

Projet interdisciplinaire ou de recherche

Conception et réalisation du app motivant ses
utilisateurs

Valentin STERN
Florian LAUER

Année 2015–2016

Projet réalisé pour Vladimir Latocha de l'équipe EDP du laboratoire IECN

Déclaration sur l'honneur de non-plagiat

Je soussigné(e),

Nom, prénom : STERN, Valentin

Élève-ingénieur(e) régulièrement inscrit(e) en 2^e année à TELECOM Nancy

Numéro de carte de l'étudiant(e) : 31205864

Année universitaire : 2015–2016

Auteur(e) du document, mémoire, rapport ou code informatique intitulé :

Conception et réalisation du app motivant ses utilisateurs

Par la présente, je déclare m'être informé(e) sur les différentes formes de plagiat existantes et sur les techniques et normes de citation et référence.

Je déclare en outre que le travail rendu est un travail original, issu de ma réflexion personnelle, et qu'il a été rédigé entièrement par mes soins. J'affirme n'avoir ni contrefait, ni falsifié, ni copié tout ou partie de l'œuvre d'autrui, en particulier texte ou code informatique, dans le but de me l'accaparer.

Je certifie donc que toutes formulations, idées, recherches, raisonnements, analyses, programmes, schémas ou autre créations, figurant dans le document et empruntés à un tiers, sont clairement signalés comme tels, selon les usages en vigueur.

Je suis conscient(e) que le fait de ne pas citer une source ou de ne pas la citer clairement et complètement est constitutif de plagiat, que le plagiat est considéré comme une faute grave au sein de l'Université, et qu'en cas de manquement aux règles en la matière, j'encourrais des poursuites non seulement devant la commission de discipline de l'établissement mais également devant les tribunaux de la République Française.

Fait à Nancy, le 19 mai 2016

Signature :

Déclaration sur l'honneur de non-plagiat

Je soussigné(e),

Nom, prénom : LAUER, Florian

Élève-ingénieur(e) régulièrement inscrit(e) en 2^e année à TELECOM Nancy

Numéro de carte de l'étudiant(e) : 31415656

Année universitaire : 2015–2016

Auteur(e) du document, mémoire, rapport ou code informatique intitulé :

Conception et réalisation du app motivant ses utilisateurs

Par la présente, je déclare m'être informé(e) sur les différentes formes de plagiat existantes et sur les techniques et normes de citation et référence.

Je déclare en outre que le travail rendu est un travail original, issu de ma réflexion personnelle, et qu'il a été rédigé entièrement par mes soins. J'affirme n'avoir ni contrefait, ni falsifié, ni copié tout ou partie de l'œuvre d'autrui, en particulier texte ou code informatique, dans le but de me l'accaparer.

Je certifie donc que toutes formulations, idées, recherches, raisonnements, analyses, programmes, schémas ou autre créations, figurant dans le document et empruntés à un tiers, sont clairement signalés comme tels, selon les usages en vigueur.

Je suis conscient(e) que le fait de ne pas citer une source ou de ne pas la citer clairement et complètement est constitutif de plagiat, que le plagiat est considéré comme une faute grave au sein de l'Université, et qu'en cas de manquement aux règles en la matière, j'encourrais des poursuites non seulement devant la commission de discipline de l'établissement mais également devant les tribunaux de la République Française.

Fait à Nancy, le 19 mai 2016

Signature :

Projet interdisciplinaire ou de recherche

Conception et réalisation du app motivant ses
utilisateurs

Valentin STERN
Florian LAUER

Année 2015–2016

Projet réalisé pour Vladimir Latocha de l'équipe EDP du laboratoire IECN

Valentin STERN
Florian LAUER
valentin.stern@telecomnancy.eu
florian.lauer@telecomnancy.eu

TELECOM Nancy
193 avenue Paul Muller,
CS 90172, VILLERS-LÈS-NANCY
+33 (0)3 83 68 26 00
contact@telecomnancy.eu

Institut Élie Cartan de Lorraine
Campus Aiguillettes, Université de Lorraine - Faculté des Sciences
et Technologies
54506, VANDOEUVRE-LES-NANCY CEDEX
+ 33 (0)3 83 68 45 64

Encadrant : Vladimir Latocha

Remerciements

Nous tenons à remercier Monsieur Vladimir Latocha pour son soutien et sa disponibilité tout au long du projet ainsi que pour ses conseils et les échanges d'idées intéressants que nous avons pu avoir lors des réunions.

Table des matières

Remerciements	vii
Table des matières	ix
1 Introduction	1
1.1 Contexte du problème	1
1.2 Enjeux	1
2 Analyse du problème	2
2.1 Causes de la situation insatisfaisante	2
2.2 Recherche d'éléments de motivation au sein de l'application	2
2.2.1 La motivation dans les jeux vidéos	2
2.2.2 La motivation selon les géants du web	3
2.2.3 Résultats de la recherche	3
2.3 Fonctionnalités de l'application	5
2.3.1 Écouter des mp3, achat et importation	5
2.3.2 Retours sur les leçons, commentaires, contact	6
2.3.3 Interactions entre les utilisateurs	6
2.3.4 Statistiques et objectifs	6
2.3.5 Blog ou Forum	6
2.3.6 Autres fonctionnalités	6
3 Organisation	7
3.1 Méthode de travail	7
3.2 Technologies utilisées	7

3.3	Répartition des tâches	9
3.4	Technique	9
4	Solution proposée	12
4.1	Résultats actuels	15
4.2	Perspectives	15
	Liste des illustrations	16

