

RAPPORT DE STAGE 2024



ETUDIANT

Johan JANIN
19 ans
MDS Lyon - B1B

ENTREPRISE

SecondStep
11 rue du Havre, Paris
[Second-step.fr](https://second-step.fr)

Table des matières

Table des matières	2
Note	2
Partie I : L'entreprise	3
Partie II : Activités professionnelles confiées par l'entreprise	5
Temporalité de l'objectif.....	5
Le projet.....	6
Objectifs Quantitatifs et Qualitatifs	6
Déroulement de l'Activité Professionnelle.....	6
Note complémentaire	11
Tâches annexes	13
Partie III : Résultats et analyse de l'expérience	23
Résultats du Projet Principal : Développement et Lancement du Site Web E-commerce	23
Résultats des Projets Annexes	24
Analyse de l'Expérience.....	24
Impact sur l'Entreprise	25
Si c'était à refaire ?.....	25
Conclusion	26
Remerciements	26
Annexe	27
Documents.....	27
Codes Sources et Dépôts	27

Note

En complément de ce rapport de stage j'ai chaque jour décrit ma journée sur un Notion, indiquant les missions que j'ai dû effectuer, des images pour chaque étape de l'avancement des projets que l'on m'a confiés, etc...

Vous pouvez retrouver le résultat de ce travail à [cette adresse](https://shorturl.at/8tBVP). (<https://shorturl.at/8tBVP>)

SECOND *step*

S econdStep est une entreprise française basée à Paris, structurée en société par actions simplifiée (SAS), ce qui lui offre une grande flexibilité de gestion et d'organisation.

Fondée en 2023 par Thomas Lorian, Megane Ragot et Constantin Kalogerakis, elle a pour mission de réduire les déchets dans l'industrie de la mode en redonnant vie à des baskets usagées. Cette initiative se situe au croisement de la mode, de l'écologie et de l'économie circulaire.

Le modèle d'affaires de SecondStep repose sur la collecte, l'authentification, et la rénovation de baskets usagées. Ces chaussures sont ensuite revendues à des prix compétitifs, souvent 40% moins chers que les articles neufs, ce qui les rend accessibles à une **clientèle soucieuse de l'environnement et de son budget**. La première année d'exploitation a montré une croissance prometteuse avec une augmentation continue du chiffre d'affaires trimestriel majoritairement générée par les ventes de baskets rénovées.

L'entreprise emploie une vingtaine de personnes entre 19 et 25 ans et favorise une structure organisationnelle plate (dite en râteau et non pyramidale) pour encourager la communication et la prise de décision rapide. Les co-fondateurs occupent des rôles clés : Thomas Lorian comme CEO, Megane Ragot comme COO, et Constantin Kalogerakis comme CTO. Les autres équipes sont dédiées à la rénovation des baskets, *au développement web*, au marketing et à la gestion de la clientèle.

Sur le marché de la mode durable, SecondStep se positionne comme un **acteur innovant et engagé**. L'entreprise se distingue par son offre unique de baskets remises à neuf, un segment qui gagne en popularité auprès des consommateurs cherchant des alternatives écologiques. Les concurrents de SecondStep incluent d'autres startups de mode durable et des entreprises traditionnelles adoptant des pratiques de recyclage et de revente (Vinted, sites de Sneakers, Wethenew, ...). Toutefois, la qualité des produits de SecondStep et sa transparence renforcent sa position concurrentielle.

En plus de ses activités principales, SecondStep collabore avec des associations et des initiatives locales pour promouvoir le recyclage et l'économie circulaire. L'entreprise utilise principalement des plateformes numériques pour atteindre une audience globale, minimisant ainsi son empreinte carbone. Dans le futur, la startup envisage d'élargir son offre en intégrant d'autres articles de mode durable, consolidant ainsi sa place sur le marché.

En synthèse, SecondStep est une entreprise dynamique et responsable, alliant mode et durabilité pour offrir des produits de qualité tout en contribuant à la protection de l'environnement.

Partie II : Activités professionnelles confiées par l'entreprise

Après avoir exploré en détail l'entreprise SecondStep, son modèle d'affaires innovant et son positionnement sur le marché de la mode durable, il est maintenant temps de se concentrer sur les activités professionnelles qui m'ont été confiées durant mon stage. Dans cette seconde partie, nous allons examiner les objectifs, le déroulement, et les diverses tâches que j'ai réalisées, en mettant en lumière les défis rencontrés et les solutions apportées.

Aujourd'hui, SecondStep n'est pas rentable pour Thomas et ses équipes. La cause de cela ? Un site web trop lent (environ 4 à 5 secondes de chargement à chaque changement de page) notamment lié au CMS utilisé (Wordpress) ainsi qu'aux nombreux plugins installés dessus.

De fait, ils ont l'intention de migrer leur backend de Wordpress vers Shopify et de coder le frontend sans CMS (on dit qu'on code le site en « headless »). C'est le cœur de ma mission : en collaboration avec le responsable du pôle web (Louis : stagiaire également qui intégrera l'équipe via son alternance), l'objectif est de développer le site web e-commerce en pure code avec intégration des informations de la boutique Shopify.

Temporalité de l'objectif

Pour Thomas, l'objectif est clair : pouvoir lancer ce site web le plus rapidement possible, idéalement début juin afin de lever le problème financier et atteindre les 300 commandes par semaines d'ici juillet 2024. Actuellement, ils ne reçoivent que 15 à 20 commandes par semaine, du fait du temps de chargement du site web, des problèmes d'UI/UX, etc....

De notre côté, lorsque Thomas nous a expliqué le projet, nous n'étions pas très optimistes quant à une version utilisable en juin, mais au fur et à mesure du projet, le lancement début juin devenait de plus en plus envisageable.

Dans le cadre de ce stage, j'ai été chargé de concevoir le site web e-commerce de l'entreprise, en le reliant à Shopify. Cette mission s'inscrit parfaitement dans les objectifs du stage, visant à appliquer et approfondir mes compétences en développement web, en particulier avec des technologies modernes et des frameworks comme Remix et React.

Le projet

Le projet principal consiste à créer un site web e-commerce performant et intuitif pour SecondStep, en utilisant Hydrogen, un framework basé sur Remix et React, spécialement conçu par Shopify pour être utilisé avec Shopify. J'ai travaillé en étroite collaboration avec Louis, responsable du pôle développement pour transformer les maquettes fournies par une agence en un site fonctionnel et optimisé.

Objectifs Quantitatifs et Qualitatifs

1. Quantitatifs

- Lancer le site dans un délai de deux mois.
- Assurer une intégration fluide avec Shopify pour gérer les produits, les commandes et les paiements (ainsi que les contenus variables du site).
- Optimiser le site pour un temps de chargement inférieur à 1,5 secondes (grand maximum).

2. Qualitatifs

- Créer une expérience utilisateur (UX) engageante et intuitive.
- Garantir une esthétique fidèle aux maquettes initiales.
- Assurer une compatibilité multi-plateformes (ordinateurs, smartphones, ...).

Déroulement de l'Activité Professionnelle

1. Contraintes

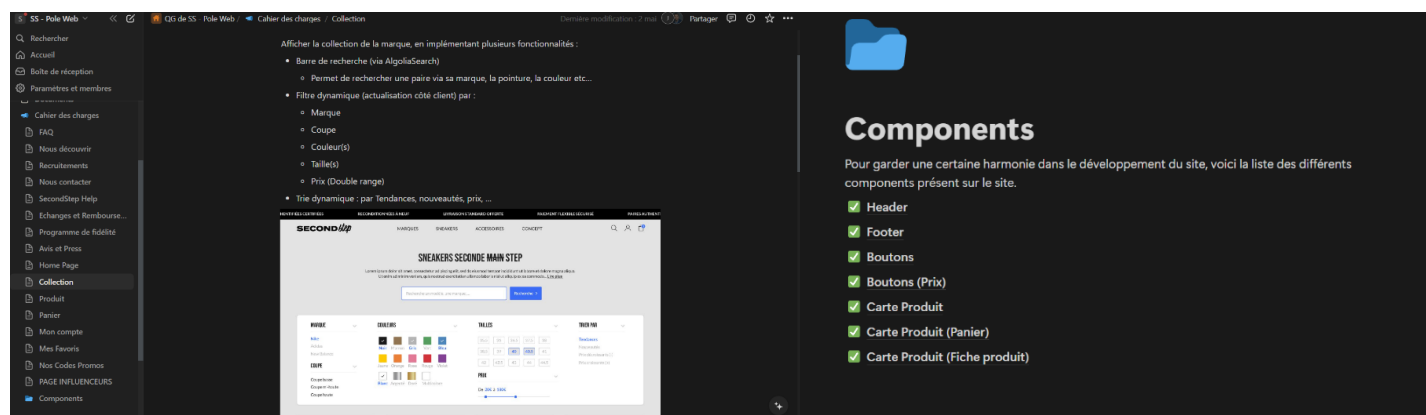
- Humaines : Coordination avec Louis et communication régulière avec l'agence de design pour garantir la fidélité aux maquettes.
- Matérielles : Accès aux outils de développement, environnements de test et ressources de Shopify.

