



Conector nativo ecommerce Shopify- Vtex : Diseño, desarrollo e implementación de una solución backend que facilite la conexión nativa entre VTEX y Shopify, empleando Express.js como tecnología de desarrollo.

Yenifer Daniela Osorio Gómez

Informe de práctica para optar el título de Ingeniera de Sistemas otorgado por UdeA

Asesor

Jeysson Pérez Gómez. Especialista en Gerencia Integral

Universidad de Antioquia

Facultad de Ingeniería

Ingeniería de Sistemas

Medellín

2024

Cita	(Osorio Gómez, 2024)
Referencia	Osorio Gómez Y. D. (2024). <i>Conector nativo ecommerce Shopify- Vtex : Diseño, desarrollo e implementación de una solución backend que facilite la conexión nativa entre VTEX y Shopify, empleando Express.js como tecnología de desarrollo</i> [Semestre de Industria]. Universidad de Antioquia, Medellín.
Estilo APA 7 (2020)	



Centro de Documentación Ingeniería (CENDOI)

Repositorio Institucional: <http://bibliotecadigital.udea.edu.co>

Universidad de Antioquia - www.udea.edu.co

Rector: John Jairo Arboleda Céspedes

Decano/Director: Jesús Francisco Vargas Bonilla

Jefe departamento: Diego José Luis Botía Valderrama

El contenido de esta obra corresponde al derecho de expresión de los autores y no compromete el pensamiento institucional de la Universidad de Antioquia ni desata su responsabilidad frente a terceros. Los autores asumen la responsabilidad por los derechos de autor y conexos.

Dedicatoria

Esta dedicatoria va para mi familia, en especial para mis padres y mi hermana por ser mi principal apoyo y motor durante todos estos años de esfuerzos. Y también para mi prima y su esposo que me brindaron la oportunidad de tener un trabajo, dándome toda la comodidad en los horarios para que yo pudiera continuar con mis estudios.

Agradecimientos

Agradecimiento especial a Tita Media por brindarme la oportunidad de realizar mis prácticas profesionales allí. A su vez a Wilmer Cardozo y al equipo de trabajo encargado de lo que ha sido este proyecto y a Jeisson Perez por el acompañamiento durante este proceso. Y por último, agradezco inmensamente a la Universidad de Antioquia por haberme acogido y educado profesionalmente durante todos estos años.

Tabla de contenido

Resumen	7
Abstract	8
Introducción	9
1 Objetivos	10
1.1 Objetivo general	10
1.2 Objetivos específicos	10
2 Marco teórico	11
3 Metodología	13
4 Resultados	16
5 Análisis.3	32
6 Conclusiones	33
7 Recomendaciones	34
Referencias	34

Lista de figuras

Figura 1	
<i>Programas a seguir a partir de la metodología Scrum</i>	14
Figura 2	
<i>Configuración del seller en ambiente VTEX</i>	16
Figura 3	
<i>Configuración de los webhooks en Shopify</i>	17
Figura 4	
<i>Arquitectura conector</i>	18
Figura 5	
<i>Flujo notificación de catálogo</i>	19
Figura 6	
<i>Flujo creación de SKU</i>	20
Figura 7	
<i>Flujo para la actualización de SKU</i>	21
Figura 8	
<i>Flujo de creación de órdenes</i>	22
Figura 9	
<i>Flujo de preparación de órdenes</i>	23
Figura 10	
<i>Flujo de facturación de órdenes</i>	24
Figura 11	
<i>Flujo de tracking de órdenes</i>	25
Figura 12	
<i>Flujo de cancelación desde el seller</i>	26
Figura 13	
<i>Flujo de cancelación desde marketplace</i>	27
Figura 14	
<i>Vista desde el módulo tiendas</i>	29
Figura 15	
<i>Vista desde el módulo productos</i>	30
Figura 16	
<i>Vista desde el módulo configuración para credenciales</i>	30
Figura 17	
<i>Vista desde el módulo configuración para envíos</i>	31

Siglas, acrónimos y abreviaturas

API	Application Programming Interface
AWS	Amazon Web Services
HTTP	Hypertext Transfer Protocol
MVP	Minimum Viable Product
SKU	Stock-keeping unit

Resumen

El proyecto se enfocó en abordar el desafío de integrar de manera fluida las plataformas de comercio electrónico VTEX y Shopify, dos líderes en el mercado, para permitir a diferentes comercios aprovechar lo mejor de ambas sin enfrentar limitaciones técnicas. Lo anterior a través de una solución backend utilizando las APIs de ambas plataformas. Con este conector se busca facilitar a los comercios la publicación de productos desde Shopify en el marketplace de VTEX, así como la gestión eficiente de inventario, precios, ventas y envíos. Esta integración busca ampliar oportunidades de expansión, garantizar una experiencia coherente para los clientes en diversos canales y maximizar las ventas.

Palabras clave: conector, Vtex, Shopify, marketplace, seller, e-commerce

Abstract

The project focused on addressing the challenge of seamlessly integrating the e-commerce platforms VTEX and Shopify, two market leaders, to allow different merchants to take advantage of the best of both without facing technical limitations. The above through a backend solution using the APIs of both platforms. This connector seeks to make it easier for businesses to publish products from Shopify in the VTEX marketplace, as well as the efficient management of inventory, prices, sales and shipping. This integration seeks to expand expansion opportunities, ensure a consistent experience for customers across multiple channels, and maximize sales.

Keywords: connector, Vtex, Shopify, marketplace, seller, e-commerce

Introducción

En el mundo actual en donde el internet toma cada vez más protagonismo, el comercio se está viendo obligado a migrar o expandirse de un enfoque tradicional de relacionamiento directo con el cliente a algo más impersonal y rápido, como lo son las compras por internet. Es en este punto que aparece el comercio electrónico y las diferentes plataformas que facilitan la migración o la expansión de los comercios. En este contexto, plataformas de comercio electrónico como VTEX, el cual está expandiéndose fuera de Latinoamérica y Shopify con millones de comercios que lo utilizan alrededor del mundo, han demostrado ser líderes indiscutibles, cada una ofreciendo una gama única de herramientas y características para impulsar la presencia en línea y maximizar las ventas. Sin embargo, a medida que las empresas buscan diversificar sus canales de venta y llegar a diferentes mercados y países, surge la necesidad de una integración fluida entre estas dos plataformas.

Este proyecto se propuso abordar este desafío mediante el desarrollo de un conector nativo entre VTEX y Shopify, que permite a los comercios aprovechar lo mejor de ambas plataformas sin tener que lidiar con obstáculos técnicos o limitaciones operativas. Al crear un conector entre estas dos plataformas líderes, este proyecto tuvo como objetivo abrir nuevas oportunidades para la expansión del negocio, la gestión eficiente de inventario, la sincronización de datos vitales y una experiencia de cliente coherente en todos los canales.

