

Instalador y despliegue de la aplicación móvil

**Breve descripción:**

En este componente, se conocerán los parámetros de desarrollo y ejecución de una “app” móvil. Se estudiarán los elementos que aseguren la calidad del montaje y despliegue de la “app”, identificando, además, fallas e inconsistencias del proceso de desarrollo y funcionamiento; junto con la definición de acciones correctivas y pruebas pertinentes para ofrecer una “app” útil, segura y acorde a necesidades de organizaciones o del mercado.

**Septiembre 2023**

Tabla de contenido

[Introducción 1](#_Toc145347923)

[1. Calidad en desarrollo de “apps” móviles 3](#_Toc145347924)

[1.1. Norma ISO 29119 6](#_Toc145347925)

[1.2. Tipos de pruebas de “software” 7](#_Toc145347926)

[1.3. Pruebas de calidad de “software” 11](#_Toc145347927)

[2. Etapas para realizar pruebas de “software” 15](#_Toc145347928)

[3. Herramientas para pruebas de “apps” móviles 16](#_Toc145347930)

[4. Validación de lineamientos para envió a las “app” store 20](#_Toc145347932)

[4.1. Google Play 20](#_Toc145347933)

[4.2. App store 22](#_Toc145347934)

[Síntesis 24](#_Toc145347935)

[Material complementario 26](#_Toc145347936)

[Glosario 27](#_Toc145347937)

[Referencias bibliográficas 28](#_Toc145347938)

[Créditos 30](#_Toc145347939)

Introducción

Las actividades realizadas por medio de las pruebas de “software” favorecen la entrega de un producto de calidad que cumpla, de manera coherente, con los requerimientos del cliente y de los usuarios. Es importante definir y utilizar pruebas estandarizadas que faciliten la realización y comprensión de las mismas.

Para la elaboración de este componente, se abordaron varios autores conocidos en instalación y despliegue de aplicaciones móviles, de quienes se han citado y referenciado conceptos y ejemplos para los fines educativos de esta materia, en el entendido de que el conocimiento es social y, por lo tanto, es para ser usado por quienes necesitan adquirirlo. Se espera que este documento sea útil para todos, aprendices y lectores en general, que estén interesados en acercarse a asuntos básicos de la programación de aplicaciones para dispositivos móviles.

1. Instalador y despliegue de la aplicación móvil



[**Enlace de reproducción del video**](https://www.youtube.com/watch?v=Lkxr2NRpojA)

|  |
| --- |
| **Síntesis del video: Instalador y despliegue de la aplicación móvil** |
| Instalador y despliegue de la aplicación móvil: En el desarrollo de aplicaciones móviles, se debe contemplar el proceso de calidad en el cual se realizan pruebas de “software”. Este proceso es considerado en las etapas finales, ya que produce datos significativos que contribuyen en la detección y corrección de las inconsistencias, fallas, errores o hallazgos. Estos luego deben ser plasmados en informes de manera clara y concisa. Finalmente, lo anterior permite contribuir al proceso de implementación y despliegue en las plataformas de publicaciones de “apps”. |

# Calidad en desarrollo de “apps” móviles

En las etapas finales del desarrollo de aplicaciones móviles, antes de crear el ejecutable, ha de tenerse seguridad en que el producto que se entrega funciona correctamente y cumple con los estándares de calidad del desarrollo de “software” ISO 9126. Para verificar los requisitos del usuario o los parámetros requeridos por el cliente, modificar la funcionalidad o evitar errores, fallas o anomalías, deben realizarse pruebas de “software” que permitan evaluar el correcto funcionamiento de la aplicación.

Conozca las acciones y aspectos implícitos del proceso de determinación de la calidad del desarrollo y funcionamiento de las aplicaciones móviles.

* **¿Qué es la calidad?.** Es el grado con el que un sistema, componente o proceso cumple con los requisitos especificados y las necesidades o expectativas del cliente o usuario. Se define la calidad como un proceso eficaz de “software” que se aplica de manera que crea un producto útil, que proporciona valor medible a quienes lo producen y a quienes lo utilizan. (Bolaños et al., 2008)
* **Evaluar cada proceso**. Realizar las pruebas de “software” es necesario para ofrecer la posibilidad de evaluar cada uno de los procesos, antes de entregar y desplegar la aplicación en las tiendas, entregando un producto con buenos estándares de calidad. Pero también, antes de enfatizar sobre las pruebas de “software”, es de gran importancia definir algunos parámetros en los estándares de calidad de “software” donde se fundamentan algunas de las pruebas que son realizadas para definir la calidad del producto.
* **Atributo:** actividad verificada y medida. El modelo de calidad ISO 9126 realiza la clasificación de eficacia de “software” dividiendo los temas en categorías, y estas categorías deben cumplir con otras características que se denominan subcategorías, mucho más específicas. Las características, a su vez, están divididas en atributos; y un atributo es una actividad que debe ser verificada o medida en el funcionamiento de la aplicación. Los atributos no están definidos en el estándar, ya que varían entre diferentes productos y desarrollos de “software”.

