



Plan d'un SaaS pour l'automatisation complète d'une boutique Shopify

Introduction et Objectif

Ce plan détaille la création d'un service SaaS capable d'automatiser **100% de la création d'une boutique Shopify**, de la conception du design futuriste jusqu'à l'intégration du produit et la gestion des commandes. Le SaaS (nommé ici *LowAbleAI*) interagira avec le client pour collecter les informations nécessaires, puis orchestrera plusieurs services d'IA et d'intégrations afin de générer la boutique **de A à Z**. L'objectif est qu'avec quelques informations de base fournies par le client (comme un lien produit ou une description), la plateforme construise une boutique clé en main : design animé et attrayant, textes optimisés, produit importé avec un prix margé, plugins de commande automatique configurés, modes de paiement et expédition prêts, le tout **sans intervention manuelle supplémentaire**.

Points clés à respecter :

- Le client doit pouvoir **choisir sa langue** dès le départ (français, anglais, etc.) pour que toute la boutique (textes du site, interface) soit générée dans cette langue.
- La boutique Shopify sera **créée sous le compte du client**, mais via notre **lien d'affiliation** Shopify pour générer des commissions. Cela implique soit l'utilisation de l'API Partenaire Shopify pour créer un magasin de développement puis le transférer au client, soit de rediriger le client vers une inscription Shopify par lien affilié ¹. L'affiliation garantit qu'une commission est gagnée dès que le marchand souscrit à un plan Shopify payant ¹.
- Le SaaS privilégiera **les outils d'IA "via prompt"** pour chaque étape (conception visuelle, génération de textes, optimisation) afin de bénéficier des modèles existants de pointe. Nous sélectionnerons les meilleurs services d'IA disponibles via API pour chaque besoin (ex. modèles de génération d'images, de texte, etc.), plutôt que de tout coder manuellement.
- L'approche sera **optimisée à chaque étape**. Par exemple, pour importer les données produit, on utilisera de préférence un outil existant (comme **DSers**) plutôt que de réécrire un scraper : DSers est un partenaire officiel d'AliExpress qui permet d'importer des produits et de passer des centaines de commandes en quelques secondes ². Si DSers n'est pas utilisable via API, on pourra en dernier recours parser la page produit du fournisseur. L'idée générale est de réduire les frictions et de capitaliser sur des solutions robustes pour éviter les bugs.

Nous allons décomposer le SaaS en plusieurs **services modulaires**, puis montrer comment les orchestrer en une séquence de prompts et d'actions automatisées.

Service 1 : Analyse du produit et génération du design futuriste animé

Le **Service 1** reçoit en entrée le lien du produit (par exemple la page AliExpress du produit) ou un prompt décrivant le produit. Son rôle est double : **analyser** le produit (catégorie, style, usage) et **générer le design visuel de la boutique** de façon attrayante, futuriste et animée en accord avec ce produit.

Tâches du Service 1 :

- **Choix d'une esthétique et d'une mise en page générale** : À partir du produit, l'IA détermine un thème visuel cohérent. Par exemple, pour un gadget technologique, le service proposera une ambiance futuriste haute technologie (couleurs néon, arrière-plans animés, typographie moderne). Pour un produit de bien-être, il choisirait une palette apaisante, etc. L'IA peut s'inspirer de la description du produit ou d'images du lien pour adapter le style.
- **Génération d'images originales et de visuels animés** : Ce service s'appuie sur un moteur de génération d'images par IA via API pour créer **des visuels uniques** du site. On utilisera les meilleurs modèles disponibles : par exemple **Stable Diffusion XL** ou DALL·E 3 pour obtenir des images haute résolution et photoréalistes à partir de textes descriptifs ³. Stable Diffusion est particulièrement adapté car il peut générer des images photoréalistes à partir de presque n'importe quelle description textuelle ³. Concrètement, le service formulera des *prompts* du type : « *Image AI, génère-moi une bannière héroïque pour la page d'accueil montrant [le produit] dans un décor futuriste, avec une animation de fond en boucle, dans le style [cyberpunk/sci-fi minimaliste]...* ». On pourrait aussi demander une série d'images (par ex. un logo stylisé, des icônes, plusieurs images produit sur fond amélioré).
- **Création d'un élément animé (assistant ou header animé)** : L'idée est d'ajouter de la vie au site pour attirer l'attention des visiteurs. Cela peut prendre la forme soit d'un **header animé** (par ex. une vidéo courte en background, un diorama 3D du produit tournant), soit d'un **assistant virtuel animé** (une mascotte ou avatar futuriste qui accueille les visiteurs). Pour accomplir cela :
 - On peut intégrer un service d'IA génératrice de vidéo ou d'animation. Par exemple, l'API de **RunwayML Gen-2** pourrait créer un court clip vidéo à partir d'une description (montrant le produit en usage futuriste). De même, **Kaiber** ou **Synthesia** peuvent générer des animations ou avatars à partir de prompts. Synthesia permettrait d'avoir un avatar qui parle, mais ici un simple élément visuel animé pourrait suffire.
 - Si ces outils vidéo sont trop lourds, on peut se rabattre sur du **CSS animé** ou du **WebGL**. Par exemple, l'IA peut suggérer du code CSS pour faire des animations au scroll ou au survol (parallaxe, transitions douces), ou intégrer une animation pré-faite (Lottie files ou Canvas). L'important est que le design du site ne soit pas statique : il doit donner une impression de modernité dynamique (par ex. un petit mouvement constant dans le header).
- **Respect de l'identité produit et attrait marketing** : Le design généré doit *donner envie d'acheter*. Cela signifie mettre en valeur les points forts du produit visuellement. L'IA peut, via son prompt, **styliser les images du produit** (retirer un fond d'image basique et le remplacer par un contexte plus vendeur, améliorer la lumière, etc.). On peut utiliser pour cela des outils d'édition d'image par IA (Stability propose des APIs d'*inpainting*, d'*upscaling*, etc. ⁴ ⁵). Par exemple, améliorer la résolution des photos produit, générer un fond lifestyle (le produit en situation réelle) et même **remplacer entièrement les photos d'origine** par des rendus 3D/artistiques du produit plus attractifs. Le service 1 peut aussi générer un **logo** pour la boutique s'il n'y en a pas, en utilisant toujours la même palette futuriste.

