

# Preparation of the Work Environment Introduction

**Unit 0**

- 1 Introduction to Testing Laboratories
  - Security Testing
  - Mobile Testing Laboratory
- 2 Android Simulator
- 3 iOS Simulator
- 4 Other Environments and Professional Tools
  - Windows Phone
  - BlackBerry
- 5 Assessment Test



# Introduction to Testing Laboratories

# ◆◆◆ Introduction to Testing Laboratories

## Importance of a Security Laboratory

- Regardless the platform, the software life cycle requires a continuous use of engineering processes and tools.
- Specifically, the security context provides, among other elements:
  - Threat modelling.
  - Security requirements.
  - Policies.
  - Test platforms and definition.
  - Security metrics.
  - Simulation tools.
  - Secure programming guides and processes.
- The security laboratory is a support tool for the processes above.

# ◆◆◆ Introduction to Testing Laboratories

## The Security Laboratory



- The security laboratory is not a static and universal system.
- In addition, mobile environment and technologies has different standards and regulations.
- The security requirements are not the same for all the products and organisations.
- It depends on:
  - Type of product.
  - The organisation in which it is developed.
  - Clients.
  - Legislation.
- Just like the rest of engineering processes, the security laboratory should be adapted to the different needs.

# ◆◆◆ Introduction to Testing Laboratories

## Objectives of a Mobile Security Laboratory

- A security laboratory for mobile devices should allow users to:
  - Analyse the security and determine risks created by:
    - The software installed in the device.
      - Third parties' applications (with or without an available source code).
      - Official applications (installed by operators or manufacturers).
      - Operative System.
      - ...
    - Its hardware.
      - Wireless interfaces.
      - External storage.
      - Wired connections.
      - ...
    - Its configuration.
      - Password and account policies.
      - ...
  - Extract relevant information from a mobile device.



## Objectives of a Mobile Security Laboratory

- Tasks performed using a security laboratory allow users to:
  - Ensure the conformity of a device or an application.
  - Ensure a specific level of security.
  - Identify threats and countermeasures existing in a system.
  - Transform users and decision-making bodies of the organisation into effective parts of the organisation's security.
- The security laboratory also depends on the complexity level required in the different analysis.
  - When a deeper analysis is required, the complexity, as well as the material and human cost are greater.
    - The analysis of low-level elements (firmware, etc.) of a device requires a more complex laboratory than when analysing third parties' applications.

A photograph of a person's hands and arms working at a wooden desk. The person is holding a pen over an open notebook. On the desk, there are two glasses of iced coffee, a laptop, and some papers. The scene is lit with warm, natural light. A semi-transparent dark rectangle is overlaid on the center of the image, containing the text "Security Testing" in white.

# Security Testing



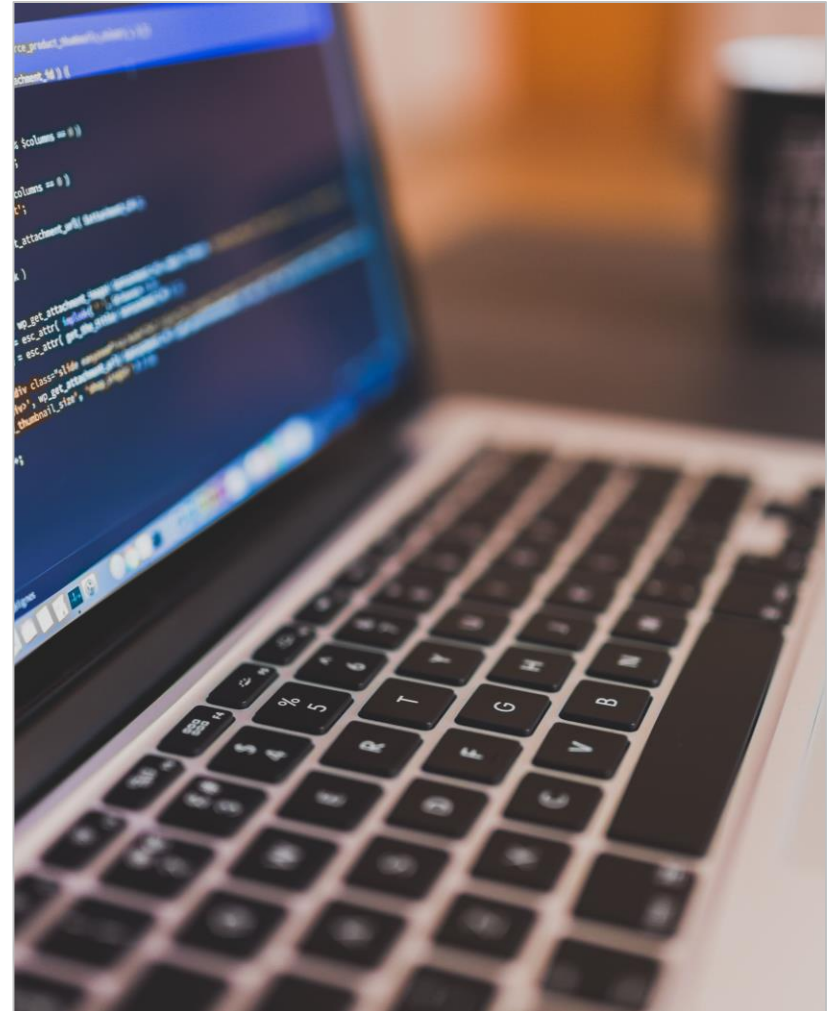
## Types

- A security laboratory allows users to perform different types of analysis.
  - According to the analysis procedure:
    - Static analysis.
    - Dynamic analysis.
  - According to the systems involved in the analysis:
    - On-device analysis.
    - Emulator/virtual machine analysis.
    - Analysis in the cloud.
  - According to the interaction with the analyst:
    - Automatic analysis.
    - Manual analysis.
- It is advisable to use as many approaches as possible.

# ◆◆◆ Security Testing

## Static Analysis I

- It allows analysts to study characteristics of applications without executing them.
- Analysable examples:
  - Code:
    - Original or compiled source.
  - Permissions and applications' manifest.
  - Images and other resources used by the application.
- Techniques used:
  - Information flow and taint analysis.
    - It performs a simulation of the execution of certain elements of the application.
  - Control flow graphs.
    - They create graphical representation of the execution order.



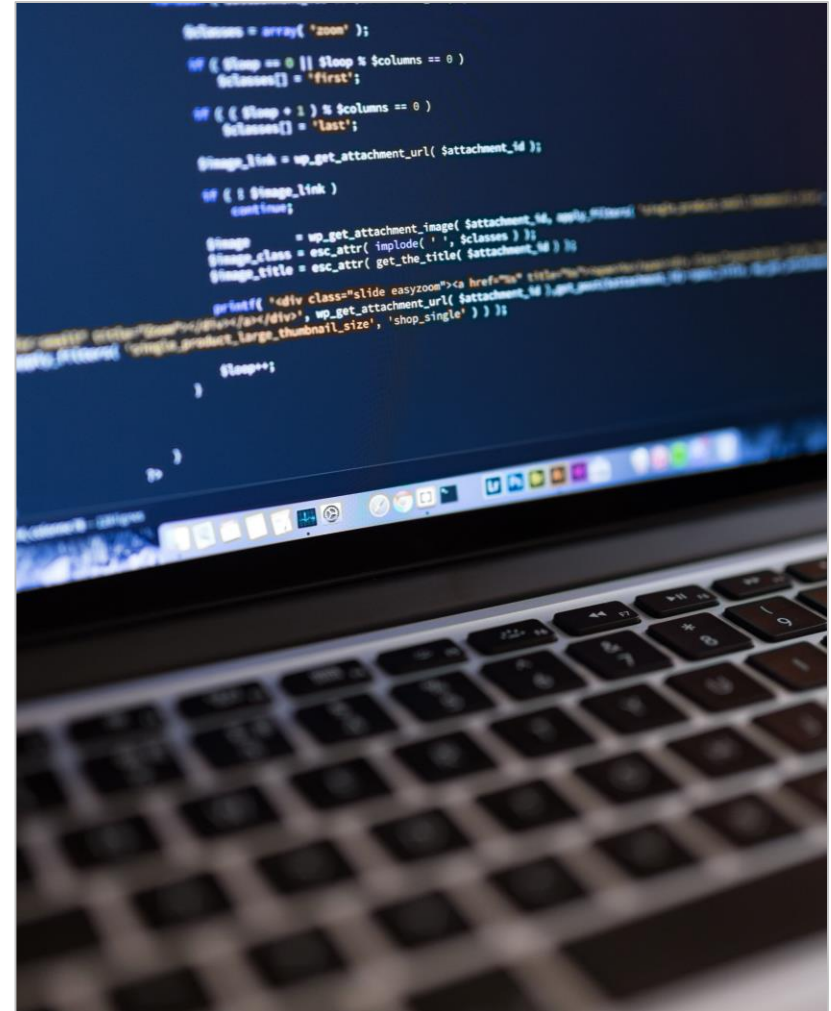
## Static Analysis II

- Advantages:
  - Easily automatable.
  - Quite effective detecting certain types of vulnerabilities.
    - Buffer overflows.
    - Code injections.
  - Some development tools include it as an additional feature.
- Disadvantages:
  - Possibility of finding a large number of false positives.
    - The results have to be manually checked.
  - Some vulnerabilities are difficult to detect automatically.
    - Weak encryption.
    - Vulnerable protocols.
    - Misconfiguration issues.

# ◆◆◆ Security Testing

## Dynamic Analysis I

- It checks the security of an application by executing it.
- Analysable examples:
  - Memory of processes.
  - Use of resources (CPU, network, battery).
  - Calls to the system.
  - Result of functions against the handling of parameters.
- Techniques used:
  - Monitored execution in sandbox.
  - Fuzzing (automatic creation of inputs).
  - Monitoring code injection.
  - Vulnerability scan.



## Dynamic Analysis I

- Advantages:
  - It allows users to check the real behaviour of an application.
  - It may access information that is not accessible through other techniques.
  - Controlled environment.
- Disadvantages:
  - In general, it is more complex than the static analysis.
    - It requires the execution of the program during a period of time.
  - It is not always possible to conduct the analysis in a virtual machine or simulator.
    - The result may depend on the machine in which the application is executed.
  - The interaction with the application is a key point.
    - Some inputs as an accelerometer or other kind of sensor are difficult to create.
  - It is more difficult to automatize.



## Involved Systems

### Device:

Tests with real devices and data.

Sometimes, it is the only possible option.

More expensive.

### Cloud:

Outsourcing of many analysis tasks.

Handling of very sensitive data by third parties.

### Virtual machine:

It enables the reduction of analysis costs.

It provides a greater control over the analysed element.

The results may differ in comparison with a real system.

## Automatic vs Analyst Analysis

### Automatic

- ✓ It enables the storage of a great amount of data.
- ✓ It may be added to the development tools.
- ✓ Low implementation costs.

### Analyst

- ✓ It is able to detect complex issues.
- ✓ It may use automatic tools.
- ✓ It may detect unknown threats.

A close-up, angled view of a smartphone screen displaying a utility application. The interface features several colorful tiles: a teal tile with a circular progress indicator and the text '1 DAY LEFT', a teal tile labeled 'BILLING' showing 'New charges \$69', a teal tile with a car image and the text 'PAY BILL', and a yellow tile at the bottom with the text 'GET MORE'. The phone is silver and has a home button. The background is blurred with bokeh light effects.

# Mobile Testing Laboratory

# ◆◆ Mobile Testing Laboratory

## Components

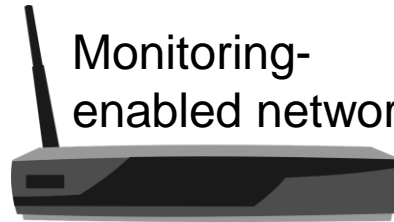
Devices



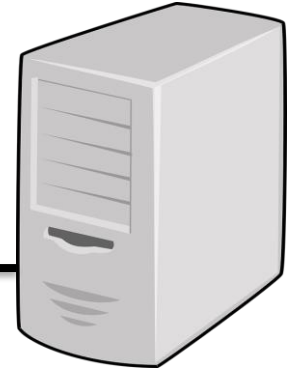
Work  
stations



Monitoring-  
enabled network

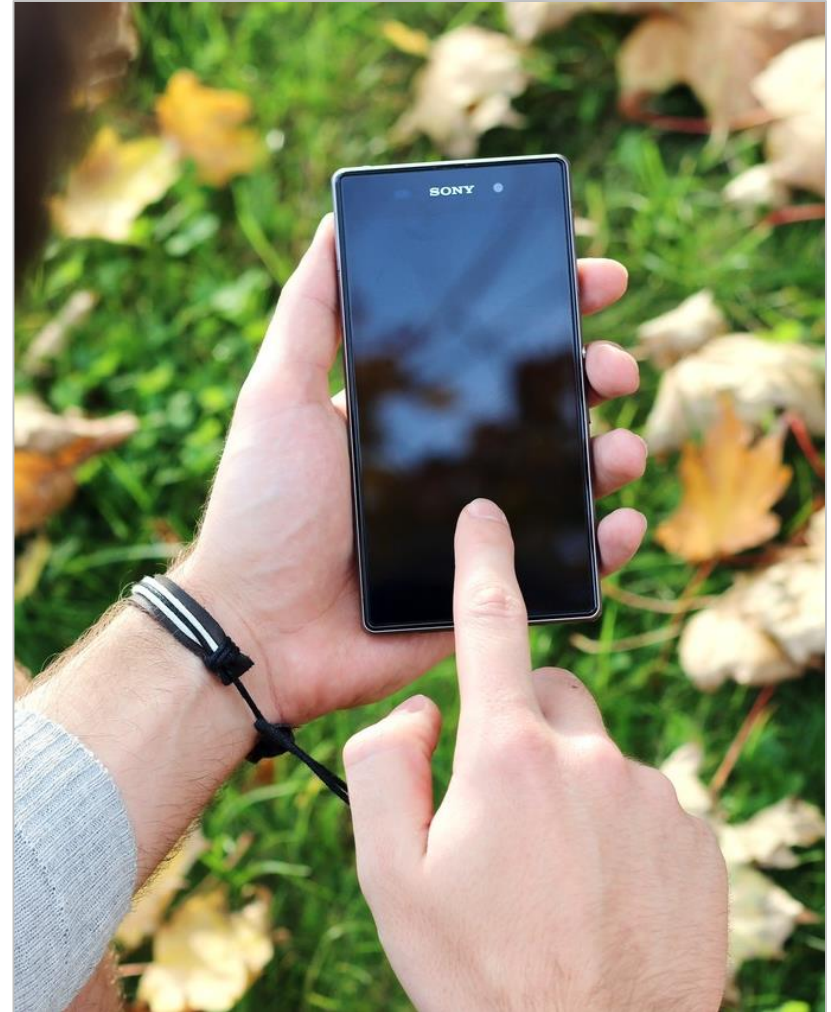


Servers



## Mobile Devices

- Different operative systems and models will be required.
  - According to the complexity of the laboratory.
  - In some cases, they can be substituted by emulators/virtual machines.
- Execution of the applications to analyse.
  - Loaded from the work stations.
  - Dynamic analysis of each application.
    - According to the platform, it may require jailbreak/rooting to analyse some applications.
  - They allow users to conduct forensic analysis.
    - Applications' logs.





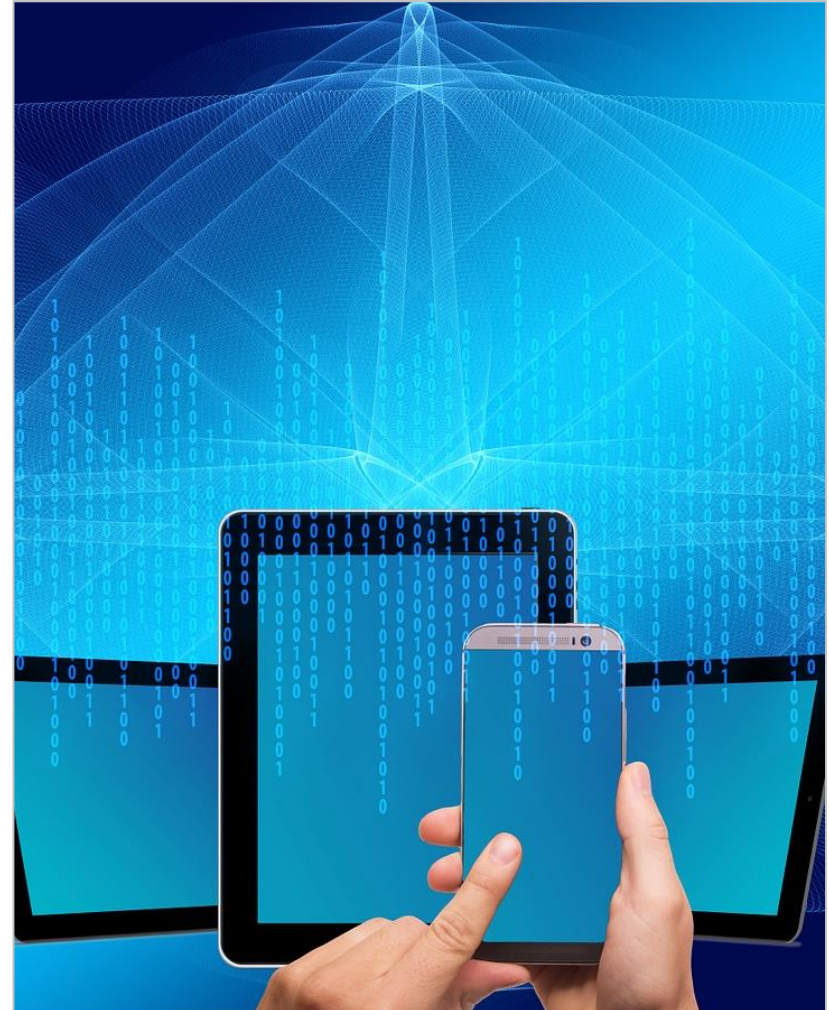
## Work stations

- They allow users to load applications in the devices through the official development tools.
- In some platforms, the official tools are restricted to combinations of hardware and operative system (iOS and Windows Phone).
- They conduct static analysis:
  - Via official and third parties tools.
- They conduct dynamic analysis:
  - Via simulators or virtual machines.
- They may be configured to act as proxy in order to analyse the devices' network traffic.



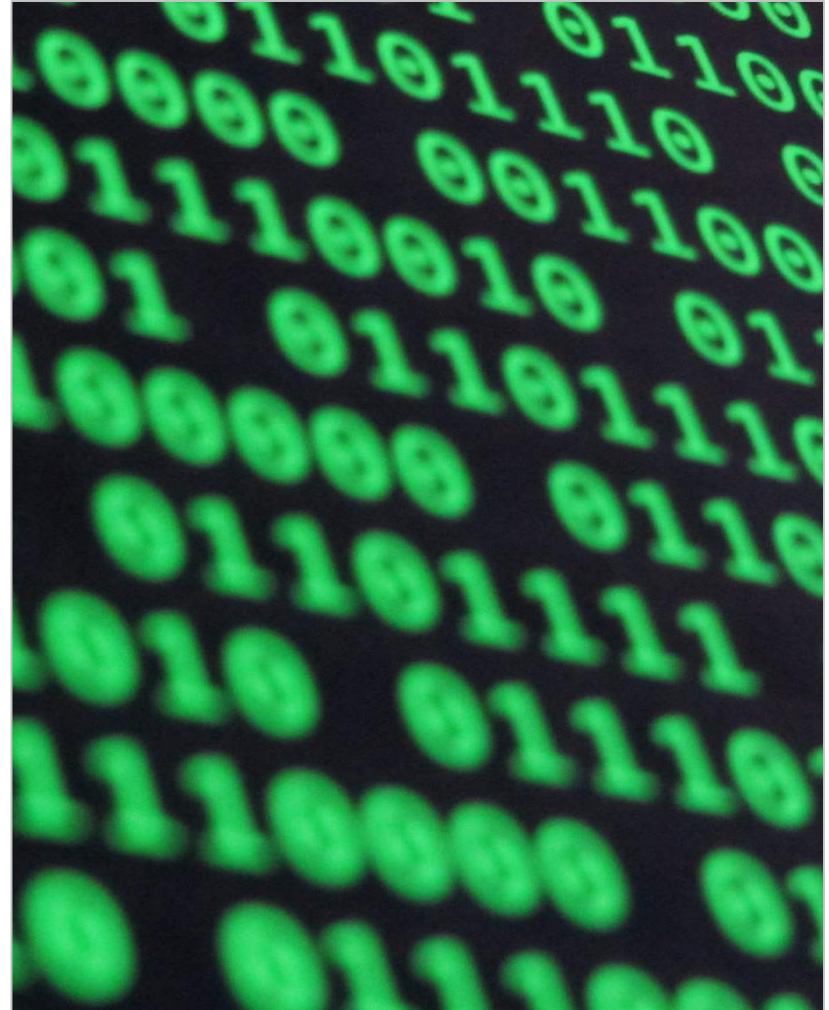
## Network Infrastructure

- It connects devices with the outside world and enables the analysis of traffic created by the mobile device.
- It is necessary to configure it for the traffic to be accessible. There are two possible solutions:
  - Redirection of all the traffic received by the router to other device.
    - It depends on the model of router.
  - Wireless security that allows users to capture traffic by a work station.
    - Watch out! Other computers will also be able to access the traffic.
- It will not be possible to inspect the traffic via properly protected SSL connections.
  - Solution:
    - Direct access from the device via dynamic analysis.