2. Étapes du Projet

A. En amont :

- Réunion initiale pour définir les besoins et les attentes.

- Élaboration du cahier des charges via Notion, détaillant chaque fonctionnalité du site.



B. Pendant :

- Développement de l'architecture du site avec Hydrogen.
- Intégration des composants React pour les différentes sections du site.
- Tests réguliers pour vérifier la performance et l'UX.

C. En aval :

- Débogage et optimisation finale.
- Formation des membres de l'équipe à l'utilisation du site et des outils de gestion de Shopify.
- Lancement du site et suivi post-lancement pour assurer son bon fonctionnement.

3. Temps Nécessaire à l'Accomplissement

Le projet s'est déroulé sur toute la durée de mon stage, soit environ deux mois, avec une phase initiale de planification de deux semaines, suivi d'un mois et demi de développement intensif, puis deux semaines de tests et optimisations.

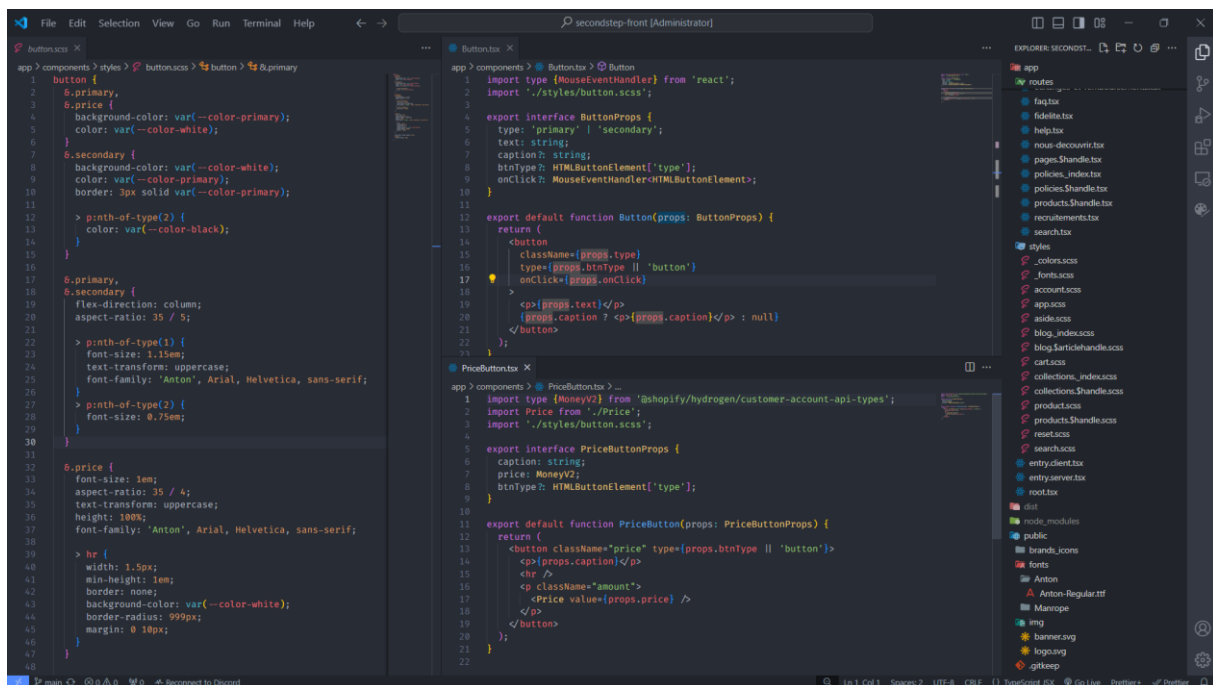
4. Mon Rôle

- Rédaction du cahier des charges sur Notion.
- Développement des fonctionnalités front-end avec React et Hydrogen.
- Intégration des APIs Shopify via GraphQL pour gérer les produits et les commandes.

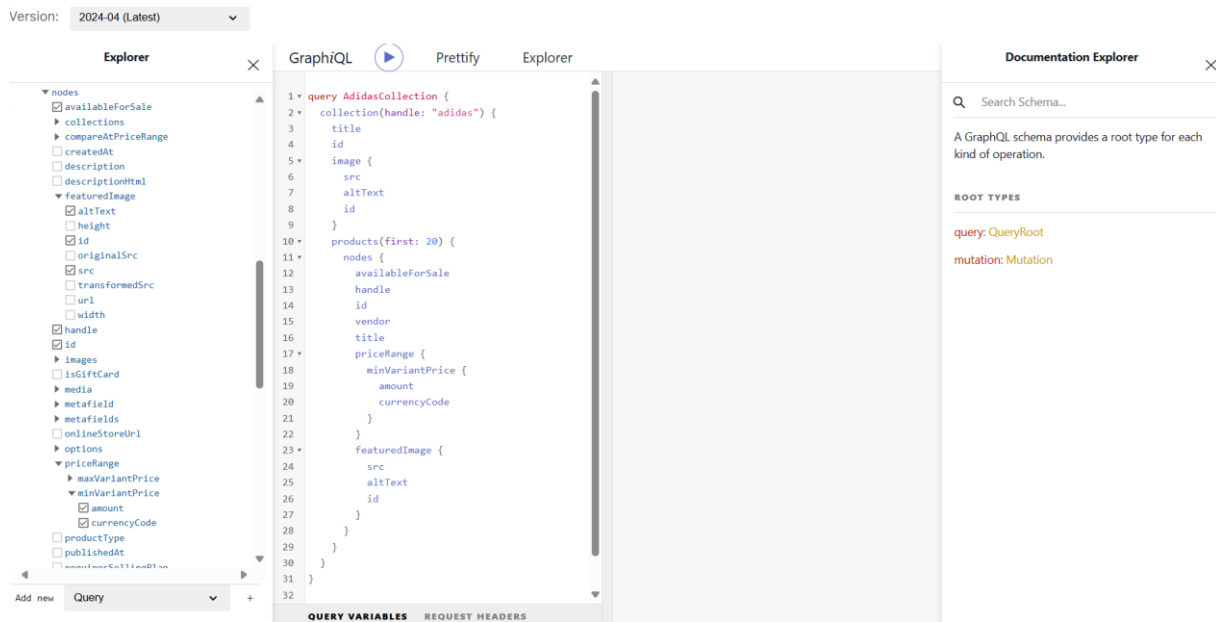
- Collaboration avec Louis pour résoudre les problèmes techniques et valider les étapes du projet.