1 Introduction

1.1 Contexte du problème

La méthode Feldenkrais est une méthode inventée dans les années 1940 visant à faire prendre conscience aux pratiquants les mouvements de leur corps dans l'espace et dans l'environnement. Elle peut apporter plus de souplesse dans les articulations ainsi que la flexibilité de la colonne vertébrale, tout autant que de fournir une organisation plus aisée des mouvements dans la vie quotidienne ou durant une pratique sportive. Elle améliore donc la respiration, les mouvements et tous les muscles et articulations du corps humain. (source Wikipédia)

Il existe un site internet permettant de récupérer des leçons de la méthode Feldenkrais et qui a été réalisé par notre encadrant M. Latocha. Ce site dispose de fonctionnalités pour acheter une leçon ou un pack de leçons en ligne et les écouter par la suite sur son téléphone, ou sur son ordinateur. Cependant, cette méthode d'accès n'est pas la méthode la plus pratique pour les utilisateurs, car il peut s'avérer long de l'acheter puis ensuite de la télécharger.

1.2 Enjeux

Les objectifs essentiels de ce PIDR sont dans un premier temps de découvrir le domaine de la recherche, et ici, en quoi la recherche peut permettre de faire en sorte qu'une application soit motivante pour ses utilisateurs. Mais dans un deuxième temps, il était également demandé de réaliser un prototype sous la forme d'une webview, qui pourra être proposé aux utilisateurs de la méthode ; ceux-ci, par la suite pourraient nous transmettre les retours sur l'utilisation de l'application : les points positifs aussi bien que négatifs, pour en dégager des améliorations possibles. Un enjeu majeur sera que les pratiquants de la méthode puissent accéder à leurs leçons via un mobile ou une tablette -et donc, où ils veulent- car la pratique peut très bien se faire en extérieur ou alors souvent dans une salle n'ayant pas de poste informatique. Un autre enjeu important serait de pouvoir atteindre davantage d'utilisateurs. L'enjeu pour les pratiquants est donc la liberté de l'utiliser partout pour rendre la pratique beaucoup plus facile pour eux, mais également les rendre régulier dans leur pratique. L'objectif de notre encadrant est de faire partager la méthode Feldenkrais à un maximum de personne. Notre objectif personnel a été de réaliser une application mobile, car ce domaine nous intéresse et car nous n'avions jamais réalisé d'application mobile complète auparavant.

2 Analyse du problème

2.1 Causes de la situation insatisfaisante

Il n'y a pas la possibilité d'écouter les leçons de la méthode Feldenkrais partout. De plus, les pratiquants peuvent être démotivés car les leçons sont longues et il est parfois difficile de trouver le temps de les réaliser, ou de penser à les réaliser par rapport au stress que peut apporter la vie active. Ces raisons rendent l'utilisation de cette méthode peu répandue, et réservée aux seules personnes qui la connaissent. La cause principale de cette situation est donc le manque de popularité de la méthode Feldenkrais car elle est très peu connue, relativement récente et a du mal à se démocratiser. Aussi les personnes qui l'utilisent peuvent la tester et s'en lasser rapidement par la suite, car elles ne sont pas assez motivées pour continuer.

2.2 Recherche d'éléments de motivation au sein de l'application

Un point important de ce projet a été de rechercher des moyens de motiver les personnes à utiliser la méthode Feldenkrais. L'essentiel de la recherche associée à ce projet consiste en ce point. De nombreuses idées nous sont parvenues grâce à des réunions mettant en relation différentes expériences personnelles par rapport aux leviers de motivation rencontrés dans la vie de tous les jours.

2.2.1 La motivation dans les jeux vidéos

Nous avons commencé par comparer l'application à un jeu vidéo. En effet, un bon jeu vidéo et différencié d'un mauvais jeu vidéo par sa capacité à motiver le joueur à continuer celui-ci. On peut retrouver des jeux très difficiles, comme par exemple Dark Souls, qui poussent le joueur dans ces derniers retranchements mais qui leur font éprouver une grande satisfaction. On peut également trouver des jeux de construction qui nous motivent à continuer à construire pour finir notre projet, par exemple Minecraft. Ce dernier ne dispose pas de but particulier et le joueur peut donc en trouver lui même et réaliser ce qu'il a envie.

Chamberlain identifie 4 archétypes de joueurs principaux : Les explorateurs, les sociaux, les leaders et les killers. Ce dernier ne sera pas exploités dans ce projet, car le principe des killers et réussir à devenir premier en descendant les autres joueurs, parfois à tout prix. Les explorateurs souhaitent découvrir tous les secrets, récupérer chaque objet possible pour les collectionner ou encore aller là où personne n'est encore allé. Ceux-ci seront à nouveau peu utilisés dans notre ap-

plications. Cependant, l'ajout de secrets permet en général de faire venir les utilisateurs sur une application, comme par exemples les secrets de Google ou Youtube. Les sociaux, eux, trouvent de l'intérêt à jouer à un jeu si ils peuvent interagir avec d'autres personnes. Il vont donc en général préférer les jeux en lignes massivement multijoueurs. Enfin, les leaders veulent arriver en haut du classement en effectuant les différentes actions pour progresser, gagner des points et battre les autres. Leur satisfaction provient de leur nom affichés à tous après un dur labeur pour progresser.

2.2.2 La motivation selon les géants du web

Après nous être penchés sur les jeux vidéos, nous nous sommes demandés pourquoi Facebook, Amazon ou encore Tinder et Snapchat sont autant utilisés de nos jours.