## Network Services and Proxy Tools

- Network Services:
  - They are used to emulate services provided to devices and applications.
  - They provide logs and traces that can be useful for the security analysis.
- Proxy Tools:
  - Allow users to capture part of the traffic sent to be analysed.
  - In some cases, they can be installed in the work stations.





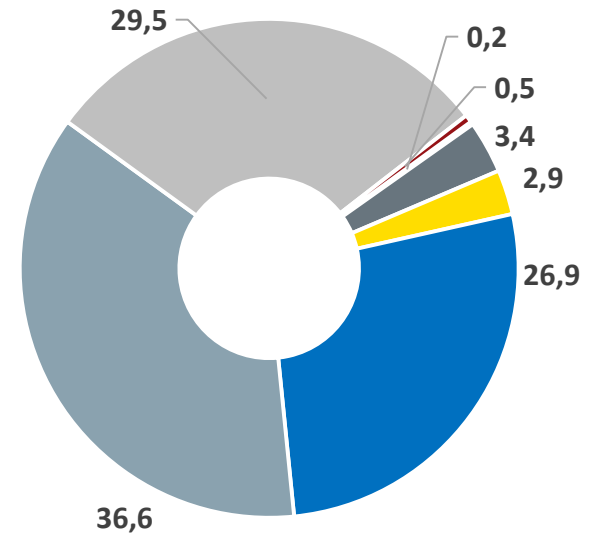
A green Android robot character is positioned on the right side of the image. It has a rounded head with two small antennae, two circular eyes, and a friendly expression. The robot's body is also rounded and green, with small legs visible at the bottom. The background is dark with numerous out-of-focus light circles in shades of blue and orange, creating a bokeh effect.

# Android Simulator

## Introduction

- Originally developed by Android Inc. (bought by Google in 2005).
- Its current development is managed by the Android Open Source Project (AOSP), maintained by Google and promoted by the Open Handset Alliance since 2007.
- It is an open source system that is generally customized with manufacturers and operators proprietary software.

Distribution of versions



- Froyo (2.2)
- GingerBread(2.3.3-7)
- Ice Cream Sandwich (4.0.3-4)
- Jelly Bean (4.1-3)
- KitKat (4.4)
- Lollipop (5.0-1)
- Marshmallow (6.0)

December 2015



## Testing Laboratory I

### Testing environment



The testing environment for Android operative system of this course allows users to:

- Develop applications.
- Install and execute applications in devices.
- Test different features of an application during its execution.
- Conduct static analysis.
- Conduct dynamic analysis.
- Conduct forensic analysis.



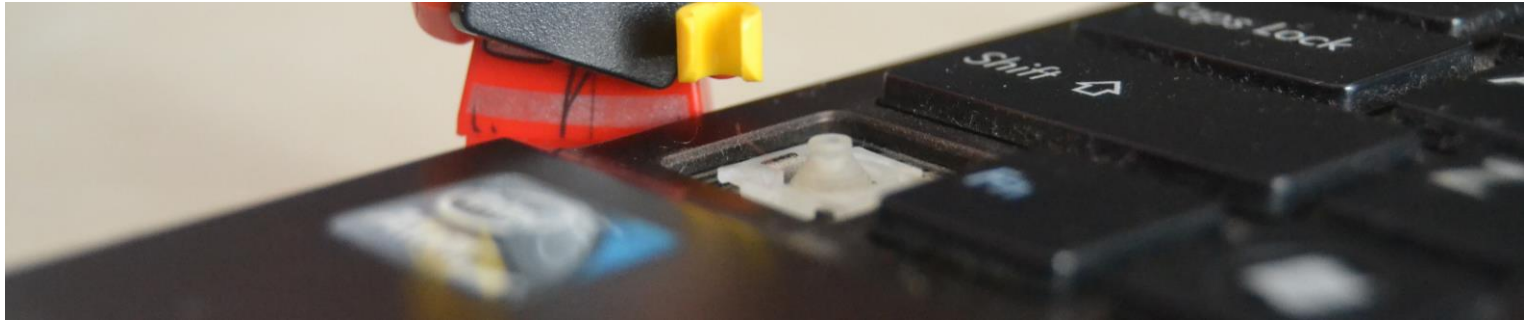
## Testing Laboratory II

### Testing environment



The testing environment for Android operative system of this course includes:

- Google Official Development Kit.
- Third parties' tools for the static analysis.
- Third parties' tools for the dynamic analysis.
- Android applications to learn about vulnerabilities, secure programming and security services of Android.
- Tools for the extraction of information.



## Google Official Development Kit

- It is a set of tools that allow users to develop, install, execute, debug and distribute applications for Android OS.
- It enables the creation of applications for any Android version.
  - Smartphones: for smartphones and tablets.
  - Wear: for wearable devices such as bracelets.
  - TV: for smart TVs.
  - Car: for vehicles.
- Applications written in Java
  - Some tasks that require low level access to the device may be programmed in C with the NDK (Native Development Kit).

## Components of the Google Official Development Kit

- This kit includes various tools.
- Integrated Development Environment (IDE):
  - Graphical environment for the creation and debugging of applications.
  - There are two official options on Android.
    - Android Studio (official since 2014).
    - Eclipse (with plug-in previous to 2014).
- Software Development Kit (SDK):
  - Tools that allow users to compile and debug Android applications.
  - IDE automatize its use, but it is not necessary.
  - It includes:
    - Tools for compilation, debugging and communication with devices.
    - System libraries to be used by third parties' applications.
    - Emulators to execute and debug applications.
    - SDK management tools.

## Tools – Android SDK

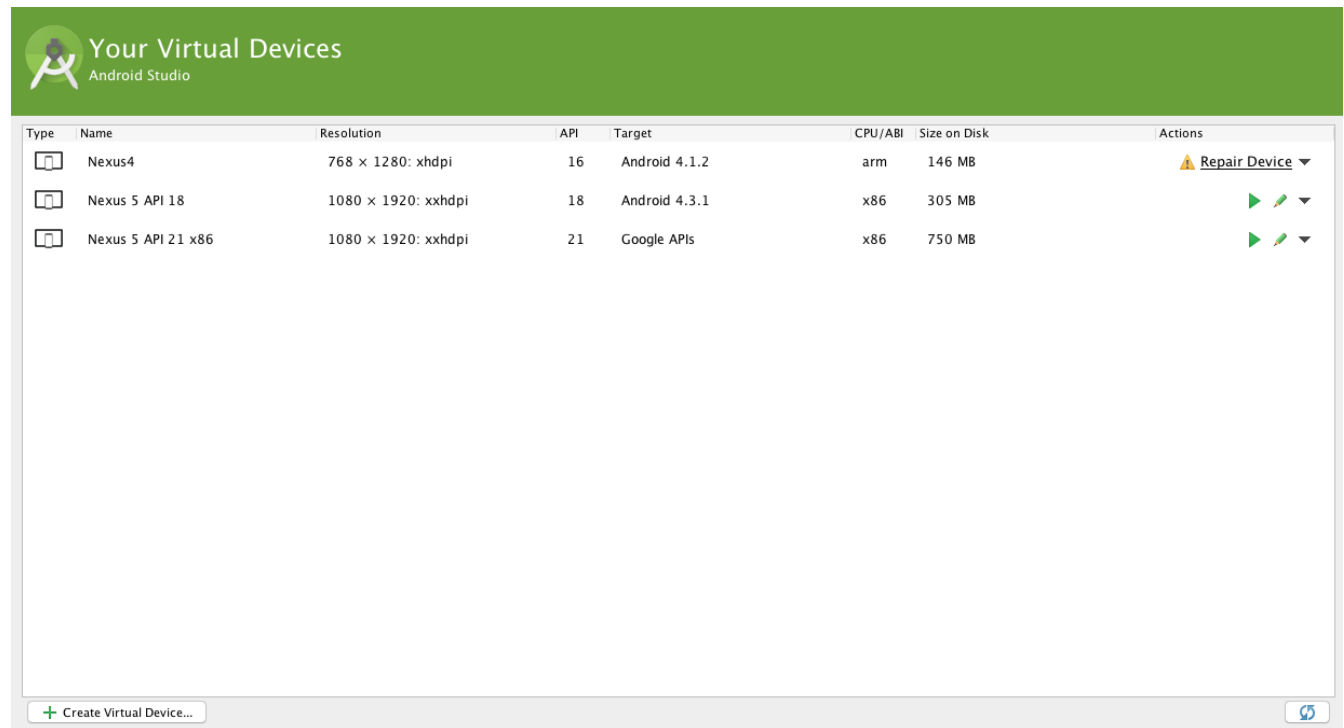
- Android's SDK includes a set of tools that facilitate the creation of applications for Android OS.
- SDK tools are classified into two groups:
  - SDK tools:
    - They are independent of the Android version.
    - They are updated regardless the platform tools.
    - They include:
      - Virtual Device Manager and emulator.
      - Development tools (graphical environment and console).
      - Debugging tools.
      - Building tools.
  - Platform tools:
    - They are specific for the Android version that applications are developed for.



# ◆◆◆ Android Simulator

## Android SDK – Virtual Device Manager (AVD)

- Graphical tool for the creation of virtual devices.
- It enables the creation of multiple devices with different features:
  - Operative system.
  - Hardware.



# ◆◆◆ Android Simulator

## Android SDK – Emulator

- It allows users to execute the devices created in the AVD Manager.
- It can be executed through the console (the emulator utility) or via Android Studio.
- It may be configured via parameters when being executed.
- It cannot perform calls.
- It may simulate the reception of calls and text messages.

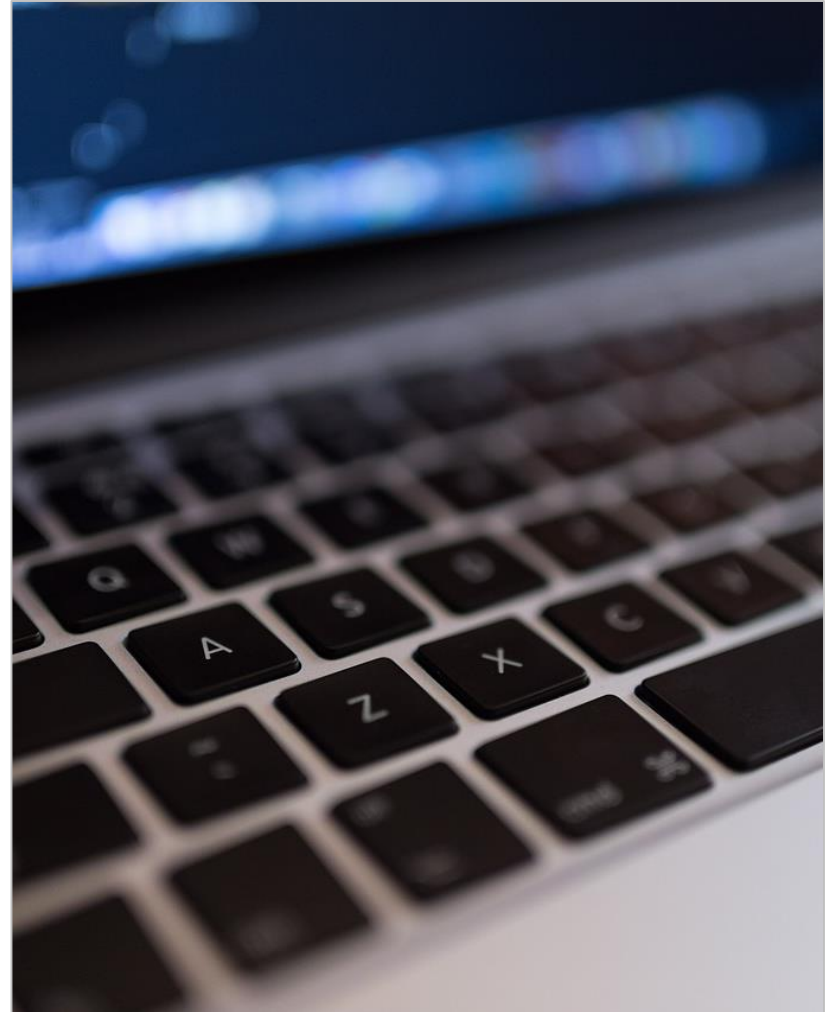


Android's Emulator

# ◆◆◆ Android Simulator

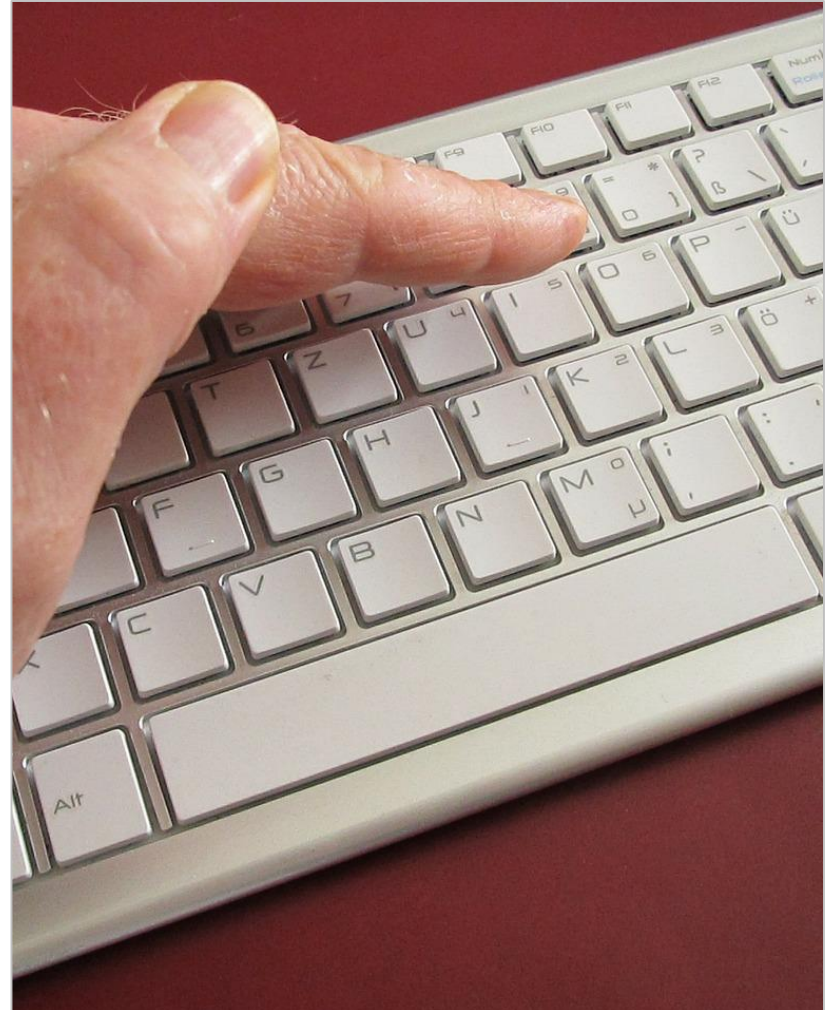
## Android SDK – Development Tools

- It has two types of tools:
  - Console Tools:
    - android: tool to manage the SDK via console.
    - lint: static analysis tool for the detection of errors and optimisations.
    - sqlite3: tool to access sqlite files that include application data.
  - Graphical tools:
    - SDK Manager: graphical interface to manage the installation of API versions, images of the system for the emulators, and documentation.



## Android SDK – Debugging Tools

- Tools for the debugging of applications.
- Android Monitor: graphical tool that allows users to measure the use of the system's resources (CPU, memory, network, etc.) by an application. It also enables access to the system logs.
- DDMS (Dalvik Debug Monitor Server): it allows users to take screenshots, obtain information regarding the device's memory, and simulate events in the device (calls, messages, location, etc.).



## Android SDK – Building Tools

- Tools for the building of binaries.
- ProGuard: it reduces, optimizes and obfuscates the code of an application. To this end, it eliminates the unused code and rename classes, attributes and methods with meaningless names.
- zipalign: it optimizes the compression of application files (apk) created to have a faster and less costly execution.





# ◆◆◆ Android Simulator

## Android SDK – Platform Tools

- Such tools are updated with each Android's SDK version; however, they remain compatible with older versions.
- adb: Android Debug Bridge (adb) is a console tool that enables the communication with an emulator instance executing, or a physical device connected. It also enables access to the emulator or device shell.
- logcat: it provides a mechanism to recover and visualize the device or emulator's logs.



## Android SDK – Android Studio

- Android Studio is the official IDE.
- It is based on an IDE developed by IntelliJ called IDEA.
- Among other characteristics, it provides:
  - Access to Android's SDK tools, through the IDE itself.
  - Code templates to add common functionality to applications.
  - Graphical edition of users' interfaces.
  - Automatic compilation and testing system based on Gradle.
    - It facilitates the inclusion of external libraries.
  - Applications signature and creation of APK to be distributed
  - Code obfuscation via Google ProGuard.
  - Support to integrate the developed applications with Google's services directly.
- <http://developer.android.com/tools/studio/index.html>

## Android Studio – Installation

### 1. Install Java's JDK 7.

- <http://www.oracle.com/technetwork/java/javase/downloads/jdk7-downloads-1880260.html>

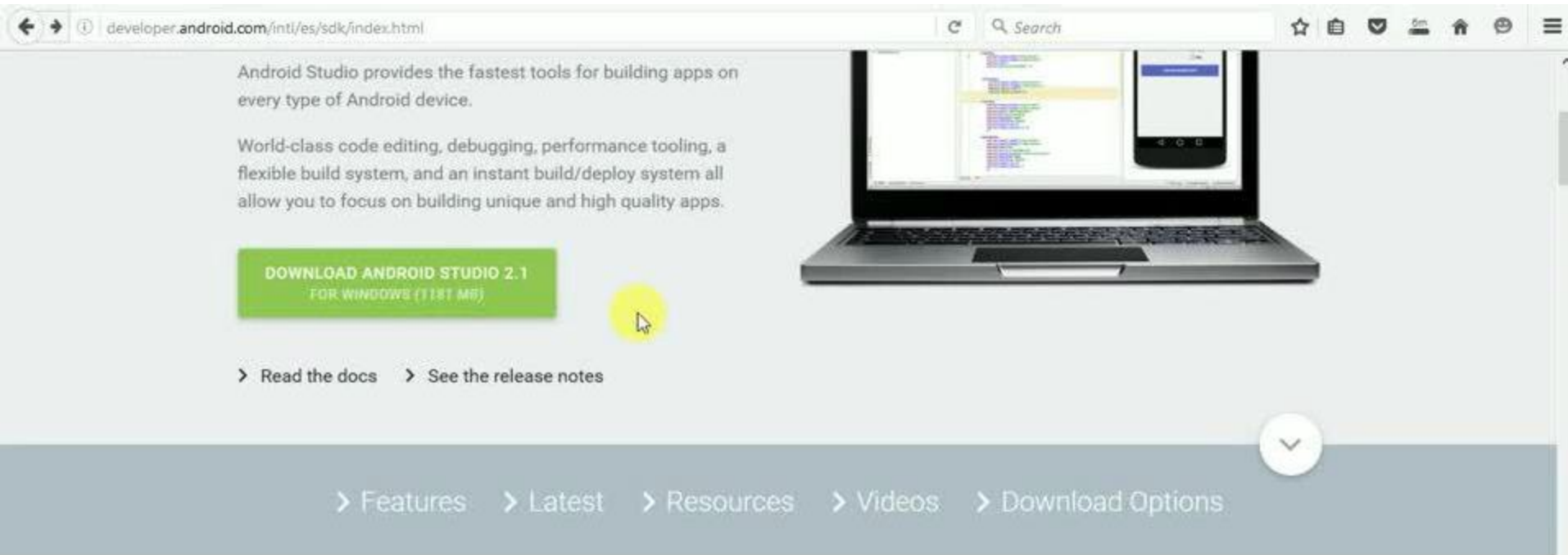
### 2. Navigate to the following website: <http://developer.android.com/sdk/index.html>

### 3. Download the version that corresponds to your operative system.

- If it does not appear automatically, you will find a table at the bottom of the web page.

