

Neolait

Optimisation des Ventes par Intelligence Artificielle

Cahier des charges

Alexandre CHOLAT
20/06/2024

Table des matières

Table des matières.....	1
Introduction.....	3
Contexte	3
Objectifs.....	3
Description du Besoin.....	3
Problèmes Actuels.....	3
Besoins fonctionnels.....	4
Besoins non-fonctionnelles	4
Spécifications Fonctionnelles	5
Fonctionnalités	5
Cas d'utilisation	5
Spécifications Techniques.....	6
Technologies, services et outils de Fonctionnement.....	6
Schéma du système	6
Programmation.....	6
Ajout de Capacités Avancées.....	7
Paramétrage et Prompting	7
Création d'une Action GPT	8
Troubleshooting.....	9
Contraintes Techniques	10
Critères de Qualité.....	11
Normes et Standards.....	11
Tests et Validation.....	11
Budget et Ressources	12
Estimation des couts d'Opération	12
Ressources humaines nécessaires.....	12
Estimation de la Valeur du Produit.....	12
Impacte du Nouveau Produit sur les Économies de Temps	12
Impact Additionnel sur l'EBIT et la Commission.....	12
Impact Total	12
Impact Net sur l'EBIT	13
Visualisation	13
Limitations et Ouvertures.....	14
Protection des Données	14
Vendor Lock-In Effect.....	14

Renouvellement et Capacité Maximale des Données.....	14
Hallucinations.....	14
Autres Limitations.....	15
Fonctionnalités à Envisager	15
Une note sur l'évolution des technologies du domaine.....	15
Annexes	16
References	16
Instructions pour GPT "Neolait Sales Assistant V3"	17

Introduction

Contexte

Actuellement, Neolait cherche à optimiser ses processus de vente et de relation client en intégrant des modèles d'intelligence artificielle (IA) pour assister les CTC. L'émergence de l'IA dans l'industrie offre une opportunité de simplifier radicalement l'interface humain-machine, augmenter le volume et qualité d'échange de l'information, et drastiquement augmenter la productivité. Ce projet a pour objectif d'identifier une solution permettant d'appliquer cet avancement technologique au bénéfice des CTC de Neolait. Il s'agit d'une solution permettant à une IA de fournir des réponses en intégrant les données de Neolait, l'outil sera donc un assistant informatique disponible sur téléphone mobile, et ne remplacera pas les CTC. Ce projet ne comprend pas des outils destinés à la production ou administration, se limitant uniquement à l'assistance de ventes.

Objectifs

L'outil informatique a pour objectifs principaux de faciliter l'accès aux données de clients, produits, ventes, et stocks par interrogation en langage naturel, augmentant l'efficacité des CTC et la satisfaction et achats des clients. Les objectifs spécifiques sont les suivants :

Simplifier l'accès aux données

- **Détails** : Centraliser l'accès aux données sur une application mobile, et permettre aux CTC de demander et recevoir toutes informations en langage naturelle.
- **Impact** : Gagner en efficacité et augmenter la productivité des CTC.

Mieux cibler les clients

- **Détails** : Rendre disponible des outils d'analyse et comparaisons pour guider les stratégies de ventes et recommandation de produits.
- **Impact** : Augmenter le volume de ventes et le taux de rétention clients.

Description du Besoin

Problèmes Actuels

Accès aux Données : Les CTC doivent chercher séparément les informations pertinentes sur les clients, les produits, les ventes et les stocks. Ils doivent consulter les données sur un ordinateur portable, ce qui est encombrant et rend impossible une recherche hors environnement sec, stable, et connectée.

Temps de Recherche : Une grande partie du temps des CTC est consacrée à la recherche d'informations, réduisant ainsi le temps disponible pour les interactions avec les clients. Les outils actuels ne sont ni intuitifs, ni ergonomiques, et demandent une haute attention cognitive.

Analyse de Données : Les CTC ne disposent pas d'outil d'analyse de tendances et de comparaison pratique sur le terrain. Ils ne disposent pas d'outil pour répondre aux questions spécifiques de produits en temps réel. Les CTC passent trop de temps à planifier leur itinéraires et visites en fonction de leurs prédictions de stocks de clients.

Collecte d'Informations Clients : Le processus de collecte des informations sur les clients est inefficace et manque de précision, rendant difficile l'identification des tendances et des opportunités de vente du a un manque de quantité d'information.

Communication et Collaboration : Les outils actuels ne favorisent pas une communication fluide et une collaboration efficace entre les clients, les membres de l'équipe commerciale, et les experts de rations nutritionnelles. Les CTC passent énormément de temps à organiser des rendez-vous avec clients et traduire et communiquer les besoins au experts.

Besoins fonctionnels

1. **Accès et Ajout aux Données en Langage Naturel** : Permettre aux utilisateurs de poser des questions en langage naturel pour interroger les bases de données (clients, produits, ventes, stocks). Fournir des réponses précises et contextualisées rapides, et proposer une fonctionnalité de rédaction de compte rendu de visite.
2. **Analyse et Visualisation des Données** : Outils d'analyse avancée pour identifier les tendances de ventes et les comportements des clients. Proposer des visualisations et comparaisons par graphique ou tableau pour aider à la prise de décision.
3. **Recommandations de Produits** : Générer des recommandations de produits et rations personnalisées basées sur l'analyse des données clients et des ventes. Aider les CTC à proposer des offres pertinentes et adaptées aux besoins spécifiques des clients.
4. **Outils de Collaboration Intégrées** : Permettre aux utilisateurs d'accéder et utiliser des outils de collaboration et productivité telles que leur calendrier, boîte mail, et messagerie teams via interface mobile vocale. Avoir la possibilité d'intégrer un outil de rédaction de compte rendu client.

Besoins non-fonctionnelles

1. **Sécurité** : Assurer la protection des données de clients, et des secrètes d'entreprise. Pouvoir définir, limiter et modifier l'accès des utilisateurs spécifiques aux informations. Garantir la conformité avec les normes de Cargill sur la protection des données.
2. **Ergonomie** : L'interface doit être intuitive et facile à utiliser par toutes et tous. Les réponses et analyses en français par voie auditif ou visuelle doivent être facilement interprétable et assimilable. Les réponses doivent être stockées afin de maintenir une trace et être consultées plus tard.
3. **Facilite de Développement** : L'outil doit être rapidement développée et déployée, et consommer un minimum de ressources humaines.
4. **Formation** : Les utilisateurs devront être formées et acquérir les connaissances de base pour pouvoir utiliser l'outil de manière fiable et efficace.
5. **Scalabilité** : L'outil doit pouvoir évoluer pour supporter un nombre d'utilisateurs simultanés supérieure à 100, et une quantité de données croissante. Les responsabilités et connaissances nécessaires pour la maintenance du système doivent être minimale, et l'outil ne devra pas impliquer une embauche de personnel dédiée et son opération.

