

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 988 473**

51 Int. Cl.:

G06F 8/76 (2008.01)

G06F 8/35 (2008.01)

G06F 11/30 (2006.01)

G06F 11/36 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **19.08.2020 PCT/CN2020/110026**

87 Fecha y número de publicación internacional: **11.03.2021 WO21042991**

96 Fecha de presentación y número de la solicitud europea: **19.08.2020 E 20860819 (0)**

97 Fecha y número de publicación de la concesión europea: **19.06.2024 EP 4024204**

54 Título: **Método de interacción no intrusiva y dispositivo electrónico**

30 Prioridad:

05.09.2019 CN 201910840922
19.09.2019 CN 201910886241

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
20.11.2024

73 Titular/es:

HUAWEI DEVICE CO., LTD. (100.0%)
B2-5 of Nanfang Factory No.2 of Xincheng
Road Songshan Lake Science and
Technology Industrial Zone Dongguan
Guangdong 523808, CN

72 Inventor/es:

WANG, JIANFENG;
CHEN, HAO;
YANG, XIAOYUN;
GUAN, CHI y
CHEN, XIAOXIAO

74 Agente/Representante:

SÁNCHEZ SILVA, Jesús Eladio

ES 2 988 473 T3

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín Europeo de Patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre Concesión de Patentes Europeas).

DESCRIPCIÓN

Método de interacción no intrusiva y dispositivo electrónico

5 Campo técnico

Esta solicitud se refiere al campo de los dispositivos electrónicos y, en particular, a un método de interacción no intrusiva y un dispositivo electrónico.

10 Antecedentes

En la actualidad, los dispositivos electrónicos tales como los teléfonos móviles se han convertido en herramientas de comunicación indispensables en la vida diaria y el trabajo de las personas. Se instala una aplicación (aplicación, APP para abreviar) en un dispositivo electrónico para mejorar las funciones del dispositivo electrónico y proporcionar una experiencia de uso más rica para el usuario. La implementación de algunas funciones de la aplicación se basa en un servicio de capacidad del dispositivo proporcionado por un sistema operativo del dispositivo electrónico.

15

Actualmente, se utiliza una forma de interacción intrusiva para la interacción entre una aplicación y un sistema operativo de un dispositivo electrónico. Específicamente, el sistema operativo del dispositivo electrónico proporciona servicios de capacidad del dispositivo utilizados para implementar diferentes funciones y proporciona las interfaces funcionales correspondientes para la aplicación. Las interfaces funcionales pueden encapsularse en un kit de desarrollo de software (kit de desarrollo de software, SDK), que se denomina paquete de interfaz funcional. Tras integrar el paquete de interfaz funcional, la aplicación puede invocar un servicio de capacidad del dispositivo correspondiente para implementar una función correspondiente.

20

Esta forma de interacción intrusiva tiene al menos las siguientes desventajas: No se puede invocar un servicio de capacidad del dispositivo en todos los sistemas operativos. Es decir, si un paquete de interfaz funcional se implementa en función de un sistema operativo (por ejemplo, un sistema operativo Android), el paquete de interfaz funcional solo puede invocar un servicio de capacidad de dispositivo correspondiente proporcionado por el sistema operativo y no puede invocar un mismo servicio de capacidad de dispositivo proporcionado por otro sistema operativo. No se puede invocar un servicio de capacidad de dispositivos en todos los dispositivos. Es decir, una aplicación instalada en un dispositivo electrónico integra un paquete de interfaz funcional, puede invocar, mediante el uso del paquete de interfaz funcional, solo un servicio de capacidad de dispositivo correspondiente proporcionado por un sistema operativo del dispositivo electrónico y no puede invocar un mismo servicio de capacidad de dispositivo proporcionado por un sistema operativo de otro dispositivo. La aplicación o el paquete de interfaz funcional deben proporcionar una buena compatibilidad. De lo contrario, incluso si la aplicación integra el paquete de interfaz funcional, la aplicación no puede invocar el servicio de capacidad del dispositivo correspondiente. El sistema operativo del dispositivo electrónico proporciona muchos tipos de servicios de capacidad del dispositivo y muchos paquetes de interfaz funcional correspondientes. En otras palabras, los paquetes de interfaz funcional están fragmentados. Esto hace que una aplicación integre muchos paquetes de interfaz funcionales.

25

30

35

40

QI TIELIN ET AL, “A Model Transformation Platform Design Based on Model Driven Architecture”, INTELLIGENT COMPUTATION TECHNOLOGY AND AUTOMATION (ICICTA), 2010 INTERNATIONAL CONFERENCE ON, IEEE, PISCATAWAY, NJ, EE. UU., (20100511), ISBN 978-1-4244-7279-6, páginas 175 a 179, se refiere al diseño de plataformas de transformación de modelos basadas en una arquitectura basada en modelos.

45

Breve descripción de la invención

Las implementaciones de esta solicitud proporcionan un método de interacción no intrusiva y un dispositivo electrónico. Una aplicación puede invocar un servicio de capacidad del dispositivo de una manera de interacción no intrusiva para implementar una función correspondiente, sin integrar varios paquetes de interfaz funcional. Además, el servicio de capacidad del dispositivo se puede invocar en todos los sistemas operativos y dispositivos.

50

La invención se define en las reivindicaciones independientes. Otras características técnicas específicas se han definido en las reivindicaciones dependientes.

55

Al usar la solución técnica anterior, después de obtener el archivo de descripción que pertenece a la aplicación y que se define mediante el uso de la descripción del protocolo no intrusiva, el dispositivo electrónico puede determinar, a partir de los componentes del dispositivo electrónico en función del archivo de descripción, el componente que puede implementar la función que debe implementar la aplicación. El componente del dispositivo electrónico es un módulo que se encuentra en el dispositivo electrónico y que está configurado en función de un protocolo no intrusivo para proporcionar un servicio de capacidad del dispositivo y puede implementar una función independiente. El dispositivo electrónico ejecuta el componente determinado en función del archivo de descripción para proporcionar el servicio de capacidad del dispositivo para la aplicación, a fin de implementar la función que debe implementar la aplicación. De esta manera, la aplicación en el dispositivo electrónico puede invocar el servicio de capacidad del dispositivo de una

60

65

manera de interacción no intrusiva para implementar una función correspondiente. Además, el servicio de capacidad del dispositivo se puede invocar en todos los sistemas operativos y dispositivos.

5 Basándose en ciertos principios descritos en esta solicitud, se puede aprovechar al máximo la ventaja multidispositivo de un lenguaje descriptivo utilizando la descripción del protocolo no intrusiva para generar el archivo de descripción.

Basándose en ciertos principios descritos en esta solicitud, se puede mejorar la eficiencia de funcionamiento del componente.

10 Basándose en ciertos principios descritos en esta solicitud, se puede usar un proceso para implementar la interacción del archivo de descripción en un escenario en el que los datos requeridos cuando un componente correspondiente proporciona un servicio varían según el entorno en ejecución.

15 Basándose en ciertos principios descritos en esta solicitud, cuando el dispositivo electrónico no tiene ningún componente que pueda proporcionar el servicio para la aplicación, el dispositivo electrónico puede enviar el archivo de descripción a otro dispositivo que pueda proporcionar el servicio correspondiente, implementando así la invocación del servicio con capacidad multidispositivo.

20 Breve descripción de los dibujos

La figura 1 es un diagrama esquemático de una estructura de un dispositivo electrónico según una implementación de esta solicitud;

25 la figura 2 es un diagrama de composición esquemático de una arquitectura de software según una implementación de esta solicitud;

la figura 3 es un diagrama de flujo esquemático de interacción no intrusiva según una implementación de esta solicitud;

30 la figura 4 es otro diagrama de flujo esquemático de interacción no intrusiva según una implementación de esta solicitud;

la figura 5(a) y la figura 5(b) son un diagrama esquemático de una interfaz de visualización según una implementación de esta solicitud;

35 la figura 6(a) a la figura 6(d) son un diagrama esquemático de otra interfaz de visualización según una implementación de esta aplicación; y

la figura 7 es un diagrama esquemático de todavía otra interfaz de visualización según una implementación de esta solicitud;

40 Descripción de implementaciones

45 Para aclarar los objetivos, las soluciones técnicas y las ventajas de la presente invención, a continuación se describen con mayor detalle diversas implementaciones con referencia a los dibujos adjuntos. No todas las implementaciones descritas a continuación se reivindican, se incluyen para ayudar a comprender el contexto de la invención. Si bien la descripción se refiere a diversas implementaciones, las realizaciones de la invención son aquellas que comprenden al menos todas las características de una reivindicación independiente. Cualquier implementación que no entre en el alcance de las reivindicaciones no forma parte de la invención, sino que se incluye como ejemplo ilustrativo que es útil para comprender la invención.

50 Las implementaciones de esta aplicación proporcionan un método de interacción no intrusiva. El método puede realizarse mediante un dispositivo electrónico que incluye una aplicación. Al usar el método, la aplicación en el dispositivo electrónico puede invocar un servicio de capacidad del dispositivo de una manera de interacción no intrusiva para implementar una función correspondiente. Además, el servicio de capacidad del dispositivo se puede invocar en todos los sistemas operativos y dispositivos.

55 Por ejemplo, la aplicación en las implementaciones de esta aplicación puede ser una aplicación integrada (es decir, una aplicación de sistema del dispositivo electrónico) o puede ser una aplicación descargable. La aplicación integrada es una aplicación proporcionada como parte de la implementación del dispositivo electrónico. La aplicación descargable es una aplicación que puede proporcionar una conexión de subsistema multimedia de protocolo de internet (subsistema multimedia de protocolo de internet, IMS) de la aplicación descargable. La aplicación descargable puede ser una aplicación preinstalada en el dispositivo electrónico, o puede ser una aplicación de terceros descargada e instalada por un usuario en el dispositivo electrónico.

65 Por ejemplo, el dispositivo electrónico en las implementaciones de esta aplicación puede ser un dispositivo como un teléfono móvil, una tableta, un ordenador de sobremesa, un ordenador portátil, un ordenador de mano, un ordenador

portátil, un ordenador personal ultramóvil (UMPC), un netbook, un teléfono móvil, un asistente digital personal (PDA), un dispositivo de realidad aumentada (AR)/realidad virtual (VR) o un reproductor multimedia. Una forma específica del dispositivo no se limita específicamente en las implementaciones de esta solicitud.

5 A continuación se describen con detalle las implementaciones de las implementaciones de la presente solicitud haciendo referencia a los dibujos adjuntos.

La figura 1 es un diagrama esquemático de una estructura de un dispositivo electrónico según una implementación de esta solicitud. Como se muestra en la figura 1, el dispositivo electrónico 1 puede incluir un procesador 110, una interfaz 10 120 de memoria externa, una memoria interna 121, una interfaz 130 de bus serie universal (bus serie universal, USB), un módulo 140 de gestión de carga, un módulo 141 de gestión de energía, una batería 142, una antena 1, una antena 2, un módulo 150 de comunicaciones móviles, un módulo 160 de comunicaciones inalámbricas, un módulo 170 de audio, un altavoz 170A, un receptor 170B, un micrófono 170C, un conector 170D de cascos, un módulo 180 de 15 sensores, un botón 190, un motor 191, un indicador 192, una cámara 193, un visualizador 194, una interfaz 195 de tarjeta de módulo de identificación de abonado (módulo de identificación de abonado, SIM) y similares.

El módulo 180 de sensores puede incluir un sensor 180A de presión, un sensor 180B de giroscopio, un sensor 180C de presión barométrica, un sensor magnético 180D, un sensor 180E de aceleración, un sensor 180F de distancia, un 20 sensor 180G óptico de proximidad, un sensor 180H de huella dactilar, un sensor 180J de temperatura, un sensor táctil 180K, un sensor de luz ambiental 180L, un sensor 180M de conducción ósea y similares.

Puede entenderse que la estructura que se muestra en esta implementación no constituye una limitación específica del dispositivo electrónico. En algunas otras implementaciones, el dispositivo electrónico puede incluir más o menos 25 componentes que los mostrados en la figura, o algunos componentes pueden combinarse, o algunos componentes pueden dividirse, o pueden usarse distintas disposiciones de componentes. Los componentes mostrados en la figura pueden implementarse mediante hardware, software o una combinación de software y hardware.

El procesador 110 puede incluir una o más unidades de procesamiento. Por ejemplo, el procesador 110 puede incluir 30 un procesador de aplicaciones (Application Processor - AP), un procesador de módem, una unidad de procesamiento de gráficos (Graphics Processing Unit - GPU), un procesador de señales de imagen (Image Signal Processor - ISP), un controlador, una memoria, un códec de vídeo, un procesador de señales digitales (Digital Signal Processor - DSP), un procesador de banda base, una unidad de procesamiento de red neuronal (Neural-network Processing Unit - NPU) y/o componentes similares. Distintas unidades de procesamiento pueden ser dispositivos independientes o pueden 35 estar integradas en uno o más procesadores.

El controlador puede ser un centro nervioso y un centro de mando del dispositivo electrónico. El controlador puede generar una señal de control de operación en función de un código de operación por instrucción y una señal de 40 secuencia de tiempo para realizar un control absoluto de una lectura de instrucción y de una ejecución de instrucción.

Además, en el procesador 110 puede disponerse en una memoria que esté configurada para almacenar instrucciones y datos. En algunas implementaciones, la memoria en el procesador 110 es una memoria caché. La memoria puede 45 almacenar instrucciones o datos que hayan sido justo usados o usados cíclicamente por el procesador 110.

En algunas implementaciones, el procesador 110 puede incluir una o más interfaces. La interfaz puede incluir una 45 interfaz de circuito interintegrado (circuito interintegrado, I2C), una interfaz de circuito interintegrado sonoro (circuito interintegrado sonoro, I2S), una interfaz de modulación por impulsos codificados (modulación por impulsos codificados, PCM), una interfaz de receptor/transmisor asíncrono universal (receptor/transmisor asíncrono universal, UART), una interfaz de procesador industrial móvil (interfaz de procesador industrial móvil, MIPI), una interfaz de entrada/salida de propósito general (entrada/salida de propósito general, GPIO), una interfaz SIM, una interfaz USB, y/o similares. 50

El módulo 140 de gestión de carga está configurado para recibir desde un cargador una entrada de carga. El cargador puede ser un cargador inalámbrico o un cargador cableado. El módulo 140 de gestión de carga también puede 55 suministrar energía al dispositivo electrónico usando el módulo 141 de gestión de energía mientras carga la batería 142.

El módulo 141 de gestión de energía está configurado para conectarse a la batería 142 y al módulo 140 de gestión de carga y al procesador 110. El módulo 141 de gestión de energía recibe una entrada de la batería 142 y/o del módulo 140 de gestión de carga y suministra energía al procesador 110, la memoria interna 121, una memoria externa, el 60 visualizador 194, la cámara 193, el módulo 160 de comunicaciones inalámbricas y componentes similares. En algunas otras implementaciones, el módulo 141 de gestión de energía puede disponerse, alternativamente, en el procesador 110. En algunas otras implementaciones, el módulo 141 de gestión de energía y el módulo 140 de gestión de carga pueden disponerse, alternativamente, en un mismo dispositivo.

Una función de comunicación inalámbrica del dispositivo electrónico puede implementarse mediante la antena 1, la 65 antena 2, el módulo 150 de comunicaciones móviles, el módulo 160 de comunicaciones inalámbricas, el procesador de módem, el procesador de banda base y componentes similares.

La antena 1 y la antena 2 están configuradas para transmitir y recibir señales de onda electromagnética. Cada antena del dispositivo electrónico puede configurarse para cubrir una o más bandas de frecuencias de comunicaciones. Además, pueden multiplexarse distintas antenas para mejorar la utilización de antena.

