# Document de réponse
# Application de chat ‘CHandler’

*Rédigé par O.Pichot*
*RT231 UHA*


# SOMMAIRE

# 1. Introduction

## a. Synopsis

Ce document de réponse à pour but d'expliquer le travail réalisé pour ce projet, nous verrons en détail les fonctionnalités disponibles que l'application CHandler offre, les limites de celles-ci, les recommandations à prendre en compte pour de futurs améliorations, etc..
Il est important de noter que ce document ne rentrera pas dans le côté technique de l'application. Pour ceci, merci de vous référer au documentations serveur & client.

## b. Contexte du projet

Vous nous avez contacté pour le développement d'un logiciel interne de communication car les solutions actuellement sur le marché ne correspondent pas à vos attentes.
L'entièreté du logiciel doit être fonctionnelle sur votre réseau local sans accès externe à une quelconque base de données, API ou service tiers.
D'après nos échanges, le serveur gérant les communications sera hébergé sur une machine d'ores et déjà présente sur votre réseau local et seuls les administrateurs auront accès au dit serveur.

## c. Objectif de l'application

L'objectif principal de l'application est d'améliorer la communication entre vos collaborateurs pour des échanges plus fluides et un échange d'informations rapide et groupé de par l'utilisation de salons de discussions regroupant de multiples employés en fonction de leur secteur de travail.

# 2. Réalisation effective du projet

## a. Description générale de l'application

CHandler est une solution permettant aux employés de l'entreprise de communiquer au sein de groupes de discussions ou bien en discussions privées (d'un utilisateur à un autre).
Celle-ci permet également une gestion basique des droits d'accès aux différents salons afin de laisser le choix à chacun de discuter, ou non, avec d'autres collègues. Nous verrons plus en détail ci-dessous les fonctionnalités relatives aux droits d'accès.
L'application à été pensée pour permettre de nombreuses évolutions dans le futur en fonction des besoins (et des retours) des utilisateurs.

## b. Fonctionnalitées implémentées

Vous trouverez ci dessous la liste exhaustive des fonctionnalitées disponible dans cette version 1.0 de CHandler :

| Type d'utilisateur | Fonctionnalité | Commentaire |
|---|---|---|
| Lambda | Inscription d'un nouvel utilisateur. | *Bien évidemment, toute la partie authentification dispose d'une gestion d'erreur avec un retour pour l'utilisateur (par exemple : Mauvais mdp, utilisateur inexistant, nom d'utilisateur invalide, etc..)* |
| Lambda | Connection sur le compte d'un utilisateur existant. | |
| Lambda | Rejoindre un salon de discussion | *N/A* |
| Lambda | Demander l'accès à un salon auquel il n'a pas accès pour l'instant. | *Chaque nouvel utilisateur n'a de base accès qu'au salon 'Général', il peut également demander à rejoindre le salon 'Blabla', la demande sera immédiatement validée sans l'aval d'un admin.* |
| Lambda | Envoyer à un autre utilisateur une demande d'ami | *N/A* |

| | | |
|---|---|---|
| Lambda | Gérer les demandes d'amis reçues d'autres utilisateurs | *Il est possible pour chaque demande de soit accepter soit refuser celle-ci* |
| Lambda | Envoyer un message sur un salon de discussion | *Si l'utilisateur à accès au salon* |
| Lambda | Envoyer un message privé à un ami | *N/A* |
| Admin | Gérer les demandes d'accès à un salon des utilisateurs | *N/A* |
| Admin | Ajouter un ami de force | *Un admin n'a pas besoin de l'aval de l'utilisateur pour l'ajouter en ami.* |
| Admin | Accès à tous les salons de discussion par défaut | *N/A* |
| Admin | Connection sur le terminal du serveur pour l'exécution de commandes | *N/A* |
| Admin | Bannir un utilisateur (action définitive et irréversible)<br><br>> /ban <nom_utilisateur> | *Non seulement l'utilisateur sera banni mais également l'adresse ip associée au client. Ceci empêche l'utilisateur banni de se recréer un compte et rejoindre de nouveau un salon.* |
| Admin | Kick un utilisateur (bannissement temporaire, la durée est choisie par l'admin)<br><br>> /kick <nom_utilisateur> <durée (secondes)> | *N/A* |
| Admin | Lister toutes les connexions actives à un moment T.<br><br>> /show_clients | *N/A* |
| Admin | Changer le mot de passe d'un utilisateur.<br><br>> /change_password <nom_utilisateur> | *Au cas où un utilisateur aurait oublié son mot de passe.* |
| Admin | Passer un utilisateur au statut d'admin<br><br>> /make_admin <nom_utilisateur> | *Attention, un admin à de grands pouvoir ceci n'est pas à prendre à la légère.* |
| Admin | Lister les threads actifs du serveur.<br><br>> /threads | *Cette commande liste les threads actuellement actifs sur le serveur.* |

