

Becoming a Pro

IN Mobile Applications Testing



Mobile APPS: **Distribution/Installation/Logs**



Android
Studio

Powered by IntelliJ Platform

Mobile APPS: **Distribution/Installation: Android .APK**



What is
Android Studio?

Android Studio is the official integrated development environment (IDE) for **Android** platform development.

The official language for Android development is **Java**. Large parts of Android are written in **Java** and its APIs are designed to be called primarily from **Java**.

It is possible to develop C and C++ apps using the Android Native Development Kit (NDK), however it isn't something that Google promotes.



What is ADB in Android Studio

Android Debug Bridge (adb) is a versatile command line tool that lets you communicate with an emulator instance or connected Android-powered device.

It is a client-server program that includes three components:

A **client**, which sends commands. The client runs on your development machine. You can invoke a client from a shell by issuing an adb command. Other Android tools such as DDMS also create adb clients.

A **daemon**, which runs commands on a device. The daemon runs as a background process on each emulator or device instance.

A **server**, which manages communication between the client and the daemon. The server runs as a background process on your development machine.

What is ADB LOCATS?

Logcat is a command-line tool that dumps a log of system messages, including stack traces when the device throws an error and messages that you have written from your app with the Log class

ANDROID MONITOR includes a logcat Monitor that displays debug messages.

The logcat Monitor displays system messages, such as when a garbage collection occurs, as well as messages that you can add to your app using the LOG class.

It displays messages in real time and also keeps a history so you can view older messages.

Mobile APPS: *Distribution/Installation/Logs*

What is ADB LOCATS?

Verbose - Show all log messages (the default).

Debug - Show debug log messages that are useful during development only, as well as the message levels lower in this list.

Info - Show expected log messages for regular usage, as well as the message levels lower in this list.

Warn - Show possible issues that are not yet errors, as well as the message levels lower in this list.

Error - Show issues that have caused errors, as well as the message level lower in this list.

Assert - Show issues that the developer expects should never happen.

**To set a LOG LEVEL : in the log level MENU
Select the Following Options**

Mobile APPS: *Distribution/Installation/Logs*

What is ADB LOGCATS?

HOMEWORK : <http://adbshell.com/commands/adb-logcat>

Some most useful commands



adb logcat *:V *lowest priority, filter to only show Verbose level*

adb logcat *:D *filter to only show Debug level*

adb logcat *:I *filter to only show Info level*

adb logcat *:W *filter to only show Warning level*

adb logcat *:E *filter to only show Error level*

adb logcat *:F *filter to only show Fatal level*

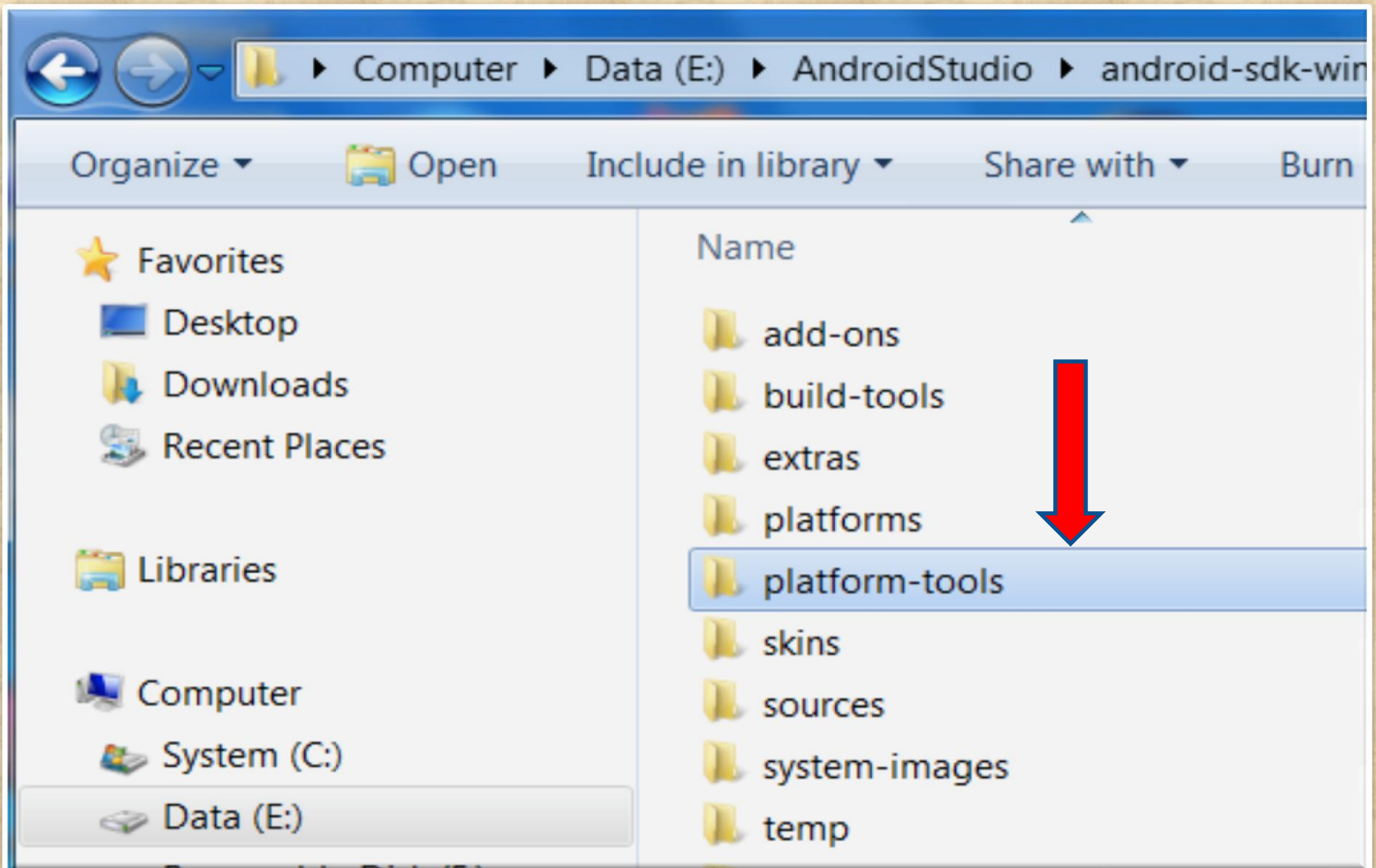
adb logcat *:T *filter to show steps leading up to errors and warnings*

Mobile APPS: **Distribution/Installation/Logs**

Organize ▾ Open Include in library ▾ Share with ▾ Burn New folder

Name	Date modified	Type
android-sdk-windows	11/17/2016 1:58 PM	File folder
android-studio	6/17/2015 5:47 PM	File folder
Projects	6/18/2015 10:16 A...	File folder

Mobile APPS: *Distribution/Installation/Logs*



Mobile APPS: **Distribution/Installation/Logs**

Computer > Data (E:) > AndroidStudio > android-sdk-windows > platform-tools

Organize ▾ Include in library ▾ Share with ▾ Burn New folder

	Name	Date modified
★ Favorites		
Desktop		
Downloads		
Recent Places		
	api	6/22/2016 11:56 A...
	lib64	6/22/2016 11:56 A...
	systrace	6/22/2016 11:56 A...
	adb.exe	6/22/2016 11:56 A...

