

Diseño y construcción de “Frontend”

**Breve descripción:**

En este componente formativo se abordan los conceptos clave para el diseño y desarrollo de aplicaciones móviles nativas, como sus elementos, vistas, maquetación, entornos de desarrollo, lenguajes entre otros.

**Septiembre 2023**

Tabla de contenido

[Introducción 1](#_Toc145420455)

[1. Análisis de requerimiento 3](#_Toc145420456)

[2. Diseño gráfico: herramientas de prototipado 8](#_Toc145420457)

[2.1. Figma 8](#_Toc145420458)

[2.2. Sketch 9](#_Toc145420459)

[2.3. Adobe XD 11](#_Toc145420460)

[2.4. Conceptos de “Material Design” 13](#_Toc145420461)

[3. Principios básicos de usabilidad 17](#_Toc145420462)

[4. Componentes de “software” 20](#_Toc145420463)

[4.1. Licenciamiento 20](#_Toc145420464)

[4.2. Seguridad 22](#_Toc145420465)

[4.3. API’s, Firebase 25](#_Toc145420467)

[4.4. Herramientas 30](#_Toc145420469)

[Síntesis 34](#_Toc145420470)

[Material complementario 35](#_Toc145420471)

[Glosario 36](#_Toc145420472)

[Referencias bibliográficas 37](#_Toc145420473)

[Créditos 38](#_Toc145420474)

Introducción

Aquí comienza el estudio del componente formativo Diseño y construcción de “Frontend”; explore la información del video que se muestra enseguida, la cual le contextualiza sobre los aspectos más importantes de los temas por desarrollar. ¡Adelante!

1. Diseño y construcción de “Front-end”



[**Enlace de reproducción del video**](https://youtu.be/11eAGCTwixQ?feature=shared)

|  |
| --- |
| **Síntesis del video:** Diseño y construcción de “Front-end” |
| Diseño y construcción de “frontend”: siempre que se aborda un objeto sin la apropiada planeación, se corre el riesgo de hacer un análisis menos efectivo. Lo que puede conducir a una brecha entre el producto entregado y lo esperado por el cliente o los usuarios finales. Al desarrollar aplicaciones móviles, esto es una realidad que con facilidad se puede encontrar, debido a que los móviles proponen otras y nuevas formas de interacción con el usuario. Cualquier frustración del usuario o el cliente termina convirtiéndose, en el mejor de los casos, en un sobreesfuerzo o incumplimiento de los tiempos programados.  Establecer correctamente los requisitos del sistema permite realizar un correcto diseño de la integración con el usuario, lo cual permite dimensionar adecuadamente el coste en tiempo y permite potenciar la inversión. Es por esto que se recomienda dos cosas previas a la realización del código:  La primera, es determinar correctamente los requerimientos, es decir, el alcance del sistema; y la segunda actividad es realizar un diseño o prototipado del sistema de tal forma que provea una visión clara de la interacción del sistema con el usuario. |

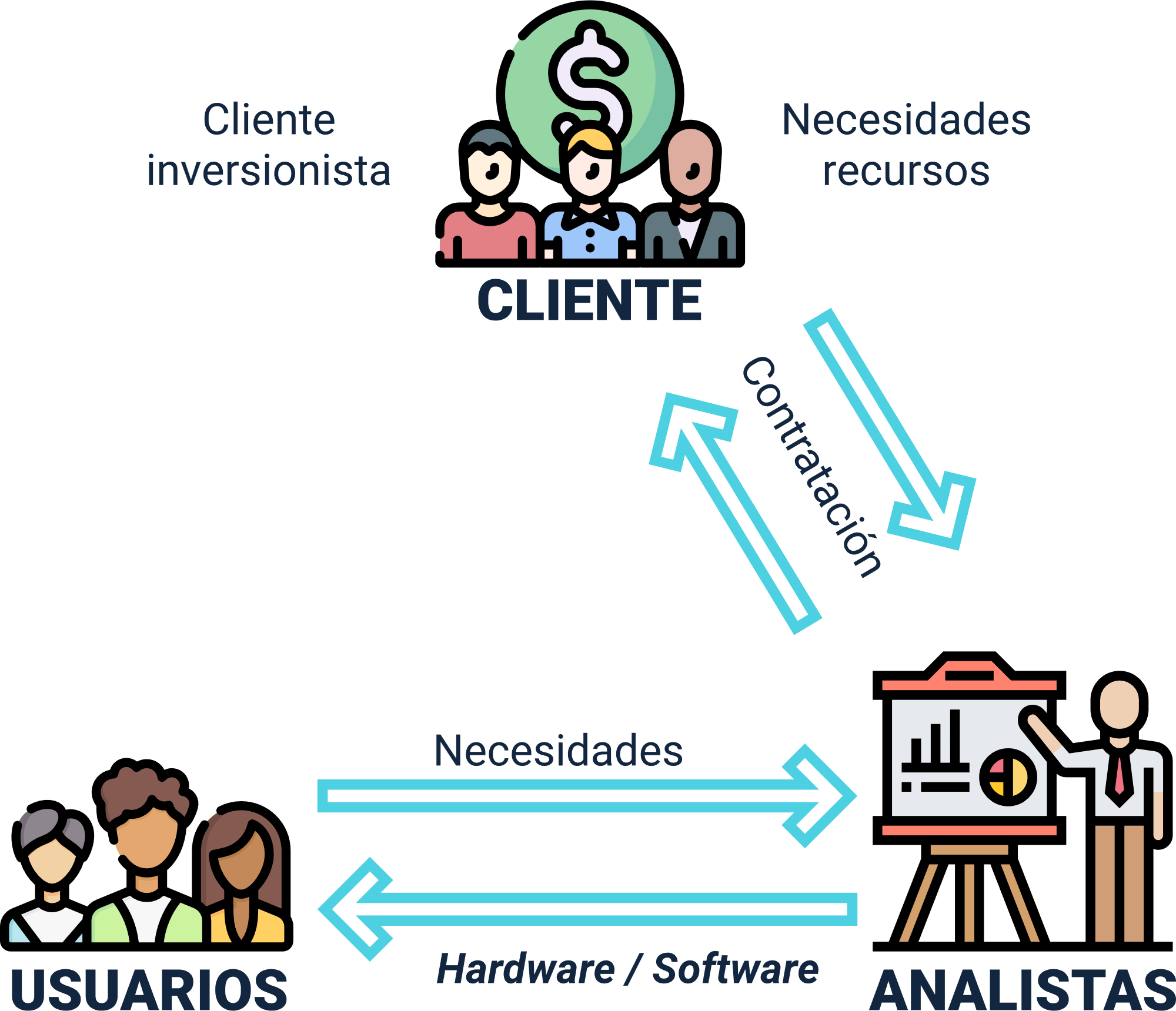
# Análisis de requerimiento

El desarrollo de aplicaciones móviles no implica únicamente la codificación o programación, también requiere de analizar y entender las necesidades del cliente, crear modelos que hagan más fácil el proceso de desarrollo, la planeación del proyecto, y otras actividades involucradas.

La aplicación de las técnicas de elicitar (transferir información) requisitos, conducen a identificar las fuentes de los mismos con los interesados en el sistema (“stakeholders”). Estas actividades sin embargo pueden derivar en que el cliente o los usuarios excedan lo que realmente necesitan, o tal vez sin considerar las limitantes a nivel de “hardware”, “software”, presupuesto, tiempo o requisitos, por lo tanto es necesario conocer con todo nivel de detalle lo que se requiere alcanzar y las necesidades reales a cubrir. De esta forma se deben detallar y modificar los requerimientos básicos elicitados, lo cual se logra en un consenso de las partes involucradas como lo indica la siguiente figura 1 y da como resultados la elaboración de un documento en el cual se debe responder:

¿Qué es lo prioritario?, ¿qué es lo esencial? y ¿para cuándo se requiere? Esto se especifica de algún modo y luego se revisa o válida para garantizar la coincidencia en la comprensión que tiene el analista del problema y los demás participantes.

