

Rapport de stage sur le développement de solutions applicatives pour smartphones



Stage effectué du 15 mai au 30 juin 2017
sous l'encadrement de Monsieur PRACA Damien

HighConnexion
365 Avenue Archimède
13290 Aix-en-Provence
04 42 24 58 24



HighCONNEXION

SOMMAIRE

| | |
|--|----|
| Remerciements..... | 3 |
| 1. Introduction..... | 4 |
| 1.1 Histoire et présentation de la société HighCo..... | 4 |
| 1.2 Histoire et présentation de la filiale HighConnexion..... | 4 |
| 1.3 Organigramme de la société..... | 6 |
| 1.4 Mission du stage..... | 7 |
| 2. Développement..... | 8 |
| 2.1 Ma place dans l'entreprise et analyse de l'état du projet..... | 8 |
| 2.2 Objectifs et deadline de l'application ClubNotifier..... | 8 |
| 2.3 Résultat du travail réalisé..... | 9 |
| 3. Conclusion..... | 14 |
| Annexes..... | 15 |
| Index..... | 20 |
| Bibliographie / Sitographie..... | 21 |

Remerciements

Après mes 7 semaines vécues au sein de la filiale HighConnexion de la société HighCo, je tenais à remercier sincèrement l'ensemble de l'équipe éducative du lycée Marie Curie ainsi que les collaborateurs de l'agence ayant permis de rendre ce stage possible et de l'avoir rendu si riche en apprentissage tout en m'offrant une vision plus réelle du monde du travail et du quotidien de salariés exerçant le métier que je convoite.

J'aimerais adresser également une pensée toute particulière à Monsieur **MARKOVITCH Yan**, qui m'a beaucoup aidé dans ma recherche de stage et m'a permis de postuler dans cette entreprise. Son écoute et ses conseils m'ont permis de cibler mes candidatures, et de trouver ce stage qui était en totale adéquation avec mes attentes.

J'adresse également mes remerciements et mes pensées à Monsieur **PRACA Damien**, ingénieur et développeur sénior au sein de la filiale HighConnexion pour son accueil, le temps passé ensemble et le partage de son expertise au quotidien. Grâce à sa confiance, j'ai pu m'accomplir totalement dans la mission qui m'a été donnée. Il fut également d'une aide précieuse dans les moments les plus délicats et durant ma première semaine, pendant laquelle l'adaptation au développement mobile fut difficile.

Pour finir, je souhaite également accorder ma gratitude à Monsieur **GUENIOT Sylvain**, ingénieur analyste développeur au sein de HighConnexion pour le soutien et l'aide qu'il m'a également apporté. Son expérience, ses compétences et son côté très pédagogue dans le développement mobile fut extrêmement enrichissants.

1. Introduction

1.1 Histoire et présentation de la société HighCo

Côté à la bourse de Paris depuis 1996, HighCo est un groupe de solutions marketing fondé en 1990.

Spécialisé dans la grande distribution et la grande consommation, HighCo est présent sur 5 types d'activités : le couponing, l'échantillonnage, les points de ventes, le traitement des données et les conseils en communication.

Sa mission est de développer des offres toujours plus mesurables et plus performantes en terme de retour sur investissements. Cela passe notamment par l'utilisation des technologies web et mobiles.

Avec plus de 700 collaborateurs et 14 filiales réparties en Europe, HighCo, dont le siège social est situé à Aix-en-Provence, se positionne comme un acteur majeur et indépendant des solutions marketing dans le monde du numérique. En 2016, la société réalise un chiffre d'affaires de plus de 155,38 millions d'euros pour une marge brute de 81,06 millions d'euros.

Profil de la société HighCo :

Président : CHABASSIEU Didier

Création : 1990

Chiffre d'affaire (2016) : 155,53 millions d'euros

Résultat : 14,10 millions d'euros

Effectif : 700 collaborateurs

Implantations : France, Belgique, Benelux, Espagne, Italie, Emirats Arabe-Unis

1.2 Histoire et présentation de la filiale HighConnexion

High Connexion est une filiale appartenant à 51 % à la filiale HighCo Shopper, qui elle-même appartient au groupe HighCo. Elle développe et maintient des applications iOS, Android & Windows Phone de type natives ou customisées en fonction de l'ergonomie et du rendu souhaité et apporte son expertise technique, fonctionnelle et graphique pour la réalisation d'applications. L'organisme conçoit et met également à disposition des outils innovants pour informer, alerter, fédérer, divertir, promouvoir ou acheter.

Sa capacité de production est de 6 ingénieurs mobiles basés à Aix-en-Provence et à Lyon qui ont déjà réalisés plus de 100 applications distinctes dans des univers très variés et pour de grands comptes tels que Carrefour, l'Olympique de Marseille, La Poste, Oasis, Bouygues Telecom, Colis privé etc. et possède une forte expertise dans la gestion complète d'un projet et de ses spécifications jusqu'à sa publication sur les stores.

On peut relever des applications phares plus en détails comme :

- L'application **RTM** permettant, entre autres, d'obtenir les horaires des lignes et les places des parkings en temps réel.

- L'application **McDo France**, permettant de passer commande depuis son mobile, puis de retirer celle-ci dans le restaurant sélectionné au préalable.
- L'application **eJobber**, société d'intérim, permettant de se positionner facilement sur des missions disponibles en fonction d'une date souhaitée.

HighConnexion est une filiale du groupe HighCo qui a su, avec l'expertise de ses dirigeants, associer l'innovation technologique et les usages mobiles à la gestion de la relation client pour faire d'HighConnexion un acteur reconnu, apprécié et leader sur le marché du marketing et du paiement mobile.