Mobile APPS: *Distribution/Installation/Logs*

The screenshot shows the Android Studio interface. At the top, a file explorer window is open, displaying the path: `Computer > Data (E:) > AndroidStudio > android-sdk-windows > platform-tools`. Below the file explorer, the main IDE window is visible, showing the menu bar (File, Edit, View, Navigate, Code, Analyze, Refactor, Build, Run, Tools, VCS, Window, Help) and the toolbar. The left sidebar contains the Project, Structure, and Captures views. The main workspace area is divided into several panels: Captures, System Information, Build Variants, and Terminal. The Terminal panel is active, showing the following text:

```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

E:\AndroidStudio\Projects\MyApplication>
```

A red arrow points from the file explorer window to the Terminal panel, indicating the location of the file being referenced in the terminal command.

Mobile APPS: *Distribution/Installation/Logs*

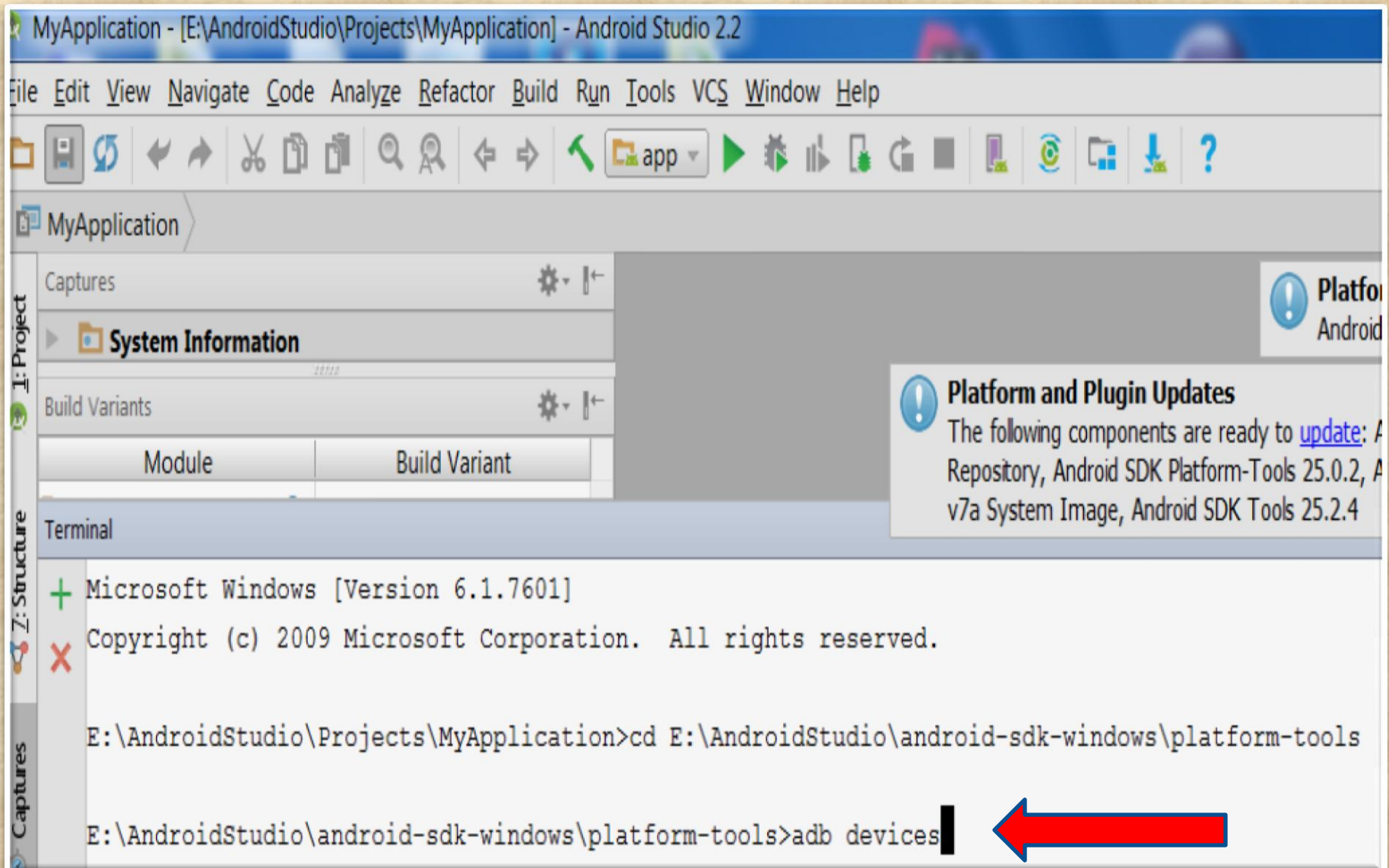
The screenshot shows the Android Studio 2.2 interface. The title bar reads "MyApplication - [E:\AndroidStudio\Projects\MyApplication] - Android Studio 2.2". The menu bar includes File, Edit, View, Navigate, Code, Analyze, Refactor, Build, Run, Tools, VCS, Window, and Help. The toolbar contains various icons for file operations, navigation, and development. The left sidebar shows the Project view with "MyApplication" selected, and the Structure view showing "Captures". The main editor area displays the "System Information" tab. A notification bubble in the bottom right corner states "Platform and Plugin Updates" with details about updates for the Android Repository, SDK Platform-Tools 25.0.2, and System Image v7a. The Terminal window at the bottom shows the following text:

```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

E:\AndroidStudio\Projects\MyApplication>cd E:\AndroidStudio\android-sdk-windows\platform-tools
E:\AndroidStudio\android-sdk-windows\platform-tools>
```

A red arrow points to the command prompt in the terminal window.

Mobile APPS: *Distribution/Installation/Logs*



The screenshot shows the Android Studio interface for a project named 'MyApplication'. The terminal window is open, displaying the following text:

```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

E:\AndroidStudio\Projects\MyApplication>cd E:\AndroidStudio\android-sdk-windows\platform-tools

E:\AndroidStudio\android-sdk-windows\platform-tools>adb devices
```

A red arrow points to the command 'adb devices' in the terminal. The interface also shows a 'Platform and Plugin Updates' notification in the bottom right corner, indicating that several components are ready to be updated.

Mobile APPS: Distribution/Installation/Logs

MyApplication - [E:\AndroidStudio\Projects\MyApplication] - Android Studio 2.2

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

MyApplication

Captures

System Information

Build Variants

Module	Build Variant
--------	---------------


Terminal

Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

```
E:\AndroidStudio\Projects\MyApplication>cd E:\AndroidStudio\android-sdk-windows\platform-tools
E:\AndroidStudio\android-sdk-windows\platform-tools>adb devices
List of devices attached
4dced50a      device
```

E:\AndroidStudio\android-sdk-windows\platform-tools>

Platform and Plugin Updates
The following components are ready to [update](#): Android Repository, Android SDK Platform-Tools 25.0.2, Android v7a System Image, Android SDK Tools 25.2.4