5 El módulo de comunicación móvil 150 puede proporcionar una solución para la comunicación inalámbrica que incluye 2G/3G/4G/5G y similares, y que se usa en el dispositivo electrónico. El módulo 150 de comunicaciones móviles puede
 10 incluir al menos un filtro, un interruptor, un amplificador de potencia, un amplificador de bajo ruido (amplificador de bajo ruido, LNA) y componentes similares. El módulo 150 de comunicaciones móviles puede recibir una onda electromagnética mediante la antena 1, realizar un procesamiento, tal como un filtrado y una amplificación, de la onda
 15 electromagnética recibida y transferir una onda electromagnética procesada al procesador de módem para su demodulación. El módulo 150 de comunicaciones móviles puede además amplificar una señal modulada por el procesador de módem y convertirla a una onda electromagnética para radiarse mediante la antena 1. En algunas implementaciones, al menos algunos módulos funcionales del módulo 150 de comunicaciones móviles pueden estar dispuestos en el procesador 110. En algunas implementaciones, al menos algunos módulos funcionales del módulo 150 de comunicaciones móviles pueden estar dispuestos en un mismo dispositivo, al igual que al menos algunos módulos en el procesador 110.

En algunas implementaciones, el procesador de módem puede ser un componente independiente. En algunas otras implementaciones, el procesador de módem puede ser independiente del procesador 110, y está dispuesto en un mismo componente al igual que el módulo 150 de comunicaciones móviles u otro módulo funcional.

El módulo de comunicación inalámbrica 160 puede proporcionar soluciones de comunicación inalámbrica, utilizadas en el dispositivo electrónico, por ejemplo, una red de área local inalámbrica (WLAN) (como una red de fidelidad inalámbrica (Wi-Fi)), Bluetooth (Bluetooth, BT), un sistema global de navegación por satélite (GNSS), modulación de frecuencia (FM), una tecnología de comunicación de campo cercano (NFC) y una tecnología de infrarrojos (IR). El módulo de comunicación inalámbrica 160 puede ser uno o más dispositivos que integran al menos un módulo procesador de comunicación. El módulo 160 de comunicaciones inalámbricas recibe una onda electromagnética mediante la antena 2, realiza un procesamiento de modulación y filtrado de frecuencia de una señal de onda electromagnética y envía una señal procesada al procesador 110. El módulo 160 de comunicaciones inalámbricas puede además recibir del procesador 110 una señal a enviar, realizar una modulación y una amplificación de frecuencia de la señal y convertirla en una onda electromagnética para radiarse mediante la antena 2.

En algunas implementaciones, la antena 1 del dispositivo electrónico está acoplada al módulo de comunicación móvil 150, y la antena 2 está acoplada al módulo de comunicación inalámbrica 160, de modo que el dispositivo electrónico puede comunicarse con una red y otro dispositivo mediante el uso de una tecnología de comunicación inalámbrica. La tecnología de comunicaciones inalámbricas puede incluir un sistema global de comunicaciones móviles (GSM), un servicio general de radio por paquetes (GPRS), un acceso múltiple por división de código (CDMA), un acceso múltiple por división de código de banda ancha (WCDMA), un acceso múltiple por división de código por división de tiempo (TD-SCDMA), una evolución a largo plazo (LTE), una tecnología BT, un sistema GNSS, una red WLAN, una comunicación NFC, una modulación FM, una tecnología IR y/o tecnologías similares. El sistema GNSS puede incluir un sistema de posicionamiento global (Global Positioning System - GPS), un sistema mundial de navegación por satélite (Global Navigation Satellite System - GLONASS), un sistema de navegación por satélite BeiDou (BeiDou Navigation Satellite System - BDS), un sistema por satélite cuasicenital (Quasi-Zenith Satellite System - QZSS) y/o un sistema de aumentación basado en satélites (Satellite Based Augmentation Systems - SBAS).

El dispositivo electrónico implementa una función de visualización usando la unidad GPU, el visualizador 194, el procesador de aplicaciones y componentes similares. La unidad GPU es un microprocesador para procesar imágenes y está conectada al visualizador 194 y al procesador de aplicaciones. La unidad GPU está configurada para: realizar un cálculo matemático y geométrico y representar una imagen. El procesador 110 puede incluir una o más unidades GPU, que ejecutan instrucciones de programa para generar o cambiar información de visualización.

El visualizador 194 está configurado para visualizar una imagen, un vídeo y similares. El visualizador 194 incluye un panel de visualización. El panel de visualización puede ser un visualizador de cristal líquido (LCD), un diodo emisor de luz orgánico (OLED), un diodo emisor de luz orgánico de matriz activa (AMOLED), un diodo emisor de luz flexible (FLED), un miniLED, microLED, un micro OLED, un diodo emisor de luz de punto cuántico (QLED) o similares. En algunas implementaciones, el dispositivo electrónico puede incluir uno o N visualizadores 194, donde N es un entero positivo mayor que 1.

El dispositivo electrónico puede implementar una función de realización de fotografías usando el procesador ISP, la cámara 193, el códec de vídeo, la unidad GPU, el visualizador 194, el procesador de aplicaciones y componentes similares.

El procesador ISP está configurado para procesar datos facilitados por la cámara 193. En algunas implementaciones, el procesador ISP puede estar dispuesto en la cámara 193.

La cámara 193 está configurada para capturar una imagen estática o un vídeo. En algunas implementaciones, el dispositivo electrónico puede incluir una o N cámaras 193, donde N es un entero positivo mayor que 1.

El procesador de señales digitales está configurado para procesar una señal digital y puede procesar otra señal digital además de la señal de imagen digital. El códec de vídeo está configurado para comprimir o descomprimir un vídeo digital. La NPU es un procesador informático de red neuronal (red neuronal, NN). La NPU procesa rápidamente la información de entrada con referencia a una estructura de una red neuronal biológica, por ejemplo, un modo de transferencia entre neuronas del cerebro humano, y además puede realizar un autoaprendizaje continuo. Una aplicación como la cognición inteligente, por ejemplo, el reconocimiento de imágenes, el reconocimiento facial, el reconocimiento de voz o la comprensión del texto del dispositivo electrónico puede implementarse mediante el uso de la NPU.

La interfaz 120 de memoria externa puede configurarse para conectar a una tarjeta de memoria externa, por ejemplo, una tarjeta micro SD a fin de ampliar una capacidad de almacenamiento del dispositivo electrónico.

La memoria interna 121 puede configurarse para almacenar un código de programa ejecutable por ordenador. El código del programa ejecutable incluye instrucciones. El procesador 110 ejecuta las instrucciones almacenadas en la memoria interna 121 para realizar diversas aplicaciones de función y un procesamiento de datos del dispositivo electrónico. La memoria interna 121 puede incluir una zona de almacenamiento de programas y una zona de almacenamiento de datos. La zona de almacenamiento de programas puede almacenar un sistema operativo, una aplicación requerida por al menos una función (p. ej., una función de reproducción de sonido o una función de reproducción de imágenes) y similares. El área de almacenamiento de datos puede almacenar datos (como datos de audio y una agenda telefónica) y similares creados cuando se utiliza el dispositivo electrónico, y similares. Además, la memoria interna 121 puede incluir una memoria de acceso aleatorio de alta velocidad o puede también incluir una memoria no volátil, por ejemplo, al menos un dispositivo de almacenamiento de disco magnético, una memoria flash o un almacenamiento flash universal (Universal Flash Storage - UFS).

El dispositivo electrónico puede implementar funciones de audio, por ejemplo, una reproducción o una grabación de música, usando el módulo 170 de audio, el altavoz 170A, el receptor 170B, el micrófono 170C, el conector 170D de cascos, el procesador de aplicaciones y componentes similares.

El módulo 170 de audio está configurado para convertir una información digital de audio en una salida de señal analógica de audio, y también está configurado para convertir una entrada analógica de audio en una señal digital de audio. El módulo 170 de audio puede además configurarse para codificar y decodificar una señal de audio. El altavoz 170A, también denominado “bocina”, está configurado para convertir una señal eléctrica de audio en una señal acústica. El dispositivo electrónico puede escuchar música o responder una llamada en un modo manos libres usando el altavoz 170A. El receptor 170B, también denominado “auricular”, está configurado para convertir una señal eléctrica de audio en una señal acústica. Cuando se responde a una llamada o se recibe información de voz utilizando el dispositivo electrónico, el receptor 170B puede colocarse cerca de un oído humano para recibir una voz. El micrófono 170C, también denominado “micro” o “micrófono”, está configurado para convertir una señal acústica en una señal eléctrica. En el dispositivo electrónico se puede disponer al menos un micrófono 170C. El conector 170D de cascos está configurado para conectar a unos cascos cableados.

El sensor 180A de presión está configurado para detectar una señal de presión, y puede convertir la señal de presión en una señal eléctrica. En algunas implementaciones, el sensor 180A de presión puede estar dispuesto en el visualizador 194. Existen una pluralidad de tipos de sensores 180A de presión, por ejemplo, un sensor de presión resistivo, un sensor de presión inductivo y un sensor de presión capacitivo. El sensor 180B de giroscopio se puede configurar para determinar una postura en movimiento del dispositivo electrónico.

El sensor 180C de presión barométrica está configurado para medir la presión barométrica. En algunas implementaciones, el dispositivo electrónico calcula una altitud en base a un valor de la presión barométrica medida por el sensor 180C de presión barométrica para ayudar en el posicionamiento y la navegación. El sensor magnético 180D incluye un sensor de efecto Hall. El dispositivo electrónico puede detectar la apertura y el cierre de una cubierta plegable usando el sensor magnético 180D. El sensor de aceleración 180E puede detectar la magnitud de las aceleraciones en varias direcciones (normalmente en tres ejes) del dispositivo electrónico. El sensor de aceleración 180E puede configurarse además para identificar una postura del dispositivo electrónico. El sensor 180F de distancia está configurado para medir una distancia. El dispositivo electrónico puede medir la distancia de manera infrarroja o de manera láser.

El sensor 180G óptico de proximidad puede incluir un diodo emisor de luz (LED) y un detector óptico, por ejemplo, un fotodiodo. El dispositivo electrónico emite luz infrarroja mediante el diodo emisor de luz. El dispositivo electrónico detecta una luz infrarroja reflejada por un objeto cercano usando el fotodiodo. Cuando se detecta suficiente luz reflejada, puede determinarse que hay un objeto cerca del dispositivo electrónico. Cuando se detecta insuficiente luz reflejada, el dispositivo electrónico puede determinar que no hay un objeto cerca del mismo.

5 El sensor 180L de luz ambiente está configurado para detectar un brillo de luz ambiente. El dispositivo electrónico puede ajustar de manera adaptativa el brillo del visualizador 194 en función del brillo de luz ambiente detectado. El sensor 180L de luz ambiente también se puede configurar para ajustar automáticamente un balance de blancos durante la realización de fotografías. El sensor 180L de luz ambiente también puede cooperar con el sensor 180G óptico de proximidad para detectar si el dispositivo electrónico está en un bolsillo a fin de impedir un tacto accidental.

10 El sensor 180H de huella dactilar está configurado para captar una huella dactilar. El dispositivo electrónico puede usar una característica de la huella dactilar captada para implementar un desbloqueo basado en la huella dactilar, un acceso de bloqueo de aplicación, una realización de fotografías basada en la huella dactilar, una respuesta a una llamada basada en la huella dactilar y acciones similares. El sensor 180J de temperatura está configurado para detectar una temperatura. En algunas implementaciones, el dispositivo electrónico implementa una política de procesamiento de temperatura usando la temperatura detectada por el sensor 180J de temperatura.

15 El sensor táctil 180K también se denomina “panel táctil”. El sensor táctil 180K puede disponerse en el visualizador 194, y el sensor táctil 180K y el visualizador 194 forman una pantalla táctil, denominada “pantalla táctil”. El sensor táctil 180K está configurado para detectar una operación táctil realizada sobre o cerca del sensor táctil 180K. El sensor táctil puede transferir la operación táctil detectada al procesador de aplicaciones para que determine un tipo de evento táctil. Puede proporcionarse una salida visual relacionada con la operación táctil en el visualizador 194. En algunas otras implementaciones, el sensor táctil 180K alternativamente puede estar dispuesto en una superficie del dispositivo electrónico en una posición diferente a la del visualizador 194.

20 El sensor 180M de conducción ósea puede obtener una señal de vibración. En algunas implementaciones, el sensor 180M de conducción ósea también se puede disponer en los cascos para obtener unos cascos de conducción ósea. El módulo 170 de audio puede obtener una señal de voz por análisis en función de la señal de vibración, que procede del hueso de vibración de la parte de cuerda vocal y que es obtenida por el sensor 180M de conducción ósea, para implementar una función de voz. El botón 190 incluye un botón de encendido, un botón de volumen y similares. El botón 190 puede ser un botón mecánico o puede ser un botón táctil. El dispositivo electrónico puede recibir una entrada de tecla y generar una entrada de señal de tecla relacionada con un ajuste de usuario y un control de función del dispositivo electrónico.

25 El motor 191 puede generar un incitador vibratorio. El motor 191 puede configurarse para producir un incitador vibratorio de llamada entrante y una retroalimentación por vibración táctil. El indicador 192 puede ser una luz indicadora y puede estar configurado para indicar un estado de carga y un cambio de energía, o bien puede estar configurado para indicar un mensaje, una llamada perdida, una notificación y similares. La interfaz 195 de tarjeta SIM está configurada para conectar a una tarjeta SIM. La tarjeta SIM puede insertarse en la interfaz 195 de tarjeta SIM o sacarse de la interfaz 195 de tarjeta SIM para implementar un contacto con, o una separación de, el dispositivo electrónico. El dispositivo electrónico puede admitir una o N interfaces de tarjeta SIM, donde N es un entero positivo mayor que 1.

30 Las siguientes implementaciones pueden implementarse en un dispositivo electrónico que tenga la estructura de hardware anterior.

35 Además, un sistema de software del dispositivo electrónico puede usar una arquitectura en capas, una arquitectura dirigida por eventos, una arquitectura de micronúcleos, una arquitectura de microservicios o una arquitectura en la nube. En las implementaciones de esta aplicación, se describe una estructura de software del dispositivo electrónico usando un sistema Android con una arquitectura en capas como ejemplo.

40 La figura 2 es un diagrama de composición esquemático de una arquitectura de software según una implementación de esta solicitud. En una arquitectura en capas, el software está dividido en varias capas, y cada capa tiene una función y una tarea definidas. Las capas se comunican entre sí mediante una interfaz de software. En algunas implementaciones, un sistema Android se divide en tres capas: una capa de aplicación, una capa de bus y una capa de servicio de componentes de arriba a abajo. En función de la arquitectura del software, una aplicación y un sistema operativo pueden invocar un servicio de capacidad del dispositivo de una manera de interacción no intrusiva. La arquitectura de software puede usarse en un escenario en el que un proveedor de servicios y un desarrollador de aplicaciones están separados en una pluralidad de dispositivos.

45 La capa de aplicación puede incluir una serie de paquetes de aplicaciones. Por ejemplo, los paquetes de aplicaciones pueden incluir aplicaciones tales como Amazon, WeChat, Settings, Calculator, Camera, Toutiao, mensajes SMS y reproductor de música. En esta implementación, algunas funciones de las aplicaciones deben implementarse invocando un servicio de capacidad del dispositivo. Cuando una aplicación necesita invocar un servicio de capacidad del dispositivo para implementar una función correspondiente, la aplicación entrega un archivo de descripción a la capa de bus. El archivo de descripción se usa para indicar la función que debe implementar la aplicación.

