



VIII – Réalité Virtuelle avec Processing sur Android

125

VIII.2 – Processing en mode Android

<https://android.processing.org/>

Processing for Android



[Home](#)

[Install](#)

[Reference](#)

[Tutorials](#)

[Books](#)

Use Processing for Android to create Android apps with ease, including live wallpapers, watch faces, and VR apps. You can run your Processing sketches on Android devices (phones, tablets, watches) with little or no changes in the code, and without worrying about installing SDK files or editing layout files. Processing for Android also lets you accessing the Android API to read sensor data, and exporting your sketch as a signed package ready to upload to the Google Play Store.

Follow [@p5android](#) and [@ProcessingOrg](#) on Twitter, and also [Processing Foundation](#) Medium's posts, for news and updates!

126

VIII.2 – Processing en mode Android sur PC

Ajout du mode « android »

autorise Processing à éditer du code Android (java)

Ajout du compilateur « android »

utilisation d'un SDK Android pour que Processing puisse compiler et exécuter du code Android

Contribution Manager

LibrariesModesToolsExamplesUpdates

Filter

StatusNameAuthor

Android Mode | Create projects with Processing for Android dev... | The Processing Foundation

Android

Java

Android

Python

Ajouter un mode...

Cannot find an Android SDK...

Processing did not find an Android SDK on this computer. If there is one, and you know where it is, click "Locate SDK path" to select it, or "Download SDK" to let Processing download the SDK automatically.

If you want to download the SDK manually, you can get the command line tools from [here](#). Make sure to install the SDK platform for API 26.

Download SDK automaticallyLocate SDK path manually

127

VIII.2 – Processing en mode Android sur PC

Installation SDK Android

Solution 1 – par Processing :

« Download SDK automatically »

Solution 2 – par installation complète d'Android Studio IDE

Puis « SDK manager » d'Android Studio :
installation API 26 – Android 8.0 (Oreo)

« Locate SDK path manually » :
C:\Users\XXX\AppData\Local\Android\Sdk

Puis installation émulateur « Processing-phone » :
A mettre dans Documents >Processing > android > avd

Vérification sous « AVD manager » d'Android Studio :

SDK download

Downloading Android SDK...

156,1 MB / 563,0 MB

27 %

Cancel download

Android Studio

Version 3.5.3

Android Virtual Device Manager

Your Virtual Devices

Android Studio

Type	Name	Play Store	Resolution	API	Target
	processing phone		480 x 800: hdpi	26	Android 8.0 (Go...)

android.emulator.image.abi=x86
android.emulator.port.phone=5554
android.gradle=1.1
android.sdk.appcompat=1.2.0
android.sdk.asm=1.2.0
android.sdk.gvml=1.80.0
android.sdk.path=C:\Users\Emmanuel\AppData\Local\Android\Sdk
android.sdk.play_services=17.0.0
android.sdk.support=26.0.2
android.sdk.target=28
android.sdk.version=1.0.0
android.sdk.vear=2.8.1
chooserc.files.native=true

Android Emulator - processing-phone5554

February 22
MONDAY 2021

9:18

Attention : le numéro doit être identique à celui du fichier « preferences.txt » de Processing !

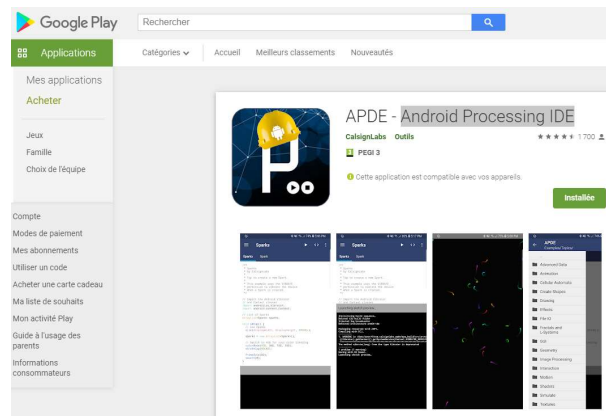
128

Emmanuel MESNARD

2

VIII.2 – Processing en mode Android sur smartphone Android

- Nouveauté : Processing a été porté sur Android
 - Ecriture directe de sketch sur tablette / smartphone !
 - APDE : Android Processing IDE
 - <https://github.com/Calsign/APDE>



129

VIII.2 – Applications sous Android

- Différents types d'application Android
 - App : application « standard »
 - VR : application de Réalité Virtuelle
 - avec prise en compte des capteurs de mouvements
 - et de la déformation des lentilles !
 - AR : application de Réalité Augmentée
 - Google AR Core
 - Wallpaper : application de fond d'écran
 - fonds d'écran dynamiques
 - Watch Face : application de type montre
- Création (optionnelle) d'un APK
 - Package signé :
 - menu *Fichier>Export Signed Package*
 - Attention au nom et aux diverses contraintes imposées pour la « distribution »
 - <https://android.processing.org/tutorials/distributing/index.html>
 - ou bien, sans signer mais fonctionnel quand même :



Building Android project...