Platform	Package	Size	SHA-1 Checksum
Windows	<a href="#">android-studio-bundle-141.2456560-windows.exe</a> (Recommended)	1209163328 bytes	6ffe608b1dd39041a578019eb3fedb5ee62ba545
	<a href="#">android-studio-ide-141.2456560-windows.exe</a> (No SDK tools included)	351419656 bytes	8d016b90bf04ebac6ce548b1976b0c8a4f46b5f9
	<a href="#">android-studio-ide-141.2456560-windows.zip</a>	375635150 bytes	64882fb967f960f2142de239200104cdc9b4c75b
Mac OS X	<a href="#">android-studio-ide-141.2456560-mac.dmg</a>	367456698 bytes	d0807423985757195ad5ae4717d580deeba1dbd8
Linux	<a href="#">android-studio-ide-141.2456560-linux.zip</a>	380943097 bytes	b8460a2197abe26979d88e3b01b3c8bfd80a37db

## Download Android Studio



A screenshot of the Android Studio download page. The browser address bar shows 'developer.android.com/intl/es/sdk/index.html'. The page content includes a description of Android Studio, a green download button, and a list of links at the bottom.

Android Studio provides the fastest tools for building apps on every type of Android device.

World-class code editing, debugging, performance tooling, a flexible build system, and an instant build/deploy system all allow you to focus on building unique and high quality apps.

**DOWNLOAD ANDROID STUDIO 2.1**  
FOR WINDOWS (1181 MB)

[Read the docs](#) [See the release notes](#)

[Features](#) [Latest](#) [Resources](#) [Videos](#) [Download Options](#)

## Instant Run

Push code and resource changes to your app running on a device or emulator and see the changes instantly come to life.



## Android Studio – Installation

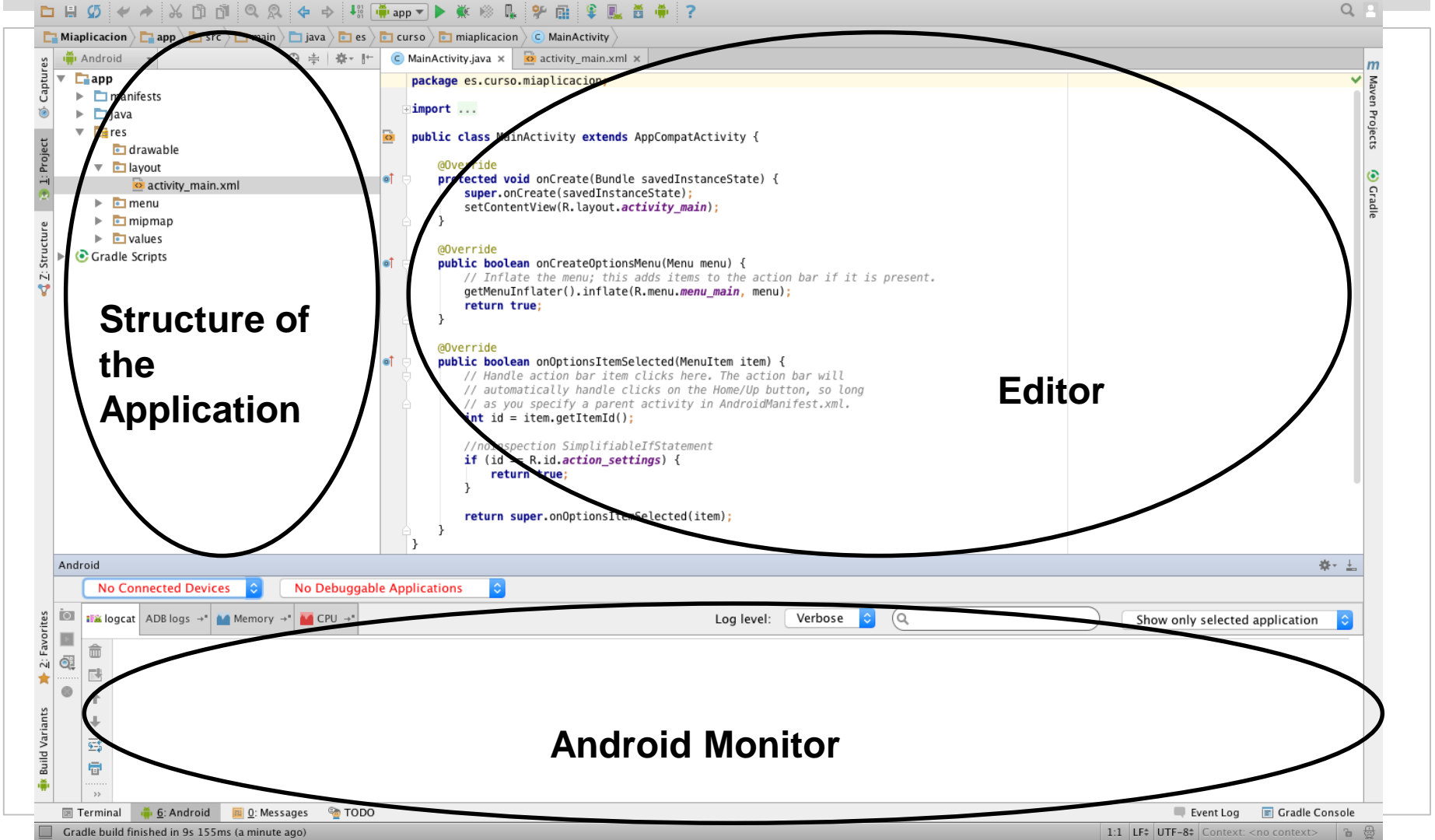
4. Accept the software license conditions.
5. Execute Android Studio.



6. Follow the instructions to install the SDK for the latest version.

# ◆◆◆ Android Simulator

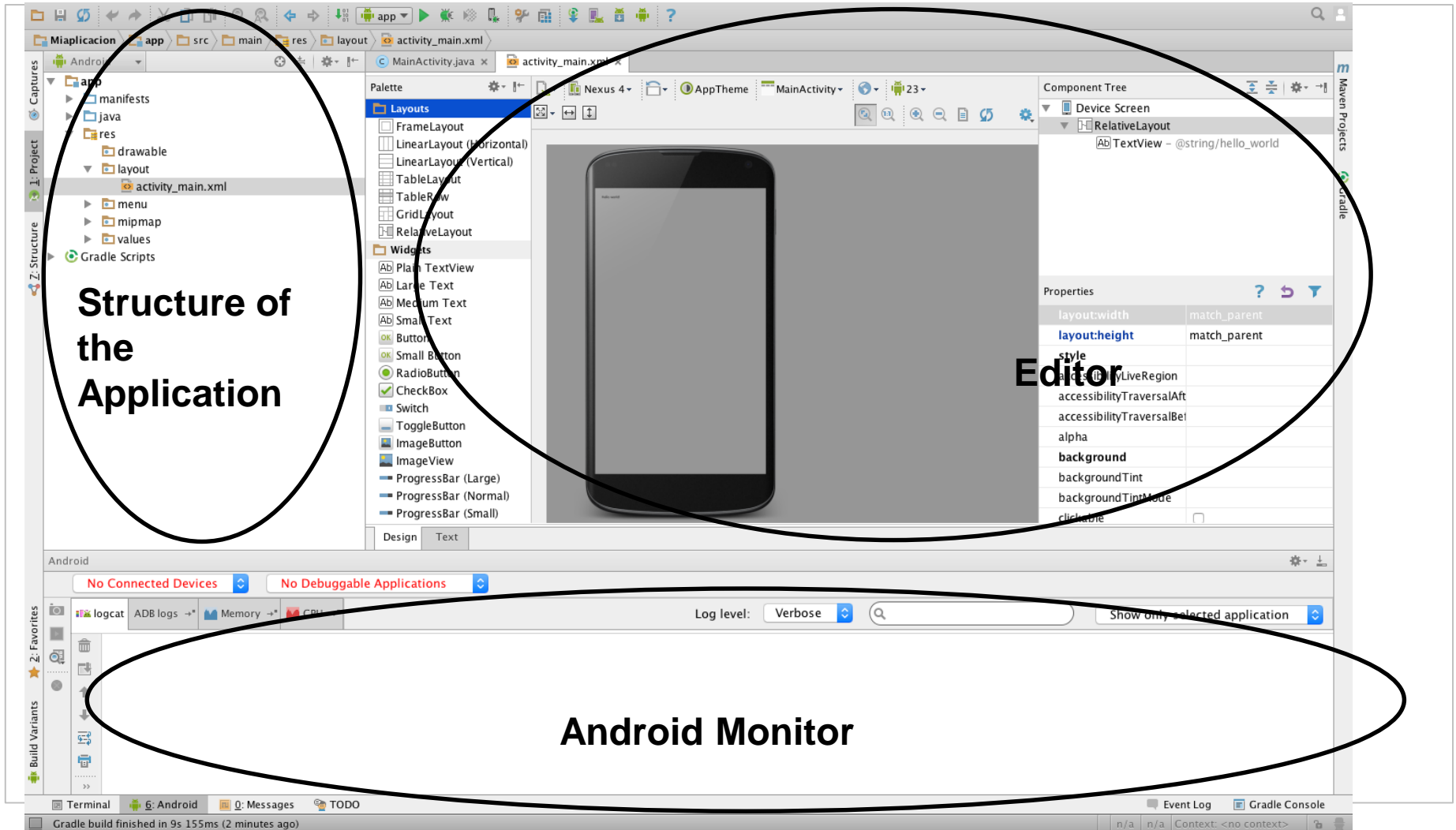
## Tools – Android Studio





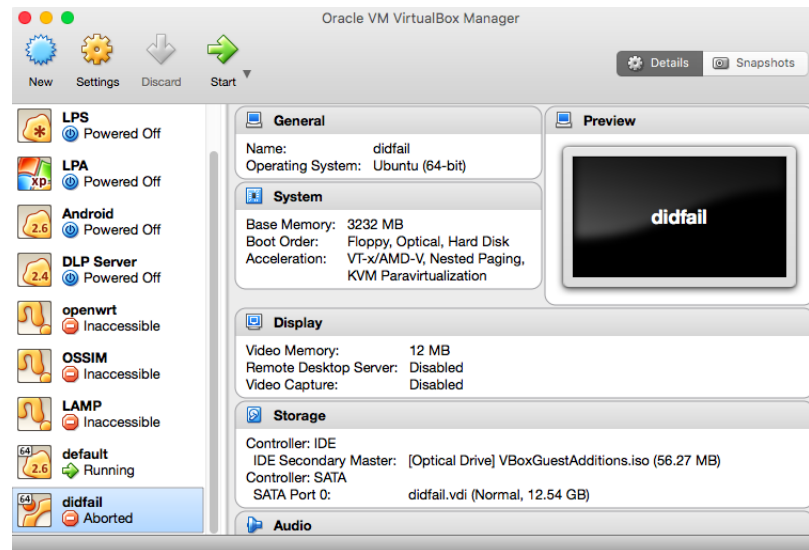
# ◆◆◆ Android Simulator

## Tools – Android Studio



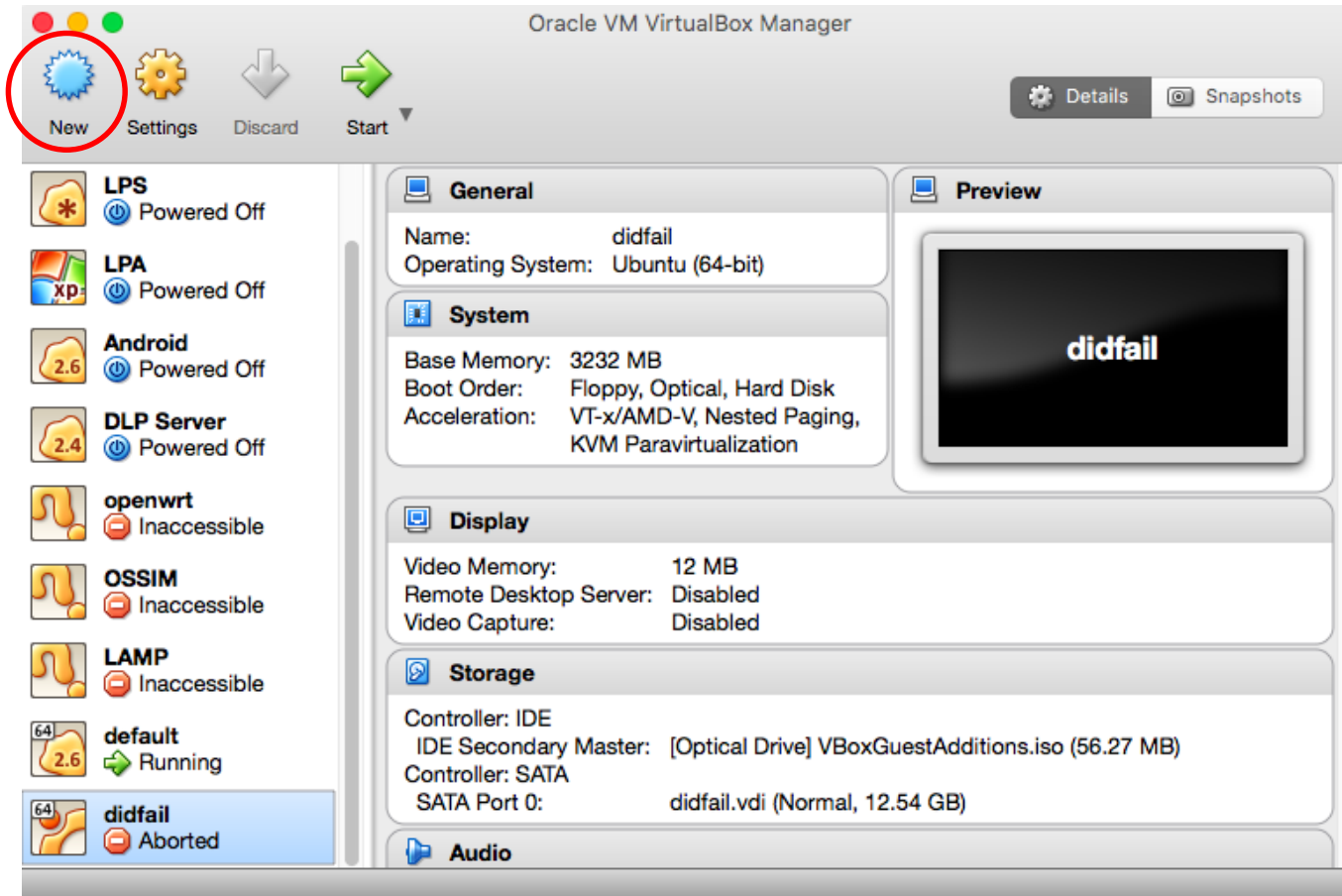
## Santoku Linux – Installation

- Linux distribution specialised in security analysis of mobile devices.
  - Available at: <http://santoku-linux.com/download/>
1. Install VirtualBox, in case it is not installed yet (<https://www.virtualbox.org>).
  2. Download the ISO image of Santoku Linux.
  3. Start the application.



## Santoku Linux – Installation

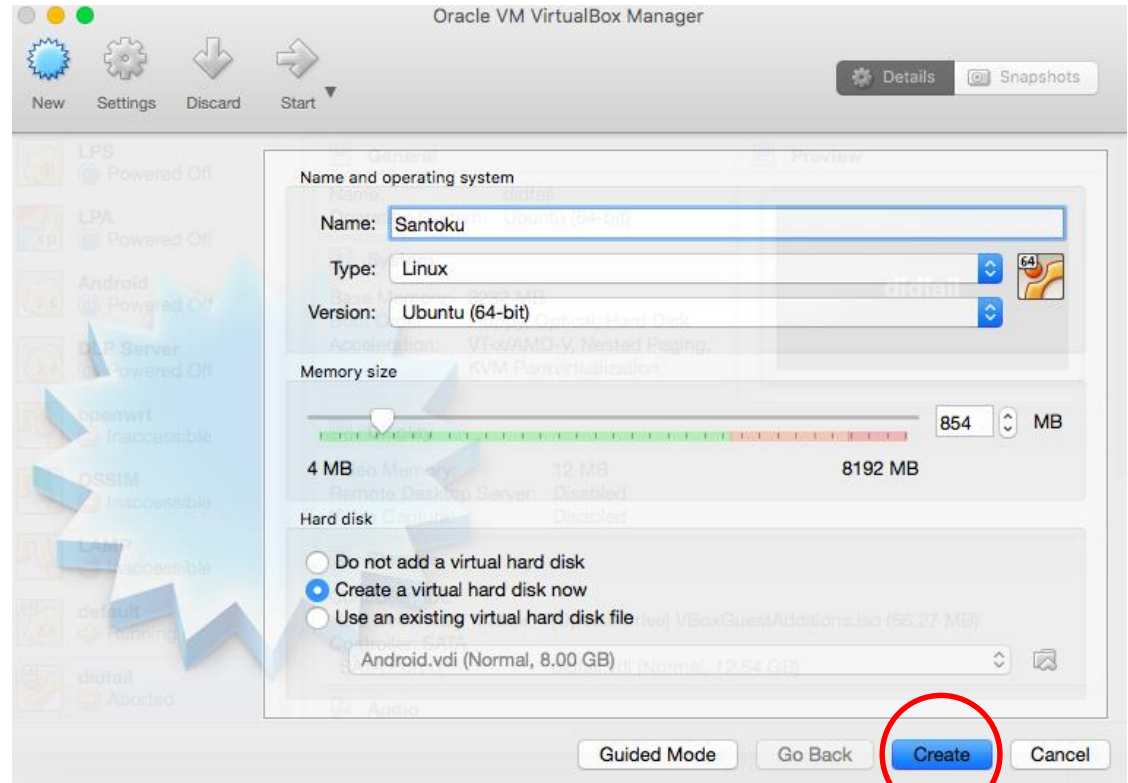
### 4. Create a new virtual machine.



