

La publicité géolocalisée dans le cadre d'une application mobile d'avertisseur de radars

Travail de diplôme réalisé en vue de l'obtention du diplôme HES

par :

Hung Hui Henri LA

Conseillers au travail de diplôme :

Peter DAEHNE, professeur HES

Michel DERIAZ, superviseur

Genève, 18.11.2011

Haute École de Gestion de Genève (HEG-GE)

Filière Informatique de gestion

Déclaration

Ce travail de diplôme est réalisé dans le cadre de l'examen final de la Haute école de gestion de Genève, en vue de l'obtention du titre du Bachelor en informatique de gestion. L'étudiant accepte, le cas échéant, la clause de confidentialité. L'utilisation des conclusions et recommandations formulées dans le travail de diplôme, sans préjuger de leur valeur, n'engage ni la responsabilité de l'auteur, ni celle du conseiller au travail de diplôme, du juré et de la HEG.

« J'atteste avoir réalisé seul le présent travail, sans avoir utilisé des sources autres que celles citées dans la bibliographie. »

Fait à Genève, le 18.11.2011

Hung Hui Henri La

Remerciements

Tout d'abord, je tiens à remercier M. Michel DERIAZ pour m'avoir proposé ce travail et pour l'encadrement apporté tout le long de ces trois mois.

Je remercie également M. Peter Daehne, professeur HEG, pour les conseils, le suivi et les corrections prodigués tout le long de ce travail.

Mes remerciements vont également à M. Julien Pession et Mlle. Anja Bekkelien pour leur aide malgré leur charge de travail.

Enfin, je tiens également à remercier les personnes ayant pris part au questionnaire et aux personnes m'ayant soutenu jusqu'à présent.

Sommaire

Ce mémoire consiste à étudier la publicité géolocalisée dans les applications mobiles plus particulièrement les applications d'avertisseur de radars. L'objectif est d'évaluer si ce modèle économique est avantageux par rapport au traditionnel système vente-achat.

Une partie de ce mémoire expliquera la méthodologie de travail utilisée. Un tour d'horizon sur les Smartphones sera présenté afin de sélectionner un système d'exploitation mobile adéquat pour le projet. Cette sélection se fera à l'aide d'une analyse multicritère. Dans notre cas, pourquoi avoir choisi de développer sous Android ?

Suite à cela, la notion de géolocalisation sera introduite suivi des outils proposés sous Android pour la localisation. Par la suite, une brève explication sera donnée concernant les outils développées et utilisés, dont « FoxyTag » pour le bon fonctionnement du travail.

Ensuite, nous enchaînerons sur la publicité géolocalisée par sa définition et la sélection d'un service dont celui d'AdMob. Toutes ces notions permettront le développement d'une API et d'applications d'avertisseurs de radars jusqu'à la publication sur l'Android Market.

Grâce aux applications, une étude sera menée à travers un questionnaire auprès de personnes de tous horizons. Ce résultat permettra de savoir si une application mobile incorporant la géolocalisation serait rentable avec la publicité géolocalisée comme source de revenu ? C'est-à-dire, une application gratuite ou semi payante serait-elle plus profitable qu'une application payante ? Si oui, par quels moyens ? Quel est le meilleur emplacement sur un écran pour rentabiliser l'application ?

Table des matières

Déclaration	i
Remerciements	ii
Sommaire	iii
Table des matières	iv
Liste des Tableaux	vi
Liste des Figures	vi
Introduction.....	1
1. La méthodologie SCRUM	2
1.1 Les acteurs	2
1.2 Déroulement	2
2. Les Smartphones	5
2.1 Situation des OS sur le marché des Smartphones	6
2.2 iOS	7
2.3 Android	8
2.4 Le choix.....	11
3. La géolocalisation sur Android	14
3.1 Type de localisation sur Android	15
3.2 Outils sur Android.....	15
4. FoxyTag	18
4.1 Pourquoi FoxyTag.....	18
4.2 Réalisation	18
5. La publicité géolocalisée.....	20
5.1 Définition.....	20
5.2 AdMob	23
6. API AndroTag	27
6.1 Fonctionnalités.....	27
6.2 Tests	28
6.3 Première application : AndroEye.....	31
7. Multi-interfaces (multi-applications)	34
7.1 Contraintes	34
7.2 Analyse et explications	34
8. Sondage	37
8.1 Résultat du sondage.....	38
8.2 Synthèse du sondage	50
9. Conclusion.....	53

Webographie.....	55
Bibliographie.....	56
Annexe 1 Interface des applications avec publicité.....	57
Annexe 2 Optimizing connections (Extract from API FoxyTag)	58
Annexe 3 Fonctionnalités l'API + Application	60
Annexe 4 Questionnaire	63
Annexe 5 Planning SCRUM	70

Liste des Tableaux

Tableau 1 Vente de téléphones par fabricant en 2011 en milliers d'unités	5
Tableau 2 Vente de Smartphones par système d'exploitation en 2011 en milliers d'unités	6
Tableau 3 Actuelle distribution des versions d'Android	10
Tableau 4 Résumé iOS - Android.....	11
Tableau 5 Pondération des critères.....	12
Tableau 7 Différences entre BuzzCity, AdFonic et AdMob	22
Tableau 8 Tailles de publicité	23

Liste des Figures

Figure 1 Déroulement de la méthodologie SCRUM	4
Figure 2 Vente de Smartphones par système d'exploitation en 2011 en %.....	7
Figure 3 Actuelle distribution des versions d'Android.....	9
Figure 4 Matrice de préférence	12
Figure 5 Exemple de géolocalisation GPS	14
Figure 6 Illustration classes	16
Figure 7 Redirection sur AdMob depuis AdSense.....	24
Figure 8 Catégorie de publicité.....	25
Figure 9 AdSense dans AdMob.....	25
Figure 10 Etat des revenus sur AdMob	26
Figure 11 Ecrans AndroTag	28
Figure 12 Exemple de pose de radars via Google™ Map.....	29
Figure 13 Exemple de pose de radars via FoxyKahna.....	30

Figure 14 Ecrans AndroEye	31
Figure 15 Aperçu d'applications publiées via la vue développeur.....	31
Figure 16 Statistiques 1 : AndroEye	32
Figure 17 Statistiques 1 : AndroEye	33
Figure 18 Publicité : Bannière	34
Figure 19 Publicité : Rectangle.....	35
Figure 20 Publicité : Leaderboard	35
Figure 21 Sexe	38
Figure 22 Tranche d'âge	38
Figure 23 Pourcentage de personnes conduisant une voiture	39
Figure 24 Pourcentage de personnes ayant utilisé une application avec publicité	39
Figure 25 Nombre de personnes ayant utilisé une application incluant de la publicité	40
Figure 26 Genre de publicité dérangeante	40
Figure 27 Pourcentage de personnes intéressé par de la publicité géolocalisée dans les applications mobiles.....	41
Figure 28 Nombre de personnes selon le sexe ayant utilisé un avertisseur de radars	42
Figure 29 Nombre de personnes selon le sexe ayant installé un avertisseur de radars	42
Figure 30 Nombre de personnes selon sexe à propos de la dangerosité d'un avertisseur de radars	43
Figure 31 Pourcentage de vote concernant le classement de l'application 1....	43
Figure 32 Pourcentage de vote concernant le classement de l'application 2....	44
Figure 33 Pourcentage de vote concernant le classement de l'application 3....	44
Figure 34 Pourcentage de vote concernant le classement de l'application 4....	45
Figure 35 Pourcentage d'avis concernant la mise à jour de l'application 1	46
Figure 36 Pourcentage d'avis concernant la mise à jour de l'application 2	46
Figure 37 Pourcentage d'avis concernant la mise à jour de l'application 3	47

Figure 38 Pourcentage d'avis concernant la mise à jour de l'application 4	47
Figure 39 Résumé des pourcentages d'avis concernant la mise à jour des applications.....	48
Figure 40 Nombre de personnes selon le sexe prêt à cliquer au moins 1 fois sur la publicité.....	48
Figure 41 Préférences concernant les genres de publicité dans le cadre d'un avertisseur de radar.....	49
Figure 42 Pourcentage de personnes selon le sexe tenté de télécharger une des quatre applications.....	50

Introduction

Aujourd'hui, la publicité est omniprésente dans notre quotidien. Elle revêt différentes formes et touche un certain public (sexe, âge et intérêt). Cette forme de publicité a su évoluer à travers le temps et avec notre mode de vie. Elle est présente à la télévision, dans nos journaux, dans la rue, sur internet et maintenant dans nos téléphones mobiles. De nos jours, la majorité des gens possède en permanence un téléphone mobile plus précisément un Smartphone. Ces Smartphones possèdent diverses fonctionnalités, dont celle de la téléphonie de base, du lecteur mp3, des caméras intégrées, des applications et d'une puce GPS.

Depuis l'apparition des Smartphones, des plateformes de téléchargement se sont mises en place pour permettre aux consommateurs de télécharger des applications gratuites ou payantes. Ces applications utilisent diverses techniques de financement. L'une d'elle est la publicité géolocalisée.

Cette forme de publicité permet d'avertir le consommateur quels sont les produits ou services disponibles à proximité de sa position géographique. En utilisant cette méthode, nous pouvons nous interroger si une application mobile incorporant la géolocalisation serait rentable avec la publicité géolocalisée comme source de revenu ? C'est-à-dire, une application gratuite ou semi payante serait-elle plus profitable qu'une application payante ? Si oui, par quels moyens ? Quel est le meilleur emplacement sur un écran pour rentabiliser l'application ?

Pour répondre à cette problématique, nous allons au préalable devoir développer des applications mobiles utilisant pleinement le système de géolocalisation afin d'y intégrer facilement le concept de publicité géolocalisée. Grâce à cela, nous pourrions étudier le comportement des utilisateurs face à ces applications.

Pour ce faire, ce mémoire va s'articuler sur 3 principaux points :

1. Une présentation des notions de base et de la méthodologie
2. La réalisation de l'expérience
3. L'analyse des résultats

1. La méthodologie SCRUM¹

Pour mener à bien ce travail, j'ai été immergé dans une équipe composée de Julien Pession, développeur informatique, Anja Bekkelien, informaticienne, et de Michel Deriaz, superviseur de projets dont celui-ci à l'université de Genève. Dans cette équipe, la méthodologie de gestion de projets employée est le SCRUM.

SCRUM est une méthode agile de gestion de projets informatiques.

1.1 Les acteurs

Dans un SCRUM, il y a quatre principaux acteurs :

- **Le SCRUM Master**, celui qui définit la priorité des fonctionnalités à développer
- **L'équipe**, composée de développeurs qui autogèrent le planning de leurs tâches
- **L'animateur**, celui qui s'assure que la méthodologie est respectée
- **Les intervenants**, ceux qui ont une vue externe du projet

Dans ma situation, le SCRUM Master et l'animateur sont joués par Michel Deriaz. L'équipe est composée de trois développeurs dont Julien Pession, Anya Bekkelien et moi. Dans le cas du travail de Bachelor, l'intervenant est Peter Daehne. Chaque développeur était occupé au développement de son propre projet.

1.2 Déroulement

Au début du projet, le SCRUM Master établit avec le développeur la date de fin de release. Une release est la période pendant laquelle le projet dure avant la livraison finale du produit. Dans notre cas, la date de réédition du travail de Bachelor est synonyme de la fin de la release.

Ensuite, le SCRUM Master établit la liste des stories à développer. Ces stories sont des objectifs à remplir afin de mener à bien le travail. Ces objectifs peuvent être classés en trois catégories : l'étude, le développement et finalement la réalisation. Cette liste est placée dans le Sand Box² en attente de validation.

1 Le terme SCRUM est emprunté au rugby et signifie mêlée.

2 C'est un bac à sable contenant les idées de tâches proposées par le développeur au SCRUM Master ou par le SCRUM Master au développeur concerné.

Après validation par le développeur concerné, cette liste est placée dans le « Product Back log » en attente d'évaluation de la charge de travail. Dans le cas contraire, le développeur discute avec le SCRUM Master pour réadapter certaines stories (fusion ou division).

Ensuite, le groupe se réunit pour évaluer la liste des stories de chaque développeur selon la charge du travail établie en nombre de jours. Le SCRUM Master détaille la description de chaque story et chacun donne son avis concernant le temps nécessaire à l'achèvement de la tâche. L'avis est donné à l'aide de cartes (planning poker) permettant à chacun de montrer son estimation sans être influencé par l'avis des autres.

Lorsque tout le monde est d'accord sur le temps nécessaire à l'accomplissement de la tâche, la story est validée et placée dans les tâches du développeur. Dans le cas contraire, chacun donne son avis afin de se mettre tous d'accord.

Après avoir évalué toutes les stories jusqu'à une charge de 9 jours/10 jours, le développeur pourra commencer son sprint le lundi qui vient. Le 10ème jour est consacré à la livraison du produit de fin de sprint au SCRUM Master. Un sprint est une période de 10 jours de travail ouvrables (2 semaines). Une release est composé de plusieurs sprints.

Chaque matinée, un daily SCRUM a lieu à 9 heures réunissant toute l'équipe. Ce daily SCRUM dure 15 minutes.

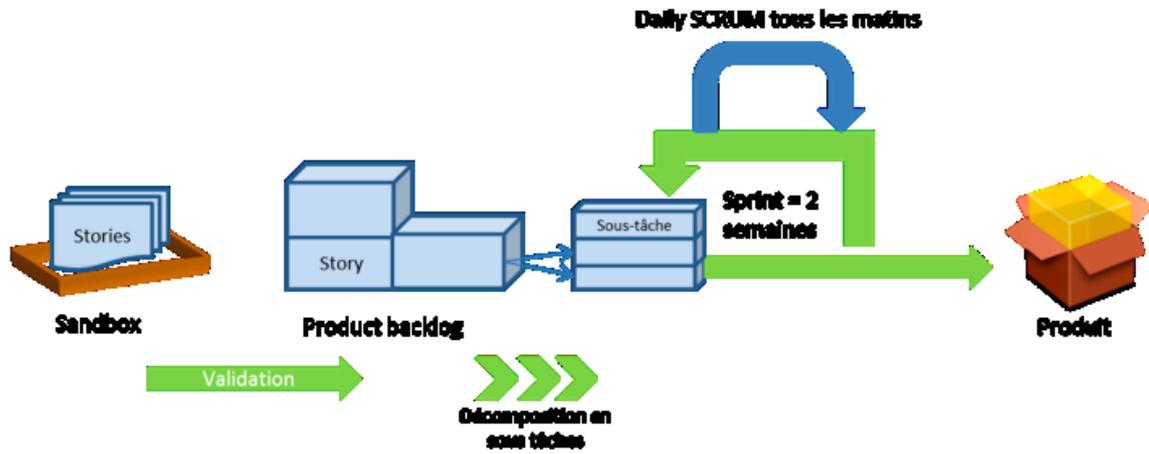
Devant tout le monde, chaque membre répond à trois questions :

1. Qu'est-ce que j'ai fait hier ?
2. Qu'est-ce que je compte faire aujourd'hui ?
3. Quelles sont les difficultés que je rencontre ?

A la fin d'un sprint, une réunion a lieu faisant état de la situation du projet pour finalement valider ou pas la réalisation effectuée.

Voici un schéma résumant la méthodologie utilisée :

Figure 1
Déroulement de la méthodologie SCRUM



2. Les Smartphones

En 2011, les Smartphones sont omniprésents dans notre quotidien. Ils permettent de communiquer textuellement ou oralement, de jouer, de prendre des photos, de filmer... En bref, ces téléphones ont pris une place significative dans notre vie. De nos jours, en observant notre vie quotidienne, nous nous apercevons que tout le monde possède sur soi un Smartphone, les jeunes, les jeunes adultes et les adultes.

Actuellement sur notre marché, les Smartphones sont tous différents : modèles, tailles, systèmes d'exploitation, fonctionnalités, puissance matérielle... La source de ces différenciations provient des fabricants de téléphones essayant de se démarquer de leurs concurrents.

Voici un tableau recensant les ventes de téléphones par fabricant en 2011 en milliers d'unités :

Tableau 1
Vente de téléphones par fabricant en 2011 en milliers d'unités

Fabricant	Unités (M)	Unités %
Nokia	97'869	22.8
Samsung	69'827	16.3
LG	24'420	5.7
Apple®	19'628	4.6
ZTE	13'070	3.0
Research in Motion	12'652	3.0
HTC	11'016	2.6
Motorola	10'221	2.4
Huawei Device	9'026	2.1
Sony Ericsson	7'266	1.7
Autres	153'662	35.8
Total	428'657	100.0

Source : Gartner (2011)

En étudiant ce tableau, nous observons que les cinq premiers fabricants de téléphones occupent plus de cinquante pourcents du marché de la téléphonie. Quelles sont les raisons qui attirent les consommateurs, l'appareil en lui-même ou le système d'exploitation utilisé ?

2.1 Situation des OS sur le marché des Smartphones

De nos jours, il y a beaucoup de fabricants de Smartphones sur le marché. Certains fabricants sont ancrés sur le marché depuis l'apparition des téléphones portables tel que Nokia par exemple alors que d'autres ont suivi la tendance comme par exemple Apple®.