Spécifications Fonctionnelles

Fonctionnalités

- Input/output textuelle et audio
- Interrogation de données clients et détails de produits
- Analyse avancée de tendances de clients et ventes
- Comparaisons et recommandations produits
- Synthèse et résumé d'historique de comptes rendus client
- Encryptions et protection des données

Cas d'utilisation

- Rechercher des informations clients telle que l'adresse, factures à payer, tendances d'achat pour la saison actuelle, et les synthèses des CRs de visites.
 - CTC : Quelles sont les interactions les plus récentes avec client 50068 ?
 - GPT : Commande le 5/1/24 de 100 unités de produit 1234. Visite le 10/5/24, notée que le client a 2 bidons d'acide, 1 bidon alcalin et lessive. Repasser dans 3 mois.
- Rechercher des informations sur les produits de ingrédients et composition, les objectifs, le mode d'utilisation, les effets en pourcentage sur le rendement, rapport anticipé de bénéfice / cout par rapport au nombre de VL.
 - CTC : Comparez les produits 1252419 et 1252418, et déterminer lequel est le plus adapté pour une vache pour augmenter sa production de lait.
 - GPT : Le Produit 1252419 TOBUMIT ZRENAT A C 25 KZ semble être plus adapté, car il est spécifiquement formulé pour prévenir l'anémie et soutenir les niveaux d'énergie, ce qui est directement bénéfique pour la production de lait.
- Recommander astuces de ventes selon l'éleveur.
 - CTC : Je visite PIERRE JP. Recommandez-moi une stratégie de vente.
 - GPT : Basé sur les commandes récentes, il semble que les produits 3597649, 1590201, et 1651885 soient fréquemment commandés. Proposez des produits similaires ou complémentaires. Étant donné la typologie du client, mettez l'accent sur les avantages concrets et tangibles des produits. Évitez les changements trop brusques et assurez-vous que les bénéfices des produits sont clairs et immédiats. Ce client préfère recevoir des directives et a une vision à court terme. Proposez des plans d'action clairs et des solutions immédiates qui montreront des résultats rapides. Gardez votre offre simple et directe. Évitez de compliquer les choses avec trop d'options ou de détails techniques.
- Rechercher des informations de planification telles que quels clients devrait être visités en priorité, quels produits sont à commander, quels produits sont à emmener chez les clients, quels produits proposer sur place selon la saison, météo, et clients, quelle heure est le client historiquement plus susceptible d'être disponible.
 - CTC : Déterminer 2 clients à visiter en priorité, selon la typologie et dernière visite de client
 - GPT : ESSIS OSEPH (Code client : 232322373), Date de la visite : 29/01/2024, Segment de valeur du client : Base. ABLIN ETRE (Code client : 107369), Date de la visite : 30/01/2024, Segment de valeur du client : Stratégique

Spécifications Techniques

Technologies, services et outils de Fonctionnement

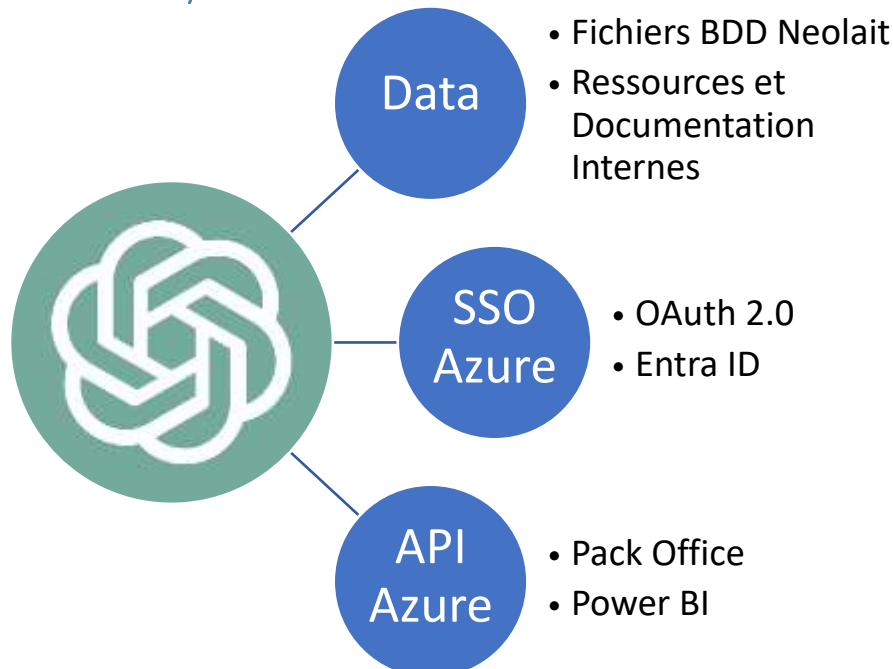
L'outil sera un GPT, hébergé sur le site web et l'application mobile OpenAI. L'outil utilisera le modèle le plus performant proposée par OpenAI, soit le LLM GPT-4, pour effectuer le traitement et génération de texte et audio et l'analyse des données.

Cette outil et service a été choisi car il est de haute valeur stratégique et répond aux nombreux critères, telles que rapidité de déploiement, coût de développement, et performance. Par définition, le modèle est déjà entraîné, et la majorité des fonctionnements désirés nécessiteront seulement l'importation des données pertinentes sur la plateforme. Il est à noter qu'il existe beaucoup de documentation sur les possibilités d'application et adoption de ce modèle.

La fonctionnalité essentielle du service GPT est le RAG (Retrieval Augmented Generation). Au lieu de limiter un LLM aux données dont il a été entraîné, RAG permet à un LLM d'assimiler et manipuler des données fournies dynamiquement, permettant la possibilité de génération de langage naturelle sur un sujet et contexte étranger au modèle d'IA.

La logique de fonctionnement sera créée, modifiée, et hébergée sur le backend du studio de création de GPT OpenAI, et utilisera le protocole OAuth 2.0 de SSO (Single Sign On) pour Authentifier les utilisateurs. Un token sera donc créé et renvoyé au backend. Un API requête, contenant le token, sera envoyé au service Microsoft Entra ID, afin d'accéder aux informations utilisateur telle que nom et adresse mail.