Quel sont les outils qu'utilisent ces géants de l'Internet pour amener toutes ces personnes à utiliser leurs site internet, et comment font ils pour les fidéliser ainsi ?









Site internet								
Rang mondial (visites)	2	3	6	7	10	39	50	NC
Estimations nombre visite par mois (M)	2668	1883	1038	909	669	195	165	48
Chiffre d'affaires (M de \$)	4000	266	107000	2,7	710	4400	NC	3500

FIGURE 2.1 – Quelques chiffres sur les géants du net

Nous avons analysé les plus vastes sites internet mondiaux. Tout le monde les connaît ou en a déjà entendu parlé, à part peut-être Stackoverflow, qui est un forum d'entraide concentré sur la programmation dans tous les langages, et Steam, qui est une plate-forme de vente de jeux vidéos dématérialisés. Ces sites ont des points communs et des particularités que nous avons décortiqués afin d'en extraire des idées, tout en essayant à chaque fois de mesurer leur pertinence par rapport à notre application. Il en a résulté la figure suivante.

2.2.3 Résultats de la recherche

Un concept important est la gratuité de l'application, au moins au début de l'utilisation ou pour son lancement. Si l'utilisateur doit payer pour y accéder, il aura moins tendance à l'essayer. Fournir un certain contenu gratuit dès son inscription est également un moyen de le motiver, et forcer un achat risque de faire fuir l'utilisateur. Il faut toujours attirer les utilisateurs et réussir à les fidéliser petit à petit pour leur montrer qu'ils ont besoin de l'application qu'ils utilisent.

Dans la continuité de ce principe, on peut également offrir des cadeaux aux utilisateurs, ou des réductions qui arrivent petit à petit en fonction de leur fréquence d'utilisation de l'application. Ils seront ainsi plus motivés à continuer et obtenir ces réductions, même si au départ cela ne leur était pas indispensable.

Une autre idée importante qu'on a pu dégager de ces sites internet est l'accomplissement de soi et la satisfaction personnelle que cela peut nous apporter. Internet met en contact des milliers de

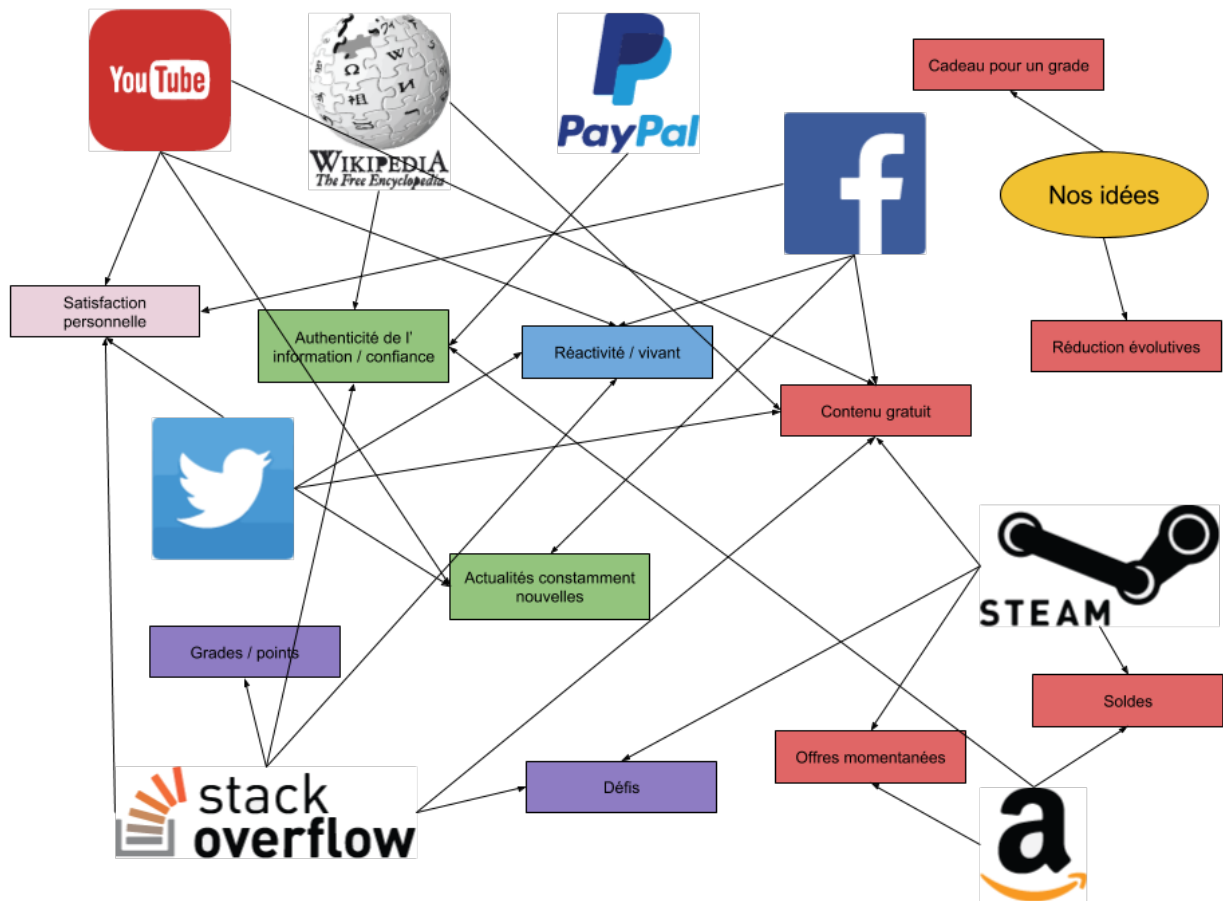


FIGURE 2.2 – Idées tirées des sites

personnes et l'on peut aider les autres assez rapidement si on en a l'envie. Sans obligations, de nombreuses personnes vont aider les autres car cela leur fait juste plaisir.