1. Atributos que abarcan las características de la norma ISO/IEC 9126



La figura representa los atributos que abarcan las características de la norma ISO/IEC 9126, y destaca seis aspectos importantes. El primero es la funcionalidad, que incluye la estabilidad, la exactitud, la interoperabilidad y la seguridad. En segundo lugar, está la usabilidad, que abarca la comprensibilidad, el aprendizaje, la operabilidad y la atracción. En tercer lugar, tenemos el mantenimiento, que se centra en el análisis, el cambio, la estabilidad y las pruebas. En cuarto lugar, se encuentra la confiabilidad, que incluye la madurez, la tolerancia a las fallas y la recuperabilidad. En quinto lugar, está la eficiencia, que se refiere al comportamiento en términos de tiempo y al uso de recursos. Finalmente, en sexto lugar, tenemos la portabilidad, que abarca la adaptabilidad, la instalación, la conformidad y el reemplazo.

* **Pruebas de “software”.** Para cumplir con estos estándares de calidad establecidos en el desarrollo de “software”, es necesario realizar actividades como las pruebas de “software”, donde se pueden identificar las fallas o falencias en cada uno de los parámetros establecidos, generando un informe con los hallazgos y errores, con el fin de solucionarlos a tiempo y lograr realizar la entrega de un producto de calidad. De esta manera, se garantiza que la aplicación logre pasar las pruebas que realizan las tiendas de las aplicaciones y seguir cumpliendo con las directrices de calidad de Apple y Android.
* **Fases de aplicación de las pruebas.** Las pruebas se pueden dividir en cuatro grandes grupos que son: las fases de inicio, planificación, diseño y ejecución. En cada una de estas fases, es posible evaluar e identificar errores. En el desarrollo de aplicaciones móviles, es pertinente realizar este tipo de pruebas y validaciones a tiempo, para no entrar en reprocesos que incrementan los lapsos de producción y los costos del desarrollo de un producto.
* **Por una “app” móvil óptima**. Las pruebas de calidad favorecen la construcción e integración de “software” y la integración de los productos asociados con su correcto funcionamiento, como la integración de las API de terceros. Es preciso utilizar un modelo de pruebas que sea claro y detallado y que se adapte a las necesidades de la aplicación. Las pruebas son elemento básico para alcanzar el desarrollo de una aplicación móvil de calidad; permiten nivelar la estrategia de trabajo, con el fin de lograr una mayor confiabilidad, mantenibilidad y facilidad de las soluciones. (Hesham y Shakeel, 2018)

## Norma ISO 29119

El objetivo de la norma ISO/IEC 29119, pruebas de “software”, es proporcionar una norma definitiva para las pruebas de “software”, que defina el vocabulario, los procesos, documentación, técnicas y un modelo de evaluación del proceso de pruebas de “software” que se puede utilizar dentro de cualquier ciclo de vida de desarrollo (Hesham y Shakeel, 2018).

El principal objetivo para realizar las pruebas es identificar si el “software” hace lo que debe hacer, de la manera correcta. Pero también probar si hace o genera otras actividades como efectos secundarios adversos, descubrir un fallo, una falencia, algún detalle que aún no ha sido identificado y que tiene consecuencias en el correcto funcionamiento de los procesos y actividades, y no sólo de la aplicación que se está evaluando, sino de todo el ecosistema en el cual está trabajando. Lo ideal es encontrar la mayor cantidad de errores en el menor tiempo posible y con la menor cantidad de recursos.

ISO/IEC 29119 consta de 5 partes:

1. Conceptos y definiciones.
2. Procesos de prueba.
3. Documentación de prueba.
4. Técnicas de prueba.
5. Pruebas dirigidas por palabras clave.

## Tipos de pruebas de “software”

Las pruebas de “software” permiten identificar la presencia de defectos en “software”; sin embargo, no garantizan, completamente, que no haya defectos ocultos en el “software”, y el hecho de que no se identifiquen tales defectos, no es una evidencia de que el “software” esté totalmente correcto.

Las pruebas, en conjunto, tienen como objetivo principal realizar una verificación y validación de cada uno de los procesos y actividades realizadas por la aplicación, independientemente de las características y el entorno donde se ejecute.

En el siguiente recurso, conozca los tipos de pruebas de “software” más importantes y profundice en los elementos que aseguran la calidad de las aplicaciones móviles híbridas y de los beneficios que estas ofrecen para organizaciones y usuarios.

