Contenido

[1. Autores del trabajo, planificación y entrega 3](#_Toc509240234)

[1.1 Autores 3](#_Toc509240235)

[1.2 Planificación 3](#_Toc509240236)

[1.3 Entrega 3](#_Toc509240237)

[2. Descripción del tipo de tecnología 3](#_Toc509240238)

[Appium: 4](#_Toc509240239)

[Espresso: 4](#_Toc509240240)

[3. Fuentes de información (documentos) 4](#_Toc509240241)

[3.1 Fuentes sobre Mobile Testing. 4](#_Toc509240242)

[3.1.1 Fuente de información 1 sobre Mobile Testing. 4](#_Toc509240243)

[3.1.2 Fuente de información 2 sobre Mobile Testing. 5](#_Toc509240244)

[3.1.3 Fuente de información 3 sobre Mobile Testing. 5](#_Toc509240245)

[3.2 Fuentes sobre APPIUM. 5](#_Toc509240246)

[3.2.1 Fuente de información 1 sobre Appium. 5](#_Toc509240247)

[3.2.2 Fuente de información 2 sobre Appium. 6](#_Toc509240248)

[3.2.3 Fuente de información 3 sobre Appium. 6](#_Toc509240249)

[3.3 Fuentes sobre ESPRESSO. 7](#_Toc509240250)

[3.3.1 Fuente de información 1 sobre Espresso. 7](#_Toc509240251)

[3.3.2 Fuente de información 2 sobre Espresso. 7](#_Toc509240252)

[3.3.3 Fuente de información 3 sobre Espresso. 8](#_Toc509240253)

[4. Fuentes de información (cursos no gratuitos) 9](#_Toc509240254)

[4.1 Cursos no gratuitos sobre Mobile Testing 9](#_Toc509240255)

[4.1.1 Introducción al Mobile Testing 9](#_Toc509240256)

[4.1.2 Testing Movible Avanzado 10](#_Toc509240257)

[4.1.3 Mobile Testing Tools and Concepts Tutorials 10](#_Toc509240258)

[4.2 Cursos no gratuitos sobre Appium 11](#_Toc509240259)

[4.2.1 Appium (Selenium) Mobile Automation Testing from Scratch 11](#_Toc509240260)

[4.2.2 Mobile Rest Automation: Appium, Cucumber and Ruby 11](#_Toc509240261)

[4.2.3 Complete Appium from begginnes to Advanced 12](#_Toc509240262)

[4.3 Cursos no gratuitos sobre Espresso 13](#_Toc509240263)

[4.3.1 Android UI Test: Espresso Fundamentals 13](#_Toc509240264)

[4.3.2 UI Testing with Espresso 14](#_Toc509240265)

[4.3.3 Create test with the Espresso Test Recorder 15](#_Toc509240266)

[5. Fuentes de información (cursos gratuitos) 15](#_Toc509240267)

[5.1 Cursos gratuitos sobre Mobile testing 15](#_Toc509240268)

[5.1.1 Curso gratuito 1 sobre Mobile testing 16](#_Toc509240269)

[5.1.2 Curso gratuito 2 sobre Mobile testing 16](#_Toc509240270)

[5.2 Cursos gratuitos sobre Appium 17](#_Toc509240271)

[5.2.1 Curso gratuito 1 sobre Appium 18](#_Toc509240272)

[5.2.2 Curso gratuito 2 sobre Appium 19](#_Toc509240273)

[5.3 Cursos gratuitos sobre Espresso 20](#_Toc509240275)

[5.3.1 Curso gratuito 1 sobre Espresso 20](#_Toc509240276)

[5.3.2 Curso gratuito 2 sobre Espresso 21](#_Toc509240277)

[6. Ayudas económicas para estudiar las tecnologías 21](#_Toc509240278)

[7. Recursos para implementar las tecnologías 22](#_Toc509240279)

[7.1 Recursos para implementar Appium 22](#_Toc509240280)

[7.1.1 Recursos gratuitos para implementar Appium 22](#_Toc509240281)

[7.1.2 Recursos no gratuitos para implementar Appium 22](#_Toc509240282)

[7.2 Recursos para implementar Espresso 22](#_Toc509240283)

[7.2.1 Recursos gratuitos para implementar Espresso 22](#_Toc509240284)

[7.2.2 Recursos no gratuitos para implementar Espresso 22](#_Toc509240285)

[8. Conclusiones 23](#_Toc509240286)

# 1. Autores del trabajo, planificación y entrega

## 1.1 Autores

Grupo de turno de mañana 1

Amir Sholkami Berube

Javier Ballesteros García

Ismael Sainz Maza Jiménez

Stephany Susan Vela Vellachich

## 1.2 Planificación

Debido a que GnattPro exige ser un usuario premium (de pago) para compartir enlaces, hemos enviado una invitación al correo electrónico para que pueda ver la planificación

## 1.3 Entrega

https://github.com/Amir95sb/TG1

# 2. Descripción del tipo de tecnología

Mobile testing es un proceso en el cual las aplicaciones diseñadas y desarrolladas para los dispositivos móviles (smartphones, tablets o móviles) son probadas en cuanto a su funcionalidad, consistencia y usabilidad.

Hay diferentes categorías sobre las cuales se puede probar el funcionamiento de estas aplicaciones:

* **Pruebas funcionales**: Es el más básico y trata sobre probar las funcionalidades de la aplicación por cada requerimiento.
* **Pruebas de rendimiento**: En esta categoría se prueba el rendimiento de la aplicación en el cliente, en el servidor y en la red.
* **Pruebas de memoria**: En este test se mide la memoria óptima que debe utilizar la aplicación para un buen funcionamiento, ya que los dispositivos móviles vienen con una memoria limitada.
* **Pruebas de interrupciones**: En esta prueba se mira cómo se comporta la aplicación frente a diferentes interrupciones: una llamada, un SMS, aviso de batería baja, etc.
* **Pruebas de instalación**: Aquí se prueba la instalación de la aplicación para que sea lo más fácil y fluida posible. Además, en esta prueba están incluidas las funcionalidades de desinstalación y actualización.
* **Pruebas de usabilidad**: Se utiliza para comprobar la eficiencia, efectividad y satisfacción de la aplicación.

Estas son algunas de las herramientas que podemos utilizar para testear las aplicaciones de dispositivos móviles:

### Appium:

[](http://appium.io/)

Appium es una herramienta de código abierto que nos permite a los usuarios testear todo tipo de aplicaciones de escritorio, web o hibridas. También nos permite automatizar estos test con emuladores o simuladores.

Tiene como característica principal que es ‘cross-platform’ esto quiere decir que permite escribir test frente a múltiples plataformas ya sea iOS, Android o Windows usando la misma API. Esto permite reutilizar código para los distintos sistemas.

### Espresso:

[](https://google.github.io/android-testing-support-library/docs/espresso/)

Espresso es una herramienta para empresas. Como herramienta de código abierto es muy fácil de usar u puede extenderse dentro del entorno de trabajo con facilidad.

Como característica principal Espresso permite crear interfaces de prueba para aplicaciones móviles sin escribir ni una solo línea de código.

# 3. Fuentes de información (documentos)

## 3.1 Fuentes sobre Mobile Testing.

### 3.1.1 Fuente de información 1 sobre Mobile Testing.

<http://www.softwaretestinghelp.com/>

Página de información en la que puedes encontrar todo tipo de información sobre software de pruebas. No es una página que solo se especialice en pruebas de aplicaciones móviles, pero si puedes encontrar mucha información sobre este tipo de pruebas.

Se trata de una página en la que la gente postea información sobre todo lo relacionado con el software de testeo, ya sea nuevos software para ello, como comparaciones entre diferentes software o guías de cómo utilizar estos.

