

명세서

발명의 명칭: 전자 장치 및 이의 동작 방법

기술분야

- [1] 일 실시 예는 전자 장치(예: 스마트폰) 및 전자 장치에 저장된 하나 이상의 어플리케이션에 관한 것이다.

배경기술

- [2] 전자 장치는 사용자에게 의해 사용될 수 있는 하나 이상의 어플리케이션을 저장할 수 있다.
- [3] 사용자가 사용 중인 어플리케이션의 오류에 의해 크래시(crash)가 발생할 수 있고, 사용 중인 어플리케이션이 강제 종료될 수 있다.

발명의 상세한 설명

과제 해결 수단

- [4] 사용 중인 어플리케이션이 강제 종료되는 경우 기존의 전자 장치는 사용자에게 해결 방법, 오류에 대한 정보, 오류 이력을 제공하지 않아, 사용자에게 불편함을 줄 수 있다.
- [5] 일 실시는 어플리케이션의 오류 원인(또는 크래시 원인)을 분석하여 사용자에게 해결 방법을 제공할 수 있는 전자 장치를 제공할 수 있다.
- [6] 일 실시는 어플리케이션의 크래시가 발생했을 때 어플리케이션의 오류 원인과 이력을 사용자에게 제공하여 사용자의 불편함을 최소화(또는 감소) 해줄 수 있는 전자 장치를 제공할 수 있다.
- [7] 일 실시는 어플리케이션의 오류 해결이 가능하지 않은 경우 사용자에게 어플리케이션의 업데이트를 안내할 수 있는 전자 장치를 제공할 수 있다.
- [8] 본 문서에서 이루고자 하는 기술적 과제는 위에서 언급한 기술적 과제로 제한되지 않으며, 언급되지 않은 또 다른 기술적 과제들은 아래의 기재로부터 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자에게 명확하게 이해될 수 있을 것이다.
- [9] 개시의 일 측면에 따르면, 전자 장치는 디스플레이; 프로세서; 및 상기 프로세서와 작동적으로 연결되고 상기 프로세서에 의해 실행되는 하나 이상의 명령어들을 저장하는 메모리를 포함할 수 있다. 상기 명령어들이 실행될 때 상기 프로세서는 상기 프로세서에 의해 실행되는 제1 어플리케이션에 발생한 오류를 기초로 오류 정보를 획득하고, 상기 획득된 오류 정보를 기초로 상기 오류의 타입과 상기 오류의 원인을 파악하며, 상기 오류의 상기 파악된 타입 및 상기 오류의 파악된 원인을 이용하여 참조 데이터 셋으로부터 상기 오류를 해결하기 위한 해결 방법을 획득하고, 상기 전자 장치의 상태 정보를 기초로 상기 획득된 해결 방법의 적합성을 검증하고, 상기 적합한 것으로 검증된 획득된 해결 방법을 기초로 상기 디스플레이를 통해 상기 획득된 해결 방법에 대한 안내를

사용자에게 제공할 수 있다. 상기 참조 데이터 셋은 하나 이상의 오류 타입, 하나 이상의 오류 발생 원인, 및 하나 이상의 오류 해결 방법을 포함할 수 있다.

- [10] 개시의 다른 측면에 따르면, 전자 장치는 디스플레이, 프로세서, 및 상기 프로세서와 작동적으로 연결되고, 참조 데이터 셋과 하나 이상의 어플리케이션을 저장하는 메모리를 포함할 수 있다. 참조 데이터 셋은 복수의 오류 타입들, 복수의 오류 발생 원인들, 및 복수의 오류 해결 방법들을 포함할 수 있다. 메모리는 상기 프로세서에 의해 실행되는 명령어들을 저장할 수 있다. 상기 명령어들이 실행될 때 상기 프로세서는 실행되는 제1 어플리케이션에 발생한 오류를 기초로 오류 정보를 획득하고, 상기 획득된 오류 정보를 기초로 상기 제1 어플리케이션이 서드 파티 어플리케이션인지 여부를 결정하며, 상기 제1 어플리케이션이 서드 파티 어플리케이션으로 결정된 것에 기초하여, 상기 획득된 오류 정보에서 상기 오류의 타입과 상기 오류의 원인을 파악하고, 상기 파악된 타입 및 상기 파악된 원인을 이용하여 상기 참조 데이터 셋으로부터 상기 오류를 해결하기 위한 해결 방법을 획득하고, 상기 전자 장치의 상태 정보를 기초로 상기 획득된 해결 방법의 적합성을 검증하고, 상기 획득된 해결 방법이 적합한 것으로 검증된 경우 상기 획득된 해결 방법에 대한 안내를 위한 팝업 창을 상기 디스플레이에 표시하는 프로세서를 포함할 수 있다.
- [11] 개시의 다른 측면에 따르면, 전자 장치의 방법은 실행되는 제1 어플리케이션에 발생한 오류를 기초로 오류 정보를 획득하는 동작, 상기 획득된 오류 정보에서 상기 오류의 타입과 상기 오류의 원인을 파악하는 동작, 상기 파악된 타입 및 상기 파악된 원인을 이용하여 참조 데이터 셋으로부터 상기 오류를 해결하기 위한 해결 방법을 획득하는 동작, 상기 전자 장치의 상태 정보를 기초로 상기 획득된 해결 방법의 적합성을 검증하는 동작, 및 적합한 것으로 검증된 상기 획득된 해결 방법에 기초하여 상기 획득된 해결 방법에 대한 안내를 디스플레이를 통해 사용자에게 제공하는 동작을 포함할 수 있다. 상기 참조 데이터 셋은 하나 이상의 오류 타입, 하나 이상의 오류 발생 원인, 및 하나 이상의 오류 해결 방법을 포함할 수 있다.
- [12] 개시의 다른 측면에 따르면, 전자 장치는 디스플레이; 프로세서; 및 상기 프로세서와 작동적으로 연결되는 메모리를 포함할 수 있다. 메모리는 하나 이상의 명령어들을 저장할 수 있다. 상기 명령어들이 상기 프로세서에 의해 실행될 때 상기 프로세서는 상기 프로세서에 의해 실행되는 제1 어플리케이션에 발생한 오류를 기초로 오류 정보를 획득하고, 제1 시간 인스턴스와 상기 제1 어플리케이션에 상기 오류가 발생한 제2 시간 인스턴스 사이의 시간 구간에 대해 상기 전자 장치의 상태 정보를 분석하며(여기서, 상기 제1 시간 인스턴스는 상기 제2 시간 인스턴스보다 빠른 시간 인스턴스임), 상기 전자 장치의 상기 분석된 상태 정보를 기초로 상기 전자 장치의 상태를 결정하고, 상기 획득된 오류 정보로부터 예외 타입 또는 오류 원인을 결정하고, (i) 상기 결정된 오류 타입 또는 (ii) 상기 결정된 오류 원인 중 적어도 하나와 상기 전자 장치의 상기

결정된 상태에 기초하여, 상기 오류에 대응되는 하나 이상의 해결 방법들을 데이터베이스로부터 획득할 수 있다.

- [13] 이 외에, 본 문서를 통해 직접적 또는 간접적으로 파악되는 다양한 효과들이 제공될 수 있다.

도면의 간단한 설명

- [14] 도 1은 일 실시예에 따른, 네트워크 환경 내의 전자 장치의 블록도를 도시한다.
 [15] 도 2는 일 실시예에 따른, 프로그램을 예시하는 블록도이다.
 [16] 도 3은 일 실시예에 따른, 전자 장치의 일 예시를 설명하기 위한 블록도이다.
 [17] 도 4는 일 실시예에 따른, 전자 장치의 운영 체제의 프레임워크와 시스템 어플리케이션의 일 예시를 설명하기 위한 도면이다.
 [18] 도 5는 일 실시예에 따른, 전자 장치가 DB로부터 오류 해결 방법을 획득하는 다른 예시를 설명하기 위한 도면이다.
 [19] 도 6은 일 실시예에 따른, 전자 장치가 DB로부터 오류 해결 방법을 획득하는 또 다른 예시를 설명하기 위한 도면이다.
 [20] 도 7은 일 실시예에 따른, 전자 장치의 어플리케이션의 크래시가 발생한 경우 해결 방법을 제공하는 일 예시를 설명하기 위한 흐름도이다.
 [21] 도 8, 도 9, 및 도 10은 일 실시예에 따른, 전자 장치의 팝업 창의 예시들을 설명하기 위한 도면이다.
 [22] 도 11 및 도 12는 일 실시예에 따른, 전자 장치가 획득한 해결 방법들의 순서 변경의 예시들을 설명하기 위한 도면이다.
 [23] 도 13은 일 실시예에 따른, 전자 장치의 동작 방법의 일 예시를 설명하기 위한 흐름도이다.

발명의 실시를 위한 최선의 형태

- [24] 이하, 실시 예들을 첨부된 도면들을 참조하여 상세하게 설명한다. 첨부 도면을 참조하여 설명함에 있어, 도면 부호에 관계없이 동일한 구성 요소는 동일한 참조 부호를 부여하고, 이에 대한 중복되는 설명은 생략하기로 한다.
- [25] 도 1은, 일 실시예에 따른, 네트워크 환경(100) 내의 전자 장치(101)의 블록도이다. 도 1을 참조하면, 네트워크 환경(100)에서 전자 장치(101)는 제 1 네트워크(198)(예: 근거리 무선 통신 네트워크)를 통하여 전자 장치(102)와 통신하거나, 또는 제 2 네트워크(199)(예: 원거리 무선 통신 네트워크)를 통하여 전자 장치(104) 또는 서버(108) 중 적어도 하나와 통신할 수 있다. 일 실시예에 따르면, 전자 장치(101)는 서버(108)를 통하여 전자 장치(104)와 통신할 수 있다. 일 실시예에 따르면, 전자 장치(101)는 프로세서(120), 메모리(130), 입력 모듈(150), 음향 출력 모듈(155), 디스플레이 모듈(160), 오디오 모듈(170), 센서 모듈(176), 인터페이스(177), 연결 단자(178), 햅틱 모듈(179), 카메라 모듈(180), 전력 관리 모듈(188), 배터리(189), 통신 모듈(190), 가입자 식별 모듈(196), 또는 안테나 모듈(197)을 포함할 수 있다. 어떤 실시예에서는, 전자 장치(101)에는, 이

구성요소들 중 적어도 하나(예: 연결 단자(178))가 생략되거나, 하나 이상의 다른 구성요소가 추가될 수 있다. 어떤 실시 예에서는, 이 구성요소들 중 일부들(예: 센서 모듈(176), 카메라 모듈(180), 또는 안테나 모듈(197))은 하나의 구성요소(예: 디스플레이 모듈(160))로 통합될 수 있다.