Profil de la filiale HighConnexion :

Président : LAURENT Bruno

Création : Février 2008

Chiffre d'affaire (2015) : 37,8 millions d'euros

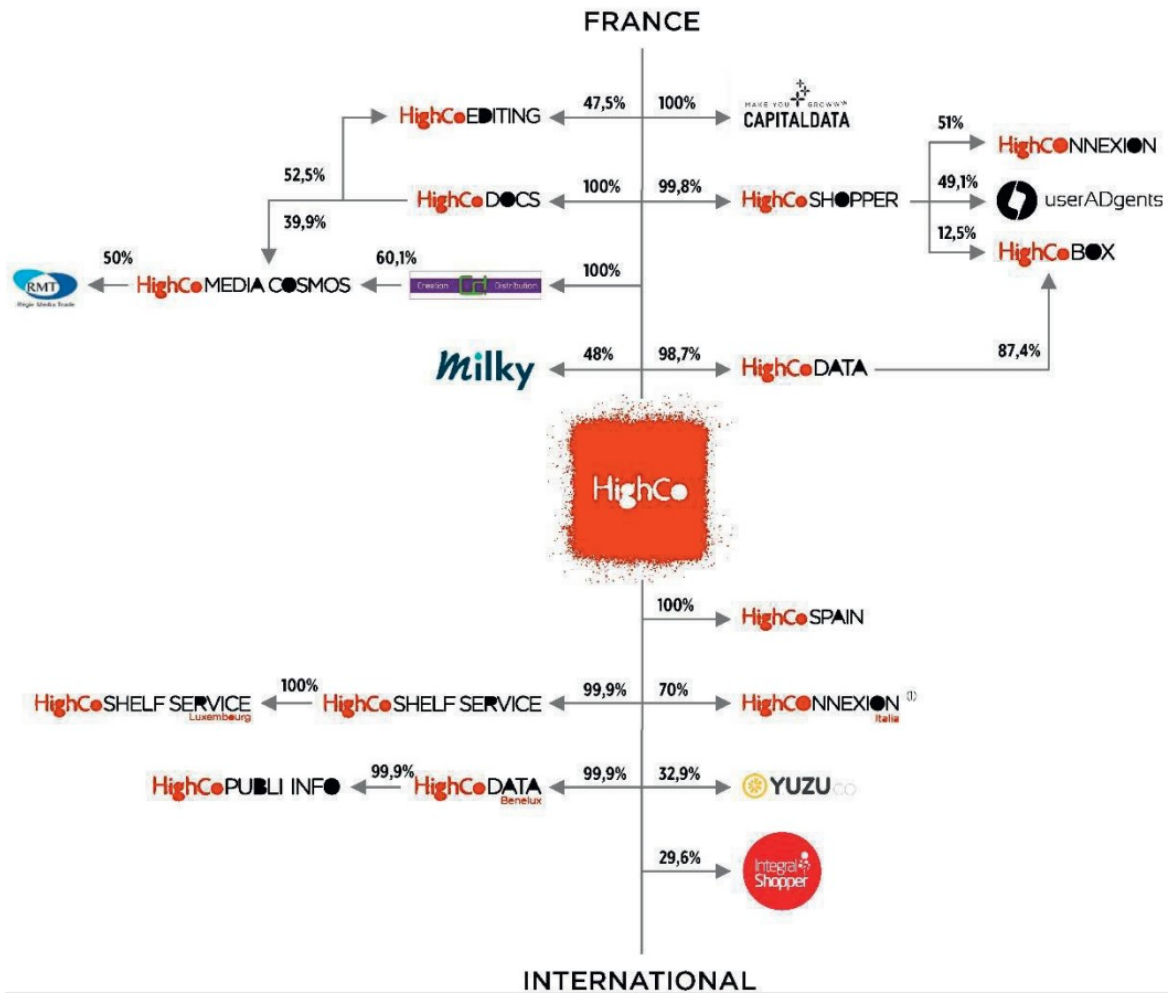
Résultat : 3,1 millions d'euros

Effectif : 30 collaborateurs

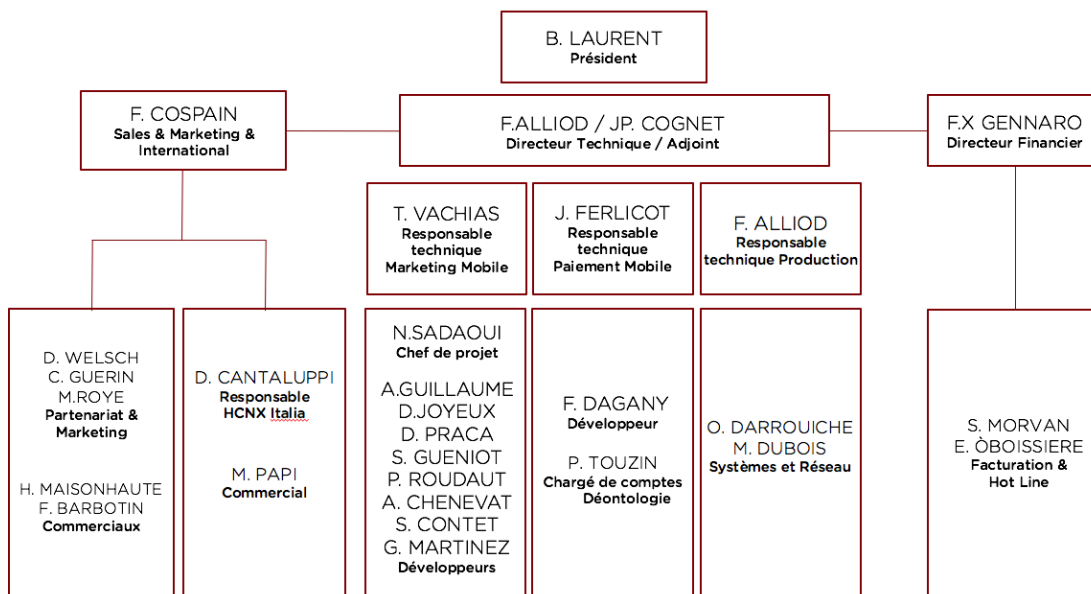
Implantations : Lyon, Paris, Aix-en-Provence, Côme (Italie)

1.3 Organigramme de la société

Organigramme de la société HighCo :



Organigramme de la filiale HighConnexion :



1.4 Mission du stage

ClubNotifier

À mon arrivée au sein de HighConnexion, il m'a été demandé de développer l'application mobile Android « ClubNotifier », un projet de marque blanche interne à l'entreprise, qui n'est donc pas une commande d'un client de l'agence.

Ce projet est un réseau social à destination des clubs de foot amateur en France et à l'étranger devant permettre aux managers et dirigeants d'un club de football de poster des articles et comptes-rendus à destination des adhérents et des parents, mais aussi de créer des événements à venir, tout en permettant de lancer des campagnes de notifications push et SMS afin d'informer rapidement les personnes intéressés d'un nouveau match, d'un changement de date ou d'une annulation à cause d'intempéries par exemple. L'application doit également permettre aux managers d'avoir une liste des contacts, leur donnant la possibilité d'informer une personne en particulier rapidement et facilement.