### 3.1.2 Fuente de información 2 sobre Mobile Testing.

<http://www.optimusinfo.com/blog/>

Optimus es una empresa diseñada para ayudar a las organizaciones con sus necesidades de información. Pero no es por esto por lo que elegimos su página como fuente de información.

Optimus es una empresa solidificada en el sector y tiene un blog en el cual postea información sobre su sector. En concreto hacen desarrollo de aplicaciones y testeo de software por lo que la mayoría de sus posts tienen que ver con la tecnología que hemos elegido.

### 3.1.3 Fuente de información 3 sobre Mobile Testing.

<https://dzone.com/mobile-app-developer-tutorials-tools-news>

Página web dedicada a la tecnología en la que podemos encontrar información de todo tipo.

Tiene una sección dedicada a las aplicaciones móviles en las cuales podemos encontrar información sobre el testeo de aplicaciones móviles.

## 3.2 Fuentes sobre APPIUM.

### 3.2.1 Fuente de información 1 sobre Appium.

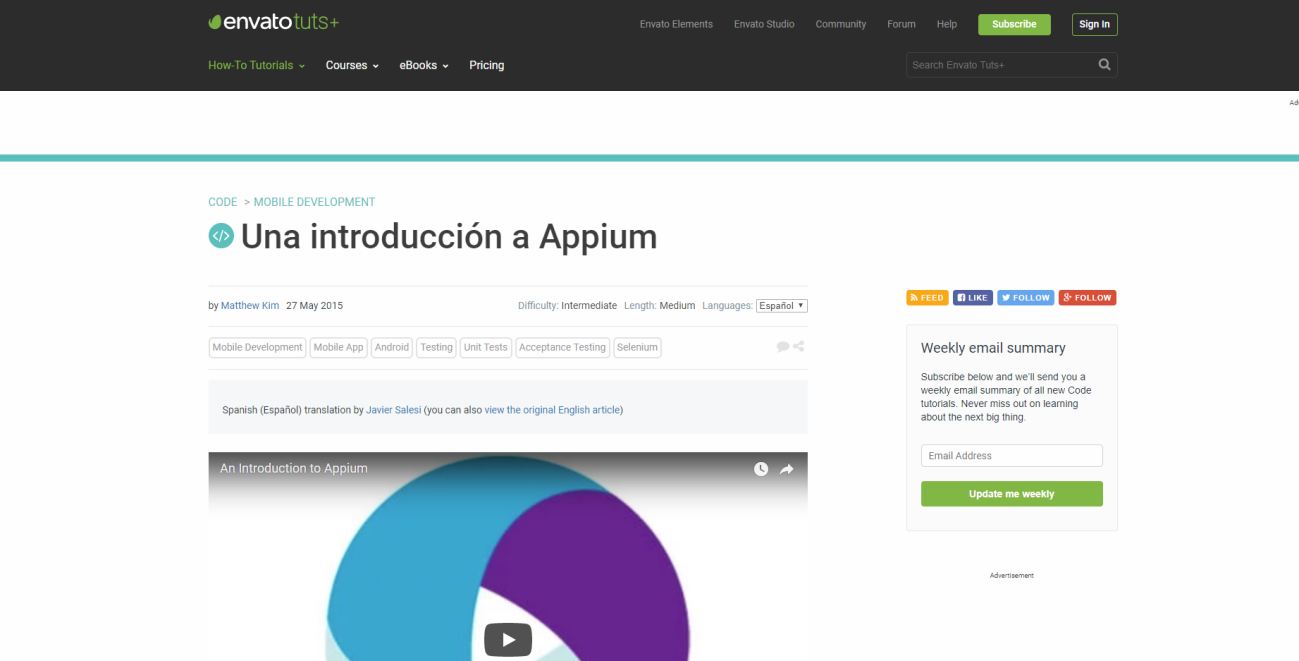
<http://appium.io/>



Página principal del software en cuestión. Aquí podemos encontrar todo tipo de información sobre la aplicación tanto información general como documentación sobre esta y tutoriales.

### 3.2.2 Fuente de información 2 sobre Appium.

<https://code.tutsplus.com/es/tutorials/an-introduction-to-appium--cms-23861>

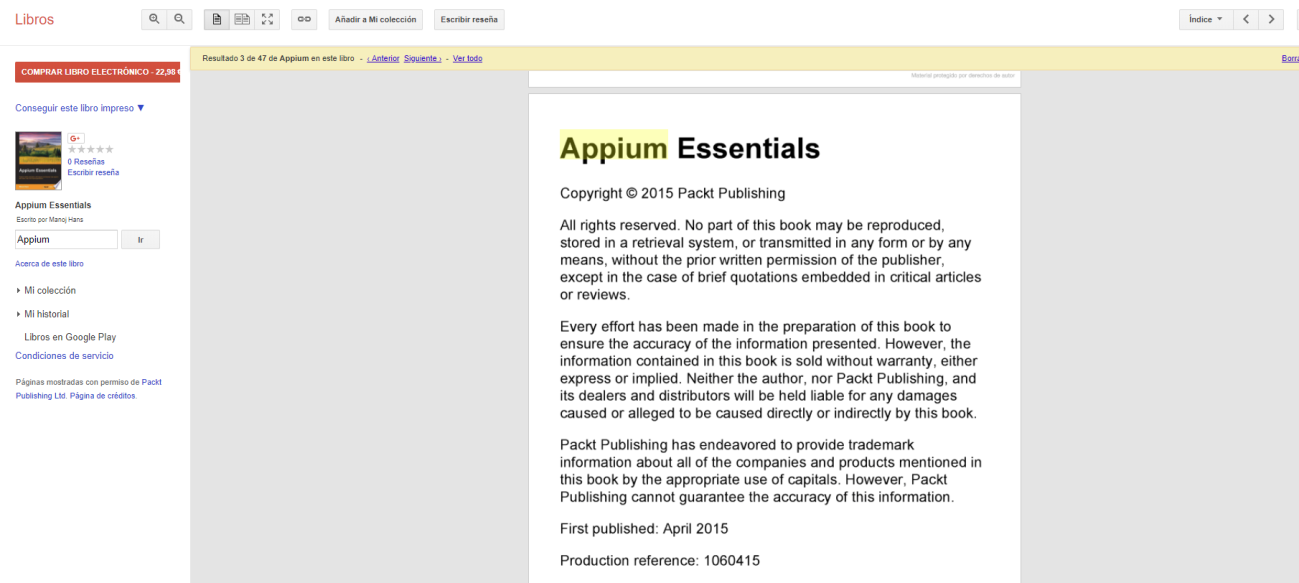


Es una página dedicada a los tutoriales, tiene de todo tipo, también de software de testeo.

En esta paginas nos encontramos un tutorial completísimo sobre cómo empezar con Appium y como configurarlo. Además, contamos con foros para resolver dudas de los expertos.