1. Esquema de acuerdos



Nota. Esquema de acuerdos, SENA (2021).

Cuando los sistemas no son tan complejos en su alcance, como lo es en la mayoría de los casos para aplicaciones móviles, se recomienda el empleo de historia de usuario, sin embargo, se propone revisar algunos aspectos que no son fáciles de identificar en un proceso de licitación de requisitos y que se mencionan a continuación ya que está enmarcados en el contexto de aplicaciones móviles, ver tabla 1.

1. Preguntas de análisis de requisitos para aplicaciones móviles.

|  |  |
| --- | --- |
| Aspecto | Pregunta modeladora |
| Aspecto físico o de “hardware” | ¿En qué tipo de dispositivos la “app” debe funcionar? Móvil, *Tablet*, *Smart watch*, etc.  ¿Qué tipo de “app” es una mejor solución?: nativa, web o híbrida  En las “apps” nativas, ¿qué sistema operativo se empleará? Android, IOS, Windows Phone, etc.  En las Web “app”, ¿qué navegadores y versiones deberán soportar las funcionalidades?  ¿A cuántos usuarios simultáneamente debe atender el sistema?  ¿Hay restricciones de nivel de acceso a Internet (ancho de banda o consumo) o aspectos de seguridad? |
| Interface de usuario | ¿Las entradas de información es proveída por los usuarios o existen otras fuentes, cuáles?  ¿La salida o consumo de información intervienen a uno o a más sistemas?  ¿Hay alguna forma preestablecida en que deben formatearse los datos?  ¿Existe alguna aplicación similar que sirva de referencia para su proyecto? |
| Aspectos humanos | ¿Qué tipo de población usará la “app”?  ¿Existen varios tipos de usuario?  ¿Qué tipo de entrenamiento necesitará cada tipo de usuario? |

Nota. Preguntas modeladoras, SENA (2021).

Cuando un requisito o conjunto de ellos hace parte de todo un ecosistema tecnológico donde intervienen diferentes tipos de aplicaciones (web, de escritorio, móvil, interacción con otros sistemas de proveedores, clientes o asociados), como, por ejemplo, los sistemas bancarios, que tienen “app” móvil, aplicación web, aplicación de terminal escritorio, aplicaciones portales para asociados etc. En un escenario así, las historias de usuario no siempre dan una visión completa de la complejidad del problema.

Estos sistemas generalmente tienen lo que denominaban “Las Reglas del Negocio” o “Conjunto de Reglas de Negocio” las cuales describe las políticas, operaciones, normas, definiciones y restricciones presentes en una organización y que los sistemas de información o aplicación deben considerar porque son las que permiten alcanzar los objetivos misionales del cliente. En ocasiones hasta se requiere un glosario de términos propios del negocio para poder especificarlos.

Este es un ejemplo de regla de negocio: un cliente al que se factura más de 10.000 al año es un cliente de tipo A. Para los clientes de tipo A se aplican descuentos del 10% en pedidos superiores a 3.000.

Pero las reglas de negocio pueden ser tan complejas como lo sean las operaciones del cliente, en casos como tal se propone emplear artefactos como los casos de uso para definir al detalle la especificación del sistema.

En un caso tal se proponen algunos artefactos a elaborar que guiarán el proceso:

#### Documento de especificación de requisitos

Permite identificar el objetivo general de cada requerimiento sin abordar demasiados detalles técnicos ni de reglas de negocio que se presenten en la operación, es una base para estimar costos y tiempos.

Revise el contenido del PDF **FS-DOC-LevantamientoRequerimientos** ubicado en la carpeta Anexos. Podrá encontrar un formato de ejemplo, que es a su vez, una guía de cómo crear o adaptar.

#### Plantilla de casos de uso

Alguno de los requerimientos especificados definidos en el documento de especificación de requisitos puede necesitar un nivel de detalle técnico superior, por ejemplo, al definir procesos síncronos o asíncronos de operación del sistema informático o casos particulares de operación, donde pueden intervenir reglas de negocio. A continuación se puede encontrar un ejemplo o guía, para que se pueda adaptar a las necesidades del proyecto.

Revise el contenido del PDF **FS-DOC-PLANTILLA-CASOS-USO** ubicado en la carpeta Anexos. Allí encontrará un formato de ejemplo que sirve como guía para crear o adaptar.

Los requerimientos de una aplicación móvil para la mayoría de los casos están cubiertos por técnica como la elaboración de historias de usuario, sin embargo, cuando la aplicación móvil pertenece a un ecosistema de solución tecnológica más compleja, el proceso requiere algún artefacto que ayude a identificar y validar los requisitos.

# Diseño gráfico: herramientas de prototipado

Un diseñador “front-end” necesita contar con recursos imprescindibles para realizar su trabajo, ya que el primer paso, generalmente es realizar una propuesta de interfaz gráfica o prototipo de interfaz de usuario (UI) para que el cliente pueda imaginar cómo se verá el sistema terminado. Actualmente no es suficiente diseñar la versión para la computadora de escritorio o el prototipo para el móvil. Ahora es necesario diseñar tres versiones distintas, escritorio, “tablet” y móvil, para que se pueda notar que el producto digital conserve o prescinda de ciertas propiedades visuales según el dispositivo en el que funciona. En orden a esta necesidad se presentan algunas herramientas que se pueden emplear para tal fin.

## Figma

Figma es una herramienta de diseño colaborativo en línea que se utiliza para crear prototipos, diseños de interfaz de usuario (UI), diseño de experiencia de usuario (UX) y más. Figma se ha convertido en una opción popular para diseñadores y equipos de diseño de todo el mundo debido a sus características y ventajas únicas.

A continuación, se describen sus principales características y ventajas:

* **Colaboración en tiempo real:** Figma permite a varios usuarios trabajar en el mismo proyecto de diseño en tiempo real. Esto facilita la colaboración entre diseñadores, desarrolladores y otros miembros del equipo, ya que pueden ver los cambios en tiempo real y comentar sobre ellos.
* **Prototipado interactivo**: Figma permite crear prototipos interactivos de alta fidelidad. Puede agregar enlaces, transiciones y animaciones a tus diseños para simular la experiencia del usuario final.
* **Bibliotecas y componentes**: Figma permite a varios usuarios trabajar en el mismo proyecto de diseño en tiempo real. Esto facilita la colaboración entre diseñadores, desarrolladores y otros miembros del equipo, ya que pueden ver los cambios en tiempo real y comentar sobre ellos.
* **Accesible desde cualquier lugar**: Figma es una aplicación basada en la nube, lo que significa que puedes acceder a tus diseños desde cualquier dispositivo con una conexión a Internet.
* **Versionamiento y control de cambios**: Cuenta con funciones de versionamiento y control de cambios.
* **Integraciones y complementos:** Se integra con muchas otras herramientas populares, como Slack, Trello y Jira.
* **Soporte para múltiples plataformas:** Figma es versátil y es adecuado para diseñar aplicaciones móviles, sitios web, interfaces de escritorio y más.
* **Seguridad y privacidad**: Cuenta con medidas de seguridad robustas y opciones de control de acceso.

Para cualquier acción de descarga, instalación, gestión o aprendizaje de uso de Figma, puede emplear el enlace de su sitio oficial: <https://www.figma.com/>

## Sketch

Sketch es una aplicación de diseño gráfico vectorial ampliamente utilizada por diseñadores, ilustradores y profesionales creativos para crear interfaces de usuario, ilustraciones, diseños de sitios web, iconos y otros elementos visuales.

Sketch era una aplicación de diseño exclusiva para macOS. Sin embargo, es importante destacar que la disponibilidad de aplicaciones puede cambiar con el tiempo debido a actualizaciones y decisiones de los desarrolladores.