En résumé : cette application doit permettre aux clubs de football amateurs de faciliter leur gestion, leur communication et leur transparence interne et à destination des familles.

2. Développement

2.1 Ma place dans l'entreprise et analyse de l'état du projet

J'ai travaillé durant ma période de stage en qualité de développeur Android, me plaçant donc sous l'encadrement de Madame **SADAoui Noémie**, chef de projet de la section développement à Aix-en-Provence.

À mon arrivée, il est important de noter que le site web de Club Notifier existait déjà en recette, bien qu'il n'intégrait pas toutes les fonctionnalités et web services nécessaires à une première version.

L'application Android, quand à elle, possédait déjà quelques fonctionnalités et se décomposait en 4 parties distinctes : « Articles et comptes-rendus », « Calendrier », « Contacts » et « Communication ».

- La partie « Articles et comptes-rendus » permettait seulement d'afficher la liste des articles, sans la prise en charge des images publiées.
- La partie « Calendrier » était fonctionnelle et permettait d'obtenir la liste des événements pour un jour sélectionné par l'utilisateur, mais manquait également de fonctionnalités, tout comme la partie « Articles et comptes-rendus »
- La partie « Contacts » était également fonctionnelle et permettait d'afficher la liste des contacts et de visualiser les détails d'une personne en particulier, mais souffrait du même manque de fonctionnalités.
- La partie « Communication », par contre, ne fonctionnait pas et affichait une erreur à son lancement. Elle est censée regrouper les événements à venir afin de permettre aux dirigeants de lancer une campagne de notifications aux personnes concernées.

Pour finir, le menu latéral de l'application ne fonctionnait qu'à moitié (2 parties sur 4 étaient cliquables).

2.2 Objectifs et deadline de l'application ClubNotifier

Après le constat et l'analyse de l'état général de l'application à mon arrivée, il m'a été demandé de faire du mieux que je pouvais pour progresser sur les fonctionnalités manquantes, qui étaient nombreuses.

Dans la partie « Articles et comptes-rendus » :

Il était nécessaire de rendre les images compatibles avec l'application afin qu'elles puissent être visibles dans la liste des actualités et dans les détails de chacune d'elle. Des détails d'une actualité qui étaient, par ailleurs, à faire quasi intégralement. On devait également pouvoir agrandir une photo au moment du clic sur celle-ci, avec la possibilité de zoomer avec ses doigts. De plus, il devait être possible de créer, d'éditer et de supprimer un article.

Dans la partie « Calendrier » :

Les fonctionnalités de création, d'édition et de suppression d'un événement étaient là encore manquantes et devaient être réalisées.

Dans la partie « Contacts » :

La recherche devait être corrigée, car celle-ci affichait systématiquement les détails du premier utilisateur de la liste et non pas ceux du contact sélectionné. Encore une fois, les fonctionnalités de création, d'édition et de suppression n'avaient pas été réalisées et nécessitaient du travail. Dans les détails d'un contact, il devait également être possible de lancer l'application « Téléphone », « SMS » et « E-mail » à partir de trois boutons distincts, permettant de contacter rapidement une personne.

Dans la partie « Communication » :

Il fallait corriger cette partie afin qu'elle affiche correctement la liste des événements à venir. Toute la partie de création de campagne de notifications devait également être réalisée.

D'autres objectifs, qui ne sont pas liés directement à une partie de l'application, étaient également prévus, à savoir :

- Une gestion des permissions permettant d'attribuer des droits différents à des parents, des adhérents mineurs/majeurs, des arbitres jeunes ou sénior, des éducateurs, des managers, des dirigeants, ou tout simplement à des équipes distinctes. Par exemple : il est évident qu'un parent ou un adhérent n'auront pas la possibilité de lancer une campagne de notifications, tout comme il est compréhensible de se dire que seuls les managers et les dirigeants auront le droit de supprimer un article ou un compte-rendu.
- Une gestion de la déconnexion d'un utilisateur dans le cas où celui-ci se connecterait au compte sur deux appareils différents.

À propos de la deadline, celle-ci était normalement inexistante, puisque mon maître de stage m'avait bien explicité à mon arrivée que j'étais censé faire du mieux que je pouvais pour leur avancer le travail. Cependant, à partir de la quatrième semaine, Monsieur **VACHIAS Thibaut**, responsable technique et marketing mobile de la section développement mobile d'Aix-en-Provence, m'a fait savoir qu'il souhaitait avoir une application fonctionnelle d'ici la fin de mon stage afin de pouvoir la mettre en ligne sur le Play Store, la boutique officielle d'Android. J'ai donc dû concentrer mes efforts sur les fonctionnalités essentielles à une mise sur le store selon les exigences de Monsieur VACHIAS, c'est-à-dire, réaliser tous les objectifs de la partie « Articles et comptes-rendus » et « Communication », ainsi que tous les objectifs non liés à une partie spécifique de l'application cités précédemment. La création d'événement, ainsi que la création/édition/suppression d'un contact étaient auxiliaires dans un premier temps pour une mise en ligne, bien que j'étais toujours amené à les développer si les fonctionnalités principales étaient terminées avant la fin de mon stage.

2.3 Résultat du travail réalisé

Au terme de mes 7 semaines passées et partagées avec les développeurs mobiles de la filiale HighConnexion, je suis à présent en mesure de revenir sur l'avancée et le travail réalisés sur le projet ClubNotifier.

Partie « Articles et comptes-rendus » :

Ce fut la partie sur laquelle j'ai travaillé en premier. J'ai réussi à répondre à tous les objectifs que nous nous étions fixé au terme des 3 premières semaines. J'ai donc intégré la première image d'un article dans la liste de tous les articles, puis j'ai travaillé sur la visualisation des détails de celui-ci. À présent, quand l'utilisateur clique sur un article, les détails de l'actualité s'affichent et la possibilité de visualiser l'intégralité des images publiées est présente. Au moment du clic sur une photo, une nouvelle fenêtre s'ouvre et permet de visualiser l'image en grand format et de zoomer avec ses doigts. Une fonctionnalité qui peut paraître anodine, mais qui n'est pas automatiquement gérée par Android. Elle doit donc être prise en charge par le développeur.