### 3.2.3 Fuente de información 3 sobre Appium.

<https://books.google.es/books?hl=es&lr=&id=bQoDCAAAQBAJ&oi=fnd&pg=PT8&dq=Appium&ots=b5dG_Kj5X8&sig=AlUYW-RLjqrMvCEVKbbVP1aLfiU#v=onepage&q=Appium&f=false>

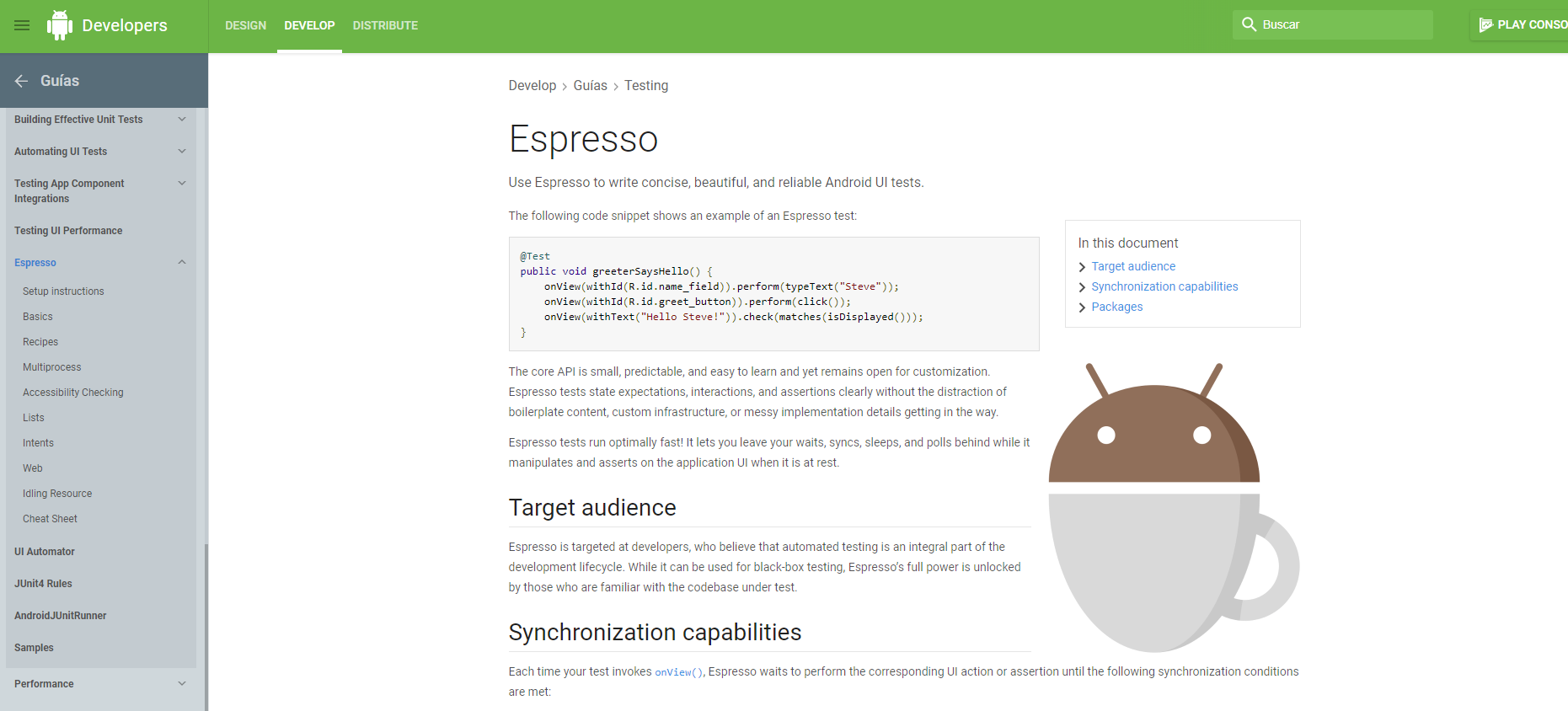


Libro online sobre lo esencial para aprender a utilizar Appium. ¡Básicamente un manual de uso gratis en la web!

## 3.3 Fuentes sobre ESPRESSO.

### 3.3.1 Fuente de información 1 sobre Espresso.

<https://developer.android.com/training/testing/espresso/index.html>

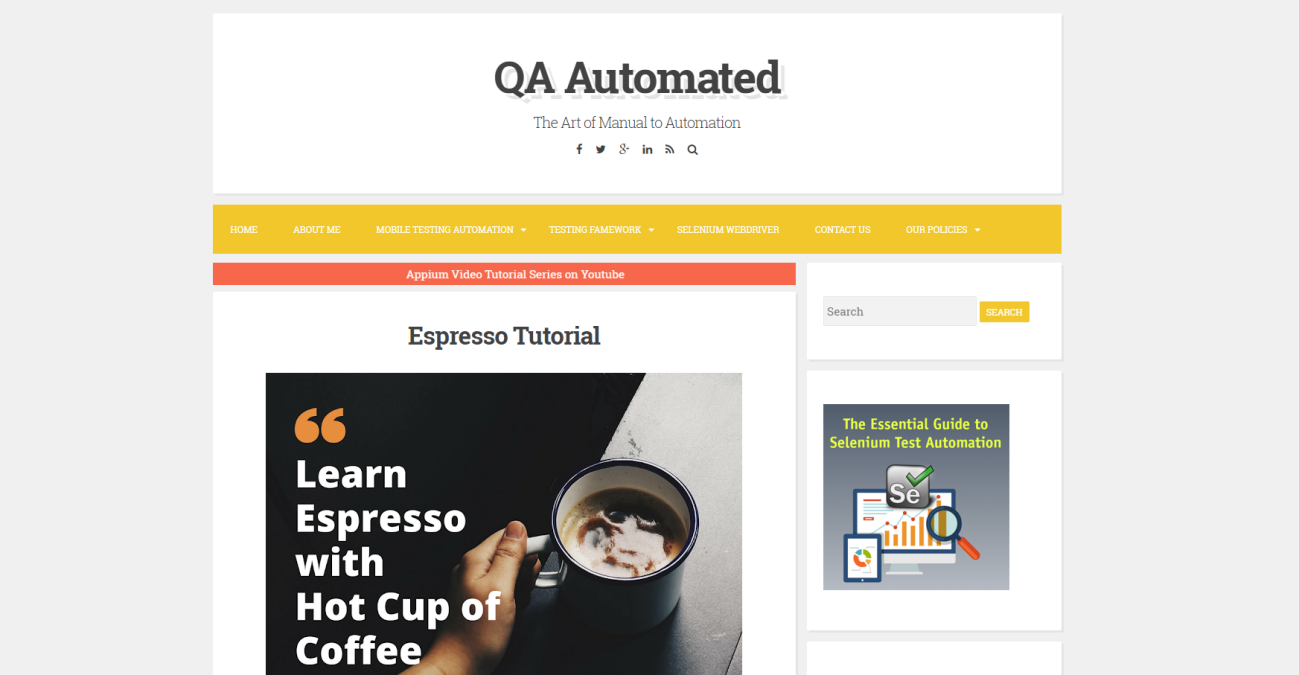


Es la página principal de Espresso donde podemos ver desde como instalar el software como información general de este y ejemplos de cómo utilizarlo.

Es una página para usuarios algo más avanzados.

### 3.3.2 Fuente de información 2 sobre Espresso.

<http://www.qaautomated.com/p/blog-page.html>



Blog dedicado a la automatización de software, una de las características principales de Espresso. En este blog podemos encontrar mucha información sobre este tipo de aplicaciones software. También contiene información sobre Appium.

En concreto tiene una zona dedicada a Espresso en la que puedes encontrar toda la información necesaria para ponerte a trabajar con Espresso.

