

Cours n°1 - Introduction

Partie 0

Les informations concernant ce cours vous ont été transmises dans le premier mail envoyé mais au cas où, vous retrouverez la répartition du module ainsi que mon adresse e-mail si vous souhaitez me contacter dans les premières diapositives de ce cours.

Partie 1 - Préambule

Deux acteurs principaux règnent sur le marché du mobile : Google avec le système d'exploitation Android et Apple avec iOS. A eux deux, ils représentent 98% du marché mobile avec un large avantage pour Google qui en possède quasiment les trois quarts.

Les smartphones ont émergé en 2007 avec la sortie du premier iPhone et ce domaine n'a cessé de connaître des évolutions tant au niveau matériel que logiciel. On a vu progressivement apparaître une multiplication des objets connectés au sens large avec les smartphones à l'avant-garde de cette tendance.

Google et Apple ont également au fil des années diversifiés leur offre d'objets connectés en proposant des tablettes, des montres connectées ainsi que des boîtiers TV.

Par effet boule de neige, les mobiles étant de plus en plus présents, le marché du développement lié à ce domaine a explosé et le nombre d'applications connaît une croissance constante (c.f. diapositive 7 et 8 ; le creux au niveau de l'année 2018 représente un nettoyage effectué sur les stores par Apple et Google d'applications inactives ou non conformes à leurs standards).

On note cependant une différence au niveau des utilisateurs de chaque écosystème ; comme on peut le constater au niveau de la diapositive 9, les revenus générés par les applications iOS sont légèrement supérieurs à ceux générés par Android, alors que seulement un quart des téléphones utilisés sont des terminaux Apple.

Partie 2 - Historique

A la fin des années 1980, Steve Jobs lance une société appelée « NeXTStep » visant à fournir des solutions logicielles et notamment un système d'exploitation du même nom basé sur un langage, créé par cette même société, appelé « Objective-C ».

En 1997, Apple rachète « NeXTStep » et utilisera ses travaux pour produire le système d'exploitation Mac OS X qui servira lui-même de base au système d'exploitation des premiers iPhones baptisé dans un premier temps « iPhone OS » puis « iOS ».

Le premier iPhone sortira en 2007 et sera le pionnier dans le domaine du smartphone bien qu'à l'époque Google travaillait déjà sur Android. Par la suite, Apple proposera des tablettes et autres objets connectés basés sur iOS bien qu'en 2019 il sera décidé de développer un autre système d'exploitation pour les tablettes de la marque baptisé « iPad OS ». (c.f. diapositives 10 et 11).

Du côté d'Android, le système est tout d'abord développé par une « startup » du même nom qui sera rachetée par Google par la suite, ce dernier investissant au même moment qu'Apple pour essayer d'être à l'avant-garde dans le domaine du smartphone mais sera dépassé de peu.

On notera également qu'Apple a eu l'avantage d'avoir la main sur le matériel et le logiciel dès le départ, chose que Google n'aura qu'au moment de la sortie des Pixels en 2016. (c.f. diapositives 12 et 13).

Partie 3 - Développement mobile

Il existe deux grandes façons de développer des applications mobiles : le « cross-plateformes » et le natif. Les avantages de l'un étant souvent les inconvénients de l'autre et vice-versa.

L'avantage du développement « cross-plateformes » est que comme son nom l'indique, il permet de développer qu'une seule fois une application pour les deux plateformes mais produit des applications moins performantes. Les outils proposant du développement cross-plateformes ne fournissent pas toujours les fonctionnalités liées à des aspects bas niveaux ou spécifiques à un système ; il faudra faire appel à du développement natif dans ce cas. Il existe de nombreux outils dans le domaine, chacun avec ses forces et faiblesses. (C.f. diapositives 14 et 15).

L'avantage du développement natif est qu'il permet de produire des applications plus performantes, mieux intégrées au système d'un point de vue visuel ou fonctionnel et est parfois incontournable comme évoqué précédemment. Cependant, il est parfois plus fastidieux qu'une solution cross-plateformes et plus coûteux en temps si plusieurs plateformes doivent être supportées. (C.f. diapositives 16 et 17).

Le développement natif permet aussi cependant de faire appel à des bibliothèques écrites dans des langages différents à partir du moment où ces langages sont compatibles avec l'environnement dans lequel s'exécute notre code natif.

Le développement Android se faisant en principe en Java, tous les langages produisant du *bytecode* pour la machine virtuelle Java (JVM) peuvent être greffés à une application Android (comme par exemple Kotlin, devenu maintenant le langage officiel pour le développement Android ou encore Scala ou Groovy).

Le développement iOS se fait en Swift et ce dernier est transformé en une représentation intermédiaire (IR) par le biais d'un outil appelé LLVM ; des compilateurs pour les langages Objective-C et C++ produisent une telle représentation intermédiaire. Il est donc possible de faire appel à des bibliothèques écrites dans ces langages depuis une application iOS. (C.f. diapositives 18 à 21)

On notera qu'il est également possible de faire appel à du code C++ dans une application Android mais pas par le même biais.

Pour ce cours, il a été choisi de se concentrer sur le développement natif iOS.

Le développement d'applications iOS se fait en Swift. Anciennement, le langage officiel était l'Objective-C mais celui-ci tend à être mis complètement de côté bien qu'il soit encore possible de trouver certaines ressources ou certains projets l'utilisant.

Apple fournit également un SDK (Software Development Kit) avec différents composants facilitant chacun des développements spécifiques. On utilisera par exemple UIKit pour la mise en place d'interfaces ou AVFoundation pour ce qui concerne l'audio-visuel (c.f. diapositive 23 et 24).