J'ai ensuite passé plus d'une semaine sur la fenêtre de création d'un article, qui était très complexe à gérer et qui a été mon premier véritable challenge technique. J'ai pu avancer à mon rythme, qui s'est accéléré de jour en jour grâce à l'expertise que m'apportait les développeurs de HighConnexion. Ce fut également ma première intégration de web services dans ma formation de développeur. L'édition, elle aussi, m'a donné du fil à retordre, car celle-ci était gérée par la même fenêtre que la fenêtre de création. Il a donc fallu que je permette au programme de différencier la création et l'édition d'un article dans une même fenêtre tout en préremplissant les cases avec les données de l'actualité en cours d'édition. Le bouton « Publier » devait aussi se transformer en bouton « Modifier », qui n'envoyait pas la même requête au serveur. La suppression, quant à elle, fut assez simple à gérer après l'expérience que j'ai acquise et la difficulté des fonctions précédemment développées.

Partie « Calendrier » :

Cette partie et la partie « Contacts » étant non essentielles à la sortie de la première version de ClubNotifier comme mentionné plus haut, ce sont les parties que j'ai traité en dernier et pour lesquelles j'ai eu le moins de temps (moins de 2 semaines). Cependant, j'ai tout de même réussi à terminer tout ce qui était à faire, à savoir la création, l'édition et la suppression d'un événement... du moins, en théorie. En effet, seule la suppression fonctionne correctement, car l'édition et la création font appel à des web services inexistantes lors de mon stage. Le travail côté client (à savoir, du côté de l'application) est néanmoins réalisé et celle-ci est prête à fonctionner, ce qui veut dire qu'une fois le web service créé par l'équipe en charge de leur développement, l'utilisateur pourra réaliser et éditer un événement sans être contraint de mettre à jour son application.

Il a également fallu faire en sorte que lorsque l'utilisateur sélectionne un jour dans le passé, le bouton « Notifier » se transforme en « Rédiger un compte-rendu », aux fonctions différentes.

J'ai également profité de mon travail sur le calendrier pour créer une vue « paysage ». À présent, dès que l'utilisateur réalisera une rotation de son smartphone en mode paysage,

il pourra profiter d'une fenêtre redimensionnée et repensée pour ce format, avec le calendrier du côté gauche de l'écran, et les événements du jour sélectionné à droite. Une spécificité non prévue au départ, mais facilitant l'ergonomie et l'utilisation.

Partie « Contacts » :

Un travail conséquent a été apporté aux contacts. Dans un premier temps, il a été nécessaire de corriger la fonction de recherche qui était défaillante et présentait des détails qui différaient du choix de l'utilisateur, car même si la liste s'actualisait en fonction des termes de la recherche, celle-ci continuait à afficher les détails dans l'ordre alphabétique de la liste non triée. Prenons un exemple : la recherche « LACROIX » renvoyait comme résultat « LACROIX Jordan », mais affichait les détails du contact « Armand Alexandre », qui était la première personne dans la base de donnée triée. Un problème qui peut sembler simple à résoudre en passant l'index de la position du contact, mais la tâche s'est avérée plus ardue que prévu, car l'utilisateur pouvait également faire une recherche, se tromper dans un caractère, effacer, et réécrire le terme de sa recherche, ce qui donnait un index erroné. Il a donc été nécessaire de trouver un juste milieu entre l'optimisation du code et l'interrogation de la base de donnée, qui est une opération coûteuse en data pour le forfait mobile d'un utilisateur lambda.

Je me suis également attardé, encore une fois, sur l'édition, la création et la suppression d'un contact qui comportaient le même souci que la partie « Calendrier » : les web services ne sont toujours pas réalisés pour ces fonctions considérées comme secondaires.

Pour finir, j'ai pu ajouter les trois boutons dans les détails d'un contact permettant de rapidement accéder à l'application « Téléphone », « SMS » et « E-mail », avec le numéro pré-rempli pour les deux premiers boutons et l'adresse e-mail pour le dernier. Des difficultés se sont encore posées pour cette fonction somme toute basique à première vue, mais qui peut rapidement devenir compliquée en prenant en compte la rétrocompatibilité avec les anciennes versions d'Android qui fonctionnaient différemment et nécessitaient une toute autre méthode pour lancer des applications systèmes. L'équation se complique encore quand on sait que depuis la version 5.0 d'Android, l'utilisateur a la possibilité de changer les applications systèmes par des applications tierces. Il était donc nécessaire, pour cette troisième possibilité, d'interroger non plus l'application intégrée par défaut, mais l'application préférentielle de l'utilisateur. Des particularités qui nécessitent finalement plus d'une cinquantaine de lignes de code pour trois boutons ayant une fonction très basique.

Partie « Communication » :

Deuxième partie essentielle à la sortie de ClubNotifier, la partie « Communication » nécessitait un travail important. Dans un premier temps, il m'a fallu corriger l'erreur s'affichant dès le lancement de la fenêtre. Une erreur due à une évolution du web service renvoyant la liste des événements futurs qui nécessitait une modification de son intégration dans l'application.

Une fois la liste des événements futurs obtenus, le plus gros du travail était à venir : le bouton « Notifier » devait permettre de lancer la fenêtre de notification que j'ai dû entièrement réaliser. La rendre fonctionnelle ne fut pas très difficile après mon apprentissage sur la partie « Actualités et comptes-rendus ». Cependant, la réalisation de

A à Z d'une interface (qui se fait en XML) fut un nouveau challenge, m'obligeant à me référer aux normes de design sous Android et à me confronter à des problèmes nouveaux, sachant qu'une interface est composée de plusieurs couches d'éléments, eux-mêmes composés de dizaines d'attributs permettant de définir leur position, leur réaction au clic de l'utilisateur, leur déplacement en cas de l'ouverture du menu latéral ou de l'ouverture d'une barre d'erreur au bas de l'écran, ainsi que leur niveau de superposition en comparaison aux autres éléments (exemple : un bouton flottant se trouve au-dessus d'une liste déroulante).

Tout comme la partie « Calendrier », j'ai également réalisé une interface pensée pour le mode paysage, avec la liste des événements à venir sur le côté gauche, et la fenêtre de campagne de notifications s'affichant sur la droite dès que l'utilisateur sélectionne un événement.