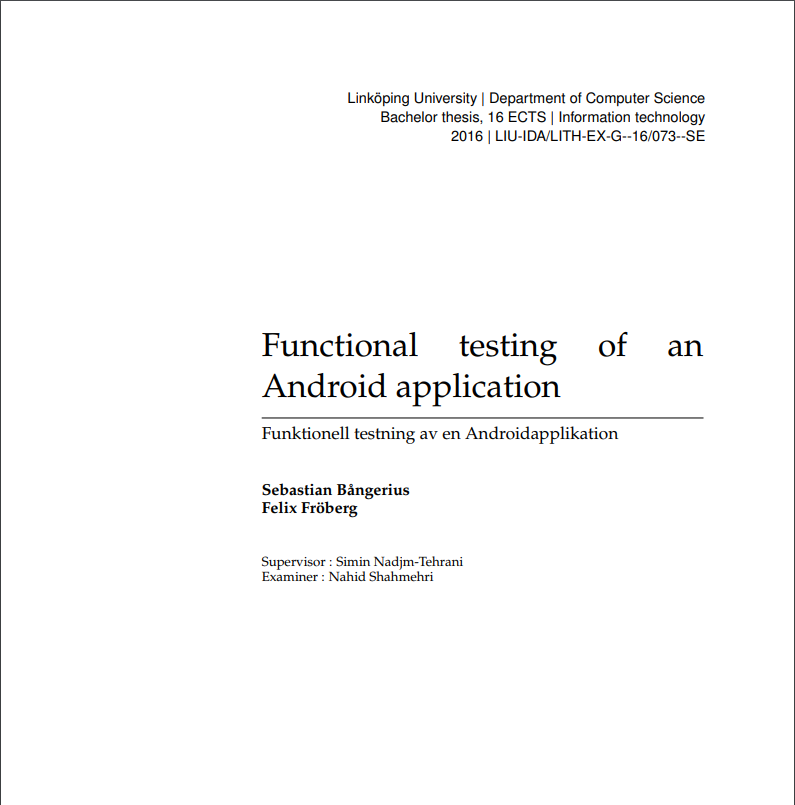
### 3.3.3 Fuente de información 3 sobre Espresso.

Web:

<http://www.diva-portal.org/smash/record.jsf?pid=diva2%3A1034457&dswid=438>

Artículo:

http://www.diva-portal.org/smash/get/diva2:1034457/FULLTEXT01.pdf

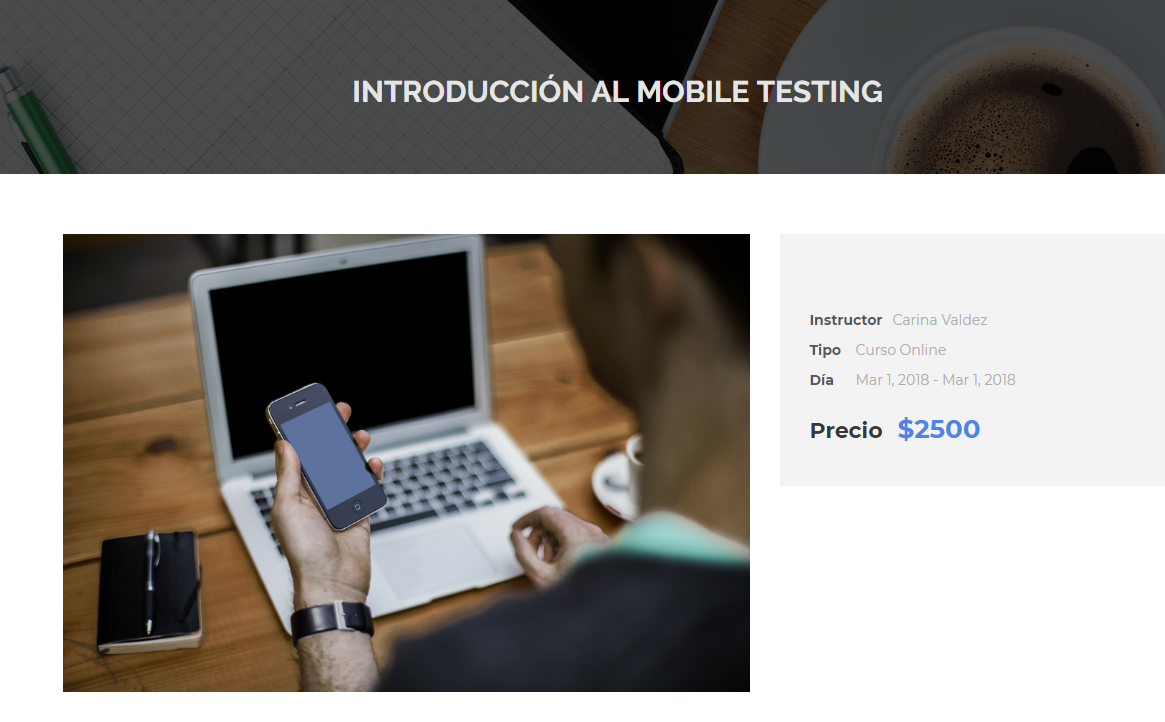


DIVA es un portal donde estudiantes de diferentes universidades presentan artículos interesantes. En este caso nos encontramos un artículo en el que se puede ver todo el proceso de pruebas de una aplicación Android utilizando Espresso.

# 4. Fuentes de información (cursos no gratuitos)

## 4.1 Cursos no gratuitos sobre Mobile Testing

### 4.1.1 Introducción al Mobile Testing

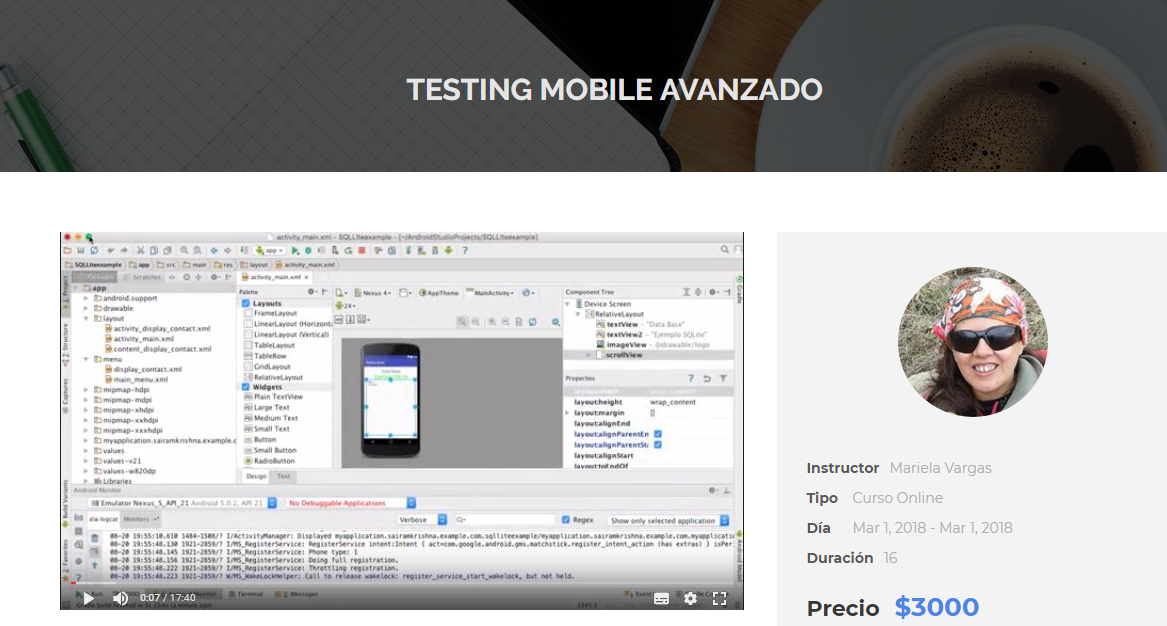


URL: <https://testingbaires.com/course/introduccion-al-mobile-testing/>

Este curso es tipo Online, por lo que es accesible desde todas las partes del mundo. Para argentinos cuesta 2500 pesos argentinos, unos 100 euros. Para el resto del mundo cuesta 220 dólares.

Se dan cinco clases, donde se explicarán contenidos relacionados con estrategias de testing, simuladores, emuladores entre otros. Además, se darán prácticas, y una clase para conocer herramientas avanzadas de esta tecnología.