Les défis peuvent pousser les utilisateurs à se surpasser, et effectuer des tâches sans s'en rendre compte. On trouve ce principe dans les succès sur Steam, ces actions à effectuer dans un jeu qui n'apporte rien d'autre que la fierté d'avoir réussi à les réaliser. Cette idée est liée avec l'idée de grades et de points, permettant de mettre en confrontation les utilisateurs et les pousser à devancer les autres. Nous allons grâce à ces deux idées compléter les attentes des personnes correspondant aux archétypes des leaders et des explorateurs.

2.3 Fonctionnalités de l'application

Il nous a fallu organiser nos tâches par niveau de priorité afin de ne pas se perdre dans le code, c'est pourquoi nous avons fait cette carte heuristique, d'abord, sur un tableau à l'école, puis nous avons numérisé toutes ces idées.

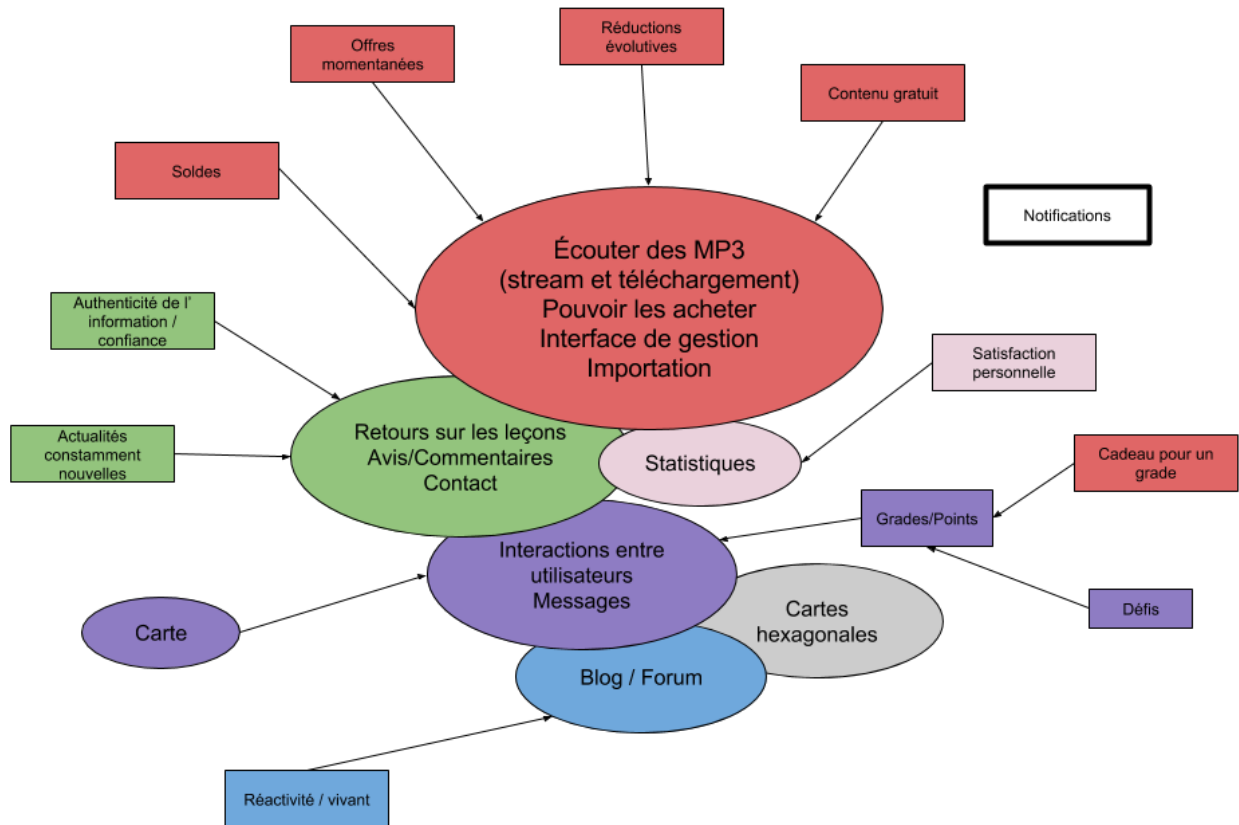


FIGURE 2.3 – Carte heuristique de l'application

2.3.1 Écouter des mp3, achat et importation

La fonctionnalité primaire de l'application était bien évidemment d'écouter les mp3 des leçons de notre encadrant sur l'application. Cependant, les lecteurs de musiques réalisés avec Ionic n'étaient pas très exploitables (style non adapté, ou ne fonctionnaient pas) il a donc fallu en réaliser un "home made". Comme les mp3 ne sont pas gratuits pour la plupart, il s'agissait aussi de pouvoir les acheter, nous avons donc réfléchi aux possibilités de paiement via une application et deux solutions étaient envisageables : Paypal, ou un paiement intégré à l'application passant par le Playstore (ou l'AppStore pour les utilisateurs d'iPhone). (import ???)