| | | |
|---|---|---|
| Admin | Couper le serveur<br><br>> /kill | *Cette commande coupe complètement le serveur et prévient chacun des clients de l'arrêt.* |

Il est possible d'obtenir des informations plus détaillées sur le fonctionnement de chacune de ces fonctions dans les documentations client & serveur.

## c. Architecture globale

Vous trouverez ci-dessous un schéma global du fonctionnement de l'application au sein du réseau de l'entreprise ou celle-ci sera implémentée.

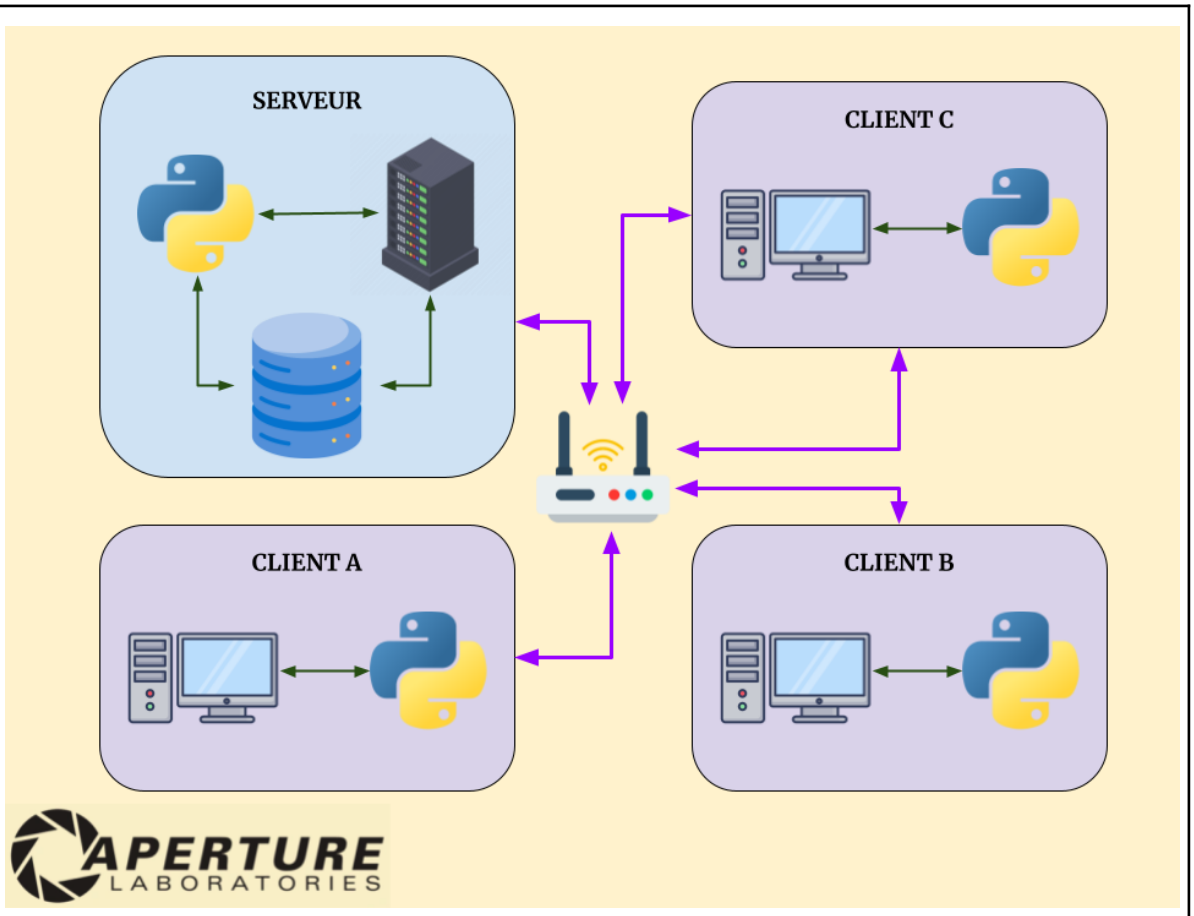


Fig.1 Schéma architecture de CHandler

## d. Défis rencontrés et solutions apportées

Les deux principaux défis rencontrés auront été le fait que les clients ne puissent pas avoir accès à la base de données directement et également la mise en place de la commande 'kill' côté serveur.