1. Tipos de pruebas de “software”



[**Enlace de reproducción del video**](https://www.youtube.com/watch?v=Uf1Kk52ONYc)

|  |
| --- |
| **Síntesis del video: Tipos de pruebas de** “software” |
| Tipos de pruebas de “software”: Las pruebas de “software” permiten identificar la presencia de defectos en él. Sin embargo, no garantiza que no haya defectos ocultos en el “software”. Las pruebas, en conjunto, tienen como objetivo principal realizar una verificación y validación de cada uno de los procesos y actividades realizadas por la aplicación, independientemente de las características y el entorno donde se ejecute.  Estos son algunos tipos de pruebas:   * Funcionalidad: se valida el correcto flujo de entrada y salida, así como los elementos del ecosistema en el que se desarrollan las actividades, identificando de esta forma si los resultados o los efectos son los correctos y acordados para el cliente y los usuarios. * Usabilidad: es el proceso que requiere el usuario para reconocer la lógica de la aplicación en cuanto a consistencia y uso con la aplicación. * Fiabilidad: es la capacidad que tiene el “software” de conservar su funcionalidad en condiciones anormales para determinar el restablecimiento de servicio y recuperación de datos de manera rápida y eficiente durante el periodo de uso. * Rendimiento: la cantidad de tiempo de respuesta que tiene el “software”, registro de eventos o “hardware” para realizar una carga de trabajo promedio con picos altos dentro de los parámetros normales de uso. * Soportabilidad: conjunto de atributos relacionados con la capacidad de un sistema de “software” para ser instalado, transferido y adaptado desde una plataforma a otra, y la posibilidad que tiene esta aplicación de adaptarse a diferentes escenarios de “hardware” y “software”. * Pruebas de componentes: busca realizar la comprobación de funcionamiento de cada uno de los elementos que integra la aplicación, tanto del “software” como de “hardware”, tratando de identificar fallas en los módulos, “software”, programas, objetos, clases, *“*script” que puedan probarse por separado. * Pruebas de integración: busca comprobar la compatibilidad y el correcto funcionamiento de la interfaz de integración entre los componentes o módulos que forman la aplicación, por ejemplo, el componente validación de usuarios con el sistema operativo, el sistema de archivos en integración con el “hardware”, etcétera. * Pruebas de “backup” y restauración de seguridad de la información: se puntualizan sobre la posibilidad de recuperación después de una falla que genere lo que se conoce como un desastre, que no es más que la pérdida de información.   Existen unas pruebas que son exclusivas para las aplicaciones desarrolladas para dispositivos móviles: procedimiento de instalación y desinstalación de aplicación, actividades multitarea alternando con otras funciones del dispositivo móvil, funcionalidad en diferentes tamaños de pantalla con diferentes tipos de resolución y visualización horizontal y vertical.  Comprobar el buen funcionamiento sobre el ecosistema de trabajo identificando la afectación a otras aplicaciones que usan los mismos recursos, actividades de función propias del dispositivo (girar, agitar, gestos táctiles sobre la pantalla y resultados), probar escenarios en los que se presentan interrupciones (ingreso de mensajes, llamadas, conexión y desconexión de cables o cambios de red), fiabilidad de los datos en los escenarios de interrupción del proceso si se cuenta con conexión a bases de datos remotas.   * Pruebas de interrupción en las aplicaciones móviles en algunos eventos que manejan jerarquías que, por lo general, pueden interrumpir y dejar en segundo plano la actividad que se está llevando a cabo en una aplicación (estos eventos, al terminar, deben dar paso a la actividad que estaba en pausa y continuar el proceso), pruebas con la interfaz de usuario (permiten verificar la coherencia de la estructura y el diseño gráfico de la aplicación frente a las necesidades del cliente y el usuario, utilizando los estándares preestablecidos para este tipo de actividades). * Pruebas de acción del usuario es la prueba de usabilidad y accesibilidad de todos los gestos táctiles de la aplicación para determinar la eficiencia, exactitud e interacción del usuario. * Pruebas de movilidad y conectividad se basa en realizar las pruebas de conexión que intervienen en la aplicación, algunas de estas pruebas son: geoposicionamiento, datos tomados y enviados por el GPS, triangulación de la posición, acelerómetro, brújula, desconexión por cambio de celdas de señal móvil o fallas de en comunicación de datos, señal de GPS, “wifi” o “bluetooth” débil o intermitente por ubicación o saturación, red analógica, 3G, 4G o 5G, conectividad con diferente intensidad y en diferentes circunstancias. * Pruebas de compatibilidad (es el tipo de pruebas que se realizan para verificar la compatibilidad y el desempeño sobre diferentes entornos de trabajo y sistemas operativos donde se utilizará la aplicación). |

## Pruebas de calidad de “software”

En el mismo sentido que se aplican todas las pruebas de “software”, se han de aplicar las pruebas de calidad del “software”, las cuales tienen unos pasos a seguir y favorecen una comprensión clara del “software” en términos de diseño, especificaciones, tiempos de trabajo, componentes, ejecución. Estas pruebas de calidad del “software” incluyen una estrategia y un plan de prueba, e incluso, el informe final de la prueba aplicada.

A continuación, afiance sus conocimientos sobre los aspectos que enmarcan y dan rumbo a la aplicación de pruebas de calidad de “software”.