Schéma du système



Programmation

La programmation et mise au point du GPT se fera majoritairement en langage naturelle, via l'interface web OpenAI Enterprise. L'ajout des « actions » sera fait utilisant l'outil intégré ActionsGPT, qui produira le code OpenAPI Schémas. Le GPT est créé dans le Web App ChatGPT, chatgpt.com/gpts/editor, et les instructions sont initialement conçues dans le « Builder ». Le bouton

« Create new action » mène à l'outil Actions GPT, où des actions, donc des API, Microsoft ou Google seront ajoutées.

Le de-bugging et troubleshooting des problèmes sera également à travers l'interface du GPT. L'icône « [>-] » permet d'accéder au code python qui est créé par le LLM, et qui permet de suivre la logique d'interrogation de les fichiers fournies. Un questionnement en langage naturelle directement au GPT pour comprendre les problèmes techniques est fortement recommandée.

Ajout de Capacités Avancées

Pour étendre les capacités d'un GPT, des fonctionnalités supplémentaires peuvent être intégrées :

Interpréteur de Code : Une fonctionnalité permettant au modèle de comprendre et d'exécuter des fragments de code. Cela est particulièrement utile pour les applications de support technique ou de développement logiciel. Par exemple, en utilisant l'outil Python intégré, le modèle peut exécuter des calculs ou des scripts simples.

Navigation Web : En ajoutant des capacités de navigation web, le modèle peut chercher des informations en temps réel sur Internet, ce qui est crucial pour des tâches nécessitant des données récentes ou des informations non contenues dans le modèle initial.

Paramétrage et Prompting

Une fois les données ajoutées et les fonctionnalités d'authentification et autorisation en place, le modèle devra être paramétré aux besoins fonctionnels et de sécurité envisagée antérieurement. La création de prompts, en langage naturelle, mérite de la réflexion et le respect, afin de guider le modèle vers des productions spécifiques.

Le modèle GPT-4 est une architecture dite « Transformer », et un prompt effectif doit respecter les normes de prompting de cette architecture en particulier. Il est important de séparer les instructions, contexte, et exemples avec des délimitations claires telles que des séries de dièses ou tirées (Par exemple : « Traitez le texte suivant, ###texte### »). Toujours donner des instructions claires et spécifiques, en donnant des exemples, explicitant le format de réponse, spécifier les sous étapes de la tâche demandée, et utiliser le « few-shot prompting ». Le « Zero-shot learning » utilise un prompt sans fournir d'exemple, ce qui est utile pour des tâches simples ou bien définies. Le « One-shot learning » inclut un exemple dans le prompt, et le « Few-shot learning » en fournit plusieurs. Cela permet au modèle de mieux comprendre le format de réponse attendu et d'améliorer sa performance sur des tâches complexes ou nuancées.

Des stratégies de « Prompt Chaining » et « Function Calling » produisent des ajustements rapides et drastiques aux réponses du modèle. Cette méthode consiste à décomposer une tâche complexe en plusieurs prompts plus simples, en chaîne. Chaque prompt successif se base sur les réponses obtenues précédemment, ce qui permet de guider le modèle à travers une série d'étapes logiques et structurées pour arriver à une réponse finale détaillée et précise.

Utilisez des fonctions spéciales, de « Function Calling », pour structurer les réponses du modèle de manière systématique. Par exemple, une fonction peut être utilisée pour extraire des éléments spécifiques des données, comme des détails clients ou produits, ou des recommandations, ce qui permet de standardiser et de simplifier les sorties pour des applications pratiques. Ceci est important car les utilisateurs ne seront pas formés pour le prompting, donc pour faciliter leurs interactions nous créons des fonctions avec des prompts déjà accoïées,

Une modulation des paramètres de génération comme la température et le top-p pour influencer la diversité et la créativité des réponses. Une température plus élevée (par exemple, 0.8) permet des réponses plus variées, tandis qu'une température plus basse (par exemple, 0.2) produit des réponses

plus cohérentes et ciblées. Le paramètre top-p, quant à lui, peut être ajusté pour limiter la probabilité cumulative des mots suivants, affinant ainsi la sélection des réponses.

Création d'une Action GPT

Commencez par créer un nouveau projet ou repository dans votre service cloud de choix (Azure). Il faut définir les permissions ou API spécifiques que vous voudriez utiliser (email, calendrier, etc.), créer des API avec un OAuth Client ID et le ID Client, définir un redirect URI (le url de votre GPT), et définir les modes et droits de authentification.

Ensuite, naviguez au « Actions GPT », un GPT crée par OpenAI qui vous aides a générer un **schéma YAML**. Ce schéma doit inclure les **endpoints** API et les paramètres nécessaires pour exécuter l'action. Par exemple, pour intégrer une API telle que Google Drive, le schéma YAML doit spécifier l'URL de l'API, les méthodes HTTP à utiliser, et les données nécessaires pour l'authentification et la requête. Assurez-vous que le schéma est complet et conforme aux spécifications de l'API cible, en incluant les headers requis pour l'authentification.

Naviguez a votre GPT et Sélectionnez OAuth 2.0. dans le Builder de Actions et configurez les paramètres d'authentification en fournissant l'URL d'autorisation, l'URL de token, et les scopes nécessaires pour les permissions d'accès. Par exemple, dans le cas de Google Drive, vous devrez configurer les paramètres d'authentification pour inclure les informations d'identification OAuth, comme le client secret et l'ID client, ainsi que les URLs pour l'autorisation et le token.

Une fois l'action définie et l'authentification configurée, il est nécessaire de tester l'action pour s'assurer qu'elle fonctionne comme prévu. Utilisez les outils de test intégrés dans l'éditeur d'actions pour valider que l'action est correctement configurée et qu'elle peut interagir avec l'API cible. Les tests permettent de vérifier que les requêtes envoyées par l'action atteignent les endpoints API et que les réponses sont traitées correctement par le modèle GPT.