En este proyecto se desarrolló una solución backend que permite la conexión entre Vtex y Shopify. Todo esto a través de las Api ofrecidas por estas dos plataformas. Con esta conexión se permitió a sellers desde Shopify publicar sus productos en el marketplace de Vtex y manejar desde inventarios y precios hasta la venta y envío de dichos productos. Permitiendo de esta manera a estas dos plataformas interactuar juntas, aprovechando las ventajas que cada una de estas tiene y brindando a las empresas las herramientas que necesitan para expandirse de manera eficiente, aumentar sus ventas y ofrecer una experiencia de cliente excepcional en múltiples plataformas. En las siguientes secciones, explicaré los detalles de este proyecto, la metodología de desarrollo y las herramientas usadas para la creación de este conector entre VTEX y Shopify.

1 Objetivos

1.1 Objetivo general

Diseñar, desarrollar e implementar una solución backend que facilite la conexión nativa entre VTEX y Shopify, empleando Express.js como tecnología de desarrollo.

1.2 Objetivos específicos

1. Configurar los entornos de prueba de Vtex y Shopify, incluyendo la adición de productos, categorías y marcas, generación de tokens y permisos necesarios para el consumo de APIs
2. Definir la arquitectura y flujos de procesos que permitan la integración fluida entre VTEX y Shopify, asegurando una conexión eficiente.
3. Desarrollar servicios del conector Vtex-Shopify correspondientes a la creación y actualización de catálogo.
4. Desarrollar servicios del conector Vtex-Shopify correspondiente a la creación, cancelación, envío y tracking de órdenes
5. Implementar la solución como una aplicación en Shopify

2 Marco teórico

En la actualidad, el internet se ha convertido en una herramienta fundamental en la vida cotidiana de las personas, generando un impacto significativo en diversos aspectos de la sociedad. Una de las consecuencias notables de esta creciente digitalización es la necesidad de las empresas de explorar nuevas formas de expandir sus negocios y llegar a un público más amplio. Con el objetivo de simplificar este proceso para las empresas dedicadas al comercio, han surgido plataformas de comercio electrónico líderes como VTEX y Shopify.

VTEX, una multinacional brasileña fundada en 2000, se especializa en soluciones de comercio electrónico. En consonancia con el artículo "VTEX and Nuvei partner amid global expansion drive" (*VTEX and Nuvei partner amid Global Expansion Drive - VTEX, s. f.*), VTEX ha evolucionado para convertirse en una plataforma global de comercio digital y mercado empresarial. Actualmente, cuenta con más de 2.400 clientes y opera en más de 3.200 tiendas en línea, abarcando 38 países. Dado su origen brasileño, VTEX ha logrado una sólida presencia en Latinoamérica.

En comparación, Shopify, una empresa canadiense de comercio electrónico que se lanzó al público en abril de 2006, ha dejado una marca significativa en el panorama del comercio en línea. Con presencia en más de 175 países, generó ingresos superiores a \$5.6 mil millones en 2022 según el informe "23 Essential Shopify Statistics You Need to Know in 2023" (Ltd, 2024)

Dada la presencia selectiva de estas plataformas en diferentes países, surgió la posibilidad de crear una conexión entre VTEX y Shopify, brindando a los negocios una alternativa adicional para su expansión. Por tanto, se propuso el desarrollo de una solución backend que permita una integración fluida y nativa entre VTEX y Shopify.

En esta integración, VTEX desempeña el papel de Marketplace, mientras que Shopify actúa como vendedor (seller). Mediante Shopify, los vendedores pueden publicar sus productos en el Marketplace de VTEX, donde también se podrían llevar a cabo las transacciones de venta.

Es fundamental señalar que el conector, elemento clave de la integración, se encuentra en una infraestructura externa. En este caso, se optó por utilizar una infraestructura AWS que, según su definición en su página principal, “es la nube más adoptada y completa en el mundo, que ofrece más de 200 servicios integrales de centros de datos a nivel global. Millones de clientes, incluso las empresas emergentes que crecen más rápido, las compañías más grandes y los organismos gubernamentales líderes, están usando AWS para reducir los costos, aumentar su agilidad e innovar de forma más rápida” (*What-is-AWS*, s. f.)

La solución backend fue desarrollada con Express.js, un marco de aplicación web altamente flexible y minimalista basado en Node.js (*Express - Node.js Web Application Framework*, s. f.). Express.js proporciona un conjunto robusto de funcionalidades para aplicaciones web y móviles, caracterizándose por su versatilidad y eficiencia. Estas cualidades permiten la construcción ágil de aplicaciones escalables y de alto rendimiento.

En cuanto a la metodología de desarrollo, se adoptó una aproximación basada en el concepto "WOW" (Way of Working), el cual busca que todo equipo de trabajo encuentre su forma de llevar a cabo sus labores, esto, basándose en metodologías existentes, pero no siguiéndola al pie de la letra: “Encontrar nuestra forma de trabajo, nuestro WoW (por sus siglas en inglés Way of Working), va a depender más de un proceso de experimentación que de seguir con estricto apego una metodología” (García, 2020). Por tanto, la forma de trabajo que se adoptó, se inspira en la metodología Scrum. Según la guía Scrum (Sutherland & Schwaber, 2020), Scrum es un marco de trabajo ágil, el cual brinda un enfoque ligero para la generación de valor a través de soluciones adaptables a problemas complejos.

Los elementos fundamentales de Scrum incluyen cinco eventos principales: Sprint, que define un período de tiempo limitado (máximo un mes) para la realización de tareas; Planning,

donde se planifican las historias de usuario a abordar durante el sprint; Daily Stand-up, una reunión diaria que sincroniza las tareas del equipo; Review, en el cual se presenta el trabajo realizado durante el sprint; y Retrospectiva, donde se analizan las posibles mejoras derivadas del sprint.

Además, la metodología Scrum establece tres roles clave: Product Owner, Scrum Master y Developers Team. Cada uno de estos roles desempeña funciones específicas para asegurar un desarrollo fluido y colaborativo.

3 Metodología

Para este proyecto se usó la WOW (Way of Working), basándose un poco en Scrum y siguiendo el siguiente programa:

Figura 1

Programas a seguir a partir de la metodología Scrum



Nota. Esta imagen fue tomada de la documentación de definición de proyecto de Tita Media

Como se puede observar, se llevaron a cabo los siguientes eventos:

Sprint: con una duración de dos semanas cada uno

Planning: iniciando cada sprint y donde se repasa lo que se hará durante las siguientes dos semanas

Seguimiento de líderes cada lunes

Daily: No se hacen todos los días, sino cada semana para hacer seguimiento de avances

Sprint Review: Se muestra a los stakeholders, en este caso VTEX, las iteraciones resultantes del Sprint.

