM Director | quiero medir el cumplimiento de los acuerdos y reportar la evolución | para poder saber que mi área puede cumplir con las necesidades del área de Producción y poder reportar a la Alta Dirección el performance del proceso |
| 9 | Sales manager | quiero medir el cumplimiento de los acuerdos y reportar la evolución | para poder saber que mi área puede cumplir con las necesidades del área de Ventas y poder reportar a la Alta Dirección el performance del proceso |
| 10 | Finance Director | quiero medir el cumplimiento de los acuerdos y reportar la evolución | para saber si mi flujo financiero cumple con lo requerido por la operación y poder reportar a la Alta Dirección el performance del proceso |
| 11 | SCM Director | Alinear, integrar y sincronizar la cadena de suministro con el mercado | Identificar necesidades de decisiones estratégicas y facilitar su resolución |
| 12 | SCM Planner | quiero basado en las restricciones, plan de negocio y evaluación financiera se generan los escenarios posibles y su viabilidad | para plantear escenarios alternativos y posibles planes de acción |
| 13 | Sales Support | quiero tener un repositorio de información de años atrás | para poder tener un histórico de información y poder hacer una previsión basado en información histórica |
| 14 | Transport Manager | quiero poder planificar la necesidad de la demanda y el abasto, y de las ubicaciones | para así poder calcular la necesidad de transporte y poder negociar una mejor tarifa con mis proveedores |
| 15 | El Sales Manager | quiere maximizar las ventas, asegurar que todos los clientes existentes tengan producto, y que la empresa pueda ganar más clientes … | Y para eso necesita hacer una previsión de ventas la más acertada posible a las necesidades de los clientes actuales y futuras, así como alinear estas necesidades a las posibles restricciones que pudiera tener la producción |
| 16 | El SCM Director | quiere conocer la proyección de ventas sobre un tiempo mayor que es su tiempo de abasto (Lead time) que para productos de compra son 7 días y para productos de plantación son 6 meses) | para poder asegurar la correcta secuenciación de los volúmenes y poder estar en sincronía con las cantidades necesarias para que ventas pueda satisfacer sus clientes |
| 17 | El Finance Director | quiere poder estimar las necesidades de capital, así como planificar la proyección del Cash Flow de la empresa | para poder asegurar la liquidez de la empresa y poder planificar ganancias y pérdidas, junto con los flujos de ingresos y egresos |
| 18 | El Sales Support | quiere poder saber las ventas pasadas y proyecciones futuras, así como las restricciones de producción y transporte | para poder aplicar métodos de análisis estadístico y modelos de previsión con los que se pueden orientar Ventas y Producción para sus planificaciones |
| 19 | El Transport Manager | quiere poder conocer cómo se conforman los flujos de producto en cuanto a fechas, ubicaciones, orígenes & destinos, volúmenes y mixes | para poder dimensionar qué transporte de dónde a dónde se requiere para qué momento de la semana / mes en el futuro |
| 20 | SCM Planner | quiere conocer todo lo anterior para poder analizar, organizar, coordinar y reportar el proceso tanto de ventas como de producción | para poder dar la mayor transparencia de información a todas las áreas de la empresa y poder planificar con un solo número alineado con la estrategia y la visión de la Alta Dirección |

Al ver las historias de usuario notamos que se describían varios roles, sin embargo, al recortar los alcances y ver en el cliente una insuficiente maduración en sus procesos, decidimos implementar permisos basados en funciones, es decir, a cada usuario se le habilita o deshabilita el permiso de editar cierta categoría de información, esta configuración es editable y puede ser modificada desde una pantalla de administración, más adelante se verá el desarrollo de esta parte.

## Tiempo

Como se mencionó anteriormente, una de las razones por las que se decidió construir algo básico, al menos dentro de una primera fase, fue el tiempo. La necesidad de esta aplicación fue expresada a inicios del mes de noviembre por parte de consultoría y se quería que la aplicación fuera terminada máximo a finales de febrero, es decir, en 4 meses. Es importante recalcar que a la par se llevaba el desarrollo de dos aplicaciones más, por lo que mi equipo y yo no estábamos dedicados al 100% para esta aplicación, como consecuencia, la fecha de inicio del desarrollo se recorrió hasta diciembre y el tiempo estaba ya muy ajustado.

Finalmente, después de delimitar el alcance y ver las actividades a realizar, se acordó terminar la aplicación en los 4 meses, así mismo, se estableció que se trabajaría con una metodología ágil desarrollando las 7 categorías o rubros de la aplicación de forma iterativa, el plan fue el siguiente:

Graphical user interface, application, table, Excel

Description automatically generated

Ilustración - Plan de desarrollo de S&OP

En el diagrama de la ilustración 2, en la parte derecha se puede observar por medio de un diagrama de Gantt que el tiempo total supera por unos días la fecha límite, esto fue preocupante para ambas partes. A pesar de que la extracción de información se haría más rápido ya que se tendría aprendizaje de fases anteriores no se pudo hacer más para acortar los tiempos de entrega, al final, la necesidad de esta aplicación, la urgencia y el presupuesto terminaron por aceptar los tiempos de entrega propuestos por nosotros.

Cabe resaltar que en cada iteración se siguió la metodología descrita en el marco teórico, el propósito era crear la aplicación agregando en cada iteración funcionalidad de forma incremental, para lograr esto, cada una de las iteraciones cuenta con las etapas básicas del desarrollo de software, es decir, todas las iteraciones pasaron por las etapas de análisis, diseño, desarrollo, pruebas e implementación, para al final, ir liberando e incrementando funcionalidad.

## Recursos

Cuando se habla de recursos, una de las principales cosas a considerar es el presupuesto, este tema queda fuera de mi alcance, pero es importante mencionar que es un factor clave, cada herramienta, persona y tiempo invertido genera un costo, este no tiene que ser mayor al precio de venta y balancearlo es importante para poder seguir operando, de lo que sí puedo hablar es de los recursos humanos, mi equipo y las herramientas que usamos para empezar a trabajar en el desarrollo.

Se asignaron 3 recursos al proyecto general para el cliente, como mencioné antes, había 2 programadores y un analista de negocio, todo el equipo estaba presente en las juntas de análisis y definición con el cliente, sin embargo, una vez analizado y discutido el requerimiento y habiendo creado una maqueta o diseño en conjunto, el desarrollo era mi responsabilidad. Adicional a los recursos humanos, para el proyecto se asignó una cuenta para poder trabajar en el “tenant” (ambiente o sitio vendido por Microsoft) del cliente, incluida la licencia para poder empezar a programar con Power Platform y una base de datos en la nube (Azure SQL). Ya con esto se empezó a trabajar con el cliente en cada una de las iteraciones.

## Iteración 1 – Consensus Forecast

La primera iteración se basaba en construir la primera de las categorías (ver tabla 1) conocida como “consensus forecast”, este conjunto de información es el número al cual se llega a acuerdo entre producción y ventas, este dato tiene que guardarse dentro de la aplicación.

A raíz de las reuniones con el cliente y con el consultor, se analizaron los siguientes puntos, estos son muy importantes ya que marcaron la base de las siguientes iteraciones.

* La información de S&OP tenía que ser mostrada de forma semanal
* No todas las personas pueden editar todas las categorías y no todas las categorías son editables.
* La información debe de ir agrupada por conjuntos de productos, las reglas nos fueron especificadas previamente para determinar que productos entran en qué grupo, llamamos a estos grupos “planes”.
* La información puede ser vista por unidades o cargas (camiones) equivalentes.
* El idioma de la aplicación debe ser inglés.

Lo primero que suelo hacer para analizar un problema es organizarlo, sabiendo el alcance y las diferentes partes del problema decido organizar la aplicación en 3 partes, secciones o menús

* Menú 1 – Summary: Contiene el valor de los indicadores.
* Menú 2 – S&OP: Pantalla principal de la aplicación que contiene los diferentes grupos de productos (planes) y cada uno de ellos el layout con su información correspondiente.
* Menú 3 – Settings: Aquí se pueden ajustar parámetros y permisos para la aplicación, solo ciertas personas tendrán acceso a esta pantalla.

Teniendo claras las partes y contenido, pasamos de lo general a lo particular, empezamos a diseñar el contenido a detalle de la aplicación, dejamos la sección de summary y settings para el final debido a que su desarrollo dependía de tener completa la información de las categorías y tener programadas ya las secciones o funciones para fijar permisos.

Para la etapa de diseño de esta iteración y las posteriores se tomó como base el layout o formato que inicialmente construyó el consultor, mantener el mismo formato tenía la ventaja que el usuario ya estaba acostumbrado al mismo, se hicieron solo unas cuantas modificaciones en cuanto a colores y organización, el diseño final consistió en 2 pantallas principales como se muestra en las siguientes imágenes.