Mobile APPS: *Distribution/Installation/Logs*

The screenshot shows the Android Studio 2.2 interface. The title bar reads "MyApplication - [E:\AndroidStudio\Projects\MyApplication] - Android Studio 2.2". The menu bar includes File, Edit, View, Navigate, Code, Analyze, Refactor, Build, Run, Tools, VCS, Window, and Help. The toolbar contains various icons for file operations and development actions. The main workspace is titled "MyApplication" and contains several panels: "Captures", "System Information", "Build Variants", and "Terminal". The "Terminal" panel shows the following output:

```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

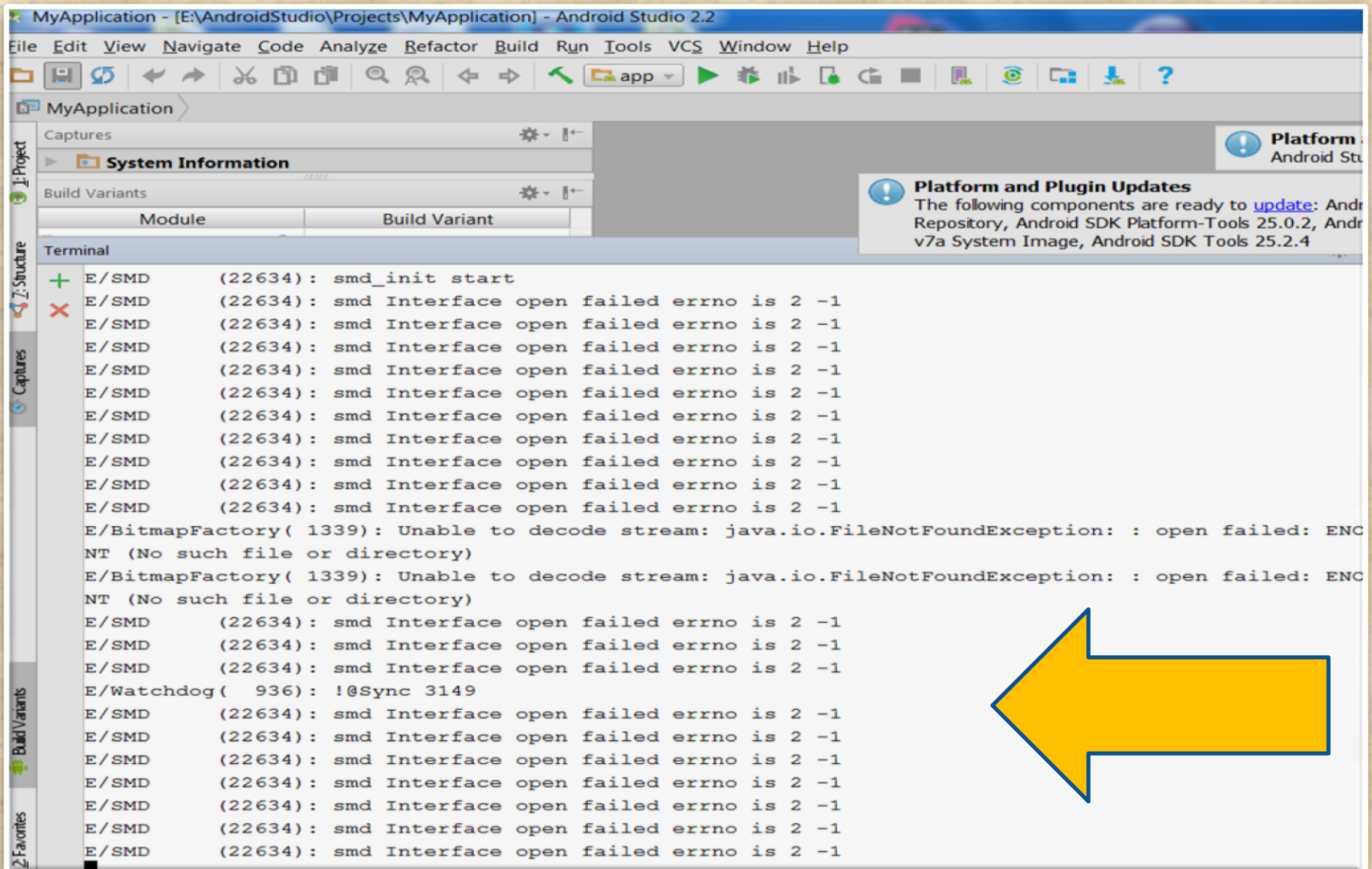
E:\AndroidStudio\Projects\MyApplication>cd E:\AndroidStudio\android-sdk-

E:\AndroidStudio\android-sdk-windows\platform-tools>adb devices
List of devices attached
4dced50a      device

E:\AndroidStudio\android-sdk-windows\platform-tools>adb logcat *:E
```

A red arrow points to the terminal output, specifically to the "adb logcat" command.

Mobile APPS: Distribution/Installation/Logs



MyApplication - [E:\AndroidStudio\Projects\MyApplication] - Android Studio 2.2

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

MyApplication

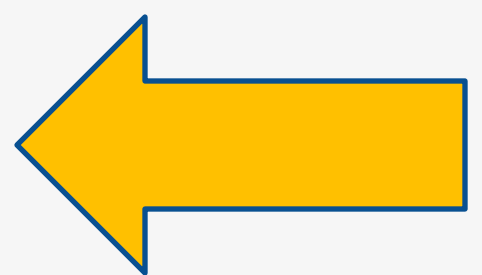
System Information

Build Variants

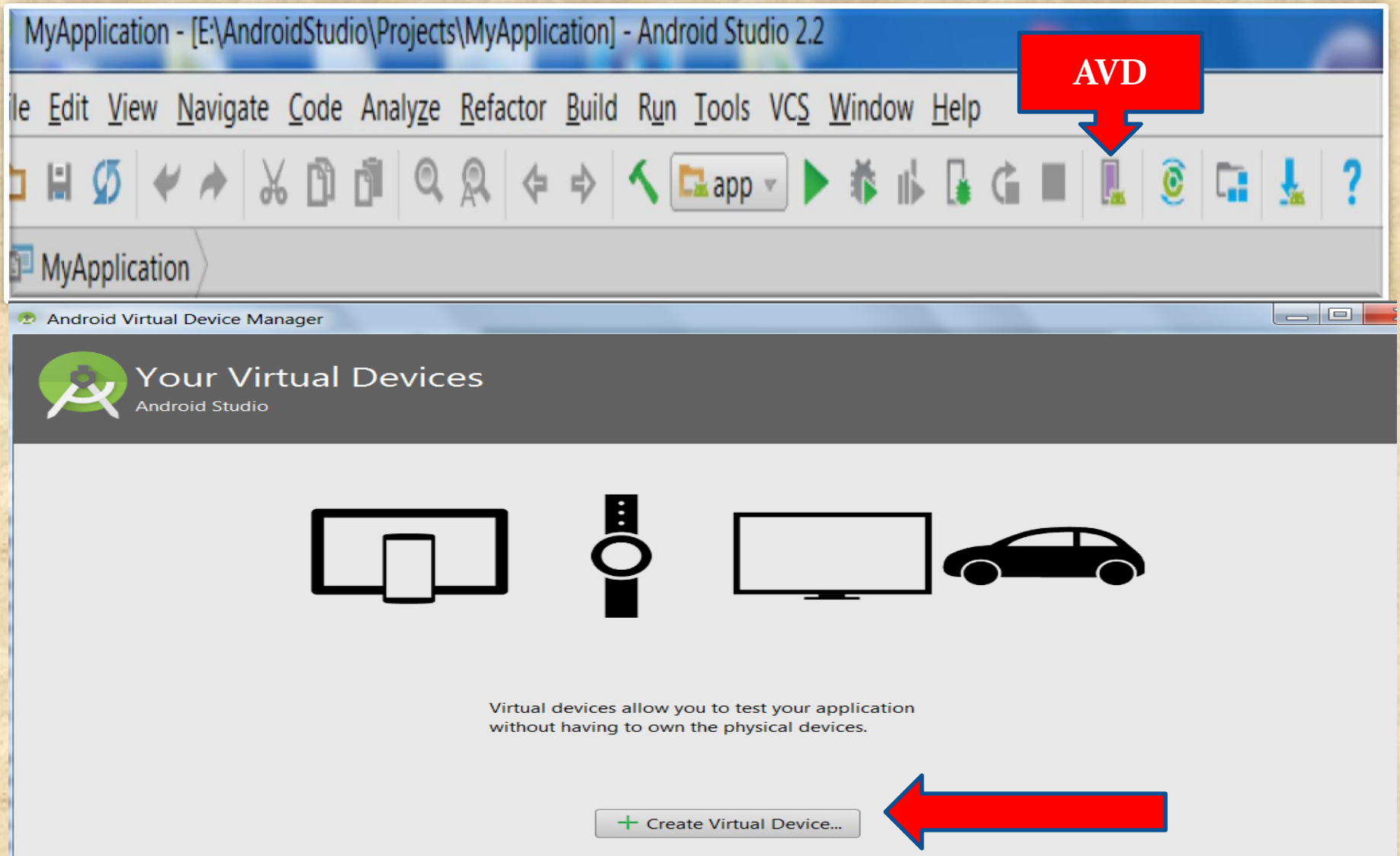
Terminal

```
+ E/SMD (22634): smd_init start
X E/SMD (22634): smd Interface open failed errno is 2 -1
E/SMD (22634): smd Interface open failed errno is 2 -1
E/SMD (22634): smd Interface open failed errno is 2 -1
E/SMD (22634): smd Interface open failed errno is 2 -1
E/SMD (22634): smd Interface open failed errno is 2 -1
E/SMD (22634): smd Interface open failed errno is 2 -1
E/SMD (22634): smd Interface open failed errno is 2 -1
E/SMD (22634): smd Interface open failed errno is 2 -1
E/SMD (22634): smd Interface open failed errno is 2 -1
E/SMD (22634): smd Interface open failed errno is 2 -1
E/BitmapFactory( 1339): Unable to decode stream: java.io.FileNotFoundException: : open failed: ENO
NT (No such file or directory)
E/BitmapFactory( 1339): Unable to decode stream: java.io.FileNotFoundException: : open failed: ENO
NT (No such file or directory)
E/SMD (22634): smd Interface open failed errno is 2 -1
E/SMD (22634): smd Interface open failed errno is 2 -1
E/SMD (22634): smd Interface open failed errno is 2 -1
E/Watchdog( 936): !@Sync 3149
E/SMD (22634): smd Interface open failed errno is 2 -1
E/SMD (22634): smd Interface open failed errno is 2 -1
E/SMD (22634): smd Interface open failed errno is 2 -1
E/SMD (22634): smd Interface open failed errno is 2 -1
E/SMD (22634): smd Interface open failed errno is 2 -1
E/SMD (22634): smd Interface open failed errno is 2 -1
E/SMD (22634): smd Interface open failed errno is 2 -1
E/SMD (22634): smd Interface open failed errno is 2 -1
```