5. Ressources Utilisées

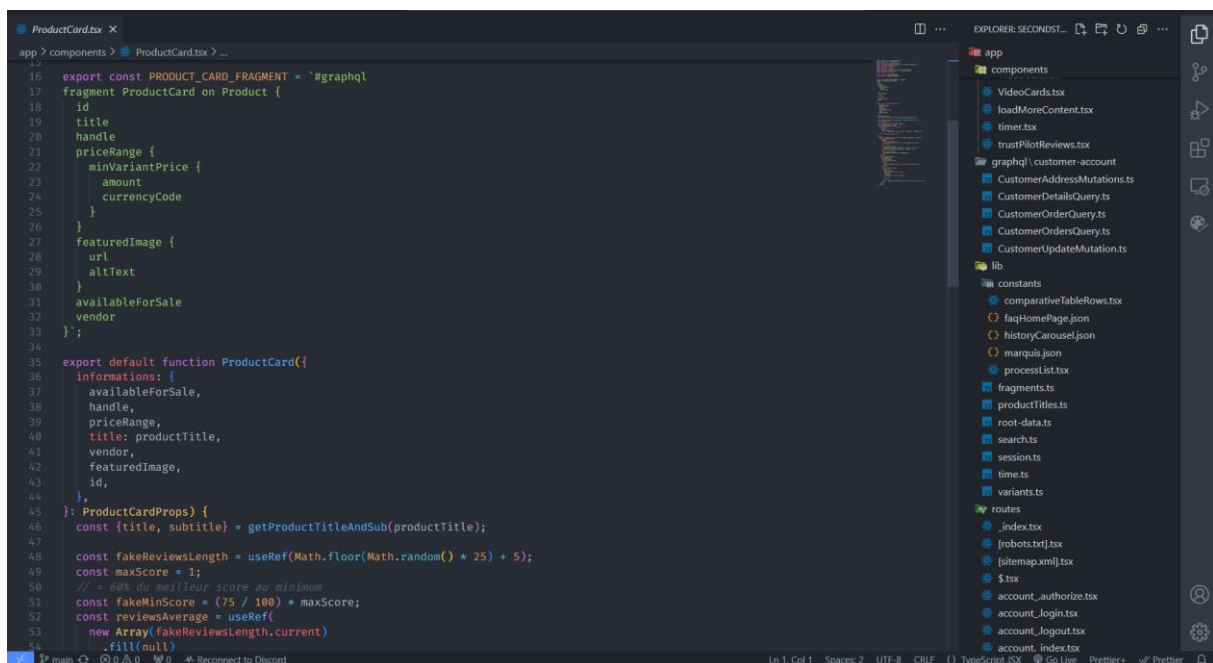
- Logiciels :
 - Remix : BFF (**B**ackend **F**or **F**rontend : un serveur créé uniquement pour le front-end du site web)
 - React : Composants UI



- SCSS : Compilateur CSS utilisé pour la mise en forme des différentes parties du site web
- Notion : Cahier des charges et feuille de route
- Shopify : Backend e-commerce (GraphQL pour récupérer les informations)



- Hydrogen : API d'interaction avec la boutique Shopify



- VSCode : IDE

- Matériel : Ordinateur personnel et double écran fourni par l'entreprise.
- Budget : Principalement alloué à l'achat de licences pour certains plugins et services tiers.

Pour mieux comprendre comment nous avons intégré les données de Shopify au site, voici un schéma représentatif de l'architecture que nous avons développé derrière la boutique de SecondStep :



AirTable est une sorte de tableur Excel avancé ayant la possibilité de créer des connexions avec à peu près n'importe quel service.

Make s'occupe de convertir et transférer les données AirTable vers une boutique Shopify.

Shopify nous sert de back-end afin de gérer les commandes, les clients et autres.

Hydrogen est le framework Shopify permettant d'interagir avec l'API de Shopify.

Le front-end est la partie que nous avons développée : le site qui s'occupe d'afficher les produits, gérer le panier du client, afficher les collections misent en avant...

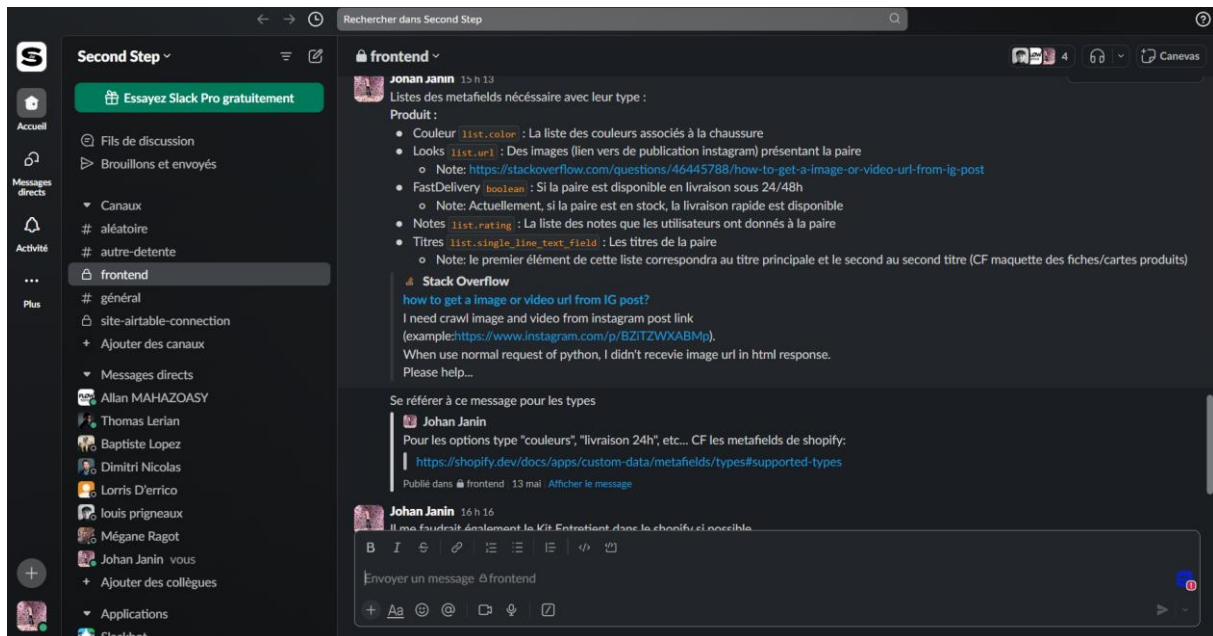
6. Formation Suivie

Une formation initiale sur l'utilisation de Hydrogen et Remix a duré environ une semaine.

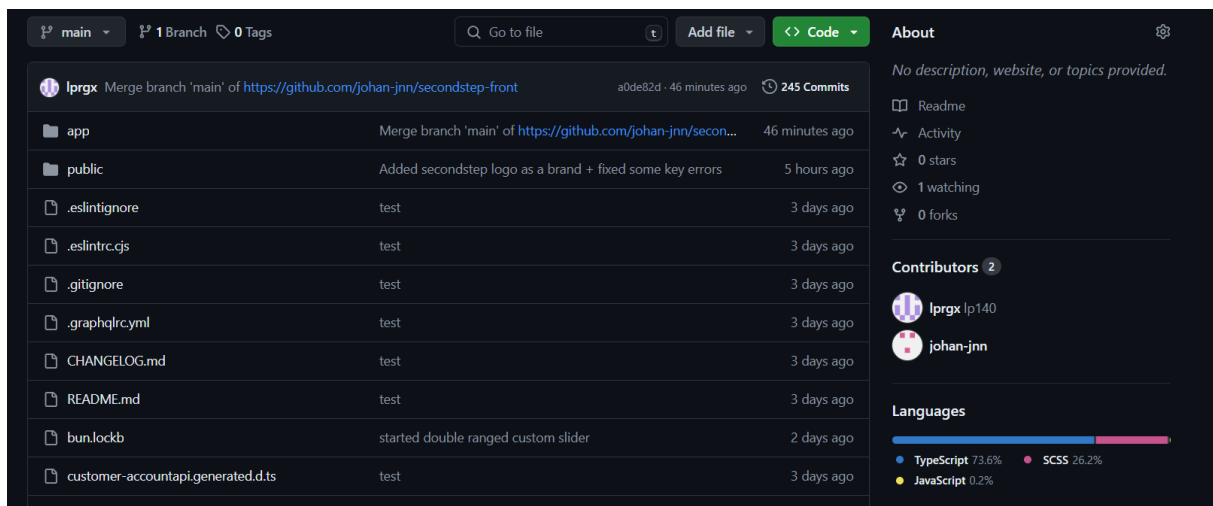
Cette formation s'est faite en autodidacte à force d'utiliser les composants mis à disposition par Hydrogen, d'aller rechercher des informations sur la documentation ou ~~encore~~ en cherchant des solutions à nos problèmes sur des forums, via des vidéos, ...