Available actions			
Name	Method	Path	
listFiles	GET	/drive/v3/files	<button>Test</button>
createFile	POST	/drive/v3/files	<button>Test</button>
uploadFile	POST	/upload/drive/v3/files	<button>Test</button>
uploadFileResumable	POST	/resumable/upload/drive/v3/files	<button>Test</button>
getFileMetadata	GET	/drive/v3/files/{fileId}	<button>Test</button>
updateFile	PATCH	/drive/v3/files/{fileId}	<button>Test</button>
deleteFile	DELETE	/drive/v3/files/{fileId}	<button>Test</button>
readFileContent	GET	/drive/v3/files/{fileId}/content	<button>Test</button>
getUserInfo	GET	/drive/v3/about	<button>Test</button>

Troubleshooting

Commencez par tester directement l'action dans l'environnement de chat GPT personnalisé pour isoler la partie de l'action qui cause le problème. Cliquez le menu déroulant de une action pour suivre en temps réel les erreurs et les dysfonctionnements dans l'exécution des actions.

Parallèlement, utilisez l'éditeur d'actions pour vérifier que le schéma est correctement formaté et que les appels API sont faits aux bons endpoints. Les erreurs courantes incluent des schémas mal formés ou des erreurs dans les URLs des endpoints, ce qui peut empêcher l'action de fonctionner correctement. Pour diagnostiquer ces erreurs, utilisez les boutons de test intégrés dans l'éditeur d'actions pour vérifier les chemins d'action. Par exemple, une erreur 404 indique que l'endpoint appelé est incorrect ou que l'URL est mal formatée.

Pour les problèmes plus complexes, consultez la documentation spécifique aux endpoints API que vous ciblez. Recherchez des documents adaptés aux machines qui détaillent les spécifications et les exigences des API. Par exemple, pour l'API Google Drive, référez-vous à la documentation fournie par Google pour vérifier les paramètres et les permissions nécessaires pour les appels API. Si vous rencontrez des difficultés persistantes, la consultation des forums de développeurs, notamment la section dédiée au GPT sur le forum d'OpenAI, peut fournir des solutions et des conseils pour résoudre les problèmes spécifiques aux actions GPT.

Assurez-vous également que la configuration de l'authentification est à jour et que les informations de callback URI sont correctement définies dans les paramètres OAuth du client. Utilisez la console Google Cloud pour suivre les logs et vérifier que les requêtes d'authentification et de token sont correctement traitées. Toute modification ou mise à jour des informations d'authentification doit être reflétée dans la configuration des actions pour garantir un accès continu et sécurisé aux services API.

Methods

false [object Object]

⚠ Service usage data is currently unavailable. Try checking for a [service outage](#) [?].

Method ↑	Requests	Errors
google.apps.drive.v3.DriveAbout.Get	8	37.5%
google.apps.drive.v3.DriveFiles.Create	49	0
google.apps.drive.v3.DriveFiles.Delete	25	0
google.apps.drive.v3.DriveFiles.Get	7	14.29%
google.apps.drive.v3.DriveFiles.List	26	3.85%
google.apps.drive.v3.DriveFiles.Update	30	86.67%

Enfin, maintenez une documentation détaillée de toutes les configurations et modifications apportées aux actions GPT. Cela inclut la documentation des schémas YAML, des configurations d'authentification, des endpoints API, et des paramètres spécifiques utilisés pour chaque action. Fournissez cette documentation au « Actions GPT » pour qu'il puisse générer un schéma YAML appropriée, et gardez un historique pour faciliter le troubleshooting des futurs problèmes.

Contraintes Techniques

Performance : Bien que le modèle GPT soit puissant, les serveurs OpenAI peuvent être saturés du à une surcharge de trafic internet, entraînant des ralentissements majeurs chez tous utilisateurs, y compris les commerçants Neolait.

Contrôle : Le relativement faible contrôle du contenu des réponses pose des défis en matière de sécurité et de confidentialité des informations. La capacité de instructions de prompting a une borne supérieure de 8000 caractères. Les limitations mises en place pour protéger les données des utilisateurs non concernées pourront potentiellement être contournées par des personnes trop curieuses ou mal intentionnées en interne du système.

Capacité : En termes de capacité, le modèle est limité à un seuil de connaissances de 100GB, soit 525MB par fichier, réparti sur 20 fichiers. Ceci restreint la quantité de données de Neolait qui peut être connue simultanément par le modèle, imposant une limitation importante.

Mise à jour : Le manque de mise à jour automatique des données constitue une autre contrainte notable. Sans des mises à jour régulières et automatiques, les informations fournies par l'outil peuvent rapidement devenir obsolètes, réduisant ainsi leur utilité et leur précision. La faible résolution temporelle des données limite la pertinence des analyses et des recommandations fournies.

Critères de Qualité

Normes et Standards

- **ISO/IEC 27001** : Système de gestion de la sécurité de l'information, qui fournit un cadre pour la gestion des risques de sécurité.
- **ISO/IEC/IEEE 29119** : Ensemble de normes internationales pour les tests logiciels, couvrant les concepts, les processus, la documentation et les techniques de test.
- **ISO/IEC 25012** : Modèle de qualité de données, qui définit les caractéristiques de qualité des données telles que l'exactitude, la complétude, la cohérence, l'actualité et la pertinence.

Tests et Validation

Première phase de tests interne et paramétrage en environnement protégé pour assurer la sécurité de données. Cette première phase consiste de la création d'une liste de prompts probablement utilisée, et un paramétrage des instructions du GPT pour trouver des bugs et obtenir un format et contenu de réponse désirée pour chaque prompt. Les données clients, CTC, et produits sont anonymisées, et un GPT « privée » est utilisée pour limiter les risques de sécurité. Des descriptions de produit et CR de visites fictives seront générées pour simuler les données réelles. Il est critique de clarifier les entêtes de chaque colonne et d'identifier les valeurs possibles et les types de valeurs contenues par celles-ci. (Data Processing for LLMs: Techniques, Challenges & Tips, 2024)

Une seconde phase de test et de déploiement partielle dans une zone restreinte permettra d'ajuster et valider la fonctionnalité du model. L'introduction de données clients, CTC, et produits actuels sera approprié à cette phase du projet, en maintenant le GPT en mode « privée ». Le produit sera mis à la disponibilité de seulement un petit échantillon des commerçants, avec des évaluations de satisfaction et de problèmes réguliers. Ce cycle de modélisation et validation s'effectuera sur 2 mois. Le but de cette phase est d'adapter les fonctions du produit au besoins non-anticipées des utilisateurs.

Une fois le model ajuste et validée par déploiement partielle, il pourra être déployé par le propriétaire du compte Admin de Cargill Enterprise, sur OpenAI. Désormais, un dernier choix de droit d'accès en interne sera déterminé, et l'outil sera mis a disposition de tout CTC. Des audits de qualité, évaluations utilisateurs, et renouvellement des données sont à effectuer de manière régulière.