Sprint Retrospectiva: Se habla sobre que inconvenientes se registraron durante el Sprint y cómo se puede mejorar para que no se presenten de nuevo

Además los roles que se manejaron fueron:

Products Owner: Cristian Cifuentes (desde Vtex)

Scrum Master: Juan Camilo Campos, Adriana Lemus (desde TITA MEDIA)

Developers: Wilmer Cardozo(lider backend), Daniela Osorio (Desarrolladora Backend), Luigi Marquez(Desarrollador backend), Giovany Jimenez (Desarrollador Backend)

4 Resultados

Así se llevó a cabo el desarrollo de los diferentes objetivos específicos:

Para lograr el primer objetivo de configurar los entornos de prueba de VTEX y Shopify, incluyendo la adición de productos, categorías y marcas, generación de y tokens y permisos necesarios para el consumo de APIs, se configuraron los diferentes entornos de pruebas: en el ambiente VTEX de prueba se hizo el mapeo de categorías y marcas, se crearon key y token para el consumo de servicios de VTEX y se creó el seller asignando la URL (Fulfillment URL), el cual es la base de la URL que se usará para exponer los endpoints que desde el conector se exponen a VTEX, esto se puede visualizar en la **Figura 2**:

Figura 2

Configuración del seller en ambiente VTEX

← Shopify | Integration Type: external | Store Visibility: Identifiable

Integration

Fulfillment URL*

https://proyectox.titamedia.com:9091/api

Now you can only create an external integration using a fulfillment URL. [Learn more](#)

Hub Settings Copy

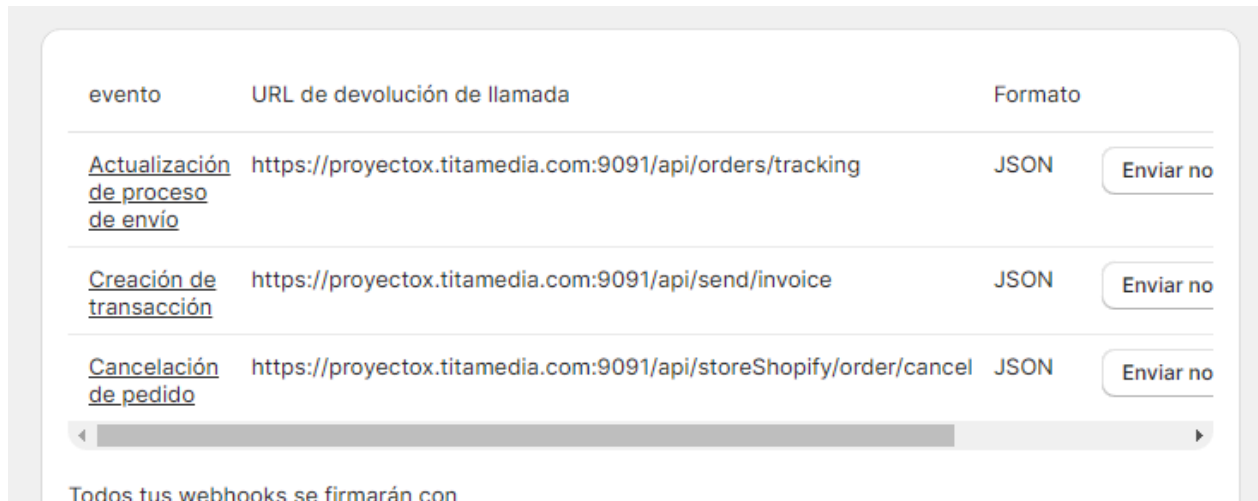
User: [Redacted]

Password: [Redacted]

Y en el ambiente de prueba Shopify. Se creó el token para el consumo de los servicios y se configuraron los webHooks que harán los llamados del lado del seller a los diferentes servicios expuestos por el conector. En la **Figura 3** se puede ver la configuración hecha en Shopify.

Figura 3

Configuración de los webhooks en Shopify



evento	URL de devolución de llamada	Formato	
Actualización de proceso de envío	https://proyectox.titamedia.com:9091/api/orders/tracking	JSON	Enviar no
Creación de transacción	https://proyectox.titamedia.com:9091/api/send/invoice	JSON	Enviar no
Cancelación de pedido	https://proyectox.titamedia.com:9091/api/storeShopify/order/cancel	JSON	Enviar no

Todos tus webhooks se firmarán con

Adicionalmente, se creó y configuró la base de datos en DynamoDB de AWS, para la persistencia del conector. En esta base de datos se crearon 4 tablas:

products_vtex: estarán almacenados los productos de los diferentes sellers y a través de esta se controla que los productos hayan sido enviados al marketplace.

shopify_account: estará almacenada la información de los diferentes sellers (cuentas Shopify).

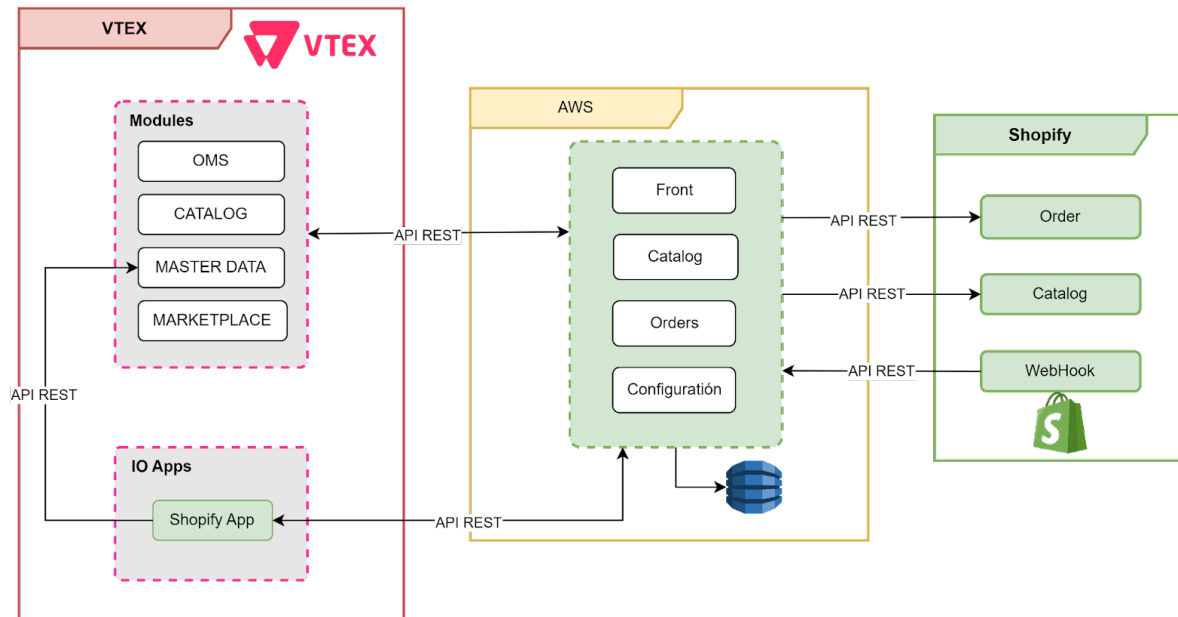
shopify_vtex_orders: estarán almacenadas las diferentes órdenes generadas desde VTEX para el manejo de sus diferentes estados.

store_shopify_vtex: estará almacenada la información de los diferentes marketplace.