Autres réalisations non liées à une partie :

J'ai dû mettre en place la gestion des droits des utilisateurs, afin d'accorder l'accès à certaines fonctions uniquement aux personnes accréditées par le serveur. Pour cela, j'ai dû faire en sorte qu'à la connexion d'une personne, l'application récupère les permissions de l'utilisateur courant et les stocke en base de données locale durant toute la durée de la session courante. Ensuite, à chaque ouverture d'une nouvelle fenêtre comportant des fonctionnalités réservées uniquement à un groupe de personnes, l'application vérifiait les droits de l'utilisateur. Ces droits se définissaient par trois caractères pour chacune des quatre parties de l'application : « r » (pour Read) si l'utilisateur avait le droit de lecture, « w » (pour « Écriture ») si l'utilisateur avait le droit de créer, modifier et supprimer du contenu, et « a » (pour « Administrator ») si l'utilisateur était un administrateur de la section en cours d'affichage, lui donnant un accès complet, même aux fonctions encore en cours de développement.

Cependant, il était parfois compliqué de gérer les droits des utilisateurs, car dans la partie « Communication », il est également possible de supprimer un événement à venir via un bouton faisant office de raccourci. Or, même si l'utilisateur a les droits d'écriture ou les droits d'administrateur sur la partie « Communication », la suppression d'un événement est liée aux droits de la partie « Calendrier », ce qui veut dire que si l'utilisateur n'a que les droits de lecture pour cette partie, il ne devrait pas pouvoir accéder ce bouton raccourci. Même problème pour le calendrier d'ailleurs, qui affiche le bouton « Rédiger un compte-rendu » pour les événements passés uniquement si l'utilisateur possède les droits nécessaires dans la partie « Articles et comptes-rendus », rendant l'implémentation de la gestion des droits assez minutieuse et nécessitant une vigilance sur chaque fonctionnalité qui n'est pas forcément liée à la partie courante.

Une fonctionnalité de gestion de droit qui est intimement liée à une autre fonctionnalité que j'ai dû mettre en place : la gestion de la déconnexion. À mon arrivée, si un responsable de HighConnexion souhaitait vérifier l'avancée de l'application ou la tester en se connectant au même compte administrateur que celui que j'utilisais, l'application se contentait de m'écrire « Token invalide » dans une barre d'erreur au bas de l'écran à chaque fois que j'essayais d'accéder à la base de données distante. Une mesure insuffisante, car je restais tout de même connecté, pouvant accéder à tout ce que j'avais déjà chargé en base de données locale et à toutes les fenêtres d'administration.

À présent, j'ai fait en sorte qu'à chaque tentative de requête vers la base de donnée distante, l'application vérifie si une erreur 401 (celle à laquelle est liée l'erreur du token invalide) est renvoyée par le serveur. Si c'est le cas, je supprime le token de l'utilisateur et l'intégralité du contenu des tables de la base de données locale, puis le redirige vers la page de connexion en supprimant toutes les fenêtres de l'application enregistrées dans la pile d'activités Android (un élément système qui enregistre toutes les fenêtres par lesquelles un utilisateur est passé pour lui permettre de revenir à celles-ci en appuyant sur la touche « Précédent » du smartphone).

À l'aide de cette fonctionnalité, l'utilisateur sera déconnecté et notifié par un message qui lui en expliquera la raison si une autre personne se connecte sur le même compte plutôt que d'afficher une simple barre d'erreur. De plus, tous les éléments qu'il avait préchargé et dont il n'a plus accès en temps que personne non connectée, sans droit de lecture, d'écriture ou d'administration, sont supprimés du téléphone.

Pour terminer, j'ai également corrigé, ajouté ou amélioré un certain de nombres de choses plus ou moins mineures, telles que le problème de certaines parties qui n'étaient pas cliquables dans le menu latéral, des erreurs d'affichage pour le mode paysage sur les parties « Articles et comptes-rendus » et « Contacts » ou encore, une réorganisation du code afin de le rendre plus lisible et facile à maintenir par la suite. De plus, le menu latéral récupère et affiche à présent le nom/prénom de l'utilisateur courant. Pour finir, les différentes parties vous affiche un message pour vous notifier qu'il n'y a aucun élément à charger s'il n'y a aucun article ou événement à venir dans votre groupe plutôt que de laisser l'utilisateur sur une page blanche, ne sachant pas si celle-ci n'arrive pas à charger ou si elle n'a tout simplement rien à afficher.

3. Conclusion

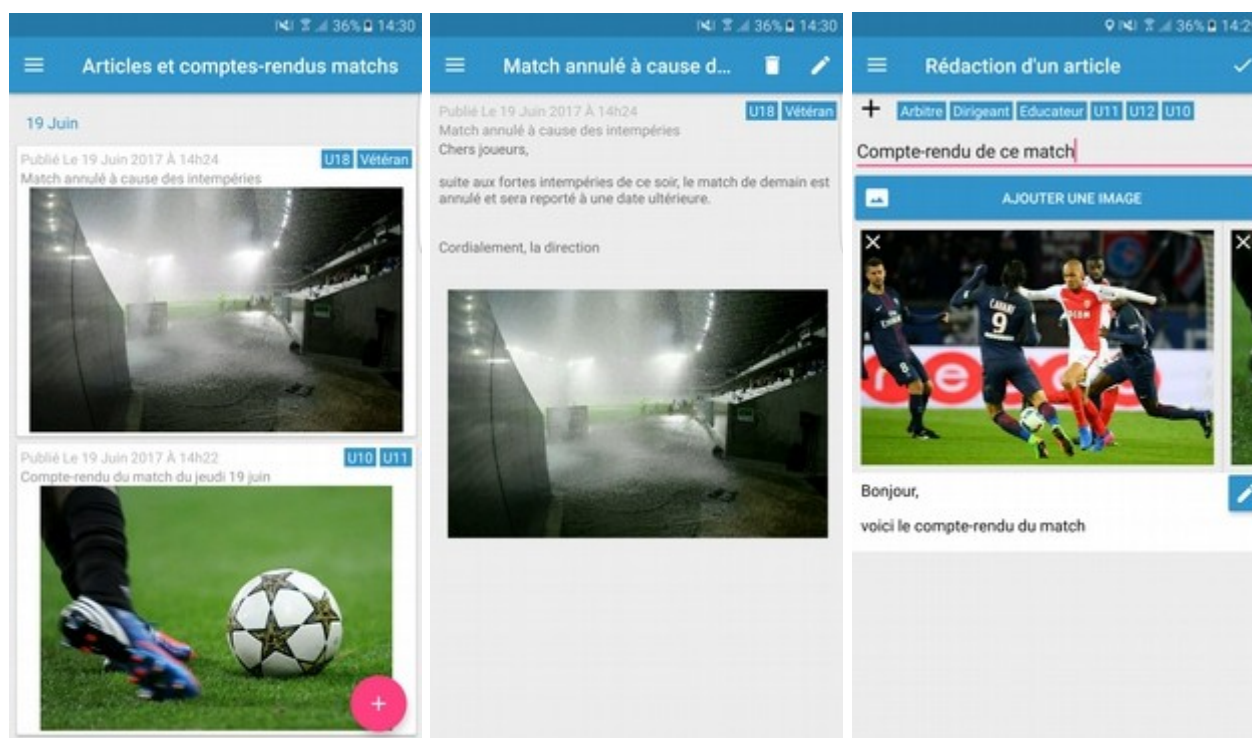
Pour conclure, j'aimerais m'attarder sur ce que mon stage, que je considère comme une réussite, m'a apporté tout au long de ces 7 semaines.