50 La capa de bus puede incluir un módulo de bus. El módulo de bus es responsable de conectar una aplicación en la capa de aplicación y un componente en la capa de servicio del componente. En esta implementación, el módulo de bus puede configurarse para recibir un archivo de descripción de una aplicación desde la capa de aplicación. Una vez recibido el archivo de descripción de la aplicación desde la capa de aplicación, el archivo de descripción puede

analizarse sintácticamente, de modo que un componente correspondiente utilizado para implementar una función que necesita ser implementada por la aplicación se encuentre a partir de los componentes de la capa de servicio del componente en función del archivo de descripción analizado. El componente encontrado se usa para proporcionar un servicio de capacidad de dispositivo para la aplicación, a fin de implementar la función correspondiente.

5 La capa de servicio de componentes puede incluir una pluralidad de componentes, por ejemplo, un componente 1, un componente 2, un componente 3,... y un componente n. Cada componente es un servicio de capacidad del dispositivo que puede completar una función independiente. Diferentes componentes pueden proporcionar un mismo servicio de capacidad de dispositivo o diferentes servicios de capacidad de dispositivo. En esta implementación, después de que el módulo de bus en la capa de bus encuentre un componente correspondiente en la capa de servicio de componente en función de un archivo de descripción entregado por una aplicación en la capa de aplicación, el componente en la capa de servicio de componente puede iniciarse para proporcionar un servicio de capacidad de dispositivo correspondiente para la aplicación, para implementar una función correspondiente. Cabe señalar que la forma portadora de un componente no está específicamente limitada en esta implementación. Por ejemplo, diferentes componentes pueden transportarse en diferentes entidades lógicas para la implementación y, alternativamente, diferentes componentes pueden transportarse en una entidad lógica para la implementación.

A continuación se describe un archivo de descripción entregado por una aplicación en la capa de aplicación.

20 En algunas implementaciones, el archivo de descripción puede definirse usando una descripción de protocolo no intrusiva y se usa para indicar una función que necesita ser implementada por la aplicación.

Por ejemplo, la descripción del protocolo no intrusivo es un lenguaje descriptivo que se usa para soportar la interacción entre la aplicación en la capa de aplicación y el módulo de bus en la capa de bus, de modo que se pueda aprovechar al máximo la ventaja multidispositivo del lenguaje descriptivo.

Según la invención reivindicada, se usa un lenguaje JSON (notación de objetos de JavaScript) para definir una estructura del archivo de descripción. El lenguaje JSON es un formato ligero de intercambio de datos basado en un lenguaje JavaScript.

30 La estructura del archivo de descripción definido mediante el lenguaje JSON puede ser la siguiente:

|-- bloque de identificación: incluye un identificador universal

35 |-- bloque lógico: una entidad lógica cuando se ejecuta un componente

|-- encabezado (encabezado): un valor fijo (Clave), utilizado por el módulo de bus en la capa de bus para analizar y comprobar el archivo de descripción

40 |-- cuerpo: una clave definida por el componente, donde el componente de la capa de servicio del componente analiza esta parte para proporcionar el servicio correspondiente

Un ejemplo de archivo de descripción es el siguiente:

45 "hwEasyGoVersion": "1.0", //Número de versión de protocolo de acceso rápido de Huawei

"client": "com.ss.android.article.news", //Identificador de una aplicación de tercera parte accedida

50 "Info signo": "ALSJFS32KAD90FJ2SAD90FJ23", //Firma

"logicEntities": //Bloque lógico: ejecución de entidades lógicas

{

55 {

"head": { //Bloque lógico: una zona de encabezado, que es una llave universal, donde se realiza una comprobación del encabezado antes de la ejecución

60 "function": "denoising",

"required": "ture",

"hasNext": "false",

65 }

```

    "body": {
5      "inpicture": "/xxx/20190408.jpg",
      "outpicture": "/xxx/20190408_d.jpg",
    }
10  },
    {
15    "head":{//Bloque lógico: una zona de encabezado, que es una llave universal, donde se realiza una
    comprobación del encabezado antes de la ejecución
      "function": "white", //Nombre de componente, que es único en un corto periodo
      "required": "ture",
20    "hasNext": "true", //Hay una función serie para ejecutar
    }
25    "body":{//Logical block: un área del cuerpo, que es una clave autodefinida y que el componente analiza
    durante la ejecución
      "inpicture": "/xxx/20190408.jpg",
30    "salidaimagen": "tmp.yuv",
      "nextFunction":{ //En un modo serie, se anida aquí una función; en un modo paralelo, esto está
    vacío
35      {
        "head": {
40          "function": "thin",
          "required": "ture",
          "hasNext": "false",
45        }
        "body": {
50          "paraSource": "last",
          "paramName": "newPicturePath.xxx",//Obtenido a partir de un valor retornado en
          una etapa previa, donde se requiere serialización
55          "nextFunction": {
            }
          }
60        }
      }
65    }
  }
}

```

}

5 En el ejemplo anterior, EasyGoVersion indica un número de versión de protocolo no intrusivo, el cliente indica un identificador de una aplicación (o un cliente) que usa un protocolo no intrusivo, SignInfo indica una firma de la aplicación que usa el protocolo no intrusivo, LogicEntities indica un conjunto de entidades lógicas, head indica un área de cabecera universal de una entidad lógica, el cuerpo indica un área de datos definida por el componente, la función indica un componente y required indica si un componente necesita ser ejecutado.

10 El archivo de descripción del ejemplo anterior sigue una versión 1.0 del protocolo no intrusivo. La versión 1.0 del protocolo no intrusivo puede incluir campos en el ejemplo anterior. Además, los campos pueden ampliarse.

15 Como se muestra en el ejemplo anterior, las líneas primera a tercera son un bloque de identificación. Se puede observar que el bloque de identificación incluye campos universales tales como un número de versión del protocolo, un identificador de aplicación y una firma de aplicación.

20 Las líneas restantes son un bloque lógico. El bloque lógico es una entidad lógica cuando se ejecuta un componente. El bloque lógico incluye una o más entidades lógicas individuales. En esta implementación, una pluralidad de entidades lógicas puede configurarse en paralelo, y la pluralidad de entidades lógicas puede configurarse alternativamente en serie. En un ejemplo, se puede usar una forma de anidamiento y se puede usar hasNext para indicar si la configuración es en serie o en paralelo. Por ejemplo, "hasNext": "false" se usa para indicar que la configuración es paralela, es decir, indicar que no hay ningún componente en serie que ejecutar; y "hasNext": "ture" se usa para indicar que la configuración es en serie, es decir, indica que hay un componente en serie que ejecutar.

25 Una sola entidad lógica en el bloque lógico representa una instancia en ejecución de un componente, es decir, se invoca un componente para proporcionar el servicio de capacidad del dispositivo correspondiente. Una entidad lógica única incluye un área de la cabeza y un área del cuerpo. El área principal incluye un nombre de componente, que puede indicarse, por ejemplo, mediante una función. El área principal puede incluir además otros campos universales relacionados con un componente, por ejemplo, un identificador que indique si el componente debe ejecutarse (que puede indicarse, por ejemplo, como obligatorio) y un identificador que indique si hay un componente en serie que ejecutar (que puede indicarse, por ejemplo, mediante hasNext). Para otro ejemplo, el área de la cabeza puede incluir además un identificador de un componente específico que se va a usar (que puede indicarse, por ejemplo, mediante un parámetro y no se muestra en el ejemplo anterior). Cuando una pluralidad de componentes puede proporcionar un servicio de capacidad del dispositivo correspondiente a una función que va a utilizar una aplicación, el identificador se puede utilizar para hacer coincidir un componente específico que va a utilizar la aplicación. El área del cuerpo incluye un campo definido por el componente (o una clave definida por el componente). Los datos incluidos en el área corporal están relacionados con una función (o un servicio) que el componente puede completar.

40 La aplicación en la capa de aplicación puede invocar un componente correspondiente entregando el archivo de descripción utilizando la descripción del protocolo no intrusiva al módulo de bus en la capa de bus.

A continuación se describe el módulo de bus en la capa de bus.

45 Haciendo referencia además a la figura 2, el módulo de bus puede incluir una interfaz de capa de aplicación universal, una especificación de protocolo, una infraestructura de protocolo, una infraestructura de componentes, un módulo de enrutamiento de componentes, un módulo de análisis/ejecución de datos, un módulo de almacenamiento de datos y un módulo de interconexión de bus remoto.

50 La interfaz de capa de aplicación universal se puede implementar mediante el uso de una interfaz PostData universal. La aplicación en la capa de aplicación puede transmitir el archivo de descripción al módulo de bus en la capa de bus utilizando la interfaz de capa de aplicación universal.

55 La infraestructura de protocolo puede ser responsable de implementar una estructura de datos básica de un protocolo no intrusivo. La estructura de datos básica es un formato estándar de un protocolo no intrusivo, que incluye clases de un bloque de identificación y un bloque lógico, una estructura de datos de una clave estándar del bloque de identificación y una estructura de datos de una clave estándar del bloque lógico. La especificación del protocolo puede ser responsable de proporcionar especificaciones lógicas y de datos específicos para un protocolo no intrusivo. La especificación del protocolo puede proporcionar además una función de verificación de normalización para un archivo de descripción entregado por una aplicación en la capa de aplicación. La normalización del archivo de descripción se puede comprobar en función de la infraestructura del protocolo y la especificación del protocolo. Por ejemplo, se puede comprobar un número de versión del protocolo, un identificador de cliente, una firma y una función en el archivo de descripción para comprobar la normalización del archivo de descripción. Por ejemplo, si el número de versión del protocolo, el identificador de cliente, la firma y la función del archivo de descripción no están vacíos, se determina que la comprobación del archivo de descripción se ha realizado correctamente. Cuando la función está marcada, se puede eliminar un espacio no válido en los datos correspondientes, se puede ignorar un caso de los datos y se pueden convertir todos los datos para que estén todos en minúsculas.

La infraestructura de componentes define un elemento básico de un componente en la capa de servicio del componente, como un mecanismo de acción y metadatos. El mecanismo de acción y metadatos se describe en la siguiente descripción del componente.

5 El módulo de análisis/ejecución de datos puede configurarse para analizar/comprobar un archivo de descripción de una aplicación en la capa de aplicación en función de la especificación del protocolo y la infraestructura del protocolo. El módulo de análisis/ejecución de datos puede configurarse además para iniciar la lógica de ejecución correspondiente en función de la especificación del protocolo y una configuración en serie o en paralelo de entidades lógicas en el archivo de descripción. Si las entidades lógicas del archivo de descripción están configuradas en serie, se ejecuta la lógica en serie, es decir, los componentes se ejecutan en serie. Si las entidades lógicas del archivo de descripción se configuran en paralelo, se ejecuta la lógica paralela, es decir, los componentes se ejecutan en paralelo. El módulo de análisis/ejecución de datos está conectado al módulo de enrutamiento de componentes y al módulo de almacenamiento de datos. Por ejemplo, después de analizar el archivo de descripción, el módulo de análisis/ejecución de datos puede transmitir un resultado de análisis correspondiente al módulo de enrutamiento del componente y/o al módulo de almacenamiento de datos.

20 El módulo de enrutamiento de componentes está configurado para implementar una función de búsqueda de un componente correspondiente entre los componentes de la capa de servicio del componente en función del archivo de descripción de la aplicación en la aplicación de la capa de aplicación. Por ejemplo, el módulo de enrutamiento de componentes puede buscar un componente correspondiente en función del resultado del análisis del archivo de descripción del módulo de análisis/ejecución de datos, e indicar el componente para proporcionar un servicio de capacidad de dispositivo correspondiente para la aplicación. El módulo de enrutamiento de componentes puede proporcionar además, para un componente, una función de consulta de datos definidos por el componente (por ejemplo, datos incluidos en el área del cuerpo).

25 El módulo de almacenamiento de datos es responsable de almacenar el resultado del análisis del archivo de descripción.

30 El módulo de interconexión de bus remoto puede conectarse a un módulo de bus en una capa de bus de otro dispositivo mediante el uso de una red. El módulo de interconexión de bus remoto puede configurarse además para transmitir un archivo de descripción desde una aplicación en la capa de aplicación. Por ejemplo, el módulo de interconexión de bus remoto puede transmitir el archivo de descripción desde la aplicación en la capa de aplicación a un módulo de bus de otro dispositivo que tenga un componente correspondiente.

35 A continuación se describen los componentes de la capa de servicio de componentes.

40 Como se describió anteriormente, un componente es un módulo que proporciona un servicio de capacidad del dispositivo y puede implementar una función independiente. El componente tiene al menos las tres características siguientes:

45 Característica 1: El componente puede analizar los datos definidos por el componente proporcionados por un protocolo no intrusivo.

Característica 2: El componente tiene información de configuración independiente. Por ejemplo, en una entidad lógica que contiene el componente, la información de configuración del componente se puede agregar en el siguiente formato (por ejemplo, el formato puede denominarse mecanismo de acción y metadatos):

```
50 <action name="com.support.component"/>
    <meta-data name="com.support.component1" value="function1"/>
    <meta-data name="com.support.component1" value="function1"/>
```

55 En la información de configuración, la primera línea se usa para indicar que los siguientes componentes son componentes que admiten el protocolo no intrusivo. La segunda y la tercera líneas indican, respectivamente, las funciones implementadas por diferentes componentes.

60 Por ejemplo, el dispositivo electrónico puede proporcionar un servicio de reproducción de vídeo y un servicio de carga de vídeo para la aplicación. En este caso, en las entidades lógicas que llevan un componente de reproducción de vídeo correspondiente y un componente de carga de vídeo correspondiente, la información de configuración del componente de reproducción de vídeo correspondiente y el componente de carga de vídeo correspondiente se puede añadir en función del siguiente formato:

```
65 <action name="com.support.component"/>
```

```
<meta-data name="com.support.component1" value="videoPlay" />
```

```
<meta-data name="com.support.component2" value="videoUpload" />
```

5 En la información de configuración, la segunda línea se usa para indicar que una función implementada por el componente es la reproducción de vídeo, es decir, el servicio de reproducción de vídeo puede proporcionarse a la aplicación. La tercera línea se usa para indicar que una función implementada por el componente es la carga de vídeo, es decir, se puede proporcionar el servicio de carga de vídeo para la aplicación.

10 Característica 3: El componente incluye una interfaz para implementar la interacción con el módulo de bus. En algunas implementaciones, se puede usar una interfaz PostData universal para la implementación. Por ejemplo, la interfaz es PostData (json easygo).

15 Basándose en la arquitectura de software proporcionada en la figura 2, la aplicación en la capa de aplicación en el dispositivo electrónico utiliza la forma de interacción no intrusiva, es decir, entrega el archivo de descripción utilizado para indicar la función que debe implementar la aplicación al módulo de bus de la capa de bus, de modo que el módulo de bus pueda encontrar, entre los componentes de la capa de servicio del componente en función del archivo de descripción, el componente correspondiente utilizado para la función correspondiente. Tras iniciarse, el componente encontrado puede proporcionar el servicio de capacidad del dispositivo correspondiente para la aplicación, a fin de
20 implementar la función correspondiente.

En esta implementación, la manera de interacción no intrusiva basada en la arquitectura de software anterior puede clasificarse en interacción estática e interacción en tiempo de ejecución.