Para el segundo objetivo que consiste en definir la arquitectura y flujos de procesos que permitan la integración fluida entre VTEX y Shopify, asegurando una conexión eficiente, se definieron la arquitectura y los diferentes flujos llevados a cabo en el proyecto:

El conector siguió la arquitectura que se muestra en la **Figura 4:**

Figura 4
Arquitectura conector



Esta arquitectura consta de tres componentes principales:

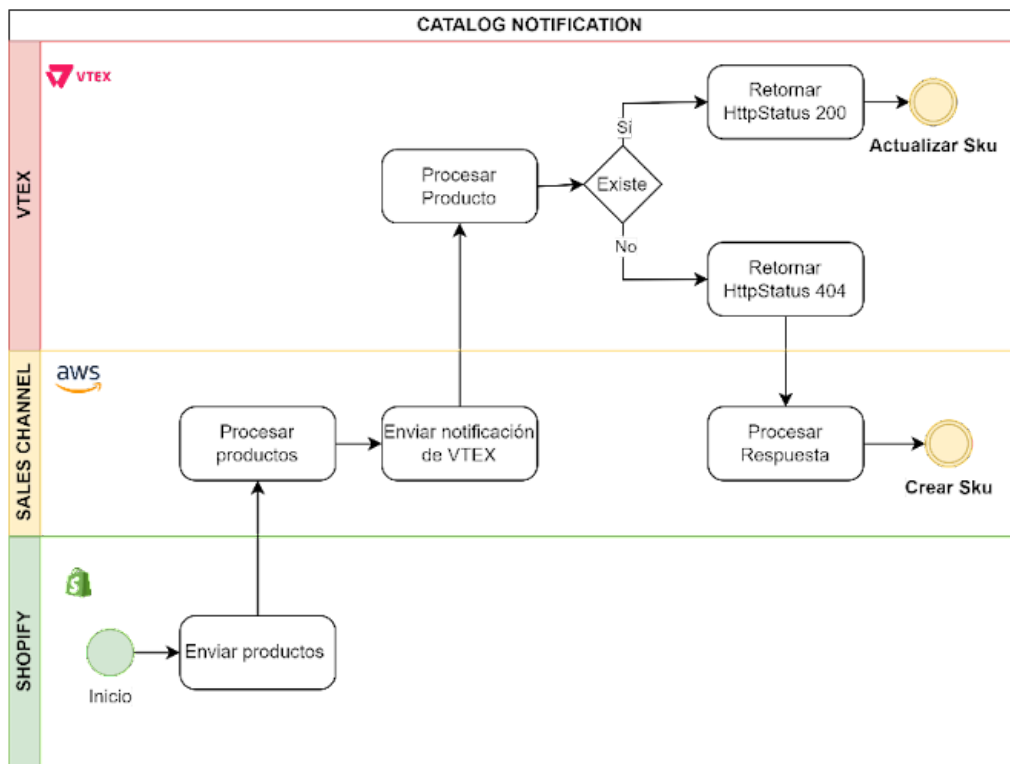
- **VTEX:** este componente contiene lo relacionado con VTEX (OMS, Catalog, Master Data, Marketplace). Además de un aplicativo en VTEX IO en el cual el seller podrá visualizar información como las órdenes creadas y las comisiones correspondientes a las misma.
- **AWS:** este componente contiene lo que estará alojado en la infraestructura AWS (Front, Configuración y servicios para crear los productos y las órdenes), es decir que aquí estará alojado el conector. Además, también estará la base de datos DynamoDB.
- **Shopify:** este componente contiene todo lo referente a Shopify (Order, Catalog, webhook), es decir, todo lo que permitirá tomar y enviar información de shopify.

- **Integración de producto:**

Creación y actualización de productos: este proceso inicia cuando el seller envía un listado de productos, el cual será tomado desde el conector y este enviará una notificación a VTEX. Si esta notificación retorna un código HTTP 200 quiere decir que el producto ya existe y es un producto para actualizar; por otro lado, si esta notificación retorna un

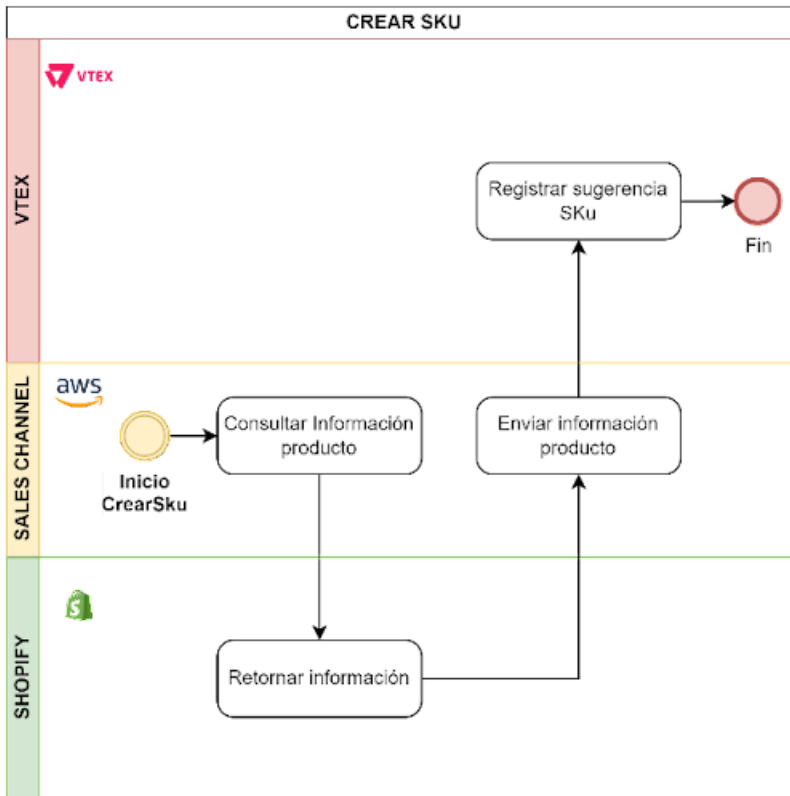
código HTTP 404, esto quiere decir que el producto no existe y es un producto para crear. En la **Figura 5** se puede visualizar el diagrama de flujo de este proceso:

Figura 5
Flujo notificación de catálogo



Creación de SKU: este proceso inicia cuando una notificación de catálogo retorna un código HTTP 404. Desde el conector, se hará a Shopify la petición de la información del producto a crear, procesa la información y la envía a VTEX, para que este haga la sugerencia de SKU (variante que tiene un producto. Por ejemplo, para una blusa un sku puede ser el color azul o rojo). Este flujo se puede ver en la **Figura 6**.

