

PhoneGap

Versus



Xamarin

Etude Technologique

Contents

Introduction.....	3
Présentation des technologies	3
Phonegap.....	3
Xamarin	3
Pour quels besoins ?.....	4
Comment ça fonctionne ? Les models	4
Phonegap.....	5
Xamarin	6
Comparatif.....	7
Développement.....	7
Les environnements de développement.....	7
La communauté, les librairies et framework additionnels.....	8
Le coût	8
Déploiement.....	8
Plateformes	8
Runtime	9
Adaptabilité aux plateformes et responsive design.....	9
Performance	9
Expérience utilisateur.....	9
Conclusion	10

Introduction

Dans le cadre de l'enseignement d'adaptation des interfaces à l'environnement, j'ai décidé de présenter une étude technologique sur deux solutions de cross plateforme mobile. La première solution vient du monde du web, il s'agit de PhoneGap, la seconde du monde de Microsoft, il s'agit de Xamarin.

Dans un premier temps nous verrons les fonctionnements de ces technologies puis dans un second temps un comparatif, dans le but de déterminer les avantages et les inconvénients de chacune d'elles.

Présentation des technologies

Phonegap

PhoneGap à l'origine est une idée développée par Nitobi software. Elle fut considérée à l'époque de ses débuts comme l'une des technologies du web les plus prometteuses, en témoigne son prix gagné à la conférence O'Reilly Media's Web 2.0 de 2009. Attiré, à raison, par son potentiel, Adobe acquit Nitobi Software en 2011. Phonegap utilise le système d'Apache Cordova pour fonctionner . À titre de comparaison, on peut considérer que Cordova est à PhoneGap , ce que WebKit est à Chrome ou Safari . À ce jour, plus de quatre cent mille développeurs utilisent Phonegap pour réaliser leurs applications mobiles.

Xamarin

Xamarin est une technologie plus récente, elle a été fondée par la société du même nom en 2011 avec pour but de proposer une alternative aux solutions web de cross plateforme. Pour cela, elle propose une solution complète avec tous les outils de développement se rapprochant du développement natif. Elle est à ce jour la solution de cross plateforme la plus utilisée par les développeurs avec plus de sept cents milles adhérents.

Pour quels besoins ?

