

Diseño y construcción de interfaces web

**Breve descripción:**

Este componente formativo ofrece herramientas para la construcción de interfaces gráficas, especialmente en entornos web. Se presentarán conceptos y una guía básica para desarrollar un proyecto desde cero utilizando una estrategia de caso de estudio.

**Julio 2024**

Tabla de contenido

[Introducción 4](#_Toc172646085)

[1. Sistema de navegación en plataformas de software 6](#_Toc172646086)

[2. Sistema de interacción en plataformas de software 9](#_Toc172646087)

[Tendencias actuales en diseño web 10](#_Toc172646088)

[3. Estructura de la interfaz de la aplicación 11](#_Toc172646089)

[Caso de estudio sobre el servicio de publicación de posts 11](#_Toc172646090)

[Instalación de Balsamiq 11](#_Toc172646091)

[Diseño de la estructura de la interfaz 12](#_Toc172646092)

[Creación de las vistas del sistema 13](#_Toc172646093)

[4. Construcción de la estructura del front-end 15](#_Toc172646094)

[5. Análisis de requerimientos 25](#_Toc172646095)

[6. Diseño gráfico: herramientas de prototipado 30](#_Toc172646096)

[Adobe XD 30](#_Toc172646097)

[Canva 32](#_Toc172646098)

[Conceptos de Material Design 33](#_Toc172646099)

[7. Principios básicos de usabilidad 36](#_Toc172646100)

[8. Componentes de software 39](#_Toc172646101)

[Licenciamiento 39](#_Toc172646102)

[Tipos de licencia de software 39](#_Toc172646103)

[8.1. Seguridad 41](#_Toc172646104)

[8.2. API Firebase 44](#_Toc172646105)

[8.3. Herramientas 45](#_Toc172646106)

[Síntesis 48](#_Toc172646107)

[Material complementario 49](#_Toc172646108)

[Glosario 51](#_Toc172646109)

[Referencias bibliográficas 52](#_Toc172646110)

[Créditos 53](#_Toc172646111)

Introducción

Este componente formativo abordará los conceptos y técnicas que respaldan el desarrollo de la interfaz gráfica de un producto de software. Se iniciará con una introducción teórica sobre el sistema de navegación e interacción de las aplicaciones. A partir de esta base, se revisará el proceso de construcción de la interfaz gráfica mediante un caso práctico, comenzando con un boceto y luego avanzando a la maquetación con herramientas de prototipado rápido y tecnologías front-end.

Además, se destacará el uso del framework CSS Materialize, el cual proporciona una interfaz dinámica, usable y de fácil aprendizaje. Se brindarán herramientas y recursos para facilitar el desarrollo de futuros proyectos de software.

Abordar un objetivo sin una planificación adecuada puede resultar en un análisis menos efectivo, generando así una discrepancia entre el producto entregado y las expectativas del cliente o los usuarios finales. Esto es particularmente crítico en el desarrollo de aplicaciones móviles, donde la interacción con el usuario ofrece desafíos únicos. Una planificación inadecuada puede resultar en frustración para el usuario o el cliente, y en el mejor de los casos, en un exceso de esfuerzo o incumplimiento de los plazos establecidos.

Establecer los requisitos del sistema de manera precisa es esencial para diseñar adecuadamente la interacción con el usuario, optimizando así el tiempo y maximizando la inversión. Por ello, se recomienda realizar dos actividades fundamentales antes de empezar a codificar:

Determinar con precisión los requerimientos (el alcance) del sistema.

Diseñar o prototipar el sistema de manera que se proporcione una visión clara de cómo interactuará con el usuario.

# Sistema de navegación en plataformas de software

Un sistema de navegación permite que los usuarios se desplacen libremente por las distintas secciones de una plataforma. No se trata simplemente de un aspecto menor de la navegación.

Considérese dos situaciones hipotéticas:

* Un aficionado a la lectura entra en una biblioteca buscando una edición antigua del Quijote.
* Un joven empresario visita una web de viajes en busca de un vuelo de ida y vuelta a Perú.

Estos escenarios presentan cómo diferentes personas acceden a la información según sus necesidades específicas. Por lo tanto, antes de diseñar cualquier sistema de navegación, ya sea web o móvil, es crucial entender cómo buscan información los usuarios.

A continuación, se explorará cómo se realiza la búsqueda de información y la navegación en sitios web y dispositivos móviles:

1. **En busca de la información**

Saber cómo buscan las personas información online presenta problemas y desafíos. La disponibilidad inmediata de los recursos en la red permite a las personas cambiar el rumbo de su búsqueda sobre la marcha. Existe una estrategia de búsqueda, en la cual a medida que se localiza nueva información, el curso de la investigación de la persona puede variar. Se suele buscar con agilidad, y es frecuente que los usuarios puedan vagar de una página a otra con el peligro de perderse en la profundidad de una jerarquía web. La manera de buscar y la frecuencia con la que el usuario puede perderse en este proceso son aspectos que deben ser tenidos en cuenta a la hora de diseñar los sistemas de navegación web (Calvo, 2015).

1. **Navegar por sitios web**

La información no solo se busca, se experimenta. El término «experiencia» implica que las emociones juegan un papel clave junto con la conducta y la cognición. Las personas ya no pueden ser vistas como seres racionales que se comportan de forma sistemática y ordenada a la hora de buscar información, teniendo en cuenta su papel crítico en la web. Al respecto muchos de los sitios tienen en la barra de navegación etiquetas o secciones con los siguientes títulos (Inicio, servicios, nosotros, acerca de y contactos), esta distribución del contenido que hace parte de la arquitectura de la información de un sistema interactivo software debe evaluarse, inicialmente.

1. **Patrón tradicional para el diseño de navegación de un sitio web**

Se debe tener siempre presente que el diseño de la navegación influye en gran medida en la experiencia de los usuarios al interactuar con el sitio. Por un lado, los diseñadores deben dedicar el tiempo necesario para familiarizarse con los patrones de búsqueda de sus usuarios, sus emociones, reacciones, entre otros. Por otro lado, este conocimiento facilita a los diseñadores la selección del tipo de navegación más adecuada para el sitio, permitiéndoles elegir entre diversas opciones disponibles.

1. **Navegar por aplicaciones móviles**

La navegación en aplicaciones móviles debe ser intuitiva y fácil de recordar, tanto para usuarios nuevos como para quienes ya la han usado previamente, asegurando que puedan moverse con facilidad entre las distintas secciones o vistas. Sin embargo, diseñar una navegación que sea fácil de descubrir y usar, representa un desafío debido a las limitaciones de las pantallas pequeñas y la necesidad de priorizar el contenido sobre los componentes de la interfaz. Aunque existen diversos patrones de navegación diseñados para abordar estos problemas, cada uno presenta desafíos de usabilidad en ciertos aspectos, según Ferraris (2018).

Es necesario implementar al menos una técnica de prototipado rápido para facilitar la navegación entre las vistas del software y mejorar la interacción. Existe una correlación evidente entre la navegación a través de las diversas vistas de un sitio o aplicación y las acciones que debe realizar el usuario, conocido comúnmente como sistemas de interacción.

# Sistema de interacción en plataformas de software

Este sistema se define como un conjunto de componentes que responden a las necesidades de los usuarios al navegar por una plataforma de software. En otras palabras, comprende el diseño de cada acción que un usuario puede realizar en un sitio web o aplicación, tales como hacer clic, scrolling o utilizar gestos (como ampliar o disminuir el tamaño). Cada acción desencadena una respuesta específica en la aplicación desarrollada, a lo que se le denomina sistema de interacción.

Adicionalmente, en las páginas de interacción, se permite a los usuarios escribir, dibujar, editar, borrar y ejecutar todo tipo de acciones. Estas pueden ser muy simples, como la página de inicio de Google, o muy complejas, integrando múltiples elementos de interacción.

* Las páginas simples son aquellas en las que la sencillez es fundamental.
* Las páginas más complejas suelen requerir un periodo de aprendizaje más extenso para que los usuarios se familiaricen con ellas y puedan aprovecharlas al máximo.

En la actualidad, muchos de los sitios más visitados presentan características adaptadas a los cambios tecnológicos recientes en la movilidad de los dispositivos. Según el artículo de la Digital Guide de IONOS (2021):

Las páginas web deben ser hoy en día más eficientes e intuitivas para poder subsistir en un mercado online extremadamente competitivo, ya sea desde un PC de escritorio, una tablet o un teléfono móvil. El diseño recargado de la década de los años 2000 ha pasado a la historia: ahora la prioridad es concentrarse en lo esencial.

### Tendencias actuales en diseño web

Esta tendencia se ha consolidado en la realidad actual, donde se observa que la gran mayoría de las consultas y actividades se realizan a través de Internet.