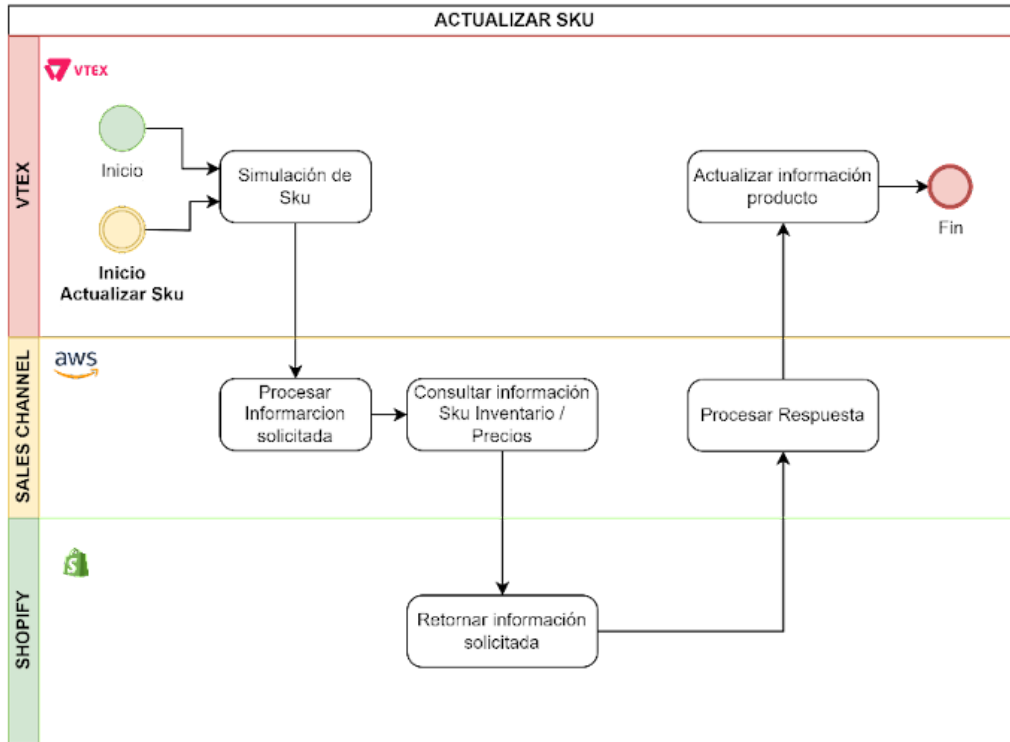
Figura 6
Flujo creación de SKU



Actualización de SKU: este proceso tiene dos puntos de partida: el primero es cuando se ha enviado una notificación desde Shopify y ha respondido un código 200 y la otra cuando VTEX cada tanto pide actualización del SKU. Esta petición llega al conector y este va a hacer la consulta a Shopify de la información del producto para procesarla y mandarla a VTEX y actualizar la información del producto. Tal como se logra visualizar en el diagrama de flujo que muestra la **Figura 7**.

Figura 7

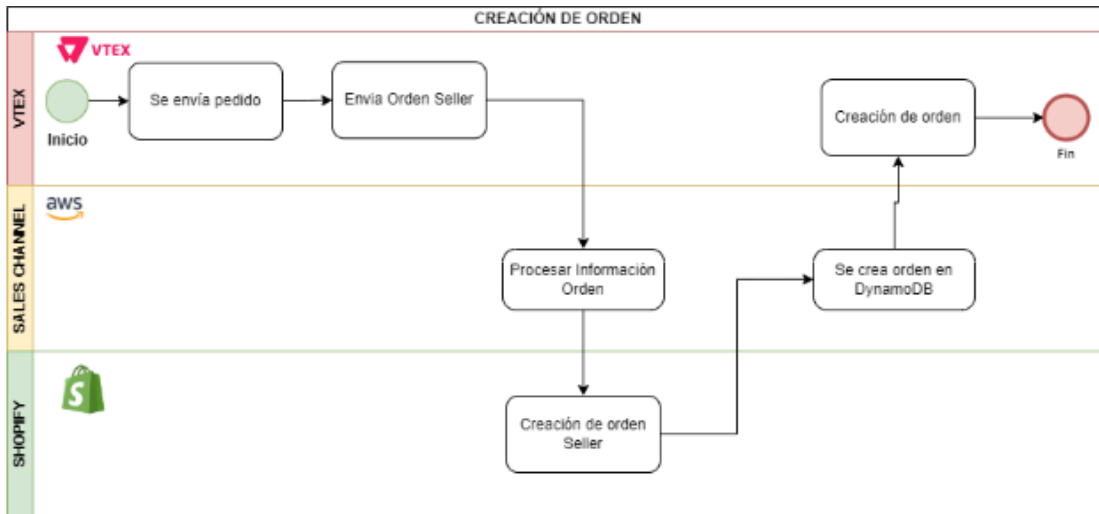
Flujo para la actualización de SKU



- **Integración de órdenes**

Creación de orden: en el conector se expuso un servicio a VTEX el cual permite que VTEX notifique que hay una nueva orden y esta se pueda crear en Shopify. Además será creada también en la base de datos DynamoDB alojada en AWS, para el manejo que se les desee dar a la misma. En la **Figura 8** se logra ver el diagrama de flujo de este proceso.

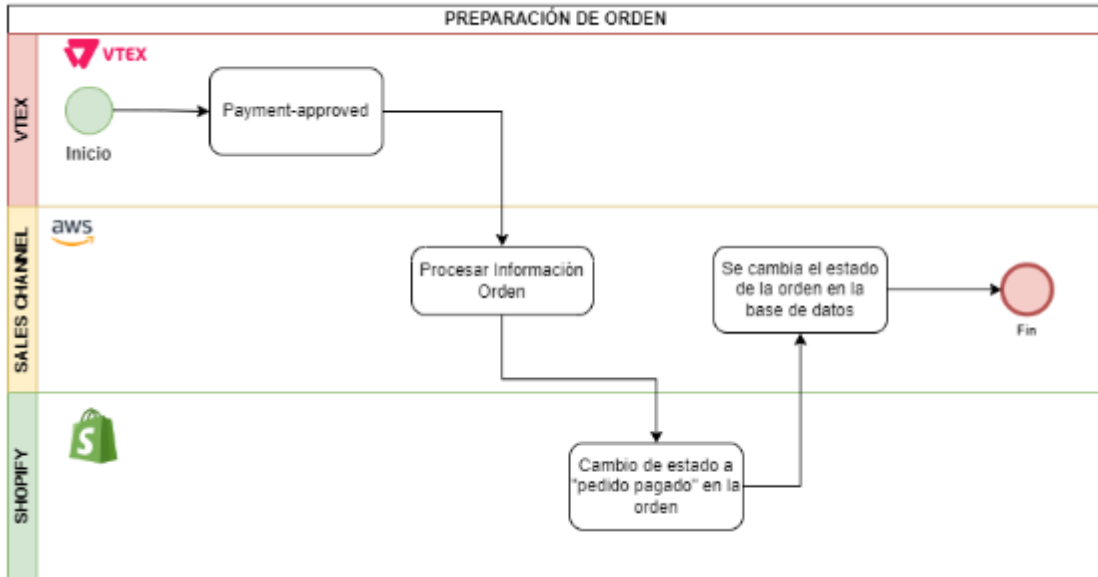
Figura 8
Flujo de creación de órdenes



Preparación de orden: en el conector se expuso un servicio a VTEX el cual invocará cuando se haya aprobado el pago de la orden. Cuando esto suceda, el conector consumirá un servicio de Shopify que aprobará el pago y creará una transacción la cual será usada para la facturación. En la **Figura 9** se logra ver el diagrama de flujo de este proceso.