Platform and Plugin Updates
The following components are ready to [update](#): Android Repository, Android SDK Platform-Tools 25.0.2, Android v7a System Image, Android SDK Tools 25.2.4



Mobile APPS: *Distribution/Installation/Logs*



Mobile APPS: Distribution/Installation/Logs

Virtual Device Configuration

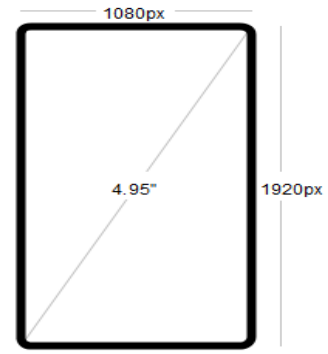
Select Hardware
Android Studio

Choose a device definition

Category	Name	Size	Resolution	Density
TV	Nexus S	4.0"	480x800	hdpi
Wear	Nexus One	3.7"	480x800	hdpi
Phone	Nexus 6P	5.7"	1440x2560	560dpi
Tablet	Nexus 6	5.96"	1440x2560	560dpi
	Nexus 5X	5.2"	1080x1920	420dpi
	Nexus 5	4.95"	1080x1920	xxhdpi
	Nexus 4	4.7"	768x1280	xhdpi
	Galaxy Nexus	4.65"	720x1280	xhdpi
	5.4" FWVGA	5.4"	480x854	mdpi
	5.1" WVGA	5.1"	480x800	mdpi
	4.7" WXGA	4.7"	720x1280	xhdpi

New Hardware Profile Import Hardware Profiles Refresh Clone Device...

Nexus 5




Size: normal
Ratio: long
Density: 420dpi

Previous **Next** Cancel Finish Help

Mobile APPS: Distribution/Installation/Logs

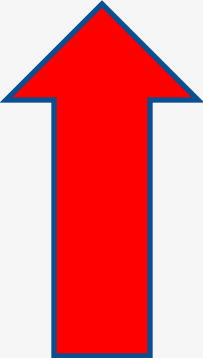
Virtual Device Configuration

 System Image
Android Studio


Select a system image

Recommended x86 Images Other Images

Release Name	API Level ▾	ABI	Target
null Download	25	x86_64	Android API 25 (with Google APIs)
null Download	25	x86	Android API 25 (with Google APIs)
Lollipop	22	x86	Android 5.1 (with Google APIs)



Lollipop



API Level
22

Android
5.1

Google Inc.

System Image
x86



Mobile APPS: Distribution/Installation/Logs

Virtual Device Configuration

Android Virtual Device (AVD)
Android Studio

Verify Configuration

AVD Name:

 Nexus 5	4.95 1080x1920 420dpi	<input type="button" value="Change..."/>
 Lollipop	Android 5.1 x86	<input type="button" value="Change..."/>

Startup orientation

Portrait Landscape


Emulated Performance: Graphics:

Device Frame Enable Device Frame

AVD Name

The name of this AVD.

Recommendation
Virtual machine acceleration driver is out-of-date.
[Reinstall Haxm](#)



Mobile APPS: *Distribution/Installation/Logs*

Android Virtual Device Manager

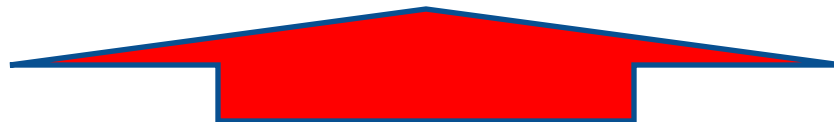


Your Virtual Devices
Android Studio

Virtual machine acceleration driver is out-of-date.

[Reinstall Haxm](#)

Type	Name	Resolution	API	Target	CPU/ABI	Size on Disk	Actions
	Nexus 5 API 22	1080 × 1920: xxhdpi	22	Android 5.1 (Goog...	x86	650 MB	



Mobile APPS: *Distribution/Installation: Android .APK*



Through an App Marketplace (*Google Play*)



by Email (*Android system recognizes the APK and displays an Install Now button in the email message*)



Through a Website (*host the release-ready APK file on your website and provide a download link*)



Google Drive



Android Studio



Test Fairy TOOL



enable allow "Unknown Sources" on the device
(*Settings > Applications > Unknown Sources*)

Mobile APPS: Distribution/Installation: Android .APK

Manually Install APK in Android Studio Emulator

1. Verify the presence of
X:\Program Files (x86)\Android \android-studio\sdk\platform-tools



2. Copy APK file into
X:\Program Files (x86)\Android \android-studio\sdk\platform-tools



3. go to Android Studio > Run Android Virtual Device Manager (AVD) > Start emulator



In this session we will use APK file get from "Candy Crush"

AVD Name	Target Name	Platform	API Level	CPU/Arch	
✓ MyAvd0	Android 4.4	4.4	19	ARM	New...
✓ instinct-emulator	Android 4.4	4.4	19	ARM	Edit...
✓ instinctcoder-phone	Android 4.4	4.4	19	ARM	Delete...
✗ A	?	?	?	?	Repair...
✗ AVD_for_10_in_WXGA_Tablet	?	?	?	?	Details...
✗ Test2	?	?	?	?	Start...