7. Méthodes de Travail

- Phoning et face-à-face : Pour les réunions d'équipe hebdomadaires et les sessions de résolution de problèmes.
- Utilisation de Notion : Pour la gestion de projet, le suivi des tâches, et la documentation.
- Utilisation de Slack : Pour communiquer rapidement et facilement avec les CEO, COO, CTO et les différents pôles de l'entreprise (marketing, design, achats, ...)



- Utilisation de Github : Pour la collaboration avec tout le pôle web sur le code du site web



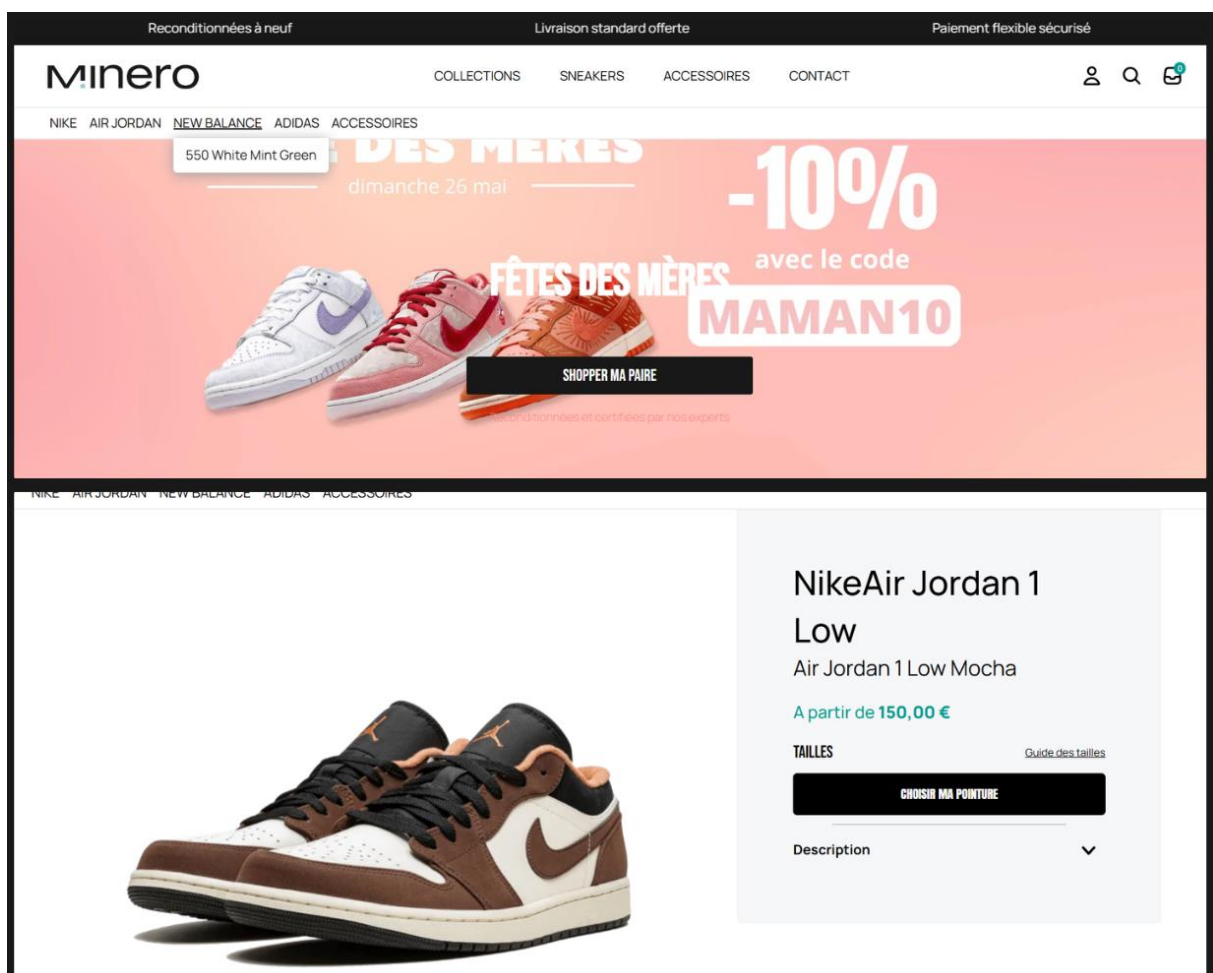
Note complémentaire

Le site web présenté ici est celui de SecondStep. Il est important de spécifier qu'en parallèle de nos travaux, Thomas a engagé une agence (nommée OOEO et réputée pour son expertise en termes de développement de site e-commerce) pour également travailler sur un site web plus propre. Nous avons commencé notre travail un peu après l'agence et avons terminé à peu près en même temps. Cependant, l'agence ayant bien plus d'expérience en termes de développement de site e-commerce, leur site était bien évidemment plus optimisé que le **nôtre** (ils ne sont pas partis sur une approche avec Hydrogen mais ont décidé d'utiliser leur

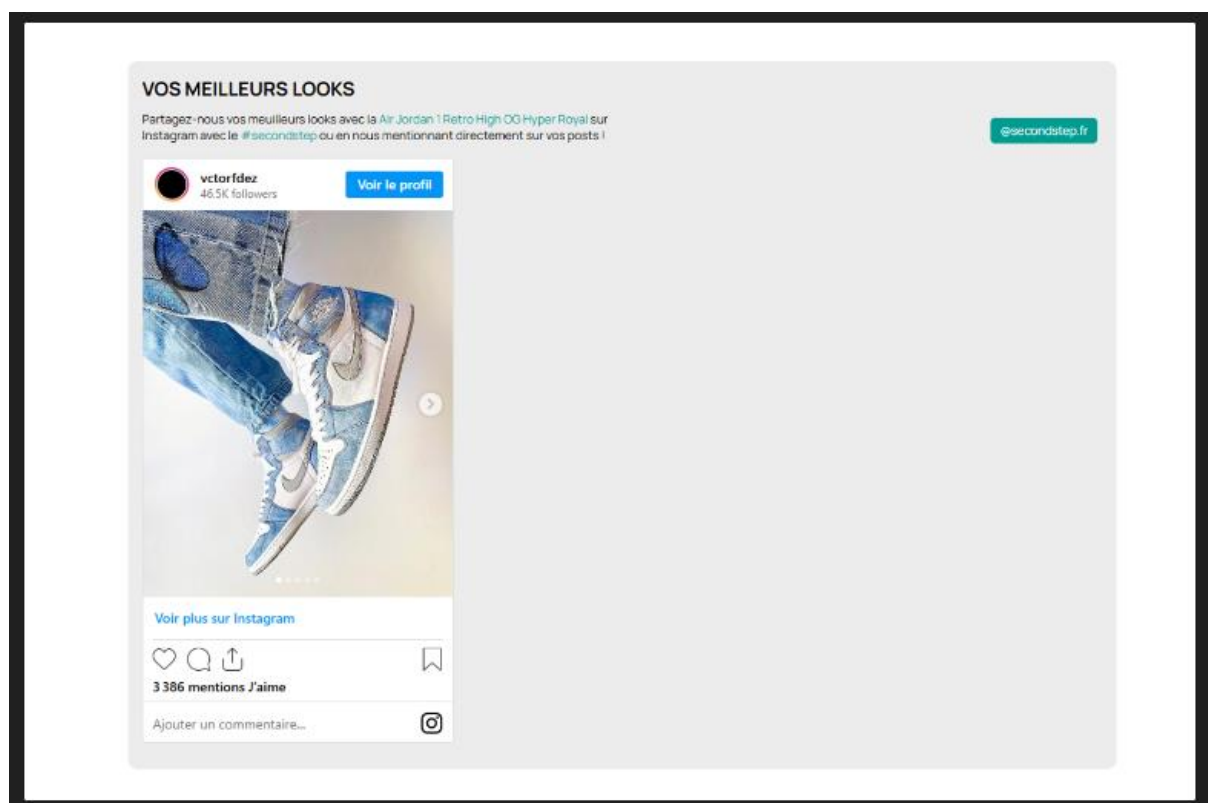
propre BBF (Backend-For-Frontend) avec Next.JS pour plus de flexibilité. Le site web de SecondStep est désormais celui créé par l'agence.