Figura 9

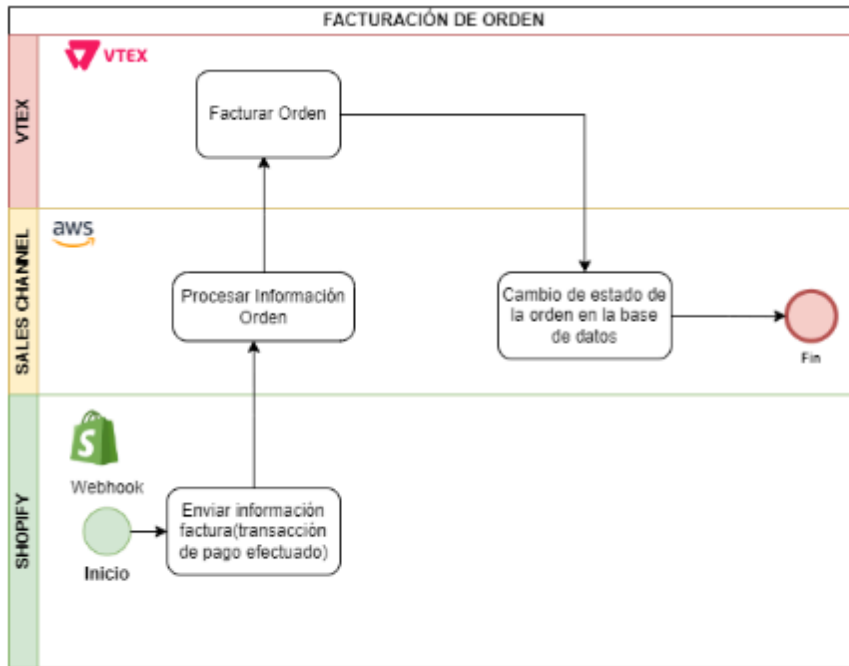
Flujo de preparación de órdenes



Facturación de la orden: cuando se crea la transacción de la realización de pago en Shopify, se ejecutará un webhook de transacción creada en Shopify, el cual hará el llamado a un servicio expuesto en el conector, el cual consumirá el endpoint de VTEX, enviando el número de factura (transacción) para la creación de la factura, para que VTEX facture la orden. En la **Figura 10** se logra ver el diagrama de flujo de este proceso.

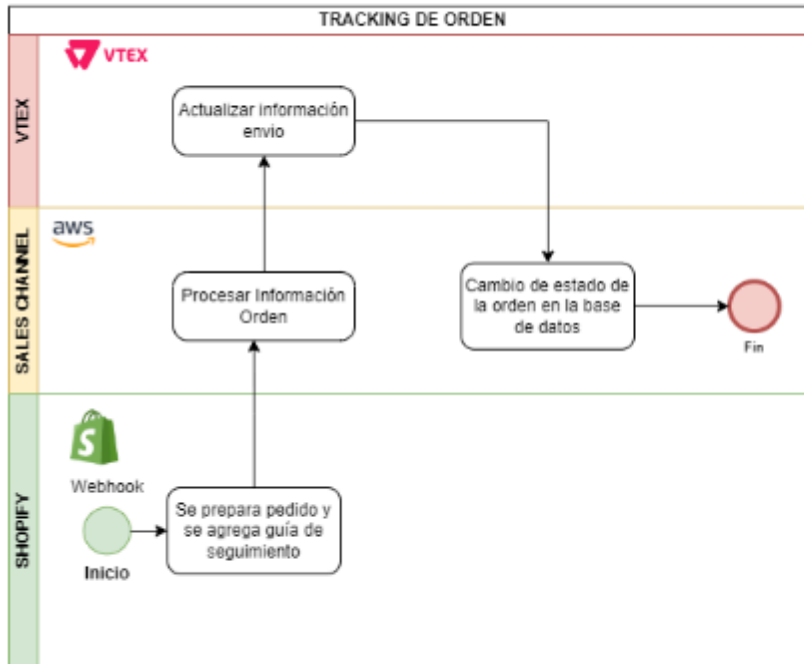
Figura 10

Flujo de facturación de órdenes



Tracking de la orden: una vez facturada la orden en VTEX, se puede preparar y generar guía de seguimiento en Shopify para el envío. Al ya haber asignado esta información desde Shopify se ejecutará un webhook en Shopify de cambio en estado del envío. Este webhook hará el llamado al servicio expuesto por el conector que buscará la orden y le asignará la información correspondiente al número de guía y a la url de seguimiento. Este proceso puede ser observado en la **Figura 11**.