Dans un premier temps, j'ai pu acquérir des connaissances en Java, le langage le plus utilisé pour le développement Android que nous n'avons pas vu durant l'année mais qui se rapproche énormément du langage C#, m'ayant permis une adaptation rapide. J'ai pu également acquérir un savoir non-négligeable sur le développement d'applications mobiles pour l'écosystème Android, qui est d'une complexité que je n'aurais pas imaginé auparavant, mais qui est extrêmement intéressant, avec une évolution constante et des mécaniques qui lui sont propres. J'ai également appris à maîtriser le champ lexical du monde du développement : commiter, pusher, merger, switcher sur le mock server, travailler en recette, les différentes branches dev, master, debug, preprod, release d'un git... des notions qui ne m'étaient pas complètement étrangères, mais qui restaient assez floues et m'empêchaient de comprendre la moitié des discussions de mes collègues les premières semaines.

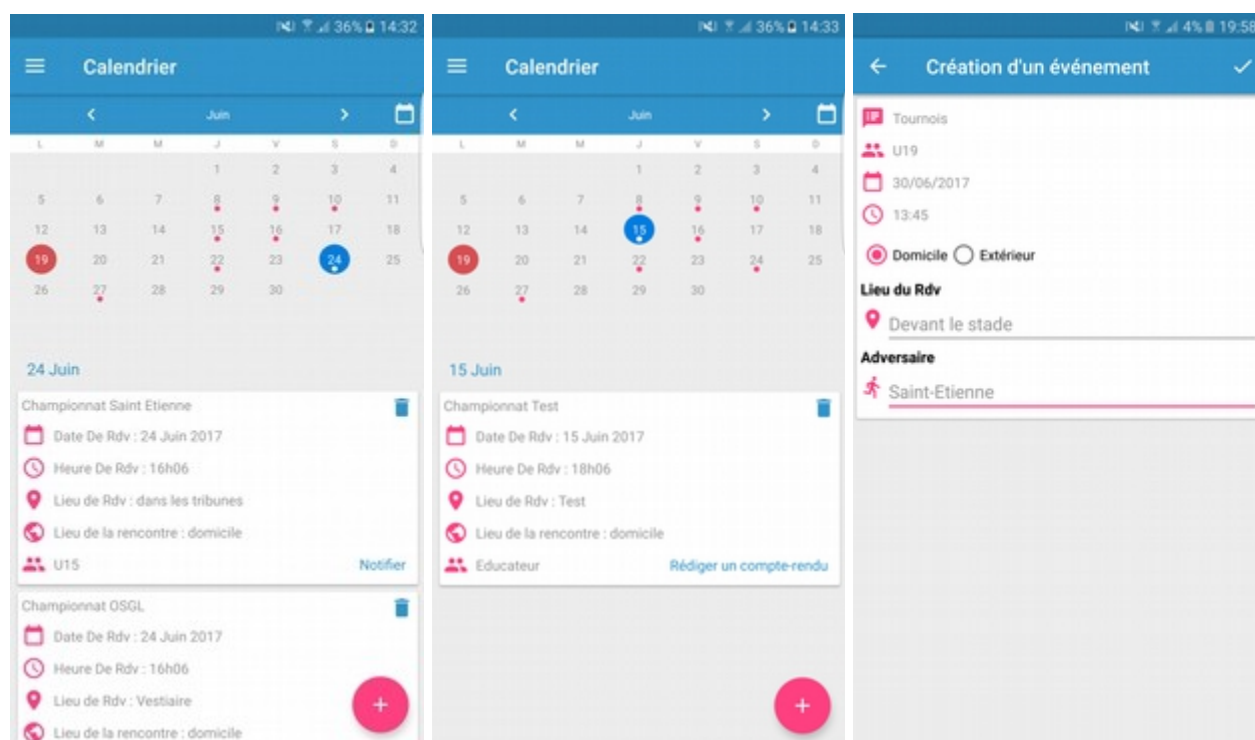
J'ai également appris à maîtriser des outils tels que Bitbucket, SourceTree, APlary, Postman ou encore MarvelApp, permettant un travail collaboratif et morcelé. Les enseignements que j'en ai tiré sont ma rigueur et mes automatismes concernant la gestion d'erreurs, qui se sont très fortement améliorés, car rien ne doit être laissé au hasard sur un projet d'une telle ampleur. Il est donc nécessaire d'essayer de se représenter toutes les possibilités d'utilisations et d'actions que l'utilisateur peut avoir afin de réduire au maximum le risque de crashes et d'exceptions. J'ai également eu l'impression que la très grande responsabilité que l'ont m'a accordé sur ce projet, sur lequel j'étais seul à travailler, m'a aidé sur ma réflexion et ma vigilance, car il est nécessaire d'anticiper les problématiques sur lesquelles nous risquons d'être confrontées avant de s'engouffrer dans la réalisation d'un algorithme qui peut rapidement s'avérer être une impasse, car la réflexion précédant son écriture n'avait pas été importante.

Pour finir, ce stage m'a réconforté dans mon choix de m'orienter vers du développement Android par la suite, car la mobilité et la modularité du système apportent des problématiques inédites et un terrain sur lequel de nombreuses innovations et bonnes idées restent encore à découvrir.