Mais pour ne pas faire de notre travail un travail inutile, Thomas nous a demandé de transformer notre travail pour le site web de Minero : une autre société lui appartenant qui est la jumelle de SecondStep à la seule différence que Minero vend des paires neuves.

Cela nous a pris environ 2 semaines afin de tout convertir pour un passer d'un SecondStep à un Minero, mais nous y sommes finalement bien arrivés (Thomas nous a d'ailleurs demandé de copier à l'identique le site proposé par l'agence) et avons pu le lancer quelques semaines plus tard :



Nous avons également réussi à créer une intégration Instagram pour chaque paire. C'est-à-dire que pour chaque paire, Thomas peut s'il le souhaite ajouter le lien d'un post à afficher sur la page de la paire afin de créer davantage de conversion :



Tâches annexes

En complément du site web e-commerce créé, j'ai réalisé des tâches annexes portant sur l'amélioration de la croissance de l'entreprise.

J'ai notamment créé une extension dont le but est de passer les musiques à un intervalle de temps non régulier sur Soundcloud. Thomas souhaite en effet améliorer les statistiques d'écoute de ses productions (il est DJ professionnel en dehors de SecondStep) et demande alors à ses employés de laisser tourner ses streams.

1. Soundcloud Track Skipper

Cette extension avait été dans un premier temps développée par Louis mais le code n'était absolument pas optimisé (il restait néanmoins fonctionnel). J'ai donc décidé de reprendre le développement de cette extension, et après avoir compris comment les extensions sous chrome fonctionnaient, j'ai commencé le développement en repartant de zéro.

J'ai commencé par rendre l'extension fonctionnelle avant de passer aux fonctionnalités annexes.

A. Côté Soundcloud

Pour ce qui est du fonctionnement de l'extension, je n'ai eu qu'à implémenter un bout de script qui s'exécute sur chaque page appartenant au domaine « soundcloud.com ». Ce script va s'occuper d'aller chercher le bouton « passer » du player de soundcloud et d'appuyer dessus comme une personne le ferait pour passer à la musique suivante, à chaque fois que la durée d'écoute dépasse la valeur aléatoire sélectionnée.

En parallèle, il a fallu vérifier que la musique qui est actuellement en train de jouer n'est pas une publicité. Pour cela j'ai remarqué un bandeau qui s'affichait lorsqu'une pub était en train de jouer. Il m'a donc fallu détecter à chaque changement de musique si ce bandeau était présent, et dans ce cas, désactiver l'extension pour la durée égale au temps de la publicité.

B. Le popup

Le popup est la partie de l'extension qui s'affiche lorsque vous cliquez sur l'icône de l'extension. J'ai avant toute chose commencé par créer un logo pour l'extension sur Illustrator, avec plusieurs variantes correspondant à chaque comportement de l'extension :



Dans l'ordre :

- Le logo de l'extension, avec le logo de Soundcloud et des flèches pour rappeler le rôle de celle-ci (passer les musiques automatiquement)
- L'icône affichée lorsque l'extension est active
- L'icône affichée lorsque l'utilisateur a mis en pause la lecture des musiques
- L'icône affichée lorsque l'extension est inactive du fait d'une publicité

J'ai ensuite voulu ajouter des paramètres de comportement à mon extension que j'ai affiché dans le popup afin que l'utilisateur puisse les modifier facilement :

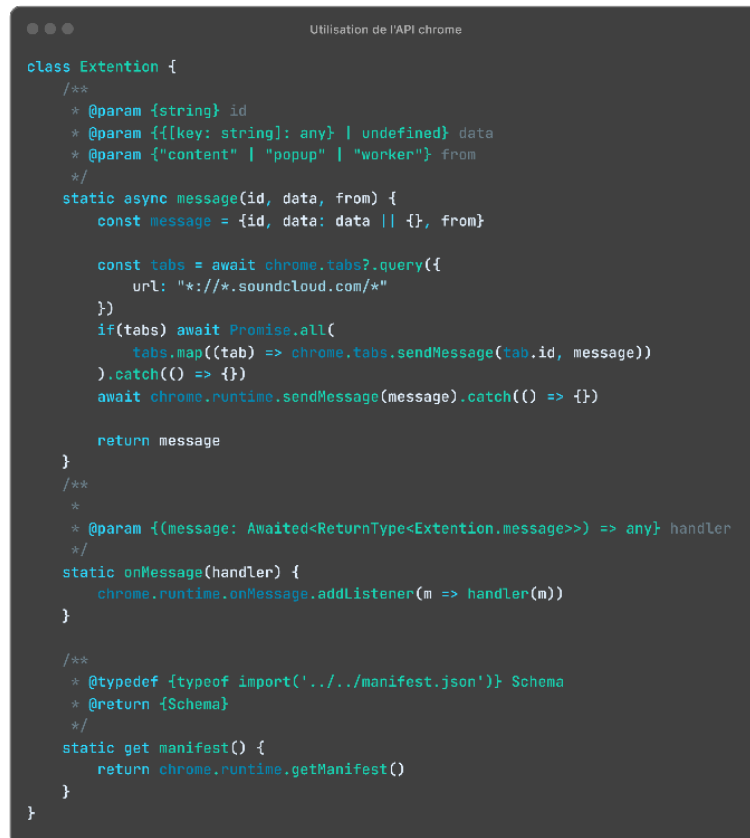


Voici à quoi correspondent les options :

- *Enable Auto Skipping* : Permet d'activer/désactiver le fonctionnement de l'extension
- *Start Playing On Launch* : Permet de démarrer le passage automatique des musiques dès que l'onglet de soundcloud est ouvert. Cette fonctionnalité peut ne pas marcher du fait de certains blocages que soundcloud met en place pour limiter l'utilisation de robots.
- *Skip Timeout* : Le temps laissé avant que la musique ne soit passé
- *Randomness* : Pour éviter d'avoir un intervalle régulier entre chaque passage de musique, ce paramètre permet d'ajouter plus ou moins de temps au paramètre *Skip Timeout*.

C. L'intégration des 2 parties

Maintenant que j'avais les deux parties de l'extension, à savoir la partie fonctionnement et la partie configuration, il a fallu par la suite relier ces deux parties afin d'avoir une extension pleinement fonctionnelle. Pour cela il a fallu envoyer des messages entre le script injecté dans la page soundcloud et le popup. J'ai alors utilisé l'API fourni par la variable « chrome » disponible dans l'exécution du code des extensions, afin d'envoyer des « messages » entre les différentes parties de l'extension :



```
Utilisation de l'API chrome

class Extention {
  /**
   * @param {string} id
   * @param {[{key: string}: any] | undefined} data
   * @param {"content" | "popup" | "worker"} from
   */
  static async message(id, data, from) {
    const message = {id, data: data || {}, from}

    const tabs = await chrome.tabs?.query({
      url: "*/**/*.soundcloud.com/*"
    })
    if(tabs) await Promise.all(
      tabs.map((tab) => chrome.tabs.sendMessage(tab.id, message))
    ).catch(() => {})
    await chrome.runtime.sendMessage(message).catch(() => {})

    return message
  }
  /**
   *
   * @param {(message: Awaited<ReturnType<Extention.message>>) => any} handler
   */
  static onMessage(handler) {
    chrome.runtime.onMessage.addListener(m => handler(m))
  }
  /**
   * @typedef {typeof import('../manifest.json')} Schema
   * @return {Schema}
   */
  static get manifest() {
    return chrome.runtime.getManifest()
  }
}
```

Finalement, l'extension était enfin fonctionnelle à la grande satisfaction de Thomas.

D. Remerciements

Je remercie Louis de m'avoir fait confiance en me confiant ce projet sur lequel j'ai découvert le monde des extensions pour navigateur. Peu de temps après avoir fini ma mission j'ai voulu créer d'autres extensions et j'ai eu l'idée de créer un SAAS (Software-As-A-Service – proposer un service et essayer de le faire rémunérer) basé sur une extension. Je suis toujours actif sur cet objectif mais je souhaitais simplement dire que c'est grâce à ce projet que je me suis lancé.