Ejemplos comunes incluyen pedir un domicilio, realizar pagos o transacciones bancarias, leer un libro, consultar el correo electrónico o asistir a reuniones virtuales. De manera natural, los diseños e interacciones están siendo orientados hacia el enfoque móvil, influenciados por la experiencia y cultura de los usuarios.

En este contexto, es crucial que las páginas web sean fáciles de usar, ya que la usabilidad y accesibilidad se convierten en elementos decisivos dentro del marco del diseño responsivo.

El diseño web moderno permite que una página o sitio web "responda o reaccione" adecuadamente al dispositivo utilizado por la persona, asegurando una presentación correcta del contenido. Es importante entender que no se trata solo de adaptar el contenido, sino de modificar la interfaz en general para mejorar la usabilidad y, por ende, la experiencia del usuario.

Con esta perspectiva, y abordando de manera sucinta estos conceptos, se propone como caso de estudio la estructura y maquetación de la interfaz de un servicio de publicación de posts. Para ello, el primer recurso que se utilizará será una herramienta de prototipado rápido llamada Balsamiq Mockups, la cual se detallará en el siguiente apartado.

# Estructura de la interfaz de la aplicación

La estructura de la interfaz de la aplicación está diseñada para ofrecer una experiencia de usuario óptima, combinando funcionalidad con estética visual. Cada elemento está pensado para garantizar una navegación intuitiva y accesible, desde los wireframes iniciales hasta los mockups finales que definen el aspecto y la operatividad del sistema.

### Caso de estudio sobre el servicio de publicación de posts

Cuando se inicia un nuevo proyecto de software, es habitual realizar sketches, bocetos o dibujos rápidos para obtener una visión preliminar de cómo será la interfaz gráfica del sistema. Este proceso es muy beneficioso para los equipos de desarrolladores y diseñadores, ya que les permite plasmar una visión concreta de lo que será el proyecto. Todo comienza con un boceto en papel. Durante esta etapa, se hacen cambios y correcciones, se descartan ideas y se generan otras mejores. Estos sketches se transforman gradualmente en el prototipo final del proyecto, finalmente evolucionando en Mockups. A este proceso se le conoce como Wireframing, que consiste en crear la estructura de una interfaz gráfica.

### Instalación de Balsamiq

Una de las herramientas más efectivas para crear prototipos, bocetos o wireframes es Balsamiq Mockups. Para comenzar a utilizar Balsamiq, se optará por el modo de servicio en la nube (versión de prueba), disponible en: <https://balsamiq.cloud/>, la instalación es muy sencilla: basta con registrarse y hacer clic en "nuevo proyecto", como se presenta a continuación:

* **Registro**

Ingrese a Balsamiq Cloud: para registrarse, ingrese su correo electrónico corporativo y seleccione 'Start Trial', accediendo así a una prueba gratuita de 30 días.

* **Creación del primer proyecto**

Para iniciar un nuevo proyecto en Balsamiq Cloud, haga clic en el ícono '+' situado en la esquina superior izquierda de su espacio de trabajo.

* **Renombrar proyecto**

Para cambiar el nombre de su proyecto en Balsamiq Cloud, escriba el nuevo nombre en el cuadro de texto y haga clic en "Rename Project".

### Diseño de la estructura de la interfaz

Esta herramienta permite generar de manera estructural las vistas de la aplicación. Es decir, se describe en palabras cuántas vistas son necesarias y cuántos objetos de interacción se requieren, proporcionando una respuesta de la estructura de la interfaz de la aplicación. Esta actividad se suele realizar tradicionalmente con lápiz y papel, sin embargo, con Balsamiq, esta tarea se puede llevar a cabo sin dificultad, como se presenta a continuación:

1. Diseño de la estructura de la interfaz



### Creación de las vistas del sistema

Para este componente formativo, el caso de estudio es un servicio básico de publicación de posts, que consta de dos vistas en la aplicación. Se ha diseñado la aplicación web de forma que sea responsive, tomando como plataforma un dispositivo móvil (tableta). La barra de herramientas de Balsamiq proporciona todo lo necesario para esta primera tarea. En la sección de material de apoyo, se adjunta un video que destaca las ventajas de esta herramienta. A continuación, se presentan las vistas de la aplicación:

1. Creación de las vistas del sistema



Para este caso, se crean dos vistas. La primera es un formulario que consta de tres (3) objetos de interacción. El primero es una caja de texto donde se introduce el título, el segundo es un área de texto para describir el contenido del post, y por último, un botón que envía estos datos a una segunda vista. Desde Balsamiq, se puede generar esta interacción para facilitar la navegación entre vistas, revisando las opciones del objeto 'button'.

Por otro lado, la segunda vista muestra de manera ordenada la lista de posts publicados por el usuario, con el formato detallado en la ilustración adjunta. Es importante resaltar que la fecha debe ser tomada automáticamente por el sistema al momento de la publicación.

# Construcción de la estructura del front-end

Para el desarrollo de la estructura del front-end, que en el ámbito de este componente formativo se limita exclusivamente a las tecnologías de HTML5 y CSS3, se utilizará el framework de maquetación llamado Materialize. Este framework CSS permite crear sitios y aplicaciones web siguiendo los principios de Material Design. La versión estándar del framework incluye CSS y JavaScript, con los estilos CSS listos para usar.

Además, Materialize incorpora buenas prácticas para el desarrollo del front-end, lo que contribuye a que la aplicación sea más eficiente y rápida en su despliegue. Por otra parte, es un framework responsive, lo que significa que se adapta a cualquier tamaño de dispositivo.

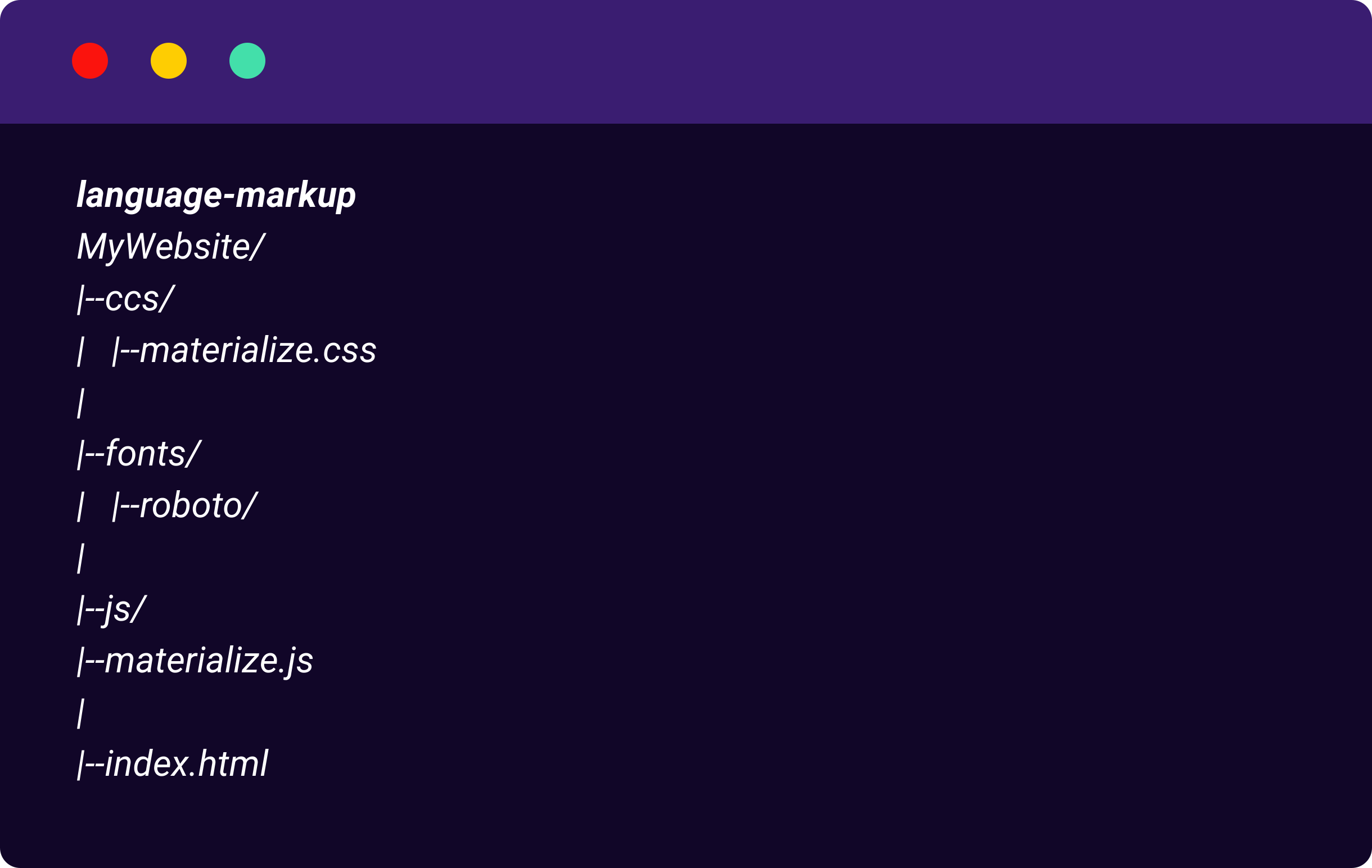
A continuación, se detallan los pasos para la instalación y manejo del framework (Materialize, 2021):

1. **Instalación de Materialize:**

Existen varias formas de instalar el framework. En este caso, se utilizará la descarga de los archivos, disponible en el enlace proporcionado en la sección de material de apoyo, con el nombre "Descarga de archivos Materialize".