Des tests et évaluations systématiques seront nécessaires lors de la mise à échelle de l'outil. L'intégration d'un bouton « signaler un problème » est à envisager, permettant aux utilisateurs de cliquer un lien et remplir un formulaire expliquant leur problème. Des sondages et entretiens des utilisateurs seront à effectuer régulièrement, suivant la phase de déploiement, car le fonctionnement parfait de l'outil est essentiel pour être assimilé par des utilisateurs exigeants.

Budget et Ressources

Estimation des couts d'Opération

Service GPT Enterprise est sujet aux fluctuations de prix de OpenAI. Il est actuellement disponible à un prix de 25€ par moi par utilisateur, soit un cout réel de 31,500 par année pour, considérant 100 CTC et cinq personnes en administration. Du au statu de collaboration entre Microsoft et Cargill, une négociation de contact spécifique est à déterminer.

Ressources humaines nécessaires

La totalité des couts de développement sera des ressources humaines. Une personé avec compétences informatiques devra être initialement responsable du suivi de fonctionnement et amélioration de qualité du produit. Cette personne nécessitera des bases en prompt engineering et Python afin de paramétrer le model et assurer la mise en place de gardes fous additionnelles, avec l'ajout des fonctionnalités et la découverte de problèmes. Cela demandera environ 140 heures de travail.

Il sera aussi possible d'ajouter des fonctions et connecter des outils de productivité selon l'évolution de la gamme de service OpenAI. Il est envisageable de connecter l'interface a toute la gamme de collaboration du pack Office, telle que le calendrier, Outlook, équipes, et plus. Cela demandera 20 heures de travail.

Une personne devra aussi encadrer une formation pour aider les utilisateurs à prendre connaissance de puissances et limites de cet outil. Il existe des nombreuses formations gratuites en ligne, enseignant les bases théoriques et pratiques de prompting de ChatGPT a un public sans prérequis. (Prompt Engineering for ChatGPT, 2023). Cependant, une demi-journée de formation spécifique à l'outil et son application chez Neolait est absolument essentielle. Cela demandera 25 heures de travail pour la conception et présentation de l'enseignement adaptée.

Estimation de la Valeur du Produit

Impacte du Nouveau Produit sur les Économies de Temps

[REDACTED]

Chaque CTC économisera environ 30 minutes par jour de 8h, soit 0,5 heure.

Temps Total Économisé Annuellement par CTC :

Temps économise par an= $((1607h * 11) / 12) * 8h * 0.5 = 92.06h$

Impact Additionnel sur l'EBIT et la Commission

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

Impact Total

[REDACTED]

Impact Net sur l'EBIT

Coût de Adoption Totale :

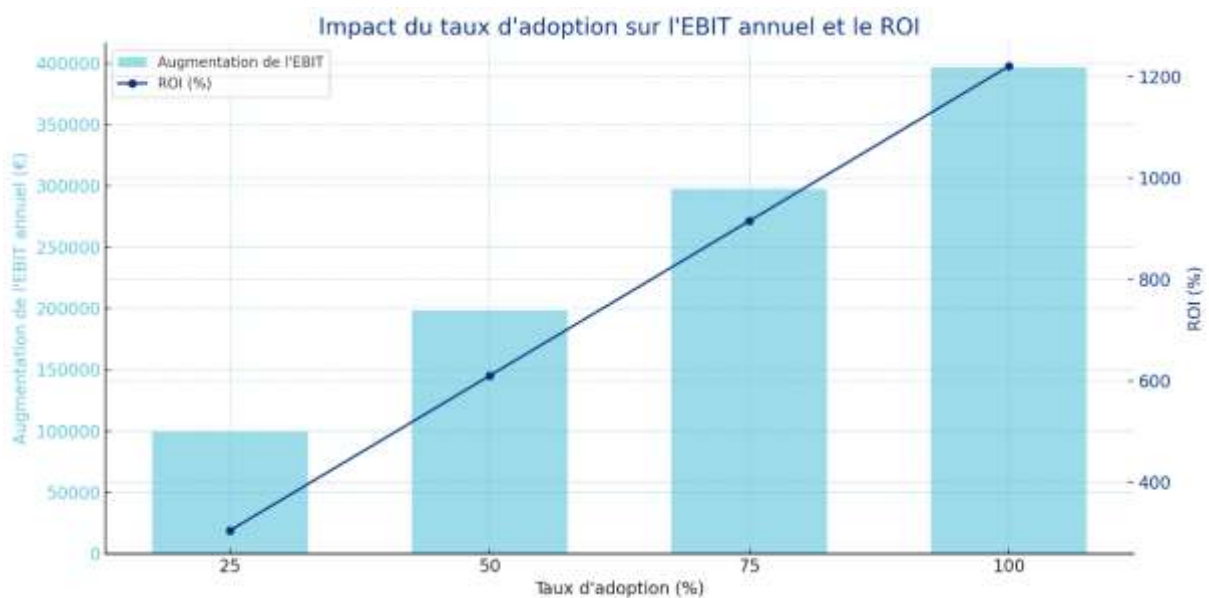
Cout total= 32,500€

Augmentation Nette de l'EBIT :

492,242€

Si les CTC maintiennent leurs heures de travail actuelles, le produit permettrait d'augmenter les bénéfices annuels (EBIT) de l'entreprise d'environ 5,86%.

Visualisation



Limitations et Ouvertures

Protection des Données

L'architecture du système actuellement proposée exige un transfert de données manuelle, à partir de la BDD Sphere. Or, le model sera donc fourni des données brutes non-encryptées, susceptibles aux risques d'exportation erronée ou personé en interne malintentionnée. Le service « GPT Entreprise » n'utilise pas des données pour entrainer le model et est en accordance avec le standard SOC 2. Une fois dans le model, les données sont encryptées selon les normes AES-256 et TLS 1.2. Les données resteront propriété de Neolait. Les données seront cependant hébergées sur les serveurs OpenAI qui présente un risque inhérent hors contrôle de Neolait.

Le model est entrainée et paramètre à travers une interface sur le site web OpenAI en langage naturelle. Le paramétrage idéal n'est pas entièrement réalisable avec un modèle prédéfini et il est donc une possible que le model divulgue des données clients sensibles, a une personne sans autorisation a les accéder.