2. Robot Discord Vente de paires

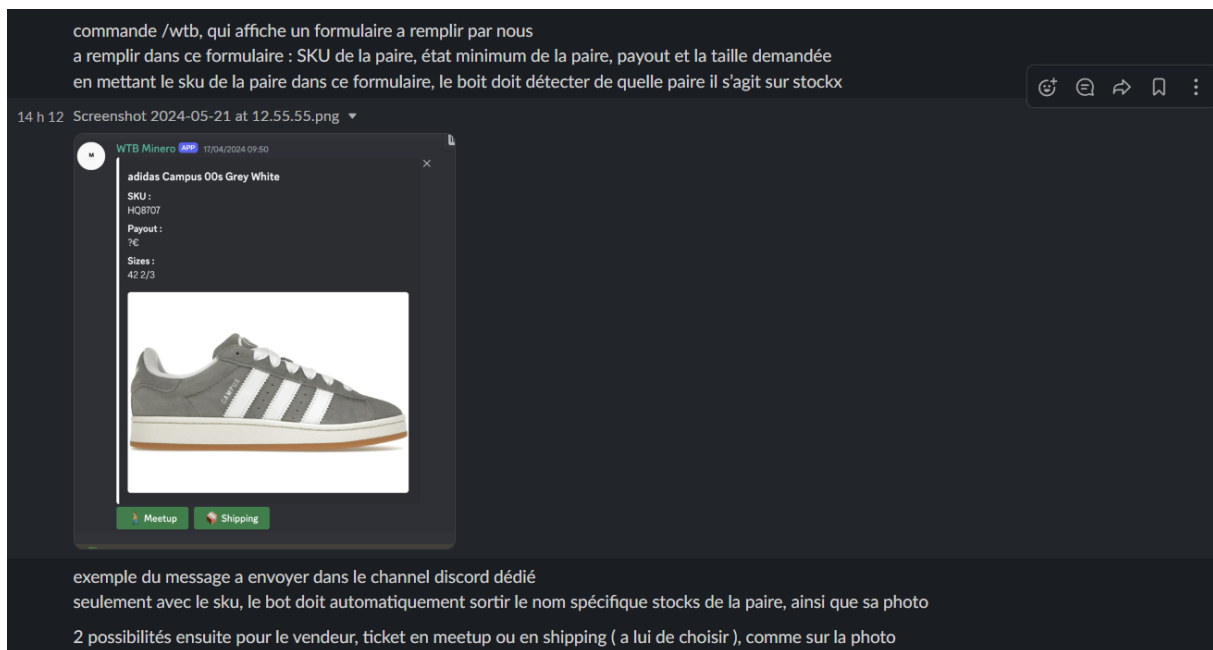
En second projet, j'ai eu à créer un robot Discord qui s'occupe :

1. D'afficher les commandes (appelées WTB) que SecondStep voudrait passer (donc les paires recherchées par l'entreprise)
2. De gérer les potentiels revendeurs à travers différents canaux.

A. Cahier des charges

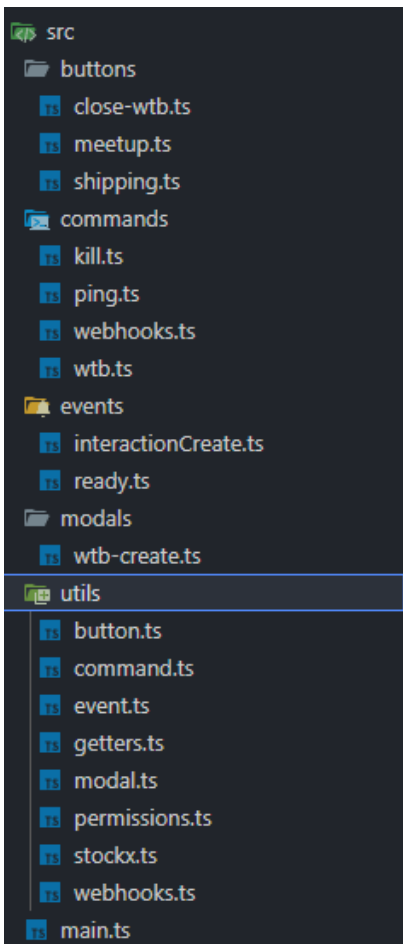
Après m'avoir demandé si ce projet était possible, ce à quoi j'ai répondu oui, sachant que j'avais déjà à titre personnel conçu des robots Discord : c'est d'ailleurs par ce biais que j'ai commencé la programmation.

On m'a alors fourni le cahier des charges suivant :



J'ai alors commencé le développement du robot. Vous pouvez d'ailleurs consulter l'avancée jour par jour sur le Notion dont l'adresse figure en page 2 de ce document.

B. Mise en place des différents composants du robot



Après avoir reçu ce cahier des charges, il a fallu que j'anticipe l'organisation des fichiers du robot afin de pouvoir convenablement développer par la suite les différents comportements de celui-ci.

J'ai opté pour un système à dossier, dont chaque dossier serait nommé avec le nom de son comportement (commandes, évènements, boutons, ...) et à l'intérieur les différents fichiers avec les différents comportements.

En complément j'ai également créé un dossier « utils » où j'ai placé les différents fichiers dont le but est d'améliorer la simplicité du code (c'est-à-dire dossier à but utilitaire).

Aussi j'ai décidé pour ce projet d'utiliser Typescript afin de ne pas me perdre dans le code et d'avoir la possibilité de typer toutes les parties de mon programme.

C. Utilisation des classes

J'ai décidé ensuite de créer des classes pour chaque type d'interaction afin d'améliorer le temps de développement et d'avoir une certaine harmonie. Par exemple pour un bouton :

```
import {
  APIButtonComponent,
  ButtonBuilder,
  ButtonInteraction,
} from "discord.js";
import WTBClient from "../main.js";
import path from "node:path";

export default class BotButton {
  readonly informations: APIButtonComponent;
  constructor(
    __filename: string,
    readonly client: WTBClient,
    button: ButtonBuilder
  ) {
    this.informations = button
      .setCustomId(path.parse(__filename).name)
      .toJSON();
  }

  get component() {
    return new ButtonBuilder(this.informations);
  }

  async onClick(interaction: ButtonInteraction): Promise<any> {}
}
```

Ici vous avez la classe utilitaire, avec le constructeur qui s'occupe d'assigner directement l'identifiant du bouton au nom du fichier afin de mieux se retrouver, ainsi que la méthode `onClick()` qui sera appelée à chaque fois que ce bouton sera cliqué (le premier argument de cette méthode est d'ailleurs un objet contenant les informations de l'interaction avec ce bouton).

Et voici le composant d'un bouton, qui correspond au bouton « shipping » : celui-ci se trouve en bas d'un WTB et permet d'ouvrir un ticket en mode « shipping » (c'est-à-dire en envoyant sa paire via colis à SecondStep).

Vous remarquerez que cette classe hérite de la classe BotButton ce qui permet de pouvoir obtenir toutes les méthodes de la classe utilitaire. Dans le constructeur je m'occupe de donner à la classe parent le nom du fichier, le client (qui contient les informations du robot) ainsi que le bouton avec sa couleur, son emoji, son label, etc....

Enfin, lorsqu'il est pressé je m'occupe d'aller chercher la fonction lié à ce bouton et de l'exécuter.

```
shipping.ts

import {
  ButtonBuilder,
  ButtonInteraction,
  ButtonStyle,
} from "discord.js";
import WTBClient from "../main.js";
import BotButton from "../utils/button.js";
import { getErrorMessage } from "../utils/getters.js";

export default class Shipping extends BotButton {
  constructor(client: WTBClient) {
    super(
      import.meta.filename,
      client,
      new ButtonBuilder()
        .setLabel("Shipping")
        .setStyle(ButtonStyle.Primary)
        .setEmoji("📦")
        .setLabel("Shipping")
    );
  }

  async onClick(interaction: ButtonInteraction): Promise<any> {
    // It's the same behavior that the meetup system
    const meetup = this.client.getComponent("buttons", "meetup");
    if (!meetup)
      return interaction.reply({
        content: getErrorMessage(
          "Je n'ai pas trouvé la fonction pour ce bouton."
        ),
        ephemeral: true,
      });
    const { onClick } = meetup;
    return onClick.bind(this)(interaction);
  }
}
```

D. Récupération des données

Selon le cahier des charges, à partir du SKU donné dans le formulaire il faut réussir à récupérer les informations de la paire depuis l'API de StockX. J'ai d'abord recherché depuis le portail développeur comment y accéder, et après avoir envoyé un formulaire pour me faire valider mon compte en tant que compte de développeur, mais n'ayant toujours pas obtenu de réponse 2 semaines plus tard, j'ai décidé d'utiliser une librairie qui va Scrapper le site web de StockX.