25 La interacción estática puede significar que, cuando la aplicación no está en ejecución, o antes de que la aplicación necesite usar un servicio proporcionado por el componente, la aplicación completa la interacción con el módulo de bus en la capa de bus para crear un archivo de descripción que se utiliza para indicar algunas funciones que la aplicación debe implementar. Cuando es necesario implementar una función correspondiente en un tiempo de ejecución de la aplicación, un componente correspondiente puede proporcionar directamente un servicio de capacidad del dispositivo correspondiente para la aplicación en función de un resultado de interacción completado de antemano, para
30 implementar la función correspondiente. Un modo de interacción estática es aplicable a un componente que se inicia automáticamente. El componente que se inicia automáticamente es un componente que se inicia automáticamente cuando se enciende el dispositivo electrónico, por ejemplo, un componente que proporciona un servicio de adaptación de la interfaz. Los datos necesarios cuando el componente que se inicia automáticamente proporciona un servicio correspondiente no varían según el entorno en ejecución. El tiempo de inactividad de la aplicación puede incluir un período tal como un período de instalación de la aplicación.

La interacción en tiempo de ejecución puede significar que, en un tiempo de ejecución de la aplicación, la aplicación completa la interacción con el módulo de bus en la capa de bus para un archivo de descripción utilizado para indicar
40 algunas funciones que la aplicación debe implementar, y proporciona un servicio de capacidad de dispositivo correspondiente para la aplicación en función del resultado de la interacción, para implementar una función correspondiente. Una forma de interacción en tiempo de ejecución es aplicable a un componente cuyos datos utilizados al proporcionar un servicio correspondiente cambian en cualquier momento del tiempo de ejecución de la aplicación. Por lo tanto, cuando la aplicación necesita el componente para proporcionar un servicio, la aplicación realiza una interacción de datos con el módulo de bus en la capa de bus.

A continuación se describen por separado la interacción estática y la interacción en tiempo de ejecución.

50 Se utiliza un ejemplo en el que el tiempo de inactividad de la aplicación es el período de instalación de la aplicación. Con referencia a la figura 2, haciendo referencia a la figura 3, un proceso de interacción estática es el siguiente: cuando la aplicación está instalada, un instalador de aplicaciones lee el archivo de descripción de la aplicación (es decir, realiza la etapa 1 en la figura 3) e invoca la interfaz de capa de aplicación universal del módulo de bus en la capa de bus, tal como la interfaz PostData, para transmitir el archivo de descripción de lectura al módulo de bus (es decir, realiza la etapa 2 en la figura 2). 3). El instalador de la aplicación es un módulo de software responsable de la instalación de la
55 aplicación. El instalador de la aplicación puede ser responsable de analizar la configuración de una aplicación, descomprimir un paquete de aplicaciones y solidificar un archivo de aplicación en un directorio de instalación de aplicaciones. El desarrollador de la aplicación configura el archivo de descripción en función de una función que debe implementar la aplicación y lo empaqueta en un paquete de instalación de la aplicación. La función puede incluir una función implementada invocando un servicio de capacidad del dispositivo proporcionado por un componente iniciado automáticamente.

Tras recibir el archivo de descripción, la interfaz PostData del módulo de bus puede transmitir el archivo de descripción al módulo de análisis/ejecución de datos (es decir, realizar la etapa 3 en la figura 3). El módulo de análisis/ejecución de datos analiza el archivo de descripción en función de la especificación del protocolo y la infraestructura del
65 protocolo, y comprueba la normalización del archivo de descripción. Tras comprobar que el archivo de descripción está normalizado, el módulo de análisis/ejecución de datos transmite los datos (por ejemplo, incluidos los valores

clave/valor estándar en un bloque de identificación y un bloque lógico en el archivo de descripción, e incluye además los datos relacionados con los componentes, como los datos de un área corporal en el archivo de descripción) obtenidos mediante el análisis al módulo de almacenamiento de datos (es decir, realiza la etapa 4 en la figura 3). El módulo de almacenamiento de datos almacena los datos recibidos. Debe tenerse en cuenta que el módulo de análisis/ejecución de datos no analiza los datos en el área del cuerpo en el archivo de descripción, es decir, los datos definidos por los componentes, sino que transmite directamente los datos al módulo de almacenamiento de datos para su almacenamiento. En este caso, se completa la interacción entre la aplicación y el módulo de bus.

Si es necesario implementar una función correspondiente en tiempo de ejecución de la aplicación, se ejecuta el componente iniciado automáticamente. Cuando se ejecuta el componente iniciado automáticamente, el componente iniciado automáticamente puede realizar una consulta en tiempo real en el módulo de almacenamiento de datos del módulo de bus mediante el uso de una interfaz como PostData (json easygo) entre el componente iniciado automáticamente y el módulo de bus, para obtener datos definidos por el componente correspondientes al componente iniciado automáticamente. Cuando se ejecuta el servicio de capacidad del dispositivo proporcionado por el componente, el componente iniciado automáticamente puede analizar los datos definidos por el componente obtenidos y proporcionar el servicio de capacidad del dispositivo correspondiente para la aplicación, a fin de implementar la función correspondiente.

Por ejemplo, en algunas pantallas especiales, como una pantalla de hipersuperficie, si un elemento mostrado incluido en la interfaz de una aplicación está muy cerca de un borde de la pantalla, puede producirse una distorsión de la pantalla o puede producirse un fenómeno de contacto accidental por parte del usuario. En tal escenario, el desarrollador de la aplicación puede configurar un archivo de descripción en la aplicación (por ejemplo, Toutiao), y el archivo de descripción incluye información que indica que la aplicación necesita implementar una función de adaptación de márgenes. Con referencia a la figura 3, la función de adaptación de márgenes puede implementarse utilizando un servicio de adaptación de hipersuperficie proporcionado por un componente A. De esta manera, cuando se instala la aplicación, el archivo de descripción puede transmitirse al módulo de bus en la capa de bus, para completar la interacción de datos. A continuación, cuando se ejecuta la aplicación, el componente A se ejecuta y consulta los datos en el módulo de almacenamiento de datos del módulo de bus, para obtener los datos definidos por el componente correspondientes al componente A (por ejemplo, realiza la etapa 5-1 en la figura 3). Al ejecutar el servicio de adaptación de hipersuperficies proporcionado por el componente A, el componente A puede analizar los datos definidos por el componente obtenidos y ajustar automáticamente un margen en función de la interfaz actual de la aplicación, para evitar la distorsión de la visualización de un elemento mostrado cerca de un borde de la interfaz actual de la aplicación y, además, evitar que el usuario lo toque accidentalmente.

Para otro ejemplo, en algunas pantallas de gran tamaño, si las páginas de la aplicación se pueden mostrar una al lado de la otra, se puede mejorar la navegación y la eficiencia operativa del usuario. En tal escenario, el desarrollador de la aplicación puede configurar un archivo de descripción en la aplicación (por ejemplo, una aplicación de compras como Amazon), y el archivo de descripción incluye información que indica que la aplicación necesita implementar una función de visualización de página en paralelo (por ejemplo, la página de inicio de una tienda y una página de detalles del producto se muestran una al lado de la otra, o una página de detalles del producto y una página de servicio al cliente se muestran una al lado de la otra) e información sobre las páginas adecuadas para mostrarse una al lado de la otra. Con referencia a la figura 3, la función de visualización de páginas una al lado de la otra puede implementarse utilizando un servicio de adaptación a pantalla grande proporcionado por un componente B. De esta manera, cuando se instala la aplicación, el archivo de descripción puede transmitirse al módulo de bus en la capa de bus, para completar la interacción de datos. A continuación, cuando se ejecuta la aplicación, el componente B se ejecuta y consulta los datos en el módulo de almacenamiento de datos del módulo de bus, para obtener los datos definidos por el componente correspondientes al componente B, como la información sobre las páginas adecuadas para mostrarse una al lado de la otra (por ejemplo, realiza la etapa 5-2 en la figura 3). Cuando se ejecuta el servicio de adaptación a pantalla grande proporcionado por el componente B, el componente B puede analizar los datos definidos por el componente obtenidos y mostrar las páginas correspondientes de la aplicación una al lado de la otra en función de una interfaz actual de la aplicación, para mejorar la navegación y la eficiencia operativa del usuario.

Con referencia a la figura 2, haciendo referencia a la figura 4, un proceso de interacción de tiempo de ejecución es el siguiente:

Cuando se ejecuta la aplicación, la aplicación genera un archivo de descripción correspondiente en función de una función que se va a implementar e invoca la interfaz de capa de aplicación universal del módulo de bus en la capa de bus, como la interfaz PostData, para transmitir el archivo de descripción generado al módulo de bus. Por ejemplo, cuando el usuario pulsa un menú en una aplicación 1 para activar la aplicación para implementar una función de reserva de entradas de cine, la aplicación 1 puede generar un archivo de descripción correspondiente en función de la operación de toque realizada por el usuario en el menú, donde el archivo de descripción se usa para indicar que una función que debe implementar la aplicación 1 es la función de reserva de entradas de cine; y transmitir el archivo de descripción a la interfaz de capa de aplicación universal del módulo de bus, tal como la interfaz PostData (es decir, realizar la etapa 1-1 en la figura 4). Por ejemplo, el archivo de descripción incluye un campo "function": "movieTicket". Para otro ejemplo, cuando el usuario realiza el desbloqueo facial, una aplicación correspondiente del dispositivo electrónico, por ejemplo, denominada aplicación 2, puede generar un archivo de descripción correspondiente, donde

el archivo de descripción se usa para indicar que una función que debe implementar la aplicación 2 es una función de reconocimiento facial; y transmitir el archivo de descripción a la interfaz de capa de aplicación universal del módulo de bus, tal como la interfaz PostData (es decir, realizar la etapa 1-2 en la figura 4).

5 Tras recibir el archivo de descripción, la interfaz de capa de aplicación universal del módulo de bus, tal como la interfaz PostData, puede transmitir el archivo de descripción al módulo de análisis/ejecución de datos (es decir, realizar la etapa 2 en la figura 4). El módulo de análisis/ejecución de datos analiza el archivo de descripción en función de la especificación del protocolo y la infraestructura del protocolo, y comprueba la normalización del archivo de descripción. Tras comprobar que el archivo de descripción está normalizado, el módulo de análisis/ejecución de datos transmite los datos (por ejemplo, incluidos los valores clave/valor estándar en un bloque de identificación y un bloque lógico en el archivo de descripción, e incluye además los datos relacionados con los componentes, como los datos de un área corporal en el archivo de descripción) obtenidos mediante el análisis al módulo de almacenamiento de datos (es decir, realiza la etapa 3 en la figura 4). El módulo de almacenamiento de datos almacena los datos recibidos. Debe tenerse en cuenta que el módulo de análisis/ejecución de datos no analiza los datos en el área del cuerpo en el archivo de descripción, es decir, los datos definidos por los componentes, sino que transmite directamente los datos al módulo de almacenamiento de datos para su almacenamiento.

Tras comprobar que el archivo de descripción está normalizado, el módulo de análisis/ejecución de datos puede transmitir además datos (por ejemplo, datos relacionados con los componentes, tales como datos en un área principal incluida en el bloque lógico del archivo de descripción) obtenidos mediante el análisis al módulo de enrutamiento de componentes, para hacer que el módulo de enrutamiento de componentes realice el enrutamiento (es decir, realizar la etapa 4 en la figura 4). Cuando se enciende el dispositivo electrónico, el módulo de bus se inicia automáticamente y escanea todos los componentes del dispositivo electrónico en función de la infraestructura de los componentes; y puede escanear además todos los componentes de otro dispositivo si el dispositivo electrónico establece una conexión con el otro dispositivo, y almacenar la información de configuración de todos los componentes escaneados en el módulo de almacenamiento de datos. Además, en el dispositivo electrónico, si hay un componente que no está configurado, por ejemplo, un componente que proporciona un servicio de sistema subyacente generalmente no está configurado, el componente puede transmitir automáticamente la información correspondiente (por ejemplo, también denominada información de configuración) del componente al módulo de bus cuando el dispositivo está encendido, para el enrutamiento de los componentes.

Al realizar el enrutamiento de componentes, el módulo de enrutamiento de componentes puede buscar un componente que pueda proporcionar un servicio de capacidad de dispositivo correspondiente a partir de los componentes de la capa de servicio de componentes en función de la información de configuración almacenada en el módulo de almacenamiento de datos. En algunas implementaciones, si el dispositivo electrónico establece una conexión con otro dispositivo, al realizar el enrutamiento de componentes, el módulo de enrutamiento de componentes puede buscar primero un componente que pueda proporcionar el servicio de capacidad del dispositivo correspondiente a partir de los componentes del dispositivo local. Si hay un componente que puede proporcionar el servicio de capacidad del dispositivo correspondiente en los componentes del dispositivo local, el módulo de enrutamiento del componente indica que el componente debe iniciarse. Por ejemplo, con referencia al ejemplo anterior, el archivo de descripción se usa para indicar que la función que debe implementar la aplicación es la función de reserva de entradas de cine. El módulo de enrutamiento de componentes encuentra un componente que puede proporcionar un servicio de por vida, por ejemplo, denominado componente 1, a partir de los componentes del dispositivo local, y el componente 1 puede usarse para implementar la función de reserva de entradas de cine. En este caso, el módulo de enrutamiento de componentes indica que el componente 1 se inicie (es decir, realiza la etapa 5-1 en la figura 4). Como otro ejemplo, con referencia al ejemplo anterior, el archivo de descripción se usa para indicar que la función que debe implementar la aplicación es la función de reconocimiento facial. El módulo de enrutamiento de componentes encuentra un componente que puede proporcionar un servicio de inteligencia artificial (inteligencia artificial, IA), por ejemplo, denominado componente 2, a partir de los componentes del dispositivo local, y el componente 2 puede usarse para implementar la función de reconocimiento facial. En este caso, el módulo de enrutamiento de componentes indica que el componente 2 se inicie (es decir, realiza la etapa 5-2 en la figura 4).

Si no hay ningún componente que pueda proporcionar el servicio de capacidad del dispositivo correspondiente en los componentes del dispositivo local, el módulo de enrutamiento del componente transmite, utilizando el módulo de interconexión de bus remoto y en función de la información de configuración almacenada en el módulo de almacenamiento de datos, el archivo de descripción recibido a otro dispositivo conectado al dispositivo (por ejemplo, transmite el archivo de descripción recibido a un módulo de interconexión de bus remoto en un módulo de bus del otro dispositivo), es decir, realiza la etapa 6 en la figura 4). El dispositivo incluye un componente que puede proporcionar el servicio de capacidad del dispositivo correspondiente. Una vez que el archivo de descripción se transmite al otro dispositivo, el dispositivo analiza y comprueba de nuevo el archivo de descripción, activa el módulo de enrutamiento del componente para que realice el enrutamiento hasta que se encuentre un componente que pueda proporcionar el servicio de capacidad del dispositivo correspondiente e indica el componente que debe iniciarse.

Cuando se ejecuta el componente, el componente puede realizar una consulta en tiempo real en el módulo de almacenamiento de datos del módulo de bus mediante el uso de una interfaz como PostData (json easygo) entre el componente y el módulo de bus, para obtener los datos definidos por el componente correspondientes al componente.

5 Cuando se ejecuta el servicio de capacidad del dispositivo proporcionado por el componente, el componente puede analizar los datos definidos por el componente obtenidos y proporcionar el servicio de capacidad del dispositivo correspondiente para la aplicación, a fin de implementar la función correspondiente. Por ejemplo, con referencia al ejemplo anterior, cuando se ejecuta, el componente 1 consulta el módulo de almacenamiento de datos para obtener los datos definidos por el componente correspondientes al componente 1 (es decir, realiza la etapa 7-1 en la figura 4). Para otro ejemplo, con referencia al ejemplo anterior, cuando se ejecuta, el componente 2 consulta el módulo de almacenamiento de datos para obtener los datos definidos por el componente correspondientes al componente 2 (es decir, realiza la etapa 7-2 en la figura 4).

