

# Android

## Projet BrainWave



**DAGGAG Chafik, DIEUDONNE Loïc,  
HAYART Robin, KERGALL Cécile**  
IUT Orléans – Licence Professionnelle  
RESTEL option Réseau et  
Télécommunication – LP2

# Sommaire

|   |    |
|---|----|
| I.Présentation du projet BrainWave.....                     | 3  |
| II.Développement.....                                       | 4  |
| 1.Connexion en Bluetooth entre le casque et l'appareil..... | 5  |
| 2.Récupération des données du casque.....                   | 6  |
| 3.Génération d'un graphique d'après les données.....        | 6  |
| 4.Ajout d'un menu de paramètres.....                        | 7  |
| 5.Génération CSV d'après un enregistrement des données..... | 8  |
| III. Problèmes rencontrés.....                              | 9  |
| IV.Conclusion.....  | 10 |

# I. Présentation du projet BrainWave

BrainWave est une application Android qui se connecte à des capteurs d'ondes cérébrales de type MindPlay. L'application récupère les données et les stocke localement. Une fois connectée au capteur, les données sont récupérées puis dessinées sous forme de courbe dans un graphique. De plus, nous pouvons créer un fichier CSV avec les données recueillies. Par ailleurs, la démarche inverse est possible, nous pouvons convertir un fichier contenant les données sur un CSV et les afficher sur un graphique. Une autre fonctionnalité que l'application offre est une gestion du Bluetooth via une interface graphique à base de bouton.

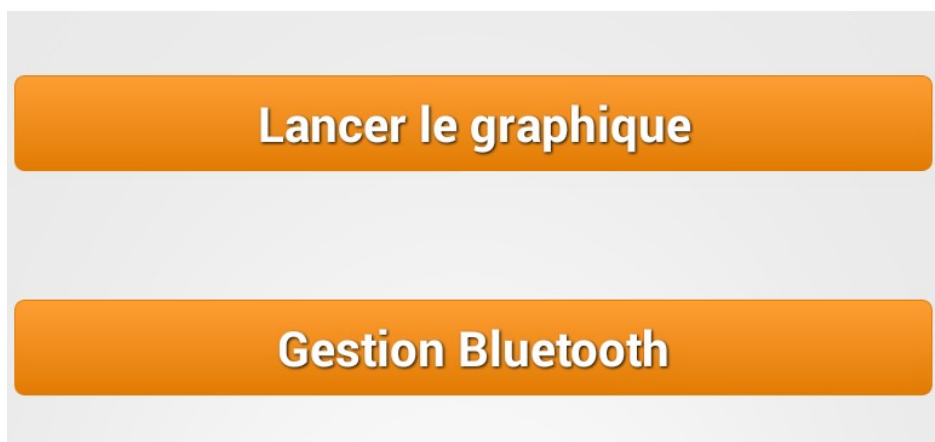
L'application se connecte en Bluetooth sur le capteur. Les données recueillies doivent être facilement compréhensibles et l'application doit être claire pour un utilisateur lambda. Par ailleurs, un export en CSV doit être possible. Pour représenter les différents niveaux de concentration et de méditation nous avons décidé d'utiliser des courbes dans des graphiques plutôt que par compteurs comme sur l'application MindWave. Ces courbes offrent une meilleure lisibilité que les compteurs, ainsi que la possibilité d'observer les données dans le temps, ce que les compteurs n'offrent pas. Nous avons donc recherché une librairie permettant de réaliser des graphiques et des courbes, nous avons trouvé [GraphView](#). De plus, grâce à l'option d'enregistrement des données des courbes dans un fichier csv, nous pouvons sauvegarder des enregistrements ainsi que comparer les courbes de deux personnes.

## II. Développement

Pour commencer le projet nous avons tous eu besoin de nous réadapter avec la programmation Android, voir pour certains qui n'avaient encore jamais pratiqué de simplement l'apprendre. Pour cela nous avons commencé par créer de simples applications de type « hello world », suivi des tutoriels sur le développement Android et étudié des projets Android existants, que des membres du groupes avaient réalisé précédemment dans leur études. Au début du projet, étant donné que nous n'avions pas encore de capteurs d'ondes cérébrales à portée de main, nous nous sommes focalisés sur la connexion en Bluetooth sous Android.