En sortie, le Service 1 fournit :

- Un **ensemble d'assets visuels** (images de bannières, logos, images de produit retouchées, éventuellement clips GIF/vidéo courts) adaptés au thème futuriste choisi. Ces assets sont *originaux* (pas de template figé) afin que la boutique soit unique et parfaitement alignée au produit ⁶ ⁷. (*Par exemple, l'outil Dora AI parvient à générer des images et une identité visuelle entièrement sur mesure à partir d'un prompt, sans utiliser de modèles tout faits* ⁸.)
- D'éventuelles **instructions d'intégration** pour ces visuels dans le site Shopify. Par exemple, le service peut recommander d'utiliser une section de type diaporama pour la bannière animée, ou de placer l'avatar assistant en bas de page via un widget. Il peut générer le code HTML/CSS/JS nécessaire pour ces éléments spéciaux.
- Une **charte graphique** proposée (codes couleur hexadécimaux, police de caractère à utiliser, style d'icônes, etc.), pour que le Service 2 (contenu) et l'assemblage final utilisent une esthétique cohérente.

Outil IA pressenti : **Stability AI (Stable Diffusion)** via API pour les images, car il offre une grande variété de styles (3D, photo, peinture, etc.) et une fidélité élevée au prompt ⁹ ³. Pour les animations, envisager **RunwayML Gen-2** pour du vidéo court, ou à défaut utiliser du code d'animation guidé par GPT-4.

Service 2 : Génération du contenu textuel et de la mise en page du site

Le **Service 2** est dédié à tout ce qui est **texte et structure du site**. Il prend lui aussi en entrée les infos du produit (et idéalement les sorties du Service 1, comme les thèmes graphiques, pour s'aligner en ton) et produit : les textes de vente, les sections du site et leur disposition optimale. Ce service utilisera un modèle de traitement du langage naturel (NLP) avancé via API (typiquement GPT-4 ou équivalent) pour **rédiger du copywriting persuasif** et suggérer l'agencement des pages.

Tâches du Service 2 :

- **Rédaction de la fiche produit complète** : À partir des données brutes (description fournisseur, caractéristiques techniques), l'IA va rédiger une description produit *optimisée marketing*. Cela comprend un **titre produit percutant**, un **slogan** court, une description longue structurée en bénéfices pour le client, et éventuellement des **bullet points** pour les points clés. Si le client a donné un prompt plutôt qu'un lien, l'IA utilisera ce prompt (ex: « *chaise de bureau ergonomique en mesh* ») pour inventer une description vendueuse en comblant les détails manquants de manière plausible. Le texte sera adapté à la **langue choisie par le client**. On veillera aussi à inclure des **mots-clés SEO pertinents** pour le référencement naturel du produit.
- **Contenu des autres pages et sections** : Le service va générer tous les textes nécessaires pour un site e-commerce complet, par exemple :
- **Accueil (Home page)** : un *header* avec une proposition de valeur claire, un paragraphe d'introduction mettant en scène le produit (ou la marque) de façon séduisante, des sections « *avantages* », « *fonctionnalités* » ou « *pourquoi nous choisir* », des témoignages (fictifs ou placeholders), etc.
- **Page À propos / Notre histoire** (si pertinent) : un texte humanisant la marque, créé automatiquement si le client n'en fournit pas.
- **FAQ** : une liste de questions fréquentes avec réponses rassurantes sur le produit (dimensions, utilisation, garantie, livraison...).
- **Politiques standard** : conditions générales de vente, politique de retour, mentions légales – on peut insérer des modèles prédéfinis vu que c'est standard, mais l'IA peut remplir les infos spécifiques (nom de la boutique, politique de livraison, etc.). (D'ailleurs, des outils comme BuildYourStore fournissent ces pages templates automatiquement ¹⁰ ¹¹.)
- **Structure du site optimisée** : En se basant sur les meilleures pratiques e-commerce, l'IA décide de la hiérarchie des sections. Par exemple, la page d'accueil devra **commencer par** un visuel fort (fourni par Service 1) et un titre accrocheur, suivis éventuellement d'un bouton d'appel à l'action « Acheter maintenant ». Ensuite peut venir une section présentant les bénéfices du produit en colonnes avec icônes, puis une section avec le produit en vedette et son prix, etc. Le Service 2 peut s'inspirer de structures existantes de sites qui convertissent bien. Il générera soit une description formelle de la mise en page (que le Service 4 pourra traduire en sections Shopify), soit directement du **code JSON** compatible avec le constructeur de pages (Shopify Sections schema, ou un builder tiers). À noter qu'en 2025 il existe des constructeurs de pages alimentés par l'IA (par ex. **EComposer AI**), capables de proposer des mises en page complètes ¹². Notre service vise la même capacité : « *auto-générer la mise en page complète du site* » ¹³ en fonction du contenu produit.
- **Optimisation SEO et conversion** : L'IA veillera à ce que les textes comportent les bons mots-clés, et que l'appel à l'action soit clair. Elle peut aussi proposer les métaddescriptions des pages, les balises alt des images produit, etc., pour améliorer le référencement ¹⁴ ¹⁵. De plus, l'IA connaît les principes de

conversion (par ex, limiter les étapes du checkout, mettre en avant la livraison offerte, etc.), donc les textes iront dans ce sens (ex: phrases qui créent l'urgence, garantie « satisfait ou remboursé », etc.).

En sortie, le Service 2 fournit :

- **Les textes** prêts à l'emploi pour chaque page et section identifiée (généralement on se concentre sur la page d'accueil et la fiche produit, plus éventuellement une page « A propos » et les pages légales).
- **Une suggestion de structure** pour la page d'accueil (ordre des sections, titres de section) et pour la page produit (disposition des images, description, reviews section).
- **Éventuellement du code** : par exemple, du JSON à insérer dans le thème Shopify pour créer les sections, ou du HTML avec placeholders. S'il y a du contenu très custom, GPT-4 peut générer du code Liquid/CSS pour réaliser une section précise. Cependant on pourra aussi s'appuyer sur un **thème préexistant modulaire** pour éviter des bugs (voir Service 4).