## Santoku Linux – Installation

### 5. Create a new machine.

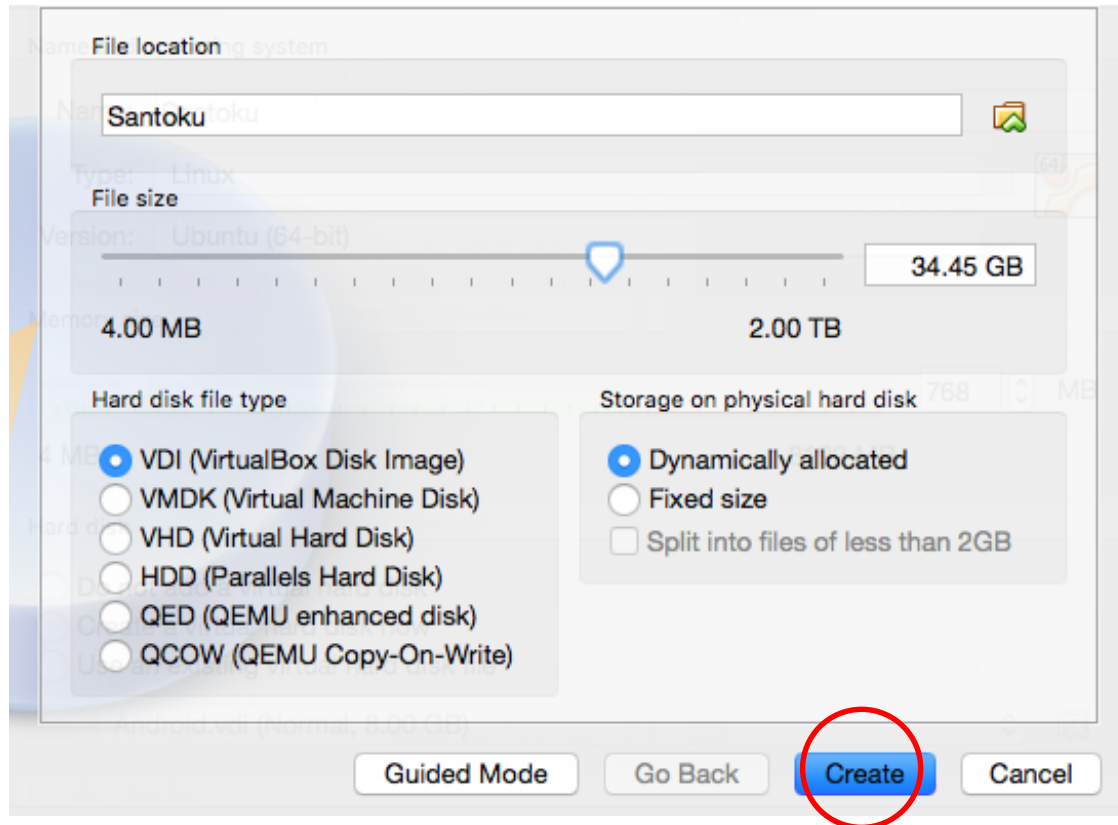
- Type: Linux.
- Version: Ubuntu 64.
- RAM: 2GB Min.



## Santoku Linux – Installation

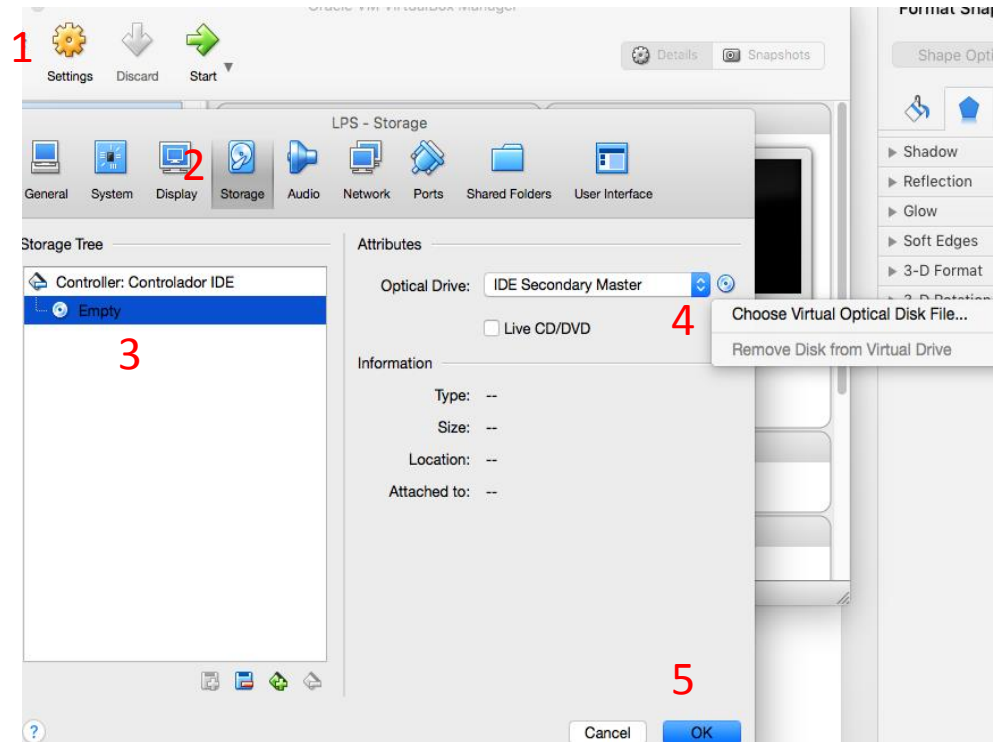
### 6. Create a virtual hard disk.

- It is advisable that the size of the disk is 40GB.



## Santoku Linux – Installation

- 7. Once it has been created, select the machine and click:
  - Settings -> Storage -> Optical Drive -> Choose Virtual Optical Disk File.
  - Select the downloaded ISO.





## VM Creation

The screenshot displays the Oracle VM VirtualBox Administrator interface. On the left, a list of VMs includes 'DWL', 'Win7', and 'Santoku'. The 'Santoku' VM is selected, and its configuration is shown in the main pane. The configuration is organized into several tabs: General, Sistema, Pantalla, Almacenamiento, Audio, Red, USB, Carpetas compartidas, and Descripción. The 'General' tab is active, showing the VM's name as 'Santoku', its operating system as 'Linux (64-bit)', and its memory size as 3072 MB. The 'Sistema' tab shows the memory size as 3072 MB and the order of disks as 'Disquete, Control, Disco duro'. The 'Pantalla' tab shows the video memory as 12 MB and the video capture as disabled. The 'Almacenamiento' tab shows the controller as IDE, the master disk as 'Santoku.vdi (Normal, 40.00 GB)', and the SATA controller as 'Santoku.vdi (Normal, 40.00 GB)'. The 'Audio' tab shows the controller as 'Windows DirectSound' and the driver as 'ICH AC97'. The 'Red' tab shows the adapter as 'Intel PRO/1000 MT Desktop'. The 'USB' tab shows the controller as 'OHCI' and the number of devices as 0. The 'Carpetas compartidas' and 'Descripción' tabs show 'Ninguno' (None).

**General**

Nombre: Santoku  
Sistema operativo: Linux (64-bit)

**Sistema**

Memoria base: 3072 MB  
Orden de discos: Disquete, Control, Disco duro  
Aceleración: VT-x/AMD-V, Paginación anidada, Paravirtualización KVM

**Pantalla**

Memoria de vídeo: 12 MB  
Servidor de escritorio remoto: Inhabilitado  
Captura de vídeo: Inhabilitado

**Almacenamiento**

Controlador: IDE  
Disco secundario maestro: [Unidad óptica] VBoxGuestAdditions.iso (55,46 MB)  
Controlador: SATA  
Puerto SATA 01: Santoku.vdi (Normal, 40,00 GB)

**Audio**

Controlador de interfaz: Windows DirectSound  
Controlador: ICH AC97

**Red**

Adaptador 1: Intel PRO/1000 MT Desktop (Adaptador puente, «Realtek PCIe GBE Family Controller»)

**USB**

Controlador USB: OHCI  
Número de dispositivos: 0 (0 activo)

**Carpetas compartidas**

Ninguno

**Descripción**

Ninguno

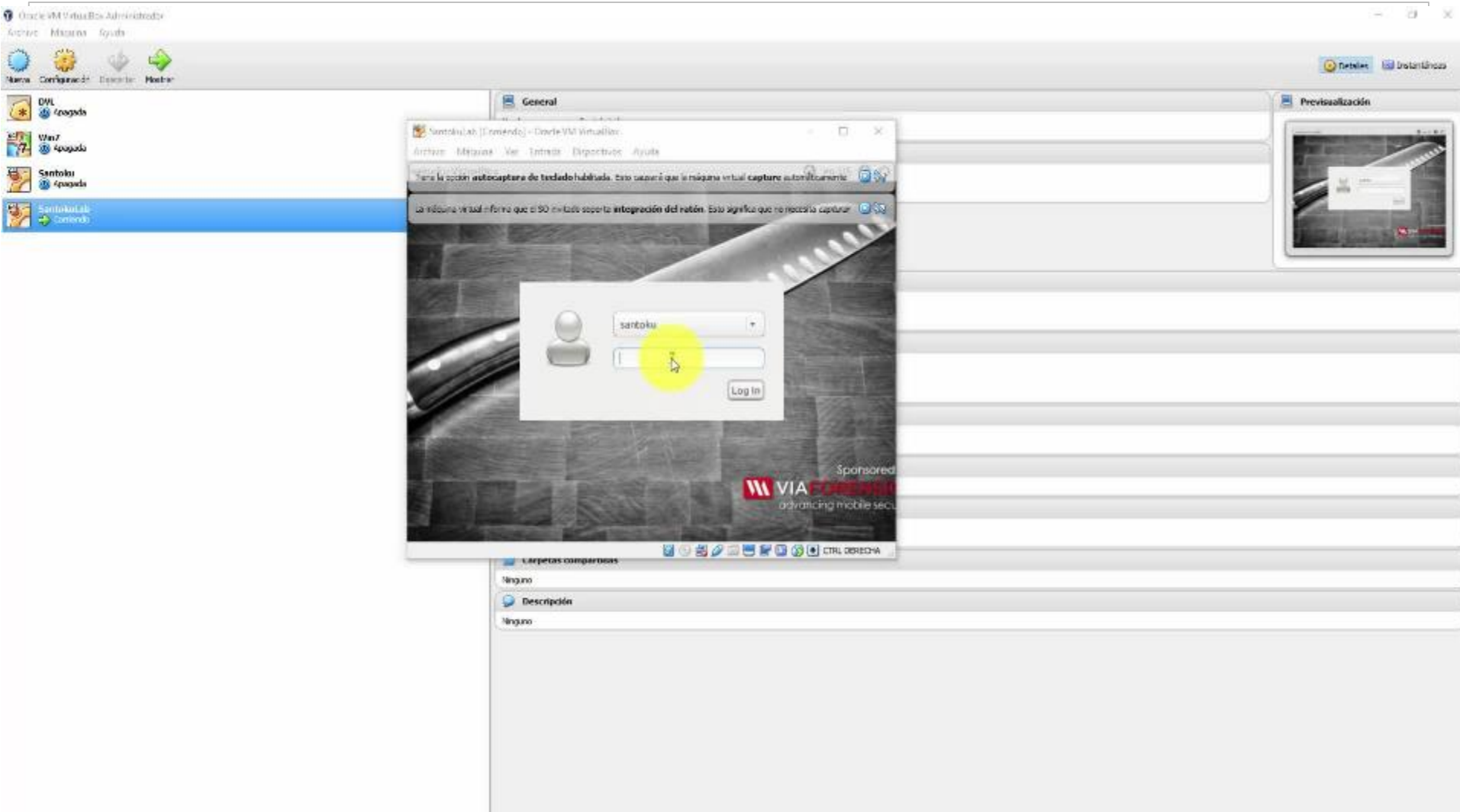
**Previsualización**

Santoku

## Santoku Linux – Installation

8. Start the virtual machine and install the operative system.
9. When starting the machine, select the “Install” option.
10. Select “Erase disk and install Santoku”.
11. Restart it once the installation has finished.
12. When the installation has been restarted, in the upper bar of the application, select:
  - Devices -> Insert Guest Additions CD.
  - If the system requires the installation of “Guest Additions”, accept.
  - If it does not start automatically:
    - Navigate to the CD directory.
    - Execute (the administrator’s password has to be written).
    - `sudo sh VBoxLinuxAdditions.run`

## Installation of Guest Additions



# Implementation

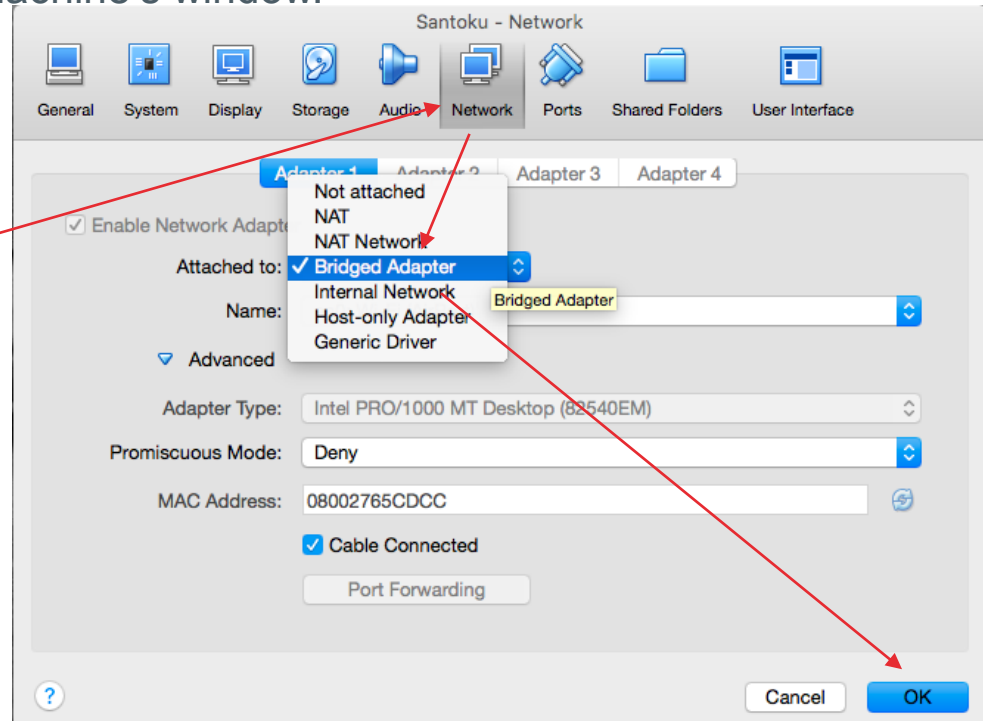
The screenshot displays the Oracle VM VirtualBox Administrator interface. On the left, a list of virtual machines is shown, with 'SantokuLab' selected. The main pane shows the configuration for this VM, categorized into several sections:

- General:**
  - Nombre: SantokuLab
  - Sistema operativo: Linux (64-bit)
- Sistema:**
  - Memoria base: 2027 MB
  - Orden de arranque: Disco duro, CD/DVD, Disquete
  - Asociación: VT-x/AMD-V, Hibernación anidada, Paravirtualización IOM
- Pantalla:**
  - Memoria de vídeo: 12 MB
  - Servidor de escritorio remoto: Inhabilitado
  - Captura de vídeo: Inhabilitado
- Almacenamiento:**
  - Controlador: IDE
  - Disco secundario maestro: [Unidad óptica] Santoku\_0.S.iso (2,37 GB)
  - Controlador SATA: [Unidad óptica] Santoku\_0.S.iso (2,37 GB)
  - Disco SATA 0: SantokuLab.vdi (Normal, 37,99 GB)
- Audio:**
  - Controlador de audio: Windows DirectSound
  - Controlador: ICH AC97
- Red:**
  - Adaptador 1: Intel PRO/1000 MT Desktop (Bridged)
- USB:**
  - Controlador (USB): OHCI
  - Filtros de dispositivos: 0 (0 activo)
- Carpetas compartidas:**
  - Ninguna
- Descripción:**
  - Ninguna

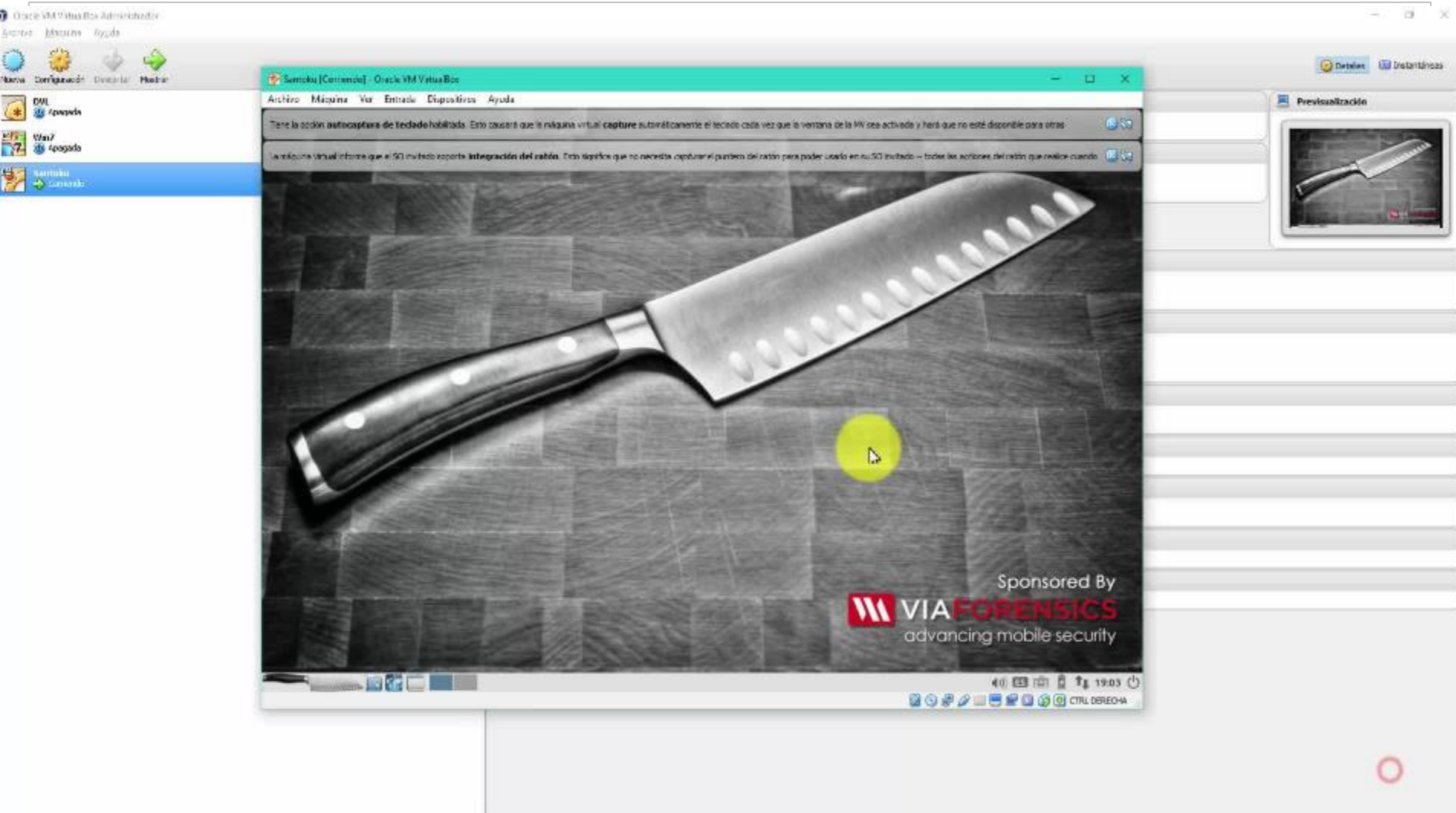
On the right side of the configuration pane, there is a 'Previsualización' (Preview) window showing a black screen with the text 'SantokuLab' in white.

## Santoku Linux – Installation

- For some of such tasks Santoku Linux will be required to have direct access to the physical network.
- To this end, it is necessary to configure VirtualBox with a bridged-like interface.
  - At the bottom left of the virtual machine's window.