* **Comprensión clara del** “software”**.** El proceso de control de calidad del “software” inicia con la comprensión concreta de la funcionalidad del “software” que se está desarrollando. Esto incluye revisar los requisitos del usuario, los requisitos no funcionales, las especificaciones funcionales, los casos de uso y los documentos que se utilizaron en el levantamiento de información. Los grupos de interés realizan el trabajo definido en horas y días, por ejemplo; y lo hacen con el objetivo de establecer expectativas claras y hacer mejores planes, estableciendo tiempos de entrega acordes con los procesos por desarrollar.
* **Estrategia y plan de prueba**. Se deben formular las siguientes preguntas: ¿Por qué realizar la prueba? ¿Qué se intenta aprender? ¿Cuántas pruebas son suficientes? ¿Cómo organizar el trabajo? ¿Qué tipo de prueba de “software” se desea realizar? ¿Cuánto tiempo se tarda en completar la tarea de prueba? Por lo tanto, se debe definir el alcance, los recursos necesarios, el entorno de prueba, el cronograma, el administrador, la metodología de gestión de riesgos y otras partes consideradas necesarias.
* **Componentes del proceso**. Con base en la estructura de componentes establecida para el “software” en desarrollo, se determina qué pruebas de calidad de “software” son apropiadas para cada “software”. Todos los resultados del ciclo de desarrollo deben someterse a revisiones de calidad, no sólo el código. Es decir, especificaciones funcionales, diseño técnico, manuales de usuario y otra documentación. El proceso incluye un diseño de caso de prueba para cada componente relacionado. Los casos de prueba deben corresponder a posibles escenarios de producción e incluir las variables requeridas. Las pruebas diseñadas incluyen pruebas de caja negra y pruebas de caja blanca.
* **Ejecución de prueba.** Esto permite realizar pruebas de calidad de “software” diseñadas y estructuradas, lo que le permite clasificar el equipo de prueba por componente o dividirlo en paquetes que serán asignados a grupos especializados en el tema. Debe haber una visibilidad de casos y problemas que le impida ejecutar otros casos de prueba y realizar acciones. Además, situación diferente con el ambiente de desarrollo y de pruebas (como errores de no disponibilidad o no relacionados con el desarrollo) debe informarse a sus respectivos equipos.
* **Informe de progreso**. Esto incluye mantener informes regulares sobre el progreso de las pruebas. Reporta casos de prueba totales, ejecución de casos de prueba y casos exitosos diariamente o varias veces al día, para proyectos críticos. Casos exitosos, casos pendientes, casos con fallas, número de incidencias y otros aspectos. Existen herramientas de gestión de pruebas de “software” que automatizan la publicación de este tipo de informes: informes de incidente automatizados.
* **Reportar las incidencias.** Es necesario tener un informe de incidente registrado. Esto incluye, al menos, la fecha y la hora, un título descriptivo, una descripción a detalle de las fallas detectadas y las inconsistencias, la ubicación y el entorno en que se detectó (incluidos los usuarios y roles que verifica) y las señales de error, si hay una ubicación, y pasos para ubicarla de manera sencilla, con el fin de corregirla.
* **Registro de procesos y elaboración informe final.** Incluye la finalización del registro y organización de las evidencias de prueba y otra documentación de los casos de prueba ejecutados. Una vez finalizada la prueba, se creará un informe final que describe la prueba realizada, incluyendo casos, resultados, problemas encontrados, lecciones aprendidas y otros aspectos que permitirán tener la trazabilidad de las fallas, errores e inconsistencias, para realizar su pronta corrección.

# Etapas para realizar pruebas de “software”

Para realizar una prueba rigurosa a las aplicaciones móviles es necesario contar con un orden y estructura que permitan cumplir con un proceso acorde con las necesidades y exigencias del proyecto. Se debe analizar cada una de las fases del proceso para lograr buenos resultados, identificando el flujo de trabajo que se debe seguir para encontrar los fallos, errores y falencias, antes de entregar el producto a los usuarios.

Las fases o etapas de realización de las pruebas de “software” son: fase de inicio, fase de planeación, fase de diseño, fase de ejecución, fase de cierre o finalización; con una fase transversal de monitoreo y control (Victoria y Ramírez, 2017).

### Etapas Pruebas de “software”

Para una mejor comprensión del flujo de trabajo en la aplicación de las pruebas de “software”, descargue el PDF denominado **Anexo-Etapas-Pruebas-de-software** que se encuentra en la carpeta de Anexos y analice el esquema que allí se presenta. Recuerde llevar registro de los elementos más destacados de este punto.

# Herramientas para pruebas de “apps” móviles

Para realizar pruebas de funcionalidad y calidad de “software”, existe una gran cantidad de herramientas que ofrecen tal servicio y que aportan a la ejecución de esta actividad, facilitando entornos de evaluación de cada uno de los módulos y componentes de las aplicaciones móviles. En su mayoría, estos instrumentos son de pago y requieren gran cantidad de esfuerzo del grupo de trabajo para lograr un óptimo resultado.

A continuación, se presentan algunas herramientas reconocidas y muy usadas en los procesos de prueba de aplicaciones móviles.

* **Herramientas de pruebas.** Herramientas de pruebas más populares.