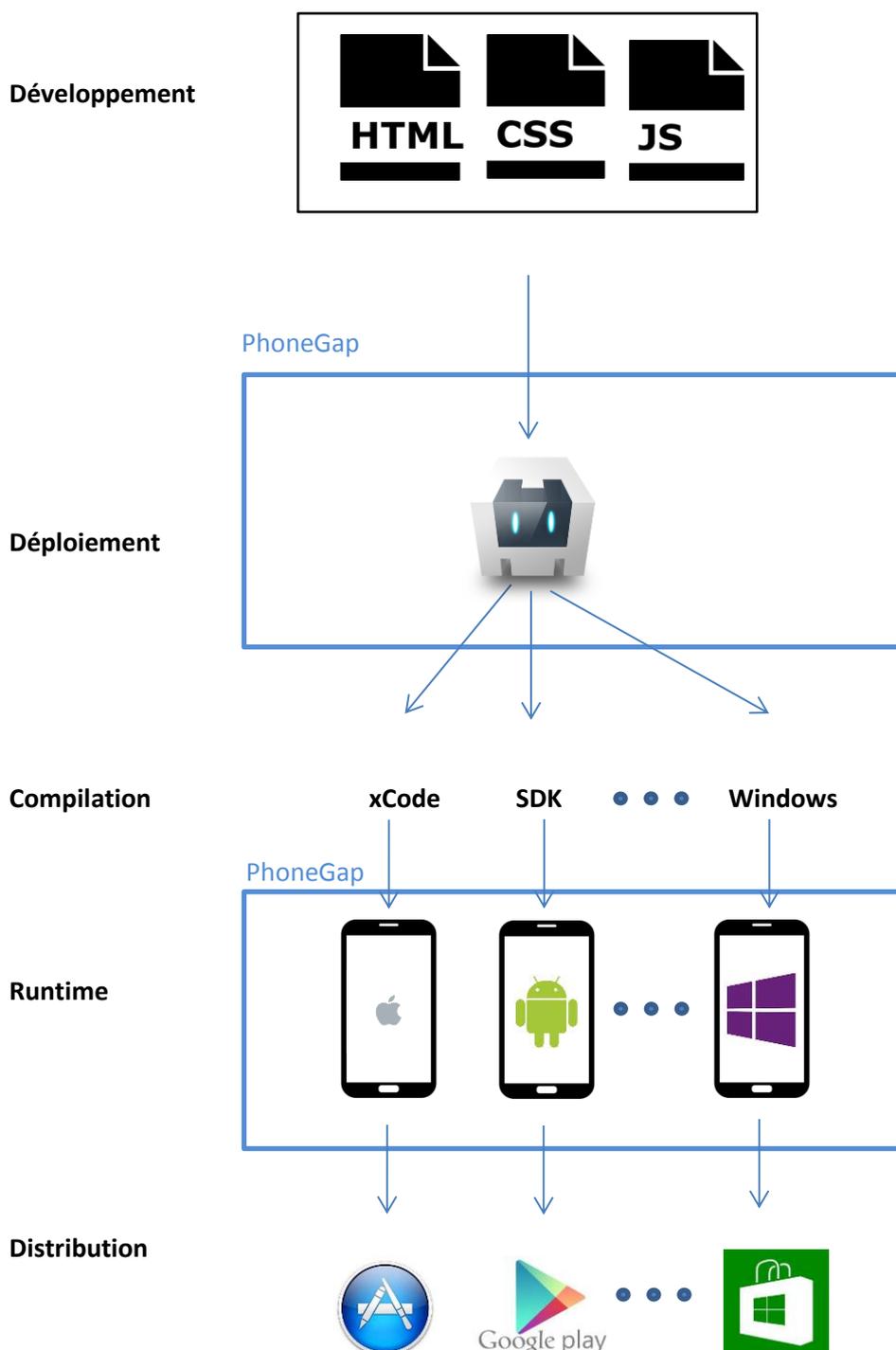
Depuis l'ascension des smartphones, tablettes et le marché des applications quel engendre, la concurrence n'a cessé de croître. Les entreprises ont donc essayé de réduire les coûts de développement et de maintenance des applications. L'objectif est donc de proposer une solution permettant de réduire le temps et les ressources à la réalisation d'une application. C'est donc dans ce contexte là et pour répondre à ces besoins que les solutions PhoneGap et Xamarin ont été créés.

Comment ça fonctionne ? Les models

Pour réaliser une application mobile utilisant des technologies de cross plateforme, il existe cinq étapes avant que l'application puisse être utilisée par l'utilisateur final. La première est le développement, vient ensuite le déploiement sur chaque plateforme, puis la compilation, l'exécution et pour finir la distribution. Chaque technologie n'intervient pas forcément dans toutes ces étapes, nous verrons cela par la suite.