Pour se différencier, certains fabricants proposent sur leurs mobiles un système d'exploitation propriétaire alors que d'autres non.

Tableau 2

Vente de Smartphones par système d'exploitation en 2011 en milliers d'unités

Système d'exploitation	Unités (M)	Unités %
Android	46'775	43.4
Symbian	23'853	22.1
iOS	19'628	18.2
Research in Motion	12'652	11.7
Bada	2'055	1.9
Microsoft	1'723	1.6
Autres	1'050	1.0
Total	107'736	100.0

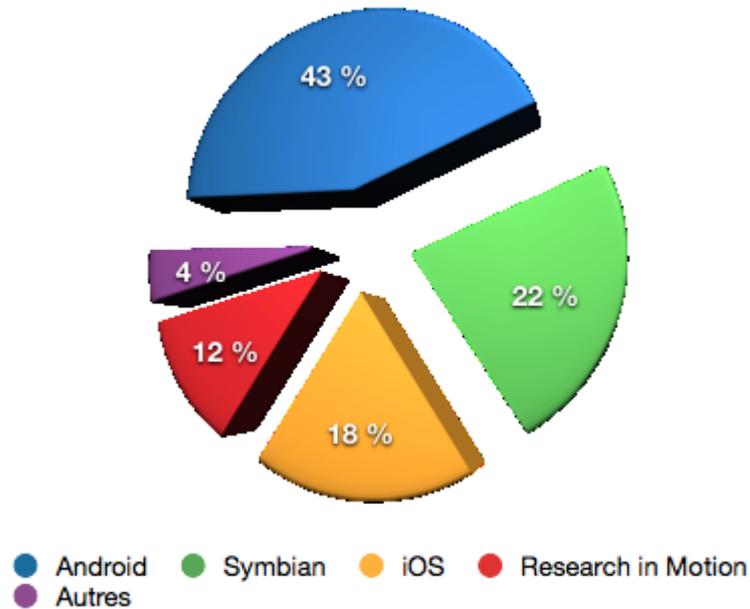
Source : Gartner (2011)

Comme montré sur le tableau ci-dessus, trois systèmes d'exploitation sont principalement utilisés sur le marché :

- Android, développé par Google™
- Symbian, développé par Nokia
- iOS, développé par Apple®

Voici un graphique représentant les données du tableau précédent :

Figure 2
Vente de Smartphones par système d'exploitation en 2011 en %



Source : Gartner (2011)

Au moment de l'écriture de ce travail, Symbian OS a été abandonné au profit du système d'exploitation Windows Phone suite à une annonce communiquée par Nokia en février 2011. Par conséquent, nous ne nous étalerons pas sur Symbian OS et Windows Phone OS. Dans le cas du premier, il est inutile de développer une future application destinée à personne. Dans le cas du second, il faudrait quelques années pour voir si cet OS séduit les consommateurs.

A propos de notre travail, nous allons principalement nous pencher sur les deux systèmes d'exploitation restant qui sont l'iOS et Android. Nous essayerons de déterminer quelle est la meilleure plateforme pour développer une application de géolocalisation utilisant la publicité géolocalisée.

2.2 iOS

L'iOS est le nom du système d'exploitation mobile produit et développé par la célèbre compagnie Apple®. Apple® est une multinationale occupant sur le marché des ordinateurs (iMac, MacBook Pro, MacBook Air), de la téléphonie (iPhones) et du divertissement (iPod, Apple® TV, iPad).

Ce système d'exploitation est uniquement utilisé sur les Smartphones d'Apple® appelés iPhone®. Depuis, cet OS s'est exporté sur les produits iPad et Apple® TV. La dernière version à ce jour est l'iOS 5.

Concernant les caractéristiques de ce système d'exploitation mobile, le SDK iOS est disponible gratuitement en téléchargement sur internet, par contre, il ne fonctionne qu'avec les systèmes d'exploitation Macintosh. Ce SDK nécessite l'IDE XCode utilisable uniquement sur le système d'exploitation d'Apple®. Par conséquent, le développement nécessite une machine spécifique. De plus, il est nécessaire de connaître le langage Objective-C pour développer.

A propos du développement d'application iOS, il faut savoir que seuls les produits Apple® sont compatibles. Par conséquent, une application développée sur cette plateforme ciblera seulement les utilisateurs de ces produits.

De plus, la majorité des applications développées sur l'AppStore³ sont payantes. Dans la mentalité des consommateurs des produits Apple®, tout ce qui se trouve sur l'AppStore est payant et les produits gratuits sont des versions incomplètes. Ces versions incomplètes sont illustrées par des versions « Lite ». Ce sentiment est aussi partagé à propos des accessoires proposés par Apple®.

Enfin, l'AppStore effectue des contrôles drastiques sur les applications publiées. L'AppStore doit approuver à 100%⁴ l'application avant que celle-ci ne soit disponible sur la plateforme de téléchargement d'Apple®. Par conséquent, il se passe un certain temps entre la fin du développement d'une application et le moment où les utilisateurs peuvent la télécharger. Le temps est non mesurable et donc totalement aléatoire. Par contre, ce contrôle assure à ses consommateurs que les applications sont de qualité et respectent leurs droits. De plus, l'accès à l'AppStore pour le développeur nécessite le paiement de 100 \$ par année.

2.3 Android

Android (ou « Android ») est le nom d'une entreprise et aussi le nom d'un système d'exploitation mobile développé par l'Open Handset Alliance. L'Open Handset Alliance

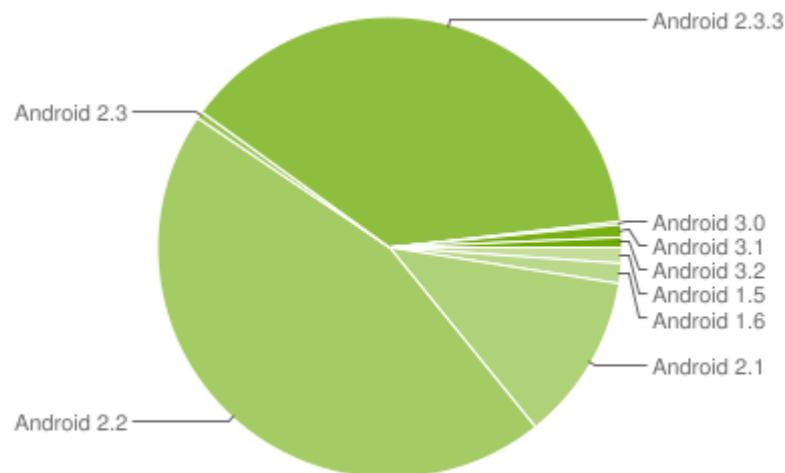
³ L'AppStore est la plateforme utilisée par les appareils Apple® (iPad, iPhone®, iPod) pour télécharger des applications, de la musique ou pour y déposer des applications.

⁴ Si une application est mal codée, elle peut être refusée par l'AppStore. De même, si l'application envoie des données importantes à l'insu de l'utilisateur.

est un consortium regroupant 84⁵ entreprises ayant pour but de promouvoir un système complet, ouvert et gratuit. En 2007, Google™ annonce avoir acheté la compagnie Android. C'est à partir de l'année 2008 qu'Android se fait connaître avec la version 1.0 de son SDK.

Ce système d'exploitation est utilisé sur les appareils mobiles tels que les tablettes, les ordinateurs portables et les Smartphones. Apparue en 2008, il a su se développer pour actuellement atteindre une version 4.0⁶ et essayer de rivaliser avec son principal concurrent iOS 5.

Figure 3
Actuelle distribution des versions d'Android



Source : Android developpers (octobre 2011)

Actuellement, un grand nombre de Smartphones utilise les versions Android 2.2 à 2.3.7. Ces Smartphones ont un format ressemblant principalement à l'iPhone® d'Apple®. La version 3.0 est exclusivement destinée aux tablettes et, à partir de la version 4, les deux types de plateformes (tablette, Smartphone) sont supportés.

⁵ Composés de : Google™, HTC, Samsung, Motorola, Dell, T-Mobile,...

⁶ « Ice Cream Sandwich », présenté le 20 octobre 2011. Les versions majeures d'Android portent des noms de desserts

Tableau 3
Actuelle distribution des versions d'Android

Plateforme	Nom de code	Distribution en %
Android 1.5	Cupcake	1.1
Android 1.6	Donut	1.4
Android 2.1	Eclair	11.7
Android 2.2	Froyo	45.3
Android 2.3 – 2.3.7	Gingerbread	38.7
Android 3.0 – 3.2	Honeycomb	1.8
Android 4.0	Ice cream sandwich	0.0

Source : Android developpers (octobre 2011)

Le SDK Android est disponible gratuitement au téléchargement sur internet. Il ne nécessite aucun matériel précis, ce qui n'est pas le cas d'iOS. Par conséquent, ce SDK est homogène sur la majorité des systèmes d'exploitation. Ce SDK est principalement couplé avec l'IDE Eclipse pour le développement, mais il peut aussi être utilisé avec l'IDE Netbeans. Concernant le développement, il suffit d'avoir la maîtrise du langage JAVA ⁷ et de prendre en main le framework d'Android.

Actuellement, ce système d'exploitation est utilisé par de nombreux fabricants de Smartphones dont : HTC ; Samsung, Google™, Motorola, LG, Sony Ericsson...

Concernant sa plateforme de téléchargement nommée Android Market, elle propose tout comme son concurrent⁸ des applications payantes et gratuites pour ses utilisateurs. Cependant, une partie des applications payantes proposée sur l'AppStore sont disponibles gratuitement sur Android Market comme par exemple le célèbre jeu « Angry Birds ». Cet aspect est particulièrement apprécié par la communauté des utilisateurs

Cette ouverture peut par contre apparaître dangereuse aux yeux des utilisateurs. Concernant le dépôt d'applications Android, la plateforme n'est pas aussi restrictive que celle d'Apple®. Il peut être plus facile de trouver des applications gratuites,

⁷ JAVA est un langage de programmation orienté objet, le plus connu et le plus accessible auprès des informaticiens

⁸ L'AppStore, son antagoniste principal

inutilisables ou dangereuses. Au début, certaines applications avaient la faculté de rendre inutilisable un téléphone.

Concernant l'accès à l'Android Market, le développeur doit payer 25 \$ par année pour soumettre ses applications.

2.4 Le choix

Il peut paraître difficile de choisir sur quel système d'exploitation nous souhaitons développer pour mener à bien notre étude.

Afin d'avoir une vue d'ensemble, voici un tableau qui résume ce qui a été présenté précédemment :

Tableau 4
Résumé iOS - Android

Caractéristiques	iOS	Android
Entreprise	Apple®	Google™
Version	5.0	4.0
Programmation	<i>Objective-C</i>	Java
Supports	Produits Apple®	Samsung, Motorola, HTC, Google™...
Parts de marché	18%	43%
Politique	Close	Open
Plateforme SDK	Mac OS X	Linux, Mac OS X et Windows
Plateforme app.	AppStore / iTunes	Android Market
Coût développeur	100 \$	25 \$

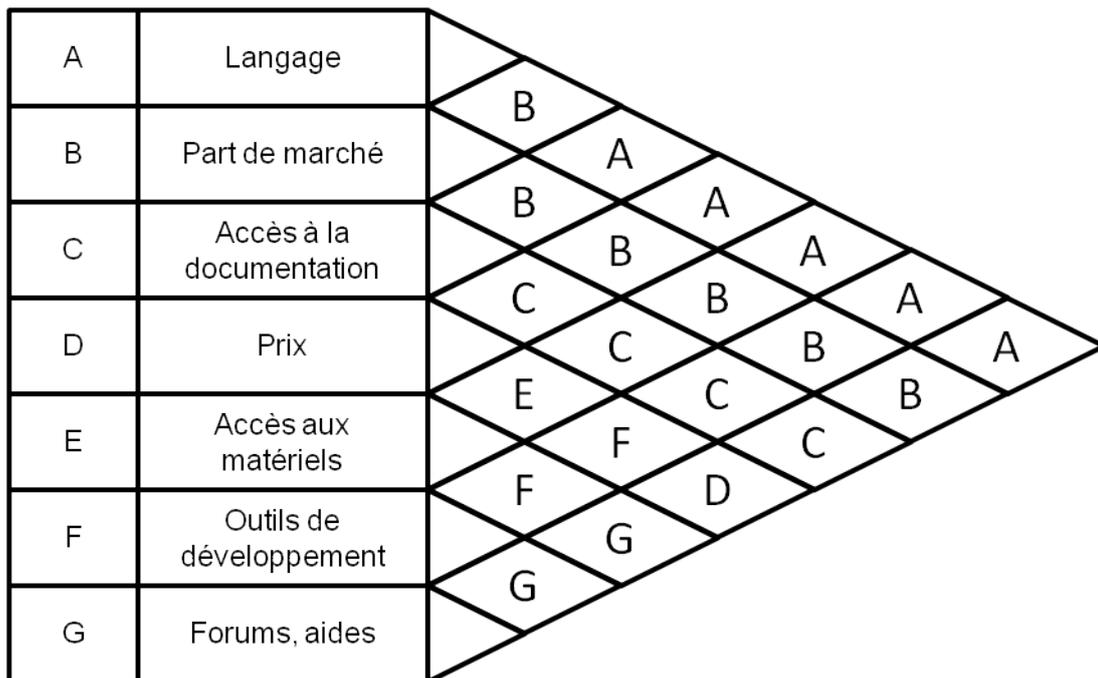
Dans le but de choisir la plateforme adéquate, plusieurs critères ont été retenus :

- Le langage : Un langage de programmation non maîtrisé risque d'apporter du retard dans la maîtrise du framework.
- L'accès à la documentation : Est-ce que la documentation est claire et mise à jour périodiquement ?
- La part de marché : Plus la part de marché est grande, plus nous pourrions récupérer de données et toucher un grand nombre d'utilisateurs.
- Le prix : Développer une application impose-t-il l'achat d'une licence de développeur ?
- L'accès aux matériels : Le développement peut imposer un matériel spécifique. Il est possible que nous n'ayons pas le matériel à disposition.

- Les outils de développement : Est-ce qu'il existe un grand nombre d'outils de développement tel que des librairies ou des IDE ? Y a-t-il des émulateurs ?
- Les forums, l'aide : Est-ce que la communauté de développeurs est assez grande pour pouvoir apporter de l'aide ? Est-elle assez développée ?

Suite à ces critères, nous avons établi des pondérations pour chacun. Pour ce faire, une matrice de préférences est utilisée :

Figure 4
Matrice de préférence



La pondération est le nombre d'apparition de chaque critère par rapport à la somme des occurrences de chacun. Ce qui nous donne :

La somme des occurrences = 21

Tableau 5
Pondération des critères

	A	B	C	D	E	F	G
Occurrence	5	6	4	1	1	2	2
Pondération	5/21	6/21	4/21	1/21	1/21	2/21	2/21
Pondération (%)	23.8	28.6	19.0	4.8	4.8	9.5	9.5

Les notes pour chaque critère sont évaluées sur trois points. Le maximum est égal à trois.

Tableau 6
Tableau multicritères iOS - Android

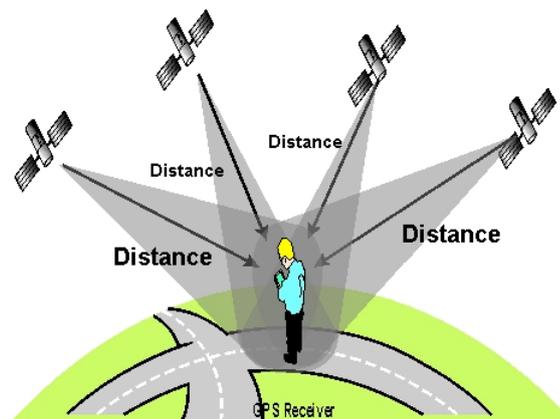
Critère	iOS		Android	
	Nombre de points	Total (Nb pts * pondération)	Nombre de points	Total (Nb pts/3 * pondération)
Langage <i>Pondération 23.8%</i>	2	0.476	3	0.714
Part de marché <i>Pondération 28.6%</i>	1	0.286	3	0.858
Accès à la documentation <i>Pondération 19.0%</i>	3	0.570	3	0.570
Prix <i>Pondération 4.8%</i>	1	0.048	3	0.144
Accès aux matériels <i>Pondération 4.8%</i>	3	0.144	3	0.144
Outils de développement <i>Pondération 9.5%</i>	3	0.285	1	0.095
Forums, aides <i>Pondération 9.5%</i>	2	0.190	3	0.285
Total (100%)		1.999 / 3		2.81 / 3

Suite aux résultats du tableau ci-dessus, il est clair que notre choix se tourne vers la plateforme de Google™.

3. La géolocalisation sur Android

La géolocalisation est un système permettant de déterminer la position d'un objet ou d'une personne sur une carte. La position est exprimée en latitude et en longitude. Il est possible de récupérer d'autres informations telles que la vitesse, la précision, l'altitude, la distance et la direction.