Uno de los más populares y utilizados es Testim. Se trata de una herramienta de prueba de “software” que utiliza el aprendizaje automático y soporta pruebas para dispositivos móviles. Otro instrumento es BlazeMeter, que se utiliza para ejecutar casos de prueba de rendimiento en aplicaciones móviles, API o aplicaciones web, para evaluar el rendimiento en cualquier etapa del desarrollo de la aplicación. Otra herramienta muy popular entre los desarrolladores de “software” es Selenium WebDriver Mobile. Uno de sus grandes inconvenientes es que sólo simula Android. Para solucionar este problema se generó un “framework Appium” que permite trabajar con Windows Phone, iOS y Android.

* **AWS “Device Farm”.** Es un servicio de prueba de aplicaciones que permite mejorar la calidad de las páginas web y aplicaciones móviles al realizar pruebas en diferentes navegadores de escritorio y diferentes sistemas operativos móviles sin tener que configurar ni administrar la infraestructura de prueba. Este servicio permite ejecutar pruebas en varios navegadores de escritorio y dispositivos reales, que aceleran la configuración y realización de pruebas, obteniendo los resultados en videos y registros para ayudar a identificar, rápidamente, problemas en su aplicación.
* **Herramienta “TestDroid”.** Esta es una herramienta en línea, de la compañía TestMatick, para pruebas de interfaz de usuario, programadas en diferentes dispositivos Android. “TestDroid” permite a los desarrolladores e investigadores documentar los “scripts” de prueba, los cuales se ejecutan automáticamente en paralelo con la aplicación, en diferentes dispositivos. Esta herramienta informa los resultados de estas pruebas y permite a los desarrolladores e investigadores identificar rápidamente las versiones de Android donde las aplicaciones pueden no funcionar. Al mismo tiempo, la herramienta se puede utilizar para identificar una amplia gama de problemas y frecuentes errores de programación asociados con cada dispositivo.

1. Pasos para la ejecución de pruebas automatizadas



La figura representa los pasos para realizar pasos de ejecución de pruebas automatizadas. En el primer paso, se crea un proyecto para llevar a cabo las pruebas automatizadas. El segundo paso implica subir la aplicación a la plataforma e interactuar con la interfaz. En el tercer paso, se selecciona el tipo de prueba a ejecutar. El cuarto paso consiste en elegir un dispositivo específico para las pruebas. El quinto paso implica especificar el estado de las conexiones, como el “WiFi” y el “Bluetooth”, y comprobar el tiempo de ejecución de la prueba y comenzar.

### “TestDroid” permite específicamente:

1. Ejecutar pruebas en un dispositivo específico en el que el cliente desea usar la aplicación.
2. Generar un informe en PDF de las pruebas realizadas por el dispositivo.
3. Evaluar la portabilidad y la compatibilidad para brindar la capacidad de realizar pruebas en una variedad de dispositivos móviles.
4. Evaluar el consumo de CPU y RAM de dispositivos móviles.
5. Evaluar el tiempo de respuesta de la aplicación, el consumo de batería, la resolución y el tamaño de la pantalla.

Contras:

1. No tiene control sobre la movilidad del usuario. Comprobación de la conexión a la red.
2. Debido a que la herramienta está en línea, los errores de conexión de red pueden afectar el rendimiento de la aplicación bajo prueba.

# Validación de lineamientos para envió a las “app store”

Las aplicaciones originales de iPhone y Android sólo se pueden distribuir a través de sus respectivas tiendas en línea (“app store”). La "app" store tiene una serie de pautas que sus aplicaciones deben seguir. Por lo tanto, al probar aplicaciones móviles, debe incluir casos de prueba para verificar el cumplimiento de estas pautas.

Para identificar estos casos de prueba, se deben consultar las condiciones impuestas por Apple para dispositivos iOS como iPhones y iPads, así como las pautas requeridas por Google para teléfonos y tabletas Android.

## Google Play

Android ofrece muchas opciones para la distribución de aplicaciones. Hay otras tiendas alternativas, pero Google Play es la plataforma principal para distribuir aplicaciones de Android. Lo primero que se debe realizar es crear una cuenta de desarrollador de Google Play para tener acceso al perfil de desarrollador y publicar la aplicación en Play Store. Esta suscripción tiene un costo que, para el año 2021, no supera los US$25 y con esa primera suscripción se tendrá acceso, de por vida, para publicar todas las aplicaciones que se quiera.

La información suministrada durante el registro aparecerá más tarde en la tienda. Cuando el usuario final navegue por la aplicación en Play Store, podrá obtener información sobre la identidad del desarrollador. La información proporcionada por Play Store es la información que declaró cuando se abrió la cuenta de editor de Google.

Conozca, ahora, aspectos de suma importancia en el proceso de validación y distribución de aplicaciones en Google Play.