Build folder: C:\Users\Emmanuel\AppData\Local\Temp\android8702084865744786144sketch

C:\Users\Emmanuel\AppData\Local\Temp\android8702084865744786144sketch\app\build\outputs\apk\debug

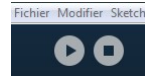
130

VIII.3 – Principe de compilation et d'exécution

■ Vers le smartphone :

- ☐ Smartphone relié à l'ordinateur

- via l'USB,
- avec mode développeur activé



- ☐ Appui sur le bouton **play** :

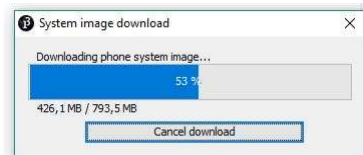
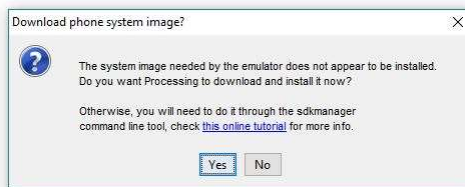
Compilation (lente...) puis installation et exécution sur smartphone
(liste des périphériques autorisés dans menu *Android>Devices*)

■ Vers l'émulateur :

- ☐ Appui sur le bouton **play** avec **Majuscule** enfoncée :

Compilation (lente...) puis exécution sur émulateur

- Rappel : APK disponible dans android...sketch > App > build > outputs > apk > debug
- Attention : demande d'installation de l'image d'un système android pour téléphone lors de la première exécution :



131

VIII.4 – Ecriture d'applications *App* pour smartphone Android

■ Modifications à apporter aux applications « Processing » : rien ! ou presque...

- ☐ Processing en « Mode Android », et application de type « App »

- ☐ **Retrait** des fonctions liées à l'objet « surface »

- `surface.setTitle("Exemple ... // Impossible sur Android`

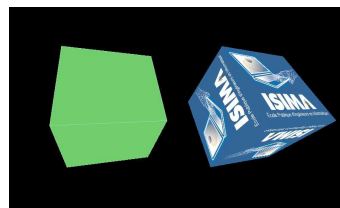
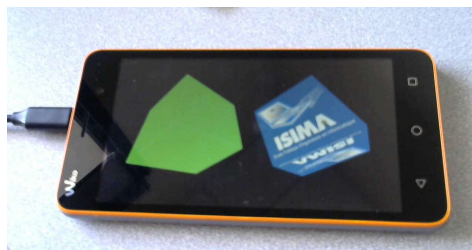
- ☐ Eventuellement, remplacement de `size()` par `fullScreen()`

- valable en 3D également : `fullScreen(P3D) // Ouverture en mode 3D`

- ☐ De ce fait, il est possible de :

- restituer sur l'écran des objets simples (2D ou 3D), tel qu'un cube tournant !
- gérer l'équivalent de la « souris » (`mousePressed`, ...)

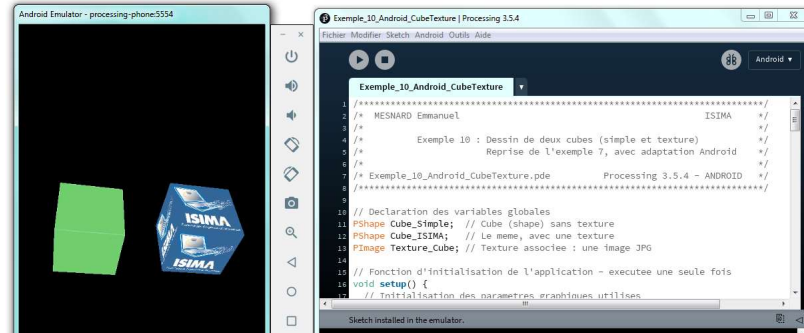
■ Reprise de l'exemple du cube texturé :



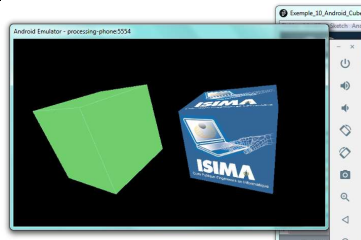
132

VIII.4 – Exemple 12 : application *App* du cube texturé

- Résultat identique sous émulateur :



- Rotation écran possible :



133

TD 9 Application Android « Classique »

- Travail demandé :
 - Faire une application au format « App » qui permet de tester les fonctionnalités d'entrées/sorties basiques de toute application : touchscreen, clavier, souris, ...
- Spécifications de l'application « AndroidPaint » :
 - Ecrire du texte sur l'écran et permettre également la saisie sur clavier (voir `displayDensity` et `openKeyboard`)
 - Gérer le touchscreen pour effectuer le tracé du dessin :
 - dessiner un point au niveau du contact sur l'écran (voir `touchStarted`)
 - éventuellement, prévoir un bouton pour effacer l'écran,
 - un petit sélecteur de couleur pour égayer les tracés, ...
 - Demander la confirmation avant de quitter (voir `backPressed`)
 - Eviter le redémarrage à cause de l'auto-rotate
 - Appel à `orientation()` dans la fonction `setup` (paramètre `LANDSCAPE` ou `PORTRAIT`)

134