Il a aussi fallu mettre en place sur nos machines personnelles l'environnement Android et faire en sorte que chacun d'entre nous soit capable d'utiliser et de développer avec les tablettes qui nous ont été fournies. Quelques temps furent nécessaires pour bien paramétrer chez chacun d'entre nous les drivers des tablettes, pour pouvoir installer et tester notre application directement sur les tablettes, ainsi que l'installation des différents SDK Android et les réglages d'Eclipse. Nous avons également mis en place un projet Github pour que chaque membre du groupe puisse directement depuis Eclipse envoyer son avancement du projet et récupérer l'avancement des autres.

Au fur et à mesure du développement nous avons fusionné nos travaux dans notre application Brainwaves. Nous avons obtenu une application décomposée en trois grandes activités. À son lancement l'application affiche un menu composé de deux boutons qui proposent le choix entre le gestionnaire de Bluetooth ou le lancement du graphique. Lorsqu'on lance le graphique l'application tente de se connecter au casque MindWave si le module Bluetooth de celui ci émet, alors la connexion se fait et l'application récupère les données de concentration et de méditation que le casque envoie, affiche ces données sur le graphique et les courbes se dessinent au fur et à mesure des données reçues. Il est ensuite possible d'appliquer différents changements grâce au menu paramètres.



*Illustration 1: Menu d'accueil de l'application*

# 1. Connexion en Bluetooth entre le casque et l'appareil

En parallèle de cette mise en place nous avons commencé l'apprentissage du Bluetooth sous Android par le développement d'une application gérant différents paramètres Bluetooth. Chafik a démarré en créant une application avec trois fonctions :

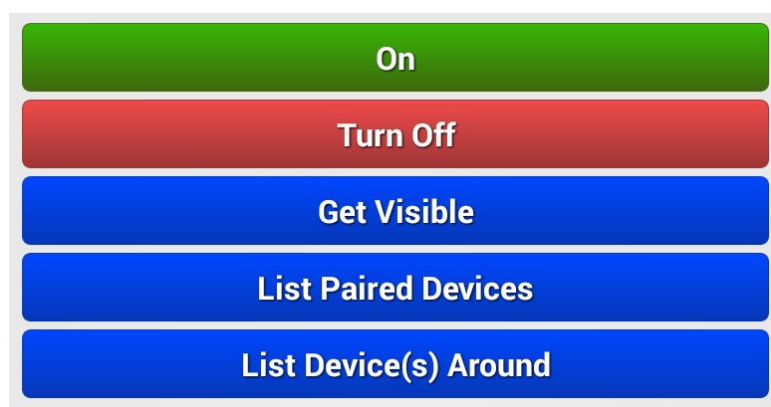
- Activer le Bluetooth
- Désactiver le Bluetooth
- Rendre notre appareil visible pour d'autres appareils Bluetooth.

À partir de cette application nous avons tous ensemble continué à développer des fonctionnalités Bluetooth, aucun d'entre nous n'étant familier avec la gestion du Bluetooth dans le développement Android. Il nous a fallu apprendre les différents éléments nécessaires au Bluetooth pour chaque nouvelle fonctionnalité. Nous nous sommes énormément aidé de la documentation Android et des différents tutoriels trouvés sur internet pour apprendre à maîtriser le BluetoothAdapter, les BroadcastReceiver, mais aussi les nouveaux éléments utiles à notre application, comme par exemple l'utilisation et le fonctionnement des listes sous Android.

Nous avons donc rajouté deux nouvelles fonctionnalités à l'application :

- le listage des appareils qui sont associés avec notre appareil
- le listage des appareils Bluetooth présents dans le rayon de notre appareil

Le Bluetooth est un moyen d'envoyer ou de recevoir des données entre deux appareils différents. En effet, Android inclut un support pour le Bluetooth qui permet à un périphérique sans fil d'échanger des données avec d'autres appareils Bluetooth. Grâce à l'API Bluetooth, nous pouvons, dans l'application BrainWave : activer et désactiver le Bluetooth, rendre notre appareil visible pour les autres, montrer la liste des appareils déjà appairés et enfin chercher les autres appareils disponibles aux alentours.