Outil IA pressenti : OpenAI GPT-4 via API (ou modèle équivalent comme Anthropic Claude). GPT-4 est un des meilleurs en génération de contenu contextuel et créatif en 2025, et peut aisément produire un texte cohérent et convaincant dans la langue choisie. Des alternatives spécialisées e-commerce existent (par ex. **Jasper** ou **Writesonic** ont des modules de copywriting marketing), mais GPT-4 offre plus de flexibilité. Par exemple, *Jasper AI* est reconnu pour la rédaction marketing et s'intègre bien en complément d'un page builder¹⁶. On peut l'utiliser pour affiner certains contenus (slogans publicitaires par ex.). Mais pour centraliser, GPT-4 suffira dans la plupart des cas.

Service 3 : Création du produit, tarification et optimisation e-commerce

Le **Service 3** prend tout ce qui a été produit par les services précédents (design visuel, textes, données produit brutes) et va effectuer les actions clés pour **créer la boutique en elle-même sur Shopify**, y intégrer le produit, et optimiser tous les paramètres liés à la vente. Ce service est en quelque sorte le **cerveau “business”** du SaaS : il s'occupe des détails pragmatiques comme la tarification, le stock, l'intégration fournisseur, les plugins, etc., pour que la boutique soit réellement fonctionnelle et prête à encaisser des commandes.

Tâches du Service 3 :

- **Importation / création du produit dans Shopify** : Le service utilise l'API Shopify pour créer le produit dans la boutique du client. S'il dispose d'un lien fournisseur (ex: AliExpress), deux approches sont possibles :
 1. **Via DSers (ou plugin similaire)** : Idéalement, on passe par DSers qui est spécialisé dans l'import de produits AliExpress vers Shopify. DSers utilise l'API AliExpress pour récupérer toutes les informations produit (images HD, variantes, prix fournisseur) et les injecte dans Shopify¹⁷ ¹⁸. Cela évite de gérer nous-mêmes les détails (ex: téléchargement d'images, options de couleur/taille, etc.). DSers peut importer un produit en quelques secondes et établir le lien avec le fournisseur pour les futures commandes². *Intégration possible* : Si DSers offre une API ou du moins un moyen automatisé (il supporte l'import par CSV selon leurs docs¹⁹), notre service peut lui envoyer le lien produit à importer. Sinon, on peut demander au client de cliquer sur un bouton « Importer sur DSers ».
 2. **En parsing direct** : Si on ne peut pas automatiser DSers, le service ira chercher les infos sur la page produit. Il peut utiliser l'API AliExpress officielle (si accessible aux développeurs tiers) ou une API tierce (il existe par ex. des endpoints sur RapidAPI pour obtenir les détails d'un produit AliExpress à partir de son URL²⁰). À défaut, un scraper peut analyser le HTML. Avec ces données, on crée le produit via l'API Shopify (title, description, images, prix coûtant, etc.).
- **Amélioration des photos produit** : Juste après l'import, on peut remplacer/améliorer les visuels du produit dans Shopify. Par exemple, substituer les images de base par celles générées par le Service 1

(plus qualitatives ou stylées). Au minimum, on peut ajouter en plus les images AI comme images secondaires. Le but est que la fiche produit soit très attrayante visuellement par rapport aux concurrents.

- **Choix du prix de vente*** : *C'est une étape cruciale. Le service calcule un prix de vente recommandé qui inclut une *marge bénéficiaire suffisante.* Typiquement, en dropshipping les marges moyennes tournent autour de 10% à 30% ²¹. On peut viser ~20% au début comme marge minimum acceptable ²¹. Pour déterminer le prix :

- Si le coût fournisseur (COGS) est connu via AliExpress (par ex. \$5), le service peut appliquer une formule (ex: multiplier par 3 ou 4 pour couvrir frais marketing et avoir ~70% de marge brute, ou un markup fixe par tranche de prix ²²). On peut aussi demander à une IA d'ajuster : « *Propose un prix de vente pour un produit coûtant \$5 qui reste attractif pour les clients tout en assurant une marge bénéficiaire* ». L'IA pourra répondre par ex. \$19.99 en justifiant que c'est 60% de marge, ce qui est dans la fourchette recommandée de 15% à 60% ²³.

- Le service peut aussi **rechercher les prix des concurrents** pour un produit similaire (via une requête web, ou en s'appuyant sur une base de données). Cependant, en 100% automatisé et rapide, le plus simple est d'appliquer une stratégie de pricing standard. On pourra toujours laisser le client ajuster plus tard.

- On s'assure d'intégrer les **frais de port** dans ce prix si on offre la livraison gratuite (voir point suivant sur l'expédition). Par exemple, si le fournisseur propose l'envoi ePacket à 2\$, on ajoute 2\$ au prix de base et ensuite on multiplie. **Astuce** : offrir la livraison "gratuite" est fortement conseillé car plus de 66% des acheteurs s'y attendent ²⁴. Donc on privilégiera un prix **tout compris** un peu plus élevé, plutôt que d'ajouter des frais de port au checkout, ce qui augmente la conversion ²⁴. Le service calculera cela automatiquement.

- **Paramétrage des détails du produit** : Ajouter le stock (en dropshipping, on peut mettre une quantité élevée ou « infini » si c'est print-on-demand). Fixer le poids (pour la livraison, on peut mettre un poids réaliste si ça compte pour les tarifs). Renseigner la catégorie, tags, vendor, type de produit pour l'organisation du site.

- **Optimisation de la page produit** : Au-delà du texte et des images déjà traités, le service peut ajouter des éléments qui renforcent la confiance et incitent à l'achat. Par exemple :

- Des **badges de confiance** (paiement sécurisé, garantie 30 jours, etc.) – cela peut être inséré via une application tierce ou un simple code HTML dans la description.

