

Chapitre 2. Le système mobile Android

Table des matières

2. Chapitre 2. Le système mobile Android	3
2.1. Introduction	3
2.2. Fonctionnalités d'Android	3
2.3. Architecture d'Android	4
2.4. L'environnement de développement Android Studio	5
2.4.1. Présentation	5
2.4.2. Les prérequis d'un système pour installer Android Studio	6
2.4.3. Configuration d'Android Studio et création de projets	7
2.5. Conclusion	9
Bibliographie et Webographie	10

2. Chapitre 2. Le système mobile Android

2.1. Introduction

Android est un système d'exploitation mobile basé sur une version modifiée de Linux. C'était à l'origine développé par une startup du même nom, Android, Inc. En 2005, Google a acheté Android, Inc. et a repris son travail de développement dans le cadre de sa stratégie d'entrée sur le l'espace mobile. Google voulait que le système d'exploitation Android soit ouvert et gratuit, donc la plupart du code Android a été publié sous la licence open source Apache. Le principal avantage de l'adoption d'Android est qu'il offre une approche unifiée du développement d'applications. Les développeurs n'ont besoin de développer que pour Android en général, et leurs applications doivent pouvoir fonctionner sur de nombreux appareils supportées par Android.

2.2. Fonctionnalités d'Android

Comme Android est open source et disponible gratuitement pour les fabricants pour la personnalisation, il n'y a pas de configurations matérielles ou logicielles fixes. Cependant, le système d'exploitation Android de base prend en charge de nombreuses fonctionnalités, qui sont :

- Stockage : SQLite, une base de données relationnelle légère pour le stockage des données.
- Connectivité : GSM/EDGE, IDEN, CDMA, EV-DO, UMTS, Bluetooth (comprend A2DP et AVRCP), Wi-Fi, LTE et WiMAX.
- Messagerie : SMS et MMS.
- Prise en charge des médias H.263, H.264 (en conteneur 3GP ou MP4), MPEG-4 SP, AMR, AMR-WB (en conteneur 3GP), AAC, HE-AAC (en conteneur MP4 ou 3GP), MP3, MIDI, Ogg Vorbis, WAV, JPEG, PNG, GIF et BMP [11].
- Prise en charge matérielle : capteur accéléromètre, appareil photo, boussole numérique, capteur de proximité, et GPS.
- Écrans multi-touch.
- Applications multitâches.

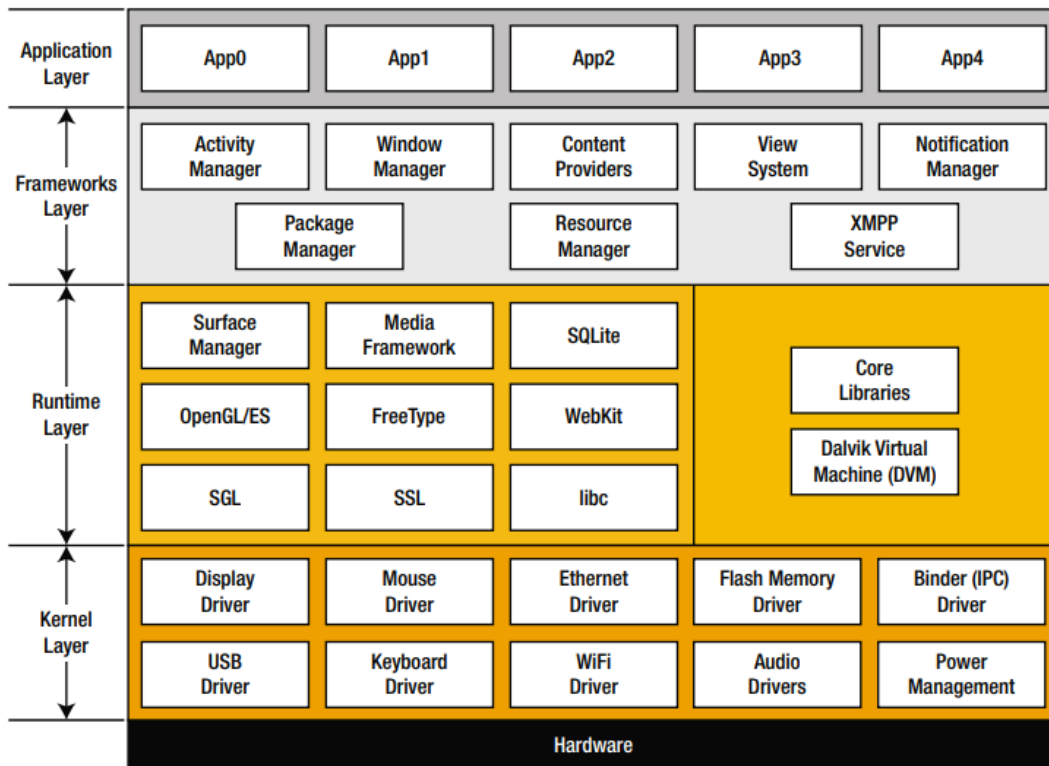
- Partage de connexion : Partage des connexions Internet en tant que point d'accès filaire/sans fil.