- [26] 프로세서(120)는, 예를 들면, 소프트웨어(예: 프로그램(140))를 실행하여 프로세서(120)에 연결된 전자 장치(101)의 적어도 하나의 다른 구성요소(예: 하드웨어 또는 소프트웨어 구성요소)를 제어할 수 있고, 다양한 데이터 처리 또는 연산을 수행할 수 있다. 일 실시 예에 따르면, 데이터 처리 또는 연산의 적어도 일부로서, 프로세서(120)는 다른 구성요소(예: 센서 모듈(176) 또는 통신 모듈(190))로부터 수신된 명령 또는 데이터를 휘발성 메모리(132)에 저장하고, 휘발성 메모리(132)에 저장된 명령 또는 데이터를 처리하고, 결과 데이터를 비휘발성 메모리(134)에 저장할 수 있다. 일 실시 예에 따르면, 프로세서(120)는 메인 프로세서(121)(예: 중앙 처리 장치 또는 어플리케이션 프로세서) 또는 이와는 독립적으로 또는 함께 운영 가능한 보조 프로세서(123)(예: 그래픽 처리 장치, 신경망 처리 장치(NPU: neural processing unit), 이미지 시그널 프로세서, 센서 허브 프로세서, 또는 커뮤니케이션 프로세서)를 포함할 수 있다. 예를 들어, 전자 장치(101)가 메인 프로세서(121) 및 보조 프로세서(123)를 포함하는 경우, 보조 프로세서(123)는 메인 프로세서(121)보다 저전력을 사용하거나, 지정된 기능에 특화되도록 설정될 수 있다. 보조 프로세서(123)는 메인 프로세서(121)와 별개로, 또는 그 일부로서 구현될 수 있다.
- [27] 보조 프로세서(123)는, 예를 들면, 메인 프로세서(121)가 인액티브(예: 슬립) 상태에 있는 동안 메인 프로세서(121)를 대신하여, 또는 메인 프로세서(121)가 액티브(예: 어플리케이션 실행) 상태에 있는 동안 메인 프로세서(121)와 함께, 전자 장치(101)의 구성요소들 중 적어도 하나의 구성요소(예: 디스플레이 모듈(160), 센서 모듈(176), 또는 통신 모듈(190))와 관련된 기능 또는 상태들의 적어도 일부를 제어할 수 있다. 일 실시 예에 따르면, 보조 프로세서(123)(예: 이미지 시그널 프로세서 또는 커뮤니케이션 프로세서)는 기능적으로 관련 있는 다른 구성요소(예: 카메라 모듈(180) 또는 통신 모듈(190))의 일부로서 구현될 수 있다. 일 실시 예에 따르면, 보조 프로세서(123)(예: 신경망 처리 장치)는 인공지능 모델의 처리에 특화된 하드웨어 구조를 포함할 수 있다. 인공지능 모델은 기계 학습을 통해 생성될 수 있다. 이러한 학습은, 예를 들어, 인공지능 모델이 수행되는 전자 장치(101) 자체에서 수행될 수 있고, 별도의 서버(예: 서버(108))를 통해 수행될 수도 있다. 학습 알고리즘은, 예를 들어, 지도형 학습(supervised learning), 비지도형 학습(unsupervised learning), 준지도형 학습(semi-supervised learning) 또는 강화 학습(reinforcement learning)을 포함할 수 있으나, 전술한 예에 한정되지 않는다. 인공지능 모델은, 복수의 인공 신경망 레이어들을 포함할 수 있다. 인공 신경망은 심층 신경망(DNN: deep neural network), CNN(convolutional neural network), RNN(recurrent neural network), RBM(restricted boltzmann machine),

DBN(deep belief network), BRDNN(bidirectional recurrent deep neural network), 심층 Q-네트워크(deep Q-networks) 또는 상기 중 둘 이상의 조합 중 하나일 수 있으나, 전술한 예에 한정되지 않는다. 인공지능 모델은 하드웨어 구조 이외에, 추가적으로 또는 대체적으로, 소프트웨어 구조를 포함할 수 있다.

- [28] 메모리(130)는, 전자 장치(101)의 적어도 하나의 구성요소(예: 프로세서(120) 또는 센서 모듈(176))에 의해 사용되는 다양한 데이터를 저장할 수 있다. 데이터는, 예를 들어, 소프트웨어(예: 프로그램(140)) 및, 이와 관련된 명령에 대한 입력 데이터 또는 출력 데이터를 포함할 수 있다. 메모리(130)는, 휘발성 메모리(132) 또는 비휘발성 메모리(134)를 포함할 수 있다.
- [29] 프로그램(140)은 메모리(130)에 소프트웨어로서 저장될 수 있으며, 예를 들면, 운영 체제(142), 미들 웨어(144) 또는 어플리케이션(146)을 포함할 수 있다.
- [30] 입력 모듈(150)은, 전자 장치(101)의 구성요소(예: 프로세서(120))에 사용될 명령 또는 데이터를 전자 장치(101)의 외부(예: 사용자)로부터 수신할 수 있다. 입력 모듈(150)은, 예를 들면, 마이크, 마우스, 키보드, 키(예: 버튼), 또는 디지털 펜(예: 스타일러스 펜)을 포함할 수 있다.
- [31] 음향 출력 모듈(155)은 음향 신호를 전자 장치(101)의 외부로 출력할 수 있다. 음향 출력 모듈(155)은, 예를 들면, 스피커 또는 리시버를 포함할 수 있다. 스피커는 멀티미디어 재생 또는 녹음 재생과 같이 일반적인 용도로 사용될 수 있다. 리시버는 착신 전화를 수신하기 위해 사용될 수 있다. 일 실시 예에 따르면, 리시버는 스피커와 별개로, 또는 그 일부로서 구현될 수 있다.
- [32] 디스플레이 모듈(160)은 전자 장치(101)의 외부(예: 사용자)로 정보를 시각적으로 제공할 수 있다. 디스플레이 모듈(160)은, 예를 들면, 디스플레이, 홀로그램 장치, 또는 프로젝터 및 해당 장치를 제어하기 위한 제어 회로를 포함할 수 있다. 일 실시 예에 따르면, 디스플레이 모듈(160)은 터치를 감지하도록 설정된 터치 센서, 또는 상기 터치에 의해 발생하는 힘의 세기를 측정하도록 설정된 압력 센서를 포함할 수 있다.
- [33] 오디오 모듈(170)은 소리를 전기 신호로 변환시키거나, 반대로 전기 신호를 소리로 변환시킬 수 있다. 일 실시 예에 따르면, 오디오 모듈(170)은, 입력 모듈(150)을 통해 소리를 획득하거나, 음향 출력 모듈(155), 또는 전자 장치(101)와 직접 또는 무선으로 연결된 외부 전자 장치(예: 전자 장치(102))(예: 스피커 또는 헤드폰)를 통해 소리를 출력할 수 있다.
- [34] 센서 모듈(176)은 전자 장치(101)의 작동 상태(예: 전력 또는 온도), 또는 외부의 환경 상태(예: 사용자 상태)를 감지하고, 감지된 상태에 대응하는 전기 신호 또는 데이터 값을 생성할 수 있다. 일 실시 예에 따르면, 센서 모듈(176)은, 예를 들면, 제스처 센서, 자이로 센서, 기압 센서, 마그네틱 센서, 가속도 센서, 그립 센서, 근접 센서, 컬러 센서, IR(infrared) 센서, 생체 센서, 온도 센서, 습도 센서, 또는 조도 센서를 포함할 수 있다.
- [35] 인터페이스(177)는 전자 장치(101)가 외부 전자 장치(예: 전자 장치(102))와