Figure 5
Exemple de géolocalisation GPS



Source : http://www.wirelessdictionary.com/wireless_dictionary_GPS_definition.html (octobre 2011)

La détermination de la position peut se faire selon plusieurs techniques :

- par wifi ; cette technique utilise la triangulation des signaux émis par les bornes wifi avoisinant l'appareil. Pour fonctionner, l'appareil doit être connecté à un réseau wifi. L'inconvénient est lorsque l'appareil se trouve dans une zone non couverte par un réseau wifi.
- par géocodage ; cette technique permet de déterminer les coordonnées à partir d'un texte décrivant l'endroit en interrogeant une base de données. Par exemple, Google™ Map utilise cette technique car il suffit de lui fournir une adresse pour qu'il nous retourne une liste de coordonnées potentielles.
- par GSM ; Cette technique utilise les ondes émises par les antennes de téléphone cellulaires afin de déterminer la position à l'aide d'une base de données.
- par GPS ; cette technique utilise les satellites orbitant autour de notre planète pour indiquer la position de l'appareil. Pour fonctionner, l'appareil nécessite une puce GPS et que au moins trois satellites soient visibles (quatre pour l'altitude).

- par marqueur, ou par puces RFID ; il s'agit d'une technique permettant de lire et de récupérer des données à l'aide d'un marqueur. Ces marqueurs peuvent être des QR Code contenant des coordonnées par exemple.

La géolocalisation est actuellement très utilisée dans le monde professionnel, notamment pour le suivi d'objets par la poste ou des sociétés de transport tel que FedEx. Cet outil est aussi notamment utilisé dans le milieu carcéral par exemple au Canada. (Bracelet électronique)

Concernant notre travail, nous allons seulement utiliser la fonction du GPS. La raison est que l'application nécessite de pouvoir récupérer à n'importe quel moment la position de l'appareil pour ainsi utiliser les services de publicité. De plus, de nos jours, tous les Smartphones ont tous des puces GPS intégrées.

3.1 Type de localisation sur Android

Sur Android, il est possible d'utiliser trois techniques de géolocalisation qui sont le GPS, GSM et Wifi.

Concernant le GPS, c'est le plus précis des trois, mais c'est le plus gourmand en batterie⁹. Il nécessite par contre une puce GPS. La majorité des téléphones Android possèdent une puce GPS utilisant le système « A-GPS¹⁰ ». Cette puce est aussi présente dans les iPhone®s.

3.2 Outils sur Android

Pour utiliser la géolocalisation GPS, il est nécessaire d'utiliser plusieurs classes fournies par l'API de Google™. Ces classes sont :

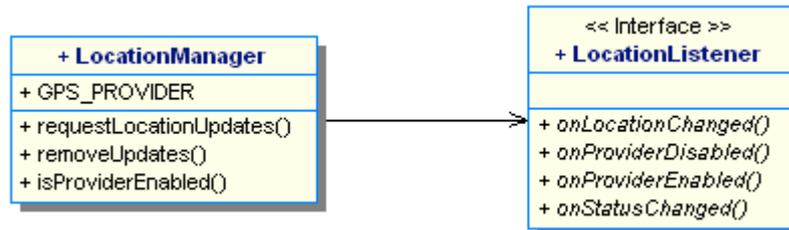
- Location, La classe contenant les informations nécessaires à la localisation. Ces informations sont la latitude, la longitude, la vitesse...
- LocationManager, Cette classe permet d'avoir accès au service de localisation du système. Elle permet aux applications d'obtenir des données périodiquement ou après avoir parcouru une certaine distance.

⁹ Ce n'est pas un problème car l'application est principalement utilisée en voiture. Qui dit voiture, dit prise allume-cigare.

¹⁰ Une technique de positionnement par satellites.

- LocationListener, Cette interface est utilisée pour recevoir les coordonnées de la classe LocationManager lorsque la position a changé.

Figure 6
Illustration classes



Plusieurs méthodes et variables de la classe LocationManager sont utilisées :

- LocationManager.GPS_PROVIDER
- requestLocationUpdates(String provider, long minTime, float minDistance, LocationListener listener)
- requestLocationUpdates(String provider, long minTime, float minDistance, LocationListener listener)
- removeUpdates(LocationListener listener)
- isProviderEnabled(String provider)

Il y a d'autres méthodes et variables disponibles dans la classe. Dans notre cas, ce sont ces méthodes et variables qui nous intéressent. Pour plus d'explications concernant ces méthodes, la documentation d'Android est disponible sur internet. (<http://developer.android.com/reference/android/location/LocationManager.html>)

L'obtention de la classe LocationManager ne nécessite pas l'instanciation de cette classe. Elle est disponible dans une classe « Activity » à l'aide de la méthode getSystemService(String name).

Pour qu'une application puisse utiliser ces fonctions, il est nécessaire d'ajouter dans le Manifest d'Android la permission d'utiliser le GPS du téléphone. Sans cette permission, des messages d'erreurs risquent de subvenir pendant l'exécution.

Avec ceci, il est possible de pouvoir localiser l'appareil en question.

Concernant le test, il y a deux possibilités : soit tester en cas réel avec un appareil soit en simulant.

La simulation s'effectue à l'aide d'un émulateur. L'émulateur permet de simuler un appareil Android en fonction de la version du SDK et il est possible de customiser le simulateur en lui fournissant quelques options. Pour fournir des coordonnées à l'émulateur, il est nécessaire d'utiliser soit le terminal ou l'outil « Emulator Control » inclus dans l'IDE d'Eclipse.

Dans le cas d'un appareil, il suffit d'exécuter une application implémentant le package « Location » et d'attendre la réception d'un signal. La réception du signal peut prendre du temps en fonction du lieu d'exécution de l'application. Sous un ciel ouvert, cela est très rapide tandis que dans un bâtiment, cela risque de prendre un certain temps dû aux obstacles rencontrés par le signal (murs, toits...)

4. FoxyTag

FoxyTag est un système collaboratif, existant depuis 2006, qui permet de signaler des radars à l'aide d'un système de tags. Ce système est utilisé sur les supports mobiles et est également disponible dans le monde entier. Cette méthode est développée par la FoxyTeam, dirigée par Michel Deriaz.

FoxyTag est une application développée en JAVA répertorient les radars et leurs positions GPS. A l'aide d'une position géolocalisée, l'utilisateur de FoxyTag pourra récupérer une liste de radars à proximité. Cette liste de radars sera mise à jour selon certains critères définis par le système. L'utilisateur pourra collaborer avec le système en taguant virtuellement les radars.

Les applications consultent les données fournies par FoxyTag à partir de la position GPS du mobile.

La majorité des applications disponibles sont payantes et consultables sur le site www.foxytag.com.

4.1 Pourquoi FoxyTag

Comme cité plus haut, le but de ce travail est de pouvoir développer une application utilisant un système de géolocalisation afin d'intégrer de la publicité géolocalisée.

FoxyTag est le système parfait pour mener cette étude car d'une part, il est gratuit pour le développement et nécessite l'utilisation de la fonction de géolocalisation du téléphone de l'utilisateur, d'une autre part, il n'est pas nécessaire que nous nous occupions de la gestion de ce système sous-entendant la base de données de radars (intégrité des données, injection SQL, serveur, etc.). Grâce à la position du mobile, il nous sera possible d'insérer de la publicité géolocalisée sans perturber l'utilisateur.

Il serait d'une part bizarre de développer une simple application et d'incorporer de la publicité géolocalisée sans lien direct avec le but principal de l'application. Par exemple, développer une application de carnet de notes qui demande la fonction de GPS à l'utilisateur serait complètement absurde pour l'utilisateur qui refuserait d'utiliser, voire d'installer l'application.

4.2 Réalisation

Pour mener à bien ce travail, il sera nécessaire de développer une API utilisant le système de FoxyTag. Grâce à cet API, il sera facile de développer rapidement divers

applications/interfaces afin d'insérer de manières différentes la publicité en utilisant une politique de prix différente de l'une à l'autre.

Cette API sera développée en JAVA afin d'être incorporée dans le SDK Android. Les fonctionnalités sont présentées plus loin dans ce document.

5. La publicité géolocalisée

La publicité géolocalisée utilise la fonction de géolocalisation d'un appareil pour fournir une publicité dynamique aux consommateurs, principalement les internautes. De nos jours, cette publicité a envahi nos Smartphones, tout particulièrement les sites web mobiles et les applications mobiles.

5.1 Définition

Dans le domaine de la publicité, il y a deux acteurs principaux: le publieur et l'annonceur.

- L'annonceur : Son objectif est de promouvoir sa marque ou ses produits à l'aide de support textuel, visuel et/ou sonore.
- Le publieur : Son intérêt est de publier de la publicité proposée par l'annonceur afin de percevoir un revenu.

Sur le marché de la publicité mobile et web, il existe des plateformes permettant à ces deux acteurs de satisfaire leurs intérêts : ce sont des agences de publicité. Ces plateformes donnent la possibilité aux annonceurs de se promouvoir à l'aide de bannières interactives. Ces bannières respectent toutes un standard de tailles (320x50 pixels par exemple). Elles permettent de pouvoir interagir directement avec le service concerné tel que la téléphonie, le sms, la vidéo ou internet. Par exemple, en cliquant sur la bannière, le client téléphone à la société concernée ou visionne la publicité de cette dernière.

Grâce à cette plateforme, les publieurs constitués d'entreprises ou de particuliers peuvent publier des annonces sur leurs sites web ou leurs applications en proposant une certaine catégorie de publicité à l'aide d'outils (API). Cet acteur reçoit en contrepartie une participation financière selon un critère de revenu. Il existe divers critères de revenu qui sont :

- Le coût par clic (Cost per click) : A chaque clic sur la publicité, le publieur perçoit un montant de l'annonceur.
- Le coût par acquisition/action : Semblable au coût par clic, sauf que le publieur perçoit un revenu lorsque le client est allé au bout de la publicité. C'est-à-dire que le client a acheté le produit, s'est inscrit sur le service...

- Le coût par milliers d'impression : Le publieur perçoit un montant en fonction du nombre de fois que la publicité soit affichée. Les annonceurs paient toutes les mille impressions pour que leur message soit exposé auprès d'un public spécifique.

A propos de ces plateformes, il existe trois types de services liés à ce genre de publicité sur le marché. Ces trois types sont :

- Blind networks : Un service facilitant l'accès à la publicité pour les éditeurs (développeurs) et ne restreignant pas les types d'annonceurs. N'importe quelle société ou personne ne peut avoir accès à ce service. Ce type de service favorise le coût par click et le coût par milliers d'impression.
- Premium blind networks : Ce service favorise les publieurs connus. Le coût par action est principalement proposé dans ce type de service.
- Premium networks : Ce service est seulement destiné aux grandes entreprises nationales (publieurs).

Chacun de ces services propose des outils d'analyse permettant par exemple de connaître :

- la démographie des clients,
- la démographie des mobiles,
- les revenus générés,
- etc.

Concernant notre choix, nous nous sommes penchés sur les "Blind networks". La raison est que notre application nécessite d'avoir une vaste base de publicités sans être focalisée sur des marques spécifiques.

A propos des « Blind networks », il existe une importante liste de services (source : <http://www.gomonews.com/mads/>). Parmi ces services, il est difficile de pouvoir établir le meilleur service au monde car il n'y en a pas. Trois services ont retenu notre attention ; ce sont les suivants :

Tableau 7

Différences entre BuzzCity, AdFonic et AdMob

		BuzzCity	AdFonic	AdMob/AdSense
Established		2006	2009	2006 (Google™ acquired in 2010)
Publisher on network		3000	+ 3000	+3000
Advertisers on network		+ 300	+ 3000	Unknown
Mobile ads served or page impression		5 billion ads per month	2 billion ad per month	10 billions per month, 32 different ads per month and 9400 per day
Geographical coverage	Countries	200	190	???
	Asia	48 %	21 %	10 %
	America	6 %	21 %	51 %
	Africa	3 %	???	???
	Europe	???	17 %	4 %
	Unknown	43 %	41 %	35 %
CPC vs CPM		95:5	95:5	Unknown
Specialism by publisher or demographic		All, Web Mobile + Applications mobile	iOS and Android	All except against morals
Option for targeting adverts	Channel	YES	Unknown	YES
	Countries	YES	YES	YES
	Devices	YES	YES	YES
	Location	Unknown	YES	YES
	Other	YES	YES, gender, age	YES
Tools		YES	YES	YES
Remuneration for publishers		65 % placement. Top publishers earns ~ 7000\$/day with 60 million ad impression/day	60 %, Unknown	Vary to size of the publisher. Example : "Backflip Studios earn 100'000 \$ per month => http://www.Google.com/ads/mobile/pdfs/casestudies/backflip.pdf
SDK + Compatible with android		YES	YES	YES

Source : <http://www.mobithinking.com/mobile-ad-network-guide/blind> (octobre 2011)

Notre choix est celui d'AdMob / Google™ Mobile Adsense.

5.2 AdMob

AdMob est une société fondée en 2006 par Omar Hamoui axé dans le domaine de la publicité mobile. Cette société fut rachetée par Google™ en 2011. Actuellement, cette plateforme est en cours de restructuration avec le service de Google™ (AdSense).

La géolocalisation est intégrée dans l'API d'AdMob¹¹ :

Comme toute autre plateforme de publicité mobile, AdMob propose divers types de bannières qui sont :

Tableau 8
Tailles de publicité

Taille	Description
320x50	Standard banner
300x250	Medium Rectangle
468x60	Full-Size Banner
728x90	Leaderboard

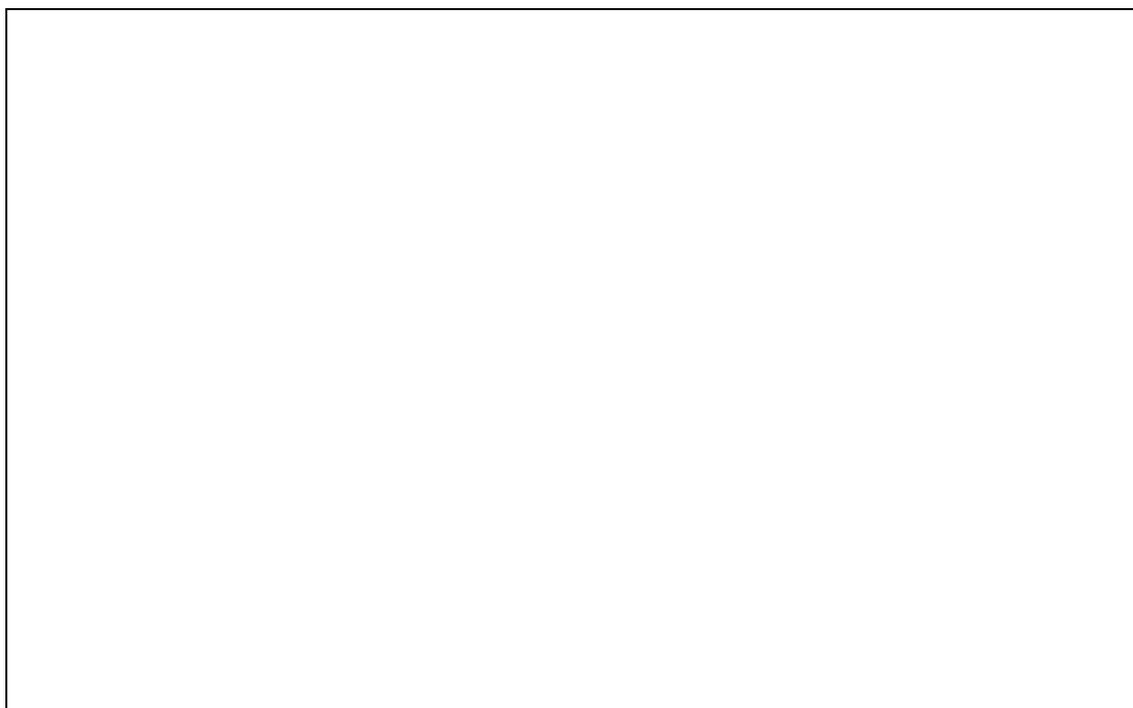
Source : <http://code.google.com/intl/fr/mobile/ads/docs/android/intermediate.html>

Etant en cours de restructuration, les offres proposées sur AdSense ne sont plus disponibles. Le site d'AdSense conseille d'aller sur AdMob.

¹¹ <http://code.Google™.com/intl/fr/mobile/ads/docs/android/intermediate.html#bannersizes>

Figure 7

Redirection sur AdMob depuis AdSense



L'utilisation d'AdMob nécessite la création d'un compte qui est gratuit. Il est possible de créer un compte AdMob ou de relier un compte Google™ déjà existant.

Après l'inscription, nous pouvons directement créer une application utilisant ces services. Il suffit d'indiquer le nom de l'application, le type d'application et de remplir quelques champs¹². De plus, il est nécessaire d'implémenter le code fourni par AdMob. Ce code est fourni dans le SDK d'AdMob. Sa version actuelle est la 4.3.1 et est téléchargeable gratuitement.

Dans l'interface d'AdMob du compte web, il est possible de filtrer les genres de publicités par catégorie.