10 A continuación se utiliza un ejemplo en el que el dispositivo electrónico es un teléfono móvil para describir esta implementación en función de diferentes maneras de interacción (interacción estática o interacción en tiempo de ejecución) con referencia a los ejemplos.

15 Ejemplo 1: Con referencia a la figura 2, una capa de servicio de componente del teléfono móvil incluye un componente de pantalla dividida. El componente de pantalla dividida puede proporcionar un servicio de pantalla dividida para implementar una función de visualización de pantalla dividida. Específicamente, en un proceso en el que un usuario navega por una aplicación, el componente de pantalla dividida puede mostrar páginas de la aplicación en pantallas divididas en función de una función de actividad (Actividad) de la aplicación. El componente de pantalla dividida puede pertenecer al componente iniciado automáticamente anterior. Se puede usar el modo de interacción estática. Cuando la aplicación no está en ejecución, la aplicación completa la interacción con un módulo de bus en una capa de bus. Por ejemplo, para una aplicación que necesita usar el servicio de pantalla dividida proporcionado por el componente de pantalla dividida, un desarrollador de aplicaciones puede configurar un archivo de descripción correspondiente y empaquetar el archivo de descripción en un paquete de instalación de la aplicación. Cuando se instala la aplicación, el archivo de descripción se transmite al módulo de bus en la capa de bus. El módulo de bus analiza y comprueba el archivo de descripción y, tras comprobar que el archivo de descripción está normalizado, almacena los datos obtenidos mediante el análisis, por ejemplo, incluidos los valores de clave/valor estándar en un bloque de identificación y un bloque lógico en el archivo de descripción, e incluye además los datos relacionados con los componentes (como los datos de un área corporal en el archivo de descripción). De esta manera, en tiempo de ejecución de la aplicación, si es necesario implementar una función de pantalla dividida, el componente de pantalla dividida se ejecuta y lee y analiza los datos almacenados para proporcionar el servicio de pantalla dividida para la aplicación, mostrando así las páginas de la aplicación en pantallas divididas.

35 En algunas implementaciones, el servicio de pantalla dividida proporcionado por el componente de pantalla dividida incluye dos modos de visualización de pantalla dividida: un modo de lista y un modo de no lista. El modo de lista es aplicable a una aplicación de flujo de información, tal como una aplicación de noticias. El modo sin lista es aplicable a un escenario en el que un solo producto se muestra en una pluralidad de dimensiones o en un modo asociado en una aplicación, y a un escenario en el que dos productos se comparan en pantallas divididas, como una aplicación de compras.

40 El modo de lista puede incluir dos modos: un modo de lista única y un modo de lista múltiple. Por ejemplo, en el modo de lista única, la visualización en pantalla dividida de las páginas de una aplicación puede ser: La pantalla del teléfono móvil se divide en un área de visualización izquierda y un área de visualización derecha, siempre se muestra una página de inicio de la aplicación en el área de visualización izquierda y se puede mostrar otra página de la aplicación en el área de visualización derecha. Por ejemplo, la aplicación es Toutiao. Cuando Toutiao está instalado, se puede transmitir un archivo de descripción correspondiente al módulo de bus, y el módulo de bus analiza y comprueba el archivo de descripción. Tras comprobar que el archivo de descripción está normalizado, el módulo de bus almacena los datos obtenidos mediante el análisis. Como se muestra en la figura 5(a), después de que el usuario abra Toutiao, el teléfono móvil muestra una página de inicio 501 de Toutiao. Cuando el usuario pulsa una opción de la página de inicio para activar la visualización de otra página, se ejecuta el componente de pantalla dividida y lee y analiza los datos almacenados para proporcionar el servicio de pantalla dividida para Toutiao, mostrando así las páginas de Toutiao en pantallas divididas. Como se muestra en la figura 5(b), el teléfono móvil muestra la página de inicio 501 de Toutiao en un área izquierda de la pantalla y muestra otra página 502 en un área derecha de la pantalla. Este modo de visualización puede ayudar al usuario a navegar rápidamente. Además, el componente de pantalla dividida puede proporcionar además un servicio para salir de la visualización en pantalla dividida. Cuando el usuario desea salir de la visualización en pantalla dividida, puede realizar un gesto correspondiente para provocar la salida de la visualización en pantalla dividida, de modo que se pueda resolver la incomodidad de la lectura a media pantalla durante la lectura de contenido prolongado. El usuario puede volver a activar la visualización en pantalla dividida del teléfono móvil realizando un teléfono móvil correspondiente.

60 Para otro ejemplo, si una aplicación tiene una pluralidad de páginas de lista, tales como una página de inicio, una página de búsqueda y una página de recomendaciones, las actividades que contienen estas páginas de lista son todas de un nivel 0, diferentes actividades llevan a cabo diferentes páginas de lista y las actividades que contienen páginas que no son de lista son todas de un nivel 1. Para una aplicación de este tipo, el modo de lista múltiple se puede usar para la visualización en pantalla dividida. Con referencia a la figura 6, la visualización en pantalla dividida de las páginas de una aplicación de este tipo puede ser: Una vez iniciada la aplicación, como se muestra en la figura 6(a), el teléfono móvil muestra una página de inicio de la aplicación en pantalla completa. Si el usuario pulsa una opción en la

página de inicio para activar la visualización de una página que no es de la lista, como se muestra en la figura 6(b), el teléfono móvil divide la pantalla en un área de visualización izquierda y un área de visualización derecha, la página de inicio se muestra en el área izquierda y la página que no es de la lista se muestra en el área derecha. Si el usuario pulsa una opción en la página de inicio para activar la visualización de otra página de lista, como una página de recomendaciones, como se muestra en la figura 6(c), el teléfono móvil muestra la página de recomendaciones en pantalla completa. Si el usuario pulsa una opción en la página de inicio en la visualización de pantalla dividida que se muestra en la figura 6(b) para activar la visualización de otra página de lista, tal como una página de búsqueda, como se muestra en la figura 6(d), el teléfono móvil muestra la página de inicio en el área izquierda y muestra la página de búsqueda en el área derecha. Además, el componente de pantalla dividida puede proporcionar además un servicio para salir de la visualización en pantalla dividida. En los escenarios de visualización en pantalla dividida mostrados en la figura 6(c) y la figura 6(d), cuando el usuario desea salir de la visualización en pantalla dividida, el usuario puede realizar un gesto correspondiente para activar la salida de la visualización en pantalla dividida. El usuario puede volver a activar la visualización en pantalla dividida del teléfono móvil realizando un teléfono móvil correspondiente. Este modo de visualización puede evitar el cambio frecuente entre la visualización en pantalla dividida y la visualización en pantalla completa.

Para una aplicación a la que es aplicable la visualización en pantalla dividida en el modo sin lista, la visualización en pantalla dividida se puede realizar en función de una regla de visualización simultánea de las actividades relacionadas. Se usa una aplicación de compras como ejemplo. Las actividades relacionadas pueden incluir una actividad con una página de detalles de un producto básico, una actividad con una página de servicio al cliente del producto, una actividad con una página de comentarios del producto básico y una actividad con una página de detalles de un producto similar al producto básico.

Por ejemplo, haciendo referencia a la figura 7, la visualización en pantalla dividida de las páginas de la aplicación de compras puede ser: Una vez iniciada la aplicación, el teléfono móvil puede mostrar una página de inicio de la aplicación en pantalla completa. Si el usuario pulsa una opción en la página de inicio para activar la visualización de una página de detalles de un producto, por ejemplo, denominada página de detalles 1, el teléfono móvil puede mostrar la página de detalles 1 en pantalla completa. Si el usuario pulsa una opción en la página de detalles 1 para activar la visualización de la página de inicio de una tienda, el teléfono móvil muestra la página de inicio de la tienda en pantalla completa. Si el usuario pulsa una opción en la página de detalles 1 para activar la visualización de una página de detalles de un producto relacionado con el producto, por ejemplo, denominada página de detalles 2, el teléfono móvil divide la pantalla en un área de visualización izquierda y un área de visualización derecha, la página de detalles 1 se muestra en el área izquierda y la página de detalles 2 se muestra en el área derecha. Si el usuario pulsa una opción en la página de detalles 1 para activar la visualización de una página de servicio al cliente del producto, por ejemplo, denominada página de servicio al cliente 1, el teléfono móvil divide la pantalla en un área de visualización izquierda y un área de visualización derecha, la página de detalles 1 se muestra en el área izquierda y la página de servicio al cliente 1 se muestra en el área derecha.

Cuando el teléfono móvil muestra la página de detalles 1 en el área izquierda y muestra la página de detalles 2 en el área derecha, si el usuario pulsa una opción en la página de detalles 1 para activar la visualización de una página de detalles de un producto relacionado con el producto, por ejemplo, denominada página de detalles 3, la página de detalles 1 continúa mostrándose en el área izquierda y la página de detalles 3 se muestra en el área derecha. Si el usuario pulsa una opción en la página de detalles 1 para activar la visualización de una página de servicio al cliente del producto, por ejemplo, denominada página de servicio al cliente 1, la página de detalles 1 continúa mostrándose en el área izquierda y la página de servicio al cliente 1 se muestra en el área derecha. Si el usuario pulsa una opción en la página de detalles 2 para que aparezca una página de detalles de un producto relacionado con el producto, por ejemplo, denominada página de detalles 3, la página de detalles 2 se muestra en el área izquierda y la página de detalles 3 se muestra en el área derecha. Si el usuario pulsa una opción en la página de detalles 2 para activar la visualización de una página de servicio al cliente del producto, por ejemplo, denominada página de servicio al cliente 2, la página de detalles 2 se muestra en el área izquierda y la página de servicio al cliente 2 se muestra en el área derecha.

Cuando el teléfono móvil muestra la página de detalles 1 en el área izquierda y muestra la página de servicio al cliente 1 en el área derecha, si el usuario pulsa una opción en la página de detalles 1 para activar la visualización de una página de detalles de un producto relacionado con el producto, por ejemplo, denominada página de detalles 2, la página de detalles 1 continúa mostrándose en el área izquierda y la página de detalles 2 se muestra en el área derecha. Si el usuario pulsa una opción en la página de detalles 1 para que se muestre una página de comentarios del producto, la página de detalles 1 continúa mostrándose en el área izquierda y la página de comentarios se muestra en el área derecha.

En este ejemplo, a continuación se proporciona un ejemplo de un archivo de descripción:

```
<?xml version="1.0" encoding="utr-8"?>
```

```
<hwEasyGo version="1.0">
```

```

<client package="com.xxx.xxx"/> <!--Identification block-package name-->
<signInfo value="xxx"/> <!--Identification block-signature, to ensure data integrity-->
5 <logicEntities> <!--Logical block-->
<entity> <!--Logical block-entity 1-->
    <head> <!--Logical block-entity 1-head area, which is a universal field-- >
10     <function name="magicWindowAdapter"/>
    </head>
15     <body> <!--Logical block-entity 1-body area,
    corresponding to self-defined data of a function-->
        <windowMode value="2"> <!--Self-defined field 1:
20 2 indicates that a screen split is
        supported, and 3 indicates that single-screen displaying is always used-->
        <switchType value="1"> <!--Self-defined field 2:
25 0 indicates the list mode, 1 indicates the non-
        list mode, and 2 indicates a universal switching mode-->
30 <activityLevel> <!--Self-defined field 3: sets an activity level-->
        <activity level="0" name="com.xxx.xxx.MainActivity">
35         <activity level="0" name="com.xxx.xxx.SearchActivity">
        <activity level="0" name="com.xxx.xxx.DetailActivity">
        <activity level="1" name="com.xxx.xxx.ChatActivity">
40     </activityLevel>
    </body>
45 </entity>
</logicEntities>
</hwEasyGo>
50

```

Cuando se instala la aplicación, el módulo de bus puede obtener el archivo de descripción y analizar y comprobar el archivo de descripción. Tras comprobar que el archivo de descripción está normalizado, el módulo de bus almacena los datos obtenidos mediante el análisis sintáctico, por ejemplo, incluyendo los datos <windowMode value="2"> indicando que está soportada una división de pantalla, datos <switchType value="1"> indicando un modo de visualización de pantalla dividida, y datos para establecer un nivel de actividad. En tiempo de ejecución de la aplicación, si es necesario implementar la función de pantalla dividida, los datos almacenados se leen y analizan para proporcionar el servicio de pantalla dividida para la aplicación, mostrando así las páginas de la aplicación en pantallas divididas.

Ejemplo 2: Con referencia a la figura 2, las capas de servicio de componentes del teléfono móvil y de una caja de sonido inteligente incluyen cada una un componente de reproducción de audio, y se establece una conexión entre el teléfono móvil y la caja de sonido inteligente. El componente de reproducción de audio puede proporcionar un servicio de reproducción de audio para implementar una función de reproducción de audio. Por ejemplo, una función de componente de reproducción de audio del teléfono móvil es PhoneMusicPlay, y una función de componente de reproducción de audio de la caja de sonido inteligente es BoxMusicPlay. Cuando el usuario inicia un reproductor de música para reproducir música, se pretende implementar la siguiente lógica: Para reproducir una pieza musical, el servicio de reproducción de audio proporcionado por el componente de reproducción de audio se usa para implementar

la función de reproducción de audio, y se prefiere la caja de sonido inteligente. En este escenario, dado que el usuario activa la reproducción de música, se puede utilizar la forma de interacción de la interacción en tiempo de ejecución. Cuando se ejecuta el reproductor de música, la aplicación completa la interacción con un módulo de bus en una capa de bus.

5 Por ejemplo, el usuario abre el reproductor de música del teléfono móvil y pulsa una pieza musical que debe reproducirse. En respuesta a la operación del usuario, el reproductor de música puede generar el siguiente archivo de descripción y transmitir el archivo de descripción al módulo de bus en la capa de bus. El módulo de bus analiza y comprueba el archivo de descripción y, mediante el análisis, el módulo de bus descubre que se prefiere un componente de reproducción de audio BoxMusicPlay. El módulo de bus busca en el dispositivo local, es decir, el teléfono móvil, el componente de reproducción de audio BoxMusicPlay, pero no encuentra el componente de reproducción de audio BoxMusicPlay. El módulo de bus transmite el archivo de descripción a un módulo de bus de la caja de sonido inteligente. Mediante el análisis, el módulo de bus de la caja de sonido inteligente aprende que se prefiere el componente de reproducción de audio BoxMusicPlay y, además, puede almacenar datos en una zona del cuerpo. El módulo de bus de la caja de sonido inteligente encuentra el componente de reproducción de audio BoxMusicPlay en el dispositivo local e inicia el componente de reproducción de audio BoxMusicPlay. El componente de reproducción de audio BoxMusicPlay se ejecuta y lee y analiza los datos almacenados en la zona del cuerpo para proporcionar un servicio de reproducción de audio para que el reproductor de música reproduzca música.

```

20 {
    "EasyGo Version": "1.0", //Version number of Huawei fast access protocol
    "client": "com.xxx.music", //Identifier of an accessed third-party application
25 "signInfo": "ALSJFS32KAD90FJ2SAD", //Signature
    "logicEntities": //Bloque lógico: ejecución de entidades lógicas
30 "or": { //Condition keyword: "o" indica que se selecciona una entidad de entre una entidad establecida en
    secuencia para su ejecución; "all" indica que todas las entidades están ejecutadas
        {
35 "head": {
            "function": "boxMusicPlay" //Identificador único de un componente de función
            "required": "ture"
40 }
        "body": { //Logical block: un área del cuerpo, que es una clave autodefinida y que el componente
        analiza durante la ejecución
45 "musicPath": "http://xxx.yyy.mp4", //Ruta de la música reproducida
        }
50 },
        {
            "head": {
55 "function": "phoneMusicPlay"
            "required": "ture",
60 }
            "body": {
65 "musicPath": "http://xxx.yyy.mp4", //Ruta de la música reproducida
        }
    }
}

```

}

}

5

}

}

10 De acuerdo con algunas otras implementaciones de esta aplicación, se proporciona además un dispositivo electrónico, y está configurado para realizar el método en las implementaciones anteriores, para implementar las funciones del dispositivo electrónico en las implementaciones anteriores. El dispositivo electrónico puede incluir una pantalla, uno o más procesadores y una memoria. Los componentes anteriores pueden estar conectados a través de uno o más buses de comunicación. La memoria almacena uno o más fragmentos de código de programa de ordenador, uno o más
15 fragmentos de código de programa de ordenador incluyen instrucciones de ordenador, y uno o más procesadores están configurados para ejecutar las instrucciones de ordenador, y pueden configurarse para implementar funciones de comportamiento del dispositivo electrónico en las implementaciones anteriores.