### 4.1.2 Testing Movible Avanzado



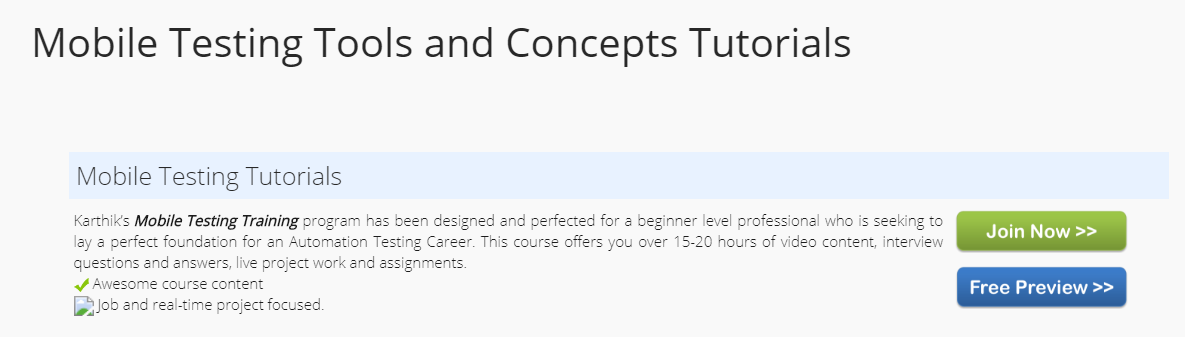
URL: <https://testingbaires.com/course/testing-mobile-avanzado/>

Este curso es la continuación del anterior. De la misma forma, es accesible online. Se usarán herramientas para IOS y Android como Xcode, Instruments, etc.

El curso se divide en 8 clases de dos horas cada uno, dividiéndose el temario en tres partes: herramientas para iOS 9, herramientas para Android y herramientas generales.

El precio si estás en Argentina es de 3000 pesos argentinos, unos 120 euros. Si se encuentra en el exterior es de 250 dólares.

### 4.1.3 Mobile Testing Tools and Concepts Tutorials



URL: <https://www.itelearn.com/mobile-testing/>

En este curso vía online ofrece vídeos de introducción gratuitos, pero el contenido importante es para miembros.

Se ofrecen varias tarifas:

* Un año de vídeos ilimitados: 500 dólares
* Una suscripción de por vida: 1500 dólares
* Personalizados: desde 20 dólares por un mes hasta packs especiales por 2500 dólares.

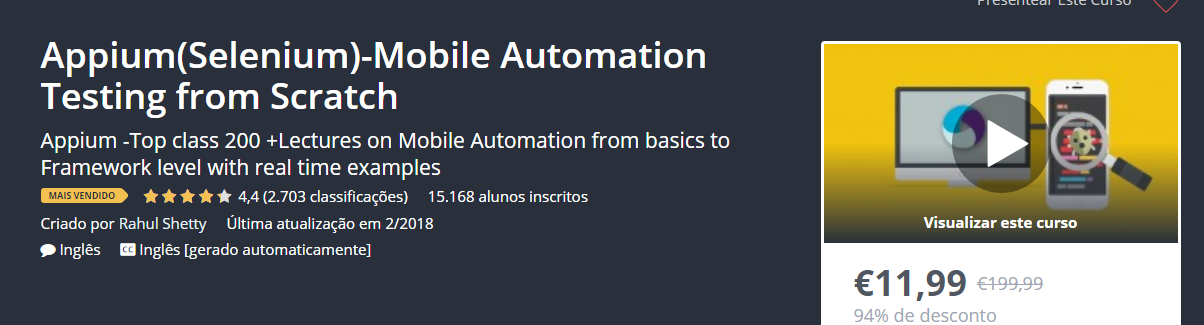
Esta página no solo ofrece este curso, si no muchos otros relacionados con la tecnología y la ingeniería.

El curso tiene una pequeña introducción con dos vídeos gratuitos, de 30 y 20 minutos respectivamente. Se aprenderá sobre los tipos de dispositivos móviles, la importancia y desafíos, las mejores prácticas, etc.

En la segunda parte se verán herramientas, emuladores, proyectos y casos.

## 4.2 Cursos no gratuitos sobre Appium

### 4.2.1 Appium (Selenium) Mobile Automation Testing from Scratch



URL: <https://www.udemy.com/mobile-automation-using-appiumselenium-3/>

Es un curso que originalmente cuesta 200 euros, pero que gracias a una oferta especial de la página web se puede adquirir a 12.

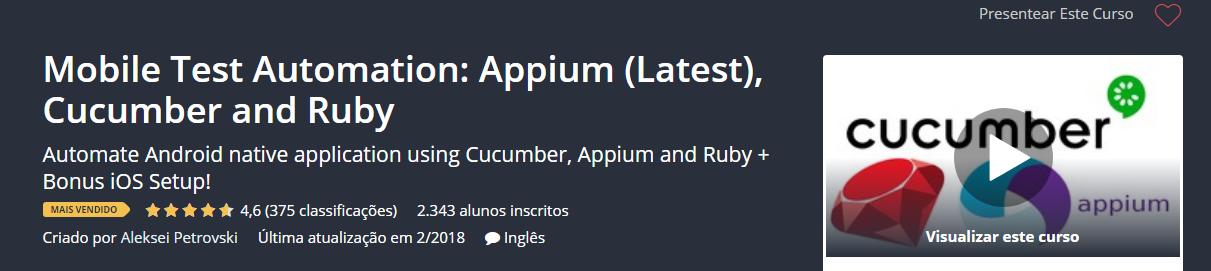
Este curso cuenta con muchos recursos tales como vídeos con un total de duración de 27 horas, 30 artículos, 11 recursos complementarios. Además, es de acceso total vitalicio una vez se compra, se puede acceder desde el móvil y la televisión y se dará un certificado al finalizar.

Con el curso se aprenderá a automatizar cualquier app en Android/iOS usando Appium, se aprenderán los diferentes estándares de codificación para optimizar lo mejor posible la automatización, se aprenderá sobre Selenium 3.0, etc.

El curso se divide en 25 temas, cada uno con sus aulas y su tiempo de trabajo.

Para este curso no se necesita saber programar. El curso ofrece lecciones vía online, e instructor de soporte para las dudas. El material está disponible para descarga.

### 4.2.2 Mobile Rest Automation: Appium, Cucumber and Ruby



URL: <https://www.udemy.com/test-automation-with-appium-cucumber-and-ruby/>

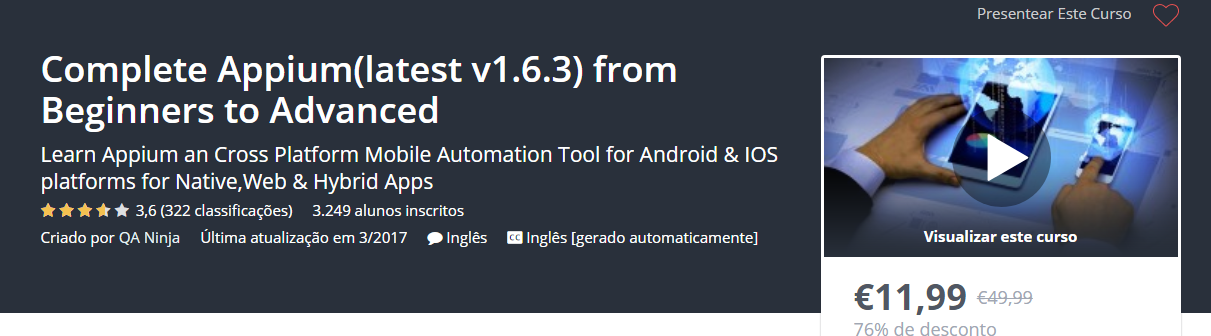
El curso cuesta originalmente 200 euros, pero de Nuevo, con la oferta se puede comprar a 12.