¹² (Pour plus de détails : <http://helpcenter.admob.com/content/pskinintroduction>)

Figure 8
Catégorie de publicité

Catégories	Paramètre	État
Abonnements	<input checked="" type="checkbox"/>	Publicités en cours
Actualités, Sport et Météo	<input checked="" type="checkbox"/>	Publicités en cours
Communication	<input checked="" type="checkbox"/>	Publicités en cours
Concours	<input checked="" type="checkbox"/>	Publicités en cours
E-commerce	<input checked="" type="checkbox"/>	Publicités en cours
Films, Télévision et Divertissement	<input checked="" type="checkbox"/>	Publicités en cours
Jeux	<input checked="" type="checkbox"/>	Publicités en cours
Jeux de hasard	<input type="checkbox"/>	Publicités filtrées
Offres d'affiliation	<input checked="" type="checkbox"/>	Publicités en cours
Outils	<input checked="" type="checkbox"/>	Publicités en cours
Photos et Vidéos	<input type="checkbox"/>	Publicités filtrées
Politique	<input type="checkbox"/>	Publicités filtrées
Portails et Références	<input type="checkbox"/>	Publicités filtrées
Religion	<input type="checkbox"/>	Publicités filtrées
Réseau d'amis	<input type="checkbox"/>	Publicités filtrées
Santé et Bien-être	<input checked="" type="checkbox"/>	Publicités en cours
Sites personnels	<input type="checkbox"/>	Publicités filtrées
Sonneries et Musique	<input checked="" type="checkbox"/>	Publicités en cours
Images publicitaires	<input checked="" type="checkbox"/>	Publicités en cours
Publicités réservées aux adultes	<input type="checkbox"/>	Publicités filtrées

Source : Mon compte AdMob

De plus, il est possible d'utiliser les services d'AdSense comme montré sur cette figure.

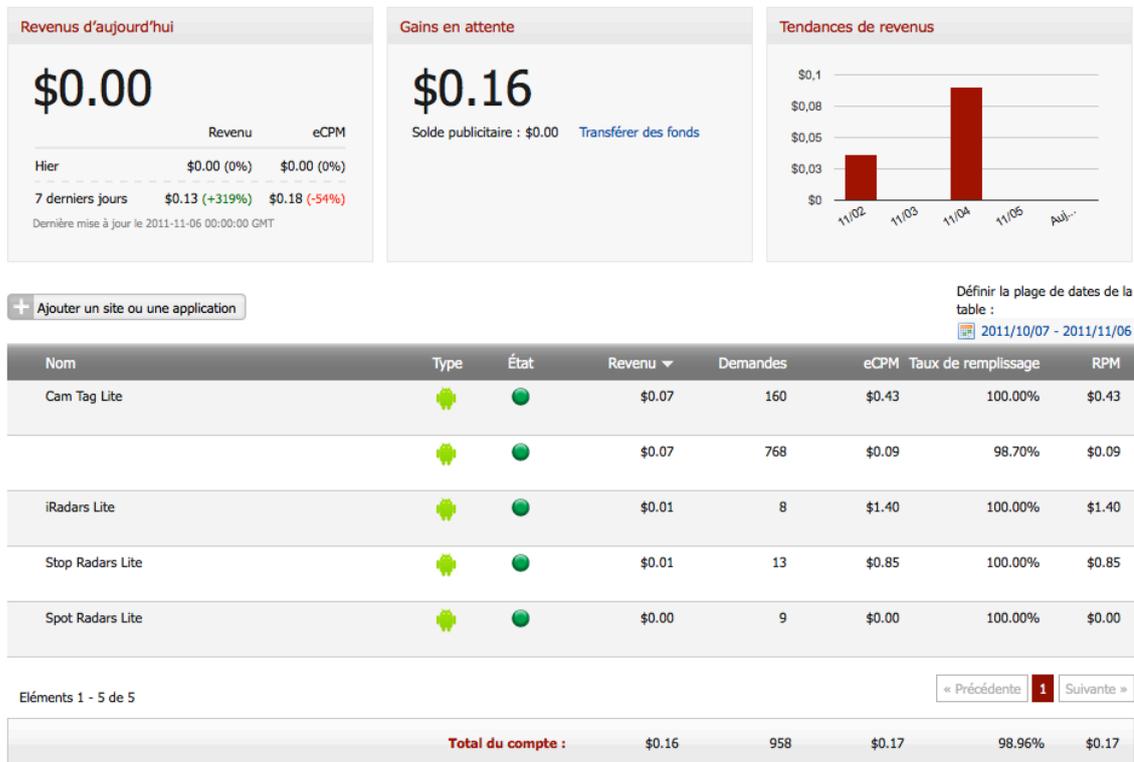
Figure 9
AdSense dans AdMob

Google AdSense : Utilisez Google AdSense pour améliorer le taux de remplissage. J'ai lu et j'accepte les [conditions d'utilisation](#) de Google AdSense.
 Ne pas utiliser Google AdSense.

Source : Mon compte AdMob

Concernant l'activité publicitaire, le propriétaire du compte peut avoir une vue d'ensemble sur le revenu perçu par la publicité pour chaque application.

Figure 10
Etat des revenus sur AdMob



Source : Mon compte AdMob

A propos du SDK, il est facile d'implémenter le code dans un projet Android. Dans le cas contraire, la documentation est disponible sur internet et une communauté d'utilisateurs est active sur les forums. (Documentation : <http://code.google.com/intl/fr/mobile/ads/docs/>)

En utilisant ce genre de service, il est difficile de trouver des données concernant le revenu potentiellement perçu¹³. Il existe des témoignages de compagnies ayant utilisé ce genre de service :

- BackFlip Studio :
<http://www.google.com/ads/mobile/pdfs/casestudies/backflip.pdf>
- Best Cool Fun : <http://www.google.com/ads/mobile/pdfs/casestudies/best-cool-fun.pdf>

¹³ A ce propos, en cliquant une fois sur une publicité, 0.03\$ ont été crédités sur mon compte AdMob.

6. API AndroTag

Cet API a pour but de fournir des outils complexes à des développeurs afin de pouvoir développer rapidement des détecteurs de radars en s'occupant simplement de l'interface utilisateur.

Avec cet API, une première version d'application a été créée.

6.1 Fonctionnalités

L'API permet de mettre à jour la base de données interne concernant les radars à proximité du véhicule en respectant un algorithme précis et l'API FoxyTag. L'algorithme consiste à optimiser le téléchargement des radars selon notre position et la direction dans laquelle nous nous dirigeons. (Voir annexe 2) De plus, il doit pouvoir communiquer (connexion, envoi et réception) les radars au service de FoxyTag.

Concernant l'affichage, l'API doit fournir un outil permettant la modélisation d'un scanner permettant d'afficher les radars dangereux. (Voir annexe 3 pour la liste des fonctionnalités complètes)

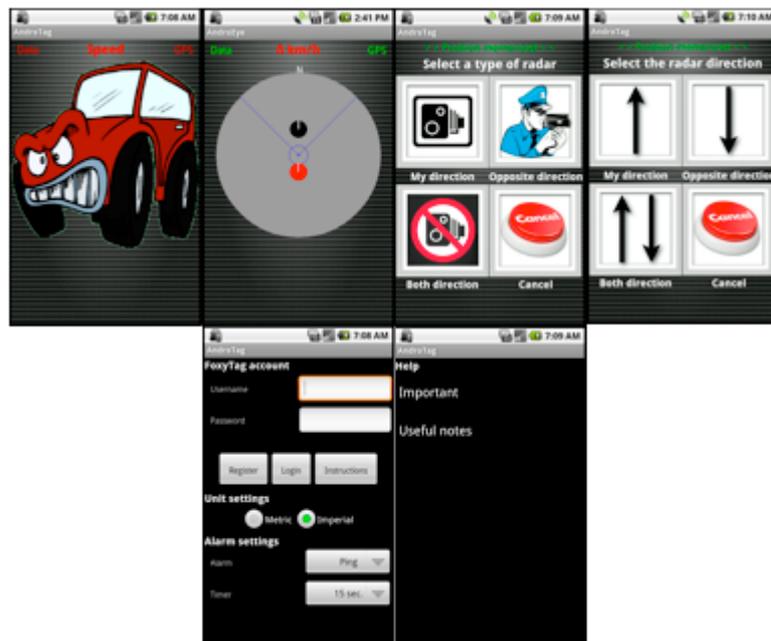
Cet API a pour but de pouvoir créer plusieurs applications possédant les mêmes fonctionnalités mais ayant des interfaces légèrement différentes.

A l'aide de cet API, une première version d'application a été créée du nom d'AndroTag.

Ces principales fonctionnalités sont :

- Afficher une page d'accueil
- Afficher les radars
- Afficher un écran d'information et d'aide
- Afficher un écran de paramétrage « développeur »
- Implémenter l'API développé

Figure 11
Ecrans AndroTag



6.2 Tests

Il est nécessaire d'effectuer le test d'une telle application au moyen d'un véritable téléphone mobile. Il est impossible d'utiliser un émulateur pour simuler un réel déplacement comprenant la vitesse et la perte de précision du GPS. Certaines fonctionnalités de base peuvent être testées avant d'aller sur le terrain tel que la connexion réseau, l'enchaînement des fenêtres, l'affichage des radars, etc.

Afin de mener à bien le test, des téléphones nous ont été prêtés. Cependant, il est assez difficile de fournir des radars de test au serveur de FoxyTag. La seule interface utilisable est le navigateur web via l'URL¹⁴ et l'utilisation de Google™ Earth¹⁵ afin de récupérer des coordonnées. Lorsqu'il fallait créer un radar, la latitude et la longitude étaient récupérées de Google™ Earth pour pouvoir construire une URL afin d'y être envoyé au serveur de FoxyTag. De plus, il fallait créer des tags dans Google™ Earth pour garder une trace des radars créés. Malheureusement, il n'est pas possible d'indiquer visuellement dans quelle direction étaient orientés les radars.

¹⁴ L'adresse web

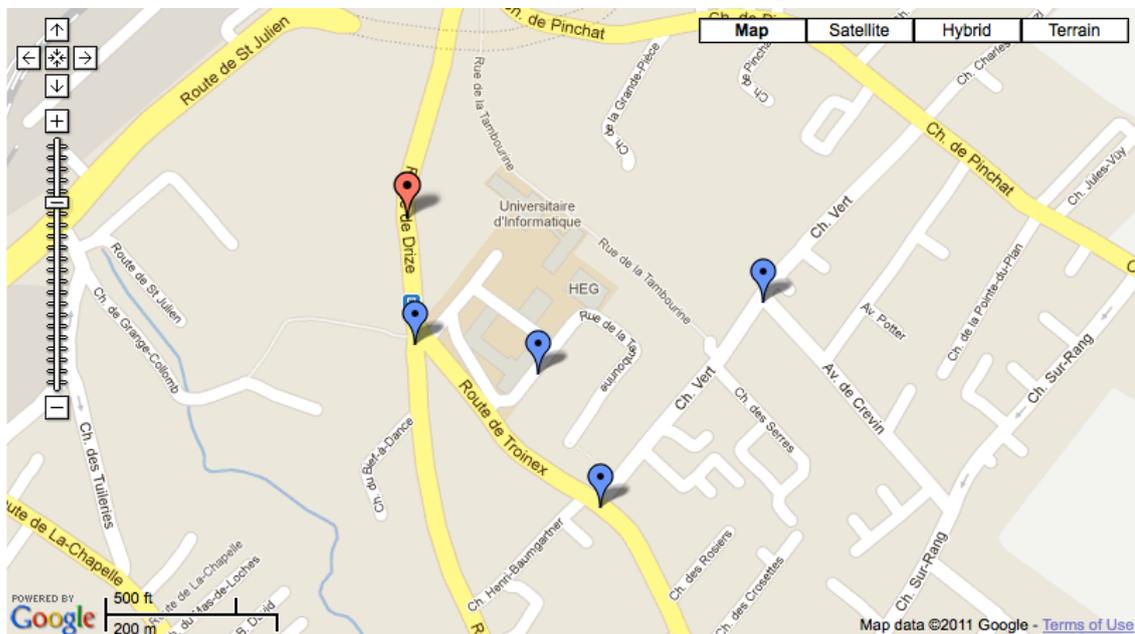
¹⁵ C'est logiciel permettant de visualiser la Terre.

Etapes :

- Récupérer la latitude et la longitude dans Google™ Earth et créer un tag
- Construire une URL :
<http://www.foxytag.com/php/dogetlab.php?cmd=tagpost&username=stag&password=&client=stag&lat=46.18000&lon=6.10000&kind=F&heading=1180&speed=80>
- L'exécuter dans le navigateur

Figure 12

Exemple de pose de radars via Google™ Map

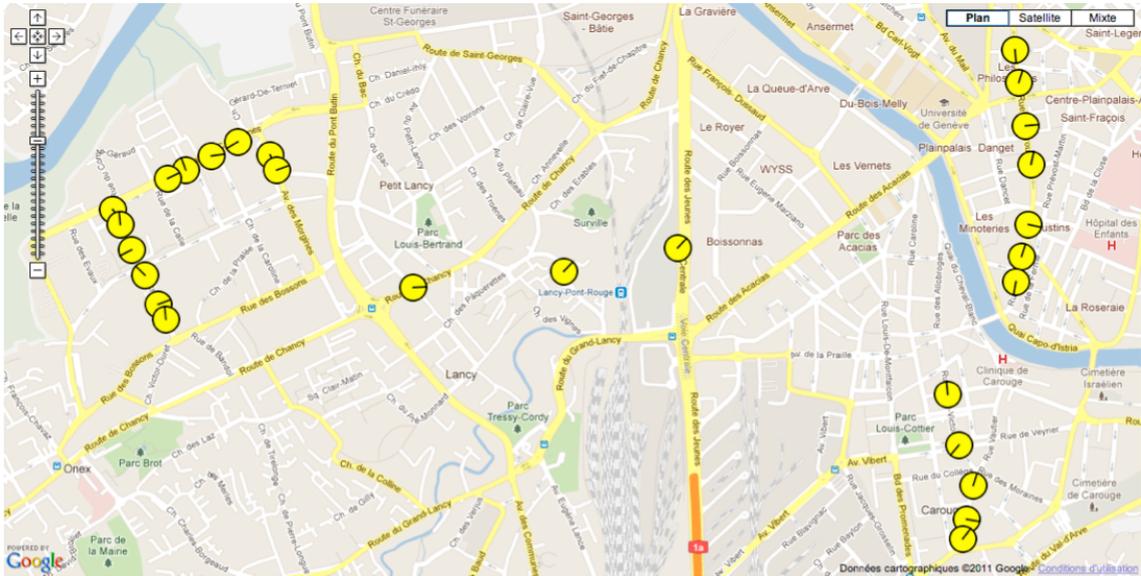


L'utilisation de ces deux interfaces est fastidieuse pour seulement poser des tags.

Pour remédier à ce problème, j'ai pu avoir accès à un système du nom de FoxyKahna me permettant ainsi de facilement poser des tags. Ce système, prêté par M. Deriaz, me permet de récupérer les coordonnées, de construire et d'envoyer des URL au système de FoxyTag à travers d'une seule interface.

Figure 13

Exemple de pose de radars via FoxyKahna



Avec ceci, il suffisait d'imprimer cette carte et d'aller tester. Cette carte m'indique dans quelle direction sont les radars.

Cependant même en ayant une base de radars, il est possible de rencontrer des problèmes pendant le test. Tel que le problème de tagging liés à de mauvaises prises de coordonnées, des problèmes de calculs de coordonnées ou bien un crash de l'application. Des imprévus peuvent survenir et il est impossible de déboguer sur le moment dans ces cas là.

Pour optimiser les tests, il a fallu en plus créer une mini base de données sur l'application afin d'écrire les logs des actions faites par l'utilisateur sur l'application. Le « LogCat », système de log pour le débogage, n'est pas convenable. Les messages écrits dans le LogCat ne sont pas stockés éternellement.

En plus, une librairie a été utilisée pour le report de crash. Cette librairie s'appelle ACRA¹⁶. Elle permet d'envoyer un report sur un compte Google™ lorsque l'application crash pour diverses raisons. Le report contient des informations sur l'appareil et sur la raison du crash. Elle donne la possibilité de faire un suivi de l'application pendant et après le développement. Ainsi, il est possible de savoir sur quel type de téléphone, l'application risque de ne pas fonctionner.

16 Application Crash Report for Android

6.3 Première application : AndroEye

Après avoir développé une version démo développeur, une première application fut créée pour la publication. Cette application du nom d'AndroEye est gratuite permettant d'avoir un feedback des utilisateurs. Tout comme AndroTag, elle possède la même structure sauf que les images sont plus soignées et la partie paramétrage est simplifiée.