A continuación, se describen sus principales características y ventajas:

* **Diseño de interfaz de usuario (UI):** Sketch se destaca en el diseño de interfaces de usuario para aplicaciones móviles y sitios web. Permite a los diseñadores crear diseños responsivos y componentes reutilizables para agilizar el proceso de diseño de interfaces.
* **Diseño de iconos e ilustraciones**: Sketch es una herramienta versátil para crear iconos y ilustraciones vectoriales. Su capacidad de trabajar con vectores garantiza que los gráficos sean escalables y de alta calidad.
* **Colaboración**: aunque Sketch no es una herramienta de diseño colaborativo en tiempo real como algunas otras, permite a los diseñadores trabajar en equipo utilizando herramientas de terceros y compartir archivos fácilmente.
* **Accesible desde cualquier lugar:** facilidad de uso: La interfaz es intuitiva y fácil de usar, lo que facilita un flujo de trabajo eficiente.
* **Versionamiento y control de cambios:** símbolos y estilos compartidos: permite crear símbolos y estilos compartidos, lo que facilita la actualización de elementos en todo el proyecto.
* **Integraciones y complementos:** integración con complementos y recursos: Sketch es altamente extensible mediante complementos y recursos adicionales.
* **Soporte para múltiples plataformas**: exportación eficiente: Ofrece opciones flexibles de exportación, como PNG, SVG y otros.
* **Seguridad y privacidad**: **Compatibilidad con dispositivos Apple:** Está optimizado para trabajar con dispositivos Apple y es una opción popular entre diseñadores que se centran en el desarrollo de aplicaciones para iOS y macOS.
* **Seguridad y privacidad**: **Comunidad activa:** Hay una comunidad activa de usuarios y recursos en línea, lo que facilita la búsqueda de tutoriales, plantillas y soluciones a problemas comunes.

Para cualquier acción de descarga, instalación, gestión o aprendizaje del uso de Sketch, puede emplear el enlace de su sitio oficial: <https://www.sketch.com/>

## Adobe XD

Adobe XD es la primera herramienta creada por la compañía Adobe, que fue específicamente diseñada para prototipar la experiencia de usuario (UX) e interface de usuario (UI). Esta herramienta ha sido el resultado de la empresa Adobe al trabajar con diseñadores de UX usan sus herramientas de diseño gráfico como Ilustrator, Photoshop y otros, de tal manera en un solo producto agrupa características que tiene como objetivo hacer que los diseñadores de interfaces sean más productivos.

Adobe XD es una herramienta de diseño de interfaces de usuario (UI) y experiencia de usuario (UX) altamente valorada por su conjunto de características y ventajas. Entre sus funciones destacadas se encuentran:

* **Diseño de Alta Fidelidad:** Adobe XD permite a los diseñadores crear diseños de alta fidelidad con facilidad, lo que les ayuda a visualizar cómo se verá y se sentirá la interfaz final.
* **Prototipado Interactivo**: los usuarios pueden crear prototipos interactivos para simular la experiencia del usuario, lo que facilita la evaluación y prueba de la usabilidad de la interfaz.
* **Diseño de Animaciones**: la herramienta permite diseñar animaciones y transiciones fluidas entre pantallas, mejorando así la interacción y la experiencia del usuario.
* **Colaboración en Tiempo Real**: Adobe XD ofrece capacidades de colaboración en tiempo real, lo que permite a varios miembros del equipo trabajar en el mismo proyecto simultáneamente.
* **Diseño Responsive**: facilita el diseño responsivo para garantizar que la interfaz se adapte adecuadamente a diferentes tamaños de pantalla y dispositivos.
* **Integración con Adobe Creative Cloud**: se integra de manera fluida con otras aplicaciones de Adobe Creative Cloud, como Photoshop e Illustrator, lo que simplifica la transferencia de diseños y activos entre aplicaciones.
* **Bibliotecas de Diseño Compartidas**: los usuarios pueden crear y compartir bibliotecas de diseño para mantener la coherencia en proyectos y equipos.
* **Comentarios y Revisión:** Adobe XD permite a los equipos y clientes agregar comentarios directamente en el diseño, facilitando así la comunicación y la revisión del trabajo.
* **Pruebas de Usuario Remotas**: esta herramienta facilita la realización de pruebas de usuario remotas, lo que resulta esencial para obtener retroalimentación valiosa durante el proceso de diseño.
* **Publicación y Compartir en la Nube**: los diseños y prototipos se pueden compartir en la nube para acceder a ellos desde cualquier ubicación y permitir que otros los revisen sin necesidad de descargar la aplicación.

Enlace web Para cualquier acción de descarga, instalación, gestión o aprendizaje del uso de Adobe XD, puede emplear el enlace de su sitio oficial: <https://helpx.adobe.com/co/xd/get-started.html>

## Conceptos de “Material Design”

“Material Design”es un estilo de diseño creado por Google. Fue anunciado el 25 de junio del 2014 en la conferencia Google I/O. Se ha implementado en el sistema operativo para móviles Android desde la versión Lollipop y también en otros servicios gratuitos de Google como Docs, Calendar, Drive, etc. (¿Qué es el Material Design?, 2021)

Google ha dispuesto al público una guía sobre el “Material Design”para que los diseñadores se animen a implementarlo en aplicaciones móviles y sitios web disponible en: <https://material.io/design>

Los objetivos de esta propuesta son:

* Emplear un lenguaje visual combinando lo principios de un buen diseño y las opciones que disponibles en las nuevas tecnologías.
* Ser un sistema que posibilite una experiencia uniforme en diferentes plataformas y dispositivos (como pantallas táctiles, control de voz, etc).

Google junto con los objetivos de “Material Design” presenta también sus tres principios fundamentales:

1. **Lo material como una metáfora.** Como lo material está unido a la realidad fusionándose con el espacio, tiempo y el movimiento, está ligado a la realidad táctil, es por esto que se puede integrar con la tecnología haciendo uso de la creatividad e imaginación.

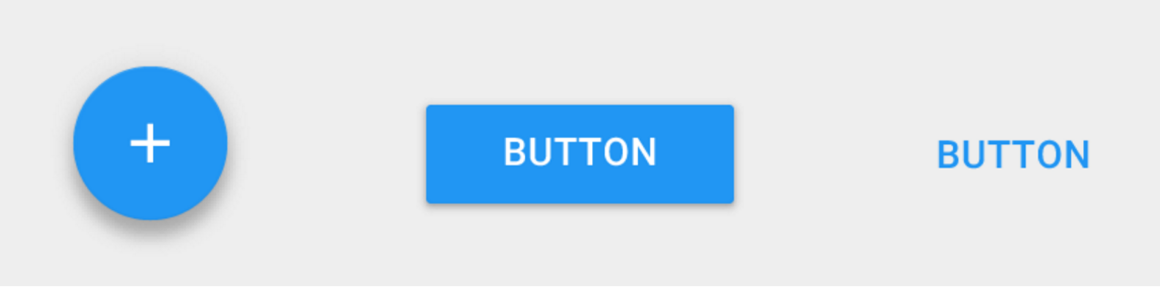
Los bordes y superficies de las cosas materiales entregan pistas visuales basadas en la realidad. El uso de atributos familiares al mundo real ayuda al usuario a entender las posibilidades de lo que se puede hacer con cada elemento presente en la interfaz. Para lograr esto se debe presentar los principios superficie, luz y movimiento, como elementos principales para transmitir cómo los objetos existen y están dispuestos a interactúan uno con el otro o con el usuario.

En conclusión, “Material Design”es una propuesta de diseño novedosa y que se está incorporando en las nuevas versiones de Android. Los principios del “Material Design”van guiados a mejorar la experiencia de usuario teniendo en cuenta las últimas tecnologías.