El curso incluye el siguiente material: 6 horas de vídeo, 37 recursos complementarios. Además de las demás condiciones de servicio, como acceso total y vitalicio, acceso desde móvil y televisión, y certificado de finalización.

Durante el curso se aprenderá a usar la framework de Appium con Android, además de Cucumber y Selenium. Se usará Appium para identificar elementos en la aplicación, como usar los gestos de Appium, etc.

Se podrá hacer uso de lecciones adicionales y más de 50 video conferencias.

### 4.2.3 Complete Appium from beginners to Advanced



URL: <https://www.udemy.com/mobile-test-automation-using-appium-selenium-30/>

El curso cuesta originalmente 50 euros, pero esta abaratado a 12 euros.

Incluye 13 horas de vídeo, 13 artículos, 2 recursos complementarios y las anteriores características mencionadas.

El curso está hecho para aprender a automatizar aplicaciones móviles en iOS y Android, implementar una framework desde cero y aprender a manejar Appium.

Para el curso se requiere tener acceso a Windows y Mac Machine y tener acceso a un dispositivo Android e IOS para probar el código en dispositivos reales.

## 4.3 Cursos no gratuitos sobre Espresso

### 4.3.1 Android UI Test: Espresso Fundamentals



URL: <https://www.pluralsight.com/courses/android-ui-tests-espresso-fundamentals>

El curso permite una prueba gratuita, pero hay contenidos de pago. Está dirigido a un nivel de inicio y dura más de una hora.

La plataforma ofrece varias tarifas:

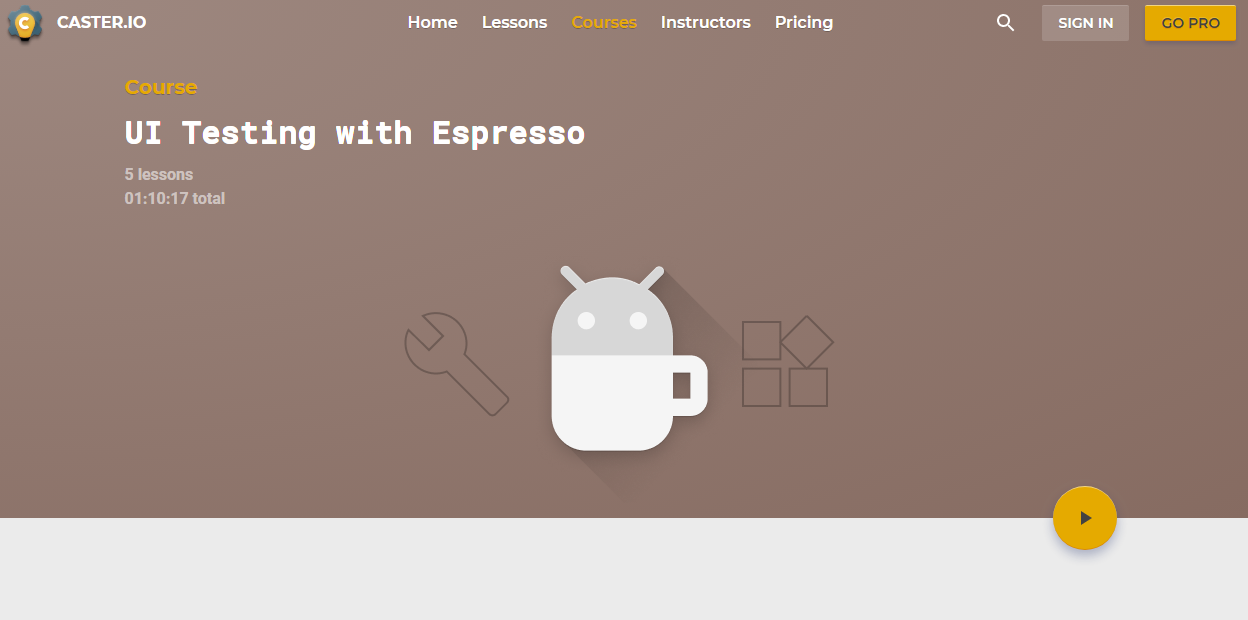
- Individual: 29 dólares al mes o 299 dólares al año.

- Empresa: 499 dólares por usuario por el pack profesional o 699 por usuario por el pack Enterprise.

En todos los packs se incluyen archivos de ejercicio, las apps de móvil y televisión, canales, etc.

El curso ofrece aprendizaje de los fundamentos de la escritura de pruebas de interfaz de usuario de Android usando Espresso. Dentro del temario se verá las acciones y combinaciones más comunes de Espresso, las aserciones y que otras herramientas serán útiles.

### 4.3.2 UI Testing with Espresso



URL: <https://caster.io/courses/espresso>

El curso permite ver la introducción de forma gratuita pero el resto de contenido es de pago, solo para miembros Pro.

Hay cuatro tipos de miembros:

- Indy: un solo usuario individual, de acceso ilimitado a las lecciones. 29 dólares al mes.

- Indy QTRL: 69 dólares por trimestre, con las mismas garantías que Indy.

- Indy yearly: 179 dólares al año, con mejor ROI y cuyo precio está bloqueado.

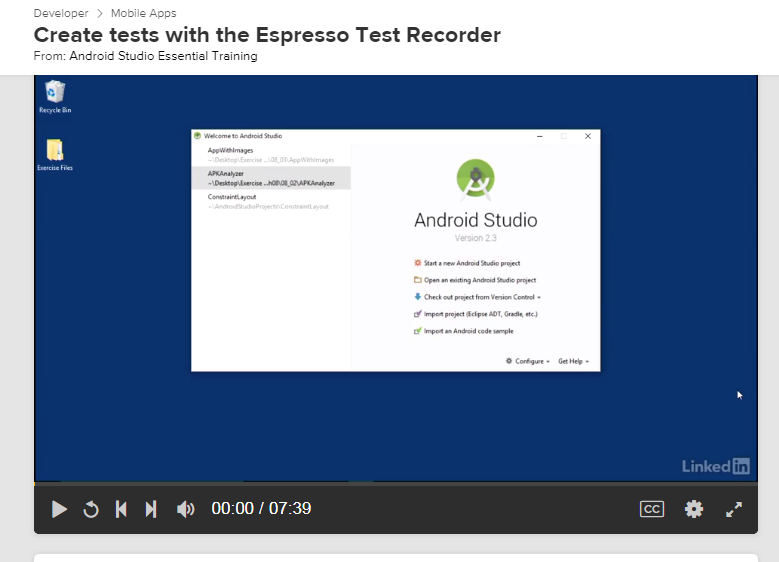
- Enterprise: 229 dólares al mes por 10 usuarios.

El curso se divide en 4 lecciones más la introducción donde se verá ViewMatchers, ViewAssetions, etc. y debug. Se simularán varias acciones de una lista de datos, se crearán ViewMatchers personalizadas y otros.

En el curso se aprenderá a usar Espresso 2.0, a usar View Actions, ViewAssertion y más. Se implementará una sencilla TDD.

El curso dirá un total de una hora y diez minutos en vídeos.

### 4.3.3 Create test with the Espresso Test Recorder



URL:<https://www.lynda.com/Android-tutorials/Create-tests-Espresso-Test-Recorder/453588/614681-4.html>

Se pone a disposición del usuario un vídeo de 7 minutos, pero para realizar los ejercicios debe ser miembro.

Hay dos tipos de miembros: basic por 15 dólares al mes o Premium por 25.

El curso dispone de una transcripción del vídeo, un FAQ y archivos de ejercicios, además da la posibilidad de verlo sin estar conectado a la red.