Ilustración - Mock up de los planes para S&OP

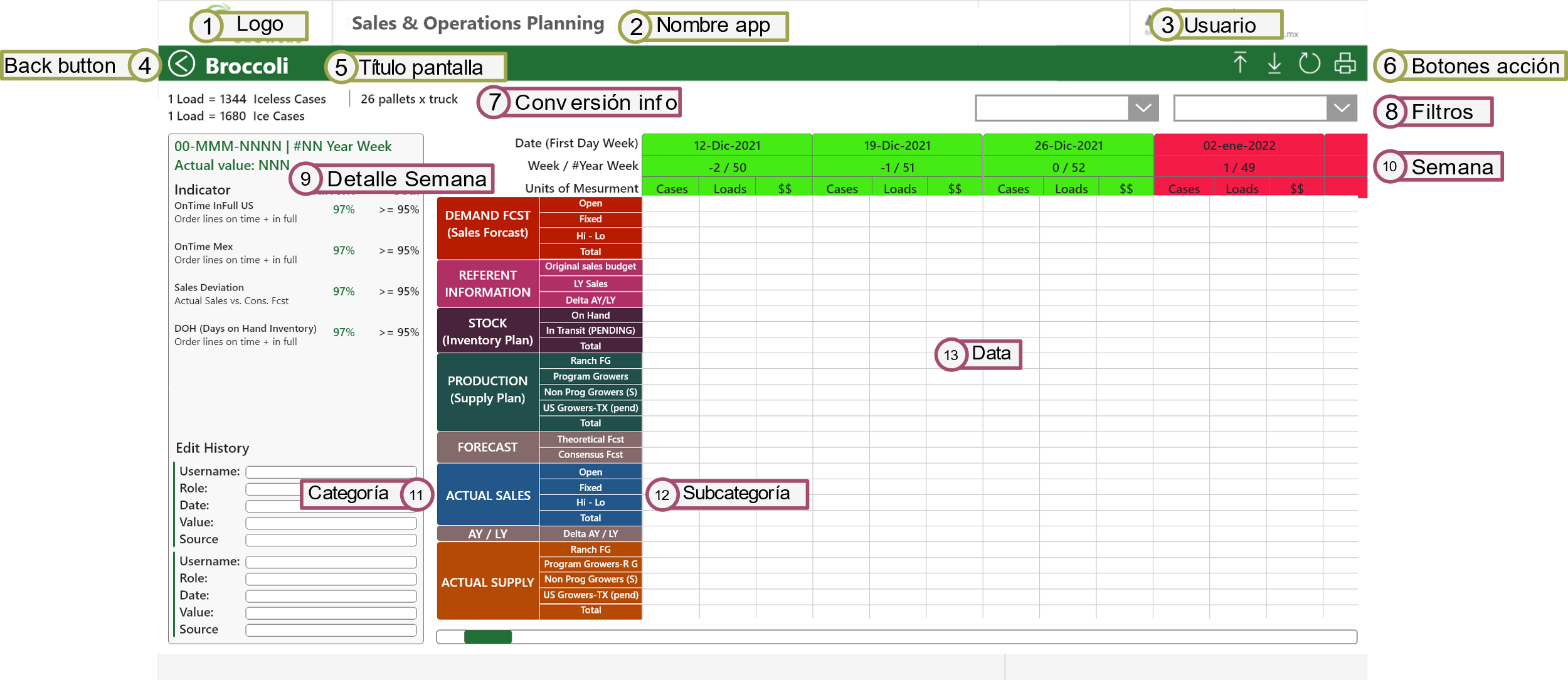


Ilustración 4 - Mock Up S&OP

Por razones de confidencialidad y privacidad de datos se ocultaron el logotipo y nombre de usuario.

Se puede observar en las imágenes anteriores que ambas tienen una estructura similar, esta estructura la heredamos de proyectos anteriores y tiene su razón de ser, en la parte superior se encuentra lo que en diseño de aplicaciones se conoce como “Title bar”, contiene el logo del cliente (1), el nombre de la aplicación (2) y la información del usuario (3) que actualmente esta usando la aplicación, se incluye la imagen, correo y nombre, estos últimos son extraídos del directorio de usuarios de Microsoft dentro del tenant de la empresa. Debajo de esta title bar esta la “action bar”, contiene el botón de regresar (4) en el caso de la ilustración 4, el nombre de la pantalla actual y los botones de acción, que nos permiten realizar funciones como actualizar datos, descargar los datos o imprimir el contenido de la pantalla actual. En el caso de la ilustración 3, la que contiene los planes, se tiene la “bar title”, que tiene los tres menús de la aplicación.

La propuesta de estas pantallas consiste en organizar los diferentes grupos de productos en planes (ilustración 3) donde cada tarjeta representa un plan, indicando algunos datos principales y una imagen alusiva al grupo, al dar clic en cada tarjeta se pasará a la segunda pantalla “ilustración 4” donde se tendrá la información de cada categoría correspondiente al plan seleccionado, esta pantalla está conformada por 2 secciones, la sección de detalles a la izquierda (9) muestra información de la semana que se seleccione, algunos indicadores y el historial de edición, la sección de datos, que está a la derecha (13) contiene la información de cada categoría (11) y subcategoría (12) correspondiente a cada semana, además unos filtros (10) para seleccionar el periodo a consultar y finalmente alguna información de referencia que nos dice el factor de conversión de cargas a unidades entre otros datos (7).

Con un proyecto anterior, además de algunos proyectos propios, comprendí que el diseño es muy importante para la construcción de aplicaciones, es por eso que pongo especial detalle en esta etapa, hay toda una ciencia detrás del diseño, cada texto, color, botón y elemento tiene que comunicar información y permitir al usuario realizar funciones de forma fácil y rápida, hablando un poco de intereses personales, tengo un gusto particular por el diseño, me gusta hacer que las aplicaciones no solo funcionen bien, también que se vean bien, que tengan consistencia en sus elementos y este todo bien organizado, este proyecto fue la oportunidad perfecta para empezar a meterme a profundidad y practicar, pues no tenía mucha experiencia en la construcción de una aplicación.

Detallar cómo fue el diseño de cada elemento no forma parte de este reporte, sin embargo, menciono que el diseño tiene cierta influencia del sistema de diseño de Google conocido como “Material Design”, es un conjunto de guías, herramientas, principios y lineamientos que nos permiten crear o diseñar aplicaciones modernas, brindando una experiencia digital acorde a los estándares de diseño de aplicaciones, la razón por la que es fácil adecuarnos a usar otras aplicaciones sin haberlas usado antes (al menos para las nuevas generaciones), es que todas ellas se comportan de manera similar, cuentan con elementos en común que han sido adaptados por la comunidad y aplicados a la mayoría de desarrollos.

Finalmente me gustaría concluir esta parte de diseño resaltando la importancia de mantener tu concepto a través de toda la aplicación, las formas, colores, botones, textos tienen que ser homogéneos a través de todas las pantallas de la aplicación y seguir comunicando o cumpliendo con el objetivo principal de la aplicación, una muestra de esto está en los colores de las cajas de texto, se usa siempre el mismo color para que el usuario sepa que al ver el color verde, puede o debe introducir datos, podría continuar con este tipo de detalles pero es un tema bastante extenso y del cual sigo aprendiendo, me limitaré a solo decir que puede que el diseño no sea perfecto pero cada elemento tiene su razón de ser y puse especial cuidado en su construcción.

Al finalizar el diseño, se expuso al cliente la propuesta, no comentó mucho al principio, mencionó que la interfaz era muy clara y amigable, también que la sección de historial de edición no era tan importante y podía ser reducida o incluso omitida, nos dio unos consejos para quitar información poco relevante y de forma general estuvo de acuerdo con nuestra maqueta, nos dio el visto bueno necesario para empezar a construir. Importante resaltar que recibir retroalimentación temprana es crucial en esta metodología, nos permite hacer correcciones tempranas y hacer cambios en el prototipo, hacerlo de esta forma es más rápido y barato que realizarlos luego en la aplicación, hay que encontrar nuestros errores antes de que sea demasiado tarde.