1. **Llamativo e intencional.**

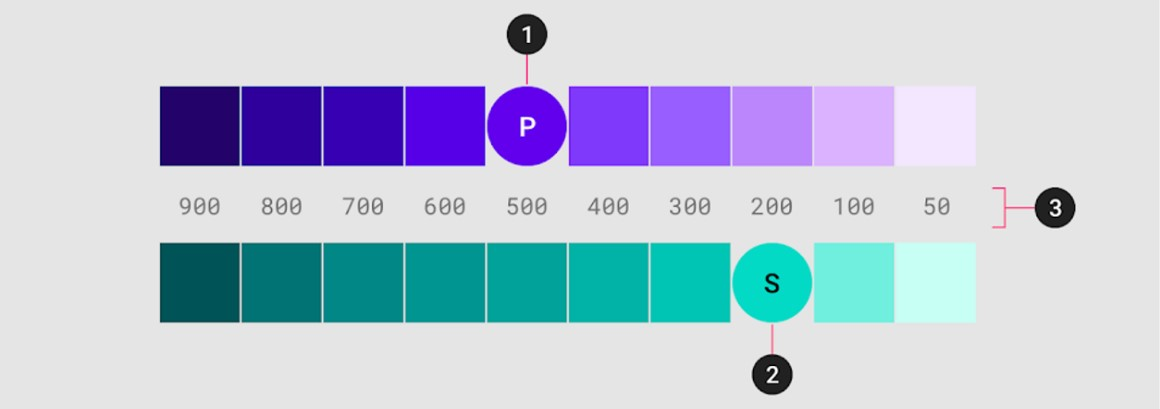
Todos los elementos visuales son diseñados por los conceptos de la teoría del diseño como el uso del color, los espacios en blanco, sobras, el sistema de tabla de datos, etc. Todos estos elementos ayudan a determinar la jerarquía y lectura visual.

Diseño de componentes



Nota. Tomada de diseño de componentes. Efecto Ripple de Material Design. (2015).

El sistema de color de “Material Design” ayuda a aplicar color a las interfaces de usuario de una manera simplificada. En este sistema, el diseñador define un color como primario y otro como secundario para representar el tema de su marca. Las variaciones de oscuras y claras para cada color se aplican a la interfaz de usuario de diferentes maneras de manera prestablecida:

1. Manejo de colores

Nota. Tomado de Manejo de colores. “Material Design” (2021).

1. **El movimiento otorga significado.**

En una aplicación web o móvil una animación tiende a captar la atención de los usuarios, sin embargo, es deseable que estas animaciones no interrumpan la experiencia de usuario. De esta forma es deseable que exista relación entre las animaciones y la propia esencia del sitio web o aplicación, por ejemplo, el efecto de presionar el botón la experiencia real de desplazar un elemento.

Para Google, el movimiento enfatiza el papel del usuario como el actor principal de la interacción con la máquina y el que dirige los movimientos en dicha interacción. Las acciones del usuario tienen el poder de cambiar el diseño o disposición de los elementos. Es así como los elementos de la interfaz son presentados al usuario sin interferir con la experiencia de usuario, incluso cuando dichos elementos se transforman y reorganizan.

En conclusión, “Material Design” es una propuesta de diseño novedosa y que se está incorporando en las nuevas versiones de Android. Los principios del “Material Design” van guiados a mejorar la experiencia de usuario teniendo en cuenta las últimas tecnologías.

# Principios básicos de usabilidad

La usabilidad es un término que recoge una valoración de la calidad de la experiencia de usuario cuando interactúa con un producto o sistema. Para medirla se hace a través del estudio de la relación que se produce entre las herramientas diseñadas y los usuarios que las usan, para determinar la eficiencia en el uso de los diferentes elementos ofrecidos en las pantallas y la efectividad en el cumplimiento de las tareas que se pueden llevar a cabo a través de ellas. En conclusión, se puede definir la usabilidad como la disciplina que estudia el diseño de aplicaciones móviles o sitios web para que los usuarios puedan interactuar de forma fácil, fluida e intuitivamente.

Existen varios aspectos relacionados con el uso de aplicaciones y la manera en que las personas se relacionan con esas aplicaciones que se les ofrecen. Básicamente estos estudios han proporcionado un conjunto de comprobación de estos aspectos, como mecanismo de variación. Dicha validación debe ser hecha a través de diferentes aspectos, entre los que se cuentan los siguientes:

* **Facilidad de aprendizaje.** Se puede emplear una unidad de tiempo para determinar lo que demoran los usuarios en usarla de manera correcta y realizar las operaciones básicas y comunes.
* **Facilidad y eficiencia de uso.** Se puede usar una unidad de tiempo para establecer cuanto tiempo se demora un usuario en realizar una operación completa de su quehacer con la herramienta de trabajo.
* **Facilidad de recordar el funcionamiento.** Capacidad de recordar las características y forma de uso de un sistema; se debe usar para volver a emplearlo en el futuro.
* **Frecuencia y gravedad de errores.** Mensajes de error a los usuarios para apoyarlos cuando deban enfrentar errores determinados en el sistema. Por ejemplo, cuando en un sistema de facturación se intenta hacer una factura sin relacionar, al menos, un producto o servicio a facturar.
* **Satisfacción subjetiva.** Indica lo satisfechos que quedan los usuarios después haber usado el sistema, debido al diseño presentado y es subjetiva porque depende de la percepción de cada usuario.

Los anteriores aspectos aplican, en principio, a cualquier tipo de aplicación web o móvil, sin embargo, el compendio de buenas prácticas para aplicaciones móviles se destaca por aspectos como los relacionados a continuación:

1. **Definir con simplicidad y claridad la estructura de los contenidos.** Generalmente, la recomendación es organizarlos de acuerdo con una jerarquía consistente temáticamente o según las relaciones y procesos de negocio de la aplicación.
2. **Establecer un diseño “limpio”.** Correcta tipografía, imágenes y recursos gráficos que hagan fácil al usuario encontrar lo que busca, separando los contenidos con una clasificación lógica.
3. **Ceder el control al usuario.** Los usuarios no sienten una experiencia positiva si en algún momento se sienten perdidos mientras interactúan con el sistema. Se debe garantizar que el usuario sepa dónde se encuentra y cómo acceder a otro enlace o funcionalidad.
4. **Facilitar la interacción.** Permitir que el usuario pueda desarrollar varias acciones de forma fácil e intuitiva aporta a la satisfacción, sin embargo, si estas opciones no se presentan como un diseño limpio, se puede lograr un efecto adverso.
5. **Simplificar y sintetizar.** La regla básica para aplicar este principio es: “todo lo que no es necesario sobra”. Con ello se evita sobrecargar el sistema con información u operaciones superfluas.
6. **Adaptar para todo tipo de dispositivos.** Otro elemento básico es la adaptabilidad a diferentes dispositivos.

# Componentes de “software”

Los sistemas de información son la agrupación de varios elementos que, operando en conjunto bajo unas premisas definidas, logran el objetivo de mantener en operación dichos sistemas. A continuación, se relacionan algunos de estos elementos necesarios para garantizar la continuidad del funcionamiento del sistema.

## Licenciamiento

Una licencia de “software” es un contrato que describe los derechos legales del uso autorizado de un sistema de información, programa u aplicativo. En otras palabras, es un acuerdo en el que el fabricante o proveedor del “software” le otorga a una persona u organización un permiso para utilizar un determinado producto.

Una acción de aceptación inequívoca de este contrato se efectúa cuando se descarga e instala a través del medio oficial de distribución del producto a instalar, es decir que, aunque no se firme un documento explícitamente cuando se realizan las operaciones descritas en el contrato (generalmente: descarga, instalación, configuración) que causan la aceptación de las condiciones de aceptación, es considerado legalmente como si se firmara un contrato de licencia.