Una vez descargada la carpeta, se encontrará una estructura de directorios muy similar a la guía que presenta Materialize en su página web.

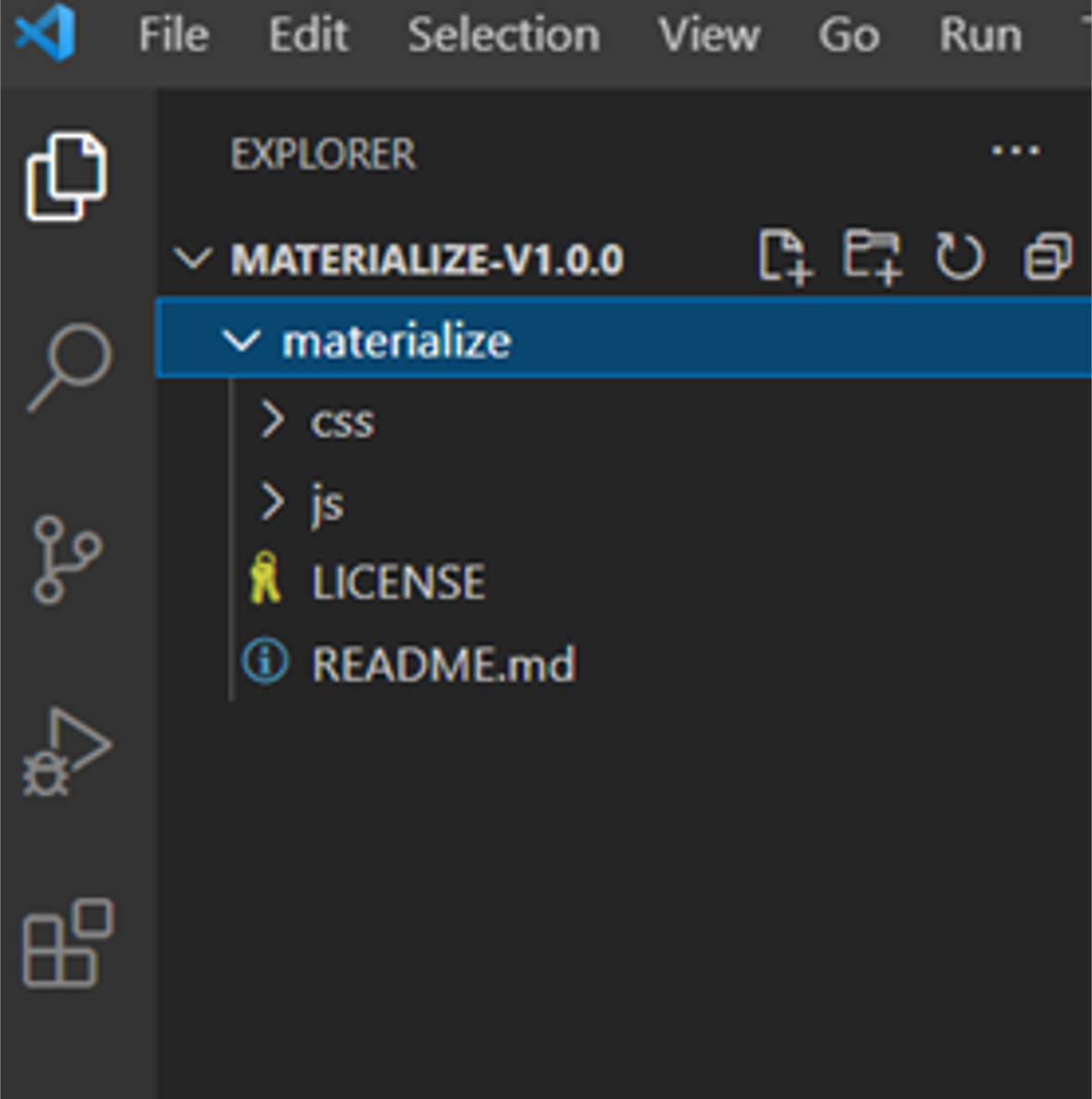
1. Estructura de directorios de Materialize



1. **Proyecto de Materialize en Visual Studio Code**

Ahora, lo siguiente que debe hacerse es incorporar o referenciar esa carpeta desde un entorno de desarrollo integrado. Para este ejemplo, se utilizará Visual Studio Code, como se presenta a continuación:

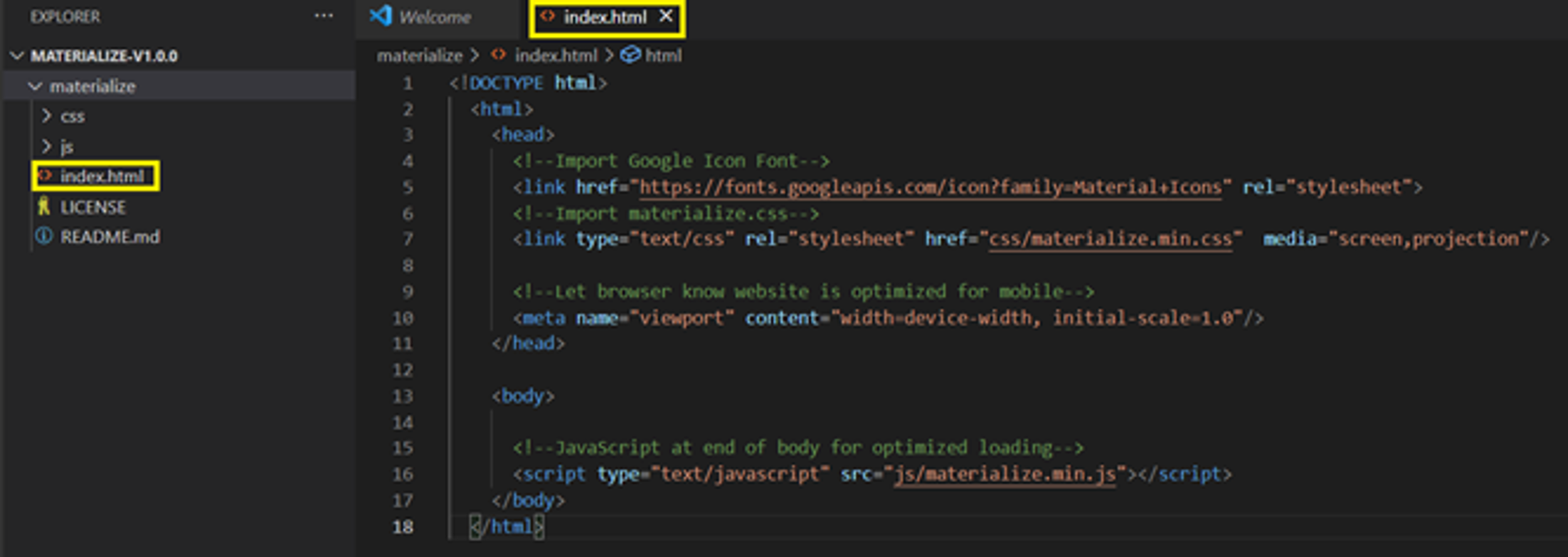
1. Proyecto de Materialize en Visual Studio Code



1. **Configuración del proyecto**

Siguiendo la guía proporcionada por el framework, que puede consultarse en el enlace indicado en la sección de material de apoyo con el nombre "Guía rápida de Materialize", se crea un archivo llamado index.html en la raíz de la estructura de directorios. Este archivo contendrá la siguiente información, incluyendo las referencias a los archivos de estilos y JavaScript que proporcionan el funcionamiento y la apariencia a la interfaz desarrollada:

1. Archivo index.html de la guía de Materialize



1. **Desarrollo del proyecto (primera vista)**

Hasta el momento, tanto el framework como el entorno para crear las vistas del caso de estudio están preparados en este componente formativo. Por lo tanto, es necesario referirse nuevamente a la primera vista diseñada para revisar los componentes necesarios para el desarrollo.

En la primera vista, se requiere lo siguiente:

* Título.
* Formulario que consta de dos cajas de texto y un botón.