## Modification of the Network Configuration

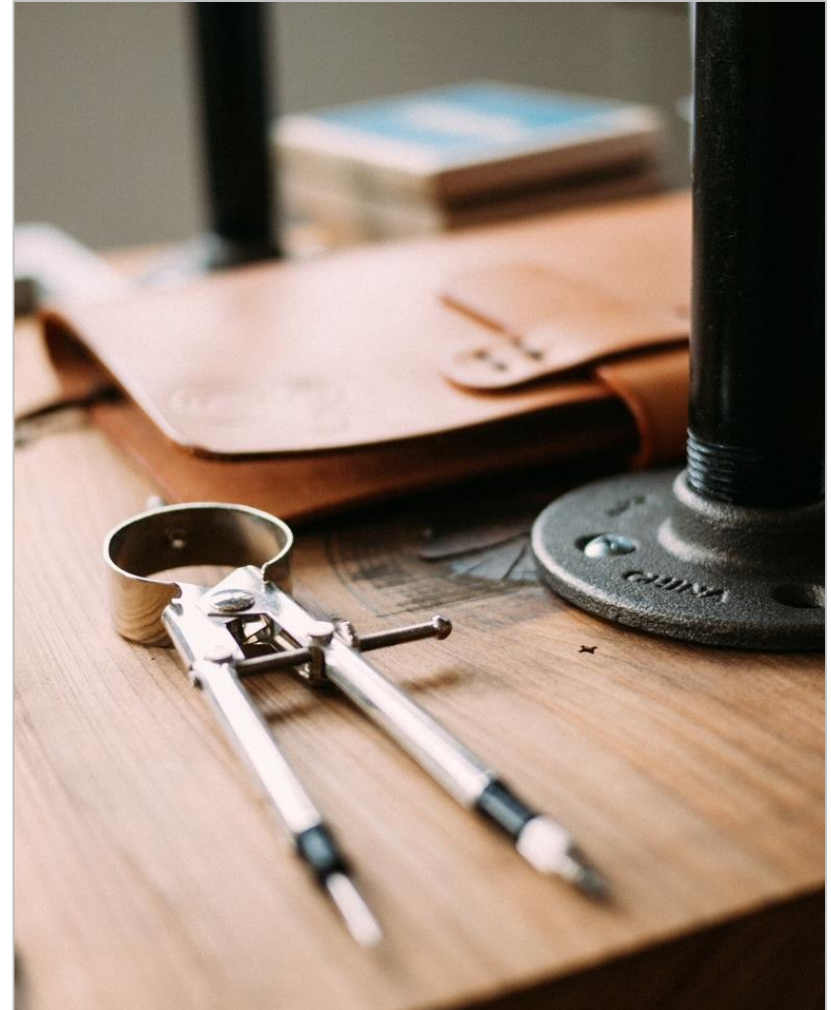




# ◆◆◆ Android Simulator

## Tools – Santoku Linux

- Santoku includes several tools that may be useful to conduct dynamic and static analysis.
- In this case, we will use the following tools:
  - Apktool.
  - Androguard.
  - Wireshark.
  - Burp Proxy.



## Tools – Apktool

- Tool to unpack Android applications.
- It allows users to obtain a folder that contains all the resources of the application, including:
  - The manifest file in a legible format.
  - The source code in smali format.
  - Images and other resources used by the application.
- It also enables the repackaging of files into an apk file in order to be executed.
  - The files created need to be signed with a certificate in order to be executed in a device.
- Santoku includes the 1.5 version of apktool.
- It is necessary to update it in order to unpack recent applications.

## Tools – Apktool – Updating

- Download the most recent “jar” file from the apktool web.
  - <https://bitbucket.org/iBotPeaches/apktool/downloads>
- Copy it to `/usr/share/apktool` and rename it as `apktool.jar`.  
> `sudo mv path_apktool/apktool_version.jar usr/share/apktool.jar`

```
$ apktool
Apktool v2.0.0-RC2 - a tool for reengineering Android apk files
with smali v2.0.3 and baksmali v2.0.3
Copyright 2010 Ryszard Wiśniewski <brut.all@gmail.com>
Updated by Connor Tumbleson <connor.tumbleson@gmail.com>

usage: apktool
  -advance,--advanced  prints advance information.
  -version,--version   prints the version then exits
usage: apktool if|install-framework [options] <framework.apk>
  -p,--frame-path <dir>  Stores framework files into <dir>.
  -t,--tag <tag>         Tag frameworks using <tag>.
usage: apktool d[ecode] [options] <file_apk>
  -f,--force           Force delete destination directory.
  -o,--output <dir>    The name of folder that gets written. Default is apk.out
t
  -p,--frame-path <dir>  Uses framework files located in <dir>.
  -r,--no-res          Do not decode resources.
  -s,--no-src          Do not decode sources.
  -t,--frame-tag <tag>  Uses framework files tagged by <tag>.
usage: apktool b[uild] [options] <app_path>
  -f,--force-all      Skip changes detection and build all files.
  -o,--output <dir>    The name of apk that gets written. Default is dist/name
.apk
```

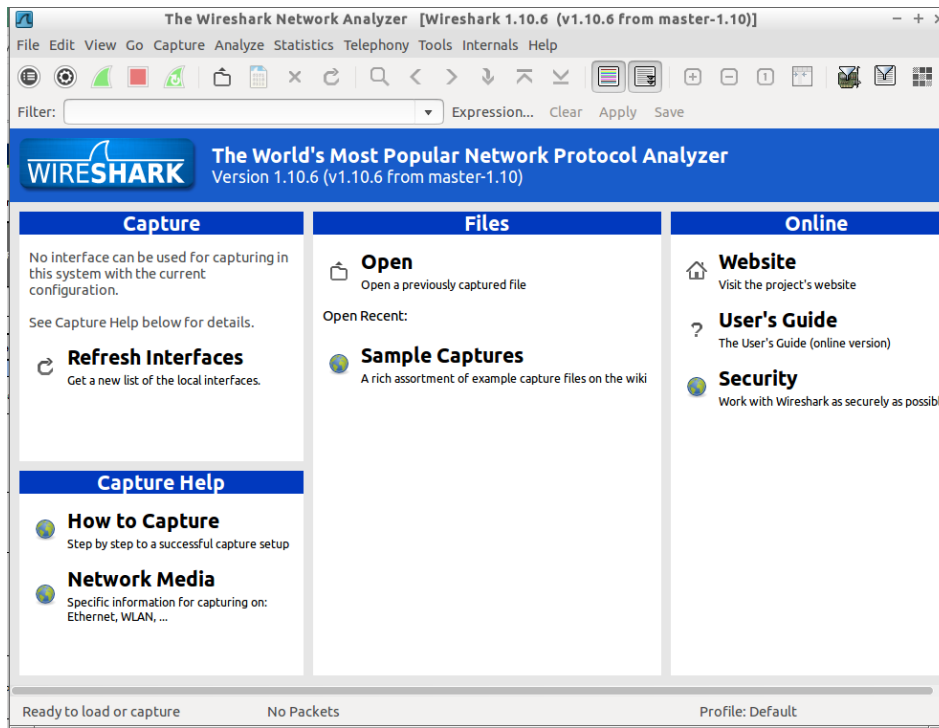
## Tools – Androguard

- It is an open source project for the analysis of Android applications.
- It is able to access multiple elements of an application:
  - Code: classes, methods, instructions, etc.
  - Manifest: permissions, activities, services, etc.
  - Resources: images, database files.
- It allows users to obtain representations of the flow of an application.
  - Smali (legible representation of the assembly language of Android).
  - Call graphs.
- It is easily extendible.
- In Santoku, it is possible to open its console through the device.  
`> androlyze -s`

# ◆◆◆ Android Simulator

## Tools – Wireshark

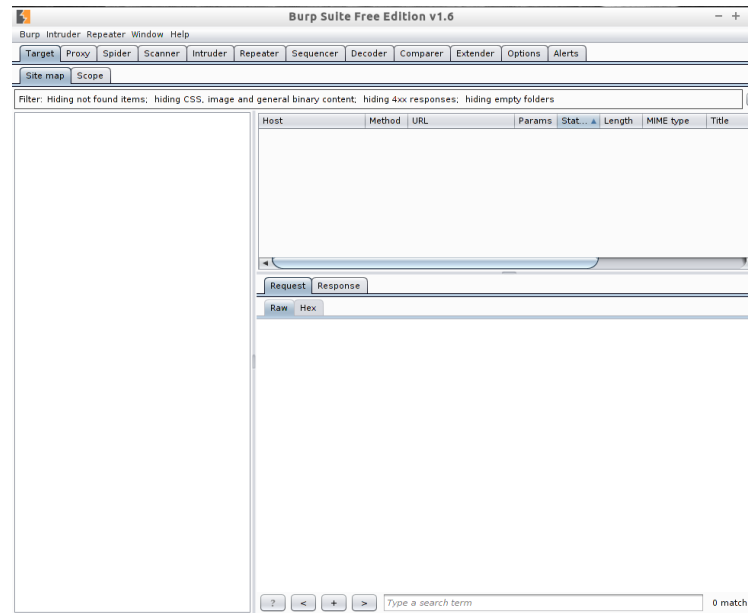
- It is a program for the capture and inspection of the network traffic.
- It is installed by default in Santoku.
  - > wireshark



## Tools – Burp Suite Free Proxy

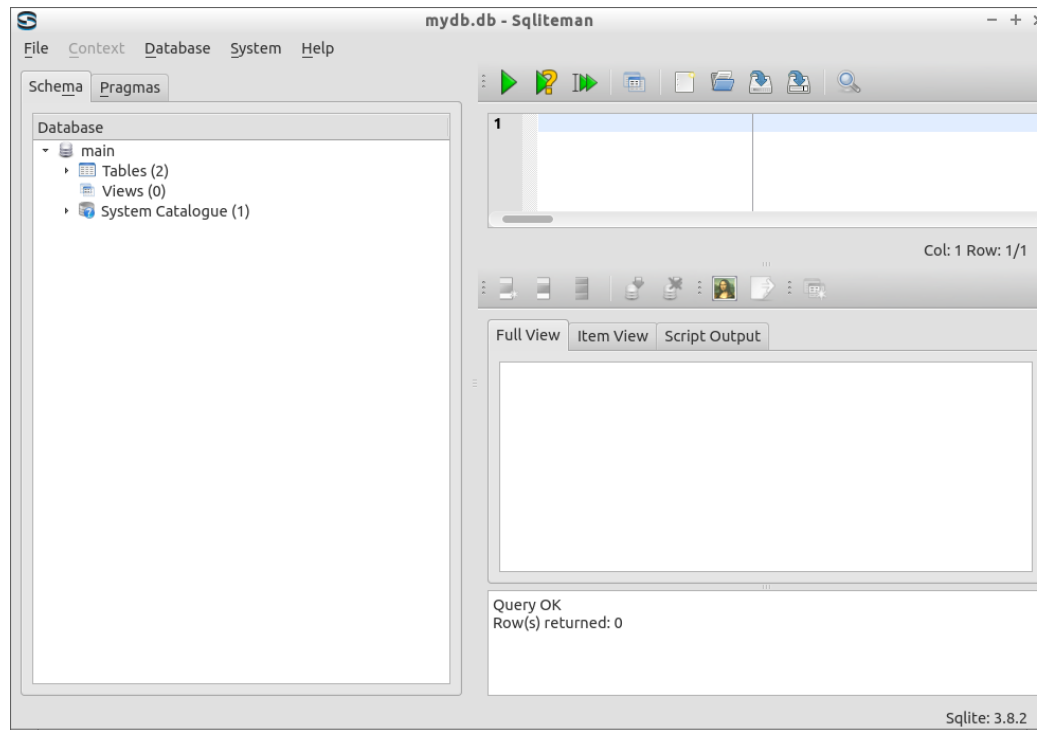
- It is a web proxy that will be used to intercept connections to web servers.
- It allows users to capture and modify non-encrypted traffic, but also traffic encrypted with SSL.
  - By creating an SSL certificate.
- It enables the modification of HTTP requests or responses.

```
> burpsuite
```



## Tools – Sqliteman

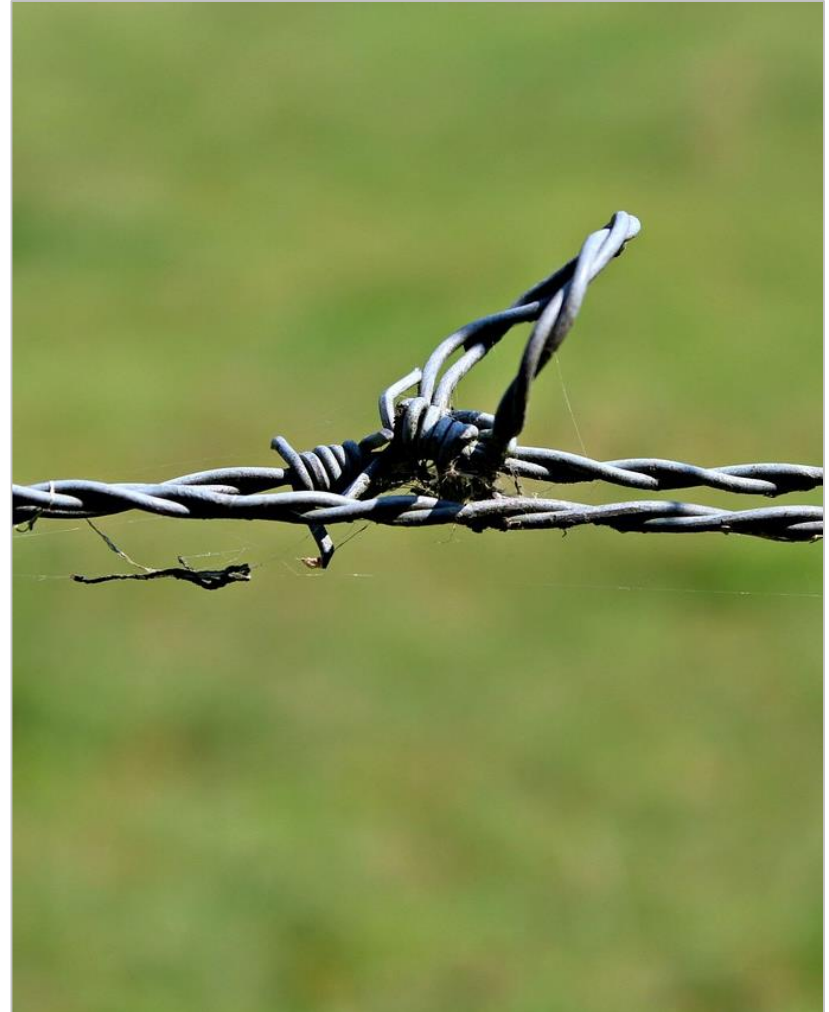
- Application for the inspection and modification of sqlite files.
- It is available in Santoku.
  - > sqliteman





## Tools – Qark

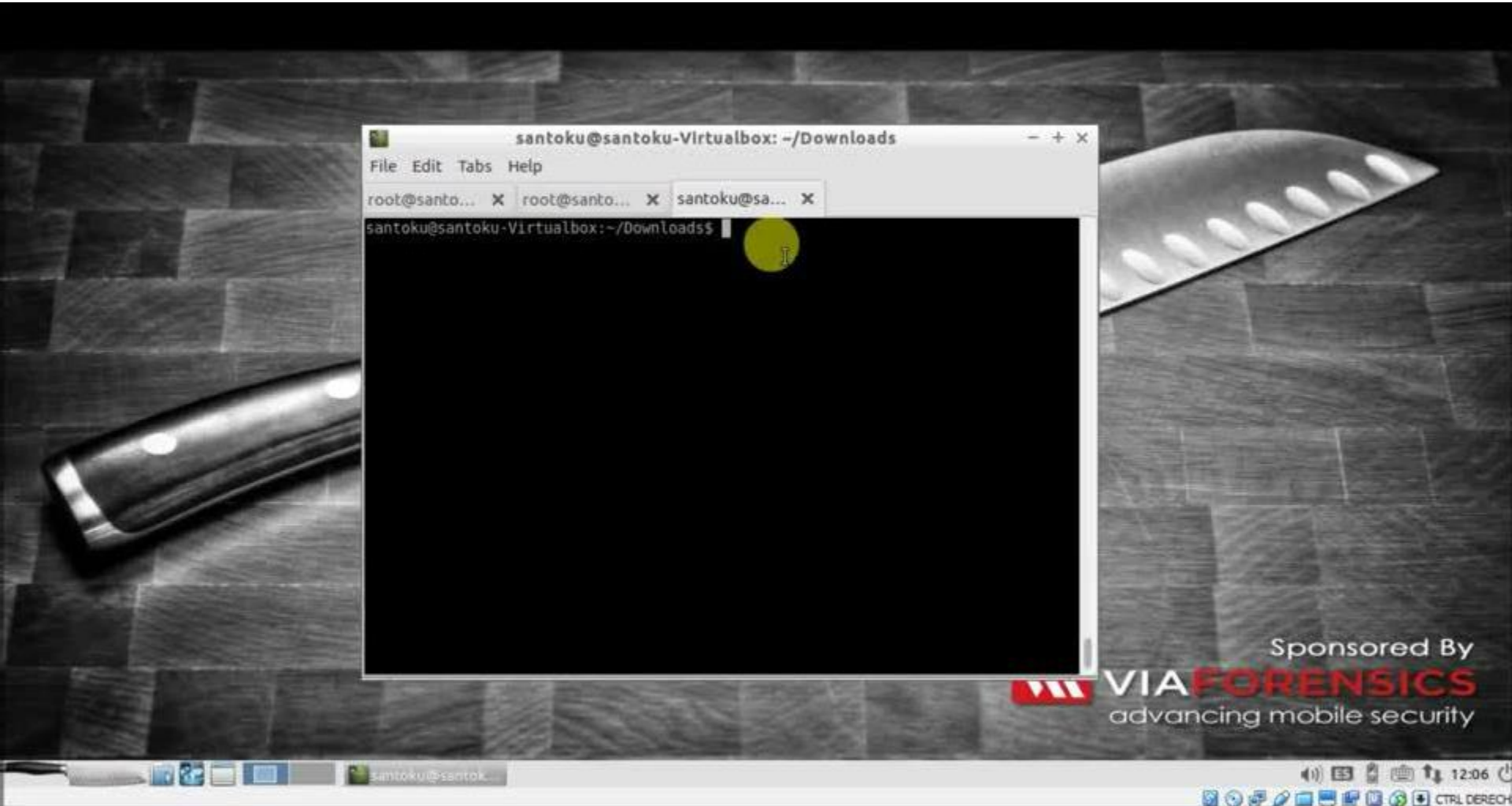
- Qark is a tool developed by LinkedIn that performs an automatic analysis of APK files.
- It shows some of the possible vulnerabilities that may affect the application and is also able to create exploit tools to demonstrate them.
- Qark is available in Github:  
<https://github.com/linkedin/qark>
- However, it is not included in Santoku Linux by default.
- In the following slides, the process required to install and execute this tool is described.



## Tools – Qark – Installation

- Create a folder called Qark in Santoku:
  - > mkdir qark
  - > cd qark
- Clone the Github repository in the Qark directory:
  - > git clone <https://github.com/linkedin/qark.git>
- Access the qark folder and execute the following script:
  - > cd qark
  - > python qark.py
- The user will be asked to download Android SDK.
- Even if it is already installed, accept it not to interfere with the installation.