2.3.2 Retours sur les leçons, commentaires, contact

Ensuite, il était évident que l'application ait un côté social, pour que l'utilisateur puisse se sentir écouté, que la communauté soit active, et que certains des points de motivation soient bien présents. C'est pourquoi il est possible de laisser des commentaires sur leçons ainsi qu'une note entre 0 et 5. Aussi, sur le site déjà existant il était possible d'envoyer un mail à M. Latocha via la partie contact, et nous avons pensé qu'il était important que cette partie demeure présente sur l'application. (confirmer quand sera fait ? ? ?)

2.3.3 Interactions entre les utilisateurs

Outre les commentaires, une idée primordiale selon nous pour rendre une application motivante était les échanges entre utilisateurs, avec la possibilité d'envoyer des messages privés ; les profils ne seraient alors pas anonymes et chaque utilisateur posséderait un compte personnel et une page profil (comme sur la plupart des réseaux sociaux actuels).

2.3.4 Statistiques et objectifs

Une autre idée qui nous est venue à l'esprit était d'afficher des statistiques, car ils sont un moteur de la motivation ; en effet, un utilisateur qui voit qu'il a effectué un cours et observe une évolution de son graphique personnel, ainsi que l'augmentation d'un score qui lui serait attribué en fonction des heures qu'il passe à faire des cours, peut lui donner envie de pratiquer encore plus, ou de revenir chaque jour sur l'application.

2.3.5 Blog ou Forum

On arrive maintenant dans les fonctionnalités un peu plus facultatives, mais auxquelles on a pensé car une partie identique était déjà présente sur le site existant mais aussi car c'est le principe de base du site Stackoverflow. Le principe serait de remplir cette partie avec des topics contenant une question suivi d'une succession de réponses pouvant être écrites par tous les utilisateurs. Enfin, nous avons pensé à intégrer une carte sur l'application car la localisation est une option présente sur beaucoup de réseaux sociaux, et, dans notre cas, elle apporterait l'avantage de savoir qui pratique aussi le Feldenkrais autour de chez soi : les utilisateurs pourraient alors organiser des rencontres entre eux pour pratiquer ou débriefer sur des leçons.

2.3.6 Autres fonctionnalités

Dans les fonctionnalités également facultatives nous avons pensé à des cartes hexagonales pour présenter les cours, dans le style d'un jeu vidéo (les cours seraient une sorte de quête à accomplir pour avancer comme sur un plateau de jeu) et les couleurs des cours pourraient s'actualiser quand celui-ci est acheté, et ensuite accompli. Néanmoins c'est une tâche demandant beaucoup de boulot et cette fonctionnalité serait peu adaptée aux petits écrans.

3 Organisation

3.1 Méthode de travail

Nous avons dans un premier temps défini les leviers de motivation que l'on voulait inclure dans l'application. Nous avons ratissé toutes les possibilités que l'on a pu imaginé durant cette période pour pouvoir avoir une vue globale de toute ce qui était utilisé. Nous avons ainsi pu choisir au mieux les principes à adopter. Nous avons ensuite sélectionner ceux qui nous paraissaient les plus pertinents. A partir de cette liste de critères, nous avons imaginé les fonctionnalités qui pouvait s'y appliquer. L'intérêt de cette méthode et de pouvoir savoir pour une fonctionnalité sera présente dans l'application plutôt qu'une autre, et ainsi savoir au mieux l'impact que celle-ci va avoir sur l'utilisateur. Ainsi, une liste de toutes les fonctionnalités principales a été crée. Celles-ci ont été ensuite regrouper dans différents groupes, en fonction du thème de celle-ci, et de son importance dans le projet. Nous voulions partir d'un noyau de base constitué des fonctionnalités les plus importantes. A ce noyau ce sont rattaché les autres fonctionnalités de moins en moins prioritaires. Ceci nous a donc permis de développer les fonctionnalités dans leur ordre d'importance et ainsi arriver petit à petit au prototype qui était prévu.

3.2 Technologies utilisées

Le résultat qui était à fournir à l'issu de ce projet était un prototype d'application. Nous avons donc choisi des technologies permettant de réaliser rapidement un prototype qui fonctionne. Pour la partie application, nous avons utilisé Ionic. Cette technologie permet de programmer une application en utilisant les technologies du web, c'est à dire HTML, CSS et Angular.JS principalement. Elle est également gratuite et dispose une belle interface utilisateur. Elle est Open Source et son ergonomie ressemble grandement à une application native. De plus, sa communauté active permet la résolution rapide des problèmes rencontrés. Elle dispose de fonctionnalités natives qui y sont intégrées et est également compatible avec la technologie Cordova qui nous as permis de disposer de toutes les fonctionnalités natives d'Android, comme par exemple l'écoute de fichiers audios, qui est la fonctionnalité principale de l'application.



FIGURE 3.1 – Logo de Ionic



FIGURE 3.2 – Logo de Node.JS

Pour ce qui concerne le serveur, nous avons utilisé le langage Node.JS. Celui-ci permet d'avoir une API REST, c'est-à-dire un certain nombre d'URL correspondant chacune à une tâche spécifique à effectuer, comme par exemple la création d'un élément ou la récupération de tous les éléments de la base de données. Ainsi, chaque fois que l'on envoie une requête au serveur, celui-ci va récupérer les données nécessaires, effectuer des traitements sur celles-ci et modifier la base de données s'il le faut, puis renvoyer le résultat au client, c'est-à-dire l'application mobile. Un avantage de ce principe de serveur est qu'il pourra tout aussi bien être utilisé dans une application mobile qu'un site web.

Grâce à ce serveur, nous avons pu développer, en plus de l'application mobile, une interface d'administration permettant à notre encadrant de créer et uploader les leçons sur le serveur et qui sont directement accessibles via l'application. Pour réaliser le design de cette partie d'administration, nous avons utilisé le framework materializecss. Celui-ci nous a permis de rendre l'interface épurée et claire, permettant ainsi de gérer au mieux les cours disponibles pour les utilisateurs.