1. **Google Play**. Google Play es una biblioteca que permite a los usuarios navegar y buscar aplicaciones según sus intereses. Cuando un usuario abre una lista de aplicaciones, sólo toma unos segundos determinar si satisface sus necesidades. Por esa razón es tan importante el ícono de la aplicación; este debe llamar la atención y hacer que sus anuncios sean atractivos para que los usuarios descarguen la aplicación.
2. **Texto descriptivo.** El texto de descripción de la aplicación es fundamental para garantizar el éxito en el reconocimiento y aceptación de los usuarios. Dicho texto descriptivo debe seguir algunos criterios y parámetros de orden técnico y publicitario. Estos son los detalles que se muestran a los clientes en Google Play.
3. **Descripción breve y capturas de pantalla.** La descripción breve es el texto del anuncio. Es el primer texto que visualiza el usuario y describe la aplicación. La descripción larga utiliza las palabras clave adecuadas para describir con precisión la aplicación, esta debe atraer a los espectadores y optimizar el proceso de adquisición de la aplicación por curiosidad o por servicio. Son importantes también las capturas de pantalla adecuadas para mejorar la imagen de la marca y mostrar las mejores funciones de su aplicación.
4. **Publicación de la aplicación**. Después de completar estos pasos, se puede descargar el binario (archivo .apk) y seguir las instrucciones en la consola de Google para iniciar la instancia de publicación de la aplicación. APK (Android Package Kit) es un formato de archivo que utiliza el sistema operativo Android para instalar y ejecutar aplicaciones. Un archivo APK contiene todos los elementos que la aplicación necesita para funcionar efectivamente en un dispositivo Android. Para realizar el proceso de publicación de la aplicación es necesario realizar el proceso de registro como desarrollador y realizar el pago correspondiente.
5. **Condiciones y requerimientos.** Es importante realizar el paso a paso, como lo indica Google, para lograr tener éxito. Es necesario seguir al pie de la letra cada una de las condiciones y requerimientos. Después de realizar el procedimiento, Google y su equipo de verificadores realizan las pruebas necesarias a la aplicación, comprobando si cumple con los requerimientos mínimos para ser publicada; esto puede tardar entre una o tres semanas aproximadamente, dependiendo de la cantidad de aplicaciones que están en proceso de aprobación.

## App store

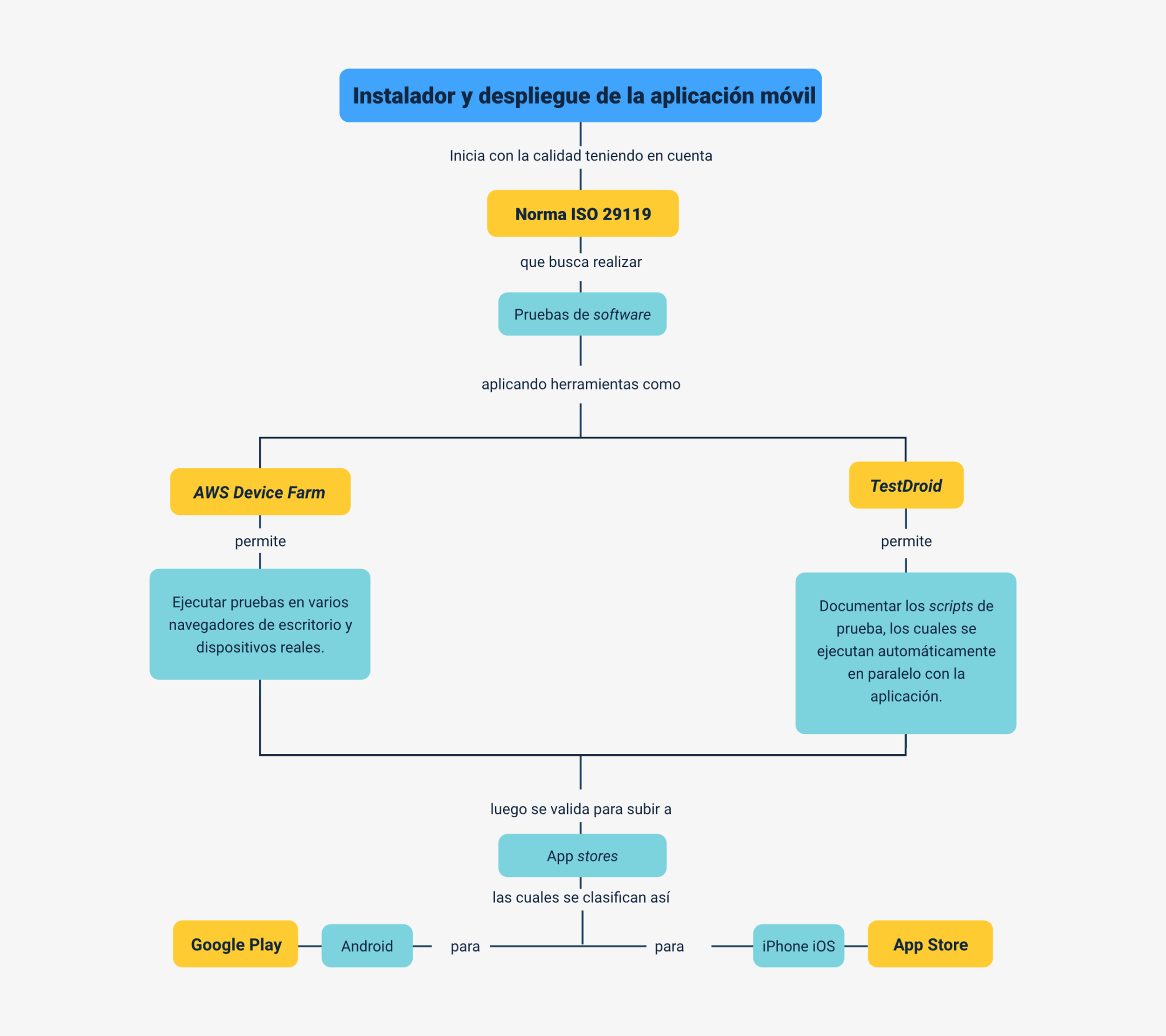
El proceso de publicación de aplicaciones en la App Store requiere una preparación adecuada. Con más de dos millones de aplicaciones en la App Store, Apple ha establecido altos estándares y pautas estrictas para mantener este ecosistema seguro para sus usuarios y ofrecer, constantemente, aplicaciones de alta calidad.