Finalizado el diseño la etapa siguiente acorde a la metodología es el desarrollo, el punto de partida es el diseño de la base datos, la arquitectura a construir debía ser capaz de guardar la información provista por los usuarios y permitir en un futuro otro tipo de planeación, el modelo de datos que construí para almacenar esta información fue el siguiente:

Table

Description automatically generated

Ilustración - Modelo datos S&OP

De manera general, se tiene una tabla de planes (c\_plan) que contendrá 1 plan por cada grupo de productos, el plan para brócoli, el plan para zanahoria, etc. Un plan está conformado por unidades de tiempo (c\_plan\_unit), en base al análisis sabemos que cada plan tiene un seguimiento semanal, sin embargo, para dar mayor escalabilidad al sistema el periodo queda abierto con 2 fechas, siendo la resolución mínima de 1 día, esto da oportunidad a tener en un futuro planes mensuales, anuales, etc. Para controlar el tipo de plan, se creó la tabla c\_plan\_type, después tenemos la tabla t\_plan\_data, en donde se almacenará la información necesaria, finalmente, las tablas de c\_plan\_category y c\_plan\_subcategory nos ayudan a identificar a que categoría y subcategorías pertenecen los datos guardados en t\_plan\_data. Dicho de otra forma más simple, el modelo de datos trata de manejar planes, cada plan abarca cierto tipo de productos, dentro de cada plan se encuentran unidades y dentro de cada unidad datos a la correspondiente duración de las unidades. A pesar de que fue la primera iteración, a la hora de crear el prototipo y el modelo de datos se contemplaron todas las iteraciones por practicidad, al final la información es muy similar en cuanto a estructura en todas las categorías.

Con el modelo de datos y el diseño terminados, la siguiente fase es el desarrollo, al igual que la etapa de diseño, no se explicará a detalle el proceso, solo se dará una redacción a nivel general y se resaltarán problemas y puntos más importantes.

En Power Apps existen ya variedad de elementos por default, etiquetas de texto, botones, drop downs, datepickers, etc. Estos objetos son configurables, ya sea estableciendo valores directamente o por medio de fórmulas con opción de poner referencias a datos u otros objetos (como en Excel), en general la programación individual de estos objetos no es muy compleja, sin embargo, si se quiere dar una funcionalidad más completa, rápida y un mejor aspecto visual hay que aplicar algunos trucos, el resumen del desarrollo es el siguiente:

* El contenido de la “tittle bar”, “action bar” y “footbar” son partes que se toman de una plantilla, esta cuenta con la funcionalidad para extraer información del usuario, de la aplicación y también contiene la configuración necesaria para los colores, todos estos datos residen en la misma base de datos configurada previamente, por lo que si se quiere comenzar a crear una nueva aplicación lo que se hace es duplicar la plantilla, ponerla en el sitio de trabajo y crear las tablas iniciales necesarias para empezar a trabajar.
* El acomodo de información fue una parte compleja, en el diseño se tiene una tabla que contempla 2 dimensiones principalmente, subcategorías en la parte vertical y fechas en la parte horizontal, adicional se tienes también las categorías y las diferentes formas de presentar la información (cargas, unidades o precio/costo), powerapps al ser una herramienta de alto nivel no permite mucho personalizar sus componentes, por lo que se optó por separar la parte de subcategoría/categoría de la información semanal, visualmente parecen un solo grupo, pero en realidad son dos objetos que no se relacionan, para que coincidiera la información se usó una columna de orden, así, aunque no estuvieran ligadas la información se ordena con un mismo criterio y valores haciendo coincidir la información, es un poco “frágil” este acomodo pero en mi experiencia usando power apps, a veces la mejor solución no es precisamente la más “elegante” o más eficiente, sobre todo cuando consideras tiempos y costos.
* Powerapps te deja trabajar con Azure SQL por medio de un conector, dicho conector hace la conexión a la base de datos con datos del el usuario de Microsoft, por lo que se tuvo que crear un pequeño stored procedure en la base de datos para dar acceso a los datos de la aplicación con permiso de lectura y escritura para cada cuenta que quisiera usar la app.
* Para generar las unidades (semanas de cada plan) se creó un pequeño ciclo de iteración sobre cada plan y semanas del año dentro de powerapps, en cada iteración se iba creando 1 semana y su respectiva numeración, fue complicado hacer que coincidiera con su forma de trabajar las semanas, sobre todo a inicios y fin de año, cuando las semanas no son completas. Se generaron registros de semanas que contemplan 10 años para cada plan y se hizo desde powerapps con 2 ciclos de iteración anidados, no recomiendo generar tablas calendario dentro de powerapps, ahora me doy cuenta de que es mucho más fácil y rápido hacerlo con lenguaje SQL, solo que en ese entonces no tenía tanto conocimiento sobre él.
* Para el guardado de los datos se decidió guardar tanto cases o units como loads, aunque es el mismo dato solo que en unidades diferentes ambos son guardados en la base de datos, en este punto me doy cuenta que no fue la mejor forma de resolver el problema, hubiera sido mejor guardar solo 1 y calcular el otro, ya sea en una vista o en una columna calculada, sin embargo, al ser pocos datos (53 al año) no representa problemas por almacenamiento y permite extraer de forma más rápida el dato.

En general el desarrollo de la aplicación fue rápido, lo más tedioso y largo fue el acomodo de información, el resultado final es el siguiente:

Graphical user interface, application, table, Excel

Description automatically generated

Ilustración – desarrollo de consensus forecast

En la ilustración 6 se puede observar que el desarrollo final para esta iteración es muy similar al mock up (ilustración 4), lo cual es muy bueno, en general la aplicación hasta este punto es muy simple, pues solo guarda el dato almacenado, ventas y producción necesitan ver todos los demás datos y debido a esto no es posible que el usuario empiece a usarla, pues no genera valor o utilidad alguna, esto suele suceder a veces cuando se realizan desarrollos siguiendo esta metodología, se tiene un mínimo producto viable (MVP por su siglas en inglés) pero tiene tan poca funcionalidad que no es útil, hay veces que incluso no es posible empezar a usar el producto hasta el final debido a que se necesitan todas sus partes para el correcto funcionamiento, es importante aclarar que una cosas es el desarrollo en iteraciones y otra cosa la implementación o pruebas, puede ser que el usuario no pueda usar aún la aplicación, pero en este punto ya se tiene desarrollado un MVP que representa un importante avance en el proyecto.

## Iteración 2 – Demand / Production Forecast

La siguiente iteración consistió en desarrollar la parte de la aplicación que extrae y guarda la información de su forecast o pronóstico, a mitad del año y unos meses antes del fin de año ventas y producción ponen en archivo de Excel lo que planean vender/producir para los siguientes 6 meses, aunque venden una mediana selección de productos realmente los productos más vendidos son unos cuantos, por lo que solamente consideran la venta de unos cuantos grupos de productos, a veces estos grupos están conformados por varios productos y otras por solo 1 producto, esto es lo que en nuestro análisis llamamos planes, para esta iteración la agrupación es relativamente sencilla dado que su archivo de Excel nos indica qué grupos son y la información ya viene separada por hojas en el archivo, no obstante había algunas categorías que variaban entre cada hoja e inconsistencia de información que poco a poco fuimos resolviendo, el desarrollo final para esta iteración consistió de 2 partes:

* Una plantilla y script en Excel para extraer la información. A causa de las inconsistencias y poca homogenización de la información se decidió crear un nuevo archivo de Excel donde ellos estarían poniendo su forecast, se estuvo debatiendo esta implementación porque el usuario quería mantener el Excel por su flexibilidad para realizar cálculos, prácticidad, uso de fórmulas, funciones para copiar y pegar información mientras que nuestro equipo empujaba que la información fuera capturada de la información y así evitar crear el script para extraer la información, al final se reconoció la necesidad del cliente por mantener el Excel pero se pusieron algunas restricciones al archivo para asegurar la integridad de la información y facilidad de extracción con Power Automate. Fue la primera vez que usaba scripts de Excel usando TypeScript, por lo que esta parte me tomó más tiempo del planeado.
* Creación de las categorías Production y Sales Forecast. Ya con la información en nuestra base de datos solo se tuvo que mostrar la información en la aplicación en el apartado correspondiente, el desarrollo fue muy similar a la primera iteración, los datos son guardados y la información mostrada en diferentes unidades, estas categorías al igual que el consensus forecast son editables, pues cada jueves ellos van ajustando los números colocados según las condiciones del mercado.

Table

Description automatically generatedIlustración - Production/Sales Forecast

Table

Description automatically generatedIlustración - Import Template

Un punto importante que no pudo ser resuelto fue el hecho de que para la aplicación se propuso un control de la modificación de datos, para ello se desarrollaron triggers en la tabla t\_plan\_data con el objetivo de registrar cambios y hacer la conversión de units a loads cada que se modificara o insertara un dato, sin embargo me encontré con que powerapps no es capaz de trabajar con tablas que tengan triggers, lamentablemente no encontré otra solución mas que trabajar la conversión dentro de la aplicación y dejar el historial de edición con información de la ultima modificación, guardando quien y cuando se realizó.