FIGURE 3.3 – Logo de Materialize



FIGURE 3.4 – Logo de MariaDB

Pour stocker les données de l'application, nous avons utilisé le système de gestion de base de données MariaDB. Celui-ci est Open Source et gratuit, et est utilisé par un grand pourcentage des sites internet car il est très répandu et sa communauté est grande et active. Afin de faire la liaison entre Node.JS et MariaDB, nous avons utilisé l'ORM Sequelize qui permet d'utiliser le principe d'Active Record qui lie ainsi tables de la base de données et objets en mémoire. Ainsi, une table de la base correspond exactement à un objet sur lequel des fonctions de recherche ou de sauvegarde ont été rajoutées. Nous pouvons ainsi modifier ces objets et les sauvegarder directement dans la base de manière aisée et efficace. De plus, si nous avons une association entre plusieurs tables, l'objet correspondant possède également les méthodes pour récupérer les autres objets associés.

Les données utiles transitent entre ces deux parties sous le format JSON qui est un format de plus en plus courant de nos jours depuis l'apparition de Node.JS. En effet, ce format permet de communiquer facilement entre le client et le serveur car ils sont tous les deux basés sur du javascript, et ce dernier arrive très bien à comprendre le JSON et utiliser le contenu des données en tant que variables. Cette étape de communication est donc très rapide

et ne nous encombre pas avec des fonctionnalités de parsing à rajouter.

3.3 Répartition des tâches

La répartition des tâches changeaient de temps en temps pour nous permettre d’avoir une vue globale de l’application. Nous ne voulions pas qu’une seule personne soit au courant d’une seule partie de l’application sans avoir d’idée sur le reste de celle-ci. Nous voulions donc être informés à tout moment de ce qui était fait, et nous avons beaucoup travaillé de paire. Un bon exemple est la réalisation d’une quelconque fonctionnalité. La plupart du temps, nous réfléchissions ensemble sur les informations nécessaires à cette fonctionnalité, puis à une interface qui lui correspondrait. Ensuite les deux parties, application et serveur, étaient codées en parallèle. Enfin, nous faisons la liaison entre les deux et nous corrigeons respectivement les erreurs.

3.4 Technique

Nous avons séparé tout notre code dans 3 gits différents. Ainsi, nous avons un dépôt qui concernait l’application mobile, et un autre concernant le serveur. Nous testions tout d’abord l’application grâce à une sorte d’émulateur offert par ionic qui permet d’ouvrir notre application dans un navigateur web. Cependant, certaines fonctionnalités natives comme la lecture de musique était indisponible durant le test sur navigateur. Il nous fallait forcément tester l’application sur nos téléphones portables. Le fait d’avoir séparer le serveur de l’application mobile nous a permis de déployer le serveur sur un serveur de test afin de l’utiliser avec l’application mobile. De ce fait, notre téléphone pouvait se connecter à ce serveur web car il n’était pas hébergé sur nos ordinateurs, inaccessibles de l’extérieur. Le dernier dépôt quant à lui nous a servi pour l’élaboration du rapport. Il est plus logique de le séparer du reste du code car il ne le concerne pas directement.

Le dossier principal concernant le développement est le dossier `www`, car il contient tous les fichiers nous permettant de donner vie à l'application. Les autres dossiers sont principalement là pour faire marche de manière général ionic et les différents plugins qui ont pu être ajoutés. Le dossier `templates` nous permet de définir toutes les vues de l'application, la forme de celle-ci. Ainsi, toutes les pages visibles par l'utilisateur sont dans ce dossier. Le dossier `js` nous permet quant à lui de définir les différentes actions associées à ces vues, c'est-à-dire tous les contrôles disponibles, le fond de l'application. Les deux parties sont liés dans les templates. Nous pouvons définir pour chaque boutons quelle est la fonction qui sera exécuté lors du clic sur celui-ci. Ainsi, lorsque qu'un utilisateur réalise une action, une fonction contenu dans fichier du dossier `controllers` va se mettre en marche. Si celle-ci a besoin de joindre le serveur afin de récupérer des données ou d'en ajouter dans la base de données, elle fera appel à une autre fonction contenu dans le dossier `services`. Ainsi, l'application est découpée selon le patron de conception Modèle (services) - Vue (templates) - Contrôleur (controllers).

Les dossiers `img` et `css` contiennnent les différentes images visibles sur le site, ainsi que les éléments de design spécifiés en CSS. Le fichier `index.html` contenu à la racine du dossier `www` permet d'inclure tous les fichiers que nous avons réalisé dans tous les autres dossier et ainsi lier l'application ensemble. Celui-ci définit également les menus permettant d'accéder au vue.