Concernant l'accès à la base de données par le serveur uniquement, il a donc fallu mettre en place un système d'envoi de 'paquets'.
En effet, pour envoyer par exemple une conversation entière d'un salon pouvant potentiellement représenter plusieurs centaines voire milliers de messages, il n'est pas possible/censé d'envoyer l'entièreté de ceux-ci en une seule et unique requête. Il pourrait y avoir avoir lors de l'envoi un problème de connexion ou bien des pertes d'informations ce qui rendrait le système non fiable.
Pour pallier cette éventualité, nous avons créé pour ce genre de situations un système préparant à l'avance un nombre X de paquets que l'on envoie ensuite un par un au client.
Le premier paquet envoie donc simplement le nombre de paquets que le client doit s'attendre à recevoir ainsi qu'un message de confirmation lui indiquant que les paquets sont prêts à envoyer une fois le OK reçu.

Pour ce qui est de la commande kill, le problème reposait sur la manière dont le serveur écoutait continuellement chaque client dans l'attente d'une requête de la part de l'un d'entre eux.
Tant qu'une commande n'est pas reçue, cette boucle d'écoute reste statique et ne peut donc effectuer aucune autre action, y compris recevoir une commande du terminal comme kill, ban, ou autre..
La solution adoptée à donc été d'implémenter un 'timeout' dans chaque itération de la boucle d'écoute (1 seconde environ) ce qui fait qu'avant la reprise de chaque boucle, on peut mettre à jour les drapeaux (flags) du serveur au cas où celui-ci n'aurait pas besoin de continuer d'écouter les clients.
Ceci permet de couper le serveur à tout moment indépendamment de la situation de chaque client actuellement connecté.

# 3. Livrables et documentations

## a. Code

Ce projet vous est livré avec l'entièreté du code nécessaire au bon fonctionnement de l'application.
Ce code est divisé en deux parties, la section CLIENT et la section SERVEUR.

La section **Client** contient les fichiers suivant :

| Fichier | Utilité |
|---|---|
| *main.py* | Fichier principal du programme (côté client), celui-ci crée une instance de client, le connecte au serveur et génère l'interface utilisateur. Tout le programme découle de ce fichier. |
| *client.py* | Ce fichier contient la Classe principale du programme (Client_handle).<br>Pour le détail des méthodes et variables de cette classe, merci de vous référer à la documentation Client. |
| *popup.py* | Ce fichier contient plusieurs classes gérant les fenêtres 'popup' autre que la page principale de l'application. |
| *sys_x.py* | Ce fichier contient les fonctions permettant une interaction directe avec le terminal client. Le fichier est court dû au fait que le client dispose d'une interface graphique donc peu d'interaction avec le terminal. |
| *styles.css* | Ce fichier contient les styles de l'application, en l'attente de retour de votre part sur l'aspect visuel de l'application, celui-ci est plutôt court car il est amené à être complété en fonction des besoin futur remontés par les utilisateurs regardant l'aspect visuel de l'application. |

La section **Serveur**, quant à elle, contient les fichiers suivants :

| Fichier | Utilité |
|---|---|
| *main.py* | Fichier principal du programme (côté serveur), celui-ci crée une instance de serveur (Server_handle), le déploie sur le réseau et démarre les premiers threads du serveur. |
| *server.py* | Ce fichier contient la classe principale du serveur (Server_handle). Pour le détail des méthodes et variables présentent au sein de celui-ci, merci de vous référer à la documentation Serveur. |
| *sql_handler.py* | Ce fichier contient TOUTES les fonctions incluant un lien avec la base de données.<br>Toute action impliquant une requête qu'il s'agisse de prise de données, de mise à jour ou de suppressions de données sera effectuée via ce fichier ci. |
| *sys_x.py* | Ce fichier contient les fonctions permettant une interaction directe avec le terminal. Pour l'instant le terminal reste très 'brouillon', étant donné que CHandler n'en est qu'à sa première version, nous avons préféré limiter la mise en forme / nettoyage du terminal afin de permettre un retour instantané en cas d'erreur ou de comportement sortant de l'ordinaire de la part du programme. |
| *database_template.sql* | Ce fichier est probablement le plus important de tous, il s'agit d'un template (modèle) de la base de données qui sera utilisée par l'application. Sans celui-ci, aucune information ne peut être stockée au bon endroit et le serveur ne sera pas en mesure de démarrer. |

