

Gestion de projet

Le code source de notre projet : <https://github.com/MLabusquiere/DrunkKeyBoard>

À la recherche d'une idée originale et désireux d'aider nos amis (et leurs amis) lors de leur soirées alcoolisées, nous avons eu l'idée de créer un clavier qui se bloquerait lorsque l'utilisateur est saoul : mots inexistants, séries de consonnes imprononçables, oublie des espaces,...

N'ayant aucune expérience dans le développement mobile nous avons commencé par lire de la documentation et trouver des exemples de code de claviers. Nous nous sommes très vite aperçu qu'il y avait un exemple de code source de clavier dans les "samples" du SDK Android.

Nos premiers tests ont été effectués avec cet exemple. Nous avons dans un premier temps parcouru le code grâce aux avantages offerts par un IDE pour comprendre le fonctionnement, la logique et l'organisation du code d'un clavier Android. Devant la réalisation que développer un clavier Android est en fait quelque chose d'assez complexe, nous avons pris la décision d'utiliser le code d'un clavier Open Source : **Hacker's Keyboard**. Nous avons donc "forké" ce projet sur GitHub, ce qui nous a permis d'avoir un clavier fonctionnel et très bien conçu comme base de nos modifications. Cela nous a permis de nous concentrer sur le code métier et les fonctionnalités de notre idée tout en n'ayant pas à nous inquiéter du fonctionnement basique du clavier.

Nous avons donc conçu quatre algorithmes permettant de détecter un utilisateur sous l'emprise de l'alcool :

- détection de frappe trop lente (l'utilisateur fait des pauses anormalement longues pendant la frappe);
- détection de chaînes de de caractères aléatoires ;
- détection d'une succession de mots n'appartenant pas au dictionnaire ;
- détection de mots trop longs (absence d'espace ou trop de fois la même lettre dans le mot).

Nous avons tous codé sur le même ordinateur soit en nous relayant soit en faisant du "pair programming" (programmation en binôme).

Après avoir compris le fonctionnement d'Android, il n'a pas été compliqué d'implémenter plusieurs algorithmes pour détecter un utilisateur ivre. Mais lorsque l'on a voulu prendre une photo nous nous sommes heurté à un problème de taille : comme le clavier est un Service il n'est pas évident d'utiliser une Activity. Cela nous a pris beaucoup de temps et de recherche pour trouver le moyen d'en lancer une. Dans un deuxième temps nous voulions que notre clavier prenne une photographie automatiquement. Malgré tous les codes que nous avons trouvés, et toutes les modifications que nous avons effectuées, il s'est révélé impossible d'utiliser la caméra en passant directement par l'API du matériel. D'après nos recherches cela pourrait être une cause matériel du au Nexus 4 que nous utilisons pour tester notre application. Nous avons donc nous rabattre sur l'application Caméra et donc c'est à l'utilisateur de décider si il veut ou non prendre une photo. Un hack est nécessaire pour forcer l'utilisation de la caméra frontale (celle en face de l'utilisateur).

Le code correspondant à cette fonctionnalité est dans une branche Git annexe **picture** car il n'est pas fonctionnel : nous n'arrivons pas à ramener l'écran sur l'application après la prise d'une photo. Ainsi notre projet avec sa dernière version fonctionnant se trouve dans la branche **master**. Et notre version beta avec la prise d'une photo se trouve dans la branche **picture**.

Nous n'avons pas eu le temps d'utiliser l'API Facebook pour partager les photographies. De plus l'objectif

de changer le niveau des messages à taper pour débloquer le clavier n'a pas été implémenté car c'est majoritairement un problème algorithmique et nous voulions nous concentrer sur les problématiques Android et "mobile".

Pour finir nous avons mis dans l'archive un fichier `git_diff.txt` qui est la différence entre le projet open source initial et notre fork.