En esta iteración se tomó más tiempo del planeado debido a la dificultad para extraer los datos del Excel, entre la configuración del entorno, programación y pruebas, también el no poder trabajar con triggers en tablas causó un poquito de retrasos en tiempos ya que se estuvieron programando hasta darme cuenta de que no era posible usarlos, el tiempo consumido adicional fue de 1 semana, como aún estábamos a tiempo no ocasionó alarma y mi equipo entendía perfectamente el trabajo necesario y tiempo para construir el proyecto, adicional a eso, también se validó el proceso de importación de datos con el usuario y quedó aprobado, a mi parecer no le quedó muy claro pero el contacto sigue estando abierto incluso hasta ahora para cualquier duda, la relación de TI con el cliente en general es muy sana.

## Iteración 3 – Actual Sales

Las ventas que tiene la empresa son registradas en su ERP como órdenes de venta o “Sales Order”, aquí se incluyen datos como el cliente, fechas, folio (Sales Order Number) etc, y también datos de los productos vendidos, cantidades e importe total entre otros datos, una vez que se confirma la orden, esta es vista por logística y envía el producto al cliente. Así es como funciona su ERP en el proceso de ventas, nuestra tarea en esta iteración fue extraer los datos de su ERP y mostrar las ventas agrupadas por semana y por productos (planes), a pesar de que teníamos acceso directo a la base de datos y un poco de conocimiento del negocio, no fue fácil extraer los datos que residían en el ERP debido a que para nosotros muchas de sus funcionalidades y datos eran y siguen siendo una caja negra para nosotros.

### Extracción

Para lograr concretar esta iteración se tuvo que pedir ayuda a otro integrante del proyecto encargado de generar indicadores, él era responsable de la generación de dashboards y entre sus tareas tenía que extraer, transformar y presentar datos en forma de indicadores, las fuentes de datos eran varias, exceles, apis, sharepoint, correos, pero principalmente el ERP de la empresa, mi compañero de trabajo se integró antes que yo al proyecto y ya tenía cierto camino recorrido y entendimiento de la estructura del ERP, al comentarle de la situación y de lo que se quería lograr el ofreció depositar en una tabla en mi base de datos la información final y necesaria a mostrar en la aplicación, al principio ese era el plan, pero después de notar que estaba muy ocupado y que el conocimiento de lo que se quería lo tenía yo, decidimos que la mejor forma de trabajar era que el extrajera la información en bruto del ERP, la pasara a mi base de datos y yo luego la transformara a como se necesitaba en la aplicación. Así se hizo y para lograrlo se utilizó la herramienta de Microsoft llamada Data Factory, con esta herramienta se programaba diario un proceso que se ejecutara a las seis de la mañana, accedía a la base de datos del ERP, copiaba las tablas necesarias (las de ventas y catálogos de productos) y las pegaba o insertaba en la base de datos de la aplicación. El flujo fue hecho en conjunto con mi compañero y quedo como se muestra en la siguiente imagen.

Diagram

Description automatically generated

Ilustración - Proceso de extracción de ventas

En la ilustración 9 se ve de una manera general el proceso de extracción de información, en realidad fueron cinco tablas las que se copian, debido a que el ERP hace una división mas precisa de las órdenes de venta, pero para simplificar el entendimiento se copian tanto las ordenes de venta que contienen productos, fechas y cantidades como el catálogo de productos que da información mas detallada del producto vendido como variedad, región, Comodity, etc.

### Transformación

Cuando la información ya reside en la base de datos podemos empezar a realizar operaciones sobre ella para transformarla, el objetivo es agrupar las ventas por planes, de forma semanal y por tipo de contrato, los parámetros y grupos son los siguientes:

* Los grupos para crear son:
  + Brócoli Crowns: Productos que son coronas completas de brócoli
  + Lechuga
  + Coliflor
  + Col
  + Apio
  + Zanahorias, 1 plan para cada tamaño de empaque (50 libras, 25 libras, 10 libras) y 2 planes adicionales para las presentaciones trozos y colosal
* Solo considerar productos con origen mexicano
* El inicio de la semana es en Domingo
* Dividir las ventas en tipos de contrato (Open, Fixed y Hi-Low)

A continuación, se muestra una imagen que contiene de manera muy general la estructura de origen y la estructura objetivo para la categoría Actual Sales.

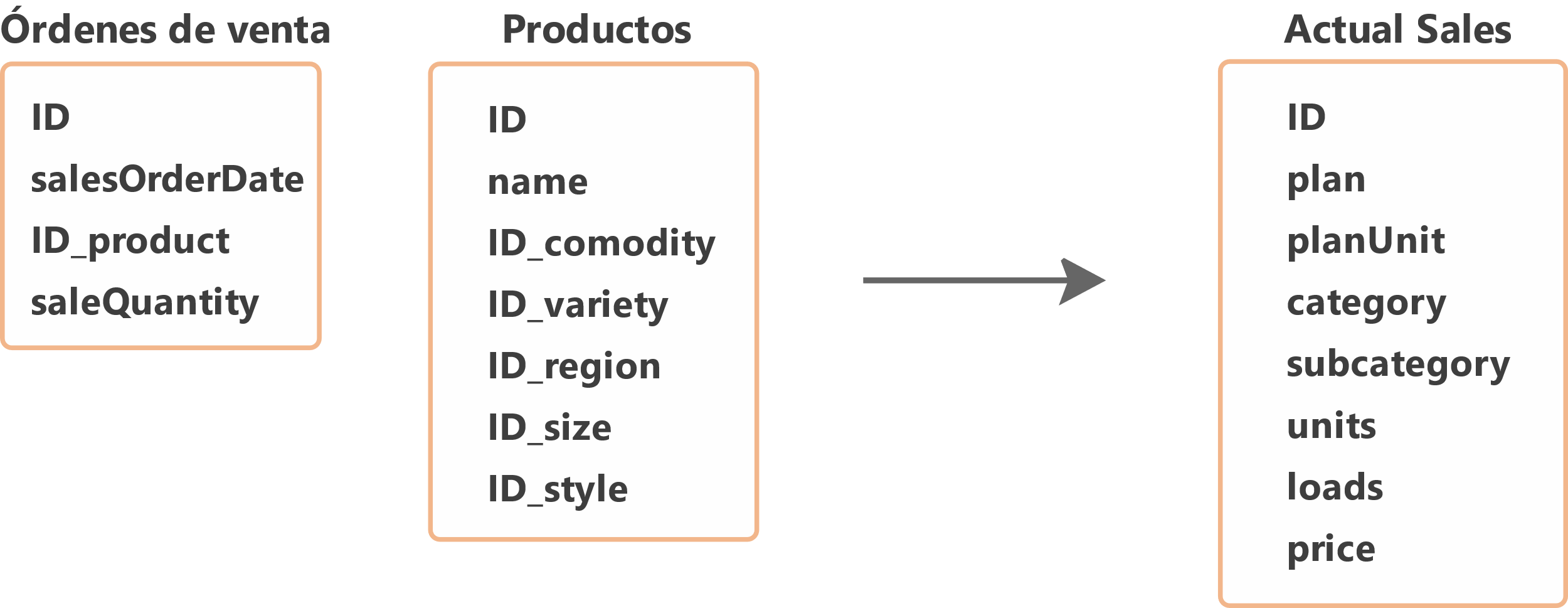


Ilustración - Transformación de Actual Sales

Lo que se hizo para llegar a la estructura de la derecha en la ilustración 10 fue primero que nada fue filtrar la información que no nos sirve, por ejemplo, aquellas ventas con cantidades en 0 o con estatus especiales, lo segundo es separar cada producto vendido y asignarlo dentro de un plan, para ello se creó otro stored procedure que toma información del producto como variedad, región, tamaño y estilo y determina si pertenece o no a un plan, este es el primer paso para determinar si la venta corresponde a los planes de interés, tercero, cuando tenemos ya identificado el plan al que pertenece la venta emepezamos a agrupar por semana la información, se hace un cruce para ver a que unidad del plan corresponde dicha venta, en este caso a que semana corresponde, esta información como se vió en la iteración 1, esta en la tabla de t\_plan\_unit delimitando el inicio y fin de la semana, posterior a eso se establece el tipo de contrato o venta a partir del identificador de la Orden de Venta, este dato es una dato alfanumérico conocido como Sales Order Number y que sigue cierta estructura, entre esta estructura esta una combinación de caracteres que nos dicen el tipo de contrato y que usamos para identificar el tipo de venta (open, fixed o hi-low) , finalmente ya que se tiene identificado el plan, semana y tipo de venta se suman las cantidades en units y loads para obtener la estructura final de Actual Sales.

El primer intento de transformar estos datos consistió en crear vistas para usarlas dentro de la app, sin embargo, era muy complicado realizar los cruces y operaciones además de que era demasiado lento, tardaba alrededor de 4 minutos en obtener y calcular los datos para 1 semana de un plan, la causa principal de este tiempo de respuesta es que se armó una vista intermedia y otra vista final que usaba la vista anterior, como resultado se tenían vistas muy complicadas que realizaban muchos cálculos en tiempo real, adicional a eso cuestioné el uso de vistas ya que la información ni siquiera estaba en tiempo real, era una copia de la base de datos y usar vistas resultaba inadecuado.