- 직접 또는 무선으로 연결되기 위해 사용될 수 있는 하나 이상의 지정된 프로토콜들을 지원할 수 있다. 일 실시 예에 따르면, 인터페이스(177)는, 예를 들면, HDMI(high definition multimedia interface), USB(universal serial bus) 인터페이스, SD카드 인터페이스, 또는 오디오 인터페이스를 포함할 수 있다.
- [36] 연결 단자(178)는, 그를 통해서 전자 장치(101)가 외부 전자 장치(예: 전자 장치(102))와 물리적으로 연결될 수 있는 커넥터를 포함할 수 있다. 일 실시 예에 따르면, 연결 단자(178)는, 예를 들면, HDMI 커넥터, USB 커넥터, SD 카드 커넥터, 또는 오디오 커넥터(예: 헤드폰 커넥터)를 포함할 수 있다.
- [37] 햅틱 모듈(179)은 전기적 신호를 사용자가 촉각 또는 운동 감각을 통해서 인지할 수 있는 기계적인 자극(예: 진동 또는 움직임) 또는 전기적인 자극으로 변환할 수 있다. 일 실시 예에 따르면, 햅틱 모듈(179)은, 예를 들면, 모터, 압전 소자, 또는 전기 자극 장치를 포함할 수 있다.
- [38] 카메라 모듈(180)은 정지 영상 및 동영상을 촬영할 수 있다. 일 실시 예에 따르면, 카메라 모듈(180)은 하나 이상의 렌즈들, 이미지 센서들, 이미지 시그널 프로세서들, 또는 플래시들을 포함할 수 있다.
- [39] 전력 관리 모듈(188)은 전자 장치(101)에 공급되는 전력을 관리할 수 있다. 일 실시 예에 따르면, 전력 관리 모듈(188)은, 예를 들면, PMIC(power management integrated circuit)의 적어도 일부로서 구현될 수 있다.
- [40] 배터리(189)는 전자 장치(101)의 적어도 하나의 구성요소에 전력을 공급할 수 있다. 일 실시 예에 따르면, 배터리(189)는, 예를 들면, 재충전 불가능한 1차 전지, 재충전 가능한 2차 전지 또는 연료 전지를 포함할 수 있다.
- [41] 통신 모듈(190)은 전자 장치(101)와 외부 전자 장치(예: 전자 장치(102), 전자 장치(104), 또는 서버(108)) 간의 직접(예: 유선) 통신 채널 또는 무선 통신 채널의 수립, 및 수립된 통신 채널을 통한 통신 수행을 지원할 수 있다. 통신 모듈(190)은 프로세서(120)(예: 어플리케이션 프로세서)와 독립적으로 운영되고, 직접(예: 유선) 통신 또는 무선 통신을 지원하는 하나 이상의 커뮤니케이션 프로세서를 포함할 수 있다. 일 실시 예에 따르면, 통신 모듈(190)은 무선 통신 모듈(192)(예: 셀룰러 통신 모듈, 근거리 무선 통신 모듈, 또는 GNSS(global navigation satellite system) 통신 모듈) 또는 유선 통신 모듈(194)(예: LAN(local area network) 통신 모듈, 또는 전력선 통신 모듈)을 포함할 수 있다. 이들 통신 모듈 중 해당하는 통신 모듈은 제 1 네트워크(198)(예: 블루투스, WiFi(wireless fidelity) direct 또는 IrDA(infrared data association)와 같은 근거리 통신 네트워크) 또는 제 2 네트워크(199)(예: 레거시 셀룰러 네트워크, 5G 네트워크, 차세대 통신 네트워크, 인터넷, 또는 컴퓨터 네트워크(예: LAN 또는 WAN)와 같은 원거리 통신 네트워크)를 통하여 외부의 전자 장치(104)와 통신할 수 있다. 이런 여러 종류의 통신 모듈들은 하나의 구성요소(예: 단일 칩)로 통합되거나, 또는 서로 별도의 복수의 구성요소들(예: 복수 칩들)로 구현될 수 있다. 무선 통신 모듈(192)은 가입자 식별 모듈(196)에 저장된 가입자 정보(예: 국제 모바일 가입자

식별자(IMSII)를 이용하여 제 1 네트워크(198) 또는 제 2 네트워크(199)와 같은 통신 네트워크 내에서 전자 장치(101)를 확인 또는 인증할 수 있다.

- [42] 무선 통신 모듈(192)은 4G 네트워크 이후의 5G 네트워크 및 차세대 통신 기술, 예를 들어, NR 접속 기술(new radio access technology)을 지원할 수 있다. NR 접속 기술은 고용량 데이터의 고속 전송(eMBB(enhanced mobile broadband)), 단말 전력 최소화 및 다수 단말의 접속(mMTC(massive machine type communications)), 또는 고신뢰도와 저지연(URLLC(ultra-reliable and low-latency communications))을 지원할 수 있다. 무선 통신 모듈(192)은, 예를 들어, 높은 데이터 전송률 달성을 위해, 고주파 대역(예: mmWave 대역)을 지원할 수 있다. 무선 통신 모듈(192)은 고주파 대역에서의 성능 확보를 위한 다양한 기술들, 예를 들어, 빔포밍(beamforming), 거대 배열 다중 입출력(massive MIMO(multiple-input and multiple-output)), 전차원 다중입출력(FD-MIMO: full dimensional MIMO), 어레이 안테나(array antenna), 아날로그 빔형성(analog beam-forming), 또는 대규모 안테나(large scale antenna)와 같은 기술들을 지원할 수 있다. 무선 통신 모듈(192)은 전자 장치(101), 외부 전자 장치(예: 전자 장치(104)) 또는 네트워크 시스템(예: 제 2 네트워크(199))에 규정되는 다양한 요구사항을 지원할 수 있다. 일 실시 예에 따르면, 무선 통신 모듈(192)은 eMBB 실현을 위한 Peak data rate(예: 20Gbps 이상), mMTC 실현을 위한 손실 Coverage(예: 164dB 이하), 또는 URLLC 실현을 위한 U-plane latency(예: 다운링크(DL) 및 업링크(UL) 각각 0.5ms 이하, 또는 라운드 트립 1ms 이하)를 지원할 수 있다.

- [43] 안테나 모듈(197)은 신호 또는 전력을 외부(예: 외부의 전자 장치)로 송신하거나 외부로부터 수신할 수 있다. 일 실시 예에 따르면, 안테나 모듈(197)은 서브스트레이트(예: PCB) 위에 형성된 도전체 또는 도전성 패턴으로 이루어진 방사체를 포함하는 안테나를 포함할 수 있다. 일 실시 예에 따르면, 안테나 모듈(197)은 복수의 안테나들(예: 어레이 안테나)을 포함할 수 있다. 이런 경우, 제 1 네트워크(198) 또는 제 2 네트워크(199)와 같은 통신 네트워크에서 사용되는 통신 방식에 적합한 적어도 하나의 안테나가, 예를 들면, 통신 모듈(190)에 의하여 상기 복수의 안테나들로부터 선택될 수 있다. 신호 또는 전력은 상기 선택된 적어도 하나의 안테나를 통하여 통신 모듈(190)과 외부의 전자 장치 간에 송신되거나 수신될 수 있다. 어떤 실시 예에 따르면, 방사체 이외에 다른 부품(예: RFIC(radio frequency integrated circuit))이 추가로 안테나 모듈(197)의 일부로 형성될 수 있다.

- [44] 다양한 실시 예에 따르면, 안테나 모듈(197)은 mmWave 안테나 모듈을 형성할 수 있다. 일 실시 예에 따르면, mmWave 안테나 모듈은 인쇄 회로 기판, 상기 인쇄 회로 기판의 제 1 면(예: 아래 면)에 또는 그에 인접하여 배치되고 지정된 고주파 대역(예: mmWave 대역)을 지원할 수 있는 RFIC, 및 상기 인쇄 회로 기판의 제 2 면(예: 윗 면 또는 측 면)에 또는 그에 인접하여 배치되고 상기 지정된 고주파 대역의 신호를 송신 또는 수신할 수 있는 복수의 안테나들(예: 어레이 안테나)을

포함할 수 있다.

- [45] 상기 구성요소들 중 적어도 일부는 주변 기기들간 통신 방식(예: 버스, GPIO(general purpose input and output), SPI(serial peripheral interface), 또는 MIPI(mobile industry processor interface))을 통해 서로 연결되고 신호(예: 명령 또는 데이터)를 상호간에 교환할 수 있다.
- [46] 일 실시 예에 따르면, 명령 또는 데이터는 제 2 네트워크(199)에 연결된 서버(108)를 통해서 전자 장치(101)와 외부의 전자 장치(104)간에 송신 또는 수신될 수 있다. 외부의 전자 장치(102, 또는 104) 각각은 전자 장치(101)와 동일한 또는 다른 종류의 장치일 수 있다. 일 실시 예에 따르면, 전자 장치(101)에서 실행되는 동작들의 전부 또는 일부는 외부의 전자 장치들(102, 104, 또는 108) 중 하나 이상의 외부의 전자 장치들에서 실행될 수 있다. 예를 들면, 전자 장치(101)가 어떤 기능이나 서비스를 자동으로, 또는 사용자 또는 다른 장치로부터의 요청에 반응하여 수행해야 할 경우에, 전자 장치(101)는 기능 또는 서비스를 자체적으로 실행시키는 대신에 또는 추가적으로, 하나 이상의 외부의 전자 장치들에게 그 기능 또는 그 서비스의 적어도 일부를 수행하라고 요청할 수 있다. 상기 요청을 수신한 하나 이상의 외부의 전자 장치들은 요청된 기능 또는 서비스의 적어도 일부, 또는 상기 요청과 관련된 추가 기능 또는 서비스를 실행하고, 그 실행의 결과를 전자 장치(101)로 전달할 수 있다. 전자 장치(101)는 상기 결과를, 그대로 또는 추가적으로 처리하여, 상기 요청에 대한 응답의 적어도 일부로서 제공할 수 있다. 이를 위하여, 예를 들면, 클라우드 컴퓨팅, 분산 컴퓨팅, 모바일 에지 컴퓨팅(MEC: mobile edge computing), 또는 클라이언트-서버 컴퓨팅 기술이 이용될 수 있다. 전자 장치(101)는, 예를 들어, 분산 컴퓨팅 또는 모바일 에지 컴퓨팅을 이용하여 초저지연 서비스를 제공할 수 있다. 다른 실시 예에 있어서, 외부의 전자 장치(104)는 IoT(internet of things) 기기를 포함할 수 있다. 서버(108)는 기계 학습 및/또는 신경망을 이용한 지능형 서버일 수 있다. 일 실시 예에 따르면, 외부의 전자 장치(104) 또는 서버(108)는 제 2 네트워크(199) 내에 포함될 수 있다. 전자 장치(101)는 5G 통신 기술 및 IoT 관련 기술을 기반으로 지능형 서비스(예: 스마트 홈, 스마트 시티, 스마트 카, 또는 헬스케어)에 적용될 수 있다.
- [47]
- [48] 도 2는 일 실시예에 따른, 프로그램(140)을 예시하는 블록도(200)이다. 일 실시예에 있어서, 프로그램(140)은 전자 장치(101)의 하나 이상의 리소스들을 제어하기 위한 운영 체제(142), 미들웨어(144), 또는 상기 운영 체제(142)에서 실행 가능한 어플리케이션(146)을 포함할 수 있다. 운영 체제(142)는, 예를 들면, Android™, iOS™, Windows™, Symbian™, Tizen™, 또는 Bada™를 포함할 수 있다. 프로그램(140) 중 적어도 일부 프로그램은, 예를 들면, 제조 시에 전자 장치(101)에 프리로드되거나, 또는 사용자에게 의해 사용 시 외부 전자 장치(예: 전자 장치(102 또는 104), 또는 서버(108))로부터 다운로드되거나 갱신될 수

있다.