#### Tipos de licencia de “software”

En Colombia, según la norma que regula este tema, se encuentra en la Ley 603 de 2000, llamada ley de licencias de “software” (o también Ley para cumplimiento de las licencias de “software”). Las licencias de “software” se clasifican según el uso final y las libertades que el autor ofrece de su producto al usuario final. Por lo tanto, es importante conocer el tipo de licenciamiento de “software”:

* **“Software” propietario o también llamado “software” de código cerrado**. Es aquel en el que el autor no transmite ninguno de los derechos de ejecución, copia, modificación, cesión o redistribución al comprador, sino que solo establece las condiciones bajo las cuales el usuario puede utilizar el programa.
* **Shareware.** Consiste en la distribución de una versión gratuita y limitada en sus funciones respecto de la aplicación que es de pago. Con esta licencia se le permite al usuario su uso por un tiempo y funciones delimitadas, pero cuando el tiempo finaliza, el usuario deberá pagar o desinstalar el programa.
* **Freeware.** Este tipo de licencia permite la instalación y uso gratuito de la aplicación y todos sus componentes sin que requiera alguna suscripción, hacer el registro es opcional pero no tiene costo o pago, sin embargo, el código fuente se mantiene limitado.
* **OEM (Original “Equipment Manufacturer”).** Esta licencia tiene la condición de que el programa debe instalarse en un equipo nuevo, por lo que generalmente aplica a los sistemas operativos.
* **Licencia corporativa por volumen.** El incumplimiento del uso de “software” no licenciado acarrea sanciones administrativas hasta de 200 salarios mínimos legales mensuales vigentes (200 SMMLV) para todos los administradores de la empresa, según lo estipule la Superintendencia de Sociedades.

## Seguridad

La seguridad de las aplicaciones, hace referencia a las medidas de seguridad, a nivel de aplicación (incluidas en la codificación), cuyo propósito es impedir el acceso no autorizado, el robo o el secuestro de datos o códigos dentro de la aplicación. Incluye las consideraciones de seguridad a tener en cuenta en el momento de desarrollar y diseñar aplicaciones, además de los sistemas y las técnicas para proteger las aplicaciones después de distribuirlas.

Para lograr el objetivo de hacer seguras las aplicaciones se deben incluir aspectos en el “hardware”, “software” y procedimiento que minimicen las vulnerabilidades de seguridad. La estrategia de publicación de la aplicación requiere el empleo de un conjunto de protocolos o procedimientos y para garantizarla se deben realizar pruebas al respecto.

En conclusión, la seguridad de las aplicaciones implica en el proceso de desarrollar, añadir y probar características de seguridad que deben tener las mismas para garantizar que respondan contra amenazas, tales como la modificación y el acceso no autorizados. Garantizar esto es importante porque las aplicaciones actuales generalmente están disponibles a través de varias redes y conectadas a la nube, esto incrementa las vulnerabilidades a las que se exponen.

### Tipos de seguridad de las aplicaciones

Existen tipos de características de seguridad en las aplicaciones, como por ejemplo la autenticación, la autorización, el cifrado de la información que consiste en ocultar el contenido a usuario no autorizado que puedan interceptar los datos, el registro y las pruebas de seguridad.

En el momento de desarrollar una aplicación es casi indispensable integrar procedimientos de autenticación y autorización para asegurarse de que solo accedan los usuarios autorizados. La forma básica de este proceso se consigue obligando a los usuarios a proporcionar un nombre de usuario y una clave para iniciar una sesión en el sistema. En el entorno de aplicaciones existen facilidades para agregar otros medios de identificación de usuario y algo que le caracteriza (una huella dactilar o el reconocimiento facial o reconocimiento de voz). Cuando además del usuario y contraseña se emplean otros de los medios de identificación de usuario se le denomina autenticación de múltiples factores.

Cuando el usuario esté autenticado, se recomienda que el sistema valide que el usuario tenga los permisos para acceder a la aplicación o componentes de esta. La autenticación se debe efectuar antes de validar la autorización. Cuando el usuario esté autenticado y durante el uso de la aplicación, se pueden proteger los datos con otras técnicas de seguridad con el fin de evitar que los ciberdelincuentes no los vean ni los usen. El cifrado de la información es la técnica que transforma la información y la transmite por las redes de tal manera que solo el emisor y el receptor puedan interpretar o descifrar el contenido real de los datos; esta técnica es la más empleada en sistemas de información que usan la internet como medio de transferencia de datos.

Finalmente, cuando se vulnera la seguridad del sistema, el registro puede ayudar a identificar quién ha accedido a los datos, cómo y cuándo. Para lo cual se deben emplear archivos de registro con toda la información relacionada.

En conclusión, los tipos básicos de seguridad de la aplicación son autenticación, cifrado y registro de incidencias. Pero según el contexto que involucre el funcionamiento de la aplicación se debe tener en cuenta:

* **Seguridad de las aplicaciones en la nube**. Las aplicaciones móviles y las aplicaciones web requieren servicios en la nube; ello plantea desafíos adicionales propios del entorno de la nube, ya que los recursos muchas veces, son compartidos. Existen dispositivos a nivel de infraestructura como los identificadores de intrusos o los cortafuegos que tienen la finalidad de identificar usos no autorizados de los recursos de red por los que funcionan las aplicaciones en la nube.
* **Seguridad de las aplicaciones móviles.** Los dispositivos móviles, generalmente también transmiten y reciben información de la red pública de Internet, de modo que son vulnerables a ataques. Existen técnicas como la que establece una Red Privada virtual (“Virtual Private Network”, VPN), que consiste en cifrar los protocolos de comunicación y todo el contenido a través de ellos simulado, creando con ello una red privada sobre una red pública.
* **Controles de seguridad de las aplicaciones**. Los controles de seguridad son las técnicas que mejoran la seguridad de una aplicación a nivel de codificación para que sea menos vulnerable a las amenazas. Las técnicas de desarrollo muchas veces propenden a establecer estos lineamientos. Por ejemplo, al trabajar con Android Studio la metodología de trabajo obliga a cumplir ciertos principios de codificación de las instrucciones que se le dan al móvil, de manera que sean más seguras.
* **Pruebas de seguridad de las aplicaciones**. Es indispensable realizar pruebas de seguridad de las aplicaciones durante el proceso de desarrollo de “software” para garantizar que no existan vulnerabilidades en cada versión antes de ser liberada al público.

## API’s, Firebase

Las organizaciones están cambiando el modelo de su arquitectura de “software” a arquitecturas basadas en componentes de microservicios. Un microservicio es un servicio pequeño encargado de una única función básica, y para realizar un proceso más complejo se emplearían varios microservicios coordinados entre sí. La ventaja de esto es que ese mismo servicio está publicado en internet no para usuarios finales sino para que sea utilizado por aplicaciones, de esta manera una aplicación móvil puede usar ese servicio de la misma forma que una aplicación web o un “chatbot”.

#### Definición de API’S

Se debe imaginar que una empresa necesita que sus clientes registren sus solicitudes por página web, aplicación móvil y “chatbot”; la principal ventaja de esto es que si se desea cambiar una regla de negocio (por ejemplo, aplicar determinados descuentos), no debe modificar los tres sistemas sino solamente los microservicios que están involucrados en dicho proceso.

Los microservicios se publican a través de lo que se denomina una API del acrónimo en inglés “Applicactions Programming Interface”, o Interfaz de programación de aplicaciones. Son un conjunto de subrutinas, funciones y/o procedimientos que se ofrecen para ser utilizados por otro “software” como una capa de abstracción (es decir de ocultación de los detalles de codificación). Por lo tanto, una API es en esencia la capacidad de comunicación entre componentes de “software”.

### Firebase

Es una plataforma en la nube de Google para el desarrollo de aplicaciones web y móviles. Fue creada en 2011 pasando a ser parte de Google en 2014, su función esencial es hacer más sencilla la creación de aplicaciones web y móviles en su desarrollo, como herramientas que facilitan el trabajo para que sea más rápido, pero sin renunciar a la calidad y la seguridad.