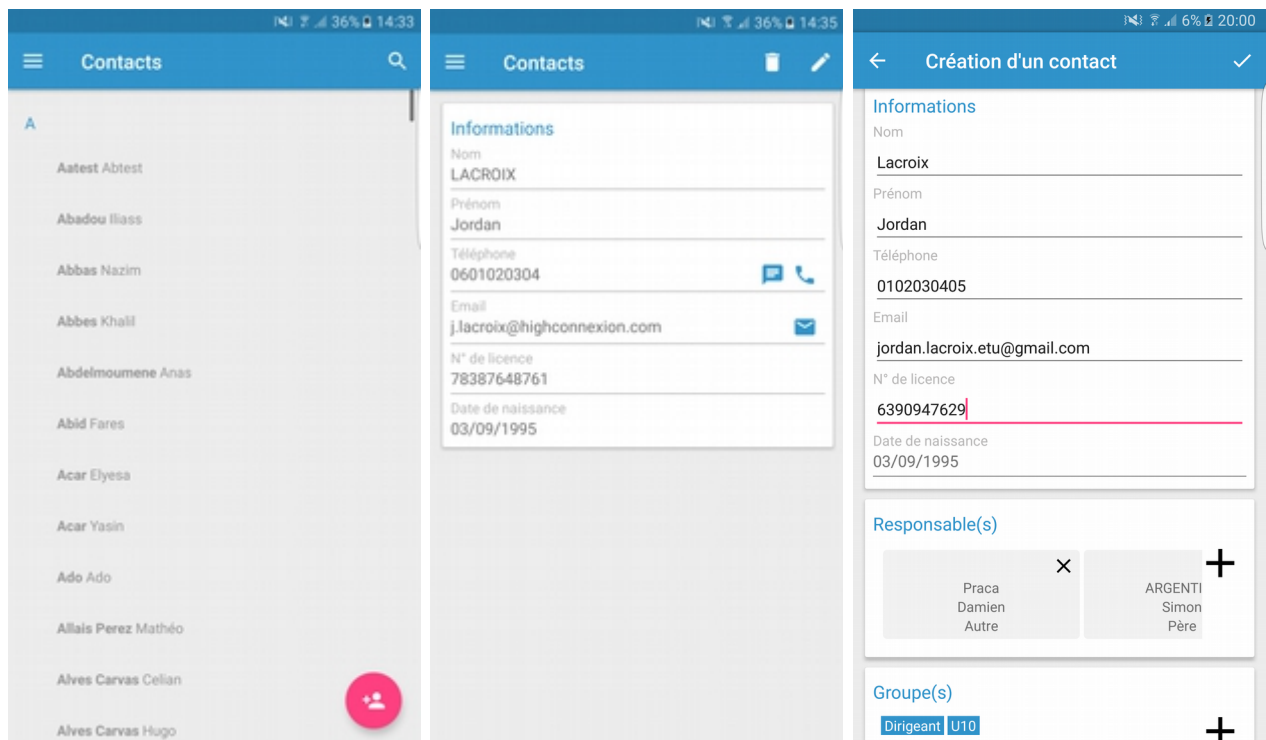
Annexes



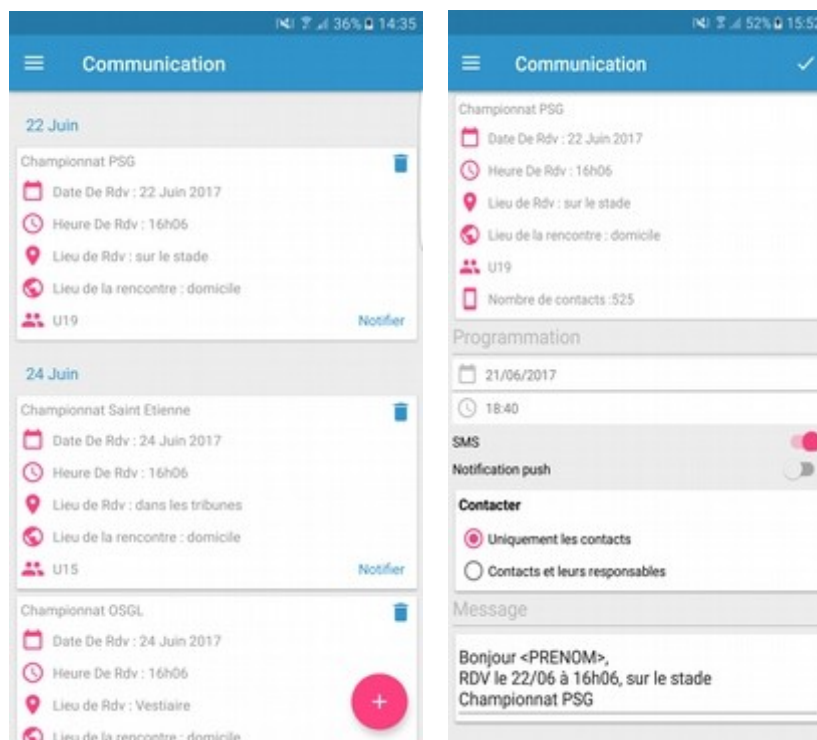
Captures d'écran de la partie « Articles et comptes-rendus »



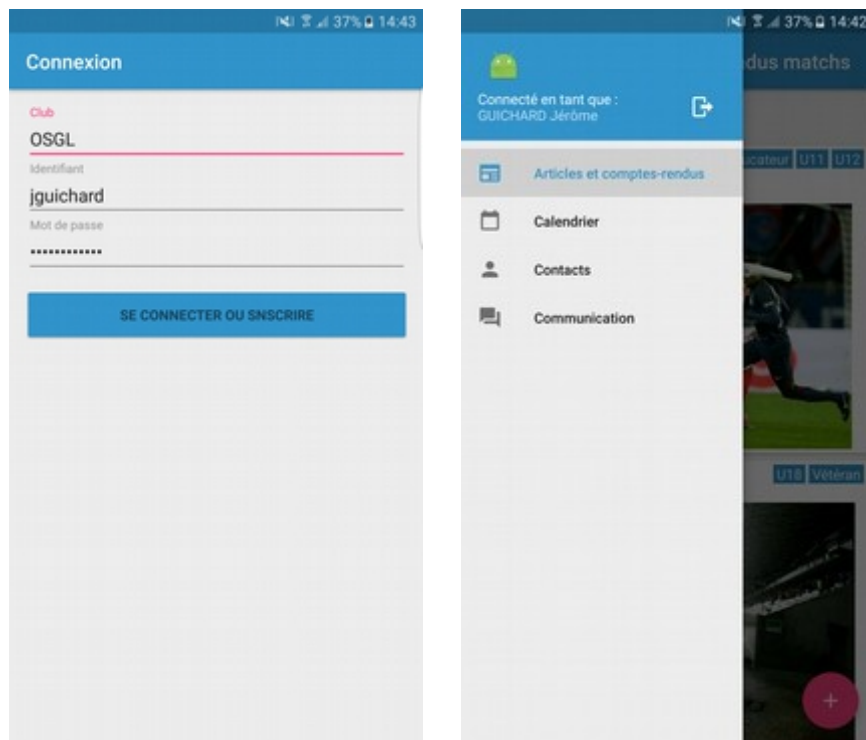
Captures d'écran de la partie « Calendrier »