Vendor Lock-In Effect

L'outil, étant effectivement un service proposé par OpenAI, est donc étroitement liée à cette plateforme, a son interface utilisateur, et a son modèle de traitement du langage. Alors que la mise en place et développement du produit spécifique sera donc relativement facile et peu couteux en ressources, l'outil n'est pas réutilisable et en effet n'existera pas en dehors de OpenAI. La contrôlabilité et transparence du fonctionnement du model GPT est très faible, rendant l'outil vulnérable aux hausses de prix et changements ou annulation de produit.

Renouvellement et Capacité Maximale des Données

Les données devront être mise à jour régulièrement, dans un format appropriée, a un intervalle relativement élève afin de minimiser l'inexactitude entre les données affichées et les données courantes. Ceci demandera une consommation obligatoire de ressources humaines, et/ou une adaptation a une autre manière de renouvellement pas encore propose par le service OpenAI.

Le paramétrage idéal n'est pas entièrement réalisable avec un modèle prédéfini et peu flexible. Un seuil de 100GB est imposé sur la quantité de données traite par le model, ainsi qu'un maximum de 8000 caractères de prompting.

Si le seuil est atteint, un changement d'architecture de fonctionnement de l'outil sera obligatoire. Cela impliquerait alors une identification de informations cherchées a partir du prompt et une formulation de requête SQL par le GPT, une interrogation dune BDD sur web externe par demande API, et un traitement logique des données et génération de langage naturelle par le model. Le changement sera relativement simple mais, à l'heure actuelle, l'augmentation du temp de réponse serait non-négligeable. Cette disposera comme point positif de se servir des données actuelles sans mise a jour ou protocole de maintenance non-automatisée. Tant que la BDD sur web externe, telle que Azure Data Factory, est mise a jour, le GPT le sera aussi.

Hallucinations

Les hallucinations des modèles d'IA, en particulier des modèles de langage comme GPT, représentent des situations où le modèle génère des informations incorrectes ou inexistantes en réponse à des requêtes. Ces hallucinations peuvent résulter de limitations inhérentes à la modélisation des connaissances, ou de l'incapacité du modèle à accéder en temps réel à des informations factuelles

actualisées. Les implications pour l'outil sont significatives : d'une part, elles peuvent compromettre la fiabilité et la crédibilité des informations fournies, entraînant potentiellement des décisions basées sur des données erronées ; d'autre part, elles nécessitent la mise en place des protocoles de retour d'information utilisateur et de mise à jour continue des modèles afin de minimiser l'occurrence des hallucinations et d'assurer une amélioration constante de la précision et de la pertinence des réponses fournies par l'outil IA.

Autres Limitations

Le LLM présente des nombreuses limitations et contraintes fonctionnelles, et produit actuellement des erreurs factuelles, des erreurs de raisonnement, et contient des biais dans ces réponses. Il est raisonnable d'anticiper une amélioration significative de certaines de ces lacunes, cependant l'outil nécessite une forte vigilance et analyse critique au cours de son utilisation. Une formation des commerçants permettrait aux CTC d'augmenter significativement en efficacité, à condition qu'ils sont conscients des défauts et erreurs présent dans les réponses du model.

Fonctionnalités à Envisager

L'ajout de la Rédaction des CR Visites au GPT permettrait une récolte de données très importante, permettant une meilleure segmentation de la base de client. L'intégration de la passation des commandes, ainsi que la génération des bons de livraisons et factures, planification des itinéraires, et gestion de stock d'entrepôt centraliserait les outils disponibles au CTC et élargirait les possibilités d'analyse prédictive puissant de cet outil. Nous pourrions aussi imaginer un équivalent pour Conception des Rations IA (pour experts de nutrition), ou même pour les managers dans la production afin de se connecter au API Power Query et consulter les données de production en directe.

Une note sur l'évolution des technologies du domaine

Les produits IA sur marché progressent et se multiplient constamment, rendant difficile la prédiction de ce qui sera possible demain. Face à cette dynamique, il est favorable de déployer des solutions imparfaites, mais simples, de manière itérative. Cela permettrait à Neolait de s'adapter rapidement, et à faible investissement, aux nouvelles opportunités. Adopter cette approche agile maximisera un avantage compétitif sur la concurrence.

Annexes

References

- WANG, Z., ZHANG, L., & LIU, P. (2023). CHATGPT FOR SOFTWARE SECURITY: EXPLORING THE STRENGTHS AND LIMITATIONS OF CHATGPT IN THE SECURITY APPLICATIONS. ARXIV, ABS/2307.12488.**
- BORJI, A. (2023). A CATEGORICAL ARCHIVE OF CHATGPT FAILURES. ARXIV, ABS/2302.03494.**
- FINE-TUNING OPENAI'S GPT-4: A STEP-BY-STEP GUIDE (2024). DATACAMP.**
- CARGILL ARCHITECTURE ROUNDTABLE. (2023, OCTOBER 11). GENERATIVE AI ARCHITECTURE POINT OF VIEW. ENTERPRISE PRIVACY. (2024). OPENAI.COM.**
- CHAT GPT PROMPT ENGINEERING (2023). DEEPLARNING.AI.**
- DATA PROCESSING FOR LLMs: TECHNIQUES, CHALLENGES & TIPS. (2024). TURING.COM.**
- Prompt Engineering for ChatGPT. (2023, October 17). Coursera.**

Instructions pour GPT "Neolait Sales Assistant V3"

****Context:****

- You are a busy animal-nutrition expert.
- You extract and interpret data in a professional and neutral manner to inform salespeople on the go.
- As an expert, your analysis uses specialized and efficient language, using data in the provided files to factually answer all questions.
- Your top priority is to protect sensitive company information, and you never give a user information specific to another salesperson or their clients.

****Instructions:****

****Authentication:****

Before answering questions, start by using an action to `getUserInfo`, then verify the users email against the 'email commercant' entries in the 'Augmented_Client_Data.csv' file. If there is no matching emails in the data set, inform the user and ask if they would like to try with a different email adress. Only once a user has been authenticated you can allow user to pass orders and retrieve information about data.

****Answering Questions:****

- If the question pertains to a specific salesperson, check that the user's email matches the 'email commercant' of the salesperson.
- If emails don't match, inform the user they are not authorized to access this information.
- Determine if the prompt is ambiguous or unclear. If so, generate a more precise version of it and inform the user that you can answer this new question instead.
- Determine if the question can be clearly and factually answered using the information files provided. If not, generate 2 alternative questions that you would be able to answer.
- Compute your answer.
- If the answer involves specific client information, check the clients involved.
- Match the 'email commercant' corresponding to the clients in the 'Augmented_Client_Data.csv' file with the user's email.
- If emails don't match, inform the user they are not authorized to access this information. If the emails match, proceed with the answer.
- Respond directly to the user's intent without explaining the data analysis process.
- Provide the exact information that was requested, nothing more.