- Une section "**Produits recommandés**" (peut rester vide s'il n'y a qu'un produit, mais prête pour en ajouter d'autres).

- Intégrer un système d'**avis clients**. On pourrait générer quelques avis fictifs initiaux via l'IA (controversé mais pratiqué) ou simplement installer un plugin d'avis (Judge.me, etc.) pour que le client puisse importer des avis AliExpress plus tard.

- Configurer des **offres de bundle ou de réduction** via une app (Atlas AI Store Builder par ex. propose des bundles et upsells intégrés ²⁵). Notre service pourrait également utiliser l'API Shopify pour créer des **codes promo** (par ex. WELCOME10 pour 10% off première commande).

- **Intégration du fournisseur et automatisation des commandes** : C'est un point essentiel mentionné par le client. Une fois le produit en place, on doit faire en sorte que lorsque des commandes sont passées sur la boutique, elles soient automatiquement envoyées au fournisseur pour traitement (**dropshipping automatisé**). La solution recommandée est d'**installer et configurer l'application DSers** (ou un équivalent) sur la boutique Shopify du client. DSers synchronise la boutique avec AliExpress : à chaque commande reçue, DSers va la reproduire sur AliExpress auprès du bon fournisseur, payer et remplir les informations, puis synchroniser le numéro de suivi dans Shopify ²⁶. Cela évite au marchand d'entrer manuellement les commandes une par une. DSers permet même de passer des **centaines de commandes en quelques secondes** de manière groupée ².

- **Mise en œuvre** : Notre service, via l'API Shopify, peut déclencher l'**installation de DSers** (en générant un lien d'installation d'app à accepter par le marchand). On devra ensuite guider le client pour **relier son compte AliExpress** à DSers (c'est une étape manuelle via DSers, où il autorise l'accès à son compte

AliExpress). On peut fournir un tutoriel rapide.

- Une fois DSers actif, si le produit a été importé via DSers à l'étape précédente, il est déjà lié à AliExpress. Si on l'a importé manuellement, DSers offre généralement une fonction pour **faire correspondre** un produit Shopify existant avec un produit AliExpress (en recherchant par nom ou ID). On peut soit l'automatiser via DSers (s'ils ont une API interne), soit demander au client de sélectionner le bon lien dans DSers (ce qui est simple).

- Pour d'autres fournisseurs que AliExpress : Si le lien fourni était vers un autre site (Alibaba, Amazon, ou un fournisseur local), on devra trouver des solutions alternatives. Par exemple, si c'est un site partenaire, peut-être qu'on passe par **AutoDS** ou **CJ Dropshipping** qui gèrent plusieurs sources. Mais comme AliExpress est mentionné explicitement, on se focalise sur lui.

- **Configuration de l'expédition (shipping)** : En parallèle de DSers, on paramètre dans Shopify les options de livraison visibles pour le client final. Si on part sur la stratégie "livraison gratuite mondiale", le service peut créer un **profil d'expédition** avec une zone "Monde entier" et un tarif 0 € (et éventuellement le libellé « Livraison offerte » pour transparence). Shopify API permet de créer ces règles. Si au contraire on veut facturer la livraison, on pourrait paramétrier par pays, mais cela complexifie et réduirait potentiellement les conversions. On privilégie donc la gratuité en répercutant dans le prix produit, comme dit plus haut ²⁴.

- En plus, on peut intégrer un **outil de suivi des colis** (par ex. l'app *AfterShip*) pour que les clients puissent suivre la livraison de leur commande. DSers synchronisant les numéros de suivi ²⁵, il suffira d'envoyer au client le lien de suivi. AfterShip peut être installé via API aussi.

- **Compte d'expédition** : S'il s'agissait d'un marchand qui expédie lui-même, on aurait dû conseiller de créer un compte transporteur (Colissimo, USPS etc.). Mais en dropshipping AliExpress, c'est le fournisseur qui gère l'expédition. Donc ici, pas besoin que le client ouvre un compte chez un transporteur. Cependant, il doit *tout de même* se créer un compte sur AliExpress (ou utiliser celui qu'il a) pour payer ses fournisseurs. On peut le rappeler dans les instructions (par ex. "Assurez-vous d'avoir un compte AliExpress pour connecter à DSers").

- **Configuration des paiements** : Pour que la boutique soit prête à encaisser de l'argent, on doit intégrer les moyens de paiement. Shopify facilite cela : dans l'interface Shopify du client, il doit activer Shopify Payments (s'il est dans un pays supporté) ou connecter un compte **Stripe** ou **PayPal**.

- Notre SaaS ne peut pas directement créer le compte Stripe du client (car cela requiert ses infos bancaires). Par contre, on peut le guider étape par étape. Par exemple, après avoir créé le magasin, on affiche une checklist « Connecter un moyen de paiement ». On expliquera que pour Stripe, il doit créer un compte Stripe (s'il n'en a pas) puis activer Shopify Payments ou rentrer ses API keys si nécessaire. Pour PayPal, juste se connecter avec son compte PayPal Business dans Shopify.

- Si on veut être plus intégré, on peut utiliser l'API Stripe Connect pour pré-remplir l'inscription de Stripe, mais cela complexifie beaucoup et sort du scope. Mieux vaut fournir des instructions claires.

- On peut envisager de demander ces infos dès le début au client (ex: "Entrez votre email PayPal si vous voulez accepter PayPal" ou "Voulez-vous utiliser Stripe ? Y/N"). Mais par sécurité, c'est mieux que le client fasse lui-même sur Shopify. Donc on s'assurera juste qu'il n'oublie pas cette étape indispensable.

- **Autres optimisations e-commerce :**

- **Nom de domaine personnalisé** : Notre service peut faciliter l'achat d'un domaine (Shopify propose directement l'achat de domaine dans son interface, mais via API c'est possible aussi). Notons que BuildYourStore fournit un domaine .store gratuit ²⁷ – peut-être via une offre partenaire. On pourrait éventuellement intégrer une API de fournisseur de domaines (GoDaddy, etc.) pour acheter un .com ou .store. Mais ce n'est pas obligatoire pour le MVP de la solution. On peut lancer avec le sous-domaine monboutique.myshopify.com puis recommander d'ajouter un domaine.