- [49] 운영 체제(142)는 전자 장치(101)의 하나 이상의 시스템 리소스들(예: 프로세스, 메모리, 또는 전원)의 관리(예: 할당 또는 회수)를 제어할 수 있다. 운영 체제(142)는, 추가적으로 또는 대체적으로, 전자 장치(101)의 다른 하드웨어 디바이스, 예를 들면, 입력 모듈(150), 음향 출력 모듈(155), 디스플레이 모듈(160), 오디오 모듈(170), 센서 모듈(176), 인터페이스(177), 햅틱 모듈(179), 카메라 모듈(180), 전력 관리 모듈(188), 배터리(189), 통신 모듈(190), 가입자 식별 모듈(196), 또는 안테나 모듈(197)을 구동하기 위한 하나 이상의 드라이버 프로그램들을 포함할 수 있다.
- [50] 미들웨어(144)는 전자 장치(101)의 하나 이상의 리소스들로부터 제공되는 기능 또는 정보가 어플리케이션(146)에 의해 사용될 수 있도록 다양한 기능들을 어플리케이션(146)으로 제공할 수 있다. 미들웨어(144)는, 예를 들면, 어플리케이션 매니저(201), 윈도우 매니저(203), 멀티미디어 매니저(205), 리소스 매니저(207), 파워 매니저(209), 데이터베이스 매니저(211), 패키지 매니저(213), 커넥티비티 매니저(215), 노티피케이션 매니저(217), 로케이션 매니저(219), 그래픽 매니저(221), 시큐리티 매니저(223), 통화 매니저(225), 또는 음성 인식 매니저(227)를 포함할 수 있다.
- [51] 어플리케이션 매니저(201)는, 예를 들면, 어플리케이션(146)의 생명 주기를 관리할 수 있다. 윈도우 매니저(203)는, 예를 들면, 화면에서 사용되는 하나 이상의 GUI 자원들을 관리할 수 있다. 멀티미디어 매니저(205)는, 예를 들면, 미디어 파일들의 재생에 필요한 하나 이상의 포맷들을 파악하고, 그 중 선택된 해당하는 포맷에 맞는 코덱을 이용하여 상기 미디어 파일들 중 해당하는 미디어 파일의 인코딩 또는 디코딩을 수행할 수 있다. 리소스 매니저(207)는, 예를 들면, 어플리케이션(146)의 소스 코드 또는 메모리(130)의 메모리의 공간을 관리할 수 있다. 파워 매니저(209)는, 예를 들면, 배터리(189)의 용량, 온도 또는 전원을 관리하고, 이 중 해당 정보를 이용하여 전자 장치(101)의 동작에 필요한 관련 정보를 결정 또는 제공할 수 있다. 일 실시예에 있어서, 파워 매니저(209)는 전자 장치(101)의 바이오스(BIOS: basic input/output system)와 연동할 수 있다.
- [52] 데이터베이스 매니저(211)는, 예를 들면, 어플리케이션(146)에 의해 사용될 데이터베이스를 생성, 검색, 또는 변경할 수 있다. 패키지 매니저(213)는, 예를 들면, 패키지 파일의 형태로 배포되는 어플리케이션의 설치 또는 갱신을 관리할 수 있다. 커넥티비티 매니저(215)는, 예를 들면, 전자 장치(101)와 외부 전자 장치 간의 무선 연결 또는 직접 연결을 관리할 수 있다. 노티피케이션 매니저(217)는, 예를 들면, 지정된 이벤트(예: 착신 통화, 메시지, 또는 알람)의 발생을 사용자에게 알리기 위한 기능을 제공할 수 있다. 로케이션 매니저(219)는, 예를 들면, 전자 장치(101)의 위치 정보를 관리할 수 있다. 그래픽 매니저(221)는, 예를 들면, 사용자에게 제공될 하나 이상의 그래픽 효과들 또는 이와 관련된 사용자 인터페이스를 관리할 수 있다.

- [53] 시큐리티 매니저(223)는, 예를 들면, 시스템 보안 또는 사용자 인증을 제공할 수 있다. 통화(telephony) 매니저(225)는, 예를 들면, 전자 장치(101)에 의해 제공되는 음성 통화 기능 또는 영상 통화 기능을 관리할 수 있다. 음성 인식 매니저(227)는, 예를 들면, 사용자의 음성 데이터를 서버(108)로 전송하고, 그 음성 데이터에 적어도 일부 기반하여 전자 장치(101)에서 수행될 기능에 대응하는 명령어(command), 또는 그 음성 데이터에 적어도 일부 기반하여 변환된 문자 데이터를 서버(108)로부터 수신할 수 있다. 일 실시예에 따르면, 미들웨어(144)는 동적으로 기존의 구성요소를 일부 삭제하거나 새로운 구성요소들을 추가할 수 있다. 일 실시예에 따르면, 미들웨어(144)의 적어도 일부는 운영 체제(142)의 일부로 포함되거나, 또는 운영 체제(142)와는 다른 별도의 소프트웨어로 구현될 수 있다.
- [54] 어플리케이션(146)은, 예를 들면, 홈(251), 다이얼러(253), SMS/MMS(255), IM(instant message)(257), 브라우저(259), 카메라(261), 알람(263), 연락처(265), 음성 인식(267), 이메일(269), 달력(271), 미디어 플레이어(273), 앨범(275), 와치(277), 헬스(279)(예: 운동량 또는 혈당과 같은 생체 정보를 측정), 또는 환경 정보(281)(예: 기압, 습도, 또는 온도 정보 측정) 어플리케이션을 포함할 수 있다. 일 실시예에 있어서, 어플리케이션(146)은 전자 장치(101)와 외부 전자 장치 사이의 정보 교환을 지원할 수 있는 정보 교환 어플리케이션을 더 포함할 수 있다. 정보 교환 어플리케이션은, 예를 들면, 외부 전자 장치로 지정된 정보(예: 통화, 메시지, 또는 알람)를 전달하도록 설정된 noti피케이션 릴레이 어플리케이션, 또는 외부 전자 장치를 관리하도록 설정된 장치 관리 어플리케이션을 포함할 수 있다. noti피케이션 릴레이 어플리케이션은, 예를 들면, 전자 장치(101)의 다른 어플리케이션(예: 이메일 어플리케이션(269))에서 발생된 지정된 이벤트(예: 메일 수신)에 대응하는 알림 정보를 외부 전자 장치로 전달할 수 있다. 추가적으로 또는 대체적으로, noti피케이션 릴레이 어플리케이션은 외부 전자 장치로부터 알림 정보를 수신하여 전자 장치(101)의 사용자에게 제공할 수 있다.
- [55] 장치 관리 어플리케이션은, 예를 들면, 전자 장치(101)와 통신하는 외부 전자 장치 또는 그 일부 구성 요소(예: 외부 전자장치의 디스플레이 모듈 또는 카메라 모듈)의 전원(예: 턴-온 또는 턴-오프) 또는 기능(예: 밝기, 해상도, 또는 포커스)을 제어할 수 있다. 장치 관리 어플리케이션은, 추가적으로 또는 대체적으로, 외부 전자 장치에서 동작하는 어플리케이션의 설치, 삭제, 또는 갱신을 지원할 수 있다.
- [56] 본 문서에 개시된 다양한 실시예들에 따른 전자 장치는 다양한 형태의 장치가 될 수 있다. 전자 장치는, 예를 들면, 휴대용 통신 장치(예: 스마트폰), 컴퓨터 장치, 휴대용 멀티미디어 장치, 휴대용 의료 기기, 카메라, 웨어러블 장치, 또는 가전 장치를 포함할 수 있다. 본 문서의 실시예에 따른 전자 장치는 전술한 기기들에 한정되지 않는다.
- [57] 본 문서의 다양한 실시예들 및 이에 사용된 용어들은 본 문서에 기재된 기술적

특징들을 특정한 실시예들로 한정하려는 것이 아니며, 해당 실시예의 다양한 변경, 균등물, 또는 대체물을 포함하는 것으로 이해되어야 한다. 도면의 설명과 관련하여, 유사한 또는 관련된 구성요소에 대해서는 유사한 참조 부호가 사용될 수 있다. 아이টে에 대응하는 명사의 단수 형은 관련된 문맥상 명백하게 다르게 지시하지 않는 한, 상기 아이টে 한 개 또는 복수 개를 포함할 수 있다. 본 문서에서, "A 또는 B", "A 및 B 중 적어도 하나", "A 또는 B 중 적어도 하나", "A, B 또는 C", "A, B 및 C 중 적어도 하나", 및 "A, B, 또는 C 중 적어도 하나"와 같은 문구들 각각은 그 문구들 중 해당하는 문구에 함께 나열된 항목들 중 어느 하나, 또는 그들의 모든 가능한 조합을 포함할 수 있다. "제1", "제2", 또는 "첫째" 또는 "둘째"와 같은 용어들은 단순히 해당 구성요소를 다른 해당 구성요소와 구분하기 위해 사용될 수 있으며, 해당 구성요소들을 다른 측면(예: 중요성 또는 순서)에서 한정하지 않는다. 어떤(예: 제1) 구성요소가 다른(예: 제2) 구성요소에, "기능적으로" 또는 "통신적으로"라는 용어와 함께 또는 이런 용어 없이, "커플드" 또는 "커넥티드"라고 언급된 경우, 그것은 상기 어떤 구성요소가 상기 다른 구성요소에 직접적으로(예: 유선으로), 무선으로, 또는 제3 구성요소를 통하여 연결될 수 있다는 것을 의미한다.