<div class="section no-pad-bot" id="index-banner">

<div class="container">

<br><br>

<h1 class="header center orange-text">Publica tu Post</h1>

<div class="row center">

<h5 class="header col s12 light">Un ejemplo de maquetación

utilizando el framework de Materialize</h5>

</div>

<br><br>

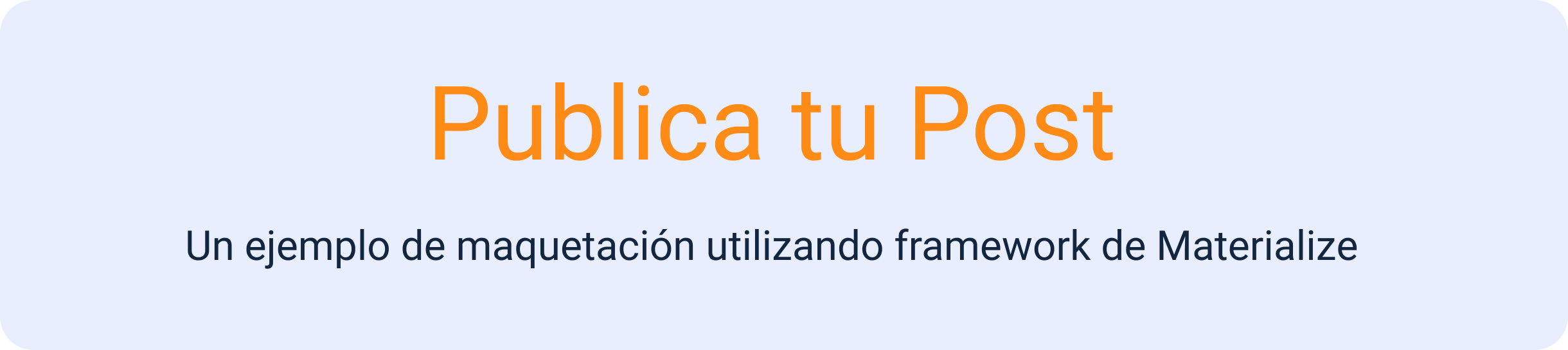
</div>

</div>

Para esta maquetación, es necesario dimensionar el grid o rejilla, lo cual ayuda a mantener el contenido de manera organizada y estructurada. Para lograrlo, se utiliza la clase "container", que centra el contenido de la página y ocupa aproximadamente el 70% del ancho de la ventana, lo que ayuda a centrar y organizar el contenido de la página.

Además, en el código se muestra el uso de colores. Por ejemplo, el título de la página utiliza el atributo "orange-text", que proporciona el color naranja al texto "Publica tu Post", como se presenta a continuación:

1. Publica tu post



Ahora continúa el desarrollo del formulario, el cual consta de dos (2) inputs: uno destinado al título del post y otro al área de texto, tal como se presenta a continuación:

<div class="row">

<div class="container">

<form class="col s12">

<div class="row">

<div class="input-field col offset-s4 s4">

<input placeholder="Título del Post" id="tituloPost"

type="text" class="validate">

<label for="tituloPost">Título del Post</label>

</div>

</div>

<div class="row">

<div class="input-field col offset-s4 s4">

<textarea placeholder="Título del Post" id="textarea1"

class="materialize-textarea"></textarea>

<label for="textarea1">Contenido del Post</label>

</div>

</div>

<button class="btn waves-effect waves-light col offset-s4

s4" type="submit" name="action">Enviar

<i class="material-icons right">send</i>

</button>

</form>

</div>

</div>

1. **Resultado de la primera vista**

Como se puede observar en el código, se sigue utilizando la estructura tradicional de un formulario en HTML. Lo que varía son los atributos de las clases que se utilizan para centrar los inputs o mejorar la apariencia de cada objeto de interacción. Por ejemplo, el atributo "col offset-s4 s4" se emplea en cada una de las etiquetas, lo que permite que cada componente del formulario se encuentre centrado.

1. Resultado de la primera vista



1. **Desarrollo de la segunda vista**

Para esta parte se utiliza el concepto de listas en HTML, como se presenta en el siguiente código de HTML:

<div class="container">

<ul class="collection">

<li class="collection-item avatar">

<i class="material-icons circle">face</i>

<span class="title">Título del Post</span>

<p>Este es un contenido para el primer Post de prueba

<br>

<i>Fecha de publicación: 1 junio de 2021</i>

</p>

<a href="#" class="secondary-content"><i class="material-

icons">grade</i></a>

</li>

</ul>

</div>

1. **Resultado de la segunda vista**

En el código HTML anterior, se muestra la estructura estándar de una lista desordenada, con los únicos cambios relacionados con el uso de los atributos propios del framework. Por lo tanto, el resultado esperado se presenta en la siguiente figura:

1. Resultado de la segunda vista



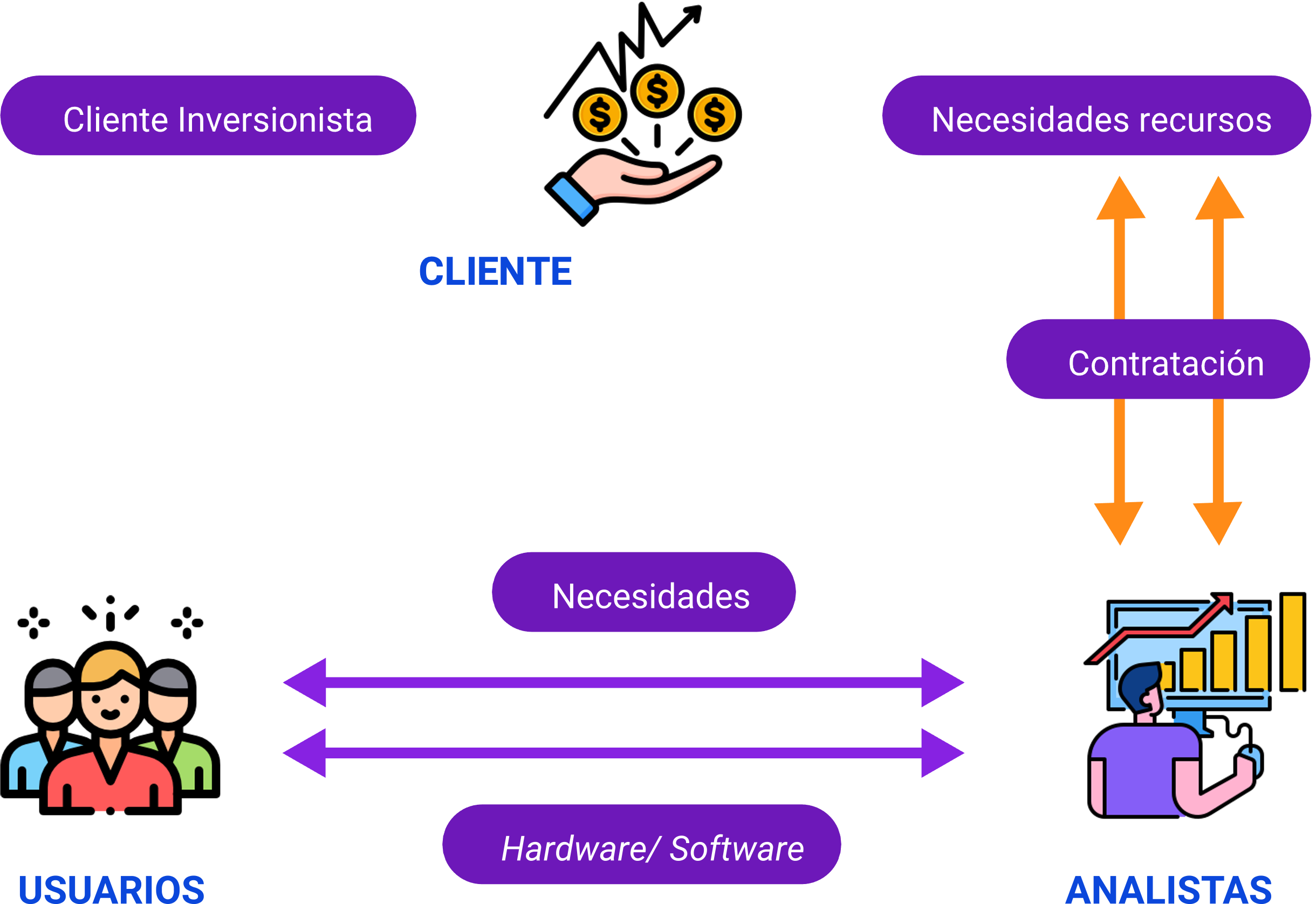
# Análisis de requerimientos

El desarrollo de aplicaciones móviles va más allá de la mera codificación o programación; también implica analizar y comprender las necesidades del cliente, crear modelos que faciliten el proceso de desarrollo, planificar el proyecto y otras actividades relacionadas.