## b. Documentations

Ce projet vous est livré avec plusieurs documentations afin de permettre à votre technicien de comprendre, modifier et installer l'application.

Chacune de ces documentations concerne un point précis du projet et est la plus complète possible sans pour autant nécessiter une lecture complète.
Il est possible de se référer dans chacun de ses documents uniquement à la partie à laquelle vous êtes intéressé.

Ceci n'est évidemment pas conseillé pour les parties installation de l'application car sauter des étapes est rarement conseillé lors de la mise en place d'un tel programme.

Vous trouverez ci-dessous la liste des documentations présente dans les livrables qui vous auront été fournis :

| Document | Contenu |
|---|---|
| Procédure d'installation de CHandler | Comment installer CHandler sur un serveur, comment installer CHandler sur chacune des machines clients. |
| Documentation développeur (Sphynx/Docstring) | Documentation générée par Sphynx via la notation Docstring présente au sein du code de l'application |
| Documentation Client | Documentation détaillé du code côté client. |
| Documentation Serveur | Documentation détaillé du code côté serveur. |
| Modèle relationnel (BDD) | Présentation de l'organisation de la base de données de l'application CHandler |
| Document de réponse | Le document que vous lisez actuellement |

# 4. Limites de l'application

## a. Sécurité

A ce stade du développement, la priorité n'aura pas été la sécurité mais le bon fonctionnement de l'application.

Les communications entre le serveur et les clients sont basées sur un protocole TCP mais ne sont pour l'instant pas encryptés. Ceci signifie que n'importe quel outil de suivi de paquets (Wireshark par exemple) au sein du réseau pourrait facilement suivre tout message transmis du serveur au client et inversement.
En effet les transmissions sont claires et le message ne nécessite donc aucun décryptage pour être lisible. Pour pallier ceci, il serait bénéfique d'implémenter à l'avenir un système d'encryption basé sur du SSL afin de rendre toute trames envoyée par l'application illisible par un utilisateur malveillant.

Clairement ceci représente pour l'instant la plus grand faille dans le système et c'est pourquoi nous espérons pouvoir aborder ce point avec vous lors d'une prochaine réunion afin de résoudre ce problème potentiel.

De plus, si les transmissions sont en claires sur le réseau, il ne faudrait pas très longtemps à un utilisateur malveillant pour récupérer des crédentiels (employé ou admin), ce qui lui donnerait ensuite accès à l'entièreté des conversations de salons mais aussi à toutes les conversations privées des utilisateurs auxquels il aurait accès.

Nous insistons sur le fait que ce point n'est pas à prendre à la légère et nous vous conseillons fortement de contacter notre secrétaire (Mme Chell) pour préparer une réunion et instaurer un devis pour la mise à jour de l'application côté sécurité.

## b. Mesures de sécurité mises en place

Notre application est dotée de trois principaux outils regardant la sécurité de l'application.

**Premièrement**, un système d'authentification empêche le démarrage de l'application tant que l'utilisateur en question n'est pas connecté et validé par le serveur.
Chaque utilisateur doit entrer ses crédentiels avant d'entamer une quelconque discussion afin d'empêcher quidam de lancer CHandler et d'accéder à des salons contenant des informations potentiellement sensibles.

**Secondement**, le système d'accès aux salons est très restreint.
Seuls deux salons sont disponibles de base à un utilisateur nouvellement créé (Général & Blabla). Ceci permet d'empêcher un utilisateur malveillant de créer un compte et d'avoir immédiatement accès à chaque salon de discussion.
S'il souhaite avoir accès à un salon plus spécialisé, il devra alors d'abord être validé par un admin et seulement à ce moment-là aura t-il accès au dit salon.
De plus, si un utilisateur fait une demande pour rejoindre chaque salon existant sur l'application, l'admin pourra clairement voir chacune de ces demandes et alors sonner l'alerte car le comportement ne sera pas celui d'un utilisateur normal.