- [58] 본 문서의 다양한 실시예들에서 사용된 용어 "모듈"은 하드웨어, 소프트웨어 또는 펌웨어로 구현된 유닛을 포함할 수 있으며, 예를 들면, 로직, 논리 블록, 부품, 또는 회로와 같은 용어와 상호 호환적으로 사용될 수 있다. 모듈은, 일체로 구성된 부품 또는 하나 또는 그 이상의 기능을 수행하는, 상기 부품의 최소 단위 또는 그 일부가 될 수 있다. 예를 들면, 일 실시예에 따르면, 모듈은 ASIC(application-specific integrated circuit)의 형태로 구현될 수 있다.
- [59] 본 문서의 다양한 실시예들은 기기(machine)(예: 도 1의 전자 장치(101))의 해 읽을 수 있는 저장 매체(storage medium)(예: 내장 메모리(136) 또는 외장 메모리(138))에 저장된 하나 이상의 명령어들을 포함하는 소프트웨어(예: 프로그램(140))로서 구현될 수 있다. 예를 들면, 기기(예: 전자 장치(101))의 프로세서(예: 프로세서(120))는, 저장 매체로부터 저장된 하나 이상의 명령어들 중 적어도 하나의 명령어를 호출하고, 그것을 실행할 수 있다. 이것은 기기가 상기 호출된 적어도 하나의 명령어에 따라 적어도 하나의 기능을 수행하도록 운영되는 것을 가능하게 한다. 상기 하나 이상의 명령어들은 컴파일러에 의해 생성된 코드 또는 인터프리터에 의해 실행될 수 있는 코드를 포함할 수 있다. 기기로 읽을 수 있는 저장 매체는, 비일시적(non-transitory) 저장 매체의 형태로 제공될 수 있다. 여기서, '비일시적'은 저장 매체가 실재(tangible)하는 장치이고, 신호(signal)(예: 전자기파)를 포함하지 않는다는 것을 의미할 뿐이며, 이 용어는 데이터가 저장 매체에 반영구적으로 저장되는 경우와 일시적으로 저장되는 경우를 구분하지 않는다.
- [60] 일 실시예에 따르면, 본 문서에 개시된 다양한 실시예들에 따른 방법은 컴퓨터 프로그램 제품(computer program product)에 포함되어 제공될 수 있다. 컴퓨터

프로그램 제품은 상품으로서 판매자 및 구매자 간에 거래될 수 있다. 컴퓨터 프로그램 제품은 기기로 읽을 수 있는 저장 매체(예: compact disc read only memory(CD-ROM))의 형태로 배포되거나, 또는 어플리케이션 스토어(예: 플레이 스토어™)를 통해 또는 두 개의 사용자 장치들(예: 스마트폰들) 간에 직접, 온라인으로 배포(예: 다운로드 또는 업로드)될 수 있다. 온라인 배포의 경우에, 컴퓨터 프로그램 제품의 적어도 일부는 제조사의 서버, 어플리케이션 스토어의 서버, 또는 중계 서버의 메모리와 같은 기기로 읽을 수 있는 저장 매체에 적어도 일시 저장되거나, 임시적으로 생성될 수 있다.

- [61] 다양한 실시예들에 따르면, 상기 기술한 구성요소들의 각각의 구성요소(예: 모듈 또는 프로그램)는 단수 또는 복수의 개체를 포함할 수 있으며, 복수의 개체 중 일부는 다른 구성요소에 분리 배치될 수도 있다. 다양한 실시예들에 따르면, 전술한 해당 구성요소들 중 하나 이상의 구성요소들 또는 동작들이 생략되거나, 또는 하나 이상의 다른 구성요소들 또는 동작들이 추가될 수 있다. 대체적으로 또는 추가적으로, 복수의 구성요소들(예: 모듈 또는 프로그램)은 하나의 구성요소로 통합될 수 있다. 이런 경우, 통합된 구성요소는 상기 복수의 구성요소들 각각의 구성요소의 하나 이상의 기능들을 상기 통합 이전에 상기 복수의 구성요소들 중 해당 구성요소에 의해 수행되는 것과 동일 또는 유사하게 수행할 수 있다. 다양한 실시예들에 따르면, 모듈, 프로그램 또는 다른 구성요소에 의해 수행되는 동작들은 순차적으로, 병렬적으로, 반복적으로, 또는 휴리스틱하게 실행되거나, 상기 동작들 중 하나 이상이 다른 순서로 실행되거나, 생략되거나, 또는 하나 이상의 다른 동작들이 추가될 수 있다.

[62]

- [63] 도 3은 일 실시예에 따른, 전자 장치의 일 예시를 설명하기 위한 블록도이다.

- [64] 일 실시예에 있어서, 도 3의 전자 장치(300)(예: 도 1의 전자 장치(101))는 메모리(310)(예: 도 1의 메모리(130)), 프로세서(320)(예: 도 1의 프로세서(120)), 및 디스플레이(330)(예: 도 1의 디스플레이 모듈(160))를 포함할 수 있다.

- [65] 일 실시예에 있어서, 메모리(310)는 하나 이상의 어플리케이션을 저장할 수 있다. 일례로, 메모리(310)는 하나 이상의 서드 파티(third party) 어플리케이션을 저장할 수 있고, 하나 이상의 시스템 어플리케이션을 저장할 수 있다. 시스템 어플리케이션은 전자 장치(300)의 제조사에 의해 전자 장치(300)에 탑재된 어플리케이션을 나타낼 수 있다. 시스템 어플리케이션은, 예를 들어, 전자 장치(300)의 관리 또는 케어를 위한 디바이스 케어 어플리케이션을 포함할 수 있으나 이에 제한되지 않는다.

- [66] 일 실시예에 있어서, 프로세서(320)는 실행 중인 제1 어플리케이션에 오류가 발생한 경우 오류 정보를 획득할 수 있다. 일 실시예에 있어서, 오류로 인해 제1 어플리케이션의 크래시가 발생하여 실행 중인(또는 사용 중인) 제1 어플리케이션이 강제 종료(FC: force close)될 수 있다. 프로세서(320)는 제1 어플리케이션의 크래시가 발생하여 제1 어플리케이션이 강제 종료된 경우, 운영

체제(예: 도 2의 운영 체제(142))의 프레임워크(framework)를 통해 오류 정보를 획득할 수 있다. 오류 정보는, 예를 들어, 제1 어플리케이션의 프로세스(process), 제1 어플리케이션의 process의 ID(identification), 제1 어플리케이션의 정보, 및 크래시 정보를 포함할 수 있다. 제1 어플리케이션의 정보는, 예를 들어, 제1 어플리케이션의 ID(예: 제1 어플리케이션의 패키지 이름), 제1 어플리케이션의 버전 정보를 포함할 수 있으나 이에 제한되지 않는다. 크래시 정보는, 예를 들어, 발생한 오류(예: 예외(exception))의 타입과 오류의 원인을 포함할 수 있으나, 이에 제한되지 않는다. exception은, 예를 들어, 사용자의 잘못된 조작에 의해 발생할 수 있는 오류 또는 개발자에 의해 구현된 로직에서 발생할 수 있는 오류를 나타낼 수 있다. 아래 표 1은 크래시 정보의 일례를 보여준다.

[67] [표1]

```
11-22 23:36:40.280 10360 14073 14073 E AndroidRuntime: FATAL EXCEPTION:
main
11-22 23:36:40.280 10360 14073 14073 E AndroidRuntime: Process: com.???katana,
PID: 14073
11-22 23:36:40.280 10360 14073 14073 E AndroidRuntime:
android.util.AndroidRuntimeException: java.lang.reflect.InvocationTargetException
.....
11-22 23:36:40.280 10360 14073 14073 E AndroidRuntime: Caused by:
java.lang.reflect.InvocationTargetException
.....
11-22 23:36:40.280 10360 14073 14073 E AndroidRuntime: Caused by:
java.lang.RuntimeException: Package not found: com.google.android.webview
.....
11-22 23:36:40.280 10360 14073 14073 E AndroidRuntime: ... 27 more
```

[68] 일 실시예 있어서, 프로세서(320)는 획득된 오류 정보에서 제1 어플리케이션에 발생한 오류의 타입(예: exception 타입)과 원인을 파악할 수 있다. 일 실시예 있어서, 프로세서(320)는 크래시 정보에서 어떤 타입의 exception이 발생하였는지를 파악할 수 있고, 오류의 발생 원인을 파악할 수 있다. 일례로, 프로세서(320)는 위 표 1의 크래시 정보의 "java.lang"를 포함하는 줄(line) "11-22 23:36:40.280 10360 14073 14073 E AndroidRuntime: Caused by: java.lang.reflect.InvocationTargetException"에서 "java.lang" 이후에 기재된 "InvocationTargetException"를 추출 또는 파싱할 수 있다. 프로세서(320)는 위 표 1의 크래시 정보로부터 "InvocationTargetException"이 발생하였음을 파악할 수 있다. 프로세서(320)는 위 표 1의 크래시 정보의 "Caused by"를 포함하는 line "11-22 23:36:40.280 10360 14073 14073 E AndroidRuntime: Caused by: java.lang.RuntimeException: Package not found: com.google.android.webview"에서

"Caused by" 이후에 기재된 "Package not found: com.google.android.webview"를 추출 또는 파싱할 수 있다.