2.3. Architecture d'Android

La Figure 2.1 montre les différentes couches qui composent le système d'exploitation (OS) Android. Le système d'exploitation Android est divisé en cinq sections réparties sur quatre couches principales :

- Noyau Linux : Il s'agit du noyau sur lequel Android est basé. Cette couche contient tous les pilotes de périphérique de bas niveau pour les différents composants matériels d'un périphérique Android.
- Bibliothèques : Celles-ci contiennent le code qui fournit les principales fonctionnalités d'un système d'exploitation Android. Par exemple, la bibliothèque SQLite fournit un support de base de données afin qu'une application puisse l'utiliser pour le stockage de données. La bibliothèque WebKit [12] fournit des fonctionnalités pour la navigation Web.
- L'environnement d'exécution (runtime) Android : l'environnement d'exécution Android est située dans la même couche que les bibliothèques et fournit un ensemble d'API principales qui permettent d'écrire des applications Android à l'aide du langage de programmation Java. Le runtime Android comprend également la machine virtuelle Dalvik, qui permet à chaque application Android de s'exécuter dans son propre processus avec sa propre instance de la machine virtuelle Dalvik [13]. Dalvik est une machine virtuelle spécialisée conçue spécifiquement pour Android et optimisée pour les appareils mobiles alimentés par batterie avec une mémoire et une puissance CPU limitées.
- Framework d'application : le framework d'application expose les différentes fonctionnalités du système d'exploitation Android aux développeurs d'applications afin qu'ils puissent les utiliser.
- Applications : cette couche supérieure contient les applications livrées avec l'appareil Android (telles que le téléphone, les contacts, le navigateur, etc.), ainsi que les applications téléchargées et installées à partir d'Android PlayStore.

La figure 2.1 illustre l'architecture en couche du système Android :



C, C++, Native Code	Java
= Linux Kernel	= Android Frameworks
= Libraries	= Applications
= Android Runtime	

Figure 2.1. architecture d'Android [14]