Para acceder a los servicios de Firebase se requiere consumir el API suministrado por Google.

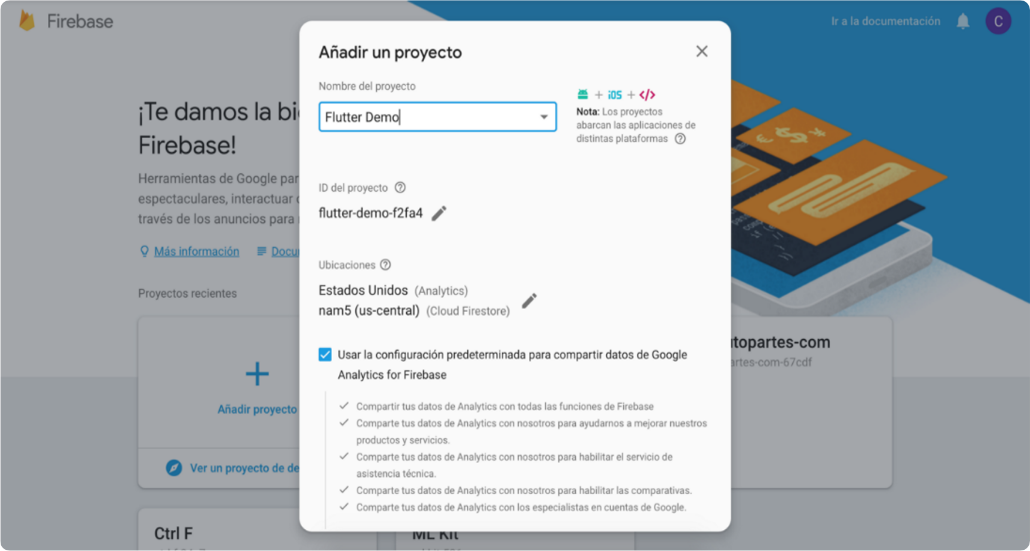
Las bases de datos en tiempo real son la principal y esencial herramienta de Firebase. Estas se alojan en la nube, son de tipo NoSQL en formato JSON. Este servicio ofrece alojar y disponer de los datos de información de la aplicación en tiempo real, manteniéndolos actualizados, aunque el usuario no realice ninguna acción.

Firebase envía automáticamente eventos, es decir, llama funcionalidades de las aplicaciones cuando los datos cambian; esto es lo que hace posible desarrollar aplicaciones con funciones que se ejecutan en tiempo real.

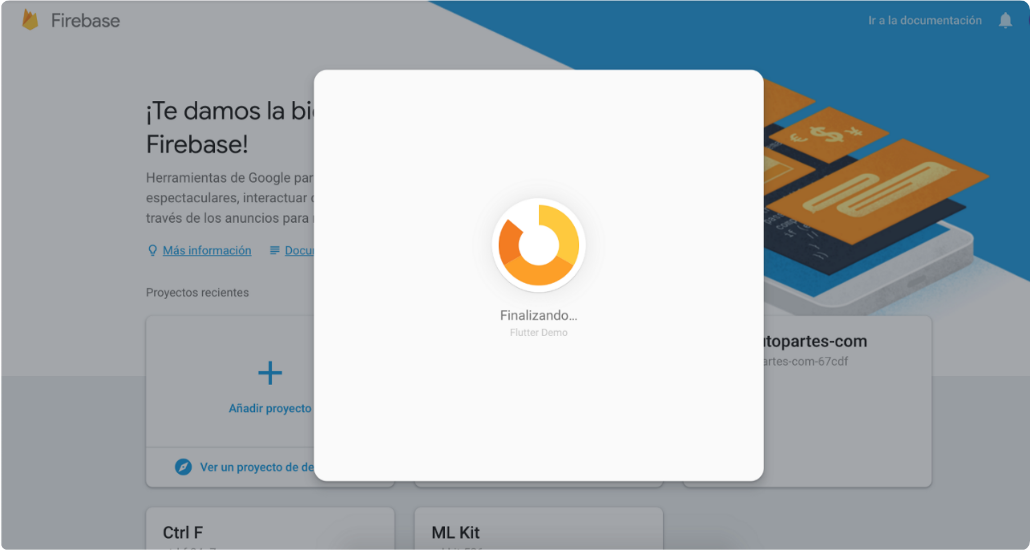
Frente al tema de seguridad ofrece un sistema de autenticación que permite tanto el “login” de usuario (mediante email y contraseña) como el acceso mediante al uso de perfiles de otros sistemas externos, generalmente de redes sociales.

A continuación, se mostrará un proceso básico para crear una base de datos en Firebase.