## 2. Récupération des données du casque

Le casque possède plusieurs capteurs qui, placés sur le front, reçoivent des ondes cérébrales et les transmettent à un boîtier Bluetooth qui les convertit en données brutes (rows data). Ces dernières sont traitées directement dans l'application. Les valeurs des rows data sont toujours comprises entre les valeurs numérique 0 et 100, elles sont donc affichées sur le graphique de l'application, graphique qui a une échelle de 0 à 100. Comme vous pouvez le voir dans la description suivante chaque tranche de 20 correspond à un pourcentage, un niveau faible, normal, élevé, etc.

```
/**
 * Renvoie une valeur entre 0 et 100 sur la capacité d'attention/méditation
 * 0 : incapacité à calculer une valeur d'attention/méditation
 * 1 - 20 : très faible attention/méditation
 * 20 - 40 : faible attention/méditation
 * 40 - 60 : valeur d'attention/méditation normale
 * 60 - 80 : attention/méditation relativement élevée
 * 80 - 100 : attention/méditation élevée
 */
```

## 3. Génération d'un graphique d'après les données

Grâce à la classe GraphViewData qui implémente GraphViewDataInterface, nous pouvons récupérer les coordonnées (x,y) correspondant à l'attention et à la méditation et les afficher grâce à la classe MainActivity. La courbe se crée en temps réel : dès que des données sont récupérées elles sont affichées sur le graphique. Grâce à la librairie GraphView, il est possible de parcourir le graphique à l'aide de l'écran tactile, on peut retourner ou à un moment précis de l'enregistrement.

Voici un exemple de graphique :

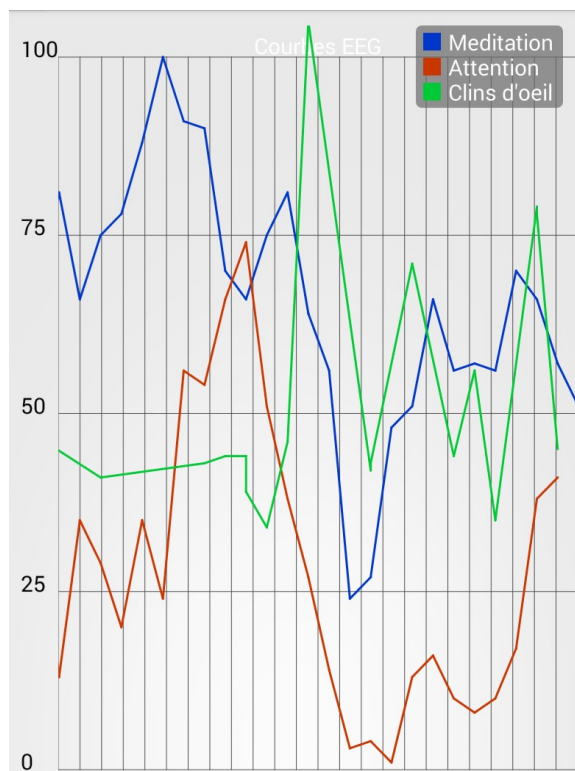


Illustration 2: Screenshot d'un graphique

## 4. Ajout d'un menu de paramètres

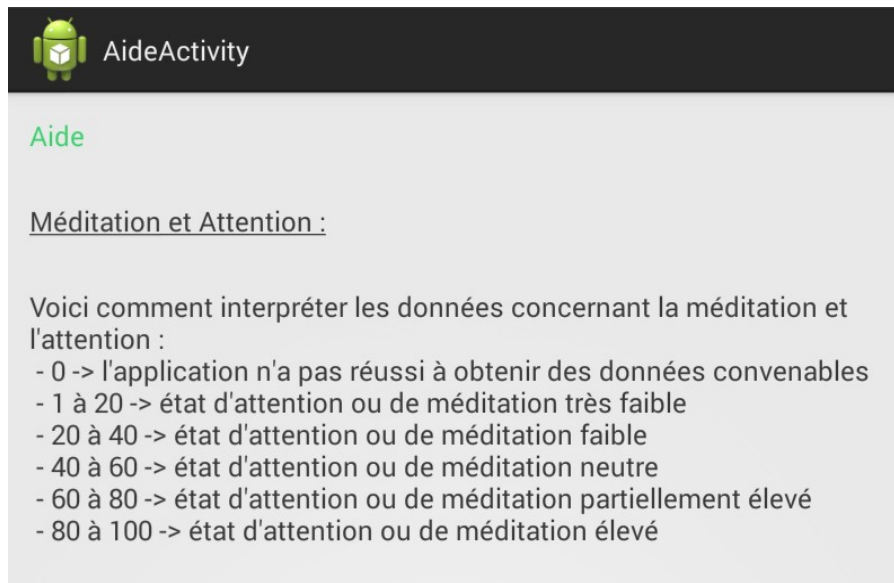
Une fois le graphique implémenté et fonctionnel, nous avons rajouter un menu Paramètres accessible un utilisant le bouton de menu disponible sur tous les appareils Android. Cette activité paramètre propose plusieurs choses : tout d'abord la possibilité, via des checkboxes, d'activer ou désactiver l'affichage des différentes courbes, ensuite on peut choisir la durée de l'enregistrement et le lancer. L'enregistrement consiste à prendre les données récupérées sur la plage de temps sélectionnée et les stocker dans un fichier csv. Ce fichier csv est enregistré dans le dossier téléchargement de l'appareil Android.