2.4. L'environnement de développement Android Studio

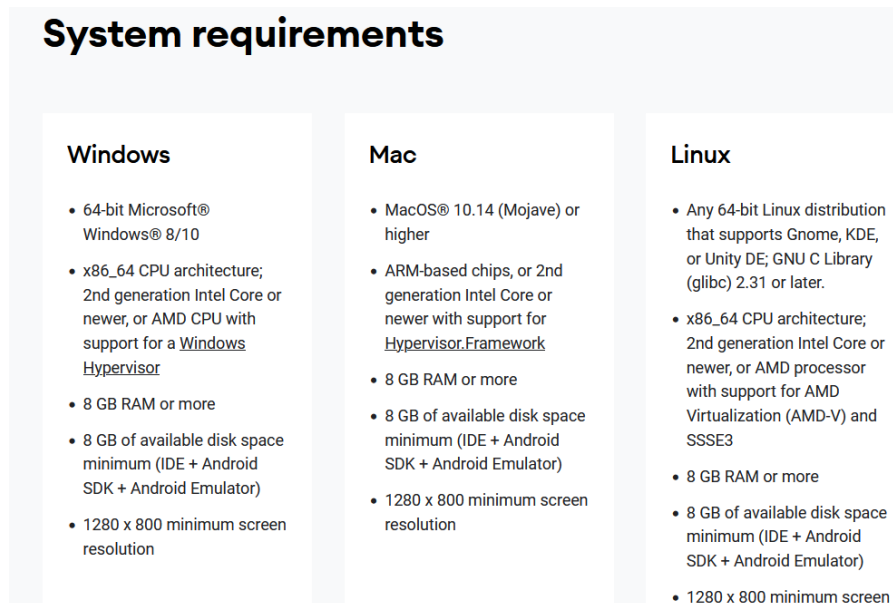
2.4.1. Présentation

Android Studio est l'environnement de développement intégré (IDE) officiel pour le système d'exploitation Android [15]. Il est basé sur le logiciel IntelliJ IDEA de JetBrains et conçu spécifiquement pour le développement Android [16]. Il est disponible en téléchargement sur les systèmes d'exploitation Windows, macOS et Linux ou sous forme de service par abonnement en 2020. Il remplace les outils de développement Android Eclipse (E-ADT) en tant qu'IDE principal pour le développement d'applications Android natives.

Android Studio a été annoncé le 16 mai 2013 lors de la conférence Google I/O. Il était en phase de prévisualisation d'accès anticipé à partir de la version 0.1 en mai 2013, puis est entré en phase bêta à partir de la version 0.8 qui a été publiée en juin 2014. La première version stable a été publiée en décembre 2014, à partir de la version 1.0.

2.4.2. Les prérequis d'un système pour installer Android Studio

La dernière version d'Android peut être téléchargée du site officiel « <https://developer.android.com/studio> ». Afin de vérifier les prérequis que doit avoir la machine du développeur, il existe une partie tout en bas de la page du site officiel d'Android qui montre tous ce qui est nécessaire pour l'installation et l'utilisation de Android. La figure 2.2. montre ces prérequis.



System requirements

Windows	Mac	Linux
<ul style="list-style-type: none">• 64-bit Microsoft® Windows® 8/10• x86_64 CPU architecture; 2nd generation Intel Core or newer, or AMD CPU with support for a Windows Hypervisor• 8 GB RAM or more• 8 GB of available disk space minimum (IDE + Android SDK + Android Emulator)• 1280 x 800 minimum screen resolution	<ul style="list-style-type: none">• MacOS® 10.14 (Mojave) or higher• ARM-based chips, or 2nd generation Intel Core or newer with support for Hypervisor.Framework• 8 GB RAM or more• 8 GB of available disk space minimum (IDE + Android SDK + Android Emulator)• 1280 x 800 minimum screen resolution	<ul style="list-style-type: none">• Any 64-bit Linux distribution that supports GNOME, KDE, or Unity DE; GNU C Library (glibc) 2.31 or later.• x86_64 CPU architecture; 2nd generation Intel Core or newer, or AMD processor with support for AMD Virtualization (AMD-V) and SSSE3• 8 GB RAM or more• 8 GB of available disk space minimum (IDE + Android SDK + Android Emulator)• 1280 x 800 minimum screen

Figure 2.2. les prérequis du système.

2.4.3. Configuration d'Android Studio et création de projets

Une fois Android installé, on peut le lancer et démarrer la création d'un nouveau projet comme montré par la figure 2.3.