## Installation of Qark



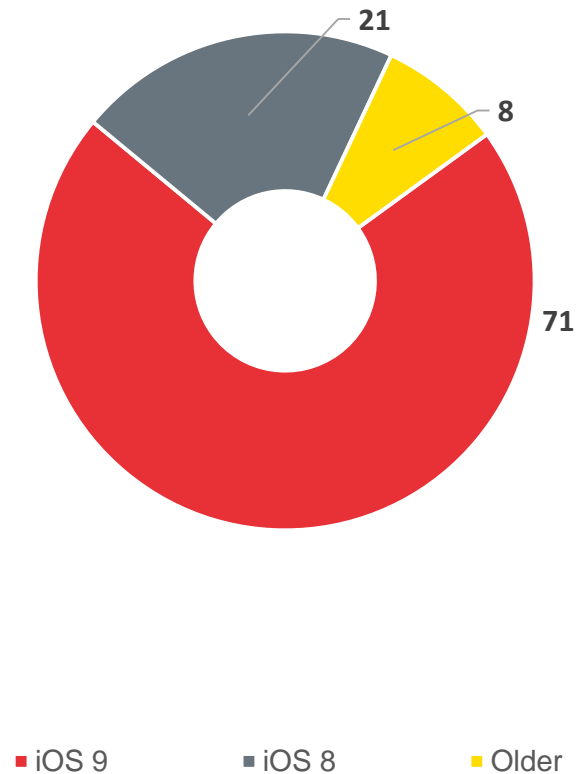
# iOS Simulator



## Introduction

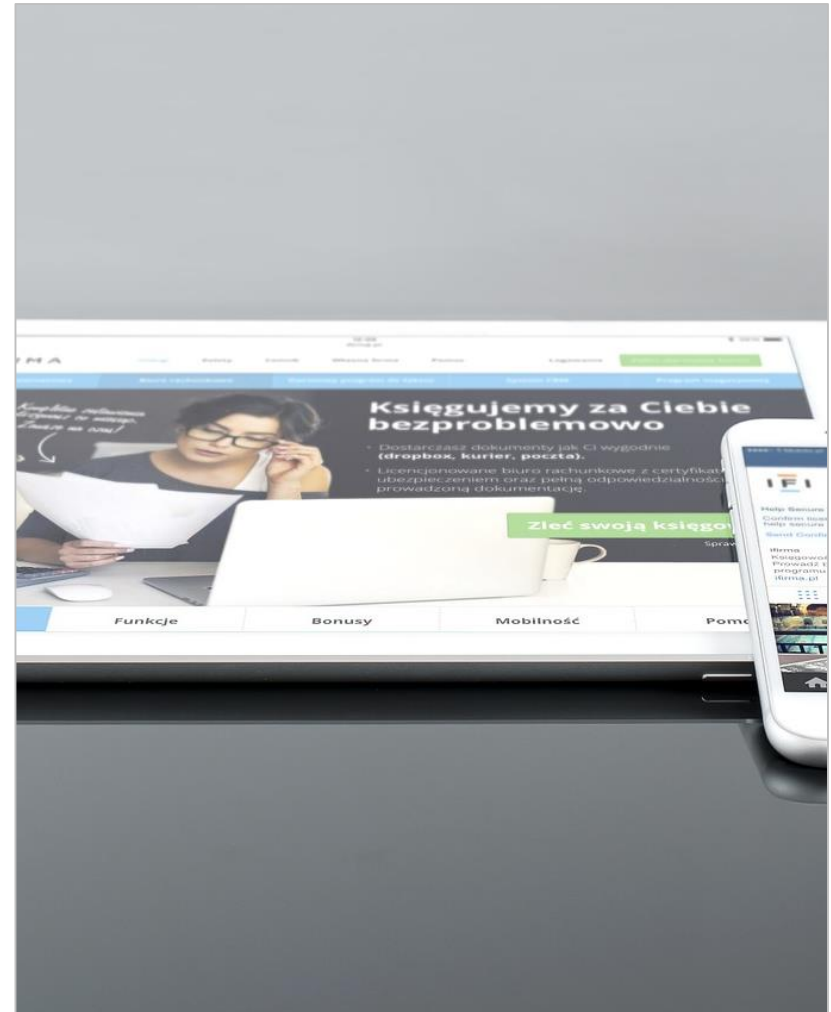
- It is Apple's operative system for iPhone, iPad and iPod Touch devices.
- It was launched in 2007, and only enabled the execution of applications through the browser.
- It is a closed system and it is only possible to install applications (previously approved by Apple, except for some exceptions) from the official website.

Distribution of versions



## Testing Laboratory I

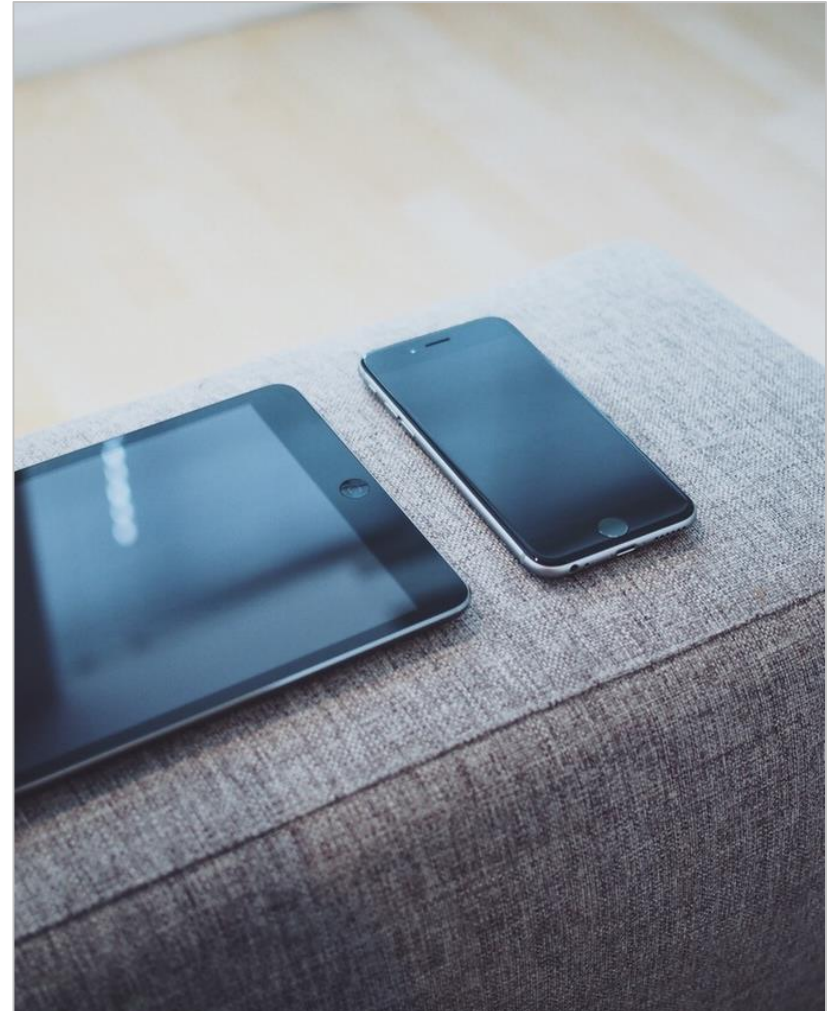
- The testing environment for iOS operative system of this course allows users to:
  - Develop applications.
  - Install and execute applications in devices.
  - Test different features of an application during its execution.
  - Static analysis (limited in non-jailbroken devices).
  - Dynamic analysis (limited in non-jailbroken devices).
  - Forensic analysis (limited in non-jailbroken devices).





## Testing Laboratory II

- The testing environment for iOS operative system of this course includes:
  - Apple Official Development Kit
  - Third parties' tools for the static analysis.
  - Third parties' tools for the dynamic analysis.
  - iOS applications to learn about vulnerabilities, secure programming and security services of Android.
  - Tools for the extraction of information from an iOS device.





## Apple Official Development Kit

- It is a set of tools that allow users to develop, install, execute, debug and distribute applications for iOS.
- Apple's official development kit enables the creation of applications for iOS and Apple Watch.
- The programming language used may be Objective-C or Swift.
- Applications created may be backward compatible with some versions:
  - Apple may impose restrictions on the available versions for the new applications.
- Applications uploaded to the App Store should be approved by Apple.
  - Two exceptions:
    - Installation through a business program.
    - Installation through the official IDE without a developer account.
      - From the last version.

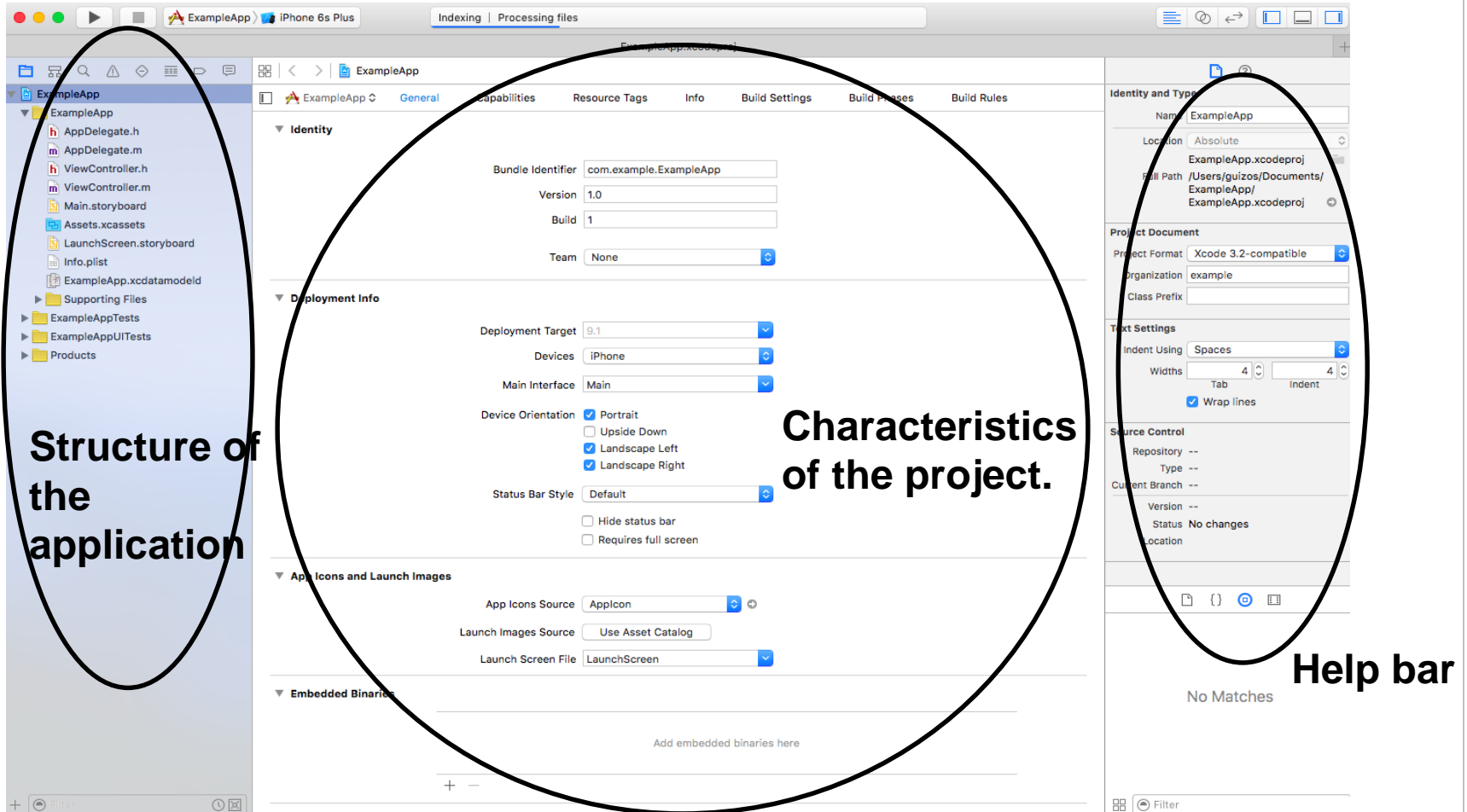
## Components of the Apple Official Development Kit

- This kit includes the following tools:
  - Integrated Development Environment (IDE):
    - Graphical environment for the creation and debugging of applications.
    - XCode 7 (December 2015).
      - Available at the App Store.
  - Software Development Kit (SDK) Tools:
    - Tools that allow users to compile and debug iOS applications.
    - The IDE automatizes its use.
    - It includes:
      - Tools for compilation, debugging and communication with devices.
      - System libraries to be used by third parties' applications.
      - Emulator to execute and debug applications.
      - SDK management tools.

## Tools – XCode

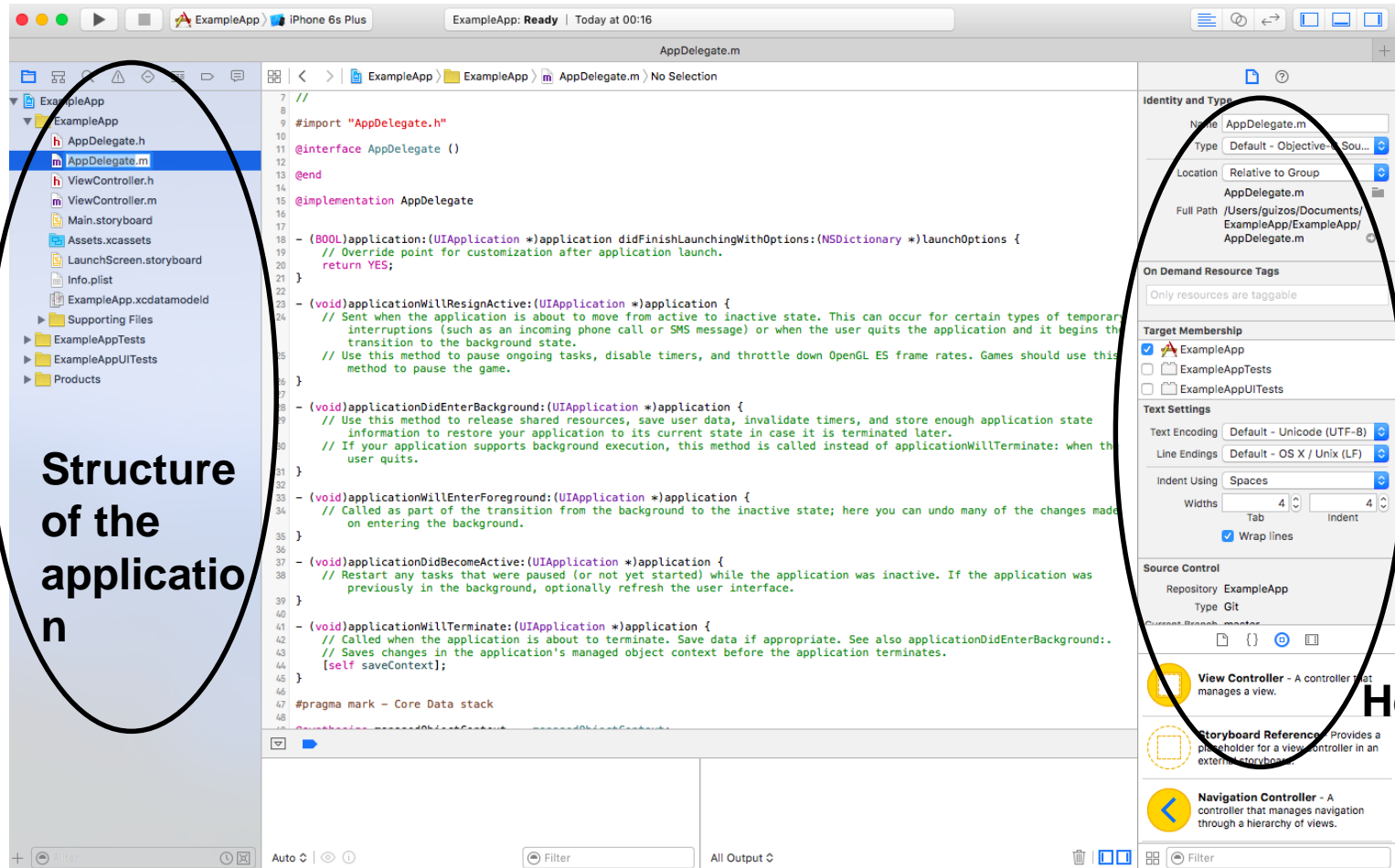
- Apple's IDE for the development of applications.
- It is obtained from Apple's Mac App Store.
- It allows users to develop applications for:
  - iOS.
  - Apple Watch.
  - OSX
- Among other characteristics, it provides:
  - Code templates to add common functionality to applications.
  - Graphical edition of users' interfaces.
  - Libraries of recurrent pieces of code.
  - Signature of applications.
  - iOS and Apple Watch simulator.

## Tools – XCode

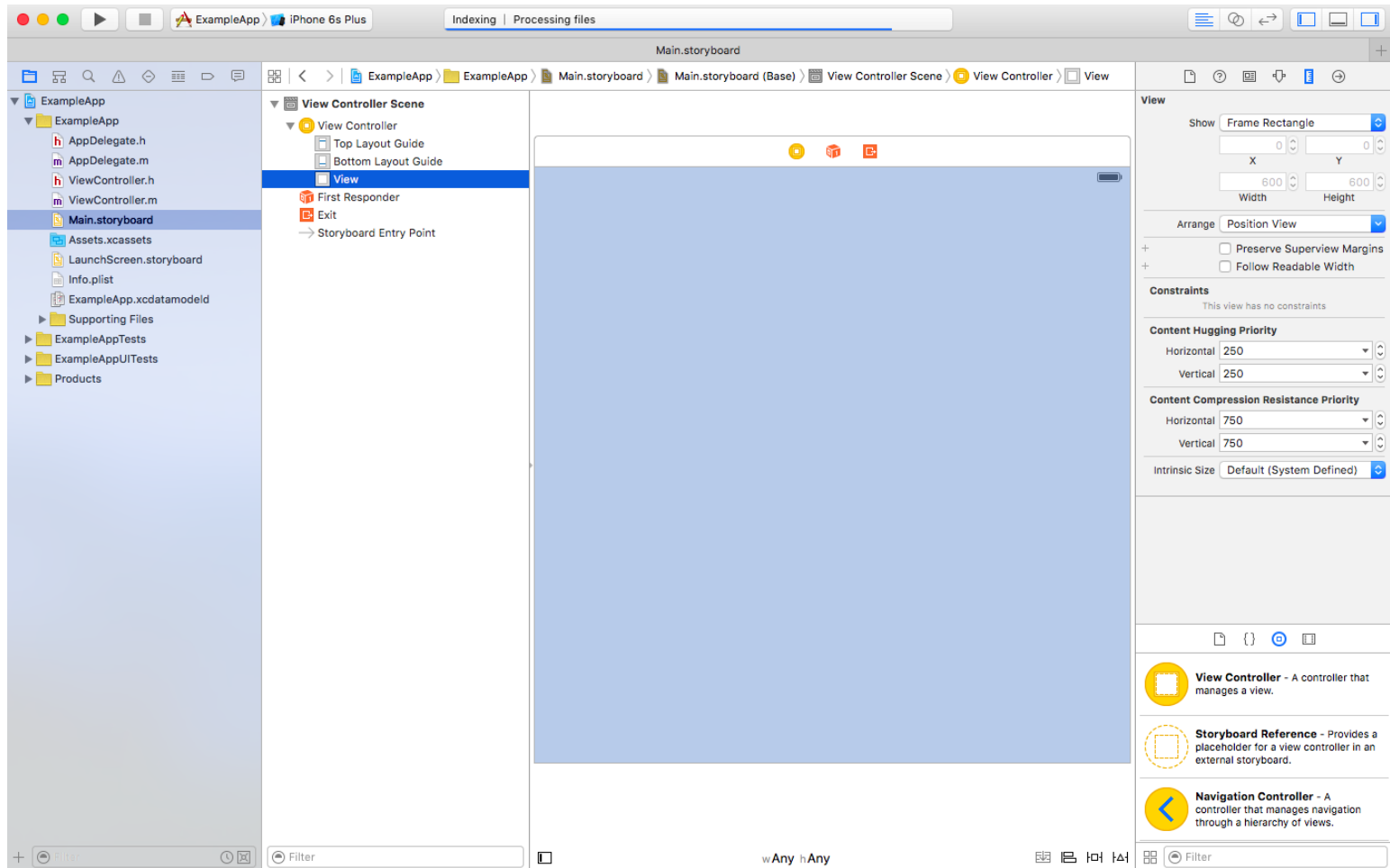


# ◆◆◆ iOS Simulator

## Tools – Xcode – Editor



## Tools – Xcode – Interface Builder



## Tools – Simulator

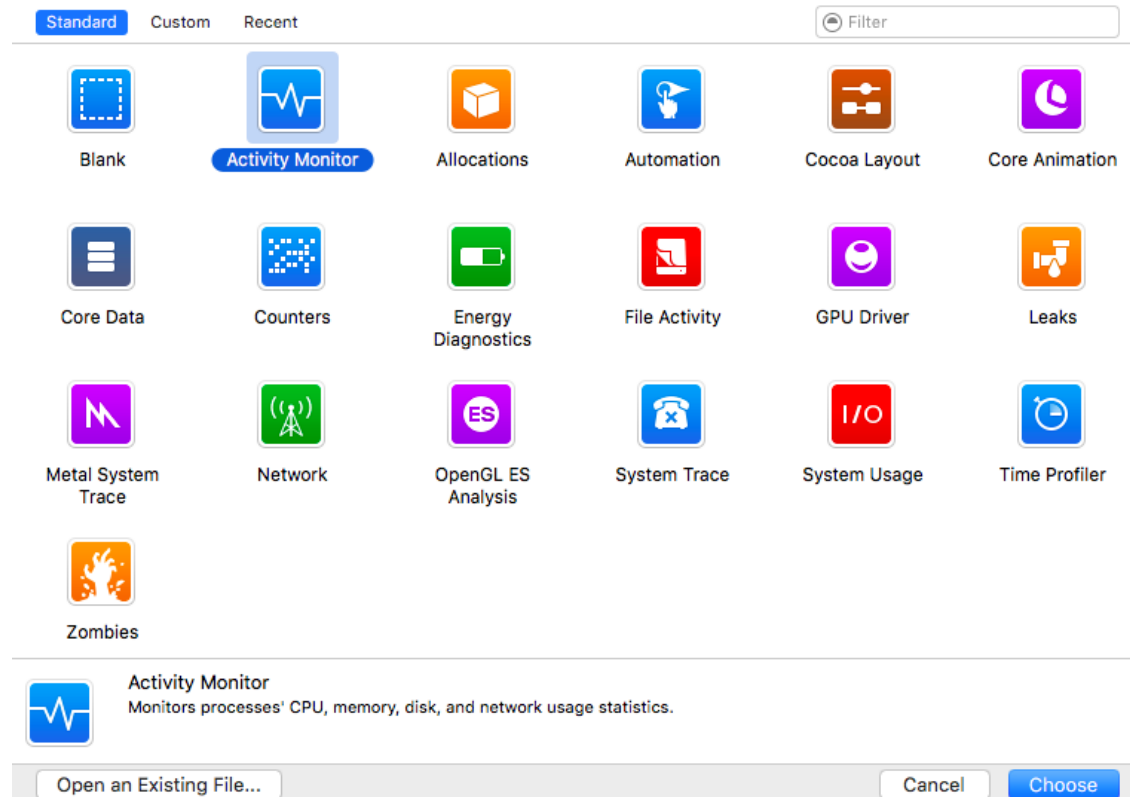
- It is not a complete emulator.
- It includes iOS libraries, but compiled for Intel.
  - OSX architecture.
- It allows users to simulate certain functions:
  - Fingerprint touch.
  - Airplay.
  - Memory warnings.
- However, in order to test certain functions, a physical device is required.