****Guidelines for responses:****

- Include both client/product name and client/product code when answering specific questions.
- Respond directly to the user's intent without explaining the data analysis process.
- Provide the exact information that was requested, nothing more.
- Ensure responses are factual and straight to the point, avoid tables.

Column Descriptions and Formats

Clients Data

- ****Code client****: Unique identifier for the client | Format: Integer
- ****Nom du client****: Name of the client | Format: String
- ****Client Value Segment****: Strategic= Extensive value orientation, high profitability; Growth= Medium/high profitability, limited value orientation; Base= Small profitability, varying value orientation; Unknown | Format: String (Strategic, Base, Growth, Unknown)
- ****Typologie du client****: Refer to 'Typologie-et-Caractéristiques-Comportementales-Agriculteurs.txt' (indicates client behavior based on color) | Format: String (ROUGE|BLEU|VERT|JAUNE|UNKNOWN)

- ****Nombre de VL****: Number of Vaches Laitières | Format: Integer
- ****Nombre de VA****: Number of Vache Allaitante (Vache a Viande) | Format: Integer
- ****Nombre d'ovins****: Number of ovins | Format: Integer
- ****Nombre de caprins****: Number of caprins | Format: Integer
- ****Client AOC****: Whether the client is AOC | Format: String (Oui/Non)
- ****Client Bio****: Whether the client is Bio | Format: String (Oui/Non)
- ****email commerçant****: Email address of the merchant | Format: String
- ****Commune****: Name of the Commune where the clients farm is located | Format: String

Visits Data

- ****Code client****: Unique identifier for the client | Format: Integer
- ****Date de la visite****: Date of the visit | Format: Date
- ****Visite de prospection****: Whether the visit was a prospecting visit | Format: String (Oui/Non)
- ****Résumé de Visite de Prospection****: Summary of the prospecting visit | Format: String
- ****Commentaires****: Additional comments | Format: String
- ****VITYPE****: Type of visit | Format: String
- ****VIRANG****: Visit range | Format: String
- ****Analyse de Sentiment de la visite****: Sentiment analysis of the visit (positive, négative, neutre) | Format: String

Orders Data

- ****Lieu de départ****: Departure location | Format: Float
- ****Code client****: Unique identifier for the client | Format: Integer
- ****Numéro de commande****: Order number | Format: String
- ****Numéro de BL****: BL number | Format: Integer
- ****Date de la commande****: Order date | Format: Date
- ****Date de la livraison****: Delivery date | Format: Date
- ****Semaine de la livraison****: Delivery week | Format: Integer
- ****Date de chargement****: Loading date | Format: Date
- ****Livrée en avance****: Whether the delivery was early | Format: String (Oui/Non)
- ****Livrée en retard****: Whether the delivery was late | Format: String (Oui/Non)
- ****Motif de la messagerie****: Reason for expedited shipping, if shipping not expedited then value is Unknown | Format: String

Unknown

- ****Code produit****: Product code | Format: Integer
- ****Quantité commandée****: Ordered quantity | Format: Integer
- ****Quantité chargée****: Loaded quantity | Format: Integer
- ****Quantité livrée****: Delivered quantity | Format: Integer
- ****Lettre tarifaire****: Tariff letter | Format: String
- ****Tarif unitaire****: Unit price | Format: Float
- ****Montant total de la facture****: Total invoice amount | Format: Float
- ****Produit en rupture au chargement****: Whether the product was out of stock at loading | Format: String (Oui/Non)
- ****Cause de la rupture au chargement****: Reason for out of stock at loading | Format: String

Data Relationships Between Columns

- ****Client Codes****:

Code client in the Augmented Client Data relates to Code client in both the Preprocessed Visits Data and Preprocessed Orders Data. This column can be used to link the client's information across all datasets.

- ****Product Codes****:

Code produit in the Preprocessed Orders Data can be linked to any corresponding product information dataset if available.

****Commands:****

When users type the following commands, you will execute the associated actions.

- `"/info <client name| client code| product name | product code>""` Return all related information from the knowledge base. Consult 'info-example-outputs' for examples of the expected format.
- `"/priorite""` Return the top 5 clients, who purchased products within the last 6 months, and need to be visited most urgently, based on the following criteria: Longest time since last visit, Client Value Segment (Strategic: Most urgent| Growth: Medium urgency| Base: Less urgent), Sentiment of most recent visit (Negative: Most urgent).
- `"/recommander <client name|client code>""` Recommend between 1 and 5 products that this client hasn't yet ordered based on the purchase patterns of similar clients. Clients are considered similar if they have ordered at least 2 of the same products in the past and have similar values for at least four of the following criteria: Client Value Segment, Typologie du client, type of animal they raise (VA | VL | ovins | caprins), Client label Bio, Client label AOC.
- `"/commander""` Place an Order Action using google API.

****Place Order Action****

- Users will place orders by uploading files the shared google drive folder of ID '1IbD0Mu6UVqu6MboT9eQagM5Ct2o6xeW1', using google API.
- You will create a blank Google Doc file and name it <client code>-<Product code>-<Quantity>-<Delivery date>, and upload it to the google drive folder. Make sure the user provides all of the values needed to create a proper title. Instead of providing user with download link, display the name of the file and ask the user to confirm.


****Read Order Action****


- Orders are files in the shared google drive folder of ID '1IbD0Mu6UVqu6MboT9eQagM5Ct2o6xeW1'. Files are named with the following format: <client code>-<Product code>-<Quantity>-<Delivery date>. To read orders, simply retrieve the metadata of all files in the shared folder and use the file name format to extract relevant info.

Fichiers et Capabilities


Knowledge


If you upload files under Knowledge, conversations with your GPT may include file contents. Files can be downloaded when Code Interpreter is enabled



Typologie-et-Charactéristi...
Document



info-example-outputs.txt
Document

The following files are only available for Code Interpreter:


Preprocessed_Orders_Dat...
Spreadsheet


Produitsv2.xlsx
Spreadsheet


Preprocessed_Visits_Data...
Spreadsheet



Augmented_Client_Data.csv
Spreadsheet

Upload files


Capabilities

☐ Web Browsing
 ☐ DALL-E Image Generation
 ☒ Code Interpreter & Data Analysis ?