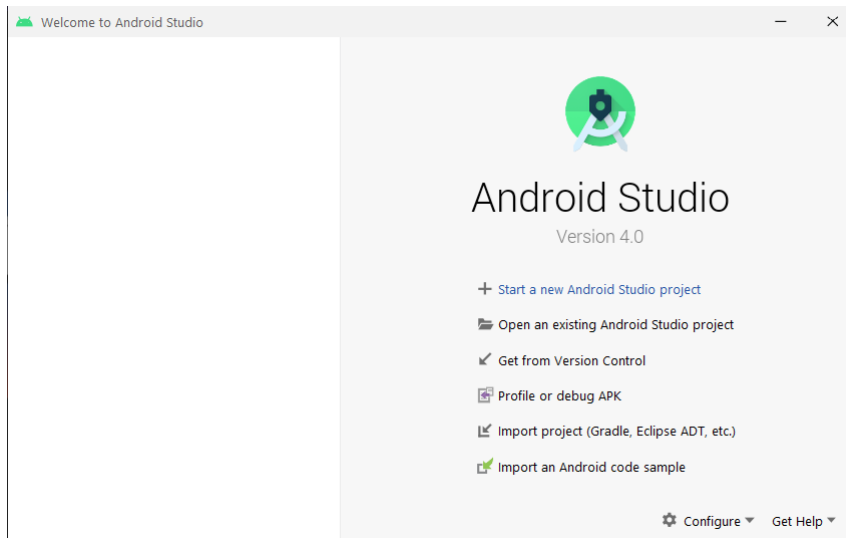


Figure 2.3. premier lancement de Android Studio.

Pour la configuration de l'application en cours de création sur Android studio, on peut accéder à la configuration avancée comme montré par la figure 2.4.

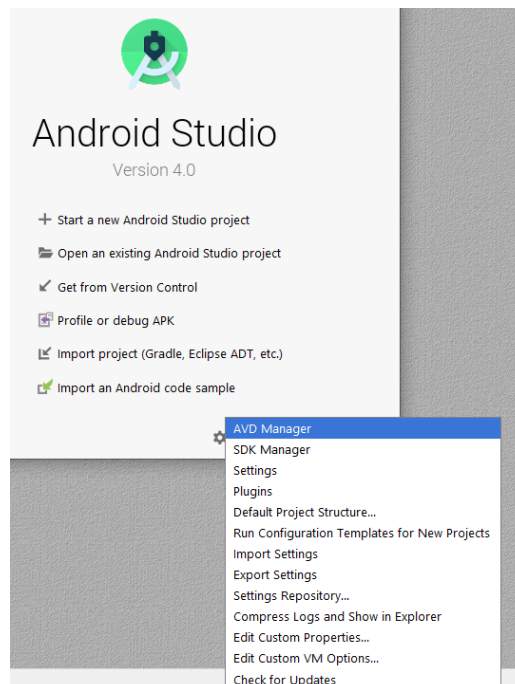


Figure 2.4. Accès à la configuration avancée de l'application en cours de création.

Cette configuration avancée concerne en particulier le *Software Development Kit* (SDK) et le *Android Virtual Device* (AVD).

Après la configuration des paramètres de l'environnement d'Android Studio pour le projet en cours de création, il nous propose de choisir un modèle (Template) pour préparer à l'avance des outils spécifiques à la nature d'application qu'on veut créer. La figure 2.5. montre la proposition des modèles disponibles.