- [69] 일 실시예 있어서, 프로세서(320)는 제1 어플리케이션의 정보 또는 크래시 정보로부터 제1 어플리케이션의 process와 제1 어플리케이션의 process의 ID를 획득할 수 있다. 일례로, 프로세서(320)는 위 표 1의 크래시 정보의 "Process"를 포함하는 line "11-22 23:36:40.280 10360 14073 14073 E AndroidRuntime: Process: com.???katana, PID: 14073"에서 "Process: com.???katana"와 "PID: 14073"를 추출 또는 파싱할 수 있다. 추출된 "Process: com.???katana"는 제1 어플리케이션의 process를 나타낼 수 있고, 추출된 "PID: 14073"는 제1 어플리케이션의 process의 ID를 나타낼 수 있다.
- [70] 일 실시예 있어서, 프로세서(320)는 오류 정보를 분석하여 제1 어플리케이션의 오류가 제1 어플리케이션이 웹뷰(webview)를 찾을 수 없어 발생한 것으로 파악할 수 있다.
- [71] 일 실시예 있어서, 프로세서(320)는 파악된 오류의 타입 및 원인을 이용하여 참조 데이터 셋으로부터 제1 어플리케이션에 발생한 오류를 해결하기 위한 해결 방법을 획득할 수 있다. 참조 데이터 셋은, 예를 들어, 하나 이상의 오류 타입(예: 하나 이상의 exception 타입), 하나 이상의 오류 발생 원인, 및 하나 이상의 해결 방법을 포함할 수 있다. 하나 이상의 해결 방법은, 예를 들어, 권한 허용(예: 파일 읽기 쓰기 권한 허용, 인터넷 사용 권한 허용), 캐시 정리, 메모리 저장 공간 확보, 재부팅, 및 인터넷 연결 중 하나 이상을 포함할 수 있으나 이에 제한되지 않는다. 하나 이상의 오류 발생 원인(또는 하나 이상의 오류 타입)은 하나 이상의 해결 방법과 맵핑되어 있을 수 있다. 일 실시예 있어서, 프로세서(320)는 크래시 정보(예: 위 표 1의 크래시 정보)로부터 추출(또는 파싱)한 exception 타입(오류 타입)(예: InvocationTargetException)이 참조 데이터 셋에 있는지 여부를 확인 또는 결정할 수 있다. 프로세서(320)는 추출한 exception 타입이 참조 데이터 셋에 있는 경우, 추출한 원인(예: "Package not found: com.google.android.webview")에 해당하는 오류 발생 원인이 참조 데이터 셋에 있는지 여부를 확인 또는 결정할 수 있다. 프로세서(320)는 추출한 원인에 해당하는 오류 발생 원인이 참조 데이터 셋에 있는 경우, 추출한 원인과 맵핑되어 있는 해결 방법을 참조 데이터 셋으로부터 획득할 수 있다.
- [72] 일 실시예 있어서, 참조 데이터 셋은 메모리(310) 또는 서버에 저장되어 있을 수 있다. 일 실시예 있어서, 참조 데이터 셋은 전자 장치(300)의 시스템 어플리케이션(예: 디바이스 케어 어플리케이션)의 데이터베이스(DB: database)에 기록되어 있을 수 있다. 다른 실시 예에 있어서, 참조 데이터 셋은 서버의 DB에 기록되어 있을 수 있다.
- [73] 일 실시예 있어서, 크래시 정보에는 오류의 발생 원인이 기록되어 있지 않을 수 있다. 일례로, 어떤 크래시 정보에는 "Caused by"를 포함하는 line이 없을 수 있다. 프로세서(320)는 크래시 정보에서 오류 타입(예: exception type)을 파악할 수

있고, 파악된 오류 타입을 이용하여 참조 데이터 셋으로부터 하나 이상의 해결 방법을 획득할 수 있다.

- [74] 일 실시예에 있어서, 프로세서(320)는 획득된 해결 방법의 적합성을 검증할 수 있다. 일 실시예에 있어서, 프로세서(320)는 획득된 해결 방법의 적합성을 전자 장치(300)의 상태 정보(예: 전자 장치(300)의 메모리 상태, 통신 연결 상태)를 기초로 검증할 수 있다. 일례로, 프로세서(320)는 참조 데이터 셋으로부터 메모리(310)의 저장 공간의 확보를 나타내는 해결 방법을 획득할 수 있다. 프로세서(320)는 메모리(310)의 저장 용량에 여유가 있는 상태이면, 메모리(310)의 저장 공간의 확보를 나타내는 해결 방법은 적합하지 않은 것으로 결정할 수 있다. 프로세서(320)는 메모리(310)의 저장 용량에 여유가 없는 상태이면, 메모리(310)의 저장 공간의 확보를 나타내는 해결 방법은 적합한 것으로 결정할 수 있다.
- [75] 일 실시예에 있어서, 프로세서(320)는 획득된 해결 방법이 적합한 것으로 검증된 경우, 획득된 해결 방법에 대한 안내를 디스플레이(330)를 통해 사용자에게 제공할 수 있다. 일 실시예에 있어서, 프로세서(320)는 획득된 해결 방법에 대한 안내를 위한 팝업 창을 디스플레이(330)에 표시할 수 있다.
- [76] 일 실시예에 있어서, 프로세서(320)는 참조 데이터 셋으로부터 제1 어플리케이션의 오류를 해결하기 위한 복수의 해결 방법들을 획득할 수 있다. 획득된 해결 방법들은 주어진 순서(given order)를 가질 수 있고, 프로세서(320)는 획득된 해결 방법들 중 최상위 순서의 해결 방법(top-ranking solution)(주어진 순서에서 최상위 위치(highest position)의 해결 방법)을 선택할 수 있다. 프로세서(320)는 최상위 순서의 해결 방법의 적합성을 검증할 수 있고, 최상위 순서의 해결 방법이 적합한 것으로 검증된 경우 최상위 순서의 해결 방법에 대한 안내를 디스플레이(330)를 통해 사용자에게 제공할 수 있다. 최상위 순서의 해결 방법이 수행되어도 제1 어플리케이션의 오류가 해결되지 않을 수 있다. 프로세서(320)는 차상위 순서의 해결 방법(주어진 순서에서 차상위 위치(next-highest position)의 해결 방법)의 적합성을 검증할 수 있고 차상위 순서의 해결 방법이 적합한 것으로 검증된 경우 차상위 순서의 해결 방법에 대한 안내를 디스플레이(330)를 통해 사용자에게 제공할 수 있다. 다른 실시예에 있어서, 프로세서(320)는 획득된 해결 방법들의 주어진 순서를 변경할 수 있는데, 순서 변경에 대해선 도 11 내지 도 12를 통해 후술한다.
- [77] 일 실시예에 있어서, 프로세서(320)는 참조 데이터 셋으로부터 제1 어플리케이션의 오류를 해결하기 위한 복수의 해결 방법들을 획득한 경우, 획득된 해결 방법들의 리스트를 디스플레이(330)를 통해 사용자에게 제공할 수 있다. 일 실시예에 있어서, 프로세서(320)는 획득된 각 해결 방법을 요약할 수 있고, 요약된 각 해결 방법의 리스트를 디스플레이(330)에 표시하여 사용자에게 제공할 수 있다. 프로세서(320)는 사용자에게 제공된 해결 방법들(또는 디스플레이(330)에 표시된 해결 방법들) 중 사용자 선택이 있는 해결 방법을

수행할 수 있다.

- [78] 일 실시예에 있어서, 프로세서(320)는 전자 장치(300)의 사용 상태를 파악 또는 확인 또는 결정할 수 있다. 일례로, 프로세서(320)는 전자 장치(300)의 운영 체제가 최신 버전으로 업데이트된 상태인지, 사용자의 다른 전자 장치 내의 데이터가 전자 장치(300)로 백업(또는 동기화)된 상태인지, 또는 전자 장치(300)가 신제품인지를 파악 또는 확인 또는 결정할 수 있다. 프로세서(320)는 제1 어플리케이션에 오류가 발생한 경우, 파악한 사용 상태에 적합한 오류 해결 방법을 사용자에게 제공할 수 있도록 제1 추천 알고리즘(예: 협업 필터링(collaborative filtering))을 이용할 수 있다. 일례로, 프로세서(320)는 제1 어플리케이션에 오류가 발생한 경우, 파악한 사용 상태에 제1 추천 알고리즘을 적용하여 제1 어플리케이션의 오류를 해결하기 위한 하나 이상의 해결 방법을 사용자에게 제공할 수 있다.
- [79] 일 실시예에 있어서, 프로세서(320)는 복수의 어플리케이션들 각각마다 어떤 오류가 발생하였는지 및 복수의 어플리케이션들 각각에 발생한 오류를 해결하기 위해 어떤 해결 방법을 제공하였는지를 메모리(310)(또는 서버)에 기록할 수 있다. 달리 표현하면, 메모리(310)(또는 서버)에 해결 방법 제공 이력이 기록되어 있을 수 있다. 프로세서(320)는 제1 어플리케이션에 오류가 발생한 경우, 사용자에게 적합한 오류 해결 방법을 제공할 수 있도록 해결 방법 제공 이력과 제2 추천 알고리즘(예: 콘텐츠 기반(content-based) 필터링)을 이용할 수 있다. 일례로, 프로세서(320)는 해결 방법 제공 이력에 제2 추천 알고리즘을 적용하여 제1 어플리케이션의 오류를 해결하기 위한 하나 이상의 해결 방법을 사용자에게 제공할 수 있다.
- [80] 일 실시예에 있어서, 획득된 해결 방법(또는 획득된 복수의 해결 방법들)이 수행되어도 제1 어플리케이션의 오류가 해결되지 않을 수 있다. 또는, 프로세서(320)는 파악된 오류의 타입이 참조 데이터 셋에 있는지 조회하였으나 파악된 오류의 타입이 없음을 나타내는 응답을 획득할 수 있다. 프로세서(320)는 제1 어플리케이션의 오류가 해결되지 않은 경우 또는 파악된 오류의 타입이 없음을 나타내는 응답을 획득한 경우, 제1 어플리케이션의 업데이트 버전이 어플리케이션 마켓에 있는지 여부를 판단할 수 있다. 프로세서(320)는 제1 어플리케이션의 업데이트 버전이 어플리케이션 마켓에 있는 경우 제1 어플리케이션의 업데이트에 대한 안내 또는 제1 어플리케이션의 업데이트 버전에 대한 안내를 디스플레이(330)를 통해 사용자에게 제공할 수 있다. 프로세서(320)는 제1 어플리케이션의 업데이트에 대한 안내 또는 제1 어플리케이션의 업데이트 버전에 대한 안내를 위한 팝업 창을 디스플레이(330)에 표시할 수 있다. 프로세서(320)는 제1 어플리케이션의 업데이트 버전이 어플리케이션 마켓에 없는 경우, 제1 어플리케이션의 오류에 대한 정보를 디스플레이(330)를 통해 사용자에게 제공할 수 있고, 제공된 정보에 대한 사용자 입력이 있는 경우 제1 어플리케이션의 오류 발생 이력과 제1

어플리케이션의 삭제를 위한 버튼을 포함하는 화면을 디스플레이(330)에 표시할 수 있다.