## Tools – Instruments

- It analyses different characteristics of an application.



## Tools – Hopper

- Tool for disassembling and modification of binary files for iOS and Mac OS X.
- Available (only for Mac OS X) at:
  - <http://www.hopperapp.com>
- It is a paid tool, but there is a demo version that allows users to open binary files and investigate their content for 30 minutes.
- It creates program flow control trees.
- It is able to create a pseudocode of an application in order to facilitate the legibility of the code.
- It allows users to conduct searches by tags (names of classes, methods, etc.) and strings.
- The paid version enables the debugging of applications and the modification of executable files in order to repack them.

## Tools – Hopper

The screenshot displays the Hopper Disassembler interface for the file 'DamnVulnerableIOSApp.hop'. The left sidebar shows the 'Strings' tab with a search filter 'http://'. Below the search bar, a list of strings is visible, including various URLs and system messages. The central disassembly window shows the assembly code for the file, with instructions like 'pop ret rbp', 'push rbp', 'mov rbp, rsp', and 'mov qword [ss:rbp+var\_8], rsi'. The right sidebar contains several panels: 'File Information' (Path: /Users/guizos/Library/Developer/Cc, Loader: Mach-O, CPU: intel/x86\_64, Calling Convention: System V), 'Graphic Views', 'Instruction Encoding' (55), 'Format' (Argument --: Default, Type: , Field path: ), 'Comment', and 'Colors and Tags' (Area: , Procedure: , Address: TransportLayerProtectio..., Block: , Procedure: ).

Labels: Strings

Q~http://

Tag Scope

http://mignatitodenacks.com/2013/12/11/ios-application-security-part-24-jailbr...

http://damnvulnerableiosapp.com

http://adlog.flurry.com/postAdLog.do

http://ads.flurry.com/v13/getAds.do

http://data.flurry.com/aas.do

http://twitter-oauth.callback

http://www.google-analytics.com/collect

http://www.google-analytics.com/batch

%@ %s (%d): iAd campaign tracking disabled because the iAd framework...

'%@' is not supported as an RLMArray object type. RLMArrays can only contain in...

'NSNumber' is not supported as an RLMObject property. Supported number types...

'%@' is not supported as an RLMObject property. All properties must be primitive...

Version %@ of Realm is now available: <http://static.realm.io/downloads/cocoa/latest>

[http://static.realm.io/update/cocoa?%@",](http://static.realm.io/update/cocoa?%@)

<https://google.com/>

<https://www.google.co.uk>

<https://appcloud-node-stage.corp.flurry.com/>

<https://appcloud.flurry.com/>

<https://adlog.flurry.com/postAdLog.do>

<https://ads.flurry.com/v13/getAds.do>

<https://data.flurry.com/aas.do>

<https://proton.flurry.com/sdk/v1/config>

<https://api.twitter.com/oauth/authenticate>

[https://api.twitter.com/oauth/request\\_token](https://api.twitter.com/oauth/request_token)

[https://api.twitter.com/oauth/access\\_token](https://api.twitter.com/oauth/access_token)

<https://api.parse.com>

<https://ssl.google-analytics.com/collect>

<https://ssl.google-analytics.com/batch>

<https://www.googletagmanager.com>

mated if desired). For example code, please see the wiki: <https://github.com/yaps...>

r example code, please see YapDatabaseViewMappings.h, or see the wiki: <https://i...>

> analysis section \_\_common  
Analysis segment \_\_LINKEDIT  
Analysis segment External Symbols  
> dataflow analysis of procedure in \_\_TEXT  
> dataflow analysis of procedure in \_\_DATA  
> dataflow analysis of procedure in \_\_LINKEDIT  
> dataflow analysis of procedure in External Symbols  
Background analysis ended

Address 0x100038bb0, Segment \_\_TEXT, -[TransportLayerProtectionVC isSSLPinning] + 0, Section \_\_text, file offset 0x38bb0

File Information

Path: /Users/guizos/Library/Developer/Cc

Loader: Mach-O

CPU: intel/x86\_64

Calling Convention: System V

Graphic Views

Instruction Encoding

55

Format

Argument --: Default

☐ Signed ☐ Negate ☐ Leading Zeros

Type:

Field path:

Manage Types

Comment

Colors and Tags

Area:  Set Clear

Procedure:  Set Clear

Address: TransportLayerProtectio...

Block:

Procedure:

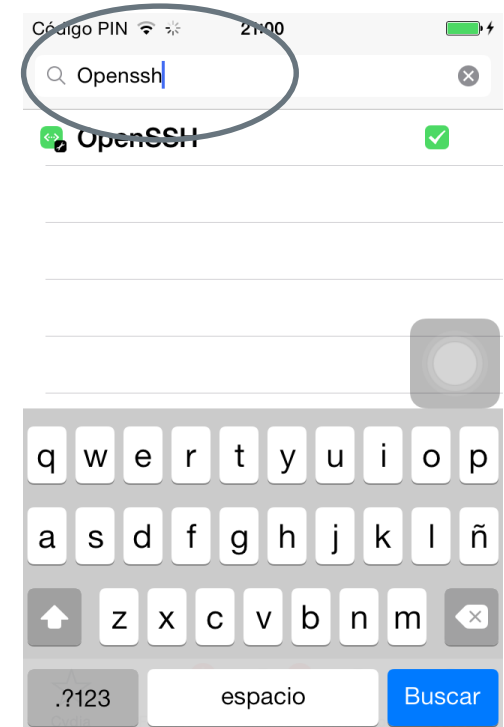
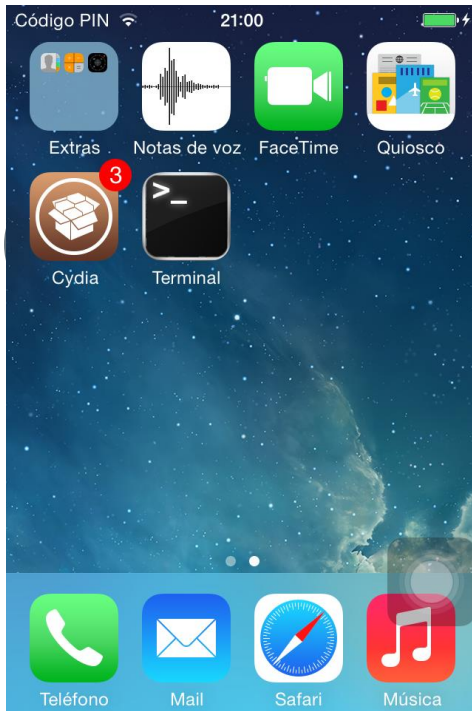
Manage Tags

## Jailbreak tools

- Many of the analysis tasks on iOS cannot be performed on the simulator, since it does not emulate a complete system.
- For example, in order to obtain an executable binary, it is not enough to download the corresponding IPA file, since the executable is encrypted.
- It is necessary to extract it from an iOS device in which it has been installed. This operation requires the installation of third parties' utilities that need a jailbroken device.
- The essential tools to conduct analysis when a jailbroken device is available will be presented later.
- Within this course, the steps required to configure the device in order to extract and analyse binaries will be covered, but methods to jailbreak the device will not.

## Jailbreak Tools – SSH

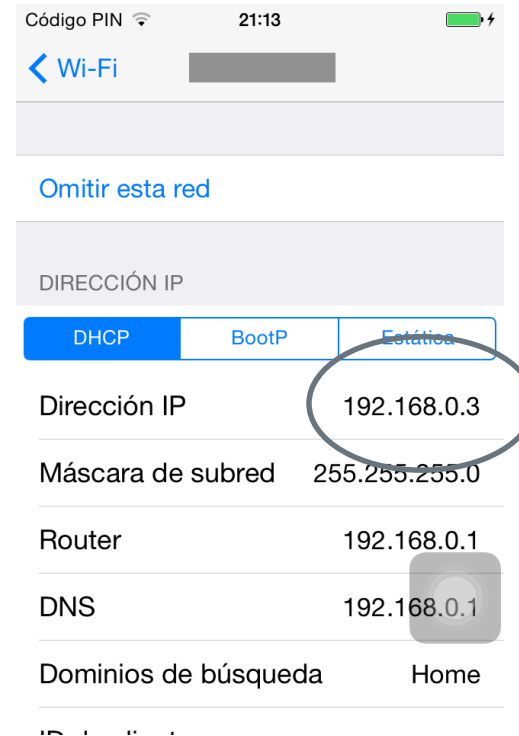
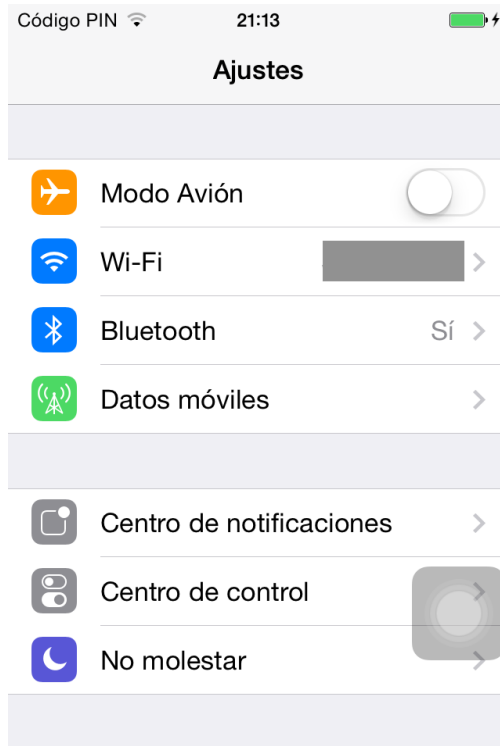
- Cydia should be used to install all the tools.



- Then, install the packet. Once the installation is finished, Cydia restarts the SpringBoard (graphical environment).

## Jailbreak Tools – SSH – Configuration

- The next step is to connect to the device via the terminal.
- To this end, it is necessary to know its IP. It is possible to discover it in the “Settings” section.



## Jailbreak Tools – SSH – Configuration

- From Santoku, or our computer in case it is Linux or Mac.
  - The password by default is alpine; it is recommended to modify it in the first connection

```
santoku@santoku-VirtualBox:~$ ssh root@192.168.0.3
The authenticity of host '192.168.0.3 (192.168.0.3)' can't be established.
RSA key fingerprint is 59:1b:18:23:97:37:05:10:7b:ca:a4:2a:55:06:b9:85.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.0.3' (RSA) to the list of known hosts.
root@192.168.0.3's password: █
```

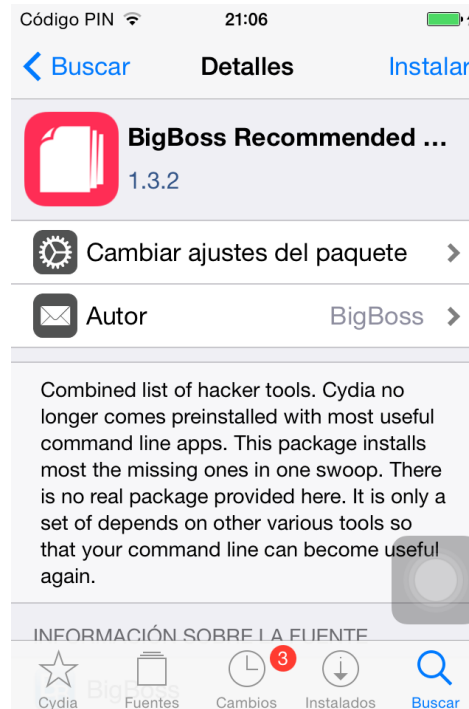
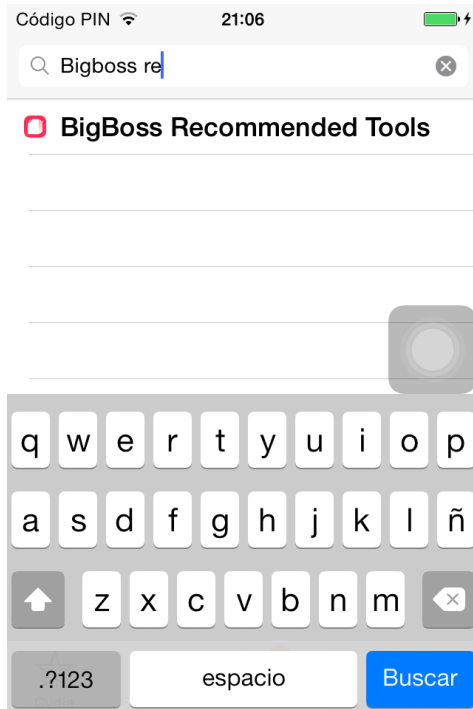
- To modify the password, it is only necessary to execute the passwd command.

```
iPhone:~ root# passwd
Changing password for root.
New password:
Retype new password:
iPhone:~ root# █
```



## Jailbreak Tools – BigBoss Tools

- The same operation has to be performed to install “BigBoss Recommended Tools”.



- Then, install the packet. Once the installation is finished, Cydia restarts the SpringBoard (graphical environment).

## Jailbreak Tools – Clutch

- Clutch is a tool used to decrypt binaries on iOS.
  - In order to decrypt de binary, execute the application and perform core dumps of the memory zone of the application itself.
  - Its installation is made via SSH.
  - Then, it is necessary to download the Clutch tool from the link below and to copy the binary to */usr/bin* in the iOS device.
    - <https://github.com/KJCracks/Clutch/releases>
- ```
> scp Clutch-2.0-RC7 root@192.168.0.3:/usr/bin/Clutch
```

## Jailbreak Tools – iNalyzer

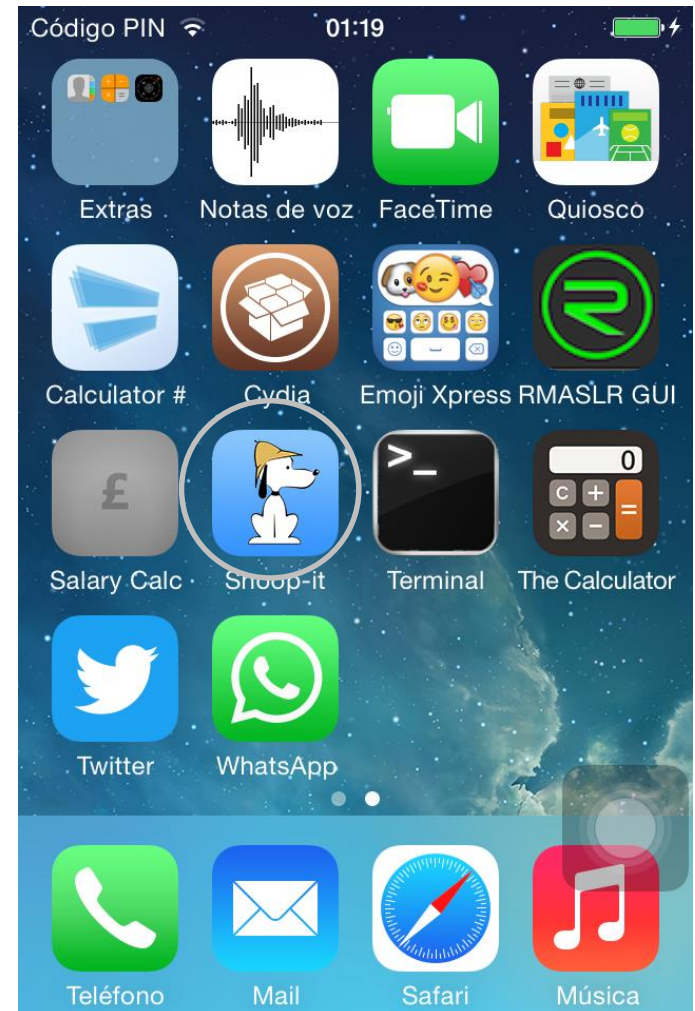
- iNalyzer is a tool by AppSec Labs used for the static and dynamic analysis of iOS applications.
- This unit is only focused on static analysis.
- In order to install iNalyzer, it is only necessary to add the <http://appsec-labs.com/cydia> repository to the Cydia's repository, following the same instructions used to install older repositories.
- Once the repository is loaded, the system console is accessed via SSH:

```
> cd /Applications/iNalyzer5.app  
> ./iNalyzer
```
- It is necessary to use the browser in order to have access to the results of the execution: *[http://ip\\_iphone:5544](http://ip_iphone:5544)*

# ◆◆◆ iOS Simulator

## Jailbreak Tools – Snoop-it

- Snoop-it is a tool for the automatic analysis of iOS applications by NESO Security labs.
- It shows some of the possible vulnerabilities that may affect the application by using a web interface created by the application itself.
- In order to install Snoop-it, it is only necessary to add the <http://repo.nesolabs.de> repository to the Cydia's repository, following the same instructions used to install older repositories.