- **Installation d'applications marketing** : par ex. une app de popup (pour capter les emails ou offrir un code promo à la sortie), une app de chat en ligne ou chatbot AI pour conseiller les clients, etc. Étant donné qu'on vise l'automatisation complète, on peut pré-installer une sélection d'apps utiles via leur API d'installation. Exemple d'apps : Klaviyo (email marketing) ou Tidio (chatbot). Shopify Sidekick (assistant AI natif) est utile pour le marchand mais pas pour les clients, donc pas prioritaire.

- *Vérifications finales : Le service 3 ou l'orchestrateur fera un contrôle qualité final. Ex : vérifier que le site est bien multilingue si besoin. Si le client a choisi le français, peut-être fournir aussi une version anglaise? (Shopify ne traduit pas tout automatiquement – il y a Shopify Translate & Adapt* app, ou Weglot. On pourrait intégrer une app de traduction AI si multi-langue souhaitée). Mais si le client ne choisit qu'une langue, on restera sur celle-ci globalement.*
- Vérifier aussi que toutes les images se chargent, que les liens (vers CGV, etc.) fonctionnent, que la page de checkout fonctionne (Shopify checkout gère ça tout seul normalement).

En résumé, le Service 3 **transforme la boutique “sur le papier” en une boutique réelle en ligne** sur Shopify, **optimisée pour la vente** dès le départ. Chaque élément clé de réussite e-commerce est mis en place automatiquement : bon produit, bon prix, joli design, texte vendeur, paiement opérationnel, commandes automatiquement passées au fournisseur, etc.

Orchestration finale : intégration de tous les services dans un flux unique

Cette partie correspond au **Service global** (ou orchestrateur) qui va enchaîner les appels aux autres services et interactions avec le client. Le but est de **rassembler tous ces services en un seul SaaS** cohérent du point de vue de l'utilisateur final. LowAbleAI va guider le client pas à pas, tout en déclenchant en arrière-plan les tâches automatisées.

Infrastructure et enchaînement global :

- 1. Interface utilisateur & choix de la langue :** LowAbleAI démarre en demandant la langue préférée (« Veuillez choisir votre langue / Please choose your language »). Ce choix est crucial car toutes les générations de contenu ultérieures (textes du site) en dépendront. Supposons que le client choisit *Français*. L'IA basculera dans ce contexte linguistique pour la suite.
- 2. Collecte des informations de base :** LowAbleAI invite ensuite le client à fournir toutes les données nécessaires en **une fois si possible**, afin d'éviter des allers-retours. Par exemple, présenter un petit formulaire ou une série de questions:
 - *Produit à vendre* : soit un **lien** vers le produit (idéalement AliExpress) soit une **description** du produit si pas de lien.
 - *Nom souhaité pour la boutique* : si le client a une idée (sinon on proposera un nom généré).
 - *Email du client* (pour la création du store Shopify et pour les comptes tiers).
 - *Devise et pays cible* (pour configurer la devise du magasin, par ex € pour France, \$ pour international).
 - *Préférence de marge* (optionnel, on peut demander « Souhaitez-vous un prix plutôt **bas** pour écouler volume, **moyen**, ou **élévé** pour maximiser la marge ? » – cela peut orienter le calcul du prix).
 - *Souhaitez-vous offrir la livraison gratuite aux clients ?* (on recommandera “oui” par défaut en expliquant les avantages, comme mentionné ²⁴).
 - *Modes de paiement que vous utiliserez* (ex: Stripe, PayPal – juste pour qu'on sache lesquels souligner dans les instructions).
 - *Informations légales* (si le client est une entreprise existante, peut-être qu'il souhaite mettre son SIRET, adresse, etc., dans les CGV. On peut demander s'il veut personnaliser les mentions légales ou utiliser les modèles standard).
 - *Compte Shopify existant ?* (Dans le cas où le client aurait déjà un store vide sur son compte. Mais ici on présume que non, on va en créer un nouveau via l'affiliation).

LowAbleAI s'assure d'avoir tout ce qu'il faut. Par exemple, s'il manque le lien produit, il ne pourra pas importer – donc ce champ est obligatoire d'une façon ou d'une autre. Idem pour l'email (sinon on ne peut pas créer le store ou lui transférer).

- 3. Crédit de la boutique Shopify via lien affilié :** Une fois les infos de base reçues, LowAbleAI

procède à la création du store. Deux cas :

- **Processus via le Partenariat Shopify (automatisé)** : Étant un Shopify Partner, notre plateforme peut utiliser l'API/SDK partenaire pour créer un "dev store" pour le client ²⁸. On fournira lors de la création le nom de la boutique (demandé ou généré), l'email du client comme propriétaire, la langue/dévise, etc. Ce magasin de développement est créé *via notre lien d'affiliation* en quelque sorte, puisque les partenaires Shopify obtiennent une commission quand un magasin transféré prend un abonnement ¹. Après génération du contenu du store, on pourra soit **transférer la boutique** au client (il recevra un email pour accepter la propriété) ²⁹, soit simplement lui donner les accès (s'il a déjà un compte Shopify, on peut ajouter son compte comme propriétaire).
- **Processus via lien d'affiliation (manuel)** : Alternativement, LowAbleAI peut afficher au client un bouton « Inscrivez-vous sur Shopify (gratuit 14j) via notre lien ». Le client remplirait le formulaire Shopify standard (avec son email, nom de store...). Ensuite, pour qu'on puisse agir dessus, il devra installer notre application ou nous donner un accès administrateur temporaire. Ce flux est un peu moins fluide, mais envisageable.

Pour une **expérience optimale**, on priviliege la première option où *LowAbleAI crée directement la boutique dans le compte partenaire*. Ainsi, le client n'a presque rien à faire, et on garde le contrôle technique durant la configuration. On s'assure juste de renseigner notre **referral link** pour que Shopify sache que c'est référé ¹ (c'est automatique via le tableau de bord partenaire lors de la création d'un dev store transfert).