Captures d'écran de la partie « Contacts »



Captures d'écran de la partie « Communication »



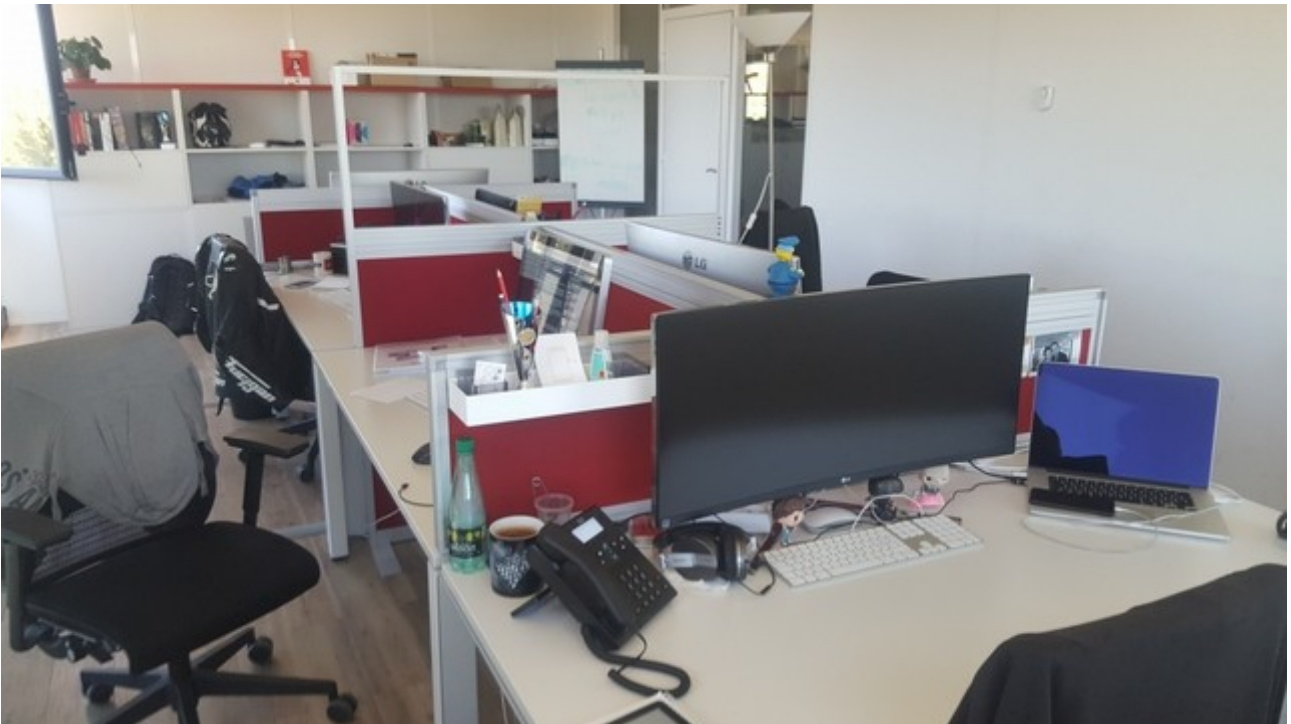
Captures d'écran de la fenêtre de connexion et du menu latéral



Locaux de HighCo – Extérieur



Locaux de HighCo – Accueil



*Locaux de HighCo – Open space de HighConnexion
(mon bureau fut celui en premier-plan à gauche)*



Locaux de HighCo - Réfectoire



Locaux de HighCo – Salle de repos

Index

Client/serveur : L'environnement client/serveur désigne un mode de communication à travers un réseau entre plusieurs programmes : l'un, qualifié de client, envoie des requêtes. L'autre ou les autres, qualifiés de serveurs, attendent les requêtes des clients et y répondent. Par extension, le client désigne également l'appareil sur lequel est exécuté le logiciel client, et le serveur, l'appareil sur lequel est exécuté le logiciel serveur.

Deadline : Date d'échéance à laquelle il faut rendre un projet.

Marque blanche : On parle de marque blanche lorsqu'une société propose un produit générique à ses clients, qui bénéficie donc du savoir-faire du fournisseur tout en ayant la possibilité de personnaliser le produit à son image (charte graphique, design, couleur...).

Natif / Native : Un programme informatique en code natif (ou langage machine) est composé d'instructions directement reconnues par un processeur.

Notifications Push : Une notification push est un message, lié à une application installée sur l'appareil qui n'est pas obligatoirement ouverte, envoyé à l'utilisateur d'un smartphone ou d'une tablette. À la différence d'un SMS qui a un coût de par sa liaison à l'infrastructure d'un opérateur, la notification push est gratuite.

Recette : Un « site recette » ou « une application recette » est une version réservée aux développeurs. Celle-ci est censé être dans un état finalisé et permet de vérifier la conformité du produit par rapport aux attentes du client avant sa mise en ligne finale.

Token : Un token pourrait être compris, en français, comme un « jeton d'authentification », qui permet d'identifier de façon unique et anonyme un utilisateur courant durant toute une session.

Web service : Le terme "web service" fait référence à des fonctionnalités de programmes informatiques permettant à des applications et des systèmes de fonctionner et d'échanger des informations à distance.

Bibliographie / Sitographie

Bibliographie :

- « Influencer le comportement du shopper », livre promotionnel réalisé et distribué par HighCo à tous les collaborateurs (*Aucune restriction de confidentialité sur celui-ci qui m'empêcherait de dévoiler certaines informations par inadvertance*)
- Rapport annuel de 2016 de la société HighCo : <https://www.highco.com/rapport-annuel/2016/index.html>

Sitographie :

- Site internet de HighCo : <https://www.highco.com/>
- Site internet de HighConnexion : <http://www.highconnexion.com/>
- Page Wikipédia de HighCo : <https://fr.wikipedia.org/wiki/HighCo>