Figure 14
Ecrans AndroEye



Cette application gratuite, validée par M. Deriaz, a été publiée le 2 novembre 2011 sur l'Android Market sur le compte de M. Deriaz. Disponible actuellement sur : https://market.android.com/details?id=app1.ch.unige.activities&feature=search_result#?t=W251bGwsMSwxLDEsImFwcDEuY2gudW5pZ2UuYWN0aXZpdGllcyJd

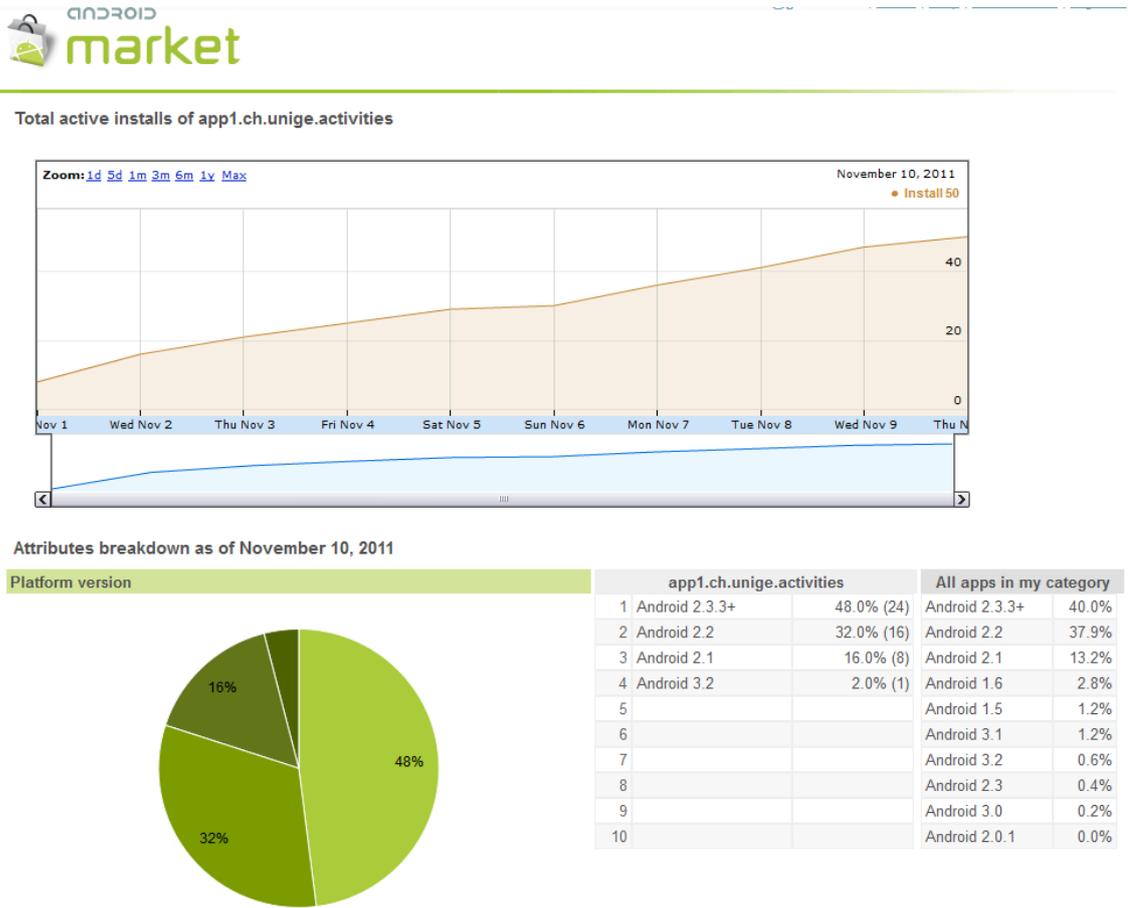
Depuis le compte développeur d'Android Market, il est possible de visualiser l'activité de l'application sur le marché à l'aide de statistiques.

Figure 15
Aperçu d'applications publiées via la vue développeur



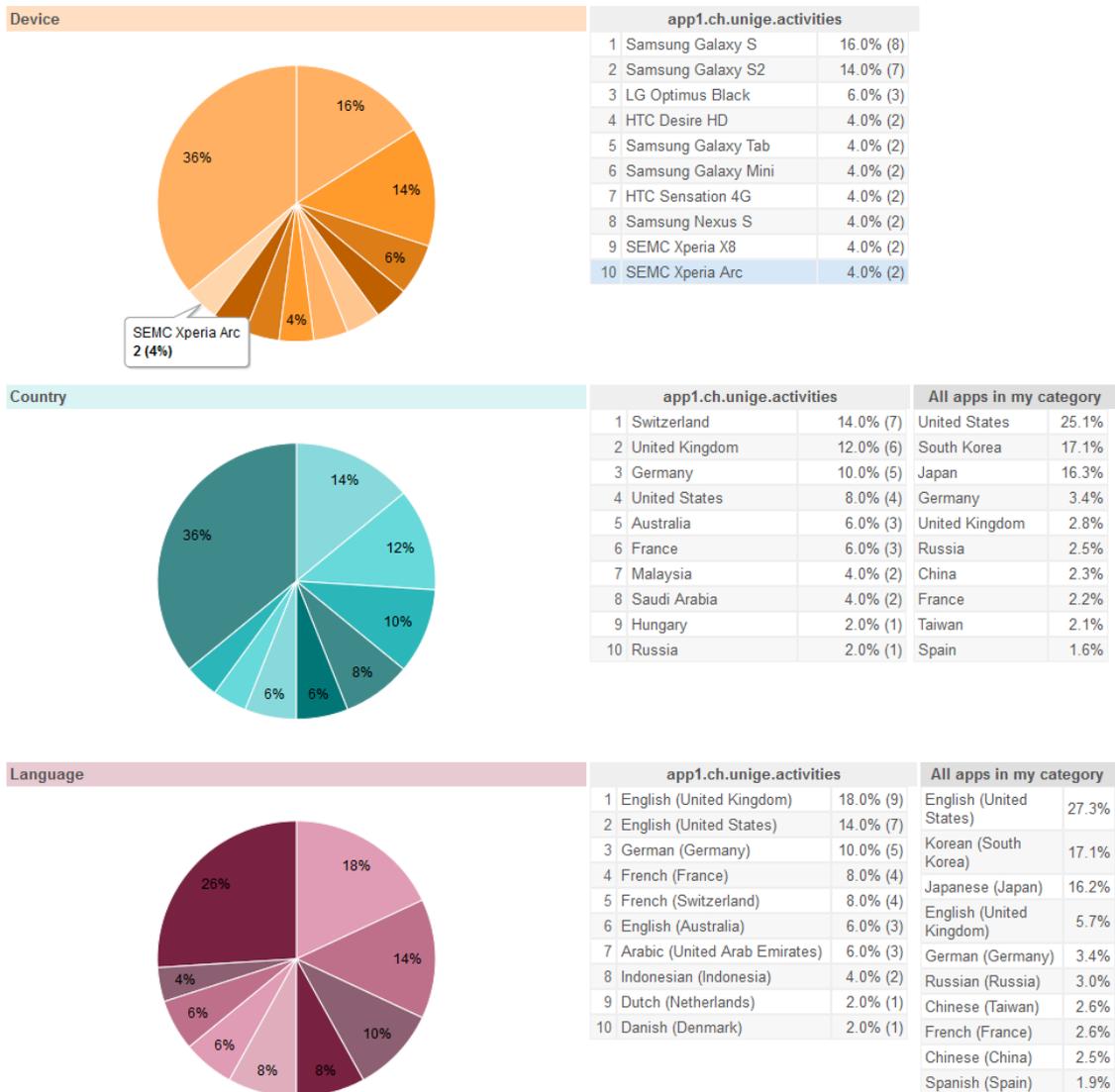
Il est possible d'observer le nombre de fois que l'application a été installée et actives.

Figure 16
Statistiques 1 : AndroEye



Ceci permet de voir quelles versions d'Android utilisent l'application.

Figure 17
Statistiques 1 : AndroEye



Comme montrée dans les figures X et Y, ces statistiques permettent de savoir quels sont :

- les types de téléphones ayant téléchargé l'application,
- la démographie des utilisateurs (langue, pays),
- le nombre de téléchargement par jour
- et les plateformes d'Android utilisant l'application.

7. Multi-interfaces (multi-applications)

L'objectif de ce chapitre est de créer plusieurs interfaces (applications) implémentant de manière différente de la publicité géolocalisée en réutilisant l'API développé.

De cette façon, il permettra de savoir quelle est la meilleure interface permettant d'obtenir un meilleur revenu et une meilleure appréciation des utilisateurs en utilisant de la publicité géolocalisée comme source de revenu.

7.1 Contraintes

L'utilisation de la publicité géolocalisée d'AdMob nécessite d'avoir :

- Un compte AdMob
- Ajouter une application Android dans le compte AdMob
- Récupérer l'identifiant de la publicité
- Implémenter le code nécessaire accompagné de l'identifiant

Pour ce projet, la version 4.3.1 du SDK AdMob est utilisée.

7.2 Analyse et explications

Après avoir publié une première version de l'application, il est temps de créer des interfaces incluant de la publicité géolocalisée. Concernant les formats publicitaires choisis, la bannière et le rectangle seront utilisés.

Figure 18

Publicité : Bannière



Le format bannière est le plus utilisé dans les applications mobiles, d'une part il est discret par sa taille et de l'autre s'adapte facilement sur les écrans mobiles.

Figure 19

Publicité : Rectangle



Le format rectangle est principalement adapté pour les tablettes électroniques. Cependant, il serait intéressant de le voir sur un environnement mobile.

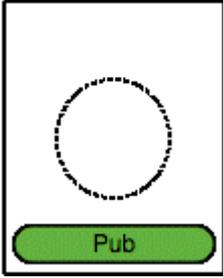
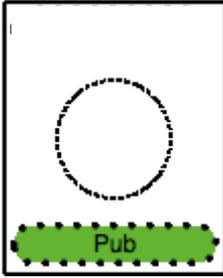
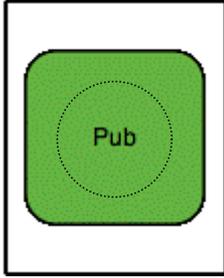
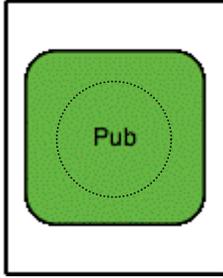
Figure 20

Publicité : Leaderboard



Le format Leaderboard est difficilement implémentable par sa taille dans ce genre d'application risquant de déranger l'utilisateur.

Voici quatre modèles d'interface que nous avons retenue :

	Application 1 : Cam Tag Lite	Application 2 : SpotRadars Lite	Application 3 : iRadars Lite	Application 4 : StopRadars Lite
Modèles				
Lieu	Page d'accueil	Page d'accueil	Page d'accueil	Page d'accueil
Apparition	Permanent	A l'arrêt du véhicule < 5 km/h	Lorsqu'il n'y a pas de radars dans les 600 m	A l'arrêt du véhicule < 5 km/h
Opinion personnel	Ne gêne pas la visibilité	Ne gêne pas la visibilité	Gêne la visibilité	Gêne la visibilité

Les applications 1 et 2 utiliseront le format des bannières tandis que les deux autres, celui des rectangles.

Toutes ces applications seront gratuites et pourront toutes être mises à jour. La mise à jour signifie le téléchargement d'une version sans publicité à condition de payer le prix. Le prix serait de 3€ par exemple.

A l'aide de ces modèles, nous pourrions les comparer selon :

- Les différents/mêmes formats de la publicité
- Le moment d'apparition de la publicité
- Le moment d'apparition selon le type de publicité
- Lequel des applications est susceptible d'être acheté

En clair, il est prévu de publier les huit applications. (Interface des applications, Annexe 1)

8. Sondage

Afin de savoir lequel de ces modèles est susceptible d'être apprécié ou non par les utilisateurs, un sondage a été effectué. Il aurait été plus appréciable de le faire dans un environnement réel via l'Android Market¹⁷.

Ce sondage a été principalement effectué dans les bâtiments de « Battelle » du 9 novembre au 11 novembre 2011.

Ce questionnaire consistait à recenser des avis à propos de la publicité géolocalisée dans le cadre d'une application d'avertisseur de radars et de savoir laquelle des applications seraient appréciée et est-ce que le public serait prêt à payer pour enlever la publicité. (Questionnaire, annexe :4)

L'objectif de cette enquête est de nous faire comprendre :

- Comment la publicité mobile est-elle perçue ?
- Est-ce que les gens apprécient la publicité dans les applications mobiles ?
- Quel serait le modèle d'application incluant de la publicité géolocalisée le plus apprécié ?
- Est-ce que les gens sont-ils prêts à payer pour supprimer la publicité d'une application ?
- Est-ce que la plupart des personnes cliquent sur la publicité?

Au final, cinquante personnes ont accepté de répondre au questionnaire composé de :

- 13 questions fermées
- 5 questions semi-ouvertes
- 3 questions fermées facultatives
- 1 question semi-ouverte facultative

A propos du mode de diffusion, la réalisation s'est déroulée par contact direct des personnes.

17 Malheureusement, il faudrait au minimum un mois pour récolter assez de données. A ce moment, il restait moins d'une semaine avant la fin du travail.

8.1 Résultat du sondage

Cinquante personnes ont participé au questionnaire. La majorité des personnes interrogées est composée de :

Figure 21

Sexe

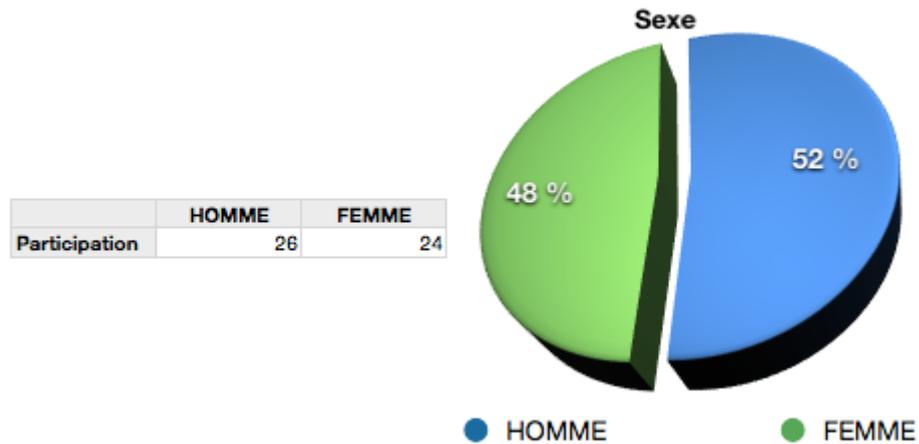


Figure 22

Tranche d'âge

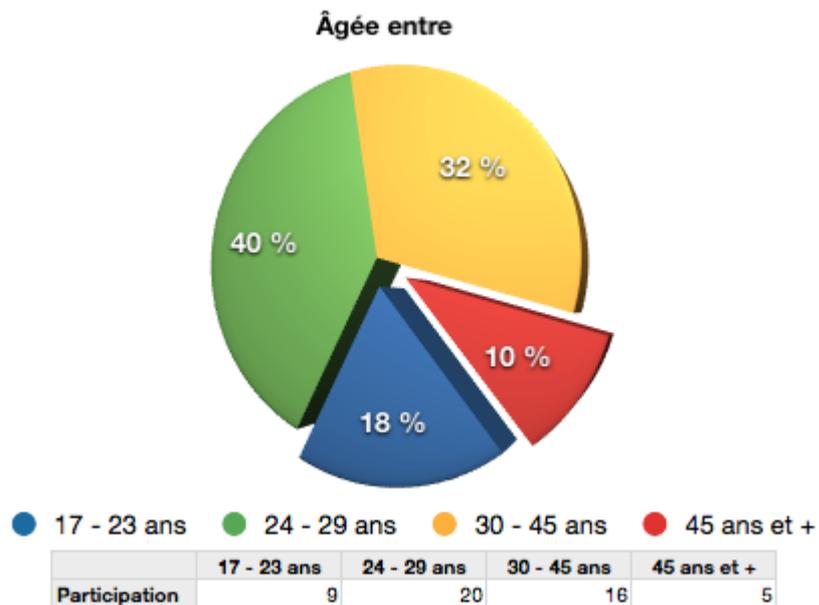


Figure 23

Pourcentage de personnes conduisant une voiture

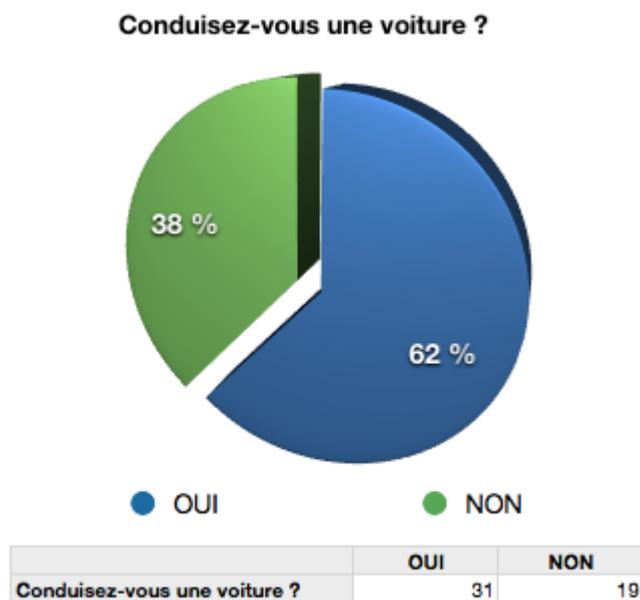


Figure 24

Pourcentage de personnes ayant utilisé une application avec publicité

Avez-vous déjà utilisé une application incluant de la publicité ?