- [81]
- [82] 도 4는 일 실시예에 따른, 전자 장치의 운영 체제의 프레임워크와 시스템 어플리케이션의 일 예시를 설명하기 위한 도면이다.
- [83] 일 실시예에 있어서, 프로세서(400)(예: 도 3의 프로세서(320))에서 운영 체제의 프레임워크(410)와 시스템 어플리케이션(420)(예: 도 2의 어플리케이션(146))이 동작할 수 있다.
- [84] 일 실시예에 있어서, 프레임워크(410)는 AMS(ActivityManagerService)(411)(예: 도 2의 어플리케이션 매니저(201)), AppErrors(412), 진단부(413), 및 DB 매니저(414)(예: 도 2의 데이터베이스 매니저(211))를 포함할 수 있다.
- [85] 일 실시예에 있어서, 시스템 어플리케이션(420)은 백그라운드에서 동작하는 IAFD(intelligent application fault diagnosis) 서비스 컴포넌트(421), 팝업 창을 제공하기 위한 팝업 제공부(422), 시스템 어플리케이션(420)의 외부 요소(예: 프레임워크(410))와 제1 DB(424) 사이의 연결을 제공하기 위한 DB 제공자(provider)(423), 및 제1 DB(424)를 포함할 수 있다.
- [86] 일 실시예에 있어서, 도 3을 통해 설명한 참조 데이터 셋은 제1 DB(424)에 기록되어 있을 수 있다. 제1 DB(424)에는 하나 이상의 어플리케이션의 버전 정보가 기록되어 있을 수 있다.
- [87] 일 실시예에 있어서, 프로세서(400)는 프레임워크(410)를 통해 시스템 어플리케이션(420) 내의 제1 DB(424)에 제1 어플리케이션의 오류를 해결하기 위한 해결 방법이 있는지 조회할 수 있다. 프로세서(400)는 제1 DB(424)로부터 제1 어플리케이션의 오류를 해결하기 위한 하나 이상의 해결 방법을 획득할 수 있다.
- [88] 일 실시예에 있어서, 프로세서(400)는 제1 어플리케이션의 크래시가 발생하여 제1 어플리케이션이 강제 종료된 경우 프레임워크(410)의 AMS(411)를 통해 오류 정보를 획득 또는 수집할 수 있다.
- [89] 일 실시예에 있어서, 프로세서(400)는 AppErrors(412)를 통해 진단부(413)를 동작시킬 수 있다. 일례로, 진단부(413)는 인스턴스에 해당할 수 있고, 프로세서(400)는 AppErrors(412)를 통해 진단부(413)에 해당하는 인스턴스를 생성할 수 있다.
- [90] 일 실시예에 있어서, 프로세서(400)는 진단부(413)를 통해 오류 정보 내의 크래시 정보(예: 위 표 1의 크래시 정보)로부터, 발생한 exception 타입과 오류의 원인을 파악할 수 있다.
- [91] 일 실시예에 있어서, 프로세서(400)는 IAFD 서비스 컴포넌트(421)를 통해 시스템 어플리케이션(420)의 일부 동작을 백그라운드(background)에서 수행할 수 있다. 백그라운드에서 수행되는 일부 동작은, 예를 들어, 후술할 제1 DB(424) 접근 동작, 제1 DB(424)로부터 해결 방법을 획득하는 동작을 포함할 수 있으나 이에

제한되지 않는다.

- [92] 일 실시예에 있어서, 프로세서(400)는 프레임워크(410) 내의 DB 매니저(414)와 시스템 어플리케이션(420) 내의 DB 제공자(423)를 통해 제1 DB(424)에 접근하여, 파악된 exception 타입과 오류의 원인에 대한 해결 방법이 제1 DB(424)에 있는지 조회할 수 있다. 프로세서(400)는 파악된 exception 타입과 오류의 원인에 대한 해결 방법이 제1 DB(424)에 있는 경우, DB 제공자(423)와 DB 매니저(414)를 통해 제1 DB(424)로부터 해결 방법을 획득할 수 있다.
- [93] 일 실시예에 있어서, 프로세서(400)는 획득된 해결 방법의 적합성을 진단부(413)를 통해 검증할 수 있고, 획득된 해결 방법이 적합한 것으로 검증된 경우 획득된 해결 방법의 안내를 위한 팝업 창을 시스템 어플리케이션(420)의 팝업 제공부(422)를 통해 디스플레이(330)에 표시할 수 있다.
- [94] 일 실시예에 있어서, 제1 DB(424)에는 획득된 해결 방법의 제공 횟수가 기록되어 있을 수 있다. 프로세서(400)는 획득된 해결 방법을 사용자에게 제공한 경우, 제1 DB(424)에 기록되어 있는 제공 횟수를 업데이트할 수 있다. 일례로, 제1 DB(424)에 해결 방법 A의 제공 횟수 10회가 기록되어 있을 수 있다. 프로세서(400)는 해결 방법 A를 사용자에게 제공한 경우 해결 방법 A의 제공 횟수를 10회에서 11회로 업데이트할 수 있다.
- [95] 일 실시예에 있어서, 프로세서(400)는 획득된 해결 방법을 사용자에게 제공한 경우, 획득된 해결 방법을 제1 어플리케이션의 현재 버전에서 제공한 것을 제1 DB(424)에 기록할 수 있다. 일례로, 오류가 발생한 제1 어플리케이션의 버전이 1.10이고, 프로세서(400)는 해결 방법 A를 사용자에게 제공할 수 있다. 프로세서(400)는 제1 어플리케이션의 버전 1.10에서 해결 방법 A를 제공한 것을 제1 DB(424)에 기록할 수 있다.
- [96] 일 실시예에 있어서, 프로세서(400)는 제1 DB(424)로부터 해결 방법들을 획득할 수 있고, 획득된 해결 방법들의 리스트를 디스플레이(330)를 통해 사용자에게 제공할 수 있다. 일 실시예에 있어서, 프로세서(400)는 획득된 각 해결 방법을 요약할 수 있고, 요약된 각 해결 방법의 리스트를 디스플레이(330)에 표시하여 사용자에게 제공할 수 있다. 프로세서(400)는 사용자에게 제공된 해결 방법들(또는 디스플레이(330)에 표시된 해결 방법들) 중 사용자 선택이 있는 해결 방법을 수행할 수 있다.
- [97] 일 실시예에 있어서, 프로세서(400)는 제1 DB(424)로부터 조회 결과가 없음을 나타내는 응답을 획득할 수 있다. 일례로, 프로세서(400)는 파악한 exception 타입이 제1 DB(424)에 없는 경우 제1 DB(424)로부터 조회 결과가 없음을 나타내는 응답을 획득할 수 있다. 프로세서(400)는 어플리케이션 마켓에 제1 어플리케이션의 업데이트 버전이 있는지 여부를 확인 또는 결정할 수 있다. 프로세서(400)는 제1 어플리케이션의 업데이트 버전이 어플리케이션 마켓에 있는 경우 제1 어플리케이션의 업데이트를 안내 또는 제1 어플리케이션의 업데이트 버전의 안내를 위한 팝업 창을 디스플레이(330)에 표시할 수 있다.

프로세서(400)는 제1 어플리케이션의 업데이트 버전이 어플리케이션 마켓에 없는 경우 제1 어플리케이션의 오류에 대한 정보를 디스플레이(330)에 표시하여 사용자에게 제공할 수 있다. 프로세서(400)는 제공된 정보에 대한 사용자 입력이 있는 경우, 제1 어플리케이션의 오류 발생 이력과 제1 어플리케이션의 삭제를 위한 버튼을 포함하는 화면을 디스플레이(330)에 표시할 수 있다.

- [98] 일 실시예에 있어서, 프로세서(400)는 제1 어플리케이션의 오류가 발생한 제2 시점(시간 인스턴스(time instance))과 제1 시점(time instance) 사이의 시간 구간(time period) 동안의 전자 장치(300)의 상태 정보를 분석할 수 있다. 제1 시점(time instance)은 제1 어플리케이션의 오류가 발생한 제2 시점(time instance)보다 빠른 시점(time instance)을 나타낼 수 있다. 전자 장치(300)의 상태 정보는, 예를 들어, 전자 장치(300)의 부하 상태, 전자 장치(300)의 발열 상태, 및 전자 장치(300)와 기지국(미도시) 사이의 전계(electromagnetic field) 상태 중 하나 이상을 포함할 수 있으나 이에 제한되지 않는다. 제1 어플리케이션의 오류가 외부 요인(또는 전자 장치(300)의 상태)에 의해 발생할 가능성이 존재할 수 있어, 프로세서(400)는 진단부(413)를 통해 제1 어플리케이션의 오류가 발생한 제2 시점(time instance)과 제1 시점(time instance) 사이의 시간 구간 동안의 전자 장치(300)의 상태 정보를 분석할 수 있다. 일례로, 전자 장치(300)에 다수의 어플리케이션들이 실행 중일 수 있어 프로세서(400)는 높은 부하 상태에 있을 수 있고, 프로세서(400)의 높은 부하 상태로 인해 제1 어플리케이션에 일시적인 오류가 발생할 수 있다. 다른 일례로, 전자 장치(300)의 발열(또는 높은 온도)로 인해 프로세서(400)의 성능이 일시적으로 저하될 수 있고, 프로세서(400)의 성능의 일시적인 저하에 의해 제1 어플리케이션의 오류가 발생할 수 있다. 또 다른 일례로, 전자 장치(300)와 기지국 사이의 전계 상태가 약전계(weak electromagnetic field) 상태로 될 수 있다. 전자 장치(300)가 예를 들어 교량 구간 또는 터널 구간에 진입하는 경우, 전자 장치(300)와 기지국 사이의 전계 상태는 약전계 상태가 될 수 있다. 약전계 상태에 의해 전자 장치(300)의 통신 오류 및/또는 프로세서(400)의 입출력 오류가 발생할 수 있고, 제1 어플리케이션의 오류가 발생할 수 있다.