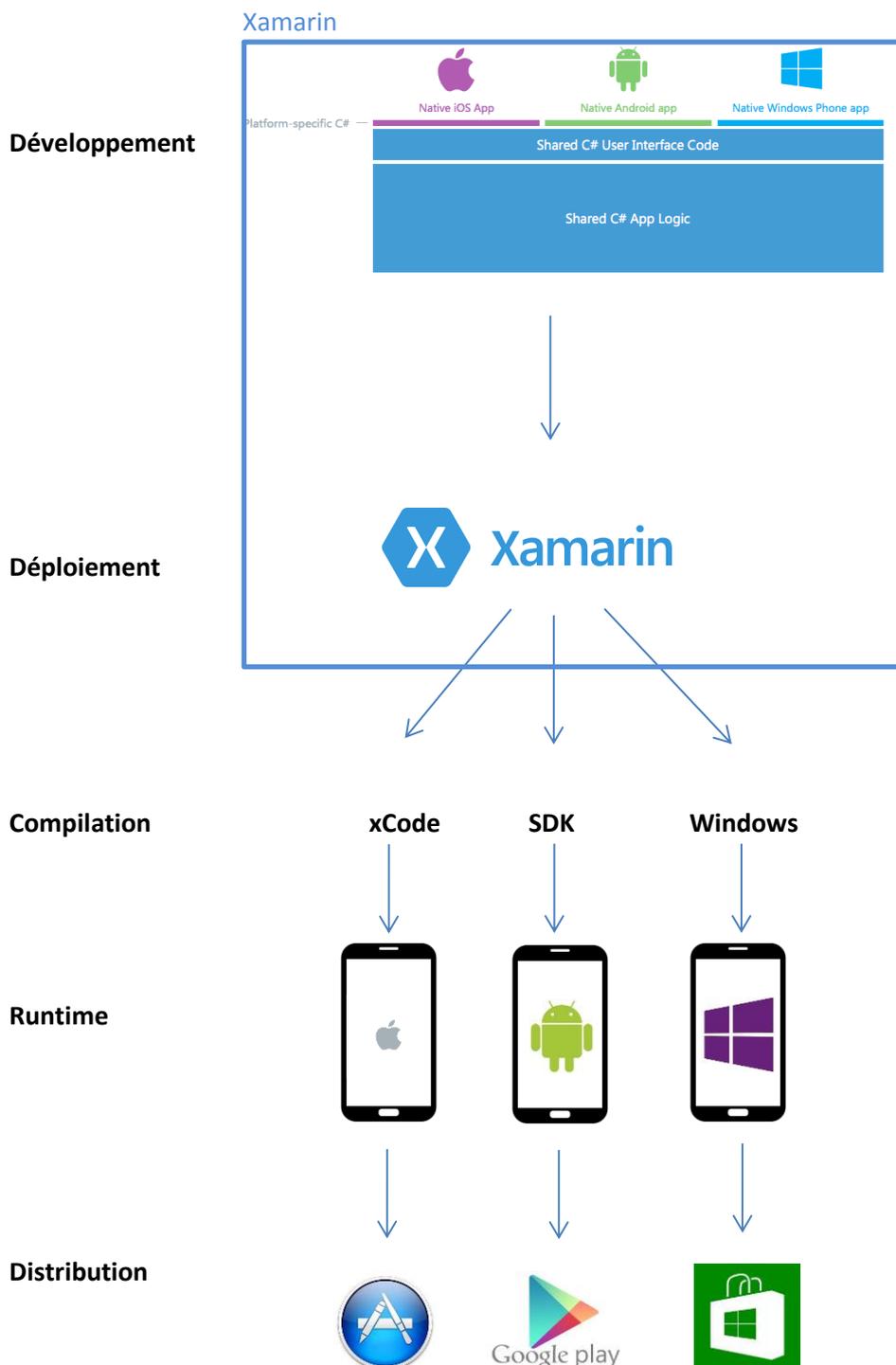
Phonegap

Phonegap est un framework utilisant les langages HTML5, Java Script et css. L'Api javascript fourni avec PhoneGap permet d'accéder aux fonctions et capteurs de l'appareil cible. Une fois le code web réalisé, l'une des différences importantes avec Xamarin est que le code n'est pas compilé. En effet, PhoneGap va juste se charger de packager le projet dans le but de le déployer sur chaque plateforme. Après que chaque framework natif (xCode, SDK Android, etc ...) est compilé le package et créé les binaires (apk, ipa, etc..), PhoneGap va intervenir ensuite lors de l'exécution, puisque le code web sera interprété par une webview PhoneGap utilisant les navigateurs web des appareils, cette webview intègre les plugins en natif permettant de faire la jonction entre le code web et le hardware. L'application s'exécute ensuite comme le ferait un site web dans une webview.



Xamarin

Xamarin possède une approche différente, contrairement à PhoneGap cette solution intègre tous les outils de développements au sein de son IDE Xamarin Studio. Dans la phase de développement, Xamarin propose de centraliser la plus grande partie de code, tel que le métier, les connexions réseau, la base de données et certaines parties d'IHM, puis Xamarin laisse la possibilité de définir des IHM spécifiques à chaque plateforme. Lors du déploiement elle génère un projet de la plateforme souhaité intégrant une librairie contenant les composants. À l'exécution, l'application agit comme le ferait une application native.