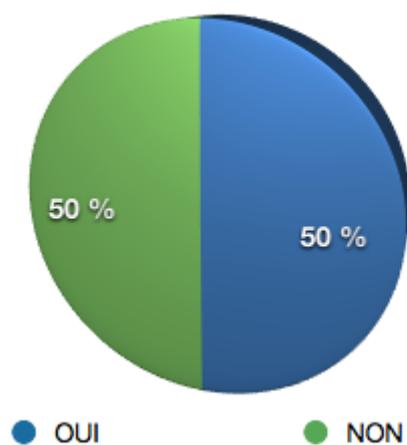
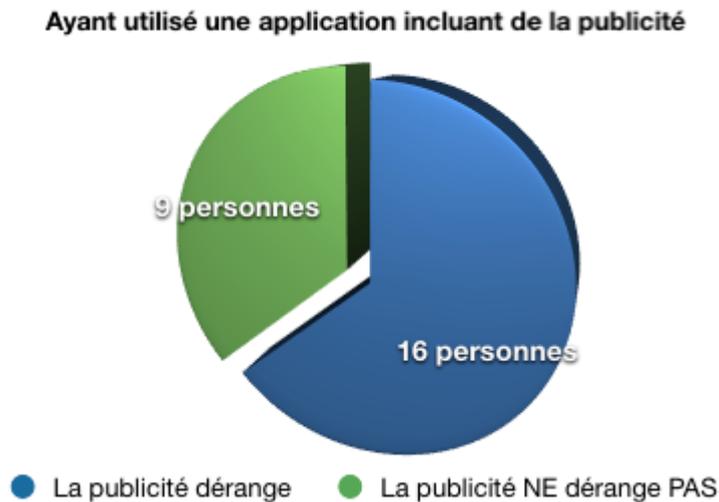


Figure 25

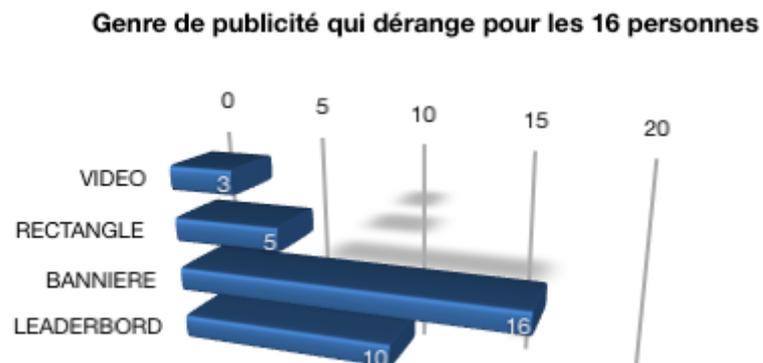
Nombre de personnes ayant utilisé une application incluant de la publicité



Pour les personnes ayant été confrontées à de la publicité dans les applications mobiles, il était intéressant de savoir si la publicité les dérangeait.

Figure 26

Genre de publicité dérangement



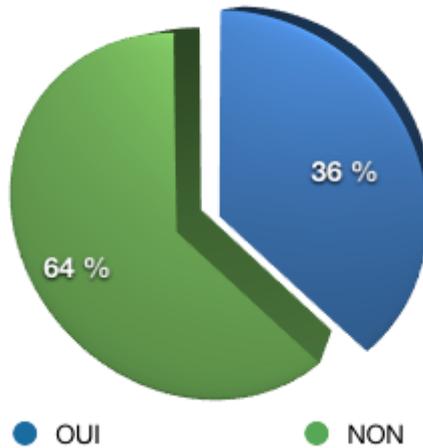
Les principaux raisons évoquées :

- Temps de chargement, ralentissement
- Mauvaise visibilité de l'application, envahissante
- Mauvaise concentration

Figure 27

Pourcentage de personnes intéressé par de la publicité géolocalisée dans les applications mobiles

Seriez-vous intéressé(e) d'avoir de la publicité géolocalisée dans les applications mobiles



Une majorité des personnes seraient intéressées par de la publicité géolocalisée dans leurs applications mobiles. Les principaux raisons favorables sont :

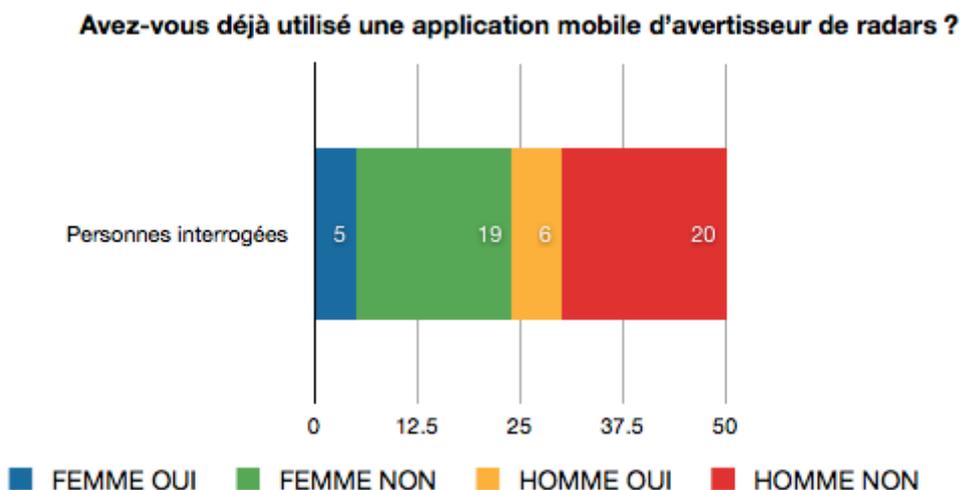
- Pratique
- Découverte
- Bons plans à proximité
- Plus pertinente

Les contre-arguments donnés sont :

- Ennuyante
- Inutile, Pas une source d'information

Figure 28

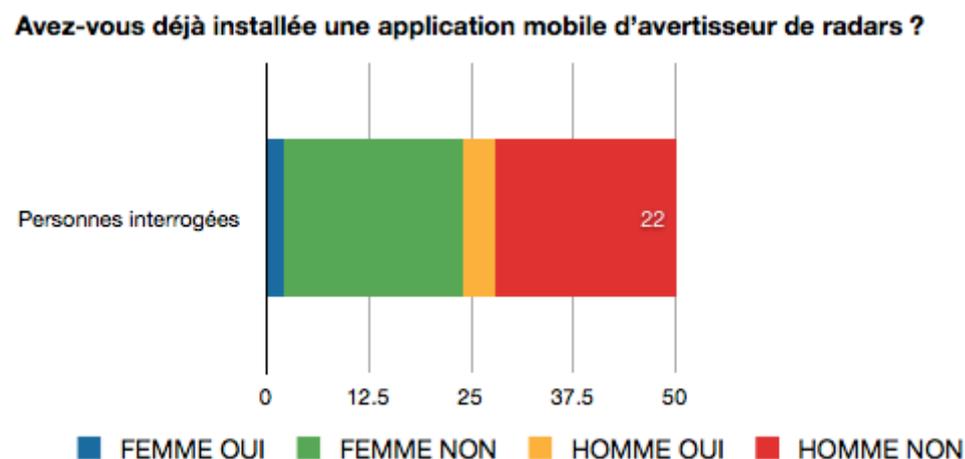
Nombre de personnes selon le sexe ayant utilisé un avertisseur de radars



Plus de la moitié des personnes n'ont jamais utilisé ce genre d'application.

Figure 29

Nombre de personnes selon le sexe ayant installé un avertisseur de radars

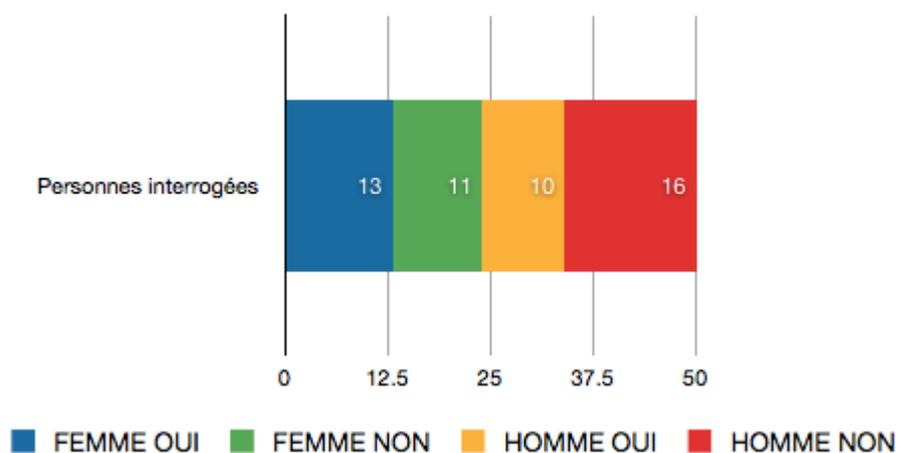


Il en va de même pour les personnes ayant installé ce genre d'application.

Figure 30

Nombre de personnes selon sexe à propos de la dangerosité d'un avertisseur de radars

Pensez-vous qu'il est dangereux d'utiliser ce genre d'application en roulant ?



La raison défavorable donnée concerne la sécurité (inattention). Pareil pour son contre-argument axé sur la prévention.

Figure 31

Pourcentage de vote concernant le classement de l'application 1

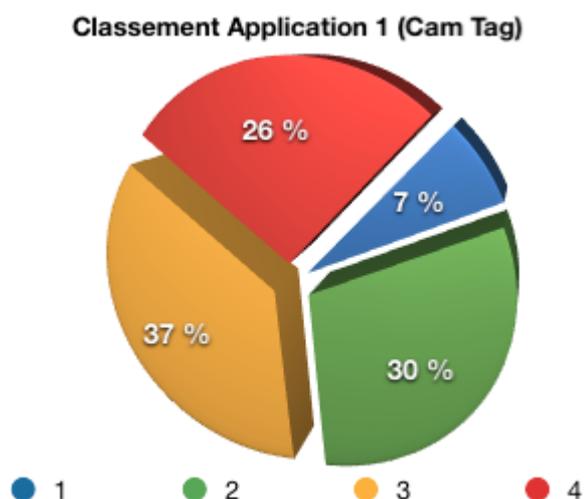


Figure 32

Pourcentage de vote concernant le classement de l'application 2

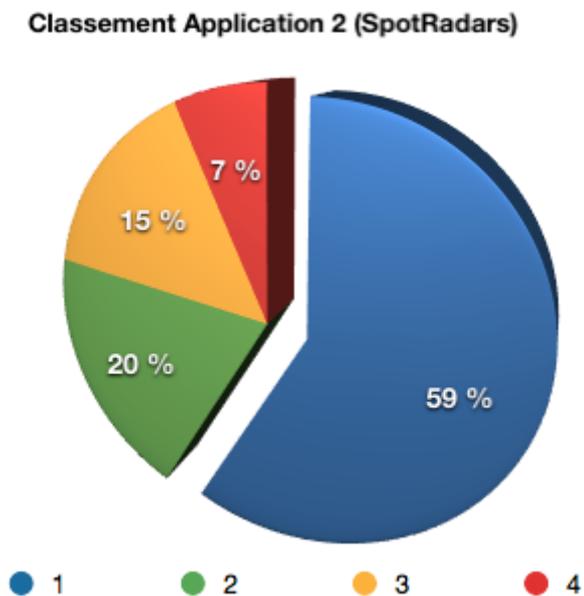


Figure 33

Pourcentage de vote concernant le classement de l'application 3

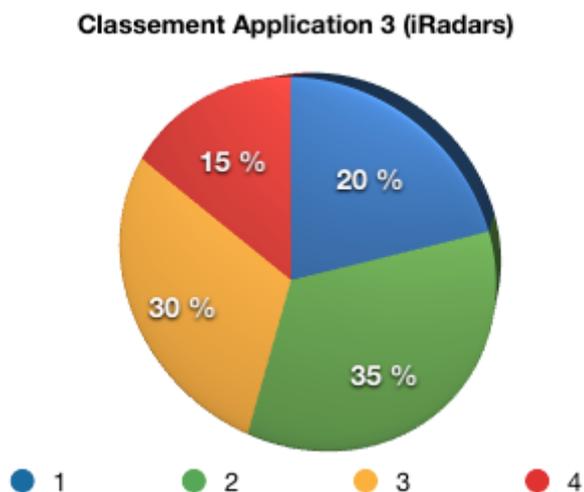
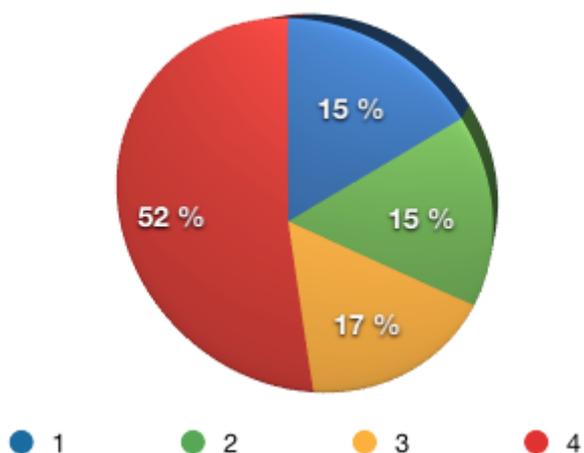


Figure 34

Pourcentage de vote concernant le classement de l'application 4

Classement Application 4 (StopRadars)

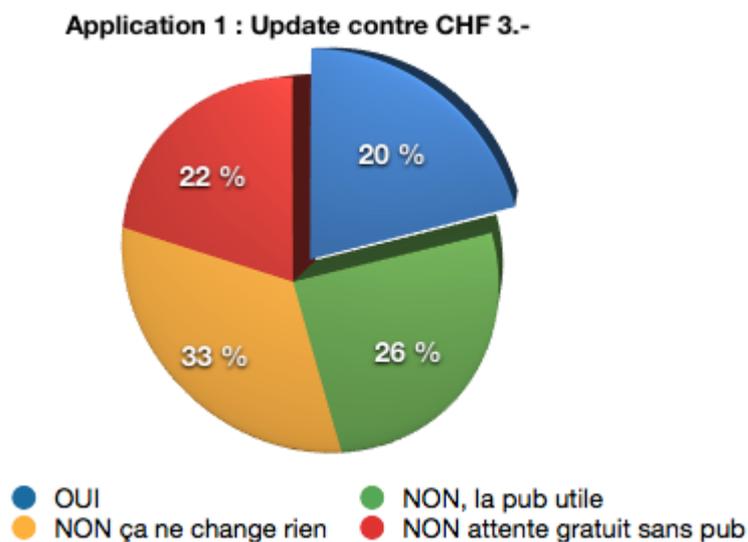


A l'aide de ces données, nous pouvons classer par ordre de préférence parmi ces quatre applications.

1. Application 2 (SpotRadars) 59 %
2. Application 3 (iRadars) 35 %
3. Application 1 (Cam Tag) 37 %
4. Application 4 (StopRadars) 52 %

Figure 35

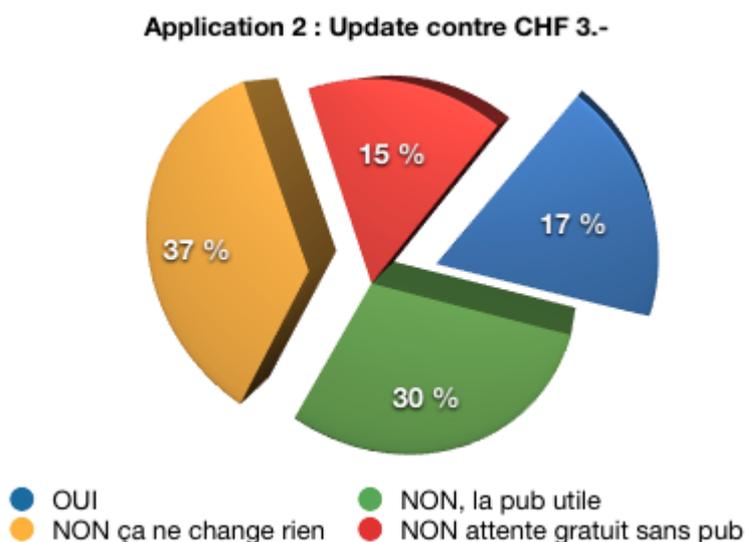
Pourcentage d'avis concernant la mise à jour de l'application 1



Seulement 20 % des personnes sont d'accord de supprimer la publicité à condition de payer CHF 3.-. Par contre, la majorité des personnes dans le « Non » (33 %) sont indifférentes à la publicité.

Figure 36

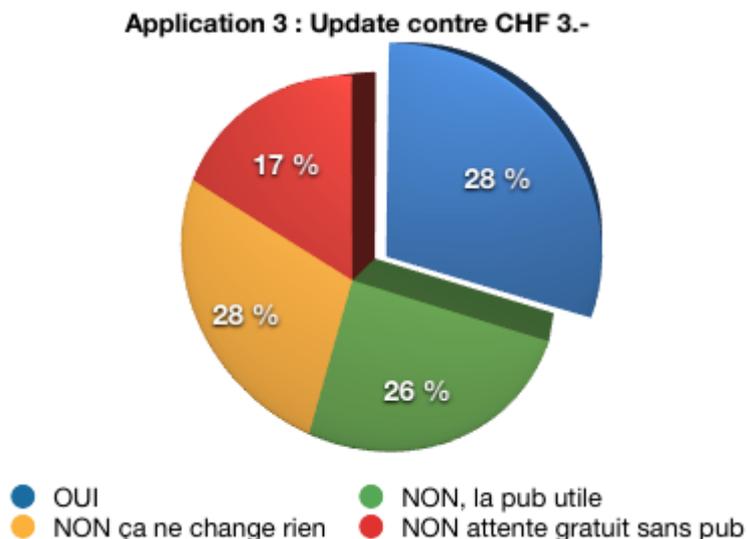
Pourcentage d'avis concernant la mise à jour de l'application 2



Même constat que pour l'application 1 concernant la catégorie des « Non ».