Después de solicitar apoyo a mi supervisor y ver las capacidades de Azure Data Factory, me decidí por hacer las transformaciones a los datos por medio de Stored Procedures, a diferencia de las vistas, los Stored Procedures permiten crear código estructurado y secuencial, tienen mas opciones para usar funciones de SQL y dado que la información no requiere ser presentada en tiempo real, estos podían ser ejecutados posterior a la extracción de datos del ERP y usarlos el resto del día. Usar Stores Procedures mejoró el rendimiento de la aplicación, ahora la consulta se genera en unos cuantos segundos (3 o 2 segundos aproximadamente) y la ejecución de estos podía ser disparada desde Azure Data Factoriy, por lo que la extracción y transformación quedaron automatizadas.

### Presentación

Para presentar la información en PowerApps se creó una vista que toma los datos finales de las tablas y algunas consultas anidadas, como la presentación de datos es en bruto no se aplica algún otro proceso mas que el formateo por decimales, millares, símbolo de $, etc.

Como conclusión de esta iteración menciono que aprendí bastante desarrollando esta categoría, por lo que he visto, los ERP o softwares de venta empresariales suelen tener estructuras similares así que la transformación aplicada podría funcionar para otros sistemas basados en ordenes de venta, eso me reconforta un poco ya que ahora se un poco mas sobre la extracción y transformación de datos y puedo aprovechar la experiencia al momento de replicarlo en otros sistemas, también tengo que resaltar toda esta transformación no fue fácil, había mucho desconocimiento de la estructura del ERP y la fuente de documentación era únicamente mi compañero de trabajo y un diagrama muy complejo del modelo de datos, localizar las columnas y tablas necesarias para el calculo de los datos fue algo muy tardado, por consecuencia, también lo fue la validación de los numero finales obtenidos, de hecho, la dificultad fue tal que no pudimos filtrar completamente la información y las cifras mostradas en Actual Sales terminaron con un error promedio del 4-5%, es algo que fue difícil encontrar y discutir con cliente, el ERP tiene muchas tablas / vistas / stored procedures que no conocemos, e incluso a los que no tenemos acceso por derechos de autor, por lo que la experiencia de mi compañero y unas horas de consultoría son lo único que tuvimos para cruzar y filtrar las ventas, el cliente al final fue consiente de eso y acepto el margen de error, con posibilidad de corregirlo en un futuro.

## Iteración 4 – Referent Information

La siguiente categoría contiene la información de apoyo o referencia, dentro de esta categoría se encuentran las subcategorías mencionadas en la Tabla 1, realmente el desarrollo de esta iteración fue fácil debido a que la información ya se tenía procesada con la iteración anterior y únicamente dos de las tres categorías originales quedaron definidas por consultoría.

El calculo de los datos fue simple, de las tablas generadas en la iteración 3, se tomaron los datos para mostrar la información de las ventas del año pasado y calcular su diferencia (en porcentaje), la única complejidad de esta iteración fue buscar una correspondencia no de semanas sino de días, puesto que la misma el número de semana del año anterior podría no corresponder a las mismas fechas de la semana corriente.

Recapitulando, el desarrollo rápido de esta iteración nos permitió recuperar parte del tiempo atrasado en Actual Sales, adicional a eso la información mostrada en esta categoría resalta algo muy importante desde mi punto de vista, que es la comparación del crecimiento o decremento de las ventas respecto al año pasado, un punto importante a considerar dentro de la empresa. A partir de este punto noté que los colores resaltaban demasiado y que tenían que ser más claros para no perder el enfoque, además, recibimos una petición-sugerencia por parte del cliente para poder seleccionar las unidades de los datos (unidades, cargas o precio-costo), por lo que se añadió un filtro para ver el resultado en diferentes unidades, la aplicación con estas modificaciones permitía ver más información y mostrar números que se podían comparar más fácilmente semana a semana, ya en la aplicación junto con Actual Sales queda como se muestra en la siguiente imagen:

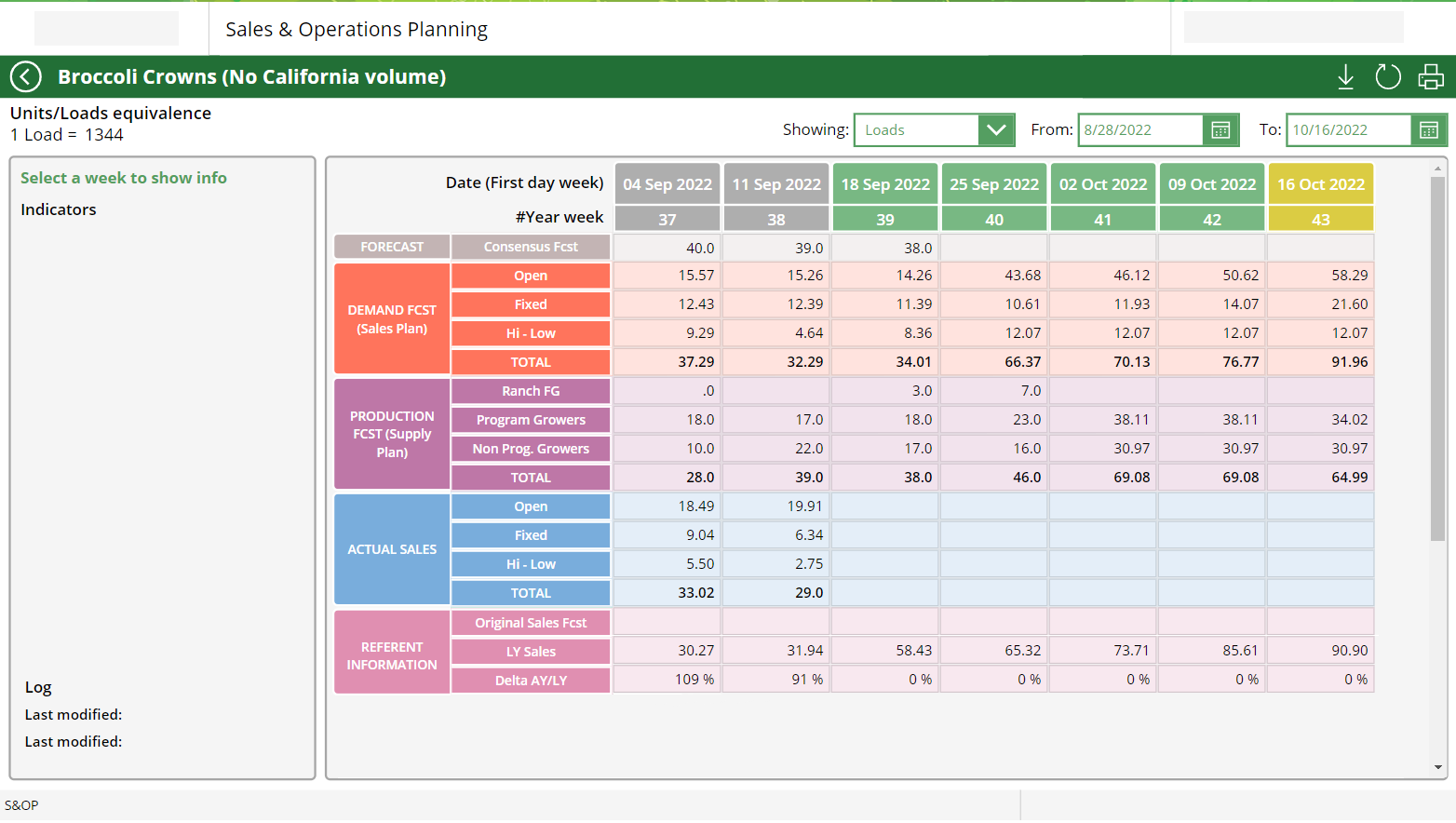


Ilustración - Actual Sales y Referent Information

## Iteración 5 – Stock

Este tema al igual que Actual Sales tuvo etapas de extracción, transformación y presentación de datos, el ERP de la empresa tiene un módulo de inventarios, este módulo registra las entradas, salidas y movimientos de los lotes de productos almacenados, registrando la cantidad disponible de ese lote y la cantidad usada, la ventaja en esta iteración es que ya se había generado un dashboard con información extraída de este módulo y con números validados que permitió agilizar el desarrollo de esta iteración, pues ya había conocimiento de la arquitectura de este módulo.