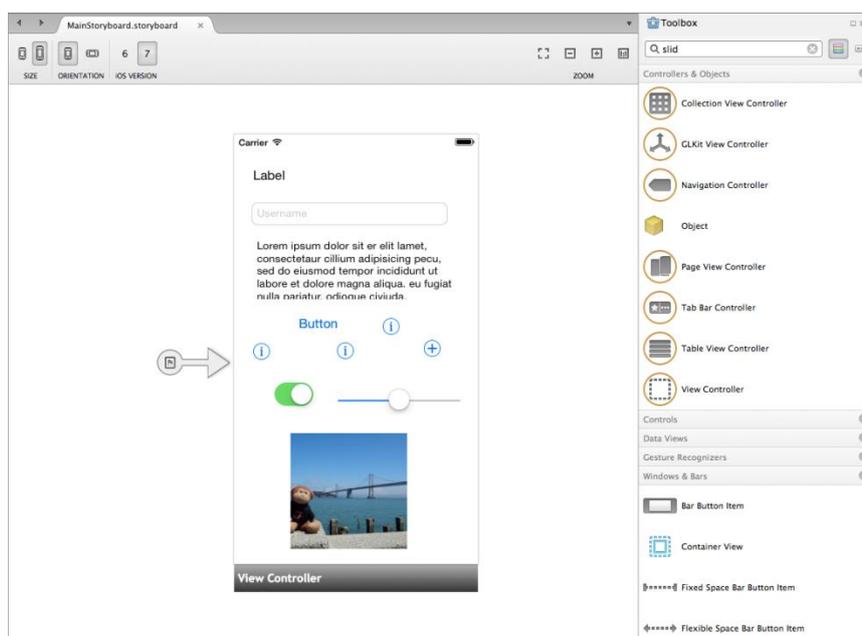
La suite de ce document portera sur un comparatif des trois étapes dans lesquelles Xamarin et PhoneGap interviennent, il s'agit du développement, du déploiement et pour finir de l'exécution de l'application.

Comparatif

Développement

Les environnements de développement

PhoneGap n'intervient pas lors du développement, lorsque l'on réalise l'application il est possible d'utiliser l'IDE de son choix, tels qu'éclipse ou IntelliJ . Xamarin intègre son propre IDE dédié au développement cross plateforme. Il fournit ainsi des outils dédiés à l'univers mobile, comme une interface intuitive pour construire ses IHM basé sur du "drag and drop" de widget. Voulant respecter l'univers du natif, l'interface est différente selon la plateforme visée, par exemple pour iOS l'interface ressemblera à celle d'Interface Builder d'Appel .



Xamarin : Interface pour construire son IHM iOS

Certains développeurs préféreront avoir la liberté de choisir l'IDE qui leur convient, d'autre comme moi préférons avoir un outil conçu pour réaliser une seule tâche et la faire bien. Donc pour ce premier point comparatif, un point pour Xamarin.

Xamarin	PhoneGap
1	0

La communauté, les librairies et framework additionnels

Xamarin est une technologie récente qui possède un cycle de développement contrôlé, et de ce fait la communauté et les librairies, qui permettent de réduire les temps de développements, sont à ce jour encore à leur début et fortement dépendant de Xamarin. Pour exemple, pour obtenir des librairies comme des composants d'IHM de la communauté, il est nécessaire de se connecter au store de Xamarin à l'adresse suivante : <http://components.xamarin.com/>, une fois le composant choisi il s'intégrera à Xamarin Studio.

PhoneGap venant de la communauté open source du web possède un avantage conséquent sur son rival. En effet, les contraintes liées à la résolution et à la taille d'écran sont des contraintes abordées depuis plusieurs années dans le web. Il existe donc de nombreuses librairies permettant de faciliter le développement comme jQuery Mobile, Kendo UI ou Onsen UI. De plus, étant basé sur du HTML et Javascript, une simple recherche Google suffit à trouver un composant IHM répondant à ses besoins.