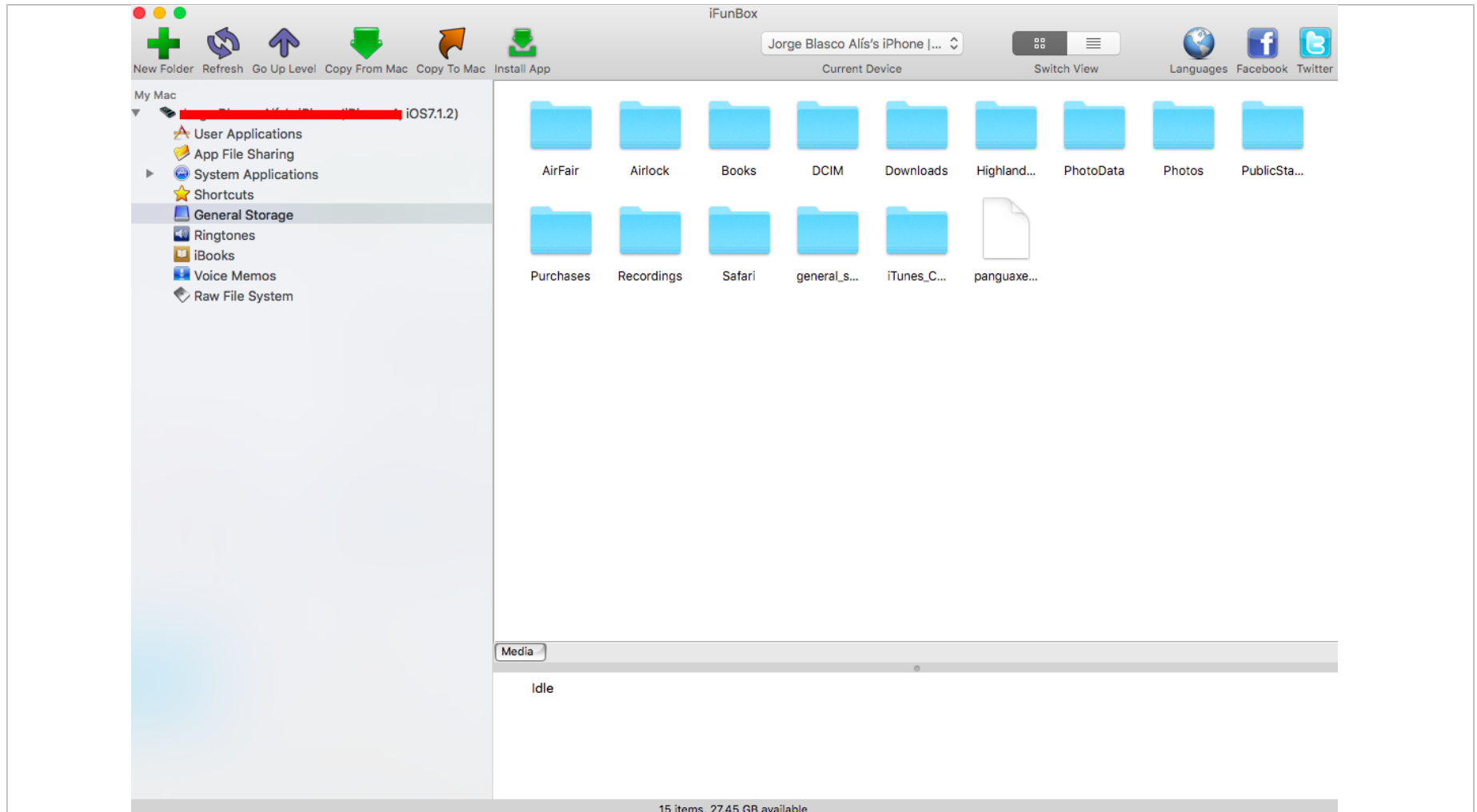


## Other Tools – iFunBox

- It does not require jailbreak.
- Available at:
  - <http://www.i-funbox.com>
- It enables the installation of applications from IPA files:
  - They should have been developed with the business program.
- It enables access to images stored in a device:
- It may use the device as a removable disc.
- It is able to access the file system of each application within the Sandbox.
- And iOS' complete file system.



## Other Tools – iFunBox







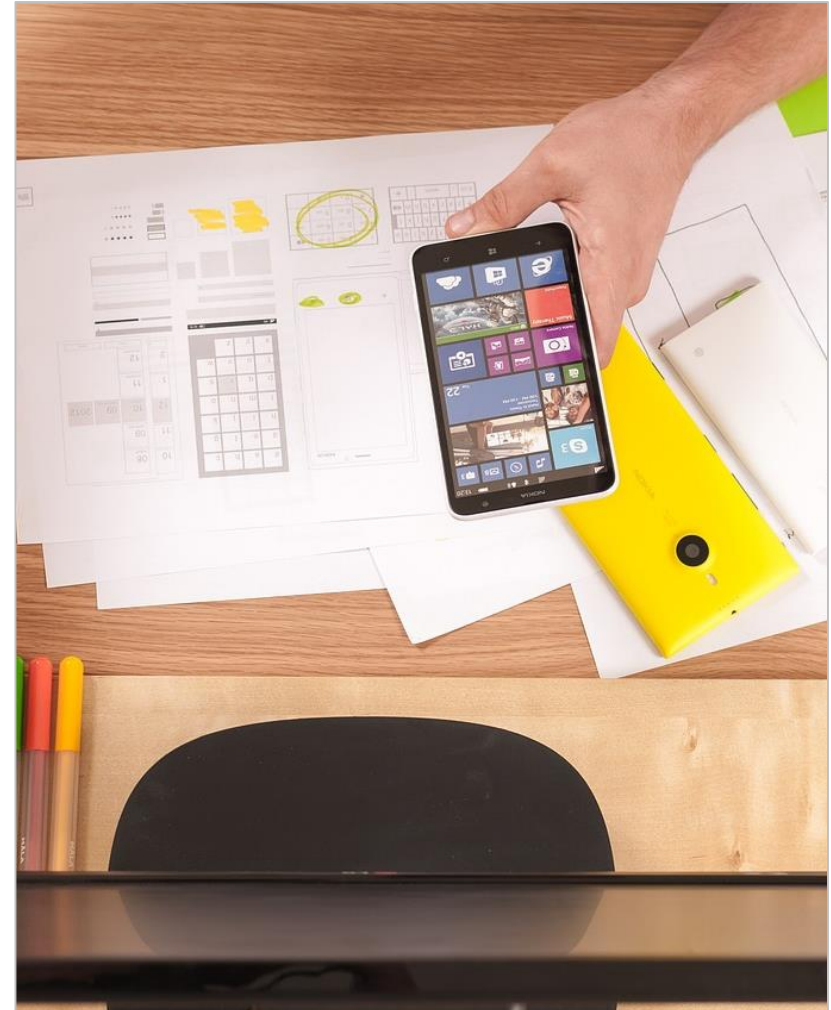
# Other Environments and Professional Tools



# ◆◆◆ Other Environments

## Introduction

- iOS and Android are the most important platforms regarding the market share.
- Other platforms that have a smaller market share among domestic users are widely used in the business context.
  - Windows Phone
    - Simple integration with other solutions in the business context that are widely extended.
  - BlackBerry
    - First company to develop smartphones with permanent connection and competitive rates for companies.



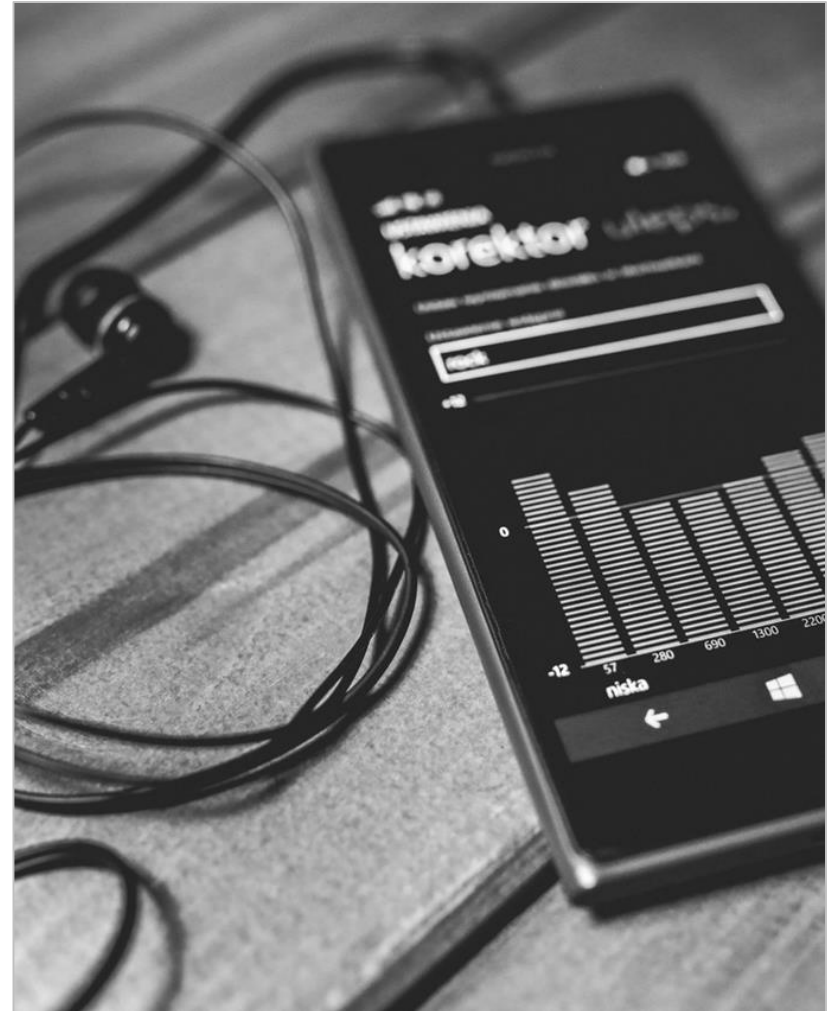
# Windows Phone



# ◆◆◆ Windows Phone

## Introduction

- Windows Mobile successor.
  - Widely used in PDAs at the beginning of 2000
- The first stable version was launched as Windows Phone 7 in 2010.
- In its last version (10), it lost the name of “Phone” and was called Windows 10 Mobile.
- It is well integrated with Microsoft services.
  - Office.
  - Skype.
  - Xbox.

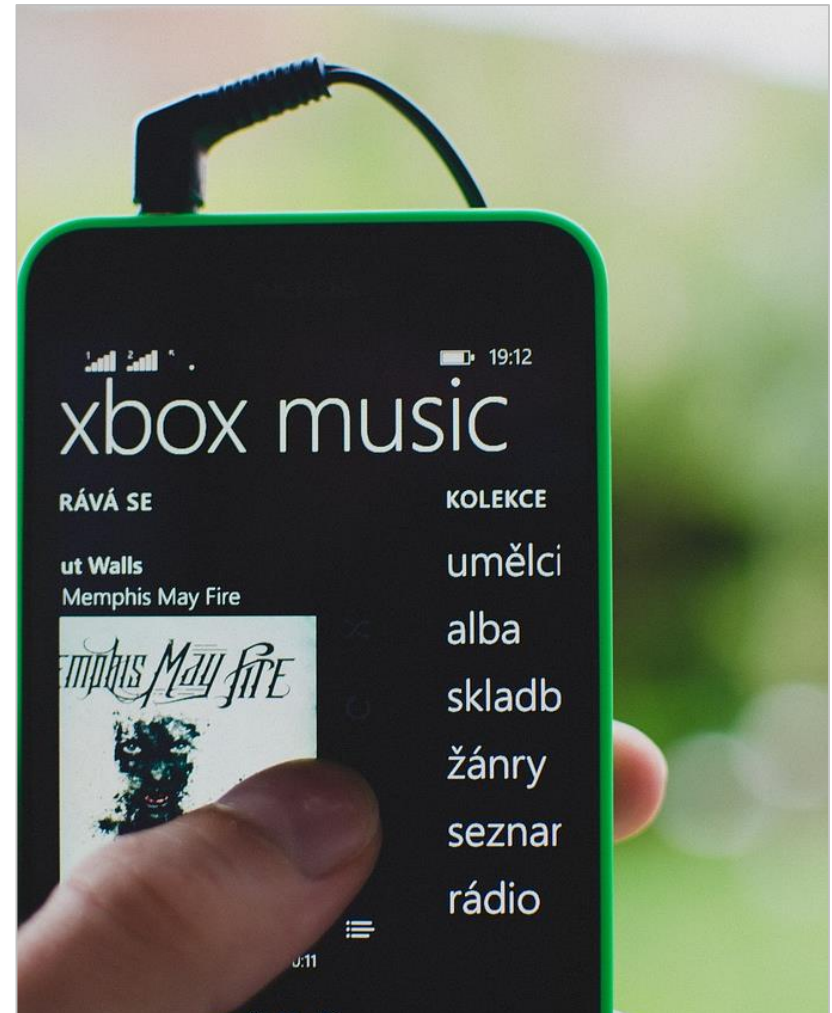




# ◆◆◆ Windows Phone

## SDK

- Windows provides a universal SDK that include all its platforms.
  - Desktop.
  - Tablets.
  - Smartphones.
  - Xbox.
- Three main tools, according to the developer:
  - Visual Studio Community.
  - Visual Studio Professional.
  - Visual Studio Enterprise.

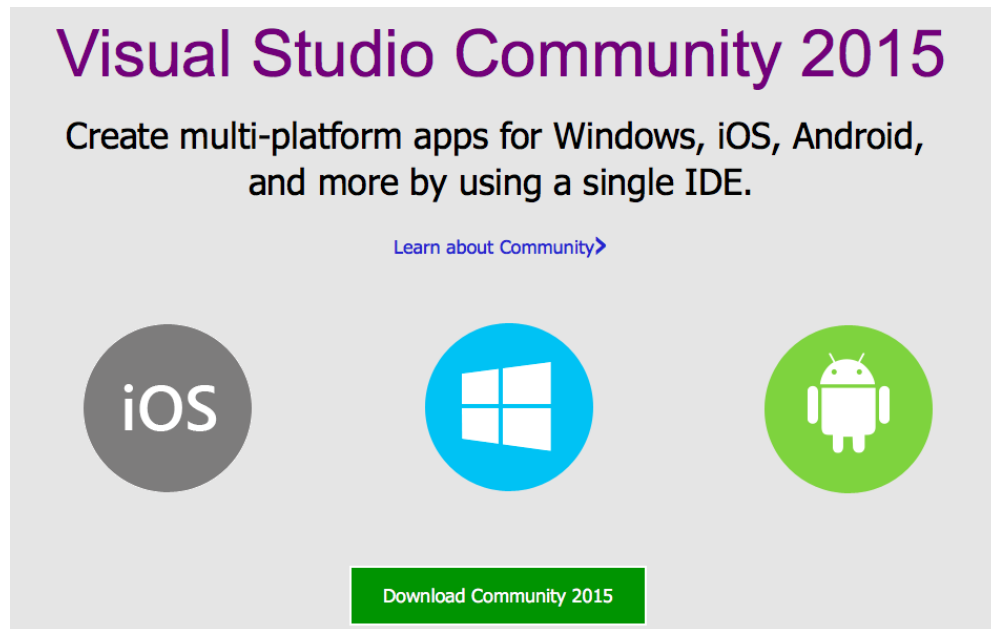


## Visual Studio Community

- Microsoft free IDE.
- Downloadable at:
  - <https://www.visualstudio.com/post-download-vs?sku=community&clcid=0x409>
- Created for small groups of developers.
- It includes:
  - Windows Phone's simulator.
  - Debugger.
  - Static analysis tools.
    - Available in Professional and Enterprise versions.
  - Microsoft specific testing tools.
  - Continuous integration and team collaboration tools.

## Visual Studio Community – Installation

- Access the web of Visual Studio (only for Windows).
- Scroll to the Visual Studio Community section.
- Execute the downloaded file to install Visual Studio.



## Emulator for Windows 10 Mobile

- Desktop application that emulates a device executing Windows 10.
- It is included in Windows 10 SDK.
- Its functioning is similar to iOS simulator.
- It allows users to simulate:
  - Data of the accelerometer.
  - SD cards.
  - Locations.
  - Wireless networks.
  - Communications via NFC.





BlackBerry

## Introduction

- Brand of the Canadian company Research in Motion (RIM), which manufactures devices with an own operative system.
- It became popular in the business context due to the communication options provided by their devices.
- Their main clients are still organisations, while domestic users are decreasing.
- Currently, this brand has three lines of devices with different operative systems.
  - BlackBerry OS.
  - BlackBerry Playbook.
  - BlackBerry 10.

## BlackBerry OS

- Operative system for BlackBerry devices until 2012 (7.1 version).
- The first version (1.0) was launched in 1997 for BlackBerry 850 (a beeper).
- It allows users to develop applications through two technologies:
  - Javascript/CSS/HTML5:
    - It uses BlackBerry WebWorks to access the native APIs of the device.
    - It is compatible with the use of Cordova for the portability to other systems such as Android and iOS.
  - Java:
    - It uses a specific SDK: the BlackBerry Java SDK.
    - Development via Eclipse through a plug-in:
      - It includes a simulator.
    - It includes a cryptographic API specially developed by RIM:
      - Files encryption.
      - Libraries for communications via TLS and SSL.

## BlackBerry Playbook

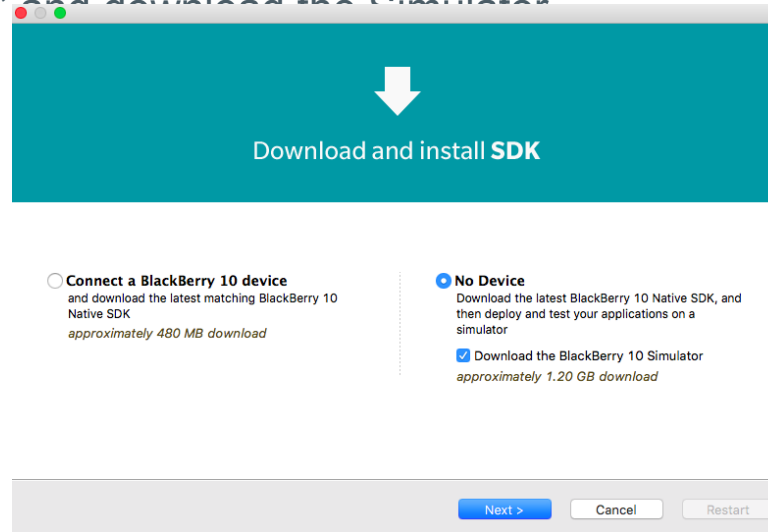
- Specific operative system of the BlackBerry Playbook tablet (2011).
- It enables the development of apps through four technologies:
  - Native SDK
    - Native platform of the device in C/C++.
    - It includes IDE, a compiler and other tools required for the development of apps.
  - Javascript/CSS/HTML5.
  - Adobe Air:
    - It enables the use of applications created with Adobe tools.
      - <http://developer.BlackBerry.com/air/download/#playbook>
  - Android:
    - Playbook OS may execute Android 2.3.3 applications with limitations.
- The Playbook simulator is provided as a virtual machine of VMWare:
  - <http://developer.BlackBerry.com/playbook/native/download>

## BlackBerry 10

- Operative system for BlackBerry launched in 2013.
- Evolution of the operative system of the QNX company (bought in 2010).
  - Integration of multiple components of Playbook OS.
- Again, there are four ways of executing applications:
  - Native SDK.
    - New SDK and IDE (Momentics, based on Eclipse).
  - Javascript/CSS/HTML5.
    - SDK WebWorks for BlackBerry 10 based on Cordova ([download](#)).
  - Adobe Air (discontinued in the latest versions).
  - Android.
    - It enables apps repackaging for their execution in BlackBerry 10.
    - From versions 10.2 and 10.3, BlackBerry 10 provides access to Google Play and Amazon Store for apps with Android 4.3 as maximum version.

## SDK – Installation

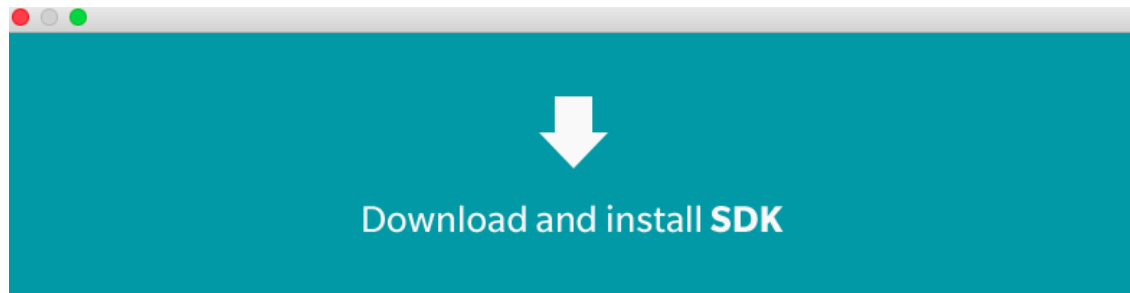
1. Navigate to the link below and download the Momentics version that corresponds to your operative system:  
<http://developer.BlackBerry.com/native/downloads/>
2. Install Momentics following the displayed instructions:
3. Execute Momentics and select a workspace.
4. When the SDK installation dialogue appears:
  - Select “No Device” and download the Simulator



## SDK – Installation

### 5. Select the recommended API level:

- Later, we will install another simulator and API level.



#### API Level

Read more about [API Levels](#)

BlackBerry 10 Native SDK 10.3.1 (Recommended) ▾

< Back

Install

Cancel

Restart



A hand holding a pen is writing on a notebook. In the foreground, a black calculator and a textbook are visible. The textbook contains math problems, including arithmetic series and sigma notation. A semi-transparent blue rectangle is overlaid on the image, containing the text "Assessment Test".

# Assessment Test

Thank you for your  
attention