Le scrapping n'étant pas une méthode exacte (il peut y avoir des erreurs, des données non trouvées, etc...), j'ai implémenté un algorithme qui s'occupe d'aller chercher les données, et si les données n'ont pas été trouvées, d'aller rechercher les données une nouvelle fois. Il répète cela un nombre défini de fois, et après ce nombre d'essais atteint, s'il n'a toujours pas trouvé les données, il renvoie la valeur null. Pour cela j'ai utilisé la technologie des fonctions récursives :

```

stockx.ts

import { StockxClient, StockxProduct } from "stockx-scraper";

const client = new StockxClient({
  countryCode: "FR",
  currencyCode: "EUR",
});

export async function getProduct(
  sku: string,
  ttl = 3
): Promise<StockxProduct | null> {
  const res =
    (await client
      .search({
        query: sku,
      })
      .then((arr) =>
        arr.length ? arr[0] : "Empty returned product list."
      )
      .catch((err) => "" + err)) || null;
  if (typeof res === "string") {
    console.error(
      `---\nImpossible trouver le produit associé au SKU "${sku}":\n${res}\nNombre d'essais
      restant : ${ttl}.\n---`
    );
    if (ttl > 0) return getProduct(sku, ttl - 1);
    else return null;
  } else return res;
}

```

E. Envoie du WTB

Afin d'envoyer un nouveau WTB, il faut remplir au préalable un formulaire pour aider à la récupération des informations de la paire. Pour cela, il suffit d'envoyer la commande /wtb dans le salon discord où le modérateur souhaite envoyer le WTB (ou spécifier le salon dans les paramètres de la commande), puis remplir le formulaire qui s'affiche à la suite de l'envoi de la commande.

OPTIONS

channel

Le salon où s'enverra le message (par défaut celui où a été exécuté la commande).

/wtb

Poster un nouveau WTB

S

/wtb | +1 option

Créer un nouveau WTB

⚠ Ce formulaire sera transmis à Sell To SecondStep. Ne donne pas de mot de passe ni toute autre information sensible.

SKU STOCKX

ETAT MINIMUM (RAMENÉ SUR 10)

'8 / 10', '17 / 20',... - Sur 10 par défaut.

PAYOUT

ex. '15', '25,6€', '95€', '230.85€', ...

TAILLES SOUHAITÉES

Séparer les différentes tailles souhaité par des virgules.

INFORMATIONS SUPPLÉMENTAIRES

Ecrivez ici les éventuelles informations supplémentaires sur la paire recherchée.

4000

Annuler

Envoyer

Le *SKU StockX* est un identifiant lié au site <https://stockx.com> qui permet d'identifier la paire.

L'*état minimum* de la paire est une note (dont le contenu va être parsé afin de toujours être ramené sur 10).

Le *payout* est le prix proposé par SecondStep (également parsé afin de toujours être dans un format à virgule en euro).

Les *Tailles souhaitées* sont les tailles que SecondStep recherche (les valeurs sont à séparer par des virgules afin d'avoir une meilleure mise au propre)

Il y a également la possibilité d'ajouter des *Informations supplémentaires* si besoin.

Après avoir rempli ce formulaire, le WTB s'envoie avec les informations de la paire dans le salon défini :

Sell To SecondStep APP 13/06/2024 11:28

Jordan 1 Retro High OG Seafoam (Women's)
SKU :
Etat Minimum :
Payout :

CD0461-002
7.0 / 10
?

Tailles :

* 38

13/06/2024 11:28

Meetup
 Shipping
 Fermer la recherche

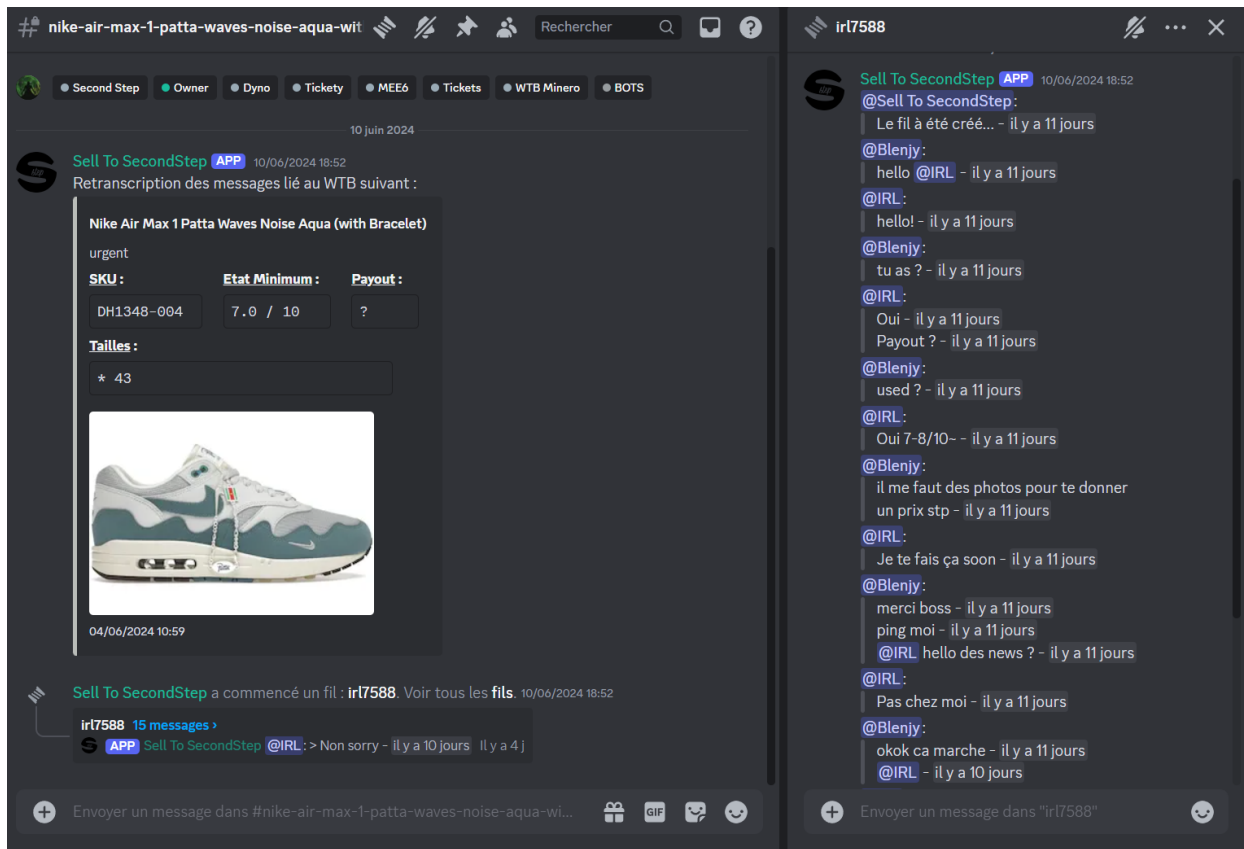
Informations :

- Le titre de la paire est cliquable et renvoie sur la page StockX du produit
- La couleur de l'encadré (à gauche) se fait en fonction de la couleur primaire de la photo de la paire (le fond blanc est exclu)
- Le bouton « Fermer la recherche » n'est cliquable que pour les modérateurs

F. Retranscription des messages

Une autre fonctionnalité demandée est la retranscription des messages. En effet, après qu'un modérateur ait fermé un WTB, il faut que les messages des différents canaux de ventes ouverts soient retranscrits dans une catégorie.