Para poder extraer los datos se utilizó el mismo servicio y se aplicó el mismo método usado en la iteración 3 – Actual Sales, con Azure Data Factory se extrajeron las tablas originales y por medio de stored procedures se transformaron los datos, en este caso los datos del stock se tomaron de la tabla de inventarios y los datos de las cargas en tránsito del módulo de compras, las transformaciones fueron las siguientes:

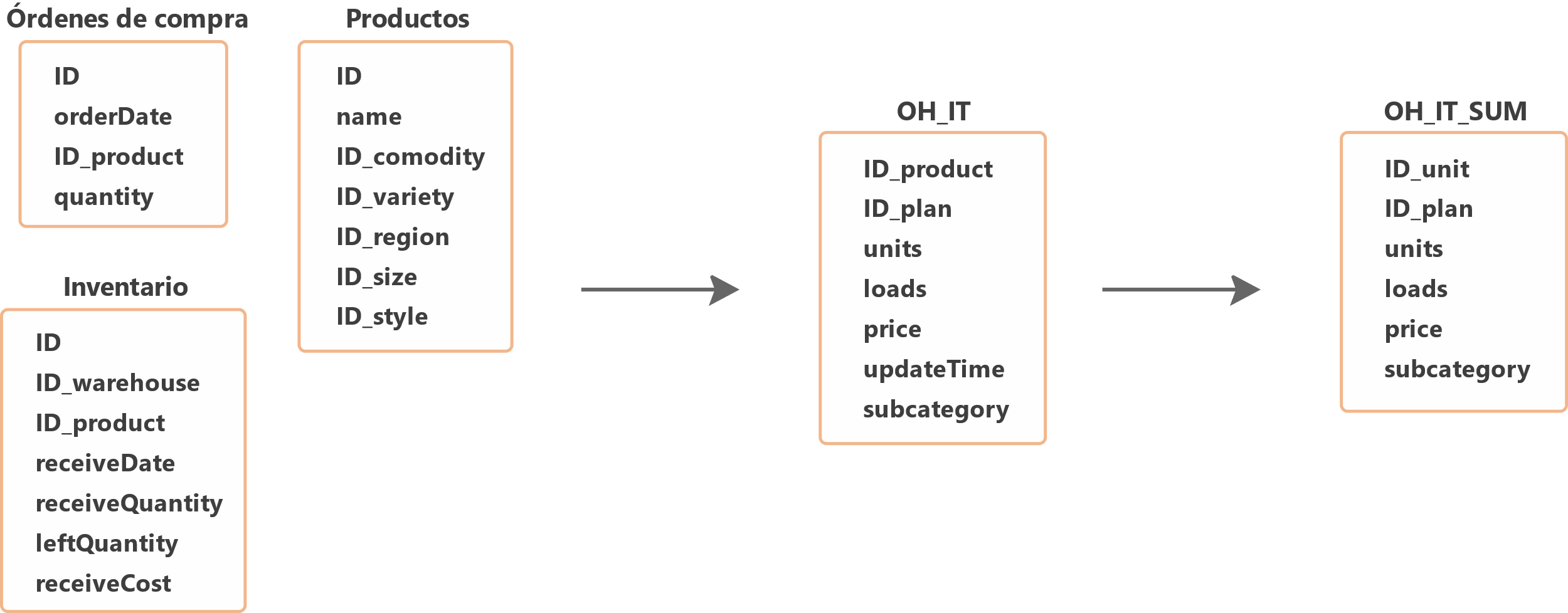


Ilustración - Diagrama transformaciones stock

En la parte izquierda de la ilustración 12 tenemos las tablas fuente, hay una tabla que tiene los cargamentos en tránsito que forma parte del módulo de compras, otra tabla que contiene los datos del inventario, esta tabla contiene todo el historial de entradas de inventario, sin embargo, solo interesan aquellos que aun tienen cantidad disponible para venta, es esta tabla se encuentran datos como la fecha de llegada del lote, la cantidad original recibida, el producto y el almacén en el que se encuentran, finalmente en el lado izquierdo se encuentra la tabla que también hemos usado previamente en la iteración 3 – Actual Sales, por lo que no fue necesario volver a extraer los datos, esta tabla es usada para poder conocer los detalles del producto en tránsito y recibido.

Posterior a extraer las tablas necesarias, se aplicaron transformaciones a la información y se depositaron en la tabla OH\_IT, entre las operaciones realizadas están la utilización del mismo stored procedure en la iteración 3 para determinar el plan al que pertenece el producto, se aplicó la conversión de loads-units, se agregó una columna para distinguir cuales eran datos de cargamentos en tránsito e inventario y se colocó una marca de tiempo para registrar la fecha en la que se extrajeron los datos, esta marca es necesaria debido a que los datos contenidos tanto en inventario como en compras son muy dinámicos, es lógico al entender que son datos que se modifican con cada venta y compra realizada por la empresa, cada que hay una venta se descuenta inventario y cada que hay compra a productores se agregan nuevos lotes y se quitan los que ya llegaron, por lo tanto, no se guardan historiales de estatus de inventario, lo que está en las tablas, es lo que esta en tiempo real, nosotros, al extraer y guardar los datos estamos guardando “capturas” de las cantidades en ese instante, datos que nos permitirán entonces consultar el historial de inventario y mercancía en tránsito. Como resultado, en la tabla OH\_IT quedan guardados los productos de inventario y envíos que había en el instante en el que se ejecutan los procesos de extracción (diariamente a las 6:00 am), indicando en cada renglón a que plan, fecha y subcategoría pertenecen. Finalmente se hace una agrupación de los datos por planes y fechas, se suman las cantidades y se hace una asociación con la semana o unidad del plan correspondiente, manteniendo la subcategoría para poder identificar y armar la vista final.

La parte más interesante de esta iteración reside en el hecho de que ahora se cuenta con un historial de inventario y envío, por decisión del consultor y el cliente, los datos mostrados en stock son los promedios de las cantidades semanales, este dato podría ser útil en algún futuro para analizar la variación de inventarios en temporadas o tal vez el crecimiento de los envíos.

## Iteración 6 – Actual Supply

La penúltima iteración, que también era la última que formaba del bloque principal de categorías, es Actual Supply, dicha categoría contiene los datos de las compras u órdenes de compras por parte de la empresa, en un inicio, se buscaba crear subcategorías de esta información dividiendo el total entre el tipo de compra, había compras que formaban parte de las propias granjas de la empresa, había otras que se realizaban a granjeros con los que se tenían ciertos acuerdos o contratos y otras compras que eran ya sea a granjeros externos o compras de emergencia a mercados nacionales o del estado de California, lamentablemente buscando en como dividir esta información nos encontramos con que el proceso era muy complicado, no se tenían bien definido el tipo de compra y el proceso era realizado de forma muy manual, por lo que junto con consultoría y el cliente se acordó empezar y reducir el alcance a solo mostrar el total de las compras, sin una subdivisión.

Al igual que las iteraciones anteriores, se tiene que extraer información de manera diaria del módulo de inventarios, solo que, a diferencia del anterior, los lotes que están en cero no los excluimos, pues es de nuestro interés conocer todo lo que llegó a los almacenes independientemente si ya fueron vendidos en su totalidad o no, la información se extrajo y se transformó como se muestra en la siguiente imagen (de manera resumida):

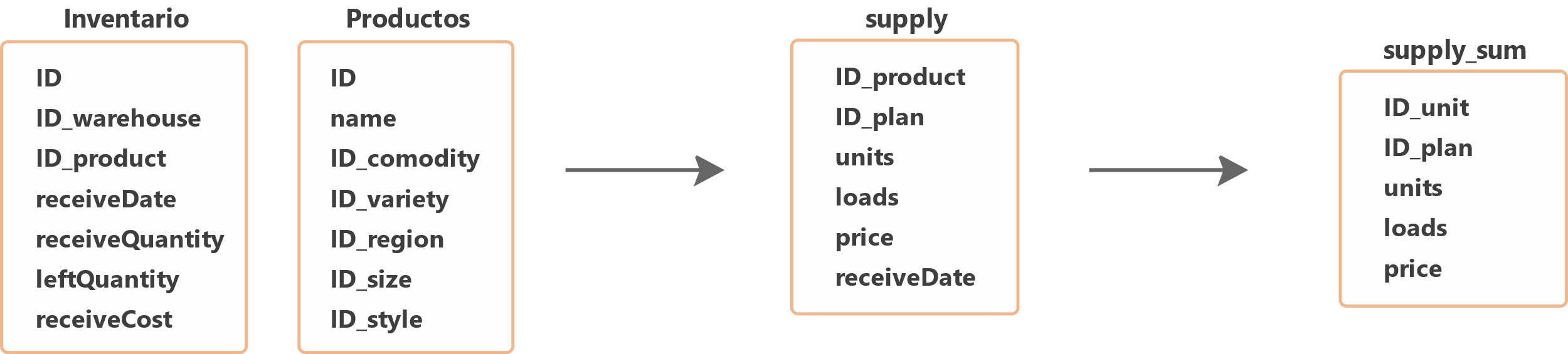


Ilustración - Extracción y transformación de Supply

De igual manera que en Stock, la información de inventario se extrae y se agrupa por días y productos, se les asigna un plan dependiendo del producto y posteriormente se vuelve a agrupar por semana, el resultado final es puesto en una vista y usado en la aplicación. Esta iteración fue corta, lo que nos permitió recuperar parte del atraso que llevábamos con el desarrollo.