La aplicación de técnicas para elicitar requisitos permite identificar las fuentes de estos con los interesados en el sistema (stakeholders). Sin embargo, estas actividades pueden llevar a que el cliente o los usuarios excedan lo que realmente necesitan o que no consideren las limitaciones a nivel de hardware, software, presupuesto, tiempo o requisitos. Por lo tanto, es fundamental conocer con detalle lo que se pretende alcanzar y las necesidades reales a cubrir. Así, se deben detallar y modificar los requisitos básicos elicitados, lo cual se logra mediante el consenso de las partes involucradas.

Se deben responder las siguientes preguntas: ¿Qué es lo prioritario?, ¿qué es esencial?, y ¿para cuándo se requiere? Esto se especifica de alguna manera y luego se revisa o válida para garantizar que la comprensión del problema por parte del analista coincida con la de los demás participantes.

1. Esquema de acuerdos



Cuando los sistemas no son tan complejos en su enlace, como suele ser el caso para aplicaciones móviles, se recomienda el empleo de historias de usuario. Sin embargo, se sugiere revisar algunos aspectos que no son fáciles de identificar en un proceso de licitación de requisitos y que se mencionan a continuación, ya que están enmarcados en el contexto de aplicaciones móviles.

1. **Aspecto físico o de hardware**

* ¿En qué dispositivos debe funcionar la aplicación? Por ejemplo: móvil, tablet, smartwatch, etc.
* ¿Qué tipo de aplicación es la solución óptima: nativa, web o híbrida?
* En el caso de las aplicaciones nativas, ¿qué sistema operativo se utilizará? Android, iOS, Windows Phone, etc.
* En el caso de las aplicaciones web, ¿qué navegadores y versiones deben ser compatibles con las funcionalidades?
* ¿Cuántos usuarios simultáneos debe soportar el sistema?
* ¿Existen restricciones en el nivel de acceso a internet (ancho de banda o consumo de datos) o requerimientos de seguridad específicos?

1. **Interfaz de usuario**

* ¿La información de entrada es proporcionada por los usuarios o existen otras fuentes? En caso afirmativo, ¿cuáles?
* ¿La salida o consumo de información implica la interacción con uno o más sistemas?
* ¿Existe algún formato preestablecido que deben seguir los datos?
* ¿Hay alguna aplicación de referencia que pueda servir como guía para el proyecto?

1. **Aspectos humanos**

* ¿Qué perfil de población utilizará la aplicación?
* ¿Hay diferentes tipos de usuarios con roles distintos?
* ¿Qué tipo de capacitación o entrenamiento necesitará cada perfil de usuario?

Cuando un requisito o un conjunto de ellos forma parte de todo un ecosistema tecnológico en el que intervienen distintos tipos de aplicaciones (web, de escritorio, móviles, interacciones con otros sistemas de proveedores, clientes o socios), como ocurre, por ejemplo, con los sistemas bancarios que disponen de aplicaciones móviles, aplicaciones web, terminales de escritorio y portales para asociados, las historias de usuario no siempre ofrecen una visión completa de la complejidad del problema.

Estos sistemas generalmente cuentan con lo que se denomina "Reglas de Negocio" o un "Conjunto de Reglas de Negocio" que describen las políticas, operaciones, normativas, definiciones y restricciones que rigen en una organización. Dichas reglas son fundamentales para los sistemas de información o aplicaciones, ya que consideran los factores críticos para alcanzar los objetivos misionales del cliente. En algunos casos, incluso se requiere de un glosario de términos específicos del negocio para poder especificar dichas reglas de forma adecuada.

* **Por ejemplo**

Una regla de negocio podría ser: un cliente al que se factura más de 10,000 al año se clasifica como cliente de tipo A; a los clientes de tipo A se les aplica un descuento del 10 % en pedidos que superen los 3,000.

Las reglas de negocio pueden llegar a ser tan complejas como las operaciones del cliente. En situaciones de esta índole, se sugiere el uso de artefactos como los casos de uso para definir con detalle la especificación del sistema. Para guiar el proceso, se proponen algunos artefactos a elaborar:

* **Documento de especificación de requisitos**

Este documento 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 la estimación de costos y tiempos de desarrollo.

* **Plantilla de casos de uso**

Algunos de los requerimientos descritos en el documento de especificación de requisitos pueden necesitar un nivel de detalle técnico superior, especialmente al definir procesos síncronos o asíncronos de operación del sistema informático o casos particulares donde intervienen reglas de negocio.

Como se puede identificar, los requerimientos de una aplicación móvil suelen estar cubiertos por técnicas como la elaboración de historias de usuario. No obstante, cuando la aplicación móvil forma parte de un ecosistema tecnológico más complejo, el proceso requiere de algún artefacto que facilite la identificación y validación de los requisitos.

# Diseño gráfico: herramientas de prototipado

Un diseñador front-end necesita contar con herramientas esenciales para realizar su labor, y el primer paso suele ser el desarrollo de una propuesta de interfaz gráfica o un prototipo de interfaz de usuario (UI) que permita al cliente visualizar cómo se verá el sistema una vez finalizado. Hoy en día, no basta con diseñar únicamente la versión para ordenadores de escritorio o el prototipo para móviles. Es necesario crear tres versiones distintas para escritorio, tablet y móvil, asegurando que el producto digital mantenga o adapte ciertas propiedades visuales de acuerdo al dispositivo en el que se utiliza. A continuación, se presentan algunas herramientas recomendadas para dicho propósito.

### 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 interfaz 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 Illustrator, Photoshop y otros, de tal manera que un solo producto agrupa características que tiene como objetivo hacer que los diseñadores de interfaces sean más productivos.

* **Instalación**

Para realizar la instalación de Adobe XD, debe ingresar a la página principal de Adobe.

* **Autenticación de cuenta**

En seguida el instalador le pedirá su cuenta de Google para poder realizar la instalación con su cuenta de usuario. Una vez realizado este proceso se activa el botón iniciar instalación.

* **Inicio de la aplicación**

En la pantalla de bienvenida, siga los pasos para empezar a crear un proyecto en Adobe XD.

* **Seleccionar prototipo**

Cuando se abre, Adobe XD despliega una pantalla que permite seleccionar un tipo de proyecto. Se debe hacer clic sobre la opción correspondiente para avanzar con la creación del proyecto.

* **Cambiar nombre al área de trabajo**

Una vez creada el área de trabajo, haga doble clic en el nombre del área, y cámbielo a "Inicio".

* **Crear fondo del sistema**

En el área de trabajo, haciendo uso del panel de propiedades ubicado en el lado derecho de la aplicación, se abre el diálogo de selección de cuadro de colores.

* **Configurar encabezado**

En el explorador de archivos, navegue hasta la imagen del encabezado y haga clic en Importar. Puede emplear los controles alrededor de la imagen para cambiar su tamaño, ajustándolo al de la pantalla.

* **Agregar rectángulo (área de contenido)**

Haciendo uso de la herramienta ubicada en el panel de propiedades de la izquierda, se creará un rectángulo debajo del encabezado.

* **Configurar fondo de área de contenido**

De la misma forma que se agrega una imagen, se incluye un cuadro de texto que se ajusta al fondo del área de contenido.