El nivel requerido para el curso es de inicial.

# 5. Fuentes de información (cursos gratuitos)

## 5.1 Cursos gratuitos sobre Mobile testing

Nuestra tecnología es Mobile Testing Tools. Hemos podido encontrar 2 cursos totalmente gratuitos.

Porque es importante Mobile Testing Tools, pues; Es importante probar cada extremo del software para evitar cualquier complicación posterior. Cierres inesperados, errores en funcionalidades clave, fallos de usabilidad… cualquiera de estas razones puede ser un motivo justificable para una mala opinión en una tienda de aplicaciones. Hoy en día existen algunas herramientas y plataformas realmente potentes para compartir con terceras personas una versión beta de una aplicación para su testeo: cuántas más personas interactúen, más fácil es detectar los bugs.

### 5.1.1 Curso gratuito 1 sobre Mobile testing



URL:

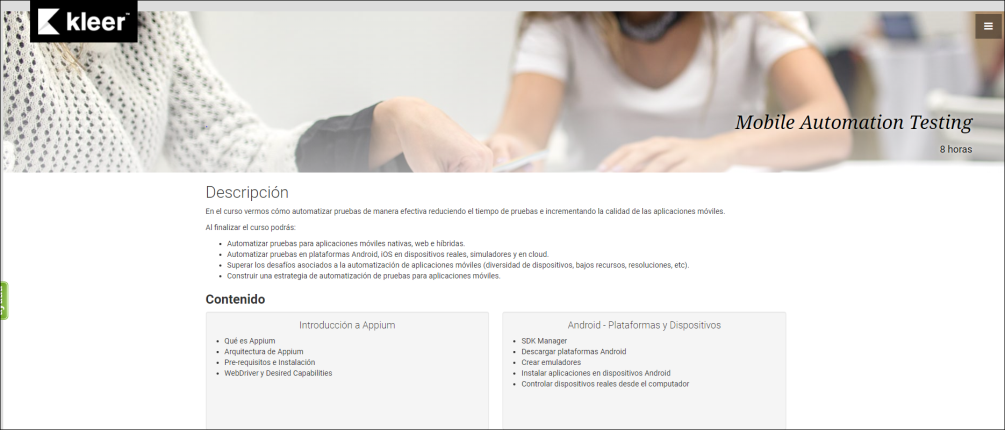
https://testingbaires.com/testing-de dispositivos-moviles-extracto-de-la primera-clase/

Mandando un correo al email que nos aparece en pantalla nos darán toda la información y podremos inscribirnos de manera gratuita

En este curso nos enseñan a realizar esas pruebas mediante aplicaciones para nuestros teléfonos móviles, podemos ver más información en el enlace y ahí nos podremos inscribir.

### 5.1.2 Curso gratuito 2 sobre Mobile testing

Aquí podemos ver otro curso gratuito donde también nos introduce conceptos a tecnologías más específicas como Appium nuevamente gratuito y con 8 horas de duración



Aprendes:

* Desarrollar scripts y pruebas automatizadas.
* Crear pruebas sostenibles, legibles y confiables.
* Aprender el uso correcto de una gran cantidad de herramientas disponibles.
* Construir una estrategia de automatización de pruebas.
* Colaborar entre desarrolladores y testers para una efectiva implementación.
* Principios y patrones de diseño de software que incrementan la testeabilidad.

Obtienes:

* Experiencia de primera mano en la implementación de pruebas automatizadas.
* 1 hora de consultoría gratuita para el equipo o empresa de cada uno de los asistentes
* Certificado de asistencia.
* Material digital e impreso de referencia para utilizar durante y luego del curso.

URL:

http://www.kleer.la/es/categoria/software/cursos/205-mobile-automation-testing

## 5.2 Cursos gratuitos sobre Appium

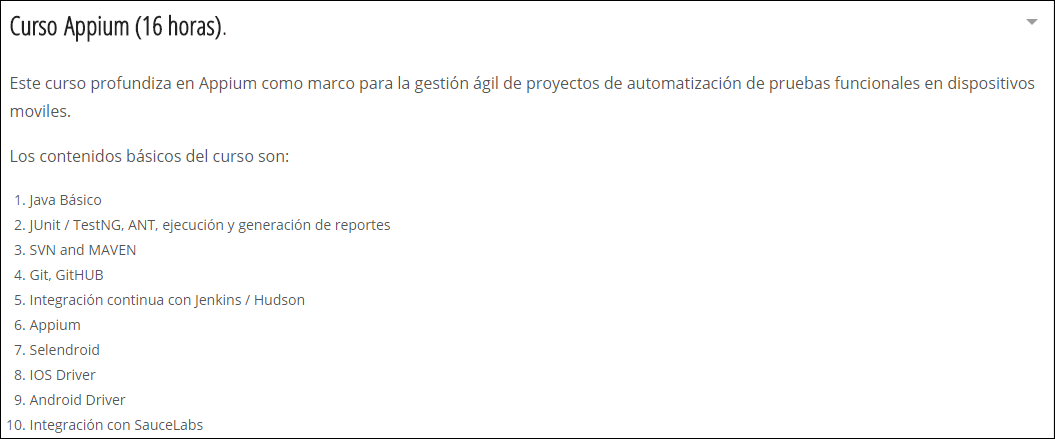
En nuestro caso hemos escogido Appium.

[Appium](http://appium.io/) es el estándar en automatización de pruebas para móviles. Se trata de un framework [open source](https://github.com/appium/appium) que permite probar aplicaciones nativas, híbridas o, como en nuestro caso, aplicaciones web. Se dice de Appium que 'es como selenium, pero para aplicaciones móviles'.

Appium se va a encargar de ejecutar las pruebas en el dispositivo móvil. Tiene varias características que lo hacen interesante:

* Nos vale tanto para dispositivos Android como para iOS.
* Podemos escribir los tests en el lenguaje que más nos guste: Java, Objective-C, Python, Ruby, C#, en todos los casos usando el API de Selenium WebDriver.
* Tiene un grado de madurez alto, y apunta a ser el estándar para pruebas en dispositivos móviles de los próximos años.

### 5.2.1 Curso gratuito 1 sobre Appium

En el siguiente curso que podemos inscribirnos de manera gratuita podemos ver todos los contenidos del curso que se darían. El pequeño problema de este curso es que solo se trataría de 16 horas para la realización del curso que se darían en muy pocas sesiones.

URL:

https://www.itnove.com/es/agile/formacion-automatizacion-pruebas-tdd-selenium-webdriver-appium

### 5.2.2 Curso gratuito 2 sobre Appium

### Captura.PNG

Aquí como podemos ver podemos realizar el curso Appium y también nos ofrecen información sobre Selenium para las pruebas de móviles y cómo podemos apreciar el curso es gratuito. El curso además incluye:

* 1 hora de vídeo bajo demanda
* Acceso de por vida
* Acceso en dispositivos móviles y TV
* Certificado de finalización

Por el contra este curso nos pide como requisitos un mínimo de conocimientos de Appium y Selenium para poder realizarlo.

URL:

https://www.udemy.com/test-data-generation-for-selenium-and-appium/

## 5.3 Cursos gratuitos sobre Espresso

Para esta segunda tecnología hemos elegido Espresso:

Espresso es un [framework](https://es.wikipedia.org/wiki/Framework) de testing open source lanzado por [Google](https://es.wikipedia.org/wiki/Google) el cual provee una [API](https://es.wikipedia.org/wiki/API) que permite crear pruebas de [interfaz de usuario](https://es.wikipedia.org/wiki/Interfaz_de_usuario) (de ahora en adelante UI por sus siglas en inglés) para simular interacciones de usuarios en una aplicación Android (en la versión 2.2 en adelante).