**Troisièmement**, chaque administrateur de l'application à la possibilité de bannir et / ou kick un utilisateur. En cas de doute sur un utilisateur, l'admin peut tout d'abord le kick pour une durée qu'il choisit, ce qui lui laisse alors le temps d'investiguer le problème, et si l'utilisateur représente bel et bien une menace, il pourra alors bannir celui-ci.
En plus de bannir simplement le compte de l'utilisateur, l'adresse IP de celui-ci sera associée au ban ce qui empêchera l'utilisateur de créer un nouveau compte via la machine à laquelle il a accès sur le réseau.

*N.B : Si cela venait à se produire, nous recommandons fortement que vous nous contactiez afin que nous effectuions un diagnostic complet de la situation.*

## c. Recommandations pour une utilisation sécurisée

Premièrement, nous conseillons au technicien de votre entreprise d'installer l'application au cas par cas pour chaque utilisateur de l'entreprise et de sécuriser les fichiers de manière à empêcher le déplacement de ceux-ci (ceci empêchera la copie des fichiers pour les exécuter depuis une machine autre que l'une de celles préparée par le technicien) ainsi que de bloquer la modification des fichiers exécutables de l'application.

De plus, il est conseillé de former vos collaborateurs sur les bonnes pratiques à adopter avec les logiciels de communication interne d'une entreprise, notamment :

- Bien verrouiller son ordinateur lorsque l'on quitte son poste de travail.
- Garder son mot de passe secret; on ne le divulgue pas, on ne le note pas physiquement sur papier ou sur un fichier de l'ordinateur, on ne rentre pas son mot de passe sous le regard d'autres employés.
- Remonter à un admin ou à un responsable de l'entreprise tout comportement suspicieux constaté de la part d'un autre utilisateur (par exemple: un collaborateur inconnu vous contacte à de multiples reprises sur l'application, spam, etc..).
- On se déconnecte et on ferme l'application dès que la journée de travail est terminée.
- Ne pas ouvrir de mail et/ou exécuter des programmes non reconnues par l'entreprise, ceci pourrait déclencher un 'cookie-graber' ou un 'keylogger' ce qui permet l'obtention des mots de passe que l'on utilise.
- Utiliser un mot de passe complexe avec caractères spéciaux et minimum 12 caractères.

Pour plus d'informations sur les bonnes pratiques à adopter nous vous conseillons de lire cette ressource mise à disposition par le gouvernement à Cybermalveillance.gouv.fr :

*Cliquer sur le lien ici*

## d. Confidentialité / Garantie de protection des données

Concernant la gestion des données fournies par le biais de CHandler, notamment :

- Informations personnelles (nom, prénom, mail, etc..)
- Informations données lors d'échanges dans un salon ou dans une discussion privée
- Liste des utilisateurs
- etc, etc..

Nous adhérons **strictement et sans exception** aux principes clés fixés par le **CNIL** via le **RGPD** pour la protection des données et la gestion de celles-ci.

Notamment :

- Finalité de l'utilisation des données : Les données récoltées à un moment T par l'application ne seront pas utilisées ultérieurement à d'autres fins.
- Pertinence de l'utilisation des données : nous ne récolterons jamais plus que ce qui est nécessaire au bon fonctionnement de l'application.
- Durée limitée de conservation : Les données seront détruites ou archivées par nos soins dans le respect des obligations légales lorsque l'application ne sera plus utile à l'entreprise.
- Sécurité du stockage des données : les données doivent être stockées de manière sûre, localement sur le territoire Français et non dans un centre de données tiers.
- Droit des personnes : Si quiconque de l'entreprise nous contacte pour supprimer, altérer ou obtenir une copie des données que nous possédons à son égard, cela sera fait immédiatement et ceci sans consultation auprès des responsables de l'entreprise.

Pour plus d'information, nous vous conseillons le site officiel de la Commission Nationale de l'Informatique et des Libertés : https://www.cnil.fr/fr

## e. Maintenance prévu à court et long-terme

Concernant la maintenance, nous vous proposons un SAV disponible chaque jour ouvrable de la semaine de 8h à 18h.