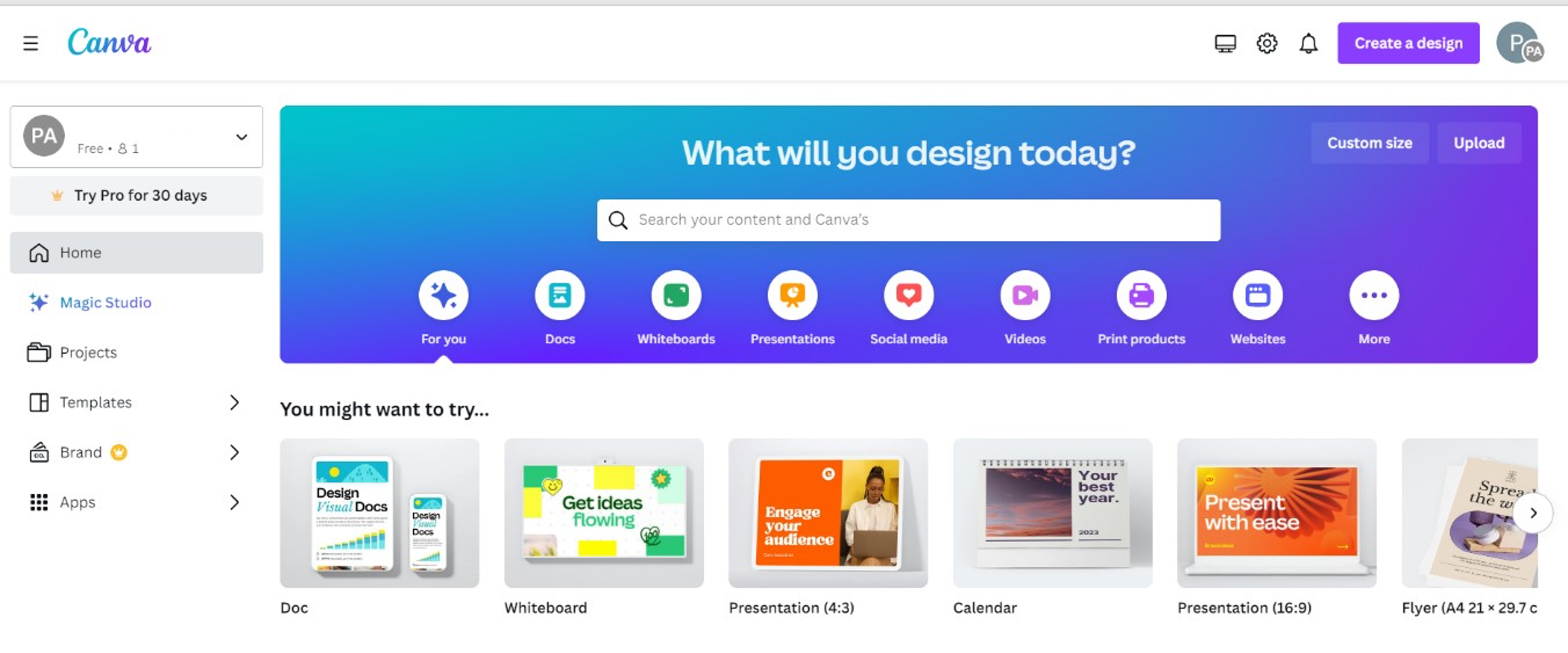
* **Avanzar con vista de prototipo**

La pestaña prototipo, ubicada en la parte superior de la aplicación, ofrece una pantalla de vista previa que permite ver y navegar por las diferentes páginas.

### Canva

Canva es una plataforma de diseño gráfico simplificada, establecida en 2012. Ofrece una interfaz intuitiva de arrastrar y soltar, y brinda acceso a una amplia biblioteca con más de 60 millones de fotografías y 5 millones de vectores, gráficos y fuentes. Para utilizar este servicio, es necesario registrarse en el sitio web. Al iniciar sesión, el usuario puede comenzar a crear diseños, como presentaciones para dispositivos móviles, utilizando el botón de crear diseño.

1. Crear presentación móvil CANVA



Canva, al igual que Adobe XD, permite diseñar interfaces para dispositivos móviles y ofrece un proceso similar. La plataforma sugiere comenzar con plantillas prediseñadas, lo que facilita el inicio de un proyecto.

Finalmente, cabe destacar que estas no son las únicas herramientas disponibles en el mercado. Se han mencionado las anteriores debido a su popularidad, aceptación en el mercado, y la amplia disponibilidad de documentación y videotutoriales en diversas plataformas en línea.

### Conceptos de Material Design

Material Design es un estilo de diseño desarrollado por Google, anunciado el 25 de junio de 2014 durante la conferencia Google I/O. Este enfoque de diseño se ha implementado en el sistema operativo Android a partir de la versión Lollipop, así como en otros servicios gratuitos de Google, incluyendo Docs, Calendar y Drive, entre otros.

* **Material Design**

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.

<https://m3.material.io/>

Los objetivos de esta propuesta son:

1. Emplear un lenguaje visual que combine los principios de un buen diseño con las opciones disponibles en las nuevas tecnologías.
2. Ser un sistema que facilite una experiencia uniforme en diversas plataformas y dispositivos, como pantallas táctiles, control de voz, entre otros.

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

* **Lo material como una metáfora**

Como lo material está unido a la realidad, fusionándose con el espacio, el tiempo y el movimiento, está ligado a la realidad táctil. Esto permite su integración con la tecnología mediante el uso de la creatividad e imaginación. Los bordes y superficies de las cosas materiales proporcionan pistas visuales basadas en la realidad, facilitando al usuario la comprensión de las posibilidades de cada elemento en la interfaz. Para lograr esto, se presentan los principios de superficie, luz y movimiento como elementos clave para transmitir cómo los objetos existen, están dispuestos e interactúan.

* **Llamativo e intencional**

Todos los elementos visuales son diseñados según los conceptos de la teoría del diseño, como el uso del color, los espacios en blanco, sombras, y el sistema de tabla de datos.

El sistema de color de Material Design facilita la aplicación de color a las interfaces de usuario de manera simplificada, permitiendo que el diseñador defina un color como primario y otro como secundario para representar el tema de su marca. Las variaciones oscuras y claras se aplican de formas preestablecidas.

* **El movimiento otorga significado**

En aplicaciones web o móviles, las animaciones captan la atención de los usuarios, pero deben evitar interrumpir la experiencia del usuario. Es deseable que las animaciones estén relacionadas con la esencia del sitio o aplicación, como el efecto de presionar un botón o la experiencia de desplazar un elemento. Google considera que el movimiento subraya el papel del usuario como actor principal en la interacción con la máquina, permitiendo que las acciones del usuario modifiquen el diseño o la disposición de los elementos sin interferir con la experiencia de 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.

# Principios básicos de usabilidad

La usabilidad es un término que evalúa la calidad de la experiencia de usuario al interactuar con un producto o sistema. Se mide a través del análisis de la relación entre las herramientas diseñadas y los usuarios que las emplean, con el objetivo de determinar la eficiencia y efectividad en el uso de los elementos ofrecidos en las interfaces y en el cumplimiento de las tareas que se realizan mediante ellas. Por tanto, la usabilidad puede definirse como la disciplina que estudia el diseño de aplicaciones móviles o sitios web para facilitar una interacción fácil, fluida e intuitiva por parte de los usuarios.

Existen varios aspectos relacionados con el uso de aplicaciones y cómo las personas interactúan con ellas. Estos estudios han derivado en un conjunto de criterios para evaluar dichos aspectos, funcionando como un mecanismo de validación. Esta validación se realiza considerando diversos factores, entre los que se incluyen 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 cuánto tiempo se demora un usuario en realizar una operación completa de su quehacer con la herramienta de trabajo.

* **Facilidad de recordar el funcionamiento**

La capacidad de recordar las características y la forma de uso de un sistema se debe usar para volver a emplearlo en el futuro.

* **Frecuencia y gravedad de errores**

Los mensajes de error se proporcionan a los usuarios para apoyarlos cuando deban enfrentar errores que se cometan en el sistema.

Ejemplo de un error predeterminado. Suponga que 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 de 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 y consistencia temática o según las relaciones y procesos de negocio de la aplicación.

1. **Establecer un diseño "limpio"**

Utilizar una tipografía correcta, imágenes y recursos gráficos que faciliten al usuario encontrar lo que busca, separando los contenidos con una clasificación lógica.

1. **Ceder el control al usuario**

Los usuarios no experimentan una sensación positiva si se sienten perdidos en algún momento mientras interactúan con el sistema. Se debe garantizar que el usuario sepa dónde se encuentra y cómo acceder a otro enlace o funcionalidad.

1. **Facilitar la interacción**

Permitir que el usuario pueda desarrollar varias acciones de forma fácil e intuitiva contribuye a la satisfacción; sin embargo, si estas opciones no se presentan con un diseño limpio, se puede lograr un efecto adverso.

1. **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.

1. **Adaptar para todo tipo de dispositivos**

Otro elemento básico es la adaptabilidad a diferentes dispositivos.

# Componentes de software

Los sistemas de información se componen de diversos elementos que, operando conjuntamente bajo premisas definidas, logran el objetivo de mantener en operación dichos sistemas. A continuación, se detallan algunos de estos elementos esenciales para garantizar la continuidad del funcionamiento del sistema.

### Licenciamiento

Una licencia de software es un contrato que establece los derechos legales del uso autorizado de un sistema de información, programa o aplicación. En otras palabras, es un acuerdo mediante el cual el fabricante o proveedor del software concede a una persona u organización el permiso para usar un producto específico.

Una aceptación inequívoca de este contrato se realiza cuando se descarga e instala el software a través del medio oficial de distribución del producto. Es decir, aunque no se firme un documento de manera explícita, las acciones descritas en el contrato (generalmente: descarga, instalación, configuración) que implican la aceptación de las condiciones, son consideradas legalmente como equivalentes a la firma de un contrato de licencia.

### Tipos de licencia de software

En Colombia, la normativa que regula las licencias de software está contenida en la Ley 603 de 2000, conocida como la ley de licencias de software o Ley para el cumplimiento de las licencias de software. Las licencias se clasifican según el uso final y las libertades que el autor concede sobre su producto al usuario final. Por lo tanto, es crucial conocer los tipos de licenciamiento:

* **Software propietario**

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**

La licencia que las empresas suelen adquirir establece un número específico de dispositivos, en los cuales el programa puede ser instalado y utilizado dentro de la misma organización.