Su principal ventaja es que nos permite la sincronización automática de las acciones de las pruebas con la interfaz de usuario de nuestra aplicación. Además, permite la ejecución de pruebas en máquinas x86 en un ambiente multihilo, solucionando problemas de concurrencia asociados al testing de UI.

### 5.3.1 Curso gratuito 1 sobre Espresso

Aquí podemos ver un curso gratuito sobre las herramientas de pruebas para móviles en este caso Android usando la Espresso

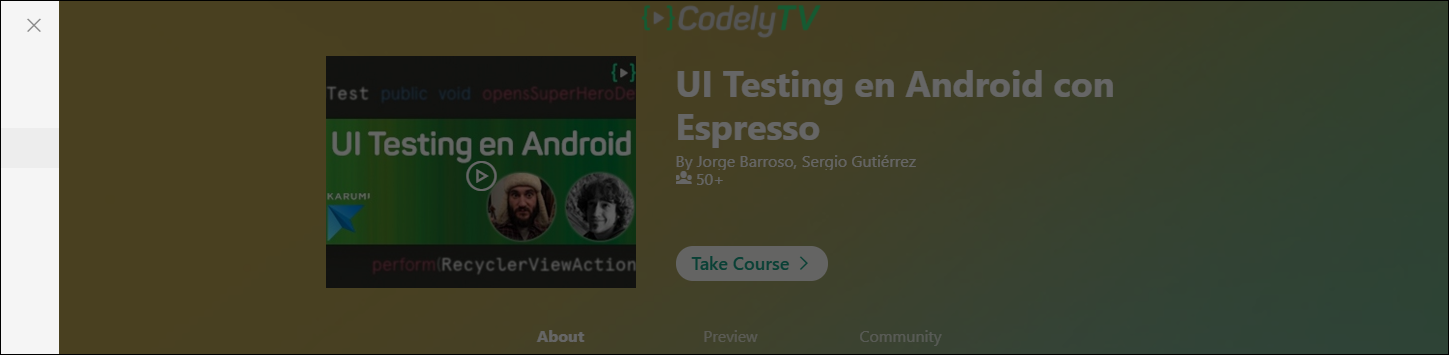


En este caso es un curso online el cual también es gratuito, pero por el contra, aunque realicemos el curso satisfactoriamente no podremos obtener ningún reconocimiento de realización. Para poder encontrar más información visitamos la web que dejo aquí.

URL:

https://www.lynda.com/Android-SDK-tutorials/Alternate-Android-testing-tools-Espresso/478377/512087-4.html

### 5.3.2 Curso gratuito 2 sobre Espresso



Aquí dejo la descripción del curso

¡Primera lección disponible sin registro!

En el mundo móvil la parte más importante y grande que debemos cubrir con tests es la interfaz de usuario. En este curso aprenderemos las claves para escribir buenos tests de UI.   
Para ello utilizaremos Espresso y veremos cuáles son las mejores herramientas y técnicas para escribir tests con esta librería de Google.

También aprovecharemos para explicar problemas comunes a la hora de escribir tests de UI y como lo solucionamos.

Y no podemos hablar de tests sin hablar de integración continua, veremos cómo echar los tests a correr dentro de nuestro ciclo de desarrollo.

Posteriormente podrás realizar el curso de [Screenshot Testing en Android](https://pro.codely.tv/library/screenshot-testing-en-android-f1bc7a25/63018/about/) para tener una visión más global de los tipos de tests que puedes realizar en Android y qué aporta cada uno.

Aprovechamos para recomendarte también el curso "[Exprimiendo las vistas de Android](https://pro.codely.tv/library/exprimiendo-las-vistas-de-android/68169/about/)" para poder sacarle el máximo provecho a los distintos componentes y layouts en Android.

URL:

https://pro.codely.tv/library/ui-testing-en-android-con-espresso/63012/about

# 6. Ayudas económicas para estudiar las tecnologías

Debido a que es una tecnología muy específica para realizar una función de pruebas de aplicaciones en teléfonos móviles, no existe ninguna ayuda económica para estudiar este tipo de tecnologías

# 7. Recursos para implementar las tecnologías

Para ambas herramientas se necesitan los SDK’s de los sistemas operativos a testear. Todos son gratuitos.

## 7.1 Recursos para implementar Appium

### 7.1.1 Recursos gratuitos para implementar Appium

* La propia aplicación Appium es de código abierto, lo que significa que es totalmente gratuita de usar.
* Un repositorio Git, donde se guardan los proyectos testing, se puede usar GitHub, que es una herramienta gratuita.
* Necesitaremos un entorno que se encargue de lanzar las tareas de los testing, como Jenkins, la cual descargara los proyectos de GitHub.

### 7.1.2 Recursos no gratuitos para implementar Appium

* Se necesitan servidores al cual se conectarán los dispositivos en los que se lanzaran las pruebas, en caso de ser iOS, se necesitara un MacOS, ya que son necesarias las librerías OS X.
* En caso de ser Android, se puede usar un servidor de cualquier sistema operativo, Windows, Linux, MacOS, o Android. También se pueden hacer los test sobre máquinas virtuales, pero se recomienda hacerlo sobre dispositivos reales.

## 7.2 Recursos para implementar Espresso

### 7.2.1 Recursos gratuitos para implementar Espresso

* Al igual que Appium, Espresso es de código abierto, por lo que es totalmente gratuita.
* Es recomendable Android Studio.

### 7.2.2 Recursos no gratuitos para implementar Espresso

* El único recurso necesario para esta herramienta serían los dispositivos Android en los cuales se realizarán los test, o en su caso, la máquina virtual.

# 8. Conclusiones

Para esta conclusión recordamos que hemos trabajado buscando información sobre Appium y Espresso.

Espresso se encuentra dentro de la aplicación y está escrito por Google con el único propósito de probar funcionalmente una interfaz de usuario de Android. En términos de estilo de prueba, la descripción de Espresso menciona "pruebas de caja blanca". Como Espresso está dentro de la aplicación, tiene acceso y sabe cómo usar el código que realmente ejecuta la aplicación para realizar pruebas más exhaustivas de cada elemento.

Appium está diseñado para ser una plataforma de prueba multiplataforma. Como tal, una de las ventajas y desventajas se centra en la "prueba de caja negra" que solo prueba lo que se ha expuesto externamente a la aplicación. Con el uso del marco de Android UIAutomator, Appium puede acceder a todos los elementos de la interfaz de usuario que un usuario verá.

Dos de los beneficios de ser externo a la aplicación son la capacidad de validar los datos que entran y salen de la capa de servicio detrás de escena, y la capacidad de ejecutar simulaciones diciendo a la aplicación que el dispositivo ha girado, o que el botón de inicio ha sido empujado.

Si bien Appium y Espresso pueden satisfacer la necesidad de realizar pruebas de UI para su aplicación Android, realmente se reduce al alcance de sus pruebas. Idealmente, podría aprovechar ambos marcos para agrandar la cantidad de pruebas realizadas a la aplicación, pero como sucede con la mayoría de las cosas, es más tiempo del que la mayoría de las personas tiene para invertir por adelantado.

Y he aquí por tanto nuestra conclusión;

Si está seleccionando solo un marco, entonces los desarrolladores que construyan una aplicación nativa de Android que tenga su alcance limitado solo a la aplicación y deseen pruebas de interfaz de usuario completas e integradas, Espresso definitivamente satisfará esa necesidad.

Si las pruebas necesitan ser compatibles con múltiples plataformas (por ejemplo, iOS, híbrido y Android), y necesita validar cómo la aplicación reacciona a factores externos como la rotación de la pantalla y / o desea ejecutar la prueba en paralelo usando un servicio, Appium satisfará mejor sus necesidades.