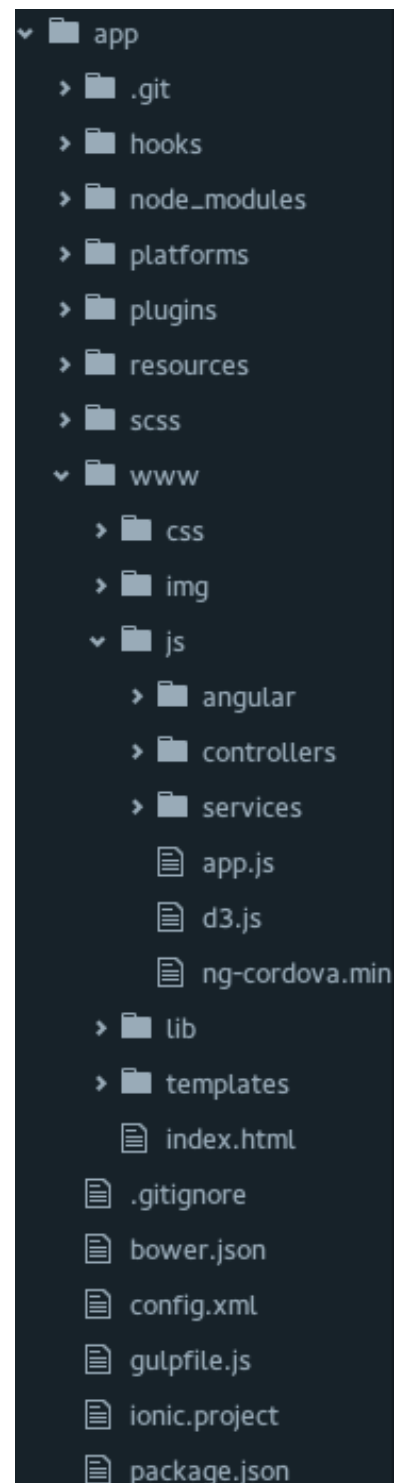


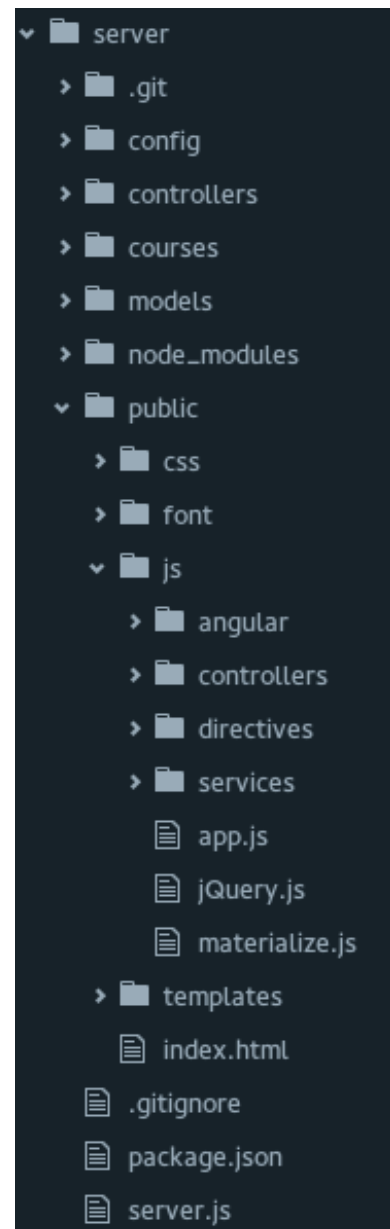
FIGURE 3.5 – Arborescence de l'application

Le dossier templates correspond au dossier www qui se situe dans l'arborescence de l'application. Il est constitué des mêmes fichiers et du même fonctionnement et ne sera pas donc plus explicité ici.

Le dossier config contient les différents fichiers de configuration. C'est-à-dire un fichier comportant les différents paramètres d'accès pour la connexion à la base de données. Il contient un fichier "router" définissant toutes les routes accessibles dans l'API. Chaque route est liée à une fonction d'un contrôleur qui va effectuer le traitement des données envoyées. Chacune des routes possède un paramètre indiquant si oui ou non l'utilisateur doit être authentifié pour y accéder, nous permettant de couper correctement l'accès aux personnes non autorisées. Ces routes nous permettent donc de bien structurer notre application pour la rendre la plus propre possible.

Du fait que le serveur soit une API REST le serveur ne prend que les parties Modèle (models) et Contrôleur (controllers) du patron de conception MVC. Le dossier models définit les différents objets que nous aurons besoin de stocker. Dès que nous lançons le serveur pour la première fois, Sequelize va parcourir ces fichiers pour créer les tables de la base de données correspondantes à chacun des objets. C'est fonctionnalité nous est très utile lors du développement car nous n'avons pas à nous soucier de la synchronisation de nos deux bases de données, et également lors du déploiement sur un serveur.

Les contrôleurs quant à eux vont recevoir la requête du client, qui est passée par les routes, et aura la possibilité de récupérer les paramètres de celle-ci. Une vérification de ces paramètres va être effectuée afin d'être sûr qu'ils sont corrects. Par la suite, il va traiter ces paramètres, soit en faisant une recherche, soit en insérant des données dans la base. Par la suite, il va renvoyer une réponse qui correspond à ce que le client attend, comme par exemple une liste d'objets qui seront affichés.



4 Solution proposée

Quand l'utilisateur lance l'application, il va arriver sur l'interface ci-contre. Il pourra alors choisir de s'inscrire s'il n'a pas déjà un compte, ou de se connecter à un compte existant.

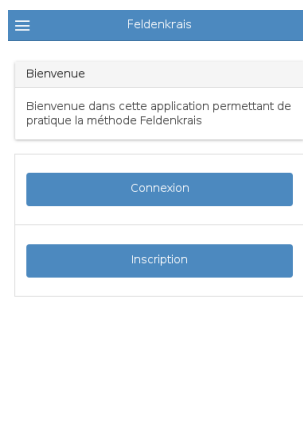


FIGURE 4.1 – Accueil

Si il choisi de s'inscrire, il pourra alors entrer un login, son adresse email pour pouvoir être contacté par la suite s'il poste des commentaires, et son mot de passe ainsi que la confirmation de celui-ci afin de s'assurer qu'il ne s'est pas trompé en l'écrivant. En s'inscrivant, le serveur va créer son compte et le connecter directement après.