El incumplimiento del uso de software no licenciado puede acarrear sanciones administrativas de hasta 200 salarios mínimos legales mensuales vigentes (200 SMMLV) para todos los administradores de la empresa, según lo estipulado por la Superintendencia de Sociedades.

## Seguridad

La seguridad de las aplicaciones se refiere a las medidas implementadas a nivel de aplicación, incluida la codificación, cuyo objetivo es prevenir el acceso no autorizado, así como el robo o secuestro de datos o códigos dentro de la aplicación. Esto abarca las consideraciones de seguridad que se deben tener en cuenta durante el desarrollo y diseño de aplicaciones, además de los sistemas y técnicas empleados para proteger las aplicaciones una vez distribuidas.

Para asegurar las aplicaciones, es fundamental incluir medidas en el hardware, software y procedimientos que minimicen las vulnerabilidades de seguridad. La estrategia de publicación de servicios de información requiere el uso de un conjunto de protocolos o procedimientos, y para garantizar la seguridad, es esencial realizar pruebas pertinentes.

La seguridad de las aplicaciones implica desarrollar, incorporar y probar características de seguridad destinadas a proteger las aplicaciones de amenazas, como la modificación y el acceso no autorizados. Es crucial asegurar esto, ya que las aplicaciones modernas suelen estar disponibles a través de múltiples redes y conectadas a la nube, lo que incrementa las vulnerabilidades a las que están expuestas.

* **Autenticación**

Procedimientos casi indispensables en el desarrollo de una aplicación para asegurar que solo accedan los usuarios autorizados. Esto comúnmente se logra mediante un nombre de usuario y contraseña, pero también puede incluir métodos adicionales como huellas dactilares, reconocimiento facial o de voz, conocido como autenticación de múltiples factores.

* **Autorización**

Una vez autenticado el usuario, es crucial validar que tenga los permisos necesarios para acceder a la aplicación o sus componentes. La autorización se verifica después de la autenticación.

* **Cifrado**

Técnica que transforma la información para su transmisión de tal manera que solo el emisor y el receptor puedan descifrar el contenido real de los datos, siendo una práctica común en sistemas que utilizan Internet para la transferencia de datos.

* **Registro de incidencias**

En caso de vulneraciones de seguridad, el registro puede ayudar a identificar quién ha accedido a los datos, cómo y cuándo. Se deben utilizar archivos de registro con toda la información relacionada para facilitar esta tarea.

En conclusión, los tipos básicos de seguridad de la aplicación son la autenticación, el cifrado y el registro de incidencias. Sin embargo, según el contexto del funcionamiento de la aplicación, se deben considerar otros aspectos relevantes.

* **Seguridad de las aplicaciones en la nube**

Las aplicaciones móviles y las aplicaciones web que requieren servicios en la nube enfrentan desafíos adicionales propios del entorno, como el hecho de que los recursos muchas veces son compartidos. Existen dispositivos a nivel de infraestructura, como los identificadores de intrusos y los cortafuegos, cuya finalidad es identificar usos no autorizados de los recursos de red utilizados por las aplicaciones en la nube.

* **Seguridad de las aplicaciones móviles**

Los dispositivos móviles generalmente transmiten y reciben información de la red pública de internet, haciéndolos vulnerables a ataques. Existen técnicas como el establecimiento de una Red Privada Virtual (VPN, Virtual Private Network), que consiste en cifrar los protocolos de comunicación y todo el contenido que pasa a través de ellos, creando así una red privada sobre una red pública.

* **Controles de seguridad de las aplicaciones**

Los controles de seguridad son técnicas que mejoran la seguridad de una aplicación a nivel de codificación para hacerla menos vulnerable a las amenazas. Estas técnicas de desarrollo a menudo establecen lineamientos específicos. Por ejemplo, al trabajar con Android Studio, la metodología de trabajo obliga a cumplir con ciertos principios de codificación para mejorar la seguridad de las aplicaciones móviles frente a intrusiones.

* **Pruebas de seguridad de las aplicaciones**

Es indispensable realizar pruebas de seguridad en 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 Firebase

Las organizaciones están adoptando modelos de arquitectura de software basados en microservicios. Un microservicio es un servicio pequeño destinado a una función específica, y para realizar procesos más complejos, se coordinan múltiples microservicios. Esto permite que el mismo servicio, aunque publicado en internet, no sea para usuarios finales sino para su uso por aplicaciones. De esta manera, una aplicación móvil puede utilizar ese servicio de la misma forma que una aplicación web o un chatbot.

* **Definición de API**

Imaginemos que una empresa necesita que sus clientes registren solicitudes a través de una página web, una aplicación móvil y un chatbot. La principal ventaja de los microservicios es que, si se desea cambiar una regla de negocio, como aplicar ciertos descuentos, no es necesario modificar los tres sistemas, sino solo los microservicios involucrados. Estos se publican mediante una API (Interfaz de Programación de Aplicaciones), que es un conjunto de subrutinas, funciones y procedimientos que se ofrecen para ser utilizados por otro software, actuando como una capa de abstracción que oculta los detalles de codificación.

* **Firebase**

Es una plataforma en la nube de Google para el desarrollo de aplicaciones web y móviles, creada en 2011 y adquirida por Google en 2014. Su principal función es facilitar la creación de aplicaciones web y móviles, proporcionando herramientas que agilizan el desarrollo sin sacrificar calidad ni seguridad. Para utilizar los servicios de Firebase, es necesario consumir la API proporcionada por Google.

La herramienta principal de Firebase son las bases de datos en tiempo real, alojadas en la nube y de tipo NoSQL en formato JSON. Este servicio permite alojar y gestionar datos de la aplicación en tiempo real, manteniéndolos actualizados incluso si el usuario no realiza ninguna acción. Firebase también envía automáticamente eventos que activan funcionalidades de las aplicaciones cuando los datos cambian, lo que facilita el desarrollo de aplicaciones con funcionalidades en tiempo real.

En cuanto a seguridad, Firebase ofrece un sistema de autenticación que permite tanto el inicio de sesión mediante correo electrónico y contraseña, como el acceso utilizando perfiles de otros sistemas externos, generalmente redes sociales.

#### Creación de una base de datos en Firebase

El proceso básico para crear una base de datos en Firebase, se presenta en el PDF "**Creación de una base de datos en Firebase**", lo invitamos a consultarlo en la carpeta Anexos.

Uno de los principales inconvenientes de Firebase podría ser la necesidad de realizar pagos, ya que no todas sus funcionalidades son gratuitas y la versión sin costo limita el número de usuarios. Sin embargo, para proyectos en sus primeras etapas, la versión gratuita de Firebase suele ser más que suficiente.

## Herramientas

Durante el desarrollo de proyectos que emplean microservicios, las pruebas de API son esenciales para garantizar la ejecución, funcionalidad, confiabilidad y seguridad de la aplicación, tanto en la fase de desarrollo como en la de consumo de la API. A continuación, se presentan varias herramientas de libre distribución que facilitan el proceso de prueba y diseño de APIs publicadas a través de los diferentes protocolos web.

* **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. Basado en Java, es empleable en cualquier sistema operativo, fácil de aprender y usar, y confiable para todos.

* **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.

* **TestNG**

Inspirado en JUnit y NUnit para el lenguaje Java. Su lema principal es proporcionar funcionalidades fáciles de usar y cumplir con todo tipo de fases de prueba como unidad, integración, funcional, entre otros.

* **JMeter**

Automatiza pruebas, pero también puede realizar pruebas de rendimiento, pruebas de estrés de los servicios RESTful y SOAP, con el uso de scripts de JMeter.