Legend:
✓ A valid Android Virtual Device. A repairable Android Virtual Device.
✗ An Android Virtual Device that failed to load. Click 'Details' to see the error.

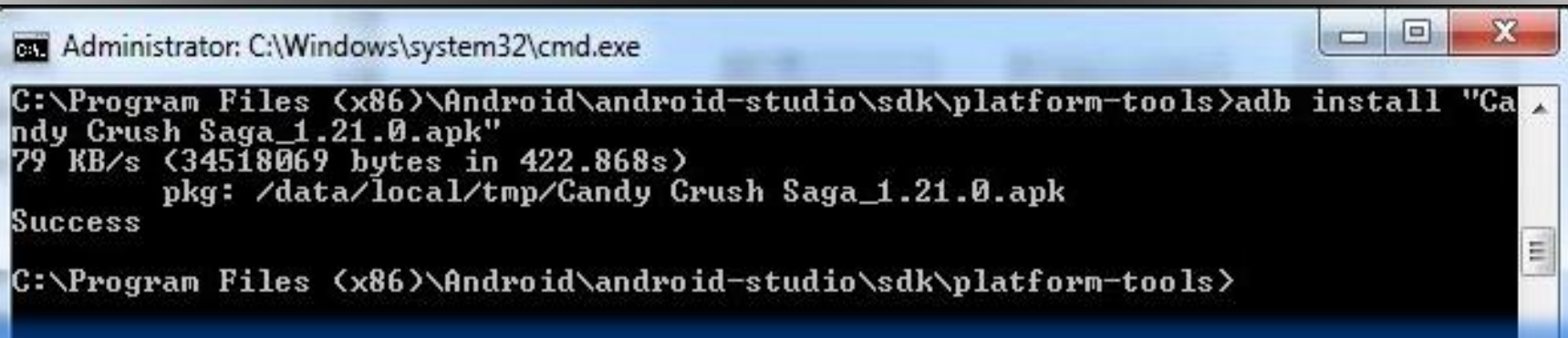
Mobile APPS: *Distribution/Installation: Android .APK*

Cont. : Manually Install APK in Android Studio Emulator

4. Go to Start > Run > Cmd

```
1 Type cd "C:\Program Files (x86)\Android\android-studio\sdk\platform-tools"  
2 Type adb install "Candy Crush Saga_1.21.0.apk"
```

5. After successfully installed, you will see result in command prompt like below



```
Administrator: C:\Windows\system32\cmd.exe  
C:\Program Files (x86)\Android\android-studio\sdk\platform-tools>adb install "Candy Crush Saga_1.21.0.apk"  
79 KB/s (34518069 bytes in 422.868s)  
  pkg: /data/local/tmp/Candy Crush Saga_1.21.0.apk  
Success  
C:\Program Files (x86)\Android\android-studio\sdk\platform-tools>
```

It will take about 5-6 minutes to install successfully

Mobile APPS: **Distribution/Installation: Android .APK**

Manually Install APK in Android Studio Emulator

6. Go to the emulator and you will see Candy Crush install in the emulator like below



Mobile APPS: **Distribution/Installation: Android .APK**

*What is
TEST
FAIRY ?*

TestFairy offers some great features for app developers. One of the stand out features is client side Video recording and not just screen shots.

TestFairy provides a video recording of the exact test from the client side, including CPU, Memory, GPS, Network monitoring, logs, crash reports and more.

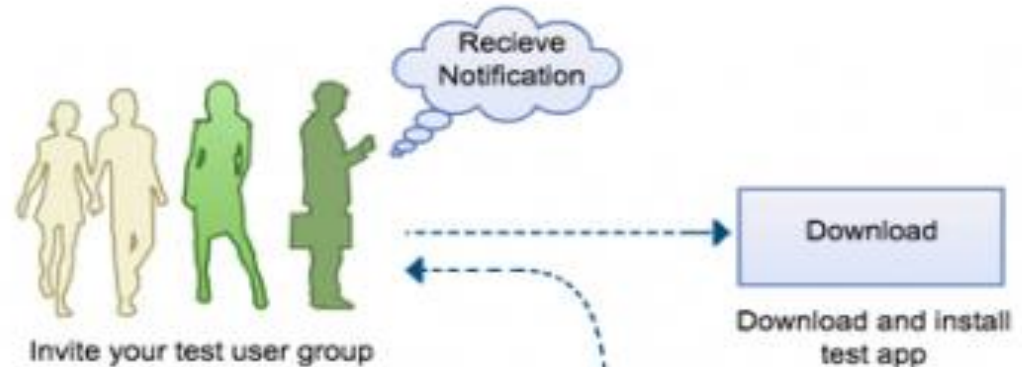
With Android apps you don't need to integrate any SDK or APIs into your app builds. You upload your APK (Android application file) to the TestFairy platform.

TEST FAIRY is for ANDROID ONLY

Mobile APPS: **Distribution/Installation:** Android .APK

TEST FAIRY

You will receive an invitation by email



Build and generate .apk
You may manually build or use build automation tools

Upload your app
You may upload using scripts, for CI builds.