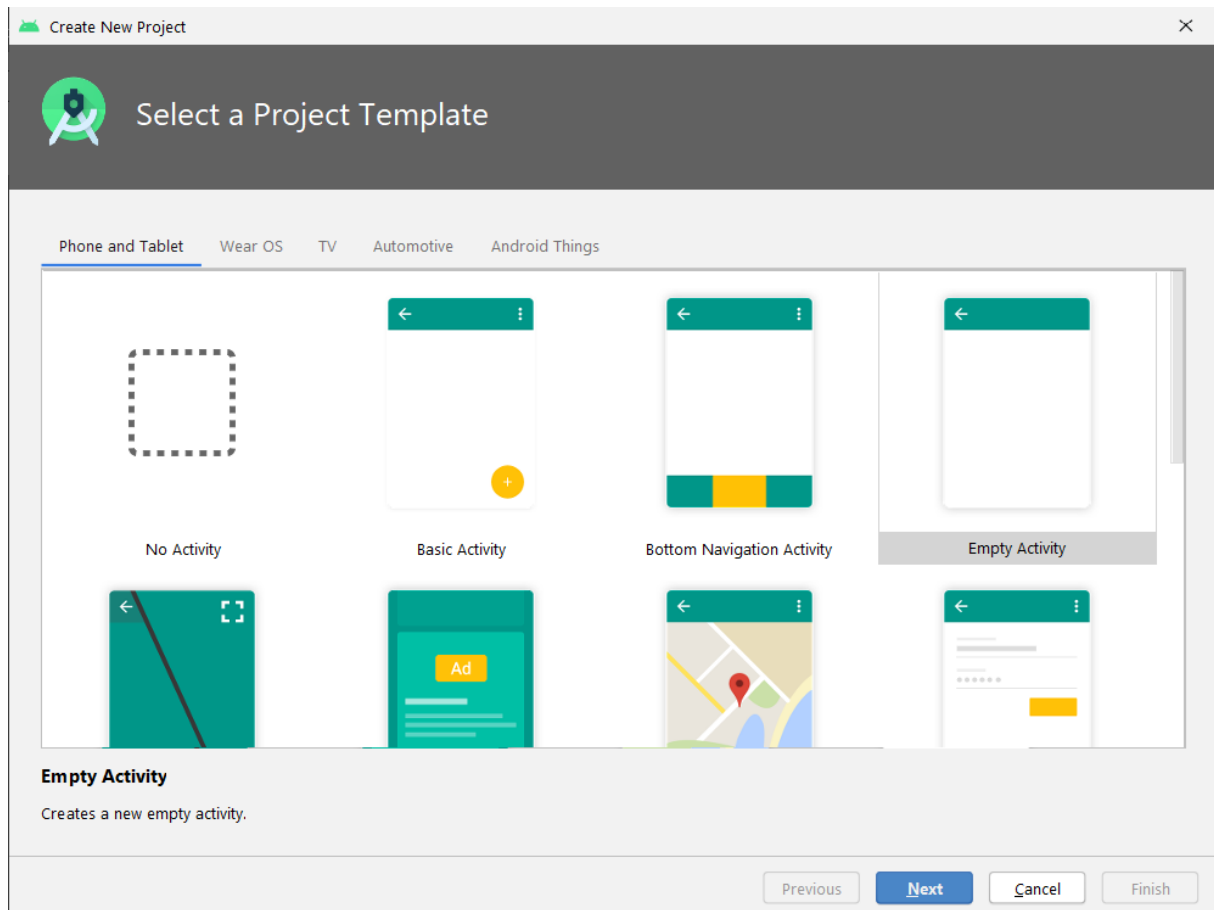


Figure 2.5. les modèles d'application disponibles.

Le reste de la configuration est lié à l'application à créer, son nom, le nom de son package, l'endroit de sa sauvegarde sur le disque dur, le langage de programmation que ce soit Java ou Kotlin [17], et à la fin la version minimum de l'OS Android supportée. La figure 2.6. montre les champs à spécifier pour la configuration de l'application à créer.