J'ai donc fait en sorte de créer un salon dans cette catégorie à la fermeture du WTB, d'aller récupérer chaque message de chaque canal ouvert lié à ce WTB et de retranscrire cela dans les threads du salon créé :



Partie III : Résultats et analyse de l'expérience

Après avoir décrit en détail les activités professionnelles confiées par l'entreprise, ainsi que les objectifs et le déroulement de ces projets, il est temps d'évaluer les résultats obtenus et d'analyser l'impact de ces expériences. Dans cette troisième partie, nous allons examiner les résultats concrets des projets réalisés, tant en termes de performances et de fonctionnalités que d'objectifs atteints, avant de conclure par une réflexion sur l'ensemble de cette expérience de stage.

Résultats du Projet Principal : Développement et Lancement du Site Web E-commerce

1. Performance et Fonctionnalité

A. Temps de Chargement :

Le site web développé en utilisant Hydrogen et React a atteint un temps de chargement moyen de 1,2 secondes, dépassant ainsi l'objectif initial de 1,5 secondes. Cette amélioration significative a contribué à une meilleure expérience utilisateur et une réduction du taux de rebond.

B. Intégration avec Shopify :

L'intégration avec Shopify s'est effectuée sans accroc majeur. Les produits, les commandes et les paiements sont désormais gérés de manière fluide, avec une interface utilisateur intuitive pour les administrateurs de SecondStep.

C. Compatibilité Multi-plateformes :

Le site a été testé et optimisé pour une utilisation sur divers appareils, y compris ordinateurs de bureau, tablettes et smartphones. Cela a assuré une expérience utilisateur cohérente et agréable quel que soit le dispositif utilisé.

2. Objectifs Quantitatifs Atteints

A. Délai de Lancement :

Le site a été lancé dans le délai prévu de deux mois. La phase de tests et d'optimisation finale a permis de déboguer et de peaufiner les fonctionnalités avant le lancement officiel.

B. Augmentation des Commandes :

Dès les premières semaines après le lancement, le nombre de commandes hebdomadaires a significativement augmenté, atteignant en moyenne 250 commandes par semaine à la fin de

mon stage. Cette croissance rapide a démontré l'impact positif du nouveau site web sur les performances commerciales de SecondStep.

3. Objectifs Qualitatifs Atteints

A. Expérience Utilisateur :

Le retour des utilisateurs a été largement positif, avec des commentaires soulignant la fluidité de la navigation et la facilité d'utilisation du site. Les maquettes initiales ont été respectées, assurant une esthétique moderne et professionnelle.

B. Formation de l'Équipe :

Une formation complète a été dispensée aux membres de l'équipe, leur permettant de gérer efficacement le nouveau site web et les outils associés.

Résultats des Projets Annexes

1. Extension Soundcloud Track Skipper

L'extension développée pour Soundcloud fonctionne de manière efficace, automatisant le passage des musiques et s'adaptant aux publicités. Cette solution a aidé Thomas à améliorer les statistiques d'écoute de ses productions musicales, sans pour autant que Soundcloud ne le détecte.

2. Robot Discord Vente de Paires

Le robot Discord a automatisé la gestion des commandes et des revendeurs, simplifiant ainsi les processus de communication interne et améliorant la réactivité de l'équipe.

Analyse de l'Expérience

1. Apprentissage et Développement Personnel

A. Compétences Techniques :

Ce stage m'a permis de renforcer mes compétences en développement web, en particulier avec des technologies modernes comme React, Remix et Hydrogen. J'ai également acquis une expérience précieuse en intégration de systèmes e-commerce avec Shopify. Cela vaut également en termes de design CSS (SCSS – SASS) : les nombreuses animations/composants du site e-commerce ont nécessité une connaissance ~~pointilleuse~~ **pointue** quant à l'utilisation et à l'écriture du style.

B. Gestion de Projet :

La gestion du projet à travers des outils comme Notion, Slack et Github m'a appris à organiser efficacement les tâches, à collaborer avec les membres de l'équipe et à suivre les progrès de manière structurée.

C. Problème et Résolution :

Travailler sur des projets annexes tels que l'extension Soundcloud et le robot Discord m'a aidé à développer mes compétences en résolution de problèmes et à aborder des défis techniques variés avec créativité et rigueur.

Impact sur l'Entreprise

1. Croissance et Rentabilité :

Les solutions développées ont directement contribué à la croissance de SecondStep (Minero), en augmentant les commandes et en améliorant l'expérience utilisateur. Cela a ~~mis~~ relancé l'entreprise sur la voie de la rentabilité.

2. Amélioration des Processus :

Les outils et automatisations mis en place ont simplifié les opérations quotidiennes et ont permis à l'équipe de se concentrer sur des tâches à plus forte valeur ajoutée.

Si c'était à refaire ?

Sans hésitation, je referai avec plaisir et intérêt un stage comme celui-ci. En effet, les 2 mois passés dans cette entreprise jeune et dynamique correspondent parfaitement à la manière de travailler et à l'ambiance dans laquelle je souhaite orienter mes futures recherches d'emploi. Dans les technologies utilisées, ne pas travailler sous Wordpress mais toucher réellement au code du contenu/style du site fut un réel défi et une opportunité que je qualifierai d'exceptionnelle surtout pour un étudiant en première année.

Conclusion

En conclusion, ce stage chez SecondStep a été une expérience à la fois enrichissante et formatrice. J'ai eu l'opportunité de travailler sur des projets variés et stimulants, allant du développement d'un site web e-commerce performant à la création d'outils d'automatisation sur mesure. Cette immersion dans le monde du développement web en entreprise a confirmé ma passion pour ce domaine et mon désir de poursuivre mes études dans cette voie. Les compétences techniques et les expériences acquises seront sans aucun doute des atouts précieux pour mes futurs projets professionnels et personnels.

Remerciements

Je tiens à remercier Thomas Lorian et Mégane Ragot de m'avoir accepté dans cette aventure et de m'avoir contacté sur LinkedIn en vue de la réalisation de ce stage, de m'avoir fait confiance dans mon travail et dans mes compétences à la réalisation de celui-ci. Cela m'a permis d'avoir un stage dans le développement web dès ma première année d'étude ce qui n'est pas si courant.

Au-delà de l'aspect purement technique de ce stage, il aura été pour moi l'occasion de découvrir le monde de l'entreprise dans ce domaine d'activité, d'intégrer une équipe jeune, dynamique et passionnée et de participer modestement au développement d'une Start Up.

Johan JANIN

Annexe

Documents

1. *Cahier des Charges*

- Document détaillant les besoins et attentes pour le projet de développement du site web e-commerce.
- Inclus les spécifications fonctionnelles et techniques, les objectifs de performance, et les maquettes du design.
- Note : Le notion étant privé, vous trouverez ci-joint un lien contenant les pages du Notion :
- <https://secondstep-cdc-2024.tiiny.site/>

2. *Documentations sur Hydrogen, Remix, GraphQL, ...*

A. API Shopify

Hydrogen : <https://shopify.dev/docs/api/hydrogen>

API Storefront (données accessibles via GraphQL) : <https://shopify.dev/docs/api/storefront>

B. Style

SASS (langage compilé en CSS) : <https://sass-lang.com/guide/>

C. Développement algorithmique

React : <https://fr.react.dev/>

Remix : <https://remix.run/>

D. Interaction avec Shopify

GraphQL : <https://graphql.org/>

Playground Shopify/GraphQL : <https://shopify.dev/graphiql/storefront-graphiql>

Codes Sources et Dépôts

1. *Dépôt GitHub pour le Site Web*

<https://github.com/johan-inn/secondstep-front/>

Contient le code source du site web e-commerce développé, incluant les composants React, les intégrations Shopify, et les fichiers de style SCSS.

Note : ce repository contient une version non terminée du site de Minéro (le nouveau repository étant privé à SecondStep).

2. *Dépôt GitHub pour l'Extension Soundcloud Track Skipper*

<https://github.com/lprgx/SoundCloud-Extension-For-Playlist>

Contient le code source de l'extension Soundcloud, incluant le script de passage automatique des musiques et le popup de configuration.

3. *Dépôt GitHub pour le Robot Discord*

<https://github.com/johan-inn/secondstep-discordbot>

Contient le code source du robot Discord, incluant les commandes, événements, et l'intégration avec l'API de StockX.