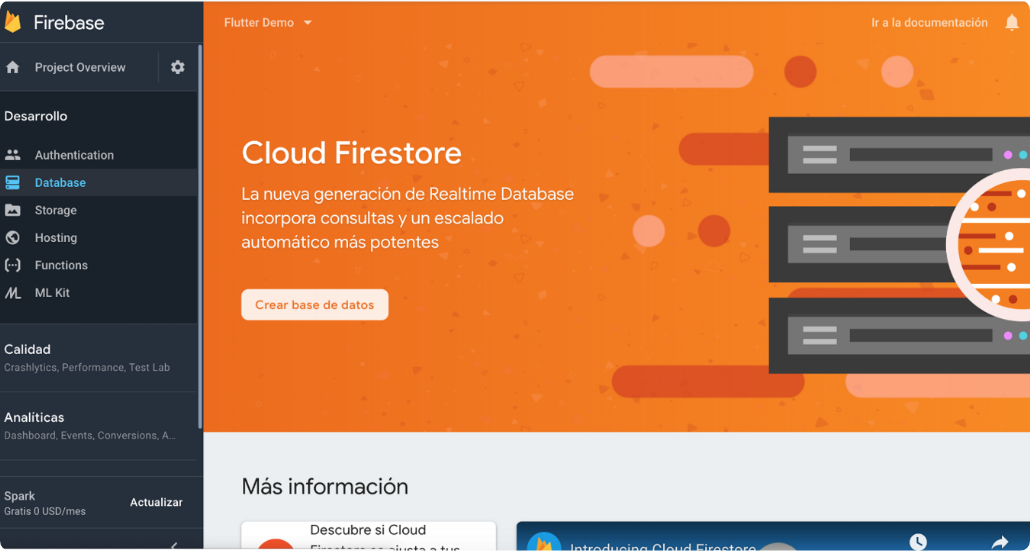
* **Nuevo proyecto Firebase.**



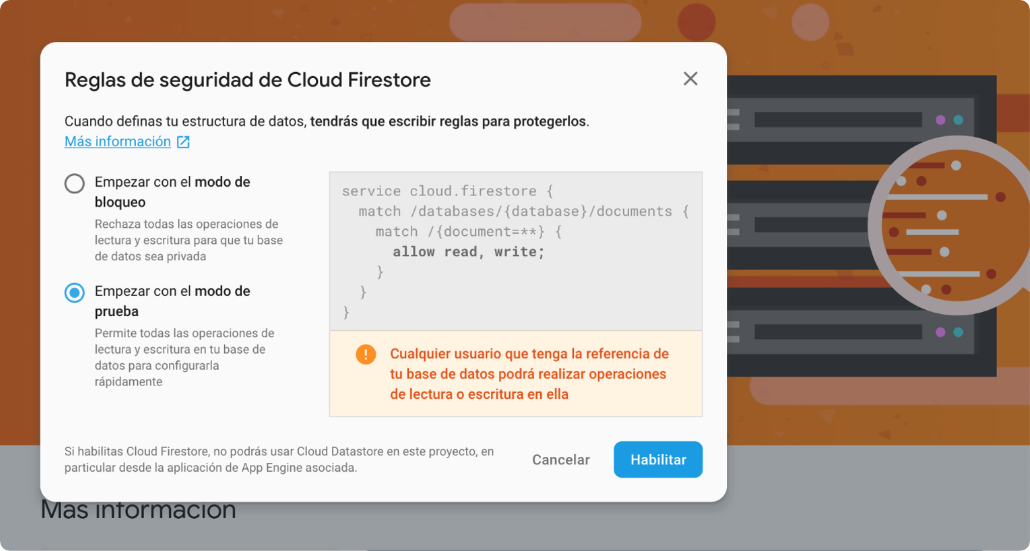
* **Esperar el proceso de setup.**



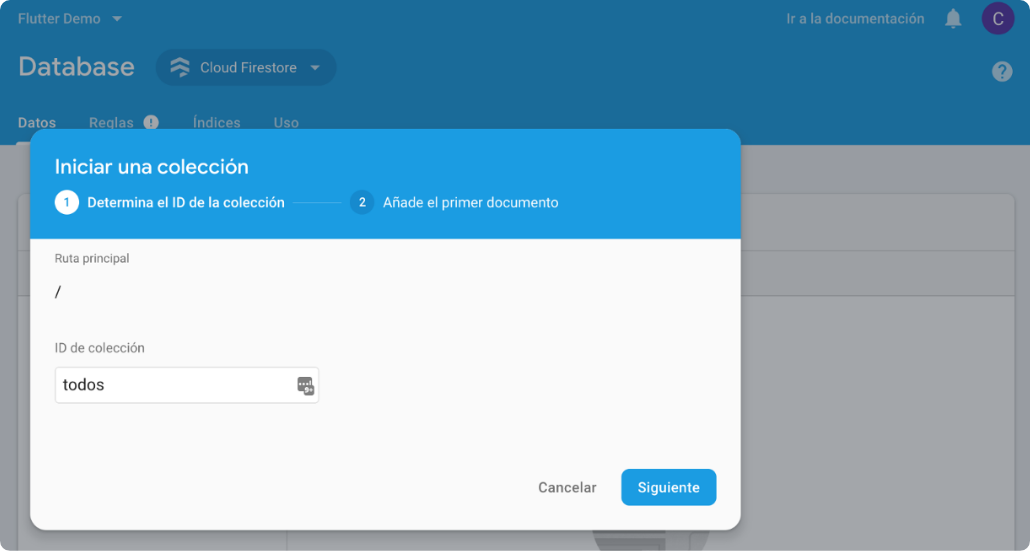
* **Panel del proyecto.**



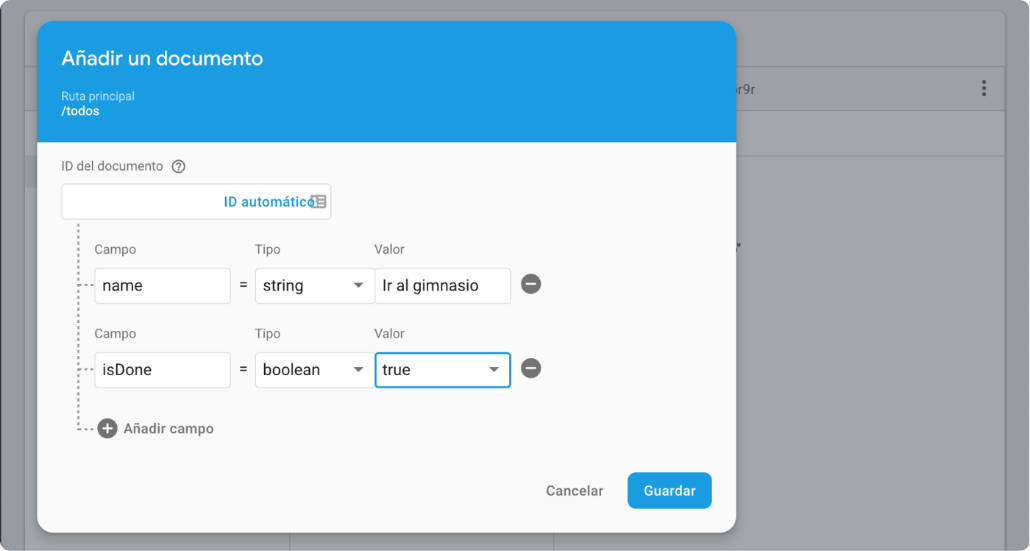
* **Modo de pruebas.**



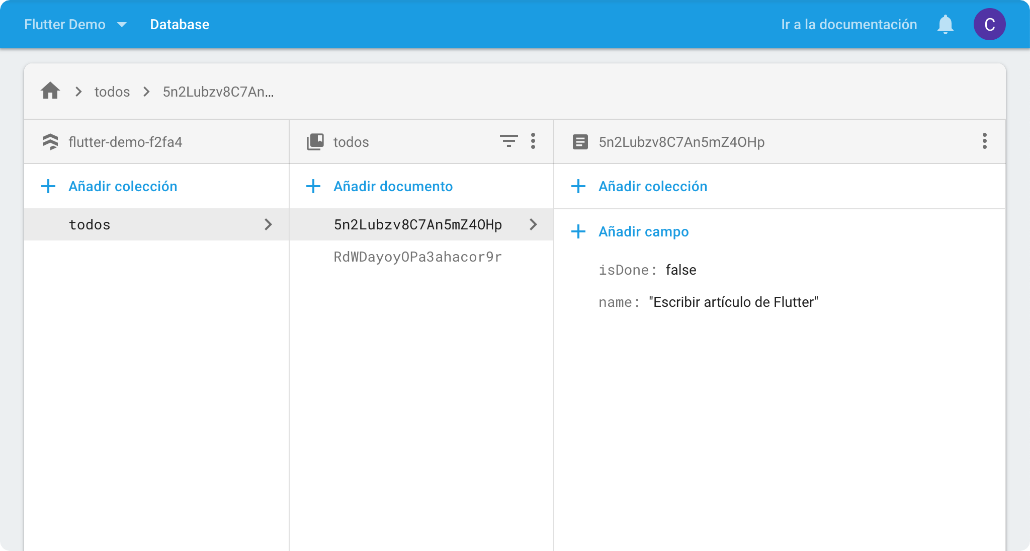
* **Crear una colección.**



* **Crear un documento.**



* **Colección y sus documentos**



Uno de los principales inconvenientes podría ser la necesidad de pago ya que Firebase no es gratuita en todas sus funcionalidades y gratuitamente limita el número de usuario. Sin embargo, para los proyectos que se encuentren en sus primeras etapas, la versión gratuita de Firebase es más que suficiente.

## Herramientas

Durante el desarrollo de un proyecto basado en microservicios, las pruebas de API se vuelven esenciales. Estas pruebas son cruciales tanto para el desarrollo como para el consumo de una API, ya que permiten evaluar y garantizar el correcto funcionamiento, la funcionalidad esperada, la confiabilidad y la seguridad de la aplicación.

A continuación, se enumeran algunas herramientas de código abierto que facilitan el proceso de prueba y diseño de APIs, utilizadas para distintos protocolos web.

1. **SOAP UI.** Puede comprobar los servicios web SOAP como los servicios web de tipo RESTful. Está disponible como una versión de código abierto y PRO. Está basado en Java, por tanto empleable en cualquier sistema operativo; sobre todo, es fácil de aprender y usar y confiable para todos.