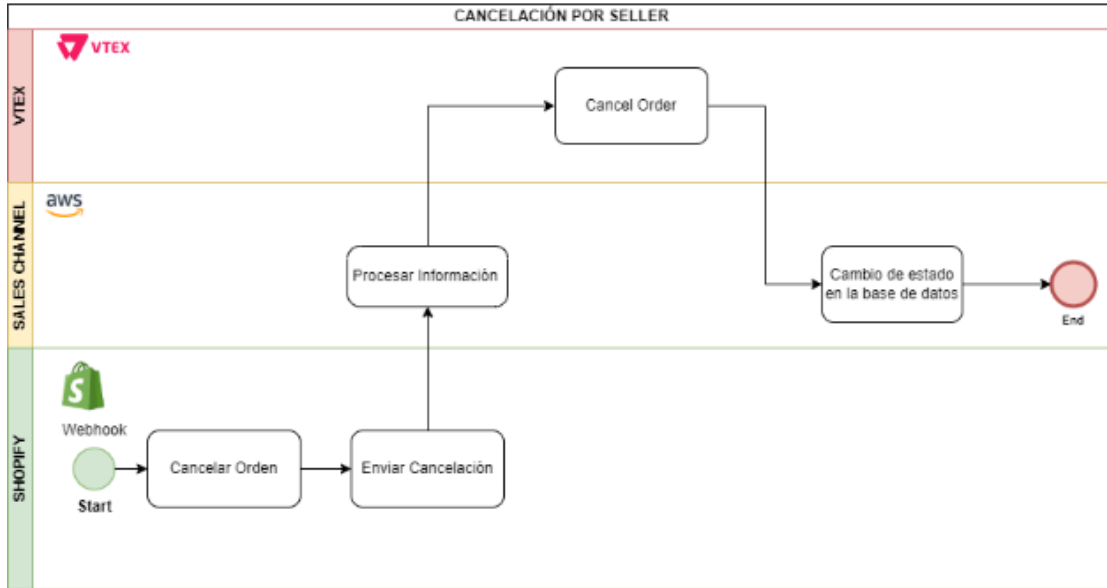
Figura 11
Flujo de tracking de órdenes



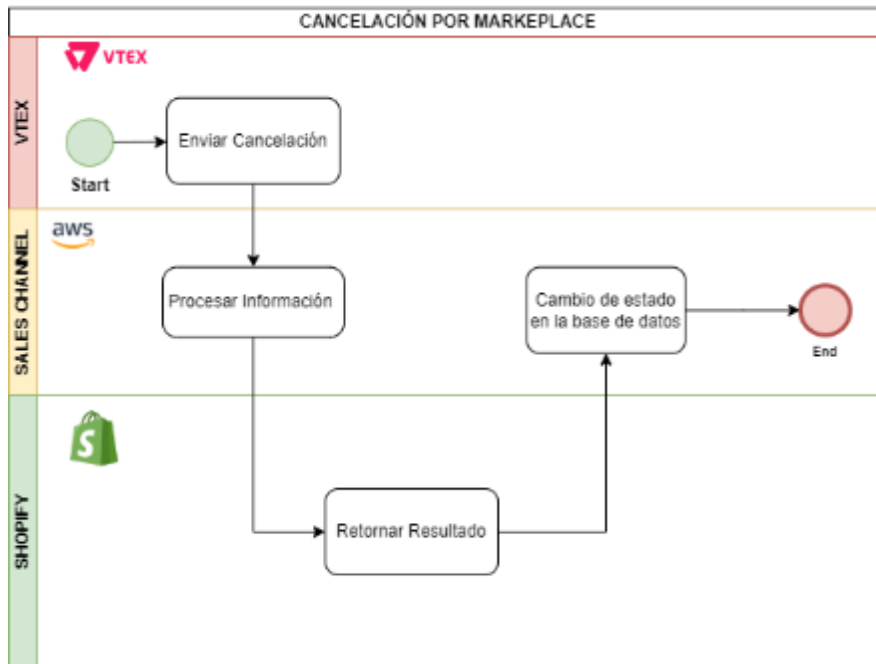
Cancelación de la orden por el seller: cuando se cancele una orden desde Shopify, se ejecutará un webhook que llamará al servicio expuesto en el conector, el cual buscará la orden, cancelará la orden en VTEX y cambiará el estado en la base de datos. El diagrama de Flujo que describe este proceso se puede ver en la **Figura 12**.

Figura 12

Flujo de cancelación desde el seller



Cancelación de orden por Marketplace: en el conector se expone un servicio a VTEX, el cual VTEX llamará cuando se cancele una orden desde el marketplace. El servicio buscará dicha orden, la cancela en Shopify y modifica el estado en la base de datos. En la **Figura 13** se visualiza el diagrama de flujo de este proceso.

Figura 13*Flujo de cancelación desde marketplace*

Para el tercer objetivo que era desarrollar los servicios para la creación y actualización de catálogo, los servicios se implementaron así:

Notificación de catálogo: este servicio recibe la cuenta Shopify correspondiente al seller, la tienda VTEX correspondiente al marketplace y un listado de las variantes o SKUs que se desean crear en VTEX; se procesa la información y se envía a VTEX cada SKU o variación a través de un endpoint expuesto por VTEX. En caso tal de que el endpoint de VTEX responda 404, indica que el SKU no está creado y el conector enviará a VTEX una sugerencia de SKU a partir de otro endpoint expuesto por VTEX, lo que hace que el SKU sea creado en el marketplace. Por otro lado, si el endpoint de VTEX responde 200, indica que el SKU ya está creado en VTEX y que se quiere actualizar el SKU.

Actualizar productos: este endpoint es expuesto a VTEX para cuando la respuesta de la notificación de catálogo sea 200. El endpoint recibirá en el body un listado de SKUs de los cuales el conector debe responder la información actualizada de estos SKUs provenientes de Shopify, para que esté siempre la información actualizada en VTEX.

El objetivo cuatro se trataba de desarrollar lo correspondiente a creación, cancelación, envío y tracking de la orden. Para esto se ejecutaron los siguientes servicios:

Procesar órdenes: este endpoint es expuesto a VTEX para cuando en el marketplace se termine de hacer una compra (checkout), VTEX consume este endpoint y el conector cree la orden en Shopify y responda la información de la orden que está siendo creada en VTEX y Shopify.

Despachar órdenes: este endpoint se expone a VTEX para cuando desde VTEX se aprueba el pago, VTEX consume este endpoint y el conector aprueba el pago y crea una transacción en Shopify, la cual será usada como factura en VTEX. El conector responde el id de la orden en el marketplace, el id de la orden en el seller y la fecha.

Notificación de factura: al crearse la transacción del pago de la orden en Shopify, se ejecuta uno de los webhooks configurados en VTEX. Este webhook consume el endpoint expuesto en el conector y este envía la información correspondiente a la transacción como factura a VTEX.

Tracking de órdenes: ya con el pago aprobado en VTEX y Shopify, el pedido está listo para preparar. En Shopify se prepara el producto y se le agrega la información de envío (número de guía del envío y la compañía encargada del envío), con esta información agregada en Shopify, se ejecuta otro de los webhooks configurados y se consume el endpoint expuesto en el conector. Este endpoint enviará la información del tracking a VTEX.

Cancelar orden desde Marketplace: el conector expone este endpoint a VTEX para cuando desde Vtex se haga la cancelación de una orden, en un primer intento retorne un json vacío, para que el conector se prepare para aprobar o no la cancelación.

Cancelar orden desde seller: el conector expone un servicio para cuando el seller desde Shopify cancele la orden. Cuando el seller cancela la orden se ejecuta uno de los webhooks configurados, el conector recibe la petición y procede a enviar la solicitud de cancelación a VTEX.

Durante el proceso de desarrollo se llegó a la conclusión que hacía falta otros servicios para lograr que la aplicación de Shopify y la aplicación VTEX IO funcionara de manera adecuada, entonces se crearon los siguientes servicios:

Creación de token para cuenta Shopify: Este endpoint se usa para asociar a la cuenta Shopify el access token que es usado para hacer el consumo de los servicios Shopify.

Creación de tiempo de envío de un pedido: A la hora de que VTEX desea actualizar la información de un producto y hace la petición al conector con el respectivo endpoint, el conector debe devolver dentro del body información del tiempo de envío, esta información no está disponible en Shopify, entonces el seller desde la aplicación Shopify debe crearla y para esto es que se expone este endpoint, para la creación del número de días de envío.

Crear tiendas: Un seller va a poder enviar sus productos a diferentes Marketplaces, entonces, este endpoint será usado para crear todos los marketplaces en la base de datos, a los que un seller desee enviar sus productos.

Obtener tiendas: Este endpoint es usado para obtener todas las tiendas o marketplaces asociadas con una cuenta Shopify o seller.

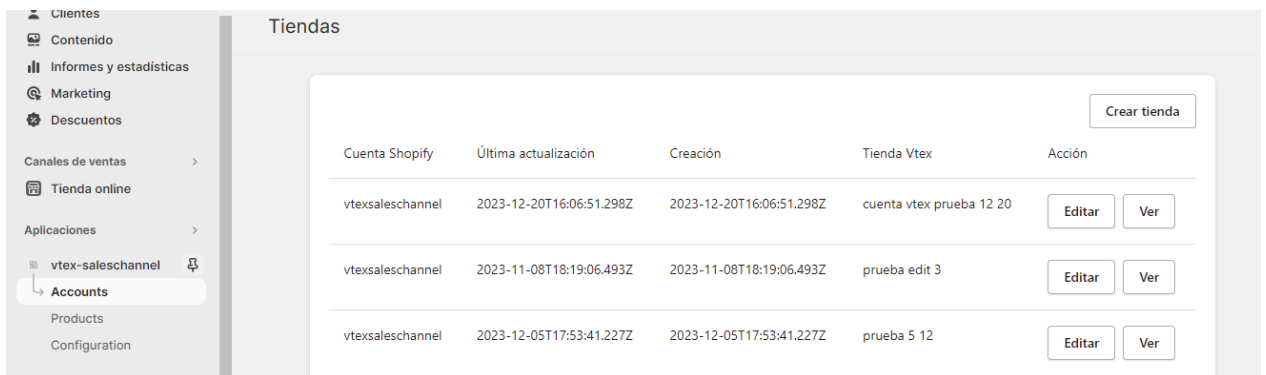
Modificar tiendas: Este endpoint es usado para actualizar la información de los marketplaces en la base de datos.