20 De acuerdo con algunas otras implementaciones de esta aplicación, se proporciona además un medio de almacenamiento legible por ordenador. El medio de almacenamiento legible por ordenador puede incluir instrucciones de software informático. Cuando las instrucciones del software informático se ejecutan en un dispositivo electrónico, el dispositivo electrónico está habilitado para realizar las etapas realizadas por el dispositivo electrónico en las implementaciones anteriores.

25 De acuerdo con algunas otras implementaciones de esta solicitud, se proporciona además un producto de programa informático. Cuando el producto de programa de ordenador se ejecuta en un ordenador, el ordenador está habilitado para realizar los pasos realizados por el dispositivo electrónico en las implementaciones anteriores.

30 De acuerdo con algunas otras implementaciones de esta solicitud, se proporciona además un aparato. El aparato tiene la función de implementar el comportamiento del dispositivo electrónico en las implementaciones anteriores. La función pueden ser implementada por un hardware o puede ser implementada por un hardware que ejecuta el software correspondiente. El hardware o el software incluyen uno o más módulos correspondientes a la función anterior, por ejemplo, una unidad o módulo de obtención, una unidad o módulo de determinación y una unidad o módulo de ejecución.
35

De acuerdo con algunas otras implementaciones de esta aplicación, se proporciona además un sistema de chips. El sistema de chip se puede usar en un dispositivo electrónico. El sistema de chip incluye un circuito de interfaz y un procesador. El circuito de interfaz y el procesador están interconectados a través de una línea. El circuito de interfaz está configurado para recibir una señal desde una memoria del dispositivo electrónico y enviar la señal al procesador.
40 La señal incluye instrucciones de ordenador almacenadas en la memoria. Cuando el procesador ejecuta las instrucciones del ordenador, el sistema de chip realiza las etapas realizadas por el dispositivo electrónico en las implementaciones anteriores.

45 El dispositivo electrónico, el medio de almacenamiento legible por ordenador, el producto de programa informático, el aparato y el sistema de chips proporcionados en las implementaciones de esta aplicación pueden realizar acciones del dispositivo electrónico en las implementaciones anteriores. Los principios de implementación y los efectos técnicos de los mismos son similares, y los detalles no se describen de nuevo en el presente documento.

50 La descripción anterior sobre las implementaciones permite a un experto en la materia entender claramente que, a efectos de una descripción conveniente y breve, la división en sólo los módulos funcionales anteriores se utiliza como ejemplo para la descripción. Durante la aplicación real, las funciones anteriores pueden asignarse a diferentes módulos funcionales para su implementación según sea necesario. En otras palabras, la estructura interna de un aparato se divide en diferentes módulos funcionales para implementar todas o algunas de las funciones descritas anteriormente.

55 En las diversas implementaciones proporcionadas en esta solicitud, debe entenderse que el aparato y el método descritos pueden implementarse de otras maneras. Por ejemplo, las implementaciones de aparato descritas son simplemente ejemplos. Por ejemplo, la división en los módulos o unidades es simplemente una división en funciones lógicas y puede ser otra división durante la implementación propiamente dicha. Por ejemplo, una pluralidad de unidades o componentes pueden combinarse o integrarse en otro aparato, o algunas características pueden ignorarse
60 o no ejecutarse. Además, los acoplamientos mutuos mostrados o discutidos o los acoplamientos directos o las conexiones de comunicación pueden implementarse a través de algunas interfaces. Los acoplamientos indirectos o las conexiones de comunicaciones entre los aparatos o las unidades pueden implementarse de modo electrónico, de modo mecánico, o de otro mod.

65 Las unidades que se describen como componentes separados pueden o no estar físicamente separados, y los componentes que se muestran como unidades pueden ser una o más unidades físicas, que pueden ubicarse en un

lugar o pueden distribuirse en una pluralidad de lugares diferentes. Algunas o todas las unidades pueden seleccionarse en función de un requisito real para lograr un objetivo de las soluciones de las implementaciones.

5 Además, las unidades funcionales en las implementaciones de esta solicitud pueden integrarse dentro de una unidad de procesamiento, o cada una de las unidades pueden existir físicamente solas, o dos o más unidades se integran dentro de una unidad. La unidad integrada puede implementarse en forma de hardware, o puede implementarse en forma de una unidad funcional de software.

10 Cuando la unidad integrada se implementa en forma de una unidad funcional de software y se vende o usa como un producto independiente, la unidad integrada puede almacenarse en un medio de almacenamiento legible. En base a tal comprensión, las soluciones técnicas de las implementaciones de esta solicitud esencialmente, o la parte que contribuye a la tecnología actual, o todas o algunas de las soluciones técnicas pueden implementarse en forma de un producto de software. El producto de software se almacena en un medio de almacenamiento e incluye varias instrucciones para indicar a un dispositivo (que puede ser un microordenador de un solo chip, un chip, o similar) o un procesador (procesador) que realice todas o algunas de las etapas de los métodos en las implementaciones de esta solicitud. El medio de almacenamiento anterior incluye: cualquier medio que pueda almacenar código de programa, tal como una unidad flash USB, un disco duro extraíble, una memoria de solo lectura (Read-Only Memory, ROM), una memoria de acceso aleatorio (Random Access Memory, RAM), un disco magnético o un disco óptico.

20 El alcance de la protección está definido por las reivindicaciones adjuntas.

REIVINDICACIONES

1. Un método realizado por un dispositivo electrónico, comprendiendo el método:
- 5 obtener, mediante el dispositivo electrónico, un archivo de descripción de la aplicación, en donde el archivo de descripción se usa para indicar una función que debe implementar la aplicación, y el archivo de descripción se define según el protocolo de interacción no intrusiva, en donde el protocolo de interacción no intrusiva es JavaScript Object Notation, JSON;
- 10 determinar, mediante el dispositivo electrónico, un primer componente en función del archivo de descripción, en donde el primer componente es un componente que está en los componentes del dispositivo electrónico y que puede implementar la función que debe implementar la aplicación, y el componente del dispositivo electrónico es un módulo que está en el dispositivo electrónico y que está configurado en función de un protocolo de interacción no intrusivo para proporcionar un servicio de capacidad del dispositivo, y puede
- 15 implementar una función independiente; y
- ejecutar, mediante el dispositivo electrónico, el primer componente en función del archivo de descripción para proporcionar un servicio de capacidad del dispositivo para la aplicación, a fin de implementar la función que debe implementar la aplicación; en donde el archivo de descripción comprende un campo usado para
- 20 identificar un componente que puede implementar la función que necesita implementar la aplicación.
2. El método según la reivindicación 1, en donde el archivo de descripción comprende un bloque lógico, en donde el bloque lógico comprende una o más entidades lógicas individuales; se utiliza una única entidad lógica del bloque lógico para indicar una instancia en ejecución de un componente, y la única entidad lógica del bloque lógico comprende un área principal; el área principal comprende un campo utilizado para identificar el componente que puede implementar la función que debe implementar la aplicación; y
- 25 la determinación, por parte del dispositivo electrónico, de un primer componente en función del archivo de descripción comprende:
- 30 determinar, mediante el dispositivo electrónico, el primer componente en función del campo comprendido en el área principal de la entidad lógica única en el bloque lógico del archivo de descripción.
3. El método según la reivindicación 2, en donde el archivo de descripción comprende además un bloque de identificación, en donde el bloque de identificación comprende los siguientes campos: un número de versión del protocolo no intrusivo seguido del archivo de descripción, un identificador de la aplicación y una firma de la aplicación; y
- 35 la determinación, mediante el dispositivo electrónico, de un primer componente en función del archivo de descripción comprende además:
- 40 comprobar, mediante el dispositivo electrónico, la normalización del archivo de descripción en función de los campos incluidos en el bloque de identificación del archivo de descripción y el campo comprendido en el área principal; y
- 45 al determinar que la comprobación de normalización del archivo de descripción tiene éxito, determinar, mediante el dispositivo electrónico, el primer componente en función del campo comprendido en el área principal de la entidad lógica única en el bloque lógico del archivo de descripción.
- 50 4. El método según la reivindicación 3, en donde la entidad lógica única del bloque lógico comprende además un área corporal, y el área corporal comprende un campo usado para los datos requeridos por un componente para implementar una función; y
- 55 la ejecución, por parte del dispositivo electrónico, del primer componente basado en el archivo de descripción para proporcionar un servicio de capacidad del dispositivo para la aplicación, para implementar la función que debe implementar la aplicación, comprende:
- 60 ejecutar, mediante el dispositivo electrónico, el primer componente en función del campo comprendido en el área corporal de la entidad lógica única en el bloque lógico del archivo de descripción para proporcionar un servicio de capacidad del dispositivo para la aplicación, a fin de implementar la función que debe implementar la aplicación.
5. El método según una cualquiera de las reivindicaciones 1 a 4, en donde la obtención, mediante el dispositivo electrónico, un archivo de descripción de una aplicación comprende:
- 65

obtener, mediante el dispositivo electrónico, el archivo de descripción preconfigurado en un tiempo de inactividad de la aplicación;

el método comprende, además:

5 analizar, mediante el dispositivo electrónico, el archivo de descripción y almacenar los campos comprendidos en el bloque de identificación del archivo de descripción, el campo comprendido en el área principal de la entidad lógica única en el bloque lógico del archivo de descripción, y el campo comprendido en el área del cuerpo de la entidad lógica única en el bloque lógico del archivo de descripción; y

10 la ejecución, por parte del dispositivo electrónico, del primer componente basado en el archivo de descripción para proporcionar un servicio de capacidad del dispositivo para la aplicación, para implementar la función que debe implementar la aplicación, comprende:

15 leer, mediante el dispositivo electrónico en tiempo de ejecución de la aplicación, el campo almacenado comprendido en el área del cuerpo de la entidad lógica única en el bloque lógico del archivo de descripción; y

20 analizar, mediante el dispositivo electrónico, el campo comprendido en el área del cuerpo de la entidad lógica única en el bloque lógico del archivo de descripción y ejecutar el primer componente en función del resultado del análisis para proporcionar un servicio de capacidad de dispositivo para la aplicación, a fin de implementar la función que debe implementar la aplicación.

25 6. El método según una cualquiera de las reivindicaciones 1 a 4, en donde la obtención, mediante el dispositivo electrónico, un archivo de descripción de una aplicación comprende:

generar, mediante el dispositivo electrónico, el archivo de descripción cuando la aplicación se está ejecutando;

30 el método comprende, además:

35 analizar, mediante el dispositivo electrónico, el archivo de descripción y almacenar los campos comprendidos en el bloque de identificación del archivo de descripción, el campo comprendido en el área principal de la entidad lógica única en el bloque lógico del archivo de descripción, y el campo comprendido en el área del cuerpo de la entidad lógica única en el bloque lógico del archivo de descripción; y

40 la ejecución, por parte del dispositivo electrónico, del primer componente basado en el archivo de descripción para proporcionar un servicio de capacidad del dispositivo para la aplicación, para implementar la función que debe implementar la aplicación, comprende:

leer, mediante el dispositivo electrónico, el campo almacenado comprendido en el área corporal de la entidad lógica única en el bloque lógico del archivo de descripción; y

45 analizar, mediante el dispositivo electrónico, el campo comprendido en el área del cuerpo de la entidad lógica única en el bloque lógico del archivo de descripción y ejecutar el primer componente en función del resultado del análisis para proporcionar un servicio de capacidad de dispositivo para la aplicación, a fin de implementar la función que debe implementar la aplicación.

50 7. El método según una cualquiera de las reivindicaciones 1 a 6, en donde el dispositivo electrónico almacena información de configuración de todos los componentes del dispositivo electrónico; y

la determinación, por parte del dispositivo electrónico, de un primer componente en función del archivo de descripción comprende:

55 determinar, mediante el dispositivo electrónico, el primer componente en función del archivo de descripción y la información de configuración almacenada de todos los componentes.

60 8. El método según la reivindicación 7, en donde el dispositivo electrónico almacena además la información de configuración de todos los componentes de un dispositivo que establece una conexión inalámbrica con el dispositivo electrónico; y

el método comprende, además:

65 si el dispositivo electrónico no determina el primer componente y determina, en función de la información de configuración almacenada de los componentes, que hay un componente que puede implementar la función que debe implementar la aplicación en los componentes del dispositivo conectado al dispositivo electrónico,

transmitir, mediante el dispositivo electrónico, el archivo de descripción al dispositivo conectado al dispositivo electrónico, en donde el dispositivo conectado al dispositivo electrónico utiliza el archivo de descripción para determinar y ejecutar un componente correspondiente, para proporcionar un servicio de capacidad del dispositivo para la aplicación e implementar la función que debe implementar la aplicación.

5
9. Un dispositivo electrónico, en donde el dispositivo electrónico comprende un procesador (110) y una memoria (121); el procesador y la memoria están acoplados, la memoria está configurada para almacenar el código del programa informático, el código del programa informático comprende instrucciones de software informático, y cuando las instrucciones del software informático son ejecutadas por el dispositivo electrónico, el dispositivo electrónico está habilitado para realizar las siguientes operaciones:

10
obtener un archivo de descripción de la aplicación, en donde el archivo de descripción se usa para indicar una función que debe implementar la aplicación, y el archivo de descripción se define según el protocolo de interacción no intrusiva, en donde el protocolo de interacción no intrusiva es JavaScript Object Notation, JSON;

15
determinar un primer componente en función del archivo de descripción, en donde el primer componente es un componente que está en los componentes del dispositivo electrónico y que puede implementar la función que necesita implementar la aplicación, y el componente del dispositivo electrónico es un módulo que está en el dispositivo electrónico y que está configurado en función de un protocolo de interacción no intrusivo para proporcionar un servicio de capacidad del dispositivo, y puede implementar una función independiente; y

20
ejecutar el primer componente en función del archivo de descripción para proporcionar un servicio de capacidad de dispositivo para la aplicación, para implementar la función que debe implementar la aplicación;

25
en donde el archivo de descripción comprende un campo usado para identificar un componente que puede implementar la función que necesita implementar la aplicación.

30 10. El dispositivo electrónico según la reivindicación 9, en donde el archivo de descripción comprende un bloque lógico, en donde el bloque lógico comprende una o más entidades lógicas individuales; se utiliza una única entidad lógica del bloque lógico para indicar una instancia en ejecución de un componente, y la única entidad lógica del bloque lógico comprende un área principal; el área principal comprende un campo utilizado para identificar el componente que puede implementar la función que debe implementar la aplicación; y

35
la determinación de un primer componente en función del archivo de descripción comprende:

40
determinar el primer componente en función del campo comprendido en el área principal de la entidad lógica única en el bloque lógico del archivo de descripción.