Nos techniciens seront disponibles et à votre écoute en cas de questions sur le fonctionnement du logiciel mais également en cas de problèmes à nous remonter si vous découvrez des bugs ou un comportement anormal de l'application.

Etant donné qu'il s'agit ici du début de notre collaboration et que nous vous proposons ici la version 1.0 de CHandler, une mise à jour hebdomadaire sera disponible notamment pour fixer les bugs qui nous auront été remontées mais également implémenter différentes options dont vous pensez qu'il serait bon d'ajouter à l'application.
Bien évidemment nos frais seront susceptibles d'évoluer en fonction des options supplémentaires que vous souhaitez implémenter.
Cependant, tout problème lié à un bug ou bien un comportement anormal de l'application sera compris dans notre forfait maintenance de base.

Nous vous proposons également de nous rencontrer de manière bi-mensuel avec l'un de vos responsables de production et votre technicien en charge afin de discuter de l'évolution à prévoir de l'application.

En cas d'échec catastrophique (arrêt complet du logiciel et impossibilité de redémarrer celui-ci), nous interviendrons sur place dans les 24 heures maximum.

# 5. Recommandations futures

## a. Améliorations possibles / suggestions pour le futur

Nous parlerons plus en détail à l'avenir des améliorations que vous souhaiteriez voir sur l'application, il sera notamment important de récupérer les feedbacks des utilisateurs pour implémenter en premier les options dont ils ont le plus besoin.
Ceci-dit, nous avons déjà plusieurs idées à ajouter à l'application, certaines d'entres-elles d'ores et déjà pré-implémenter au sein du code :

- Possibilité de création de salons par un admin ou de discussions de groupes entre amis.
    - *Le code est d'ores et déjà prévu pour fonctionner avec d'autres salons en plus des 5 présents de base.
- Ajout de statut de connexions des utilisateurs (présent/absent/ne pas déranger, etc etc)
- Système d'appels via VOIP entres deux utilisateurs
- Implémentation d'une encryption des données via SSL pour une sécurité accrue.
- Customisation de profil utilisateur (photo, description, etc, etc)
- Interface graphique de serveur.
- Sauvegarde / création de 'backups' de la base de données à la fin de chaque journée.

## b. Evolution en fonction des besoins changeants

Comme mentionné précédemment, nous prévoyons une évolution constante de l'application en fonction des besoins des utilisateurs / de la production.
Un feedback constant entre les utilisateurs, votre direction et nos équipes sera le plus important pour la pérennité du logiciel. En effet, les besoins regardant des changements à apporter se feront connaître après de nombreuses heures de test et un retour constant des utilisateurs.
L'un des avantages de posséder votre propre solution de communication est le fait que celle-ci soit réalisée spécifiquement en fonction de vos besoins et ce de manière organique en évoluant dans le temps.

# 6. Conclusion

## a. Bilan global du projet

Ce projet aura été une expérience enrichissante et fructueuse du début à la fin, chaque problème engagé nous aura forcé à développer une application robuste et polyvalente à la fois.

Cette version 1.0 de CHandler est d'ores et déjà fonctionnelle avec tout ce qui nous avait été demandé dans le cahier des charges transmis et pour autant, à la vue du projet et l'engouement que celui-ci aura généré dans nos équipes, nous avons déjà un produit plus complet que nécessaire ce qui montre bien notre investissement de A à Z pour parvenir à cette solution finale.

Le projet n'en est cependant qu'à ses débuts et nous aimons à penser que votre technicien saura reconnaître au sein du code de l'application une organisation rigoureuse prête pour d'autres améliorations et de nombreux ajouts à venir.

Tout ceci nous mène donc à la partie finale de ce document;

## b. Remerciements et disponibilité pour des ajustements futurs

Nos équipes d'Aperture souhaitent vous remercier pour la confiance que vous nous avez accordée pour le développement de cette application.
Nous espérons sincèrement que celle-ci deviendra l'une des pierres angulaires de la productivité de votre compagnie et espérons refaire affaire avec vous sous peu.

Nous restons disponibles à toute heure pour avoir votre retour et déjà commencer à planifier le futur de cet outil afin de continuer de l'améliorer pour qu'il devienne, à terme, parfaitement taillé à vos besoins.

Cordialement, l'équipe ***Aperture Labs***.