Figure 37

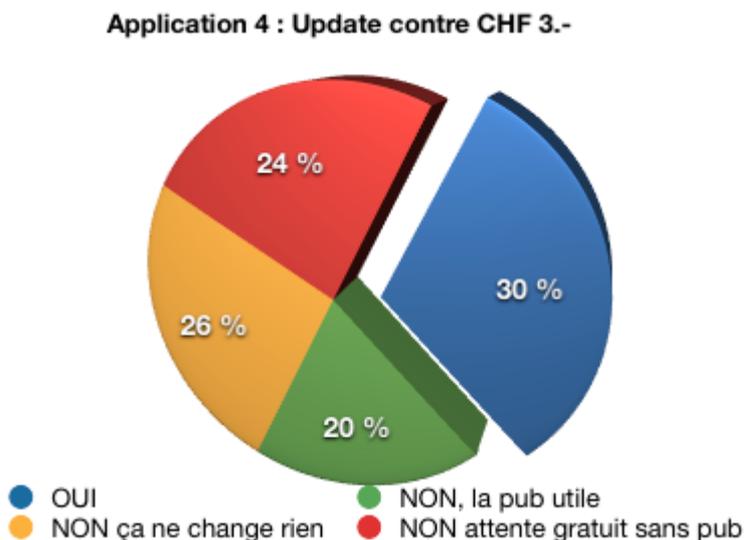
Pourcentage d'avis concernant la mise à jour de l'application 3



Égalité concernant la catégorie des « NON » et « OUI ». 28 % des personnes seraient d'accord de payer alors que d'autre sont indifférentes à la publicité.

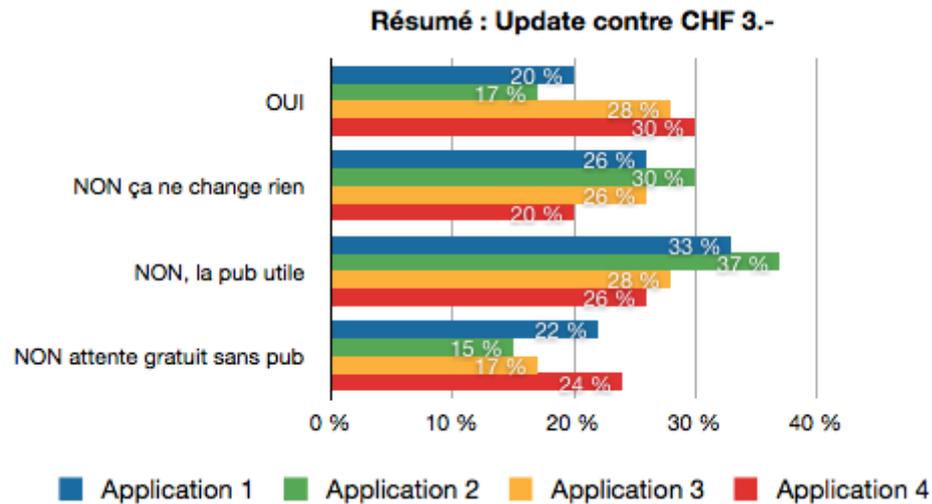
Figure 38

Pourcentage d'avis concernant la mise à jour de l'application 4



Cette application est la moins apprécié des autres. Par conséquent, la majorité des gens sont prêts à déboursier CHF 3.- pour supprimer la publicité.

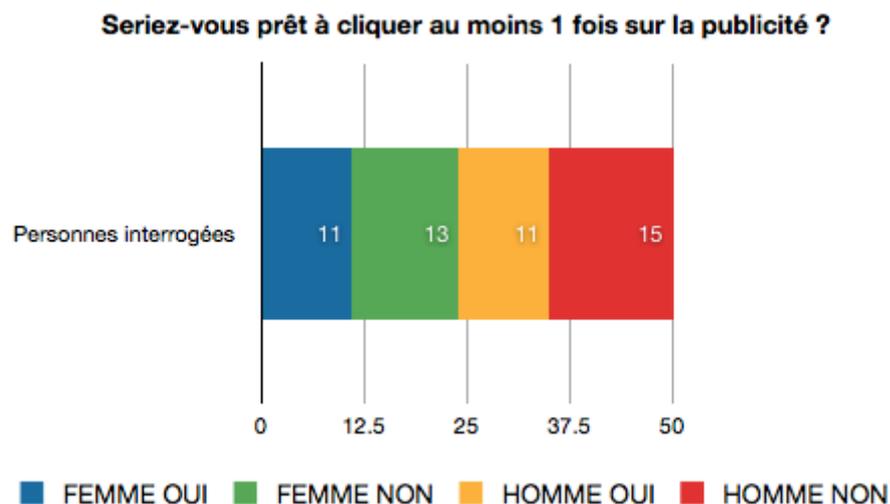
Figure 39
Résumé des pourcentages d'avis concernant la mise à jour des applications



En clair, l'application 4 a tendance à inciter l'utilisateur à effectuer une mise à jour en acceptant de payer le prix. Elle est aussi celle qui pousse les gens à attendre qu'une application soit gratuite.

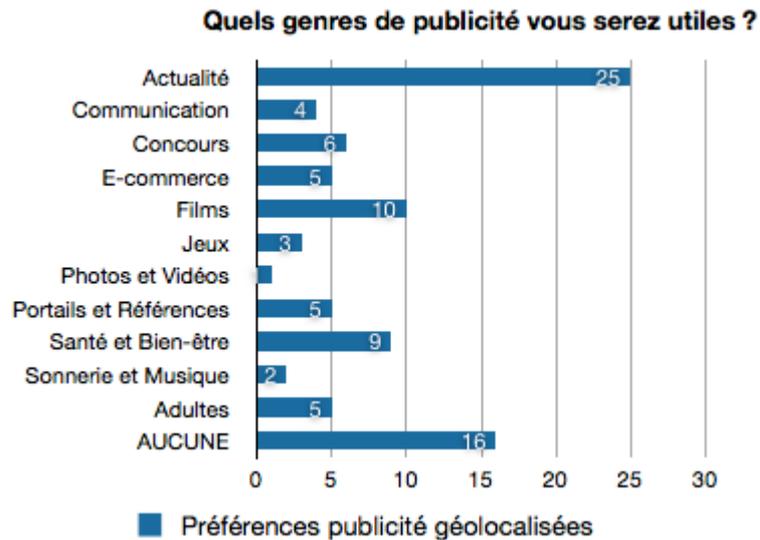
L'application 2 est l'application qui possède la majorité pour l'utilité de la publicité et pour son indifférence.

Figure 40
Nombre de personnes selon le sexe prêt à cliquer au moins 1 fois sur la publicité



Une majorité pour le « NON » et une égalité pour le « OUI » des sexes.

Figure 41
Préférences concernant les genres de publicité dans le cadre d'un avertisseur de radar

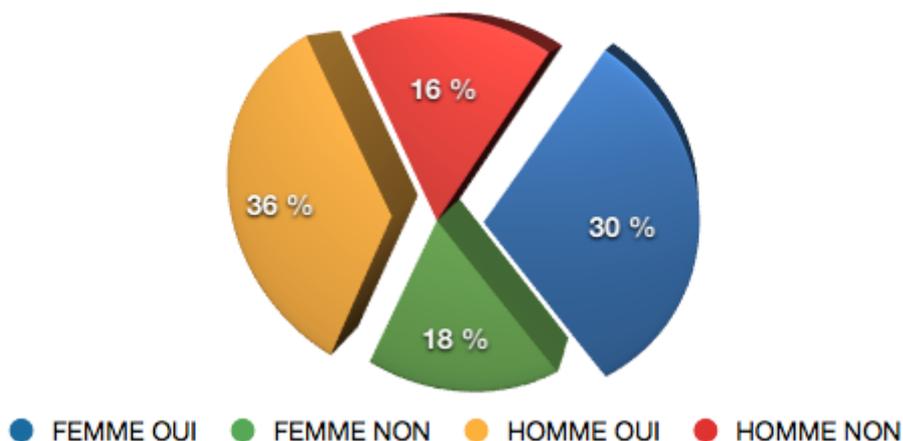


Il y a 16 personnes qui n'approuvent pas la publicité. Pour le reste, l'actualité, les films et la santé les intéressent. Omis cela, les utilisateurs seraient aussi intéressés d'avoir de la publicité lié à la même compagnie ou des applications faisant parties de la même famille (application de trafics-route, navigation,...)

Figure 42

Pourcentage de personnes selon le sexe tenté de télécharger une des quatre applications

... Seriez-vous tenté de télécharger une des quatre applications présentés avant ?



Après avoir présenté les modèles des quatre applications, la majorité des personnes sont tentées de télécharger une des applications proposées. A priori, l'application risquant d'être téléchargée serait l'application 2.

8.2 Synthèse du sondage

Les observations que nous pouvons faire sont :

- La publicité en générale n'est pas fortement appréciée par les gens. Par contre, 64 % des personnes interrogées trouveraient utiles d'avoir de la publicité pertinente (lieu, cible) c'est-à-dire géolocalisée.
- Concernant la moitié des personnes interrogées ayant déjà été confrontées par de la publicité dans les applications mobiles, elles (16 personnes) n'apprécient pas la publicité de type bannière et leaderbord. Ces raisons sont dues à la mauvaise gestion de l'ergonomie des applications utilisées. Par contre concernant notre étude, il y a eu 16 personnes refusant d'avoir de la publicité dans les applications.
- 50 % personnes ont déjà vu de la publicité dans leurs applications mobiles. Cependant, nous ne savons pas si cette publicité était géolocalisée.

- La majorité des personnes n'avaient jamais installé ou utilisé une application d'avertisseur de radars. Cependant, 66 % des sondés seraient d'accord de télécharger une de nos quatre applications.
- 27/50 personnes pensent qu'une application de ce genre ne présente pas de risques pour la conduite. C'est à chacun de faire son opinion, nous n'allons pas débattre sur ce sujet et dire qui a raison ou tort.
- A propos des quatre applications, celle qui a le plus de chance de plaire aux utilisateurs est l'application 2 (SpotRadar). 59 % des sondés ont classé l'application dans la liste de leur premier choix.
- L'idée précédente est confortée par la deuxième application appréciée qui est l'application 3. Certes de genre différent, la publicité est affichée lorsqu'il n'y a pas de dangers.
- L'application la moins appréciée est sans conteste l'application 4. La majorité des personnes nous ont fait part oralement d'une publicité très intrusive d'une part par sa taille et son contexte d'apparition. Cependant, c'est celle qui incite les gens à vouloir payer pour supprimer cette gêne.
- Concernant le format publicitaire proposé, nous pouvons suggérer qu'il n'y pas vraiment d'impact sur l'appréciation des personnes. La majorité des sondés ont répondu en se basant principalement sur le contexte d'apparition. Les deux meilleures applications appréciées ont des formats différents, par contre, elles ont un contexte assez similaire.
- Concernant l'application 3, nous pouvons observer qu'elle fait partie des deux applications qui pousseraient les gens à mettre à jour l'application. Nous pouvons suggérer que le format de publicité n'est pas très apprécié des utilisateurs.
- Presque la moitié des personnes (22/50) seraient prêtes à cliquer sur la publicité. Si ces 22 personnes cliquaient une fois par jour sur une publicité, le développeur pourrait percevoir au minimum $0.66 \$ (0.03 \$ * 22)$ par jour. Ce chiffre serait insignifiant mais avec par exemple 1 million d'utilisateurs, ceci serait différent ($0.03 \$ * 1'000'000 * 44\% = 13'200 \$/\text{jour}$).
- Nous pouvons noter que les publicités « Photos et Vidéos », « Sonneries et musiques » et « Jeux » n'intéresse pas les sondés. Par contre, un élément

intéressant est que les gens apprécient d'avoir de la pub liée au type d'application utilisé.

- Pour finir, si les utilisateurs souhaiteraient télécharger une des quatre applications, 66 % d'entre eux seraient d'accord. Avec cela, nous serions sûrs que l'application 2 aurait été choisie.

Suite à ces observations, nous pouvons conclure que l'application la plus appréciée est l'application 2 (SpotRadars).

Cette application, contrairement aux autres, utilise une publicité de type bannière apparaissant de manière « intelligente ». La publicité apparaît à un moment évitant de gêner la visibilité de l'utilisateur. Son jumeau de forme, l'application 1 (Cam Tag), affiche constamment de la publicité, ce qui est vraisemblablement inapproprié en raison de l'exposition permanente de l'utilisateur au message publicitaire.

L'application 4 est la moins appréciée de toutes. Ceci pourrait être dû à son contexte d'apparition qui obstrue une information considérée comme importante par l'utilisateur.

En général, les sondés sont prêts à cliquer sur la publicité dans le cas d'une publicité liée au contexte de l'application.

9. Conclusion

La majorité des systèmes d'exploitation mobiles ont déjà implémenté le système de géolocalisation poussant ainsi les développeurs à développer des applications de géolocalisation. Ceci ouvre la voie à la publicité géolocalisée.

Cette publicité est un format marketing très utile pour les compagnies ou développeurs souhaitant percevoir un revenu à travers ce service. Ceci est aussi utile pour les compagnies souhaitant se promouvoir. Par contre, il peut s'avérer difficile de choisir une plateforme parmi toutes celles disponibles actuellement. Il n'existe pas encore de monopole sur le marché. Le jour où une entreprise aura monopolisé ce service, ceci sera profitable pour tous les acteurs économiques (clients, publieurs, annonceurs).

Concernant le développeur, il lui est aisé d'avoir accès à ce service. La majorité des services proposent des SDK facilement implémentable pour leurs applications.

A propos de l'application d'avertisseur de radar incluant de la publicité géolocalisée, le meilleur moyen d'avoir un revenu sur le long terme serait de proposer l'application 2 (SpotRadars) à condition d'avoir beaucoup d'utilisateurs. Dans ces conditions, la publicité serait un moyen rentable sur la durée. Mais il est aussi possible d'avoir une autre stratégie à double tranchant qui est celle de proposer l'application 4 (StopRadars) incitant les gens à acheter une et une seule fois l'application ou à les faire fuir de l'application.

Comme nous l'avons constaté, le format publicitaire peut avoir un effet sur l'appréciation d'une application de la part de l'utilisateur ainsi que sa pertinence d'apparition. Une application mal appréciée peut avoir des conséquences néfastes à long terme pour le développeur. Une publicité qui gêne la visibilité ou qui agresse directement ou indirectement l'utilisateur peut faire naître un sentiment de dégoût pour la publicité. C'est comme jouer à un jeu vidéo et de voir subitement apparaître une publicité qui n'a rien à voir avec ce qui se passe sur l'écran. Il est important d'introduire discrètement et au moment opportun la publicité dans une application mobile sans perturber l'ergonomie de l'application.

A mon avis, est-ce que ces résultats concorderont avec la réalité ? Si oui, est-ce que cette stratégie s'applique forcément à tous les applications mobiles ? Est-ce qu'on peut reproduire cette technique sur tous les supports mobiles et pourquoi pas sur les tablettes numériques ?

Personnellement, ce travail était très enrichissant car il touche différents profils de l'informaticien de gestion. Il m'a permis de pouvoir découvrir la notion de géolocalisation lié à la publicité et aux applications mobiles en effectuant des recherches d'information. Grâce à cette notion, j'ai également eu la possibilité de développer une application Android, de l'apprentissage à sa publication. De plus, j'ai eu la possibilité de pouvoir appliquer une méthodologie, le SCRUM, dans un contexte professionnel. Pour finir, j'ai eu l'occasion de pouvoir effectuer une étude auprès de personnes.

Webographie

LARS, Vogel, *Android Development tutorial [en ligne]*, 04.07.2009, 24.10.2011. http://www.vogella.de/articles/Android/article.html#first_uelement (consulté le 06.09.2011).

Mind the Robot, *Android Guts: Intro to Loopers and Handlers [en ligne]*, 03.06.2010, 03.06.2010. <http://mindtherobot.com/blog/159/android-guts-intro-to-loopers-and-handlers/> (consulté le 10.09.2011)

Movable Type Scripts, *Calculate distance, bearing and more between Latitude/Longitude points [en ligne]*, 2011, <http://www.movable-type.co.uk/scripts/latlong.html> (consulté le 27.09.2011)

DRUET Nicolas, *[Android][Java] Rendre son application compatible avec tous les Android-phones (ou presque)[en ligne]*, (2010), <http://blog.developpez.com/android23/p8665/android/rendre-son-application-comptible-avec-to/#c41874> (consulté le 08.09.2011)

Google, *App Galaxy Le Guide par Google [en ligne]*, 2011, http://www.guidetotheappgalaxy.com/?utm_source=admob&utm_medium=web&utm_campaign=ag#/developersguide/2, (consulté le 08.09.2011)

Google, *Welcome to AdMob Help [en ligne]*, 2005-2009, <http://helpcenter.admob.com/>, (14.10.2011).