45 11. El dispositivo electrónico según la reivindicación 10, en donde el archivo de descripción comprende un bloque de identificación,

en donde el bloque de identificación comprende los siguientes campos: un número de versión del protocolo de interacción no intrusivo seguido del archivo de descripción, un identificador de la aplicación y una firma de la aplicación; y

50
la determinación de un primer componente en función del archivo de descripción comprende además:

comprobar la normalización del archivo de descripción en función de los campos comprendidos en el bloque de identificación del archivo de descripción y el campo comprendido en el área principal; y al determinar que la comprobación de normalización del archivo de descripción tiene éxito, determinar el primer componente en función del campo comprendido en el área principal de la entidad lógica única en el bloque lógico del archivo de descripción.

55 12. El dispositivo electrónico según la reivindicación 11, en donde la entidad lógica única del bloque lógico comprende además un área corporal, y el área corporal comprende un campo utilizado para los datos requeridos por un componente para implementar una función; y

60
ejecutar el primer componente en función del archivo de descripción para proporcionar un servicio de capacidad del dispositivo para la aplicación, para implementar la función que necesita implementar la aplicación, comprende: ejecutar el primer componente en función del campo comprendido en el área corporal de la entidad lógica única en el bloque lógico del archivo de descripción para proporcionar un servicio de capacidad del dispositivo para la aplicación, para implementar la función que debe implementar la aplicación.

65

13. El dispositivo electrónico según una cualquiera de las reivindicaciones 9 a 12, en donde la obtención de un archivo de descripción de una aplicación comprende: obtener el archivo de descripción preconfigurado en un tiempo de inactividad de la aplicación;
- 5 cuando las instrucciones del software informático son ejecutadas por el dispositivo electrónico, el dispositivo electrónico está habilitado para realizar además las siguientes operaciones:
- 10 analizar el archivo de descripción y almacenar los campos comprendidos en el bloque de identificación del archivo de descripción, el campo comprendido en el área principal de la entidad lógica única en el bloque lógico del archivo de descripción y el campo comprendido en el área del cuerpo de la entidad lógica única en el bloque lógico del archivo de descripción; y
- 15 la ejecución del primer componente en función del archivo de descripción para proporcionar un servicio de capacidad de dispositivo para la aplicación, para implementar la función que debe implementar la aplicación, comprende:
- 20 leer, en tiempo de ejecución de la aplicación, el campo almacenado comprendido en el área del cuerpo de la entidad lógica única en el bloque lógico del archivo de descripción; y
- 25 analizar el campo comprendido en el área del cuerpo de la entidad lógica única en el bloque lógico del archivo de descripción y ejecutar el primer componente en función del resultado del análisis para proporcionar un servicio de capacidad de dispositivo para la aplicación, a fin de implementar la función que debe implementar la aplicación.
14. El dispositivo electrónico según una cualquiera de las reivindicaciones 9 a 12, en donde la obtención de un archivo de descripción de una aplicación comprende: generar el archivo de descripción cuando la aplicación se está ejecutando;
- 30 cuando las instrucciones del software informático son ejecutadas por el dispositivo electrónico, el dispositivo electrónico está habilitado para realizar además las siguientes operaciones:
- 35 analizar el archivo de descripción y almacenar los campos comprendidos en el bloque de identificación del archivo de descripción, el campo comprendido en el área principal de la entidad lógica única en el bloque lógico del archivo de descripción y el campo comprendido en el área del cuerpo de la entidad lógica única en el bloque lógico del archivo de descripción; y
- 40 la ejecución del primer componente en función del archivo de descripción para proporcionar un servicio de capacidad de dispositivo para la aplicación, para implementar la función que debe implementar la aplicación, comprende:
- 45 leer el campo almacenado comprendido en el área del cuerpo de la entidad lógica única en el bloque lógico del archivo de descripción; y
- 50 analizar el campo comprendido en el área del cuerpo de la entidad lógica única en el bloque lógico del archivo de descripción y ejecutar el primer componente en función del resultado del análisis para proporcionar un servicio de capacidad de dispositivo para la aplicación, a fin de implementar la función que debe implementar la aplicación.
15. El dispositivo electrónico según una cualquiera de las reivindicaciones 9 a 14, en donde la memoria almacena además la información de configuración de todos los componentes del dispositivo electrónico; y
- 55 la determinación de un primer componente en función del archivo de descripción comprende: determinar el primer componente en función del archivo de descripción y la información de configuración almacenada de todos los componentes.
16. El dispositivo electrónico según la reivindicación 15, en donde la memoria almacena además la información de configuración de todos los componentes de un dispositivo que establece una conexión inalámbrica con el dispositivo electrónico; y
- 60 cuando las instrucciones del software informático son ejecutadas por el dispositivo electrónico, el dispositivo electrónico está habilitado para realizar además la siguiente operación:
- 65 si no se determina el primer componente, y se determina, en función de la información de configuración almacenada de los componentes, que hay un componente que puede implementar la función que debe implementar la aplicación en los componentes del dispositivo conectado al dispositivo electrónico, transmitiendo el archivo de descripción al dispositivo conectado al dispositivo electrónico, en donde el

dispositivo conectado al dispositivo electrónico utiliza el archivo de descripción para determinar y ejecutar un componente correspondiente, para proporcionar un servicio de capacidad del dispositivo para aplicación e implemente la función que debe implementar la aplicación.

- 5 17. Un medio de almacenamiento legible por ordenador, que comprende instrucciones de software de ordenador, en donde cuando las instrucciones de software de ordenador se ejecutan en el dispositivo electrónico, el dispositivo electrónico está habilitado para realizar el método de interacción no intrusiva según cualquiera de las reivindicaciones 1 a 8.