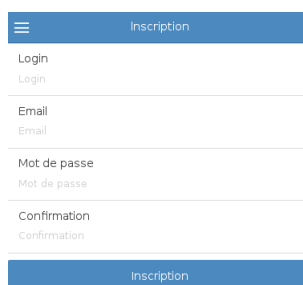
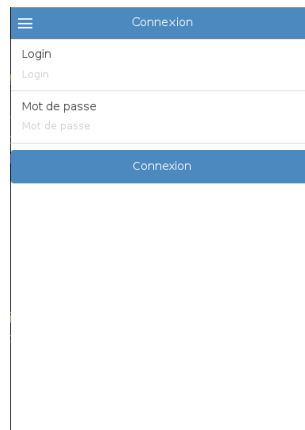


FIGURE 4.2 – Inscription

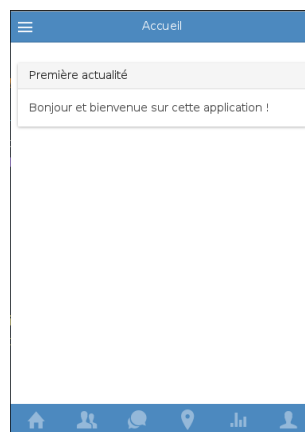
Si l'utilisateur choisit de se connecter, il pourra entrer son login ainsi que son mot de passe. Une fois connecté, il n'aura plus besoin de se connecter à chaque fois qu'il lance l'application et pourra également y accéder sans avoir accès à Internet.



The screenshot shows a mobile application interface for login. At the top, there is a blue header bar with a hamburger menu icon on the left and the word "Connexion" in the center. Below the header, there are two input fields: the first is labeled "Login" and the second is labeled "Mot de passe". Both fields have placeholder text that matches their respective labels. Below these fields is a blue button labeled "Connexion". The background of the screen is white.

FIGURE 4.3 – Connexion

Une fois connecté, l'utilisateur arrive sur la liste des actualités. Celles-ci sont créées par l'administrateur, c'est à dire notre encadrant.



The screenshot shows the home screen of the mobile application. At the top, there is a blue header bar with a hamburger menu icon on the left and the word "Accueil" in the center. Below the header, there is a list of news items. The first item is titled "Première actualité" and contains the text "Bonjour et bienvenue sur cette application !". Below this list, there is a blue footer bar with several icons: a home icon, a group of people icon, a speech bubble icon, a location pin icon, a bar chart icon, and a person icon. The background of the screen is white.

FIGURE 4.4 – Actualités

Il pourra ensuite aller sur la liste des packs de leçons, lui permettant de choisir un pack gratuit qu'il débloquent par défaut, ou de parcourir les différents packs payants qui sont disponibles et les acheter s'ils le désirent.

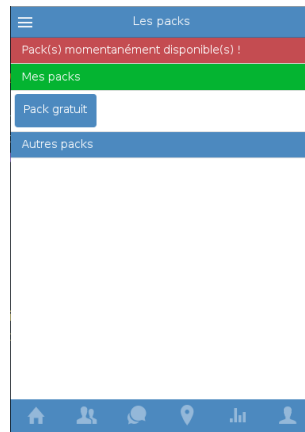


FIGURE 4.5 – Liste des packs de leçons

Il peut également aller sur la page de stats afin de renseigner son objectif à accomplir, c'est à dire un nombre de leçon à faire par jour, par semaine ou par mois, selon son choix.



FIGURE 4.6 – Objectif

4.1 Résultats actuels

Nous disposons actuellement d'un prototype d'application permettant d'écouter des leçons en streaming. Elle permet aussi d'avoir des statistiques par rapport à l'objectif que l'on s'est fixé, afin de se motiver et d'effectuer des leçons régulièrement.

La partie d'administration est composée de la création des différents packs de cours et les cours avec pour information le prix du pack, ainsi que sa disponibilité, le nom du pack, le nom de chaque cours, et leur durée et le fichier audio associé. Il est également possible d'ajouter des actualités qui apparaîtront à tous les utilisateurs lors de leur connexion.

Ce projet nous a permis personnellement de mieux utiliser la technologie Ionic et également se spécialiser dans AngularJS. Nous avons également découvert des outils permettant de gérer au mieux la conception et la réalisation d'un projet, et ainsi de savoir s'organiser et ne pas se perdre dans l'étendu de celui-ci, et dans les contraintes techniques associées.

4.2 Perspectives

Les principales perspectives de l'application consiste à développer les différentes fonctionnalités qui ont été imaginées mais qui ne sont pas encore présentes dans celle-ci. Nous pourrions également mettre le prototype à l'essai pour quelques pratiquants afin qu'ils le testent quelques temps pour obtenir des retours et ainsi savoir quelles améliorations nous pourrions faire.

Liste des illustrations

2.1	Quelques chiffres sur les géants du net	3
2.2	Idées tirées des sites	4
2.3	Carte heuristique de l'application	5
3.1	Logo de Ionic	7
3.2	Logo de Node.JS	8
3.3	Logo de Materialize	8
3.4	Logo de MariaDB	8
3.5	Arborescence de l'application	10
3.6	Arborescence du serveur	11
4.1	Accueil	12
4.2	Inscription	12
4.3	Connexion	13
4.4	Actualités	13
4.5	Liste des packs de leçons	14
4.6	Objectif	14