ACRA, *Application Crash Report for Android [en ligne]*, 2011, <http://code.google.com/p/acra/>, (consulté le 12.10.2011),

AdMob, *Start - AdMob for developers [en ligne]*, 2010, 29.11.2010, http://developer.admob.com/wiki/Main_Page, (consulté le 14.10.2011).

GoMo News, *Top 51 Mobile Advertising firms plus top 9 Mobile Advertising and Search providers*, 10.11.2010, <http://www.gomonews.com/top-51-mobile-advertising-firms-plus-top-9-mobile-advertising-and-search-providers/>, (14.10.2011).

mobiThinking, *mobiThinking guide to mobile advertising networks (2011)[en ligne]*, (2011), <http://mobithinking.com/mobile-ad-network-guide>, (14.10.2011).

Android, *Android Developers [en ligne]*, (2008-2011), <http://developer.android.com/index.html>, (05.09.2011).

Google, *Android Market [en ligne]*, 2008-2011, <https://market.android.com/?hl=fr>, (consulté le 25.10.2011).

JACKSON Rob, *Video Tutorial: Creating An Android Application Icon [en ligne]*, 04.06.2009, <http://phandroid.com/2009/06/04/video-tutorial-creating-an-android-application-icon/>, (consulté le 22.10.2011).

Bibliographie

GUIGNARD Damien, CHABLE Julien, ROBLES Emmanuel, *Programmation Android De la conception au déploiement avec le SDK Google Android 2*, Paris, Éditions Eyrolles, 2011.

MOTTIER Cyril, *Développez pour Android : [des applications mobiles optimisées]*, Paris, Éditions Digit Books, 2011.

DERIAZ Michel, *Server API*, 28.07.2009, 24.10.2011

Annexe 1

Interface des applications avec publicité

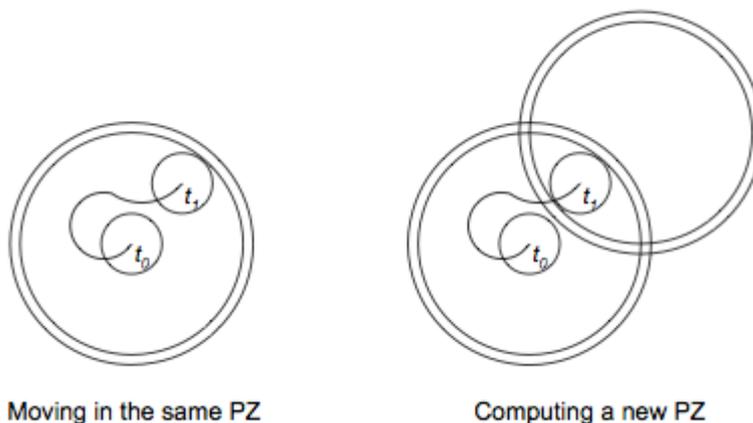
	Démarrage	En mouvement sans radars à proximité	A l'arrêt sans radars à proximité (vitesse <= 5km/h)	En mouvement avec radar(s) à proximité (vitesse > 5km/h)	A l'arrêt avec radar(s) à proximité (vitesse <= 5km/h)	Tagging 1	Tagging 2
Application 1 (Cam Tag)							
Application 2 (SpotRadars)							
Application 3 (iRadars)							
Application 4 (StopRadars)							

Annexe 2

Optimizing connections (Extract from API FoxyTag)

The following algorithm can be used in order to reduce the number of connections to the server: Let's call protected zone (PZ) the area that contains the downloaded tags. A new connection is then made only every given delay (here it is 5 minutes) or if the user requests tags that are outside of the protected zone. A protected zone is defined by 5 main attributes:

- pzLat and pzLon: Two numbers, the latitude and the longitude of the center of the PZ. A PZ is always a circle. A user can give its current position if he wants to see the tags around him, but can also give coordinates of another point to see the tags in another place.
- radiusOut: This is the radius of the external circle of the PZ. It defines the area where the tags have been downloaded.
- radiusIn: This is the radius of the internal circle of the PZ. This value is used as a trigger to ask a new connection to the server. Since connections to a remote server are not instantaneous, this internal circle allows the application to update its protected zone before the user exits the former PZ.
- timeout: This is the maximum delay between two connections to the server. Even if the user doesn't quit its PZ, a connection is made time to time in order to guaranty the freshness of the information.



The first time tags are requested, the algorithm will compute a PZ centered on the current position (or on the position where the tags are requested). But in order to limit the communications with the server, the next time a request is made, the PZ will be shifted according to the last moves of the user. In the left figure we see an example. The user starts his application at time t_0 . The little circle represents the radius of

visibility (simply called radius), or the area where the tags are visible on the mobile device. The big inner circle is the internal circle of the PZ (called radiusIn), which acts as a trigger (when the user requests tags that are outside, a new PZ is computed). The big outer circle is the external circle of the PZ (called radiusOut), which is the area where the tags are actually present in the memory of the mobile device. While the user moves close to the center of the PZ, the tags are already in memory. If the user is still at the center of the PZ when the timeout is reached, a new connection is made to the server in order to update the tags, but the PZ stays at the same position (centered on the user). But if the user moved away, then the new PZ will be shifted in the same direction. For instance, if our user reaches the inner circle of the PZ at time t_1 , a new PZ is requested and the latter will be shifted like it is shown in the right figure. The distance of the shift is proportional of the distance that the user moved since last update, so if he didn't move, the shift is null (and the new PZ is centered on the current position).

Annexe 3

Fonctionnalités l'API + Application

- **Tag Buffer** : A buffer that contains the tags around the user and that connects automatically to the server to update its content when necessary. Necessary means when the user is going to quit the covered area or every five minutes, as required by the server API.
- **Main Screen** : Default screen, when there is no speed camera around. Shows a picture, the current speed in green if ≥ 20 km/h otherwise in red, the word "GPS" in green if we have precise position otherwise in red, and the word "DATA" in green if we are up-to-date with the data otherwise in red. An additional message is shown when all the previous texts are in green (speed, GPS, DATA): "Tap the screen to add a speed camera, but only when you are as close as possible to it". When the user taps the screen, the application switches to the Tagging screen.
- **Tag Screen** : The screen to show when a visible (in front of the driver plus-minus 45° , with heading difference $< 45^\circ$) camera appears closer than 600 meters. Switches back to main screen when all cameras disappeared from the screen. EyeRadars should be taken as an example, but the Test mode text shall not be shown. The current speed, the words GPS and DATA are shown like in the main screen and if they are all green an additional message is shown: "Tap the screen to add, confirm or delete a speed camera, but only when you are as close as possible to it". When the user taps the screen, the application switches to the Tagging screen.
- **Tagging Screen** : The application allows to tag only if the user entered a valid username and password. If not, the application shows the following message: "Participating by tagging or confirming existing tags will automatically create trust links between you and people tagging in a similar way, and you will therefore benefit from more reliable data in return. To participate, please register." The word register is a link to the Settings screen. If the user can tag, then he first chooses the type of speed camera and then its direction. On both screens, a very visible message "Position memorized" must be shown.

- Settings screen : Tag Buffer : A buffer that contains the tags around the user and that connects automatically to the server to update its content when necessary. Necessary means when the user is going to quit the covered area or every five minutes, as required by the server API.
- Main Screen : Default screen, when there is no speed camera around. Shows a picture, the current speed in green if ≥ 20 km/h otherwise in red, the word "GPS" in green if we have precise position otherwise in red, and the word "DATA" in green if we are up-to-date with the data otherwise in red. An additional message is shown when all the previous texts are in green (speed, GPS, DATA): "Tap the screen to add a speed camera, but only when you are as close as possible to it". When the user taps the screen, the application switches to the Tagging screen.
- Tag Screen : The screen to show when a visible (in front of the driver plus-minus 45° , with heading difference $< 45^\circ$) camera appears closer than 600 meters. Switches back to main screen when all cameras disappeared from the screen. EyeRadars should be taken as an example, but the Test mode text shall not be shown. The current speed, the words GPS and DATA are shown like in the main screen and if they are all green an additional message is shown: "Tap the screen to add, confirm or delete a speed camera, but only when you are as close as possible to it". When the user taps the screen, the application switches to the Tagging screen.
- Tagging Screen : The application allows to tag only if the user entered a valid username and password. If not, the application shows the following message: "Participating by tagging or confirming existing tags will automatically create trust links between you and people tagging in a similar way, and you will therefore benefit from more reliable data in return. To participate, please register." The word register is a link to the Settings screen. If the user can tag, then he first chooses the type of speed camera and then its direction. On both screens, a very visible message "Position memorized" must be shown.
- Audio alarm : When a visible tag comes closer than 15 seconds to the user, according to the current speed, an audio alarm warns the user. A tag can launch only once an audio alarm while visible on the screen; in other words, a tag must first completely disappear from the screen before being able to launch a second alarm.

- A menu on the bottom of the different screens (except the screens that appear during the tagging process) proposes to switch between some screens. The items are "Cameras", "Settings", "Help".
- The help screen shows a page with some useful information.

Annexe 4

Questionnaire

Enquête : Publicité géolocalisée

Ce questionnaire est **anonyme**, les résultats nous permettront de savoir comment vous percevez la publicité géolocalisée dans une application mobile.

Ce questionnaire dure au maximum 10 minutes.

Ce questionnaire permet d'avoir une base d'étude pour le travail de bachelier mené actuellement.

Un questionnaire facultatif est disponible à la fin si vous souhaitez y répondre.

Merci de votre participation

A savoir

La publicité géolocalisée est une publicité (textuelle, sonore, visuelle) utilisant la position géographique de l'utilisateur. Par exemple, un consommateur peut recevoir une publicité locale (Restaurants, Fast Food, etc.) à proximité de sa position géographique donnée par le GPS.

Exemple d'application mobile d'avertisseur de radars.

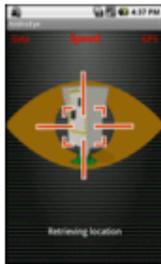
Une application d'avertisseur de radars permet de signaler les radars à proximité à l'aide d'un signal sonore. Elle utilise un système collaboratif qui permet de récupérer les radars recensés par les conducteurs.

L'utilisateur de l'application a la possibilité de taguer un radar afin d'avertir les autres utilisateurs de l'application.

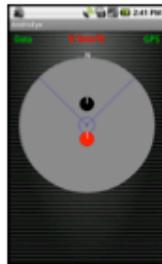
Taguer signifie que l'utilisateur envoie les coordonnées du radar qu'il a rencontré.

En clair, le conducteur lance l'application et pose son téléphone à un endroit visible. Il peut observer l'apparition de radars sur son écran ou être averti lorsqu'il s'approche d'un radar ou avertir d'un radar (le taguer) pour faire profiter la communauté d'utilisateurs.

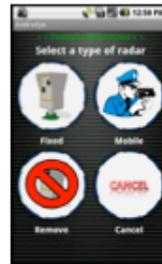
Pour illustrer le fonctionnement :



Lorsqu'il n'y a pas de radar à portée.



Des radars sont affichés sur un scanner lorsqu'elles sont à proximité de nous. Ces radars sont représentés sous forme de TAG.



Le conducteur peut signaler un radar ou le confirmer en touchant la fenêtre.



Début du questionnaire

Q1. Êtes-vous :

- un homme une femme

Q2. Vous avez entre :

- 17 - 23 ans 24 - 29 ans 30 - 45 ans 45 ans et +

Q3. Conduisez-vous une voiture ?

- Oui Non

Q4. Avez-vous déjà utilisé une application mobile incluant de la publicité ?

- Oui Non

Q4a. Si oui,

La publicité me dérangeait. Pourquoi ?

La publicité ne me dérangeait pas.

Q4b. Quels étaient les formats de publicité d'application mobile rencontrés ? (Plusieurs réponses possibles)



Vidéo



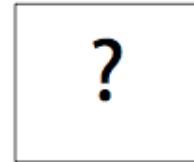
Rectangle



Bannière



Splash screen



Autre

Q5. Seriez-vous intéressé(e) d'avoir de la publicité géolocalisée dans les applications mobiles ?

Oui Non

Pourquoi ?

Q6. Avez-vous déjà utilisé une application mobile d'avertisseur de radars ?

Oui Non

Q7. Avez-vous déjà installé une application mobile d'avertisseur de radars ?

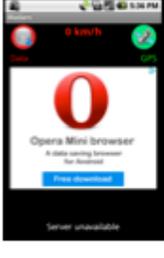
Oui Non

Q8. Pensez-vous qu'il est dangereux d'utiliser ce genre d'application en roulant ? (Vous-même ou le passager)

Oui Non

Pourquoi ?

Q9. Vous souhaitez télécharger une application d'avertisseur de radars. Seulement quatre applications vous sont proposées affichant de manière différente de la publicité géolocalisée. Ces applications sont gratuites.

	En mouvement sans radar(s) à proximité	A l'arrêt sans radar(s) à proximité (vitesse ≤ 5 km/h)	En mouvement avec radar(s) à proximité (vitesse > 5 km/h)	A l'arrêt avec radar(s) à proximité (vitesse ≤ 5 km/h)
<p>Application 1 (Cam Tag)</p> <p>Le contenu de la publicité est mise à jour à intervalle régulier et la publicité ne peut pas disparaître de l'écran</p>				
<p>Application 2 (SpotRadars)</p> <p>A l'arrêt du véhicule, la publicité apparaît à l'écran et disparaît dans le cas contraire</p>				
<p>Application 3 (iRadars)</p> <p>La publicité apparaît à l'écran lorsqu'il n'y a pas de radars à proximité.</p>				
<p>Application 4 (StopRadars)</p> <p>La publicité apparaît à l'écran lorsque le véhicule est à l'arrêt même si un radar est à proximité.</p>				

Classez ces applications par ordre de préférence. (1 à 4)

Application 1 (Cam Tag)	<input type="text"/>
Application 2 (SpotRadars)	<input type="text"/>
Application 3 (iRadars)	<input type="text"/>
Application 4 (StopRadars)	<input type="text"/>

Q10. Si vous avez installé l'**application 1 (Cam Tag)**. Cette application est complètement fonctionnelle et répond à vos attentes (tous les radars sont signalés, l'alarme sonore est assez forte,...). Seriez-vous prêt à payer CHF. 3.- pour supprimer la publicité ?

- Oui, je ne supporte pas la publicité.
- Non, la publicité peut m'être utile (resto-route, stations services, promotions, etc.
- Non, avec ou sans ça ne change rien.
- Non, j'utilise cette application en attendant de chercher une autre application gratuite sans publicité.

Q11. Même question pour l'application 2 (SpotRadars).

- Oui, je ne supporte pas la publicité.
- Non, la publicité peut m'être utile (resto-route, stations services, promotions, etc.
- Non, avec ou sans ça ne change rien.
- Non, j'utilise cette application en attendant de chercher une autre application gratuite sans publicité.

Q12. Même question pour l'application 3 (iRadars).

- Oui, je ne supporte pas la publicité.
- Non, la publicité peut m'être utile (resto-route, stations services, promotions, etc.
- Non, avec ou sans ça ne change rien.
- Non, j'utilise cette application en attendant de chercher une autre application gratuite sans publicité.

Q13. Même question pour l'application 4 (StopRadars).

- Oui, je ne supporte pas la publicité.
- Non, la publicité peut m'être utile (resto-route, stations services, promotions, etc.
- Non, avec ou sans ça ne change rien.
- Non, j'utilise cette application en attendant de chercher une autre application gratuite sans publicité.

Q14. Seriez-vous prêt à cliquer au moins 1 fois sur la publicité ?

Oui (intérêt)

Non. Pourquoi ?

Q15. Quels genres de publicité vous seraient utiles ?

Actualités, Sport et Météo

Communication

Concours

E-commerce

Films

Jeux

Photos et Vidéos

Portails et Références

Santé et Bien-être

Sonnerie et Musique

Adultes

Autres :

Q16. Vous cherchez une application d'avertisseur de radars. Seriez-vous tenté de télécharger une des quatre applications présentés avant ?

Oui

Non. Pourquoi ?

Merci d'avoir pris la peine de répondre à ce questionnaire

Facultatif

F1. Possédez-vous un téléphone Android ?

Oui. Version du SDK ou modèle de téléphone

Non. Un autre smartphone ? (iPhone, blackberry)

F2. A quelle fréquence téléchargez-vous des applications mobiles ?

Jamais

Rarement

Souvent

F3. Quel(s) type(s) d'applications téléchargez-vous ?

Majoritairement

Gratuit

Payant

Utilitaires

Voyages

Jeux

Réseaux sociaux

Nouveautés

Outils

Finances

Shopping

Divertissement

Info pratiques

Autres :

F4. Devant deux applications proposant les mêmes fonctionnalités, quels seraient vos critères de choix ?

Design (Couleurs, formes)

Notes des utilisateurs / Commentaires

Conseils d'un(e) ami(e)

Gratuit

Payant

Au hasard

Merci encore !

Annexe 5 Planning SCRUM



PrintScreen via IceScrum