A continuación, reconozca elementos de suma importancia que le orientarán sobre el proceso de publicación de aplicaciones en la App Store.

1. **Cuenta de desarrollador de Apple.** De la misma manera que en Google Play es necesario realizar un registro, se requiere hacerlo como desarrollador de Apple para crear aplicaciones iOS y publicarlas en la respectiva App Store. Esta cuenta tiene un valor, para el año 2021, que no supera los US$99 y debe renovarse anualmente. Si no se renueva, se corre el riesgo de que las aplicaciones vinculadas a la cuenta se desvinculen, por lo cual es recomendable hacer los pagos de manera oportuna. Unirse a este programa para desarrolladores también le permite crear aplicaciones iOS y probarlas en su dispositivo.
2. **Envío a la App Store.** Es importante leer las recomendaciones y las pautas de revisión de la App Store antes de comenzar el proceso de envío. De hecho, el equipo de revisión de Apple, por ejemplo, revisa cada aplicación antes de aprobar la versión. La mayoría de las negativas de Apple obedecen al incumplimiento de los conceptos básicos.
3. **App Store Optimization (ASO).** Es necesario proporcionar toda la información requerida, como íconos, vistas previas de aplicaciones / capturas de pantalla y metadatos, nombre de la aplicación, categoría, descripción detallada y palabras clave, ASO adicionales (procesos de optimización de las “apps”), antes de enviar la solicitud para revisión.
4. **Detalles de impacto y persuasión.** La primera frase (las tres primeras líneas) son las más importantes. Esto se debe a que los detalles de la aplicación se pueden confirmar sin que el usuario la descargue. En este paso es muy importante ser específicos, claros y concisos para transmitir el valor fundamental de la aplicación a los usuarios. El nombre de la cuenta de desarrollador debe ser el mismo que el del propietario de la aplicación. Este es el nombre que aparecerá debajo de la aplicación en la tienda.

Síntesis

A continuación, se presenta a modo de diagrama una síntesis de lo desarrollado en este componente formativo.



La figura representa la síntesis del componente formativo relacionado con la instalación y lanzamiento de aplicaciones, enfocándose en la calidad según la norma ISO 29119. Este proceso busca realizar pruebas de “software” utilizando herramientas como AWS “Device Farm”, que permite ejecutar pruebas en múltiples navegadores, escritorios y dispositivos reales. También se utiliza “TestDrive” para documentar los scripts de prueba, los cuales se ejecutan automáticamente en paralelo con la aplicación. Una vez validado todo, las aplicaciones se suben a las tiendas correspondientes: Google Play para Android y App Store para iPhone.

Material complementario

|  |  |  |  |
| --- | --- | --- | --- |
| Tema | Referencia | Tipo de material | Enlace del recurso |
| Calidad en desarrollo de apps móviles | Tech Community Day. (2020, 13 junio). Guillermo Zepeda - Despliegue de Aplicaciones Móviles con AppCenter [Video]. YouTube. | Video | <https://www.youtube.com/watch?v=SC1sRe3h-Qs> |
| Validación de lineamientos para envío a las app stores | Moure, B. [MoureDev by Brais Moure]. (2019, 24 octubre). APP STORE Cómo PUBLICAR tu primera APP | Tutorial Español | MoureDev | Video | <https://www.youtube.com/watch?v=tg6YcD52jNA> |
| Google Play | Luján, E. D. [EDteam]. (2020, 16 junio). ¿Cómo subir tu app a la Play Store? [Video]. YouTube. | Video | <https://www.youtube.com/watch?v=EOkKc1Oc92M> |

Glosario

**APK:** es el formato de una aplicación desarrollada exclusivamente para el sistema operativo de Google, Android. APK es el acrónimo de “Android Application Package” y, en comparación, es como los archivos .EXE en Windows.