Processing Build

Build ready
Now build is ready to download

TEST FAIRY is for ANDROID ONLY



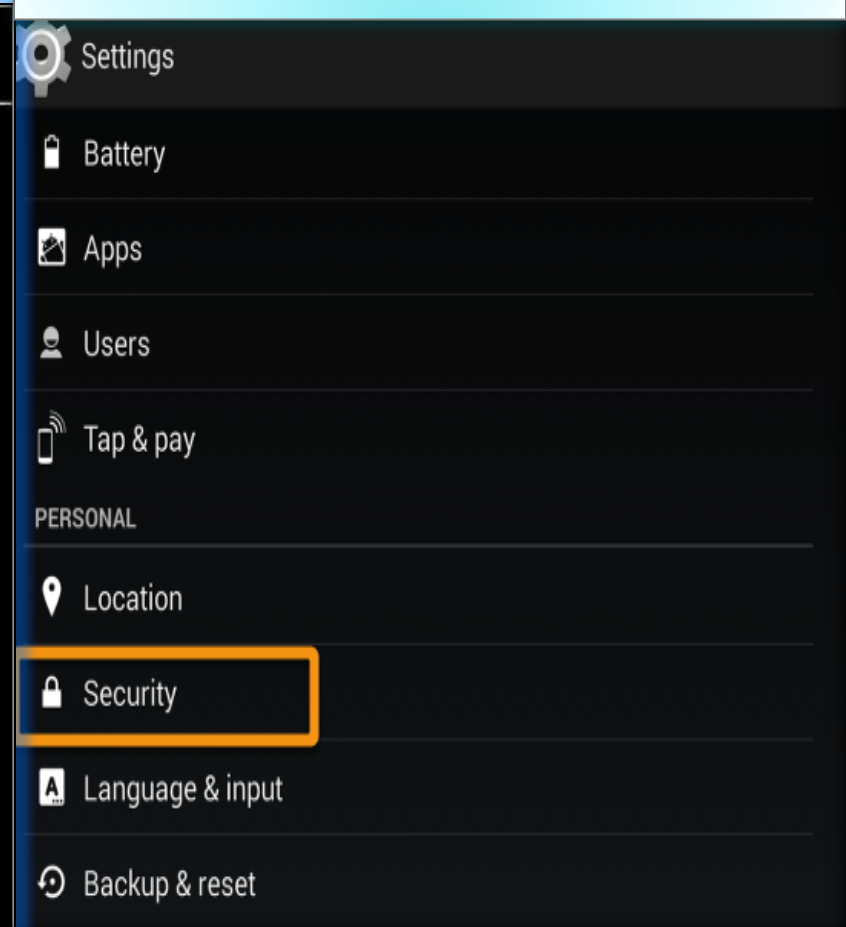
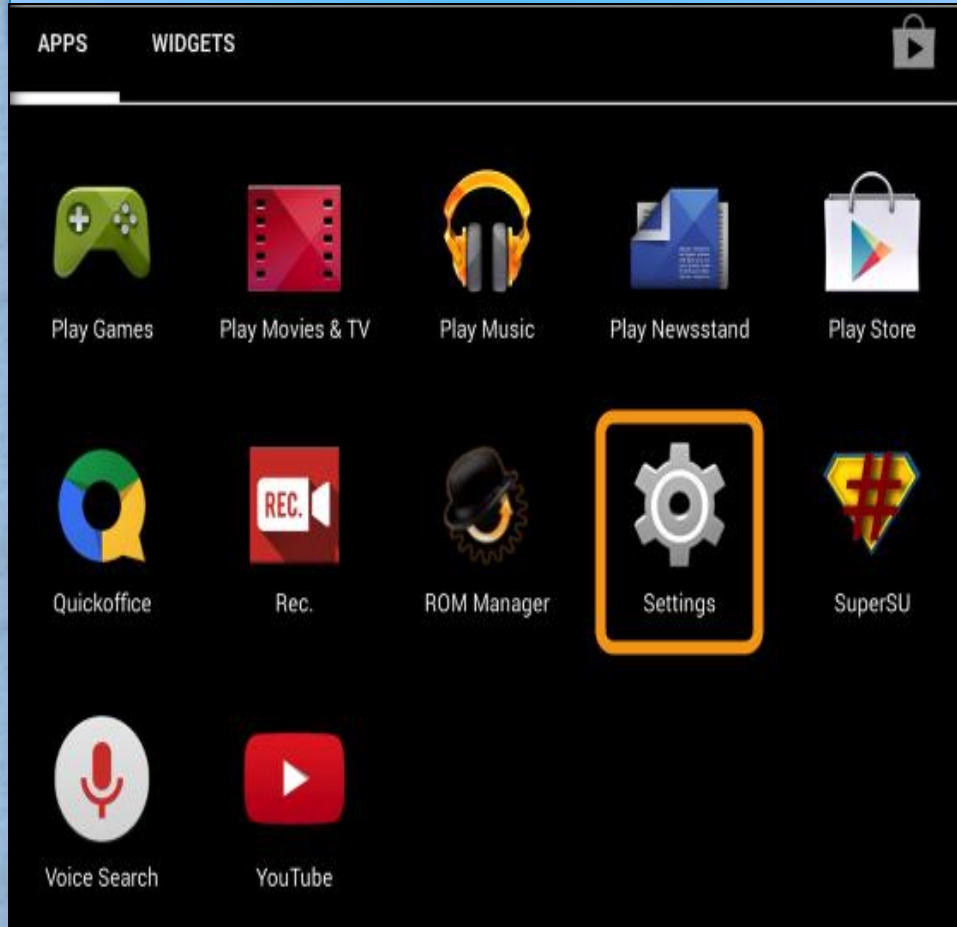
Test Fairy
Smart Android testing

Mobile APPS: *Distribution/Installation: Android .APK*

Installing THIRD PARTY APK

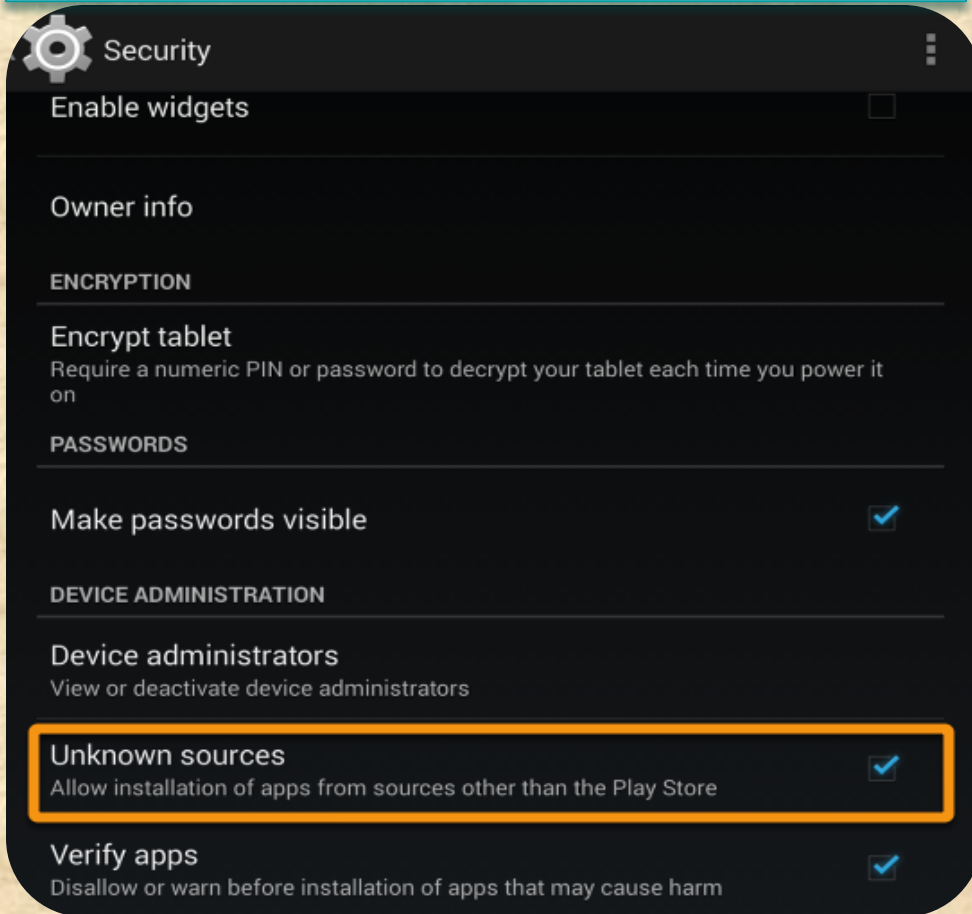
1. Go to the "Settings" application on your Android device

2. Choose the "Security" option located under the heading "Personal"



Installing THIRD PARTY APK

3. Under "Device Administration" place a checkbox next to the option "Unknown Sources"



This allows you to install applications on your Android device that are not downloaded directly from the Google Play store

Mobile APPS: *Distribution/Installation: Android .APK*

Transferring APK to your Device

1. On your computer, attach the ".apk" file to an e-mail and send it to an account that you can access via your Android device.