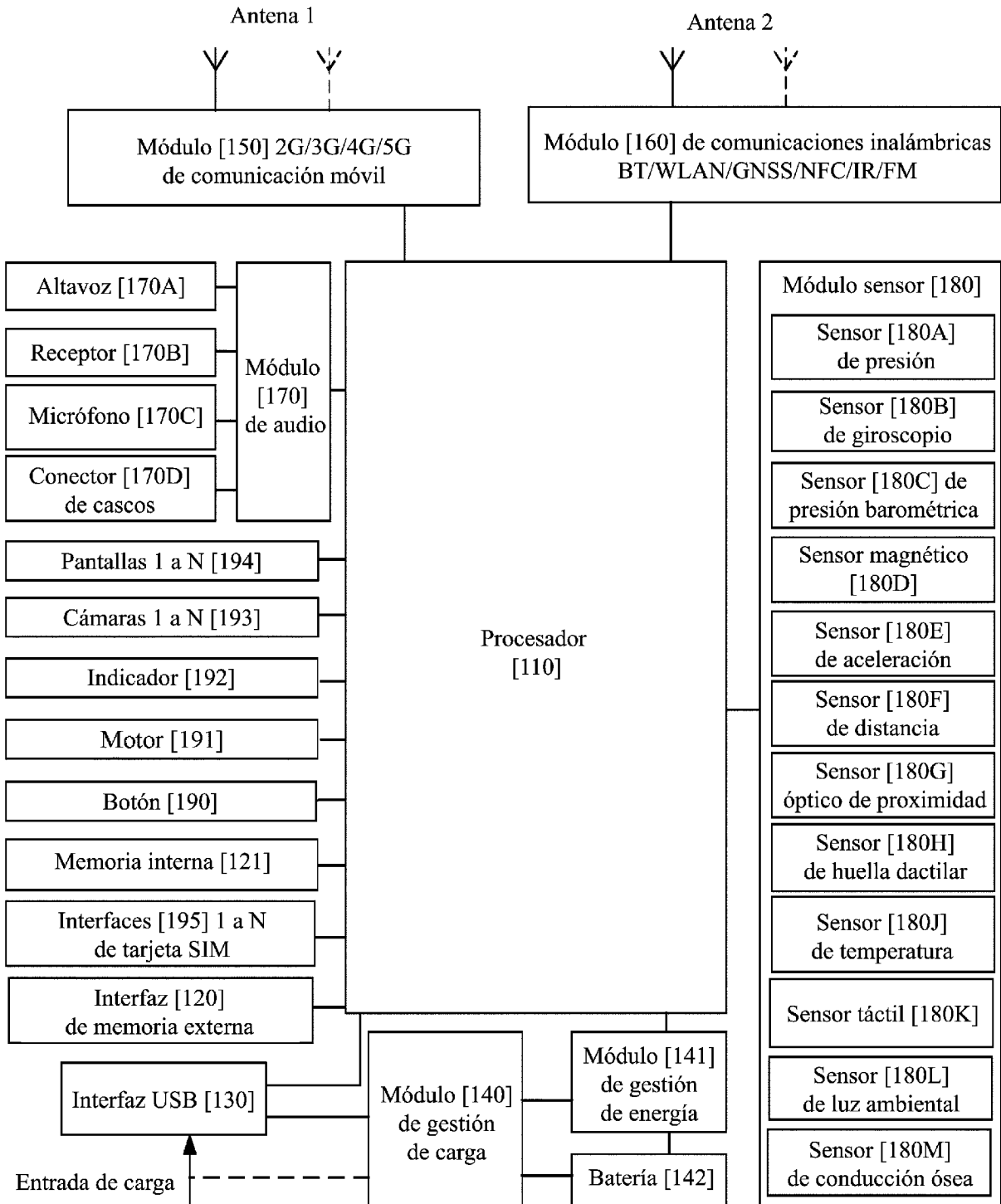


Figura 1

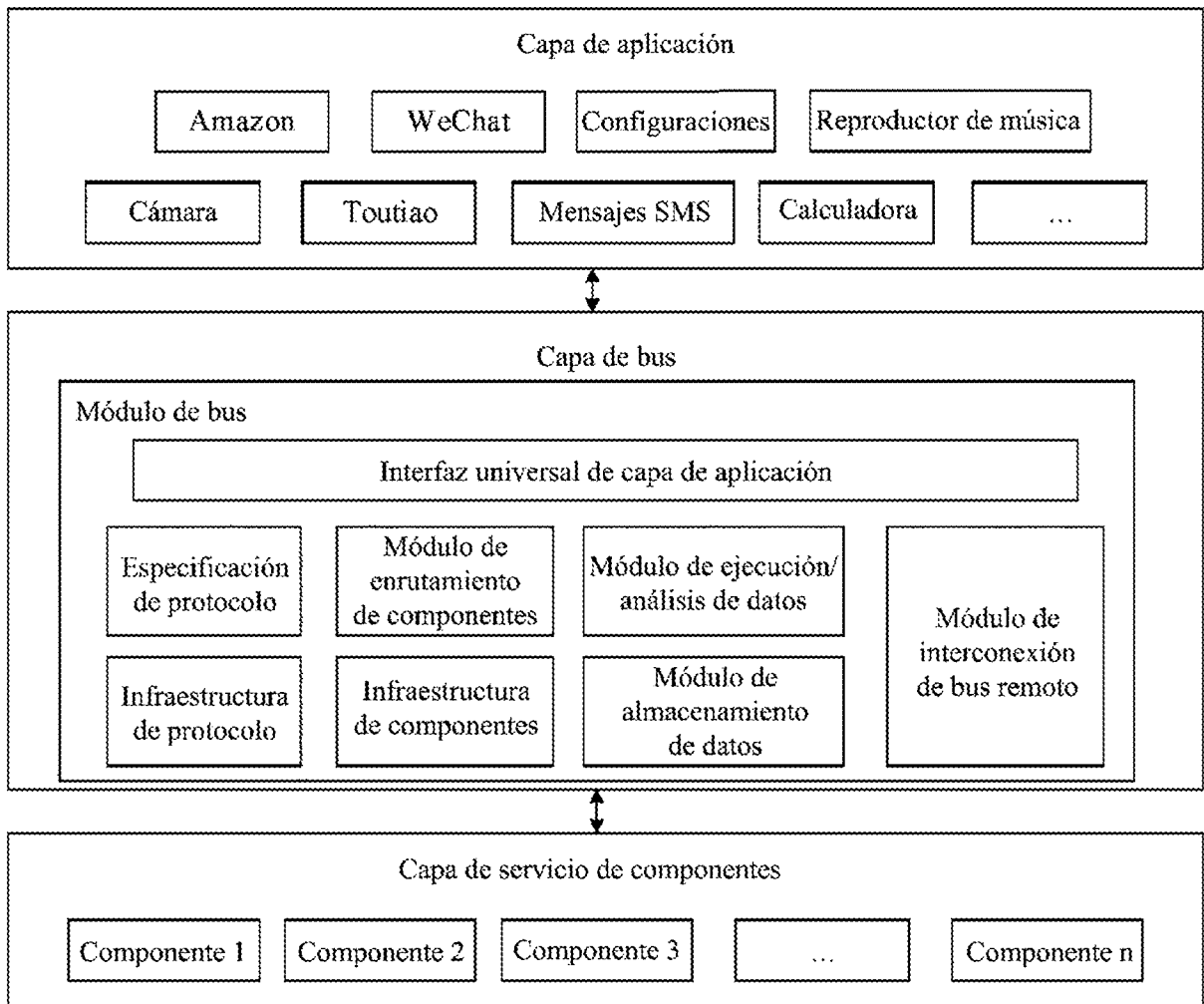


Figura 2

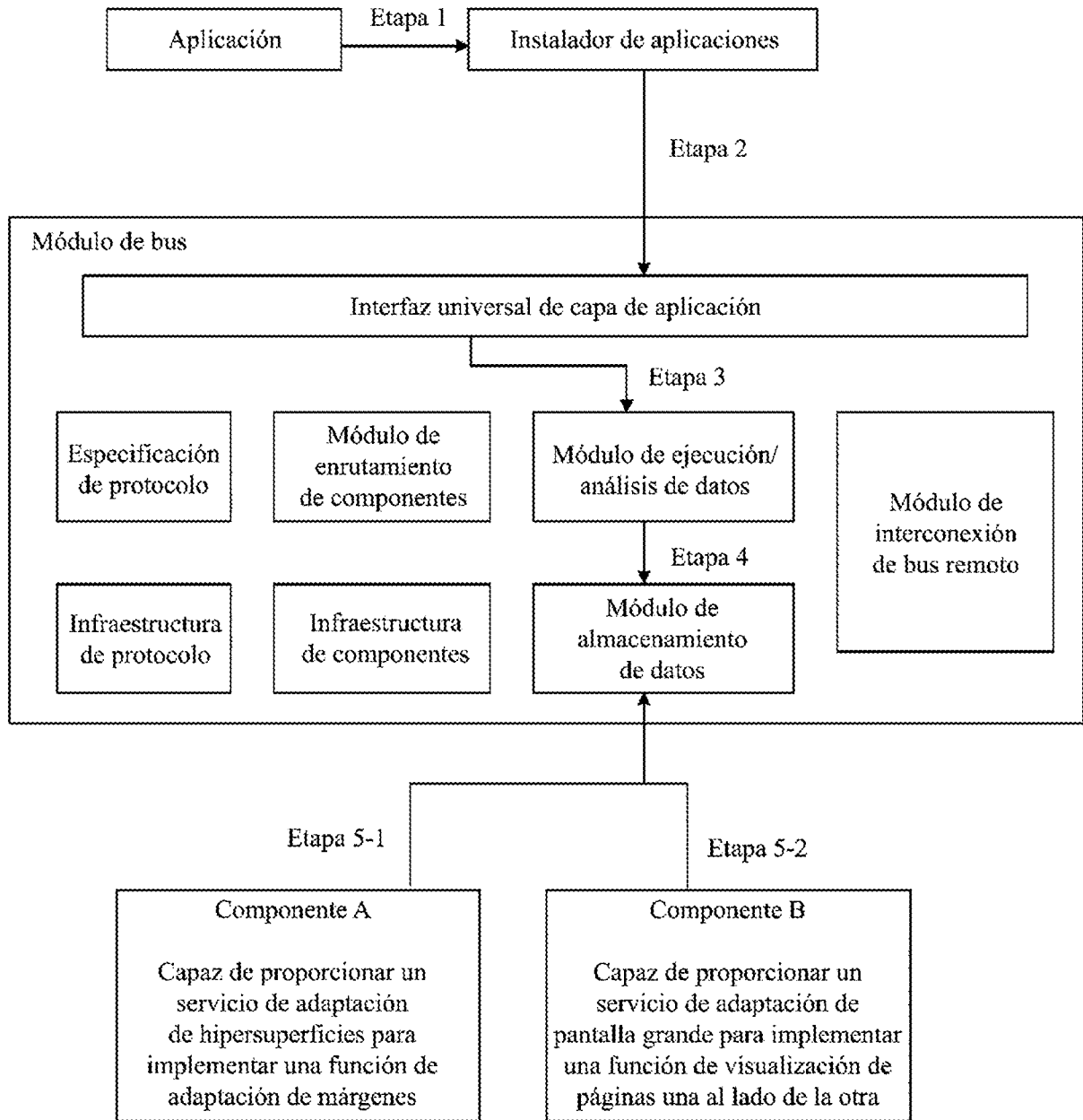


Figura 3

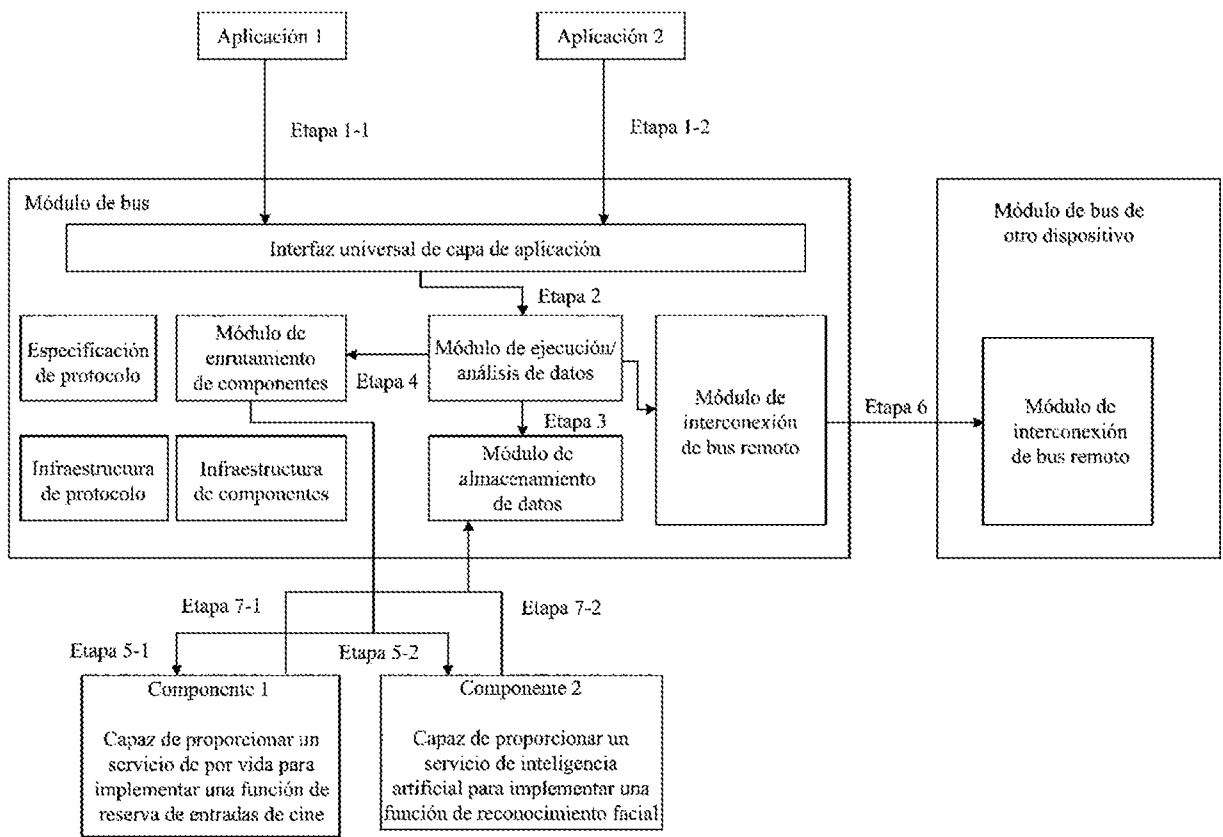
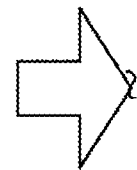
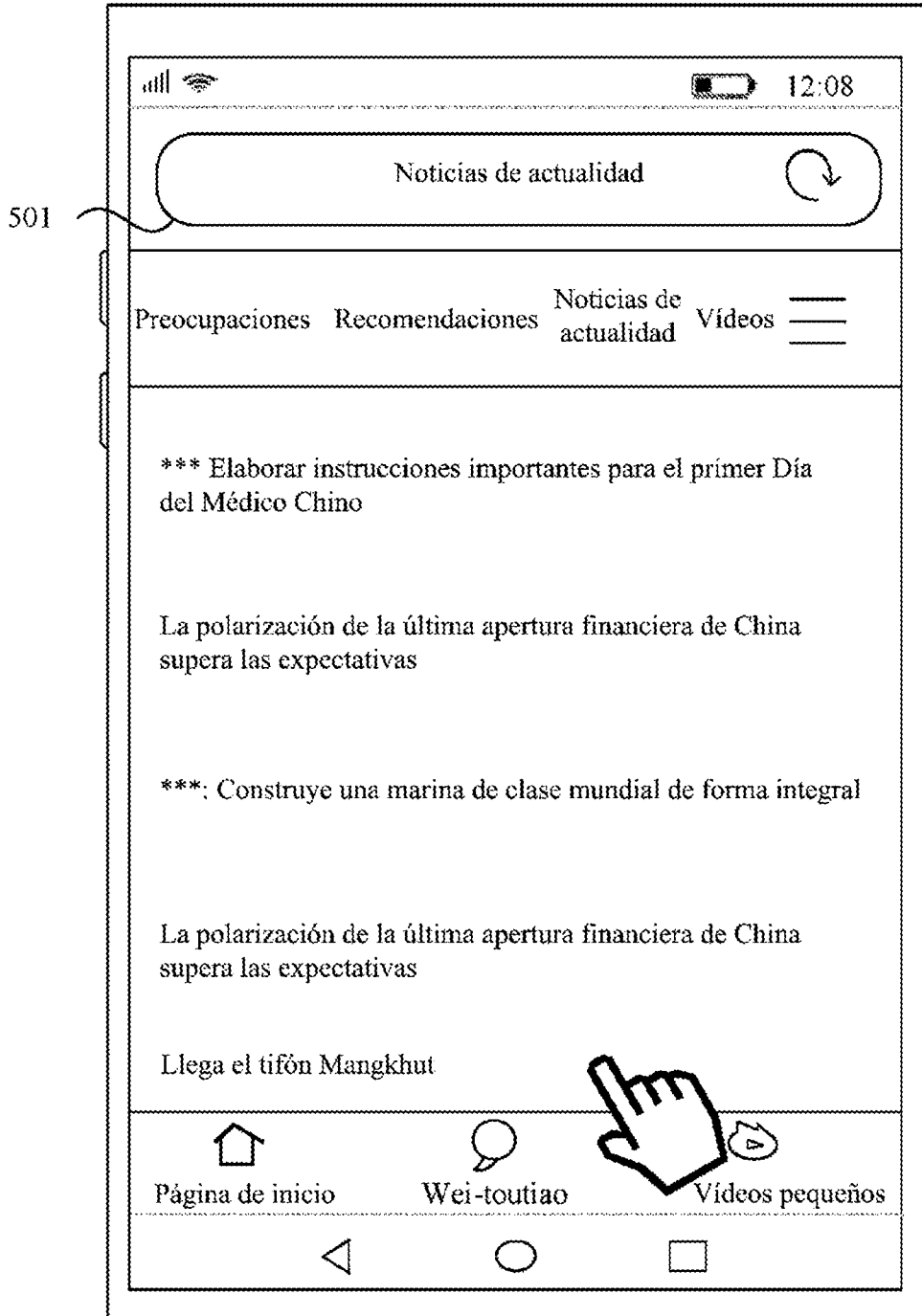


Figura 4



A FIG. 5(b)

Figura 5(a)

Cont. DESDE
FIG. 5(a)

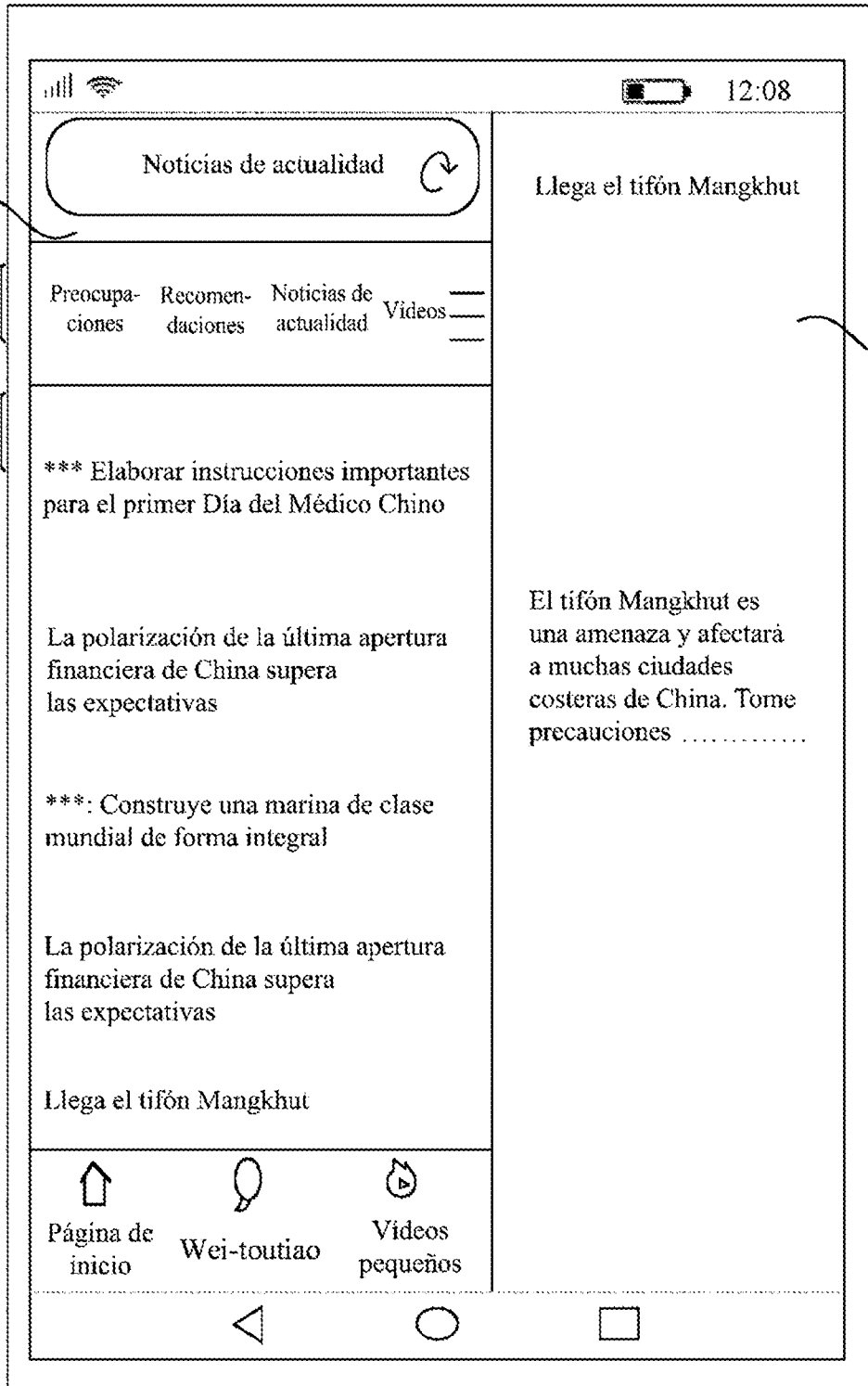


Figura 5(b)

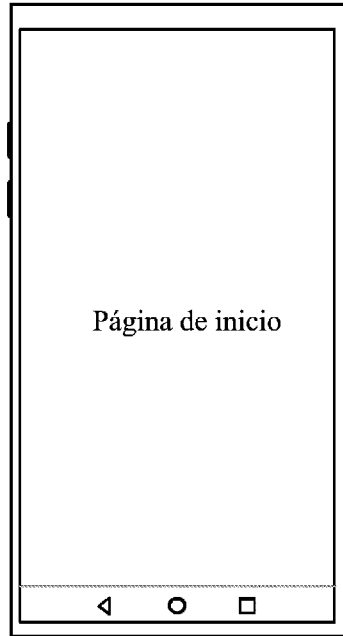


Figura 6(a)

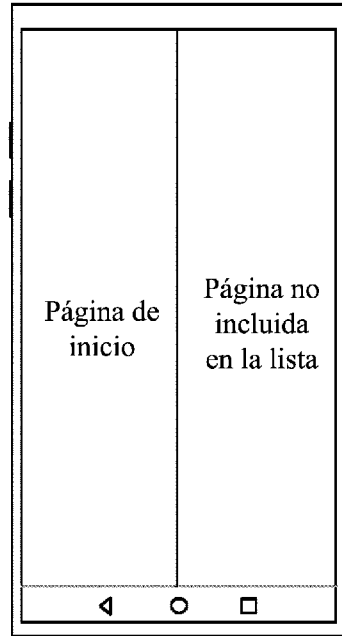


Figura 6(b)

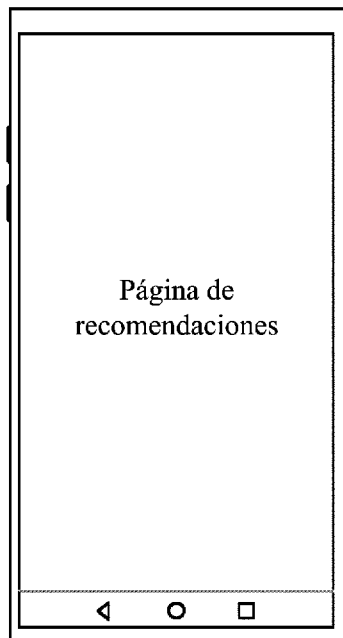


Figura 6(c)

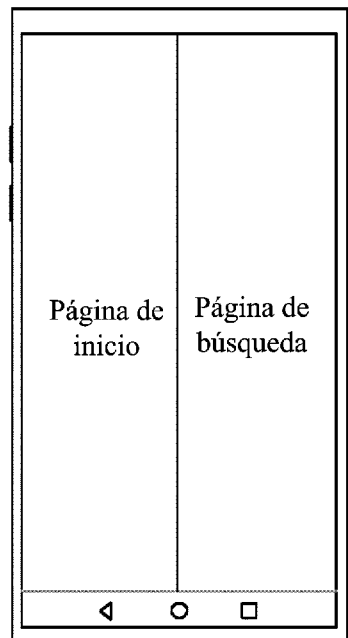


Figura 6(d)