**ASO:** proceso de optimizar aplicaciones móviles para tener una posición más alta en las tiendas y es descrito frecuentemente como el SEO de las “apps”. Es el proceso de optimización para lograr una aceleración en la visibilidad de la aplicación e incrementar descargas.

**“Backup”:** es una copia de seguridad o una copia de respaldo que se realiza sobre los datos originales, con el fin de contar con un elemento que permita la recuperación de los datos, en caso de su pérdida.

**ISO:** es la Organización Internacional de Normalización, una institución encargada de la creación de estándares internacionales, compuesta por diversas organizaciones nacionales de normalización.

**Soportabilidad:** también se puede encontrar como "portabilidad" es una de las definiciones utilizadas en programación de alto nivel. Se define como la característica que posee un “software” para ejecutarse en diferentes plataformas.

Referencias bibliográficas

Bolaños, D., Sierra, A., y Alarcón, M. (2008). *Pruebas de Software y JUnit*. Pearson Prentice Hall.

Correa, M. (2013). *Composición del sistema operativo móvil IOS de Apple y el hardware y software que lo utilizan.* Universidad Católica de Pereira. <https://docplayer.es/1521033-Composicion-del-sistema-operativo-movil-ios-de-apple-y-el-hardware-y-software-que-lo-utilizan-melissa-correa-rios.html>

Hesham, A. & Shakeel, A. (2018). *Overview of Software Testing Standard ISO/IEC/IEEE 29119*. International Journal of Computer Science and Network Security, 18(2), p. 112-116. <https://www.researchgate.net/profile/Shakeel-Ahmed-46/publication/323759544_Overview_of_Software_Testing_Standard_ISOIECIEEE_29119/links/5e5923c84585152ce8f64d83/Overview-of-Software-Testing-Standard-ISO-IEC-IEEE-29119.pdf>

ISO 25000. (s.f.). *ISO/IEC 25040*. <https://iso25000.com/index.php/normas-iso-25000/iso-25040?start=3>

Minguet, J. y Cueva, J. (2003). *La calidad del software y su medida*. Editorial Universitaria Ramón Areces.

Noriega, D., Gabino, N. y Hernández, C. (2007). *Calidad de Software*. Universidad de Matanzas "Camilo Cienfuegos" <http://www.monografias.com/trabajos59/calidad-software/calidad-software.shtml>

Puglla, G. (2017). *Marco de trabajo para control de calidad en el desarrollo de aplicaciones móviles bajo metodologías ágiles*. Universidad Técnica Particular de Loja. <https://docplayer.es/91113144-Universidad-tecnica-particular-de-loja-la-universidad-catolica-de-loja-area-tecnica.html>

Victoria, I. & Ramírez, F. (2017). *Definición de un proceso de pruebas para pequeñas organizaciones que desarrollan aplicaciones para móviles*. Universidad del Cauca. <http://repositorio.unicauca.edu.co:8080/bitstream/handle/123456789/1743/DEFINICI%C3%93N%20DE%20UN%20PROCESO%20DE%20PRUEBAS%20PARA%20PEQUE%C3%91AS%20ORGANIZACIONES%20QUE%20%20DESARROLLAN%20APLICACIONES%20PARA%20M%C3%93VILES.pdf?sequence=1&isAllowed=y>

Créditos

| Nombre | Cargo | Regional y Centro de Formación |
| --- | --- | --- |
| Claudia Patricia Aristizábal | Líder del Ecosistema | Dirección General |
| Rafael Neftalí Lizcano Reyes | Responsable de Línea de Producción | Regional Santander -Centro Industrial del Diseño y la Manufactura |
| Miroslava González Hernández | Diseñadora Instruccional | Regional Santander - Centro Industrial del Diseño y la Manufactura |
| Flor Tinoco Torres | Diseñador de Contenidos Digitales | Regional Santander - Centro Industrial del Diseño y la Manufactura |
| Camilo Andrés Bolaño Rey | Desarrollador Fullstack | Regional Santander - Centro Industrial del Diseño y la Manufactura |
| Daniela Muñoz Bedoya | Animador y Producción audiovisual | Regional Santander - Centro Industrial del Diseño y la Manufactura |
| Camilo Andrés Bolaño Rey | Locución | Regional Santander - Centro Industrial del Diseño y la Manufactura |
| Camilo Andrés Bolaño Rey | Actividad Didáctica | Regional Santander - Centro Industrial del Diseño y la Manufactura |
| María Ruiz Torres | Validador de Recursos Educativos Digitales | Regional Santander - Centro Industrial del Diseño y la Manufactura |
| Luis Gabriel Urueta Álvarez | Validador de Recursos Educativos Digitales | Regional Santander - Centro Industrial del Diseño y la Manufactura |
| Daniel Ricardo Mutis Gómez | Evaluador para Contenidos Inclusivos y Accesibles | Regional Santander - Centro Industrial del Diseño y la Manufactura |