2. On the Android device, click on the ".apk" attachment in the e-mail in order to download it.



3. Follow the on-screen instructions to install the application.

Do you want to install this application? It will get access to:



read phone status and identity



modify or delete the contents of your USB storage
read the contents of your USB storage



find accounts on the device

DEVICE ACCESS



full network access
receive data from Internet
view network connections
view Wi-Fi connections

Cancel

Install



Mobile APPS: Android LOGS : EXTRA

Installing LOGCAT APP to your Device

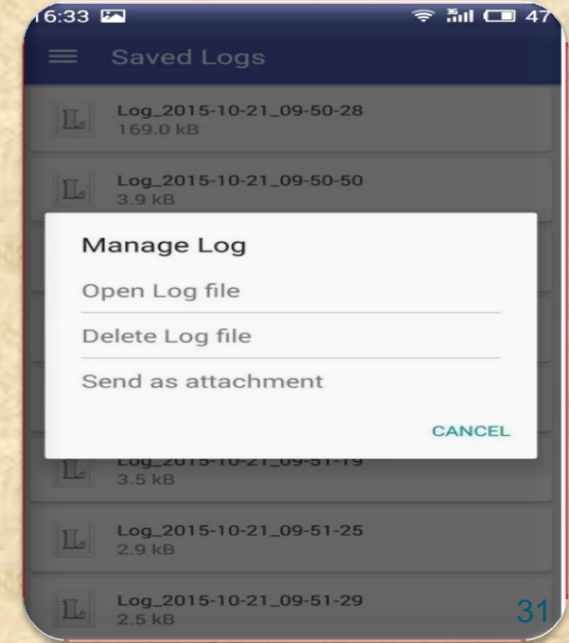
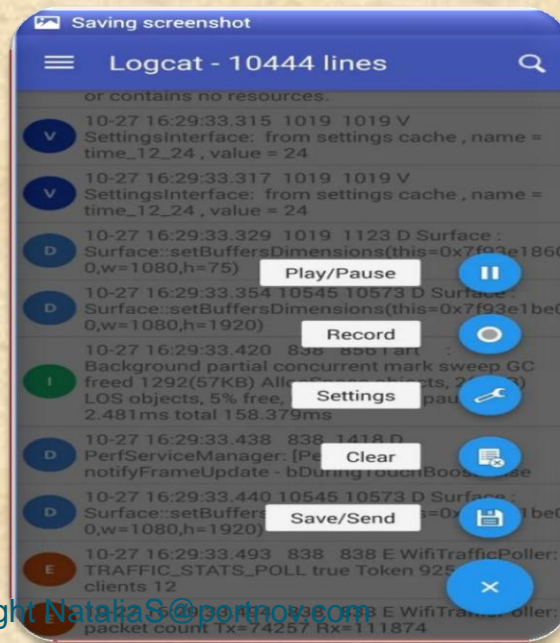
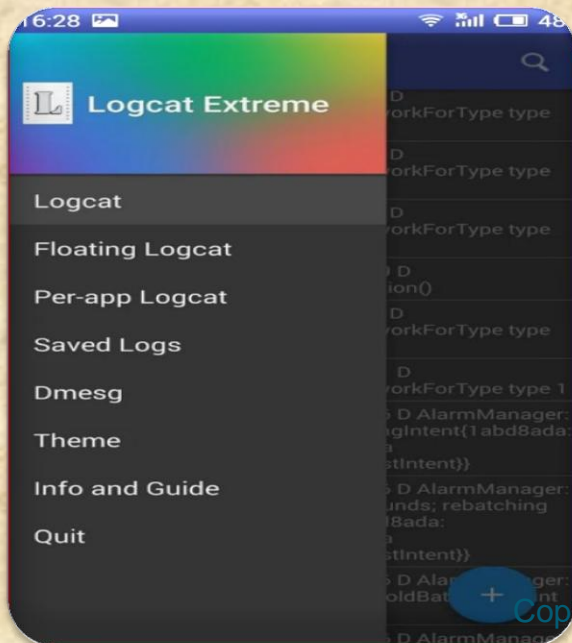
Download
LOGCAT EXTREME
from PlayStore to
your Device directly
(*this is only an
example, there are
many other similar
apps*)



LOGCAT EXTREME
is an enhanced
Logcat reader and
Logcat recorder
which comes with a
rich set of features
and handy user
interface.



Please note: From
Android 4.1 onwards
ANY logcat app needs
root access in order to
show logs properly.



Mobile APPS: *Distribution/Installation: iOS .IPA*



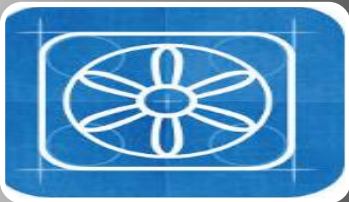
Through an App Marketplace (APP STORE)



XCODE



ITUNES



TEST FLIGHT

Mobile APPS: **Distribution/Installation: .IPA**



What is XCode?

Xcode is an Intergrated Development Environment by Apple containing a suite of software Development Tools for macOS, iOS, WatchOS and tvOS

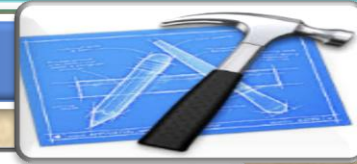
Xcode supports source code for programing languages C, C++, Objective-C, Objective-C++, Java, AppleScript, Python, Ruby, ResEdit (Rez), and Swift

Also supports variety of programming models, including but not limited to Cocoa, Carbon, and Java



Mobile APPS: *Distribution/Installation: .IPA*

Distributing .IPA through XCode ?



Connect the device to your Mac.



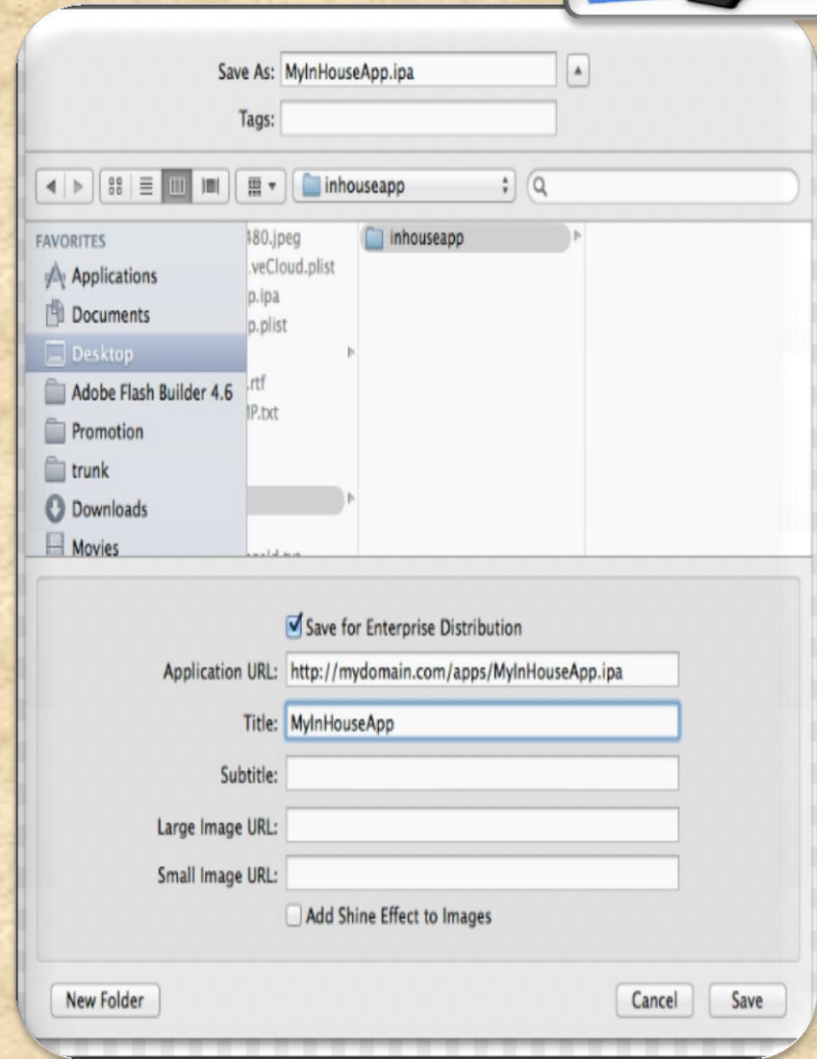
In Xcode, choose Window > Devices and select the device under Devices.



In the dialog that appears, choose the iOS App file and click Open.



In the Installed Apps table, click the Add button (+) below the table.



Mobile APPS: *Collecting LOGS: .IPA*



How to do it through Xcode on MAC ?