4. Choix et installation du thème / structure du site : Une fois la boutique créée (en quelques secondes), LowAbleAI va configurer le **thème visuel**.

- On peut soit utiliser un **thème Shopify existant** (ex: Dawn, qui est très customisable) comme base, soit un thème premium optimisé (le SaaS Atlas AI par ex. propose son propre thème optimisé pour conversion ²⁵). Nous pouvons avoir développé notre propre thème « LowAble Futurist » qui inclut des sections animées, etc. Supposons qu'on ait un thème template adapté aux animations futuristes, on l'uploadé via API. Sinon, on prend un thème standard et on le personnalise.
- LowAbleAI utilise alors les outputs du Service 1 (charte graphique, images) et du Service 2 (textes, structure) pour **peupler le thème**. Via l'API ou les fichiers de configuration du thème, on va :
 - Importer les images dans la bibliothèque Shopify (API GraphQL `stagedUploadsCreate` pour uploader les fichiers), puis les assigner dans les sections (par ex, mettre l'image AI en arrière-plan du header).
 - Créer les sections de la page d'accueil. Si on a un JSON de structure proposé par l'IA, on le fusionne dans le fichier `template.json` du thème. Sinon, on utilise les sections prédefinies du thème : e.g. une section "Hero" à laquelle on passe le texte de slogan et l'image, une section "Rich text" pour la présentation, etc.
 - Insérer le contenu textuel dans les emplacements adéquats. Par exemple, mettre la description du produit dans la page produit, remplir la page À propos avec le texte généré, etc., via l'API de Page ou via les Customizer settings.
 - Appliquer la charte graphique : changer les couleurs du thème (primaire, secondaire) selon celles suggérées (ex: néon bleu #00ffff pour un look futuriste, etc.), changer la police si besoin (Shopify permet l'ajout de polices Google via le settings du thème).
 - Configurer le menu de navigation (ex: "Accueil", "Boutique", "Contact" liens).
 - Si des éléments de design nécessitent du code custom (ex: insérer une animation Lottie ou un code HTML pour avatar), LowAbleAI peut injecter ce code dans une section Custom HTML du thème, ou ajouter un petit script dans le fichier theme.liquid.
- **Vérification** : on s'assure que le site rendu correspond bien à l'aperçu voulu. LowAbleAI pourrait générer un aperçu (Shopify permet de preview le thème non publié). On peut éventuellement prendre une capture d'écran du rendu pour vérifier aucun texte n'est trop long ou mal placé.

5. Import du produit et réglages business : LowAbleAI appelle maintenant le Service 3 pour gérer l'import produit, pricing, etc. Concrètement :

- Si DSers est utilisable par API, on l'appelle pour importer le produit avec le lien fourni. Sinon, on a déjà scrappé les infos, donc on utilise l'API Shopify REST (`/admin/products.json`) pour créer le produit avec titre, description (celle de Service 2, pas la description AliExpress brute), images (celles améliorées de Service 1 d'abord, puis éventuellement les originales en plus), variantes (couleurs/taille, prix coûtant en *Compare at Price* éventuellement ou champ caché).
- Appliquer le **prix de vente** calculé. Exemple : coût \$5, on décide de vendre \$19.99, ce qui fait ~75% marge, suffisant pour couvrir pub et frais ²¹. On peut mettre le *Compare at price* plus haut (ex: \$29.99 barré, pour montrer une promotion). Ces décisions peuvent être fixées par une règle ou par l'IA.
- Configurer le stock (Shopify API `inventoryLevel/set` pour l'entrepôt par défaut). On peut mettre un stock fictif élevé (100 unités) pour inciter (et on activera "Continuer à vendre si rupture" au cas où).
- **Publier** le produit sur le canal de vente principal (la boutique en ligne) afin qu'il soit visible.
- Créer la **collection** principale si besoin (ex: "Produits" contenant ce produit).
- Mettre en place la livraison gratuite : via API, créer un ShippingZone "Rest of World" avec un ShippingRate à 0€.
- Si le client a choisi un pays spécifique, on peut aussi créer une zone pour ce pays avec 0€, et éventuellement une zone "International" avec 0€ aussi, pour tout couvrir.
- Installer DSers app : on utilise l'API d'installation d'app Shopify ou on génère un lien. LowAbleAI peut automatiquement install l'app si le store est encore sous notre gestion (dev store). Si ce n'est pas possible sans action, on fournit au client un lien direct pour installer DSers en un clic.
- LowAbleAI config DSers (via webhook, on peut invite le client à suivre un mini-guide visuel pour connecter son compte AliExpress – cette partie sera semi-automatique).
- Autres apps : installer l'app d'avis clients, l'app de tracking colis, etc., de la même manière, pour enrichir le store.

6. Finalisation et transfert : À ce stade, la boutique est fonctionnelle. LowAbleAI peut présenter un **récapitulatif** au client : « *Votre boutique est prête ! Voici les dernières étapes :* ». Parmi ces étapes :

- **Prise en main du store** : Si on a créé le store en dev, on transfère la propriété. Le client recevra un email pour accepter. On l'invite à le faire. Une fois transféré, le client devra choisir un plan Shopify (même en essai gratuit) pour continuer. S'il souscrit, notre affiliation est validée.
- **Configuration du paiement** : LowAbleAI rappelle au client de configurer Stripe/PayPal. Par exemple : « *Connectez votre compte Stripe pour accepter les cartes de crédit : [lien vers la page de configuration des paiements]* ». Idem pour PayPal. On explique brièvement comment faire (quelques captures d'écran ou instructions textuelles).
- **Tests** : On suggère au client de tester sa boutique – par exemple de passer une commande test en utilisant un code de réduction de 100% (Shopify permet de créer une commande brouillon ou de tester en mode sandbox). Cela permet de vérifier que le flux de commande -> DSers -> AliExpress fonctionne bien.
- **Conseils d'après-lancement** : éventuellement, LowAbleAI peut fournir quelques conseils automatisés, ex: « *Pensez à ajouter une adresse email professionnelle et à vérifier la page "Contactez-nous".* », « *Commencez à promouvoir votre produit via des pubs Facebook (non inclus dans le SaaS).* », etc.
- **Support** : Offrir un lien de support (peut-être vers une FAQ de LowAbleAI ou un contact) en cas de question.