Como último objetivo, se implementó una aplicación Shopify la cual se divide en tres partes:

1. **Accounts:** Un seller puede enviar sus productos a diferentes marketplaces. Es en este módulo donde el seller podrá guardar, editar y visualizar las tiendas o Marketplaces a los que podrá enviar sus productos. En la **Figura 14** se puede visualizar cómo se ve este módulo.

Figura 14

Vista desde el módulo tiendas



Cuenta Shopify	Última actualización	Creación	Tienda Vtex	Acción
vtexsaleschannel	2023-12-20T16:06:51.298Z	2023-12-20T16:06:51.298Z	cuenta vtex prueba 12 20	<button>Editar</button> <button>Ver</button>
vtexsaleschannel	2023-11-08T18:19:06.493Z	2023-11-08T18:19:06.493Z	prueba edit 3	<button>Editar</button> <button>Ver</button>
vtexsaleschannel	2023-12-05T17:53:41.227Z	2023-12-05T17:53:41.227Z	prueba 5 12	<button>Editar</button> <button>Ver</button>

2. **Productos:** En este módulo estarán listados los productos disponibles en Shopify, los cuales podrán ser enviados a los marketplaces o tiendas Vtex creados en el módulo “Accounts”. Este módulo también tendrá la información de dichos productos y si ya fueron o no enviados a los marketplaces. En la **Figura 15** se puede visualizar cómo se ve el módulo de productos.

Figura 15
Vista desde el módulo productos

Nombre producto	Variante	Estado	Precio	Cantidad
broken sea	Pequeño / Blanco	No enviada	3.98	0
broken sea	Grande / Negro	No enviada	3.98	0
broken sea	Grande / Blanco	No enviada	3.98	0
broken sea	Grande / Rojo	No enviada	3.98	0
Camiseta manga corta	M / Negro	Enviada	3.50	5
Camiseta manga corta	M / Blanco	Enviada	3.50	6
Camiseta manga corta	S / Negro	Enviada	3.50	6
Camiseta manga corta	S / Blanco	Enviada	3.50	7
Camiseta manga corta	L / Negro	Enviada	3.50	8
Camiseta manga corta	L / Blanco	Enviada	3.50	9

3. **Configuration:** Este módulo se divide en dos, la primera parte es para asociar a la cuenta Shopify el access token para poder consumir la Api de Shopify, y la segunda parte es para poder crear o editar el número de días para los envíos. En la **Figura 16** se puede visualizar cómo se ve este módulo.

Figura 16
Vista desde el módulo configuración para credenciales

Shopify Configuration

Credenciales Envios

Credenciales

Access token

Guardar

Figura 17

Vista desde el módulo configuración para envíos



The image shows a screenshot of the 'Shopify Configuration' interface. At the top, there are two tabs: 'Credenciales' and 'Envíos', with 'Envíos' being the active tab. Below the tabs, the section is titled 'Envíos'. Underneath, there is a label 'Tiempo estimado de envío (en días)' followed by a text input field containing the number '0'. At the bottom of this section, there is a 'Guardar' button.

5 Análisis.3

Este proyecto ha logrado exitosamente establecer un puente entre VTEX y Shopify, proporcionando a los comercios que utilizan Shopify la capacidad de expandir sus operaciones al marketplace de VTEX. Asimismo, ahora tienen la capacidad de gestionar las órdenes generadas en el marketplace directamente desde sus cuentas en Shopify, lo que confirma el logro del objetivo general del proyecto.

Cada uno de los objetivos específicos delineados para este proyecto se ha cumplido de manera efectiva. Esto incluyó la configuración de los entornos de prueba tanto en VTEX como en Shopify, el diseño de la arquitectura y flujos básicos necesarios para la implementación, la creación de diversos servicios dentro del conector para facilitar los flujos previstos, y la elaboración de la aplicación en Shopify que permite la configuración de las tiendas, cuentas Shopify, y el envío de productos desde Shopify al marketplace de VTEX. En resumen, se han alcanzado todas las metas establecidas al inicio del proyecto.

Durante los seis meses de desarrollo, se enfrentaron desafíos puntuales que llevaron a la suspensión temporal del progreso. La interacción con la API de VTEX presentó dificultades debido a la falta de claridad en algunos puntos de la documentación. En consecuencia, fue necesario plantear preguntas específicas a VTEX mediante la apertura de tickets para obtener respuestas y resolver las dudas que surgieron en el proceso.

Además, la documentación de definición presentaba flujos con ambigüedades, lo que condujo a la realización de reuniones periódicas con el equipo para aclarar y concretar los flujos de trabajo. Estas sesiones de trabajo continuas resultaron fundamentales para garantizar la correcta implementación y funcionamiento del conector entre VTEX y Shopify.

6 Conclusiones

- El desarrollo del conector VTEX- Shopify abarca el flujo desde el envío de los productos de un seller a diferentes marketplaces, hasta la generación de órdenes, facturas, cancelación y tracking de las mismas.
- Desde la aplicación Shopify se pueden administrar los diferentes marketplaces, enviar los productos y hacer las diferentes configuraciones para la cuenta Shopify.
- Por los inconvenientes que se tuvieron durante el proyecto con la documentación brindada, se pudo evidenciar la importancia de tener la documentación clara y actualizada de las diferentes Apis que sean publicadas.
- Este proyecto es un MVP y la idea de Tita Media es seguir con el desarrollo de diferentes mejoras, para que este producto lo puedan usar diferentes comercios con la intención de ampliar sus mercados.
- Es importante dejar planteado dentro de la documentación inicial cuál es el alcance del proyecto, con especificaciones claras para que no haya lugar a ambigüedades y todas las partes tengan claridad de qué es exactamente lo que se va a entregar.

7 Recomendaciones

Para futuras versiones, se espera mejorar la interfaz gráfica de la aplicación Shopify, además de agregar a parte del envío de uno o varios productos, un cargue masivo de productos a partir de un archivo excel.

Referencias

Express - Node.js Web Application Framework. (s. f.). <https://expressjs.com/>

García, O. (2020, 3 septiembre). Disciplined Agile: Cómo encontrar mi way of working WOW. Proyectum.
<https://www.proyectum.com/sistema/blog/disciplined-agile-como-encontrar-mi-way-of-working-wow/>

Ltd, S. S. (2024, 5 enero). 23 Essential Shopify statistics you need to know in 2024. The Social Shepherd. <https://thesocialshepherd.com/blog/shopify-statistics>

VTEX and Nuvei partner amid Global Expansion Drive - VTEX. (s. f.). VTEX.
<https://vtex.com/sg-en/press/vtex-customers-globally-experience-33-gmv-growth-in-november-outperforming-the-ecommerce-market-2/#:~:text=As%20a%20global%20enterprise-grade,online%20stores%20in%2038%20countries.>

What-is-AWs. (s. f.). [Video]. Amazon Web Services, Inc.
<https://aws.amazon.com/es/what-is-aws/>