Puede acceder a la herramienta en su sitio oficial mediante el siguiente enlace: <https://www.soapui.org>

Le recomendamos ingresar al video **Tutorial SOAPUI** donde explican el paso a paso del uso de la herramienta: <https://youtu.be/rfIi9hdT-vk>

1. **Estudio Katalon.** Automatiza procesos de pruebas de servicios web y APIs para dispositivos móviles. Reconocida como la mejor en el área de automatización de este tipo de pruebas**.**

Puede acceder a la herramienta en su sitio oficial mediante el siguiente enlace: <https://katalon.com/>

Le recomendamos ingresar al video **Katalon Studio: Creando nuestro primer proyecto,** donde explican el paso a paso del uso de la herramienta: <https://www.youtube.com/watch?v=Ho80rqLBb1w&feature=youtu.be>

1. **TestNG.** Inspirado en JUnit y NUnit para el lenguaje Java. El lema principal de esto es proporcionar funcionalidades fáciles de usar y cumplir con todo tipo de fases de prueba como unidad, integración, funcional, etc. (TestNG, 2021).

Puede acceder a la herramienta en su sitio oficial mediante el siguiente enlace: <https://testng.org/doc/>

Le recomendamos ingresar al video **Pruebas con Selenium WebDriver, TestNG y Java proyecto,** donde explican el paso a paso del uso de la herramienta: <https://www.youtube.com/watch?v=-VHNJcoMBek>

1. **JMeter.** Automatiza pruebas pero también puede realizar pruebas de rendimiento, pruebas de estrés de los servicios RESTFul y SIAP, con el uso de scripts de JMeter.

Puede acceder a la herramienta en su sitio oficial mediante el siguiente enlace: <https://jmeter.apache.org/>

Le recomendamos ingresar al video **JMeter desde Cero | Pruebas de Rendimiento - Parte 1,** donde explican el paso a paso del uso de la herramienta: <https://www.youtube.com/watch?v=E2zwM8s7thY&feature=youtu.be>

1. **PostMan.** Tal vez el más popular porque está enfocado en el proceso de desarrollo, lo que lo hace muy fácil de usar.

Puede acceder a la herramienta en su sitio oficial mediante el siguiente enlace: <https://www.postman.com/>

Le recomendamos ingresar al video **"Postman en Español: ¡El tutorial definitivo para el éxito en las pruebas API!,** donde explican el paso a paso del uso de la herramienta: <https://www.youtube.com/watch?v=qsejysrhJiU&feature=youtu.be>

Se puede concluir que existen muchas herramientas, para todas las fases de desarrollo e implementación de aplicaciones web y móviles con diversidad de licenciamientos y generalmente con funcionalidades de libre distribución que permiten dar marcha un proyecto sin incurrir en costos de entrada.

Síntesis

Aquí finaliza el estudio de los temas de este componente formativo. En este punto, analice el esquema que se muestra enseguida y realice su propia síntesis de lo estudiado. ¡Adelante!

Esquema general de las temáticas de este componente formativo. Tema principal: Diseño y construcción de Frontend. Temas integradores: análisis del requerimiento, diseño y prototipado, usabilidad, componentes de software.
El componente formativo abordó conceptos clave para el diseño y desarrollo de aplicaciones móviles nativas, sus elementos, vistas, maquetación, entornos de desarrollo, lenguajes entre otros.
Establecer correctamente los requisitos del sistema permite realizar un correcto diseño de la interacción con el usuario, lo cual permite dimensionar adecuadamente el coste en tiempo y permite potenciar la inversión.

Material complementario

|  |  |  |  |
| --- | --- | --- | --- |
| Tema | Referencia | Tipo de material | Enlace del recurso |
| 2. Diseño gráfico: herramientas de prototipado | FalconMasters. (2018). Curso de Adobe XD | 1. Introducción y que Aprenderemos en el Curso | Video Tutorial | <https://youtu.be/5HfqCSm9k3Y> |
| 2. Diseño gráfico: herramientas de prototipado | EDteam. (2017). Minicurso de Balsamiq Mockups (1) - Los *wireframes* y la interfaz. | Video Tutorial | <https://youtu.be/4WqfxF1QjTM> |
| 2. Diseño gráfico: herramientas de prototipado | Aprende, G. (2020). COMO USAR CANVA: Canva tutorial - Aprende a diseñar FÁCIL y de manera PROFESIONAL. | Video Tutorial | <https://youtu.be/kdE6ZmT_5MI> |
| 2. Diseño gráfico: herramientas de prototipado | Conde, J. (2016). 01.- Material Design con Android Studio. ¿Qué es Material Design? | Video Tutorial | <https://youtu.be/wt0Jzc9UHNw> |
| 4. Componentes de ”software” | Novato, P. (2019). Firebase 01.- Qué es Firebase?. | Video Tutorial | <https://youtu.be/0Bhrwc8VfGw> |

Glosario

**“Wearable”:** dispositivo conectado que se puede llevar puesto y que se conecta al teléfono móvil.

**“Wireframing”:** es un esquema de página o plano de pantalla, como guía visual que representa el esqueleto o estructura visual de una interfaz.

Referencias bibliográficas

Aula formativa. (2019, julio 29). ¿Qué es el Material Design?. [Web log post]. blog.aulaformativa.com. <https://blog.aulaformativa.com/que-es-el-material-design>

Beust, C. (2019). TestNG. <https://testng.org/doc>

Material.io. (2021). Material Design. <https://material.io/design/color/the-color-system.html#color-usage-and-palettes>

Saenz, S. (2018). Ripple estilo Material Design.https://sergiosanz.dev/blog/material-design-ripple/

Créditos

| Nombre | Cargo | Regional y Centro de Formación |
| --- | --- | --- |
| Claudia Patricia Aristizabal | Responsable del Ecosistema | Dirección General |
| Rafael Neftalí Lizcano Reyes | Responsable de Línea de Producción | Centro Industrial del Diseño y la Manufactura - Regional Santander |
| Henry Eduardo Bastidas Paruma | Experto temático | Centro de teleinformática y producción industrial - Regional Cauca |
| Jonathan Guerrero Astaiza | Experto temático | Centro de teleinformática y producción industrial - Regional Cauca |
| Peter Emerson Pinchao Solis | Experto temático | Centro de teleinformática y producción industrial - Regional Cauca |
| Zulema Yidney León Escobar | Experta temática | Centro de teleinformática y producción industrial - Regional Cauca |
| Oscar Absalón Guevara | Diseñador instruccional | Centro de gestión industrial - Regional Bogotá |
| Alix Cecilia Chinchilla Rueda | Diseñadora instruccional | Centro de gestión industrial - Regional Bogotá |
| José Gabriel Ortiz Abella | Evaluador instruccional | Centro para la industria y la comunicación gráfica - Regional Distrito Capital |
| Julia Isabel Roberto | Evaluadora instruccional | Centro para la industria y la comunicación gráfica - Regional Distrito Capital |
| Ana Catalina Córdoba Sus | Metodólogo para formación virtual | Centro Industrial del Diseño y la Manufactura - Regional Santander |
| Fabián Leonardo Correa Díaz | Diseñador Instruccional | Centro Industrial del Diseño y la Manufactura - Regional Santander |
| Carmen Alicia Martínez Torres | Animador y Productor Multimedia | Centro Industrial del Diseño y la Manufactura - Regional Santander |
| Camilo Andrés Bolaño Rey | Locución | Centro Industrial del Diseño y la Manufactura - Regional Santander |
| Yerson Fabian Zarate Saavedra | Diseñador de Contenidos Digitales | Centro Industrial del Diseño y la Manufactura - Regional Santander |
| Francisco José Lizcano Reyes | Desarrollador Full-stack | Centro Industrial del Diseño y la Manufactura - Regional Santander |
| Daniel Ricardo Mutis Gómez | Evaluador para Contenidos Inclusivos y Accesibles | Centro Industrial del Diseño y la Manufactura - Regional Santander |
| Zuleidy María Ruíz Torres | Validador de Recursos Educativos Digitales | Centro Industrial del Diseño y la Manufactura - Regional Santander |
| Luis Gabriel Urueta Álvarez | Validador de Recursos Educativos Digitales | Centro Industrial del Diseño y la Manufactura - Regional Santander |