|                                   |                                     |
|-----------------------------------|-------------------------------------|
| Courbe des clins d'oeil           | <input checked="" type="checkbox"/> |
| Courbe de l'attention             | <input checked="" type="checkbox"/> |
| Courbe de la méditation           | <input checked="" type="checkbox"/> |
| Temps d'enregistrement (secondes) | 30                                  |
| Débuter l'enregistrement          | <input type="checkbox"/>            |
| Frequence Raw Data                | 50                                  |

*Illustration 3: Paramètres*

Nous avons aussi rajouté un menu "aide" : ce menu ouvre une activité qui est un texte d'explication sur ce que représentent les niveaux des courbes. Voici à quoi ce menu d'aide ressemble :



*Illustration 4: Aide*

## 5. Génération CSV d'après un enregistrement des données

Enfin de pouvoir garder une trace des données, nous avons décidé d'implémenter la possibilité de les exporter dans un fichier CSV. Pour faire cela nous avons utilisé la création de fichier Android, nous nous sommes renseigné sur la documentation Android. De plus le but n'est pas d'enregistrer toute la courbe, mais de faire un enregistrement sur une durée définie. C'est pour cela que nous avons fait en sorte que l'enregistrement se fasse sur une durée que l'utilisateur peut définir dans le menu Paramètres (par défaut la durée de l'enregistrement est de 30 secondes). Nous avons défini que le fichier csv sera enregistré dans le dossier "Téléchargement" (Download) de l'appareil car c'est un dossier qui existe sur tous les systèmes Android et qui est très simple à retrouver.



### III. Problèmes rencontrés

Nous avons rencontrés de gros problèmes de configuration et d'utilisation de EGIT (Git sur Eclipse). Il nous a d'abord fallu du temps pour bien comprendre son fonctionnement. Mais même une fois que nous avons compris l'outil, beaucoup de problèmes de conflits et de merge insolubles apparaissent, ce qui nous a souvent fait perdre du temps. Car lorsqu'un de ces problèmes arrive il nous faut passer du temps à régler le problème à la main ce qui nous a parfois pris beaucoup de temps. Sinon la solution est de supprimer notre local projet en cours et de reprendre le projet actuel sur Git, en ayant fait une sauvegarde de nos modifications à coté, on fait une sorte de réinitialisation du workspace local pour contourner les problèmes rencontrés.

Il nous avait également été demandé de pouvoir afficher un graphique à partir d'un fichier csv précédemment enregistré mais nous n'avons pas pu le faire à cause d'un manque de temps. Nous avons cependant eu le temps de commencer des recherches sur les FilePickers Android pour aller rechercher le fichier que l'on à créer en CSV.

Nous devions aussi créer un option pour afficher des points plus régulièrement afin d'obtenir de meilleures courbes. La classe Interpolation contient tous les calculs nécessaires pour cet affichage mais nous n'avons pas eu le temps de la mettre en relation avec le reste de l'application. La classe utilise pour le moment des tableaux contenant des Doubles, une fois que l'import de fichier csv sera possible il suffira de procéder à quelques petites adaptations pour l'utiliser.

## IV. Conclusion

Au final, nous avons pu réaliser une application Android fonctionnelle. Notre application BrainWave permet donc de gérer les paramètres Bluetooth de l'appareil et aussi de se connecter au casque BrainWave et de récupérer les données qu'il envoie. On peut visualiser les données en direct sur un graphique, et éventuellement les stocker dans un fichier csv. Il y a également un menu permettant de choisir les courbes que l'on veut afficher, la durée de l'enregistrement csv, mais aussi les crédits de l'application et une explication sur la valeur des points des courbes.

L'application est claire, fonctionnelle et facile d'utilisation, le design est relativement épuré. La première page Menu rassemble l'application de gestion du Bluetooth et celle de la gestion des données.