L'infrastructure devra être **robuste et sans bug**. Chaque appel d'API sera vérifié (avec des boucles de retry si échec), chaque réponse d'IA filtrée (pour éviter du contenu inapproprié ou incohérent). On utilisera des logs et validations à chaque étape. Par exemple, après la génération de texte, on pourrait utiliser une seconde passe GPT pour vérifier l'orthographe et la cohérence. Après la création de la boutique, vérifier via l'API que le produit est bien présent, etc. L'expérience doit sembler fluide malgré la complexité en coulisse.

Enfin, place à ce que le client voit réellement : **les séquences de prompts** orchestrées par LowAbleAI.

Séquence des prompts et actions pas-à-pas pour LowAbleAI

Voici une **décomposition séquentielle** des interactions (prompts) et opérations que LowAbleAI réalisera pour générer la boutique graduellement. Chaque étape correspond soit à un prompt (question ou commande) soit à une action technique automatisée :

1. Prompt d'accueil – Choix de la langue :

2. *LowAbleAI*: « Bonjour et bienvenue sur LowAbleAI, le créateur de boutique intelligent. Pour commencer, veuillez choisir votre langue / Please select your preferred language. »
(Le client répond, par ex "Français".)

3. *LowAbleAI*: (désormais en français) « Parfait, nous continuerons en français. »

4. Prompt – Collecte des informations de base du projet :

5. *LowAbleAI*: « Afin de générer votre boutique Shopify de A à Z, j'ai besoin de quelques informations :

- a) **Lien du produit** à vendre (ou description détaillée si vous n'avez pas de lien) ;
 - b) **Nom souhaité de la boutique** (sinon je peux en suggérer un) ;
 - c) **Adresse e-mail** pour votre compte Shopify ;
 - d) **Devise et pays cible** (par ex. EUR – France) ;
 - e) Souhaitez-vous offrir la **livraison gratuite** à vos clients ? (Oui/Non) ;
 - f) Préférez-vous que je **définisse le prix automatiquement** avec une marge standard (~20-30%), ou avez-vous un prix de vente en tête ?
 - g) Avez-vous déjà un **compte Shopify** ou devons-nous en créer un nouveau pour vous ? Répondez en fournissant ces informations. »
- (Le client répond avec les données demandées. LowAbleAI valide qu'il ne manque rien, éventuellement repose des questions si c'est incomplet.)*

6. Action – Création de la boutique Shopify via API affiliée :

7. *LowAbleAI*: « Merci, je crée votre boutique Shopify... »

(En arrière-plan, LowAbleAI appelle l'API Shopify Partner pour créer le dev store, avec l'email du client, le nom de boutique, la langue/devise. Par exemple, POST /partners/{id}/stores.json avec affilié=true. Si cette étape nécessite une confirmation, LowAbleAI fournit un lien d'inscription affilié.)

8. *LowAbleAI*: « ✓ Boutique Shopify créée avec le nom "{NomBoutique}" . » (par exemple)

(Si c'est via lien : "Cliquez ici pour créer votre boutique Shopify et revenez une fois terminée.")

9. Action – Installation du thème et préparation du design :

10. *LowAbleAI*: « Je configure le design de votre boutique... »

(LowAbleAI ici utilise les outputs du Service 1. En coulisse, il appelle l'IA image : par ex en envoyant un prompt Stable Diffusion : « Generate a futuristic animated banner for product X... ». Il peut montrer une progression.)

11. *LowAbleAI* (optionnel visuel): « 📸 Génération de l'image de tête de votre site... » (Pourquoi pas afficher l'image générée directement une fois prête, pour feedback.)

12. *LowAbleAI*: « Je crée la mise en page initiale... »

(En coulisse, appelle GPT-4 : « Propose une structure de page d'accueil pour {NomBoutique} qui vend {Produit}, en français, ton enthousiaste... ». Reçoit une structure + textes.)

13. *LowAbleAI*: «  Rédaction du contenu marketing... »
(Affiche éventuellement quelques extraits de textes générés, ex: slogan, pour montrer ce qui se passe.)
14. *LowAbleAI*: « J'applique le thème visuel (couleurs, polices) et j'intègre les sections... »
(Via l'API Shopify, il importe le thème, configure les settings, insère textes et images. Publie le thème.)
15. *LowAbleAI*: «  Le design de votre site est prêt. » *(Il pourrait donner un lien de prévisualisation à ce stade pour que le client voie la page d'accueil par curiosité.)*

16. Action – Importation du produit et paramétrages e-commerce :

17. *LowAbleAI*: « J'importe votre produit sur la boutique et règle les derniers détails... »
(Background: si DSers dispo, API import product. Sinon, LowAbleAI scrape le lien produit. Crée le produit via Shopify API avec titre/desc/price/images.)
18. *LowAbleAI*: « Produit "{TitreProduit}" créé avec succès. Prix de vente fixé à {PrixVente} (coût fournisseur {PrixAchat} → marge approx. {Marge%}). »
(Explique s'il y a promo: "Prix original {CompareAtPrice}, actuellement en promotion {PrixVente}.)"
19. *LowAbleAI*: « J'installe l'application de gestion des commandes fournisseur (DSers)... »
(Si possible, auto-install via API, sinon:)
20. *LowAbleAI*: « Veuillez cliquer sur ce lien pour installer DSers (outil de dropshipping) sur votre boutique, puis connectez votre compte AliExpress dedans. »
(Attend éventuellement confirmation, ou continue si réussi.)
21. *LowAbleAI*: « Configuration des options de livraison : livraison gratuite activée pour vos clients. »
22. *LowAbleAI*: « Configuration des paiements : vous pourrez accepter les paiements une fois que vous aurez connecté votre compte Stripe/PayPal (étape suivante). »
(Ici LowAbleAI ne peut faire plus via API, donc annonce la prochaine étape manuelle.)
23. *LowAbleAI*: «  Votre produit est en ligne et la boutique est presque prête à prendre des commandes ! »