1. Install XCode

2. Connect your iPhone to the Mac

3. Select Trust this computer on the iPhone pop-up request

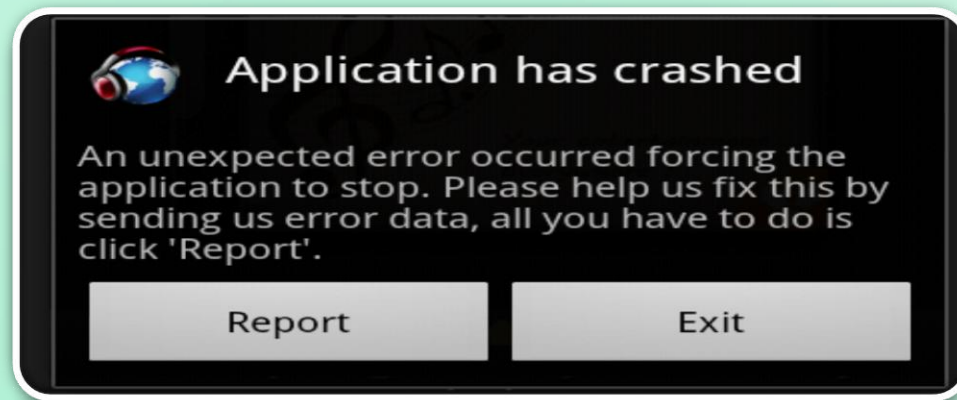
4. Start xCode (Menu) → Window → Devices (Select your iPhone and press the arrow button in the bottom right)

5. Reproduce the problem

6. Press the arrow button and download the logs

How can I debug a deployed app without Xcode debugger ?

- ❖ A: Once you have deployed your app, either through the *App Store* or as an *Ad Hoc* or *Enterprise* build, you won't be able to attach *Xcode's* debugger to it.
- ❖ To debug problems, you need to analyze *Crash Logs* and *Console* output from the device.



Getting Crash Logs and Console Output

Getting Crash Logs Directly From a Device

Without Xcode

Users can retrieve crash reports from their device and send them to you via email by following these instructions.

(It is not possible to get device console logs directly from a device)

- ❖ Open Settings app
- ❖ Go to Privacy, then Diagnostics & Usage
- ❖ Select Diagnostics & Usage Data
- ❖ Locate the log for the crashed app. The logs will be named in the format: *<AppName>_<DateTime>_<DeviceName>*
- ❖ Select the desired log. Then, using the text selection UI select the entire text of the log. Once the text is selected, tap Copy
- ❖ Paste the copied text to Mail and send to an email address as desired

Mobile APPS: *Distribution/Installation: .IPA*



What is iTunes?

iTunes is a media player, media library, online radio broadcaster, and mobile device management application developed by Apple Inc.

It is used to play, download, and organize digital downloads of music and video (as well as other types of media available on the iTunes Store) on personal computers running the macOS and Microsoft Windows operating systems.

What is iTunes?



Mobile APPS: **Distribution/Installation: .IPA**

Drag-and-drop IPA file into 'Apps' tab of iTunes BEFORE you connect the device



Connect your device

Select your device on iTunes

Select 'Apps' tab

Search app that you want to install

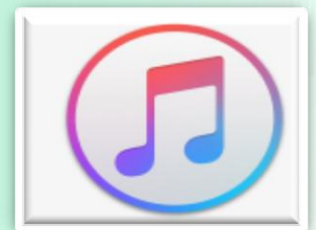
Click on 'Install' button. This will change to 'Will Install'

Click on 'Apply' button on right corner

LOGS: Collecting from iTunes

- ❖ Sync your device with iTunes on your desktop.
- ❖ After syncing, look for crash logs in the correct directory.

The Following few slides will give an instructions . Lets start !



LOGS: Collecting from iTunes

❑ Mac OS X:

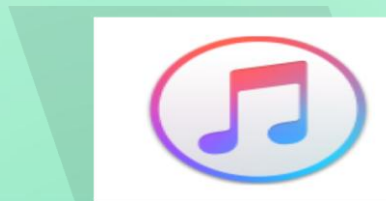
- ❖ Open Finder (found in the Dock)
- ❖ Click on the 'Go' menu at the top of your screen, and select 'Go to Folder'
- ❖ Type (or paste): `~/Library/Logs/CrashReporter/MobileDevice/<DEVICE_NAME>`
- ❖ Open the folder with the same 'name' as your device.
(Note: your device name appears in iTunes on the left side, under 'Devices').
- ❖ Open the folder called 'Retired'
- ❖ You will see at least one item starting with 'ReadItLaterPro'.



LOGS: Collecting from iTunes

❑ Windows Vista or 7:

- ❖ Open any Windows Explorer Window (My Computer, My Documents, etc.)
- ❖ Enter %appdata%, and press enter → Navigate to Roaming
C:\Users\<USERNAME>\AppData\Roaming\Apple
- ❖ *Computer\Logs\CrashReporter\MobileDevice\<DEVICE_NAME>*
- ❖ (Note: your device name appears in iTunes on the left side, under 'Devices')
- ❖ You will see at least one item starting with 'ReadItLaterPro'.



LOGS: Syncing your device with iTunes

❑ Windows XP:

- ❖ Locate your Application Data folder.
- ❖ Navigate to Apple computer *C:\Documents and Settings\<USERNAME>\Application Data\Apple*
- ❖ *Computer\Logs\CrashReporter\MobileDevice\<DEVICE_NAME>*
- ❖ (Note: your device name appears in iTunes on the left side, under 'Devices')
- ❖ You will see at least one item starting with 'ReadItLaterPro'.

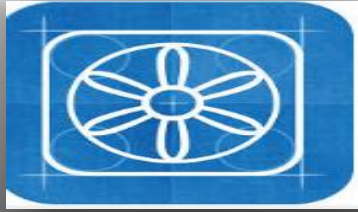


LOGS: Collecting from your Device

- To obtain iOS crash logs, please select your device and go to:
- *Settings > Privacy > Diagnostics & Usage (for iOS 8 or newer)*
- *Settings > General > About > Diagnostics & Usage (for iOS 7 or older)*

- ❖ Select a *Chrome* crash from the list.
This will start with “*Chrome_*” and contain the timestamp of the crash.
- ❖ Tap on the crash and you will see a text field with a crash log.
Long press to *Select All* and then *Copy* the crash text.
- ❖ Paste it into something you can get off of your device (for example, an email to yourself).

Mobile APPS: **Distribution/Installation: .IPA**



What is TestFlight?

TestFlight is an online service for over-the-air installation and testing of mobile applications, currently owned by Apple Inc and only offered to developers within the iOSDeveloper Program

Developers signed up with the service to distribute applications to internal or external beta testers, who could subsequently send feedback about the application to developers

The TestFlight SDK additionally allows developers to receive remote logs, crash reports and tester feedback.

HOMEWORK : READ AN ENTIRE ARTICLE ABOUT TestFLIGHT

<https://www.raywenderlich.com/48750/testflight-sdk-tutorial>



Overview: Mobile APPS

➤ Categories

➤ Types

➤ Distribution/Installation/Logs

➤ Mobile Test Industry Standards

➤ Remote Device Access (RDA)

➤ Emulators

➤ Simulators

➤ Troubleshooting Guide

➤ App Risk Analysis

Mobile Test Industry Standards :

Testing Strategies for Mobile Apps

Challenges

Device fragmentation

In-house vs. outsourced testing

Availability of mobile testing tools

Application Lifecycle Testing

- Like any desktop or web application testing, mobile application testing is also focused on the quality and performance of the final product.
- However, mobile app testing becomes far more challenging because of the following key factors