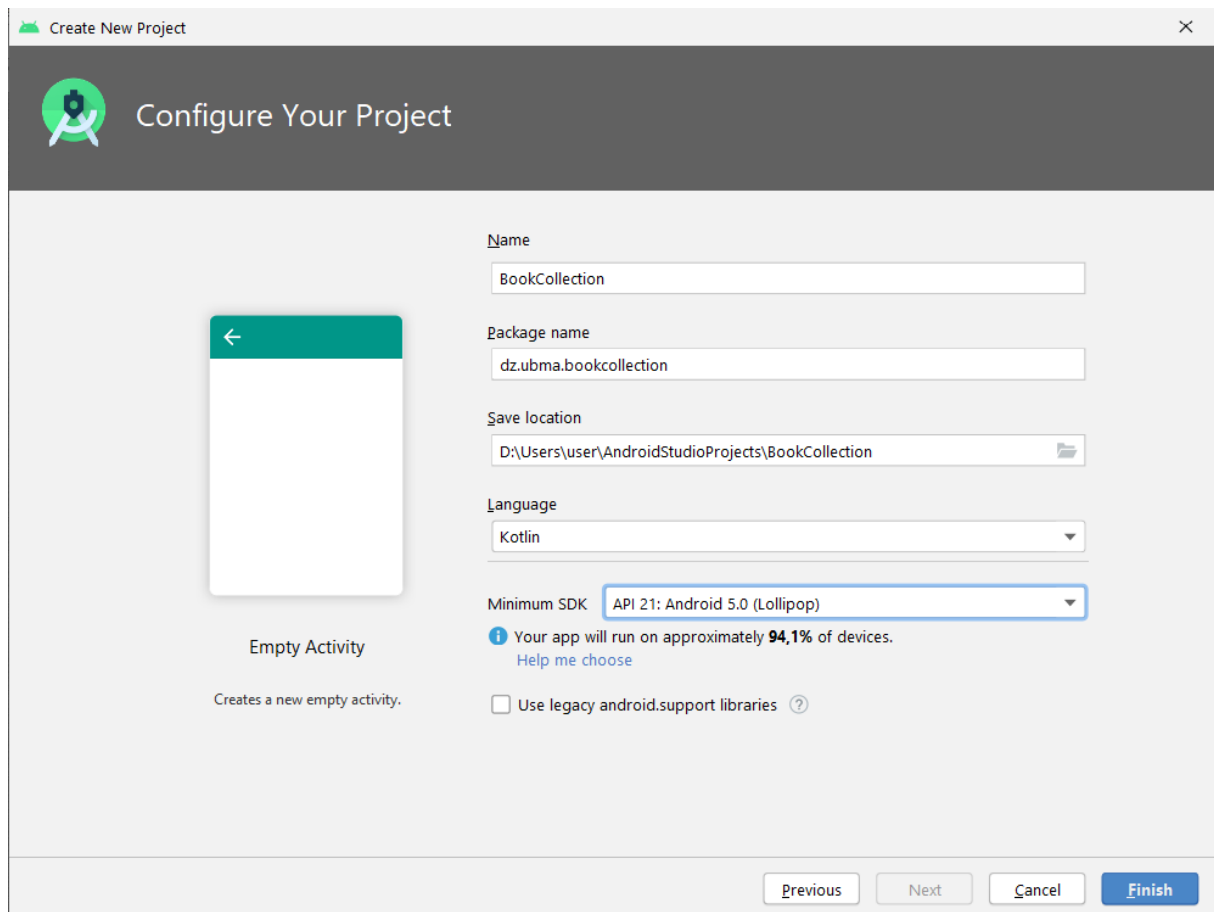


Figure 2.6. Configuration de l'application Android à créer.

Après avoir installé Android Studio, On peut utiliser le gestionnaire de SDK pour télécharger et installer plusieurs versions du SDK Android. La disponibilité de plusieurs versions du SDK permet d'écrire des programmes qui ciblent différents appareils. Par exemple, on peut créer une version d'une application qui cible spécifiquement Android Pie, mais étant donné que cette version d'Android est présente sur moins de 39.5 % des appareils, on peut sélectionner la version Android Lollipop ou KitKat qui sont supportés par plus de 94% des appareils fonctionnant avec Android.

2.5. Conclusion

Dans ce deuxième chapitre, on a présenté le système Android, son architecture, et l'outil officiel pour le développement de ses applications natives. A la fin de ce chapitre composé du cours et de TP (Voir l'Annexe), l'étudiant est capable de concevoir les interfaces de son application avec WireFrameSketcher, installer et configurer Android studio notamment le SDK et le AVD, créer un nouveau projet et de comprendre sa structuration.

Bibliographie et Webographie

- [11] « Freescale codec release note in Android ». https://www.nxp.com/docs/en/supporting-information/i_MX_Android_Codec_Release_Notes.html (consulté le 11 février 2022).
- [12] « WebKit ». <http://www.standard-du-web.com/webkit.php> (consulté le 12 février 2022).
- [13] H.-S. Oh, B.-J. Kim, H.-K. Choi, et S.-M. Moon, « Evaluation of Android Dalvik virtual machine », p. 10.
- [14] S. Gunasekera, « Android Architecture », in *Android Apps Security*, S. Gunasekera, Éd. Berkeley, CA: Apress, 2012, p. 1-12. doi: 10.1007/978-1-4302-4063-1_1.
- [15] « Le studio Android est-il conçu par JetBrains ? », *OS Today*, 5 avril 2021. <https://frameboxxindore.com/fr/non-class%C3%A9/est-le-studio-android-fait-par-jetbrains.html> (consulté le 12 février 2022).
- [16] « Android Studio: An IDE built for Android », *Android Developers Blog*. <https://android-developers.googleblog.com/2013/05/android-studio-ide-built-for-android.html> (consulté le 12 février 2022).
- [17] « Kotlin Programming Language ». <https://kotlinlang.org/> (consulté le 12 février 2022).