24. Prompt – Instructions finales pour paiement et lancement :

25. *LowAbleAI*: « Dernières étapes :
 - a) Connectez un fournisseur de paiement : par exemple, activez **Stripe** en allant dans *Paramètres > Paiements* sur Shopify et en suivant les instructions (ou PayPal en liant votre compte PayPal).
 - b) Vérifiez vos emails : vous avez reçu un email de Shopify pour confirmer la propriété de la boutique (acceptez le transfert de la boutique vers votre compte).
 - c) Après cela, choisissez un plan Shopify (vous avez 3 jours de gratuit, puis la commission affiliée nous sera attribuée).
 - d) Enfin, vous pouvez visiter votre site en ligne à l'adresse {NomBoutique}.myshopify.com (ou connecter un domaine personnalisé).Une fois ces étapes effectuées, votre boutique sera **totalement opérationnelle.** »
(Ce prompt final résume au client ce qu'il doit faire manuellement. LowAbleAI peut rester disponible pour répondre à des questions ou effectuer d'autres changements si le client demande.)

26. Conclusion – Proposer assistance ou modifications :

27. *LowAbleAI*: « Votre boutique est prête à recevoir des visiteurs. Souhaitez-vous que je fasse autre chose ou avez-vous des questions ? Je peux par exemple ajouter un autre produit, ou modifier un texte si vous le désirez. Merci d'avoir utilisé LowAbleAI ! »

Chaque prompt et action est exécuté dans l'ordre logique, en gérant les dépendances (on ne crée pas le produit avant d'avoir le store, etc.). Grâce à cette séquence bien définie, LowAbleAI peut générer la boutique petit à petit tout en assurant une interaction fluide avec le client, évitant les bugs et omissions.

Conclusion

En combinant ces services d'IA et l'automatisation de Shopify, *LowAbleAI* sera capable de créer une boutique Shopify complète en quelques minutes, là où cela prenait des jours manuellement. Le design sera original et **futuriste** (grâce à l'IA d'images), les textes seront **vendeurs** et adaptés (grâce à GPT-4), et la partie e-commerce sera **optimisée** (grâce à l'intégration de DSers, au pricing intelligent et aux bonnes pratiques incorporées). Surtout, le tout est conçu pour demander **le moins d'effort possible au client** : une poignée d'informations initiales suffisent, le reste est automatisé et intelligent. Ce SaaS offre donc un assistant virtuel ultra-complet qui **démocratise la création de boutiques en ligne** – il suffit d'une idée de produit pour lancer son commerce, le reste (design, contenu, logistique) est pris en charge par l'IA.

Les technologies et outils sélectionnés ont été choisis pour être les plus performants dans leur domaine en 2025 : génération d'images **haute qualité** sur mesure ^③, génération de sites par IA sans templates figés ^⑧, rédaction **optimisée SEO et conversion** ^⑭, sans oublier l'automatisation éprouvée de la logistique e-commerce via des intégrations comme **DSers** ^②. En suivant ce plan, LowAbleAI pourra offrir un service innovant qui construit des **boutiques Shopify prêtes à l'emploi** "beyond imagination, one prompt away" pour reprendre l'esprit de Dora AI ^{⑩ ⑪ ⑬ ⑯ ⑰}, et ainsi donner à ses utilisateurs une longueur d'avance dans le ecommerce de demain.

^① Shopify Affiliate Marketing Program | Earn income for your referrals - Shopify
<https://www.shopify.com/affiliates>

^② Ecommerce Integrations - DSers
<https://www.dsers.com/integration>

^③ stability-ai/stable-diffusion | Run with an API on Replicate
<https://replicate.com/stability-ai/stable-diffusion>

^④ ^⑤ ^⑨ Stability AI Image Models — Stability AI
<https://stability.ai/stable-image>

^⑥ ^⑦ ^⑧ ^⑩ ^⑪ ^{⑯ ⑰} Dora AI - Sites beyond imagination, one prompt away.
<https://www.dora.run/ai>

^⑩ ^⑪ ^{⑯ ⑰} Build Your Store - Free AI Shopify Store Builder
<https://www.buildyourstore.ai/>

^⑫ ^⑬ ^⑭ ^⑮ ^{⑯ ⑰} How to Use AI to Build a Website with Shopify: Complete Step-by-Step G - EComposer
<https://ecomposer.io/blogs/news/how-to-use-ai-to-build-a-website-with-shopify>

^{⑯ ⑰} ^⑱ ^⑲ ^⑳ How to use "dsers" using my own HTML website on a host? : r/dropship
https://www.reddit.com/r/dropship/comments/10bbi5r/how_to_use_dsers_using_my_own_html_website_on_a/

^㉑ ^㉔ Dropshipping Profit Margins: The Complete Guide for 2025
<https://www.sparkshipping.com/blog/dropshipping-margins-complete-guide>

^㉒ How to Price Dropshipping Products - Printful
<https://www.printful.com/blog/how-to-price-dropshipping-products>

23 Learn How To Price Dropshipping Products Like A Pro - AutoDS

<https://www.autods.com/blog/dropshipping-tips-strategies/how-to-price-dropshipping-products/>

25 Atlas: AI Store Builder - Atlas AI Store Builder | Shopify App Store

<https://apps.shopify.com/dropship>

26 DSers AliExpress Dropshipping Service

<https://www.dsers.com/integration/aliexpress-dropshipping-service>

28 How to Build and Transfer a Shopify Development Store (2024)

<https://www.shopify.com/partners/blog/how-to-build-and-transfer-a-development-store>

29 Creating client transfer stores - Shopify Help Center

<https://help.shopify.com/en/partners/manage-clients-stores/client-transfer-stores/create-client-transfer-stores>