A partir de este punto, la aplicación ya contenía información suficiente para considerarse un MVP o mínimo producto viable, extraía automáticamente la información y guardaba el forecast, lo restante era desarrollar la sección de indicadores y una pantalla de ajustes. Con el MVP listo y las pruebas pertinentes, se empujó a que el usuario empezara a usar la aplicación y fuera comentando su uso, en general no hubo mucha retroalimentación, pues la funcionalidad era muy básica respecto al guardado de información, donde hubo mas dudas fue acerca del proceso de importación, al parecer no les quedó muy claro el funcionamiento de la plantilla, pero con una capacitación se resolvió.

## Iteración 7 – Indicadores y ajustes

Con la iteración 6 terminada y el MVP en producción con 2 usuarios se empezó a desarrollar la parte de indicadores, PowerApps tiene una función para realizar cambios en la aplicación, mantenerlos y liberarlos hasta que se use dicha función, cuando un usuario abre la aplicación en modo vista, ve la última versión liberada, cuando la abre en modo edición ve la última versión guardada, así, PowerApps permite trabajar en la aplicación mientras se mantiene la ultima versión liberada en productivo, aprovechando esta ventaja se empezó a trabajar en los indicadores junto con el cliente, estos debían contener la siguiente información:

### On Time InFull US y MX:

Este par de indicadores corresponden a el porcentaje de viajes que llegaron a cliente a tiempo y con la cantidad correcta, el de US se refiere los viajes del almacén a los clientes (US) y el de MX a los viajes de las empacadoras al almacén, la información de los viajes es provista por una empresa contratada por el cliente para monitorear sus viajes. Para el desarrollo de esta parte recurrí a mi compañero de trabajo que ya tenía los valores calculados, pues había trabajado un dashboard que contenía esta información, en realidad, lo que se hizo fue copiar la tabla que él había extraído y transformado y yo me encargué de agrupar los datos por semana. El valor de estos indicadores es independiente del plan, en un inicio se buscó que también correspondieran, es decir, que discriminara entre los viajes de brócoli, zanahoria, etc, sin embargo, los viajes a veces contenían más de un producto o eran más de 2 cargas (camiones), detalle que no se especificaba en la tabla fuente, por lo que se decidió que fuera un indicador global.

### Sales deviation y production deviation

Sales Deviation es la variación en porcentaje entre el Consensus Forecast y Actual Sales, Production Deviation es la desviación entre Consensus Forecast y Actual Supply, este par de indicadores nos dicen cuanto nos desviamos en ventas y producción de lo acordado, esto solo puede ser consultado una vez que el día haya terminado, pues Actual Sales y Actual Supply son datos pasados.

El desarrollo de estos dos indicadores fue sencillo, solo se dividieron las cantidades y se multiplicaron para sacar porcentajes, la información de Actual Sales ya se tenía calculada en las tablas/vistas pasadas y el Consensus Forecast almacenado en la misma base de datos, por lo que no fue problema calcularlo.

### Week min freshness

El porcentaje de producto fresco (el mínimo de la semana), este indicador nos dice cuanto del producto en stock (en almacén) estaba fresco al momento de extraer la información del stock, a diferencia del indicador On Time, este indicador si hace una discriminación por plan. El primer paso a considerar en este desarrollo fue el conjunto reglas para determinar cuándo se consideraba fresco un producto y cuando no, este es valor es binario y se determina con la diferencia entre la fecha de llegada y la fecha actual, los días máximos en los que un producto podía considerarse fresco variaban en función de cada Comodity, las reglas y el calculo de este indicador fue también simple, pues teníamos los datos ya del stock, al principio se considero mostrar el promedio de la semana, sin embargo, con retroalimentación del cliente vimos que era mas útil mostrar el mínimo por temas de calidad.

### Orders returned

El último indicador fue aquel que daba información de las órdenes de venta que fueron devueltas por el cliente (el cliente de la empresa), en un inicio se buscaba el porcentaje de ordenes explícitamente rechazadas por cliente, es decir aquellas con cantidades, producto o calidad indebidas, sin embargo, no se contaba con algún registro en ventas específico para diferenciar entre estos casos, así que este indicador también incluye ordenes de venta que fueron regresadas por ajustes, errores o algún otro motivo, no es precisamente lo que se buscaba en un inicio, pero es algo que se acerca a ello y que se podía usar por el momento.

Para poder diferenciar aquellas órdenes de venta que fueron devueltas se usó el código de la orden (conocido como Sales Order Number), este código es un conjunto alfanumérico que contiene el número de orden, el tipo de contrato, el estatus, entre otras cosas; No era un código estrictamente que seguía una nomenclatura especifica u organizada completamente, pero era suficiente para poder identificar las órdenes devueltas.

El procedimiento para construir el indicador fue primero, identificar los caracteres que son usados para marcar una orden devuelta, estos fueron provistos por consultoría, en segundo, distinguir aquellos productos vendidos que correspondían al plan y que fueron de ordenes devueltas, después, agrupar por semana y sumar las cantidades vendidas que satisfacen las condiciones anteriores, así, tenemos la suma de que productos que fueron devueltos correspondientes a cada semana y plan, finalmente colocamos la información en la vista que contiene los indicadores y la mostramos dentro de la aplicación quedando de la siguiente forma:



Ilustración - Indicadores S&OP

El desarrollo de los indicadores fue rápido, varios de los datos ya los teníamos en nuestra base de datos como resultado de extracciones anteriores y otros como resultado de la propia captura de información en la aplicación, para darle un mejor formato y que se viera más presentable se agruparon por colores (ilustración 14) y se pusieron a petición del consultor, tanto en la parte inferior como en la parte izquierda, estos números nos reflejan datos importantes a considerar durante las reuniones que tienen para fijar sus números, además nos permiten dar seguimiento a los pronósticos y calcular desviaciones. Hay un código de colores que se usa y que fue más difícil de lo que parece, pues hubo que considerar varias condiciones, si el valor del indicador reflejaba un buen desempeño entre más alto o bajo era, si era un porcentaje y si entraba dentro de unos rangos predifinidos como bueno (verde), malo (rojo) o regular (amarillo). Finalmente, estos y otros parámetros se colocaron en una sección dentro de la aplicación que se describe a continuación.

### Settings

Ya con la información validada y el despliegue de la aplicación a unos pocos usuarios, lo siguiente era desarrollar en la aplicación una sección de ajustes, en esta parte se colocarían los parámetros de cada plan, de los kpi’s o indicadores y los respectivos permisos para cada usuario.

#### KPI’s

A pesar de que los indicadores dentro de la aplicación son calculados en vistas, se usa una tabla que nos sirve como catálogo de los mismos, en la iteración 7 se crearon columnas adicionales en esa misma tabla para parametrizar algunas funciones de la aplicación, como los colores, formatos, etc, entre estas columnas se encuentras los rangos para los cuales un indicador se considera aceptable, malo o bueno, el nombre de los responsables, el orden, su descripción y la bandera que se usa para ver si es mejor o peor valor según la magnitud del indicador; todos estos parámetros mencionados fueron puestos en la aplicación para que pudieran ser editados

#### Planes

A pesar de que las semanas son fijas y los parámetros para distinguir entre planes también, hay ciertos parámetros que pueden ser editables en cada plan:

* Short/Mid Term Horizon Days: Cantidad de días a futuro considerados como corto y mediano plazo respectivamente, estos días aunque no son cruciales para el funcionamiento de la aplicación nos permiten colorear y por consecuencia identificar que periodo de tiempo deben considerar en la aplicación, por ejemplo, se definió que para todos los planes se considerara un corto plazo equivalente a 4 semanas, es decir, su planeación cada semana tiene ver 4 semanas a futuro y establecer nuevos valores o revisar las cantidades previamente proyectadas según sea el caso, para el largo plazo se hace algo muy similar pero es un temas más de previsión como su nombre lo indica, a mediano plazo, en realidad no hay una función en la aplicación que te obligue a rellenar las semanas consideras a corto plazo, como lo mencioné antes, es mas una forma de indicar visualmente periodos de tiempo que deben tomar en cuenta.
* Equivalencias entre unidades (units) y cargas: Este dato también fue usado en iteraciones anteriores cada que se hacían cálculos para sumar cantidades vendidas, en stock o cualquier otra categoría donde se manejaran cantidades de productos, al final vimos que este parámetro no iba a cambiar mucho pero lo dejamos parametrizable en caso de que hubiera modificaciones en cuanto a la presentación o empaque de algunos productos, este dato no es algo que debería poder cambiar cualquiera, pues tiene impacto directo en los números y cálculos de la aplicación, es por esto que a continuación se presenta la sección de permisos.