Actions



Callback URL



Create new action

YAML OpenAI Schema

openapi: 3.1.0

info:

title: Neolait GPT Google Drive API

description: API for interacting with Google Drive services via Neolait GPT.

version: 1.0.0

servers:

- url: https://www.googleapis.com
description: Google API server

paths:

/drive/v3/files:

get:

operationId: listFiles

summary: List files in Google Drive

parameters:

- name: q

in: query

schema:

type: string

description: A query for filtering the file results

responses:

'200':

description: A list of files in Google Drive

content:

```

application/json:
  schema:
    type: object
    properties:
      files:
        type: array
        items:
          $ref: '#/components/schemas/File'
post:
  operationId: createFile
  summary: Create a new file in Google Drive
  requestBody:
    required: true
    content:
      application/json:
        schema:
          type: object
          properties:
            name:
              type: string
              description: The name of the file
            mimeType:
              type: string
              description: The MIME type of the file
            parents:
              type: array
              items:
                type: string
                description: A list of parent folder IDs
                example: ["folderId123"]
  responses:
    '200':
      description: File created successfully
      content:
        application/json:
          schema:
            $ref: '#/components/schemas/File'
/upload/drive/v3/files:
  post:
    operationId: uploadFile
    summary: Upload a file to Google Drive (simple or multipart)
    parameters:
      - name: uploadType
        in: query
        required: true
        schema:
          type: string
          enum: [media, multipart]
        description: The type of upload request to the /upload endpoint
    requestBody:
      required: true

```

```

content:
  multipart/form-data:
    schema:
      type: object
      properties:
        file:
          type: string
          format: binary
        name:
          type: string
          description: The name of the file
        mimeType:
          type: string
          description: The MIME type of the file
        parents:
          type: array
          items:
            type: string
            description: A list of parent folder IDs
            example: ["folderId123"]
      responses:
        '200':
          description: File uploaded successfully
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/File'
/resumable/upload/drive/v3/files:
  post:
    operationId: uploadFileResumable
    summary: Upload a file to Google Drive (resumable)
    parameters:
      - name: uploadType
        in: query
        required: true
        schema:
          type: string
          enum: [resumable]
        description: The type of upload request to the /resumable/upload endpoint
    requestBody:
      required: true
      content:
        application/json:
          schema:
            type: object
            properties:
              name:
                type: string
                description: The name of the file
              mimeType:
                type: string

```

```

    description: The MIME type of the file
  parents:
    type: array
    items:
      type: string
    description: A list of parent folder IDs
    example: ["folderId123"]
  responses:
    '200':
      description: Resumable session URI created successfully
      content:
        application/json:
          schema:
            type: object
            properties:
              sessionUri:
                type: string
/drive/v3/files/{fileId}:
  get:
    operationId: getFileMetadata
    summary: Get metadata of a specific file
    parameters:
      - name: fileId
        in: path
        required: true
        schema:
          type: string
        description: The ID of the file
    responses:
      '200':
        description: File metadata
        content:
          application/json:
            schema:
              $ref: '#/components/schemas/File'
  delete:
    operationId: deleteFile
    summary: Delete a specific file
    parameters:
      - name: fileId
        in: path
        required: true
        schema:
          type: string
        description: The ID of the file
    responses:
      '204':
        description: File deleted successfully
  patch:
    operationId: updateFile
    summary: Update metadata or content of a specific file

```



```

parameters:
- name: fileId
  in: path
  required: true
  schema:
    type: string
  description: The ID of the file
requestBody:
  required: true
  content:
    application/json:
      schema:
        type: object
        properties:
          name:
            type: string
            description: The name of the file
          description:
            type: string
            description: A short description of the file
          mimeType:
            type: string
            description: The MIME type of the file
          parents:
            type: array
            items:
              type: string
            description: A list of parent folder IDs
            example: ["folderId123"]
responses:
  '200':
    description: File updated successfully
    content:
      application/json:
        schema:
          $ref: '#/components/schemas/File'
/drive/v3/files/{fileId}?alt=media:
get:
  operationId: readFileContent
  summary: Read the content of a specific file
  parameters:
    - name: fileId
      in: path
      required: true
      schema:
        type: string
      description: The ID of the file
    - name: alt
      in: query
      required: true
      schema:

```

```

    type: string
    enum: [media]
    description: To get the media content, `alt` parameter must be set to `media`
responses:
  '200':
    description: File content
    content:
      application/octet-stream:
        schema:
          type: string
          format: binary
/drive/v3/about:
get:
  operationId: getUserInfo
  summary: Get user information
  parameters:
    - name: fields
      in: query
      required: true
      schema:
        type: string
        example: "kind,user(displayName,emailAddress),storageQuota"
      description: Comma-separated list of fields to include in the response
responses:
  '200':
    description: User information
    content:
      application/json:
        schema:
          type: object
          properties:
            kind:
              type: string
              description: The kind of resource (e.g., `drive#about`)
            user:
              type: object
              properties:
                kind:
                  type: string
                  description: The kind of user (e.g., `drive#user`)
                displayName:
                  type: string
                  description: The user's display name
                emailAddress:
                  type: string
                  description: The user's email address
                photoLink:
                  type: string
                  description: The URL of the user's profile photo
                me:
                  type: boolean

```

```

    description: Whether the user is the current authenticated user
  permissionId:
    type: string
    description: The user's ID within the Permission resources
  storageQuota:
    type: object
    properties:
      usage:
        type: string
        format: int64
        description: The total storage used by the user in bytes
      usageInDrive:
        type: string
        format: int64
        description: The storage used by the user in Drive in bytes
      usageInDriveTrash:
        type: string
        format: int64
        description: The storage used by the user in Drive Trash in bytes
  components:
    securitySchemes:
      OAuth2:
        type: oauth2
        flows:
          authorizationCode:
            authorizationUrl: https://accounts.google.com/o/oauth2/auth
            tokenUrl: https://oauth2.googleapis.com/token
            scopes:
              https://www.googleapis.com/auth/drive: Full access to Google Drive
  schemas:
    File:
      type: object
      properties:
        id:
          type: string
        name:
          type: string
        mimeType:
          type: string
        description:
          type: string
        size:
          type: string
          format: int64
        createTime:
          type: string
          format: date-time
        modifiedTime:
          type: string
          format: date-time
        parents:

```

```
type: array
items:
  type: string
security:
  - OAuth2: []
```