Sur ce point-ci, net avantage à PhoneGap.

Xamarin	PhoneGap
1	1

Le coût

Lorsque l'on souhaite réaliser une application en cross plateforme la raison principale est de pouvoir réduire les coûts de développement. PhoneGap propose un seul développement pour chaque plateforme. Xamarin propose de centraliser la plus grande partie de code, tel que le métier, les connexions réseaux, la base de données et certaines parties d'IHM et ensuite de définir des IHM spécifiques à chaque plateforme. Ce souci de contextualisation de l'application demande un temps de développements plus conséquent et de maîtriser les paradigmes propres à chaque OS, cela demande donc une personne fortement qualifiée et donc couteuse.

Point pour PhoneGap.

Xamarin	PhoneGap
1	2

Déploiement

Plateformes

PhoneGap est du développement web, de ce fait une plateforme contenant un navigateur est candidat à recevoir une application PhoneGap, à condition néanmoins que la webview PhoneGap est été réalisé pour la plateforme. À ce jour, il est possible de déployer sur Android, iOS, Blackberry, Windows Phone, Bada.

Xamarin demande un support plus conséquent, chaque composant Xamarin Studio générant un composant natif. De ce fait, les plateformes supportées sont plus réduites, il s'agit des trois principales, iOS, Android et Windows Phone.

De mon point de vue, lorsqu'on réalise du développement mobile, les trois principales plateformes sont suffisantes pour toucher l'ensemble du public visé. Néanmoins, PhoneGap proposant plus de plateformes remporte la manche.

Xamarin	PhoneGap
1	3

Runtime

Adaptabilité aux plateformes et responsive design.

PhoneGap, via les bibliothèques existantes, propose des solutions responsive design clef en main. L'un des principaux défauts que l'on reproche à PhoneGap, à raison, est qu'il n'est pas possible de prendre en compte le contexte de la plateforme d'utilisation. Xamarin, via son cycle de développement contrôlé permet de réaliser une interface spécifique aux OS, bien sûr les applications générées sont responsive design. Point pour Xamarin

Xamarin	PhoneGap
2	3

Performance

Xamarin déploie des composants natifs, de ce fait les applications sont optimisées pour chaque plateforme. PhoneGap interprète le code HTML 5 et Javascript dans une Webview, les performances dépendent donc fortement des webkits intégrés dans chaque OS, et ainsi dépendent des performances du smartphone.

À ce jour le natif reste plus performant que l'interprétation d'une page web, donc avantage à Xamarin.

Xamarin	PhoneGap
3	3

Expérience utilisateur

Nous allons aborder maintenant le dernier point, qui est pour moi le plus important, l'expérience utilisateur. Les applications qui sont massivement adaptées par le grand public ont plusieurs points

en commun, notamment la facilité d'utilisation, une interface fluide, des codes de navigation que l'utilisateur reconnaît. Et sur ces points, PhoneGap arrive à les atteindre sur une plateforme, mais des compromis seront fait sur les autres, Xamarin via les deux points vus précédemment permet de passer outre ces compromis et de proposer une solution spécifique à l'utilisateur et à son environnement d'utilisation. Xamarin gagne ce dernier round.

Xamarin	PhoneGap
4	3

Conclusion

Xamarin	PhoneGap
Environnement de développement	Communauté, librairies, framework
Adaptabilité aux OS et responsive design.	Coûts
Performance	Plateformes supportées
Expérience utilisateur	
4	3

Sur les sept critères abordés, Xamarin l'emporte, néanmoins le choix final se fera selon une autre composante, il s'agit du background de l'ingénieur ou développeur qui va utiliser la technologie. Une personne venant de l'univers du développement web choisira certainement Phonegap, il sera plus à l'aise avec les langages de développement comme HTML5 et Javascript à contrario, un développeur mobile natif ou un développeur Microsoft choisira Xamarin.