#### Usuarios y permisos:

No todo el mundo debería poder acceder o modificar los datos y parámetros de la aplicación, fue necesario controlar el acceso a ciertas partes de la aplicación, respecto a este tema de seguridad y permisos sigo sin conocer mucho, pero me encontrado con una forma “sencilla” de lidiar con este tema, que podría ser tan extenso como uno quisiera.

Revisando los alcances de la aplicación (los primeros propuestos por el consultor), no hay una necesidad explicita de controlar el acceso, sin embargo, esta es la prueba de que muchas veces este tema queda implícito, pocas han sido las veces en que los permisos o controles de acceso forman parte del levantamiento de requerimientos, el usuario suele omitir dar este requerimiento asumiendo que como es una aplicación, es fácil controlar el acceso a ciertas partes de ella, y no es así, en esta ocasión tuvimos que preguntar y definir después de todas las iteraciones, el control de acceso, (una práctica que no recomiendo y de la cual ya he aprendido), se nos dijo que no todas los usuarios debían poder editar toda la información, los usuarios pertenecientes a ventas debían únicamente poder editar la información de ventas, es decir, el forecast de ventas, caso homologo, los usuarios de producción únicamente debían poder editar el forecas de producción, respecto al forecast consensuado entre las dos partes, solo ciertas personas, incluido el director de ventas y el operador responsable de la aplicación debían poder editar este dato. Algo muy parecido eran algunas otras secciones o permisos como el importar datos de la plantilla o cambiar los parámetros.

A partir del análisis de estos requerimientos, decidí implementar un sistema de seguridad basado en acceso a funciones, es decir, definir las acciones, pantallas o funciones de la aplicación que requerían un control de acceso y especificar para cada usuario si tenía acceso cada uno de estos elementos, la forma de implementarlo fue la siguiente, primero se listan todos los elementos que requieren un control, segundo, crear una tabla donde se cree una columna de tipo booleano o bit para cada uno de los elementos de esa lista y tercero, adicionar la columna que nos permita identificar al usuario, en este caso, se utilizó la propia cuenta de Microsoft del usuario, el correo para ser más específicos.

Finalmente se tenía una matriz donde las columnas representaban los módulos, o funciones de la aplicación y los renglones los permisos que tenía cada usuario de la aplicación, las propias características de PowerApps nos facilitaron que se desarrollara de esta manera, pues ya Microsoft cuenta con varios tipos de autenticaciones y elementos para identificar a un usuario dentro de la app.

Todas estas columnas y sus valores fueron puestas dentro de la aplicación para que un administrador lo pudiera consultar y editar, otorgando así permisos a quien lo requiera, por default cada que se quiera compartir la aplicación, se debe hacer por el mecanismo tradicional de PowerApps, y además, también compartirla por medio de la misma aplicación, teniendo así 2 capas de seguridad, una provista por el propio Microsoft, y otra creada por nosotros para controlar la aplicación de forma interna, el resultado en la aplicación se muestra a continuación:

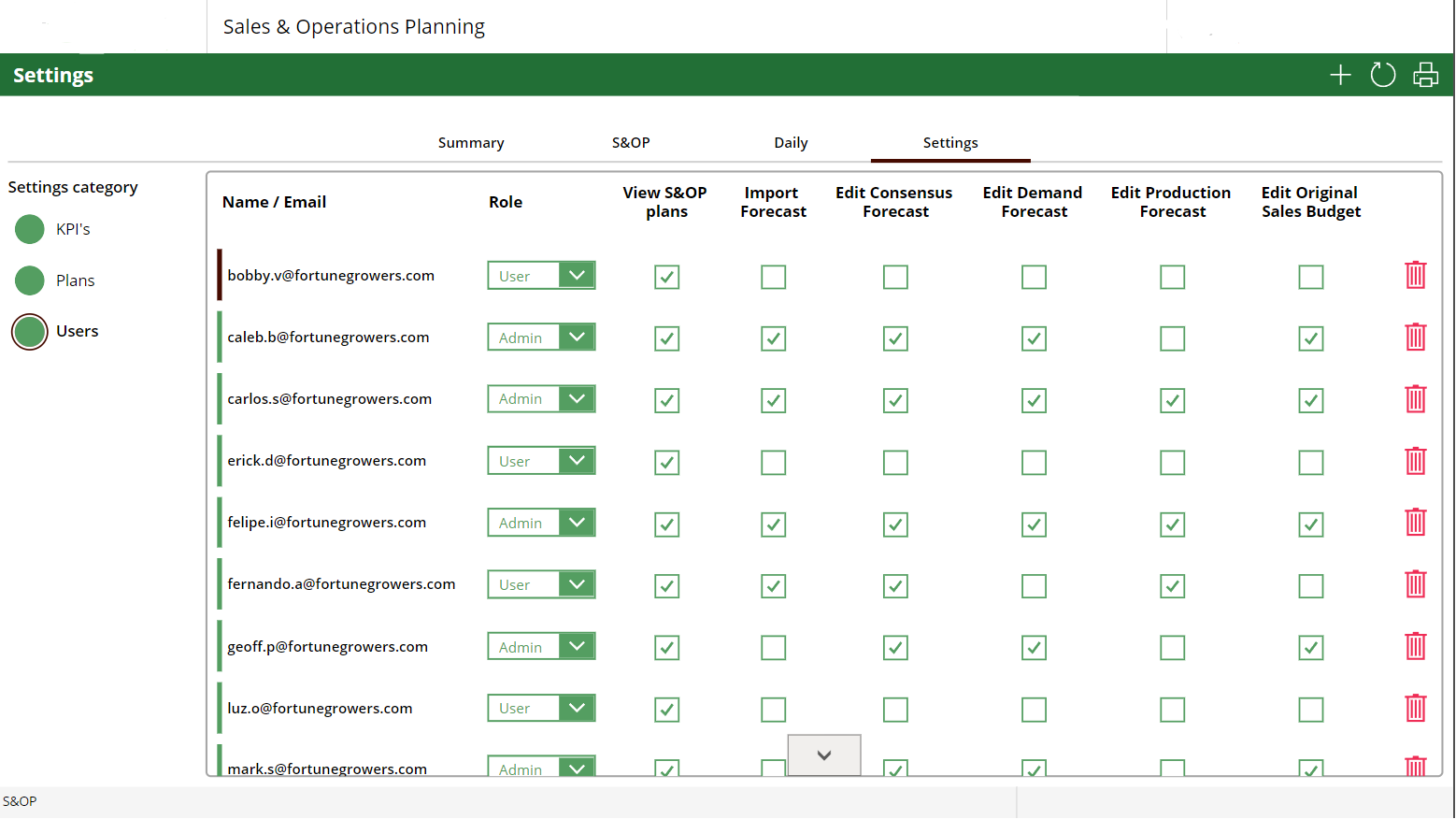


Ilustración – Settings

Hay una columna no mencionada que se muestra en la ilustración 15, que es el rol, esta columna se uso para determinar que personas pueden acceder a esta sección de settings, solo los usuarios administradores pueden acceder a esta pantalla y realizar ajustes en los permisos de otros.

# Resultados obtenidos

Asd

# Conclusiones

1. (Cox, 2002) [↑](#footnote-ref-1)
2. (Antonio Márcio Tavares Thomé, 2012) [↑](#footnote-ref-2)
3. (Dreyer, Kiil, Dukovska-Popovska, & Kaipia, 2018) [↑](#footnote-ref-3)
4. (Mentzer, 2005) [↑](#footnote-ref-4)
5. (Cambridge Dictionary, 2022) [↑](#footnote-ref-5)
6. (APICS Dictionary, 2013) [↑](#footnote-ref-6)
7. (Cambridge Dictionary, Sales, 2022) [↑](#footnote-ref-7)
8. (Microsoft, 2022) [↑](#footnote-ref-8)
9. (Microsoft, 2022) [↑](#footnote-ref-9)
10. (Microsoft, 2022) [↑](#footnote-ref-10)
11. (Microsoft, 2022) [↑](#footnote-ref-11)
12. (Microsoft, 2022) [↑](#footnote-ref-12)
13. (Rafael Camps Paré, 2005) [↑](#footnote-ref-13)
14. (Rafael Camps Paré, 2005) [↑](#footnote-ref-14)
15. (Microsoft, Microsoft Docs, 2022) [↑](#footnote-ref-15)
16. (Sommerville, 2011) [↑](#footnote-ref-16)
17. (Antonio Márcio Tavares Thomé, 2012) [↑](#footnote-ref-17)
18. [↑](#footnote-ref-18)