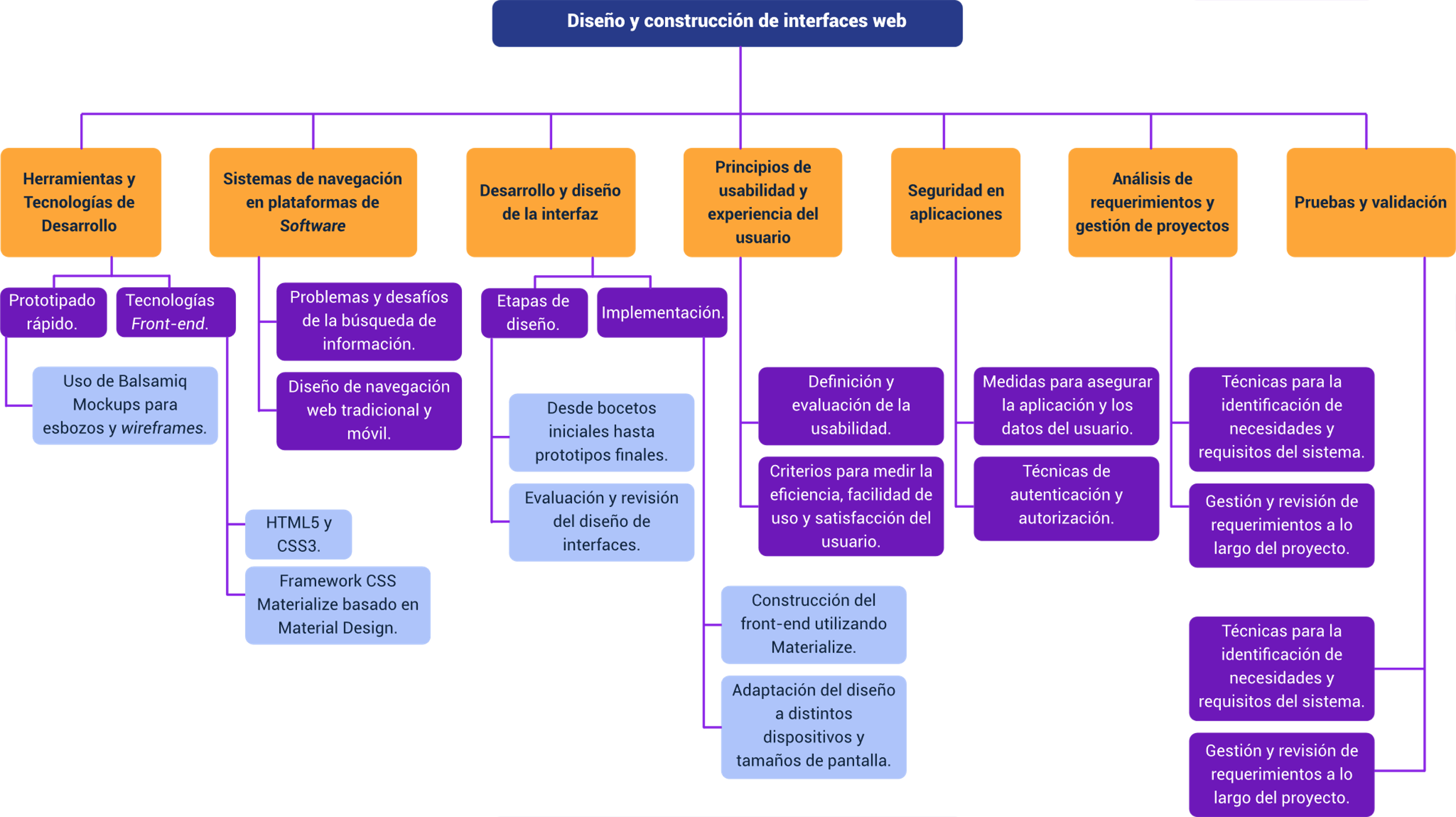
* **PostMan**

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

Se puede concluir que existe una amplia variedad de herramientas disponibles para todas las fases de desarrollo e implementación de aplicaciones web y móviles. Estas herramientas ofrecen diversos tipos de licenciamiento, y muchas de ellas son de libre distribución, lo que permite iniciar un proyecto sin incurrir en costos iniciales significativos.

Síntesis

A continuación, se presenta una síntesis de la temática estudiada en el componente formativo.



Material complementario

| Tema | Referencia | Tipo de material | Enlace del recurso |
| --- | --- | --- | --- |
| Sistema de interacción en plataformas de software | UXTips. (2018). ¿Qué es Arquitectura de la Información? [Vídeo]. YouTube. | Video | <https://www.youtube.com/watch?v=twVvK1eYmig> |
| Estructura de la interfaz de la aplicación | yReyALD. (2020). BALSAMIQ Mockups / Instalación y maquetación de una aplicación móvil - FACIL (música GOT de fondo) [Vídeo]. YouTube. | Video | <https://www.youtube.com/watch?v=idX3aFx67Vc> |
| Estructura de la interfaz de la aplicación | Bluuweb! (2018). Materialize CSS Material Design framework [Curso tutorial Español] [Vídeo]. YouTube. | Video | <https://www.youtube.com/watch?v=YJKWkUi2r5g> |
| Construcción de la estructura del front-end | Materialize.Zip (1.0.0). (2021). [framework CSS]. | Archivo | <https://github.com/Dogfalo/materialize/releases/download/1.0.0/materialize-v1.0.0.zip> |
| Construcción de la estructura del front-end | Materialize. (2021). Getting Started - Materialize. | Guía de instalación | <https://materializecss.com/getting-started.html> |
| Diseño gráfico: herramientas de prototipado | FalconMasters. (2018). Curso de Adobe XD | 1. Introducción y que Aprenderemos en el Curso [Video] Youtube. | Video | <https://www.youtube.com/watch?v=5HfqCSm9k3Y&feature=youtu.be> |
| Diseño gráfico: herramientas de prototipado | EDteam. (2017). Minicurso de Balsamiq Mockups (1) - Los wireframes y la interfaz. [Video] Youtube. | Video | <https://www.youtube.com/watch?v=4WqfxF1QjTM&feature=youtu.be> |
| Diseño gráfico: herramientas de prototipado | Aprende, G. (2020). COMO USAR CANVA: Canva tutorial - Aprende a diseñar FÁCIL y de manera PROFESIONAL. [Video] Youtube. | Video | <https://www.youtube.com/watch?v=kdE6ZmT_5MI> |
| Diseño gráfico: herramientas de prototipado | Conde, J. (2016). 01.- Material Design con Android Studio. ¿Qué es Material Design? [Video] Youtube. | Video | <https://www.youtube.com/watch?v=wt0Jzc9UHNw> |
| API Firebase | Novato, P. (2019). Firebase 01.- Que es Firebase. [Video] Youtube. | Video | <https://www.youtube.com/watch?v=0Bhrwc8VfGw> |

Glosario

**Accesibilidad**: la accesibilidad web hace referencia a un diseño web que permite que las personas con cierta limitación puedan percibir, entender, navegar e interactuar con la Web, aportando a su vez contenidos. La accesibilidad web además beneficia a otras personas, incluyendo a adultos mayores que han visto mermadas sus habilidades a consecuencia de la edad.

**Material Design**: se puede entender como un lineamiento o principio de diseño enfocado en la visualización del sistema operativo Android. Este concepto también abarca para el diseño en la web y en cualquier plataforma. Fue desarrollado por Google.

**Usabilidad**: dentro del contexto de la informática y tecnologías, es un concepto utilizado para mencionar la facilidad con que las personas pueden utilizar una herramienta software particular o cualquier otro objeto tecnológico fabricado por humanos con el fin de alcanzar un objetivo concreto o satisfacer una necesidad.

**Wereable**: 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>

Calvo, L. (2015). Sistemas de navegación. Sistemas de Navegación, Madrid, España.

Curso Wordpress. (2014). Mapa del Sitio, Navegación y Contenidos. Curso WordPress Gratis. <https://cursowpress.wordpress.com/mapa-del-sitio-y-estrategia-de-navegacion/>

Ferraris, J. C. (2018). Patrones básicos de Navegación en Apps Móviles. Medium. <https://medium.com/@juancaferraris/patrones-b%C3%A1sicos-de-navegaci%C3%B3n-en-apps-m%C3%B3viles-5b0b160ed1bb>

IONOS. (2021). Tendencias de diseño web. IONOS Digital Guide. <https://www.ionos.es/digitalguide/paginas-web/diseno-web/tendencias-de-diseno-web/>

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

Materialize. (2021). Getting Started - Materialize. <https://materializecss.com/getting-started.html>

Créditos

| Nombre | Cargo | Centro de Formación y Regional |
| --- | --- | --- |
| Milady Tatiana Villamil Castellanos | Responsable del Ecosistema | Dirección General |
| Olga Constanza Bermúdez Jaimes | Responsable de Línea de Producción | Centro de Servicios de Salud - Regional Antioquia |
| Santiago Muñoz de la Rosa | Experto Temático | Centro de Teleinformática y Producción Industrial - Regional Cauca |
| Peter Emerson Pinchao | Experto Temático | Centro de Teleinformática y Producción Industrial - Regional Cauca |
| Paola Alexandra Moya Peralta | Evaluadora Instruccional | Centro de Servicios de Salud - Regional Antioquia |
| Andrés Felipe Herrera Roldán | Diseñador de Contenidos Digitales | Centro de Servicios de Salud - Regional Antioquia |
| Edwin Sneider Velandia Suárez | Desarrollador Fullstack | Centro de Servicios de Salud - Regional Antioquia |
| Edgar Mauricio Cortés García | Actividad Didáctica | Centro de Servicios de Salud - Regional Antioquia |
| Luis Gabriel Urueta Álvarez | Validador de Recursos Educativos Digitales | Centro de Servicios de Salud - Regional Antioquia |
| Margarita Marcela Medrano Gómez | Evaluador para Contenidos Inclusivos y Accesibles | Centro de Servicios de Salud - Regional Antioquia |
| Daniel Ricardo Mutis Gómez | Evaluador para Contenidos Inclusivos y Accesibles | Centro de Servicios de Salud - Regional Antioquia |