- [99] 일 실시예에 있어서, 프로세서(400)는 제1 어플리케이션의 오류가 발생한 제2 시점(time instance)과 제1 시점(time instance) 사이의 시간 구간 동안의 전자 장치(300)의 상태 정보를 분석함으로써 전자 장치(300)의 부하 상태, 발열 상태, 및 전계 상태 중 하나 이상과 같은 전자 장치(300)의 상태를 체크 또는 결정할 수 있다. 프로세서(400)는 전자 장치(300)가 높은 부하 상태에 있는 경우, 전자 장치(300)의 온도가 높은 경우, 및/또는 전자 장치(300)가 약전계 상태에 있는 경우, 전자 장치(300)가 비정상 상태에 있는 것으로 분석 또는 판단할 수 있다.

- [100] 일 실시예에 있어서, 프로세서(400)는 진단부(413)를 통해 크래시 정보로부터 exception 타입과 오류의 원인을 파악 또는 결정할 수 있고, 파악한 exception 타입, 파악한 오류의 원인, 및 전자 장치(300)의 상태 정보의 분석 결과를

이용하여 제1 어플리케이션에 발생한 오류를 해결하기 위한 하나 이상의 해결 방법 또는 제1 어플리케이션에 발생한 오류에 대응되는 하나 이상의 해결 방법을 획득할 수 있다. 일례로, 프로세서(400)는 파악한 exception 타입 및 파악한 오류의 원인에 대한 하나 이상의 해결 방법을 제1 DB(424)로부터 획득할 수 있다. 프로세서(400)는 전자 장치(300)의 비정상 상태를 해소(또는 완화)하기 위한 방법(예: 여러 어플리케이션들이 실행 중인 경우 일부 어플리케이션의 실행을 중단, 온도를 낮추기 위해 전자 장치(300)의 사용을 잠시 중단)을 생성할 수 있다. 어떤 실시 예에 있어서, 제1 DB(424)에 전자 장치(300)의 비정상 상태를 해소(또는 완화)하기 위한 하나 이상의 방법이 기록되어 있을 수 있다. 프로세서(400)는 전자 장치(300)가 비정상 상태에 있는 경우, 제1 DB(424)로부터 비정상 상태를 해소(또는 완화)하기 위한 하나 이상의 방법을 획득할 수 있다.

- [101] 일 실시예에 있어서, 시스템 어플리케이션(420)의 업데이트를 통해 제1 DB(424)가 업데이트될 수 있다.
- [102]
- [103] 도 5는 일 실시예에 따른, 전자 장치가 DB로부터 해결 방법을 획득하는 다른 예시를 설명하기 위한 도면이다.
- [104] 일 실시예에 있어서, 도 5의 전자 장치(500)(예: 도 3의 전자 장치(300)) 내의 프로세서(510)(예: 도 3의 프로세서(320))는 제1 어플리케이션의 크래시가 발생하여 제1 어플리케이션이 강제 종료된 경우, 프레임워크(511)(예: 도 4의 프레임워크(410))를 통해 오류 정보를 획득할 수 있다. 프로세서(510)는 프레임워크(511)를 통해 획득된 오류 정보로부터 exception 타입과 오류의 원인을 파악할 수 있다. 도 5에 도시된 예에서, 프로세서(510)는 파악한 exception 타입과 오류의 원인에 대한 해결 방법을 획득하기 위해 시스템 어플리케이션(예: 도 4의 시스템 어플리케이션(420))을 이용하지 않을 수 있다.
- [105] 일 실시예에 있어서, 메모리(520)(예: 도 3의 메모리(310))에 도 3을 통해 설명한 참조 데이터 셋이 저장되어 있을 수 있다. 도 5에 도시된 제1 DB(521)에 참조 데이터 셋이 기록되어 있을 수 있다.
- [106] 일 실시예에 있어서, 프로세서(510)는 파악한 exception 타입과 오류의 원인에 대한 해결 방법이 제1 DB(521)에 있는지 조회할 수 있다. 프로세서(510)는 파악한 exception 타입과 오류의 원인에 대한 해결 방법이 제1 DB(521)에 있는 경우, 제1 DB(521)로부터 하나 이상의 해결 방법을 획득할 수 있다.
- [107] 일 실시예에 있어서, 프로세서(510)는 획득된 해결 방법의 적합성을 검증할 수 있다. 프로세서(510)는 획득된 해결 방법이 적합한 것으로 검증된 경우 획득한 해결 방법의 안내를 위한 팝업 창을 프레임워크(511)를 통해 디스플레이(330)에 표시할 수 있다.
- [108] 일 실시예에 있어서, 제1 DB(521)에는 획득된 해결 방법의 제공 횟수가 기록되어 있을 수 있다. 프로세서(510)는 획득된 해결 방법을 사용자에게 제공한 경우, 제1 DB(521)에 기록되어 있는 제공 횟수를 업데이트할 수 있다. 일례로, 제1

DB(521)에 해결 방법 A의 제공 횟수 8회가 기록되어 있을 수 있다.

프로세서(510)는 해결 방법 A를 사용자에게 제공한 경우 해결 방법 A의 제공 횟수를 8회에서 9회로 업데이트할 수 있다.

- [109] 일 실시예에 있어서, 프로세서(510)는 획득된 해결 방법을 사용자에게 제공한 경우, 획득된 해결 방법을 제1 어플리케이션의 현재 버전에서 제공한 것을 제1 DB(521)에 기록할 수 있다. 일례로, 오류가 발생한 제1 어플리케이션의 버전이 2.10이고, 프로세서(510)는 해결 방법 A를 사용자에게 제공할 수 있다. 프로세서(510)는 제1 어플리케이션의 버전 2.10에서 해결 방법 A를 제공한 것을 제1 DB(521)에 기록할 수 있다.
- [110] 일 실시예에 있어서, 프로세서(510)는 제1 DB(521)로부터 해결 방법들을 획득할 수 있고, 획득된 해결 방법들의 리스트를 디스플레이(330)를 통해 사용자에게 제공할 수 있다. 일 실시예에 있어서, 프로세서(510)는 획득된 각 해결 방법을 요약할 수 있고, 요약된 각 해결 방법의 리스트를 디스플레이(330)에 표시하여 사용자에게 제공할 수 있다. 프로세서(510)는 사용자에게 제공된 해결 방법들(또는 디스플레이(330)에 표시된 해결 방법들) 중 사용자 선택이 있는 해결 방법을 수행할 수 있다.
- [111] 일 실시예에 있어서, 프로세서(510)는 제1 DB(521)로부터 조회 결과 없음을 나타내는 응답을 수신할 수 있다. 프로세서(510)는 조회 결과 없음을 나타내는 응답을 수신하는 경우, 어플리케이션 마켓에 제1 어플리케이션의 업데이트 버전이 있는지 여부를 확인 또는 결정할 수 있다. 프로세서(510)는 제1 어플리케이션의 업데이트 버전이 어플리케이션 마켓에 있는 경우 제1 어플리케이션의 업데이트를 안내하기 위한 팝업 창을 디스플레이(330)에 표시할 수 있다. 프로세서(510)는 제1 어플리케이션의 업데이트 버전이 어플리케이션 마켓에 없는 경우 제1 어플리케이션의 오류에 대한 정보를 디스플레이(330)에 표시하여 사용자에게 제공할 수 있다. 프로세서(510)는 제공된 정보에 대한 사용자 입력이 있는 경우, 제1 어플리케이션의 오류 발생 이력과 제1 어플리케이션의 삭제를 위한 소프트 버튼을 포함하는 화면을 디스플레이(330)에 표시할 수 있다.
- [112] 일 실시예에 있어서, 프로세서(510)는 제1 어플리케이션의 오류 발생 시점과 제1 시점 사이의 시간 동안의 전자 장치(500)의 상태 정보(예: 전자 장치(500)의 부하 상태, 전자 장치(500)의 발열 상태, 및 전자 장치(500)와 기지국 사이의 전계 상태 중 하나 이상)를 분석할 수 있다. 전자 장치(500)의 상태 정보의 분석에 대한 실시예들은 도 4를 통해 설명한 전자 장치(300)의 상태 정보의 분석에 대한 실시예들이 적용될 수 있어, 상세한 설명을 생략한다.
- [113] 일 실시예에 있어서, 프로세서(510)는 크래시 정보로부터 exception 타입과 오류의 원인을 파악할 수 있고, 파악한 exception 타입, 파악한 오류의 원인, 및 전자 장치(500)의 상태 정보의 분석 결과를 이용하여 제1 어플리케이션에 발생한 오류를 해결하기 위한 하나 이상의 해결 방법을 획득할 수 있다. 전자

장치(500)의 상태 정보의 분석 결과를 이용하여 해결 방법을 획득하는 실시 예들은 도 4를 통해 설명한 전자 장치(300)의 상태 정보의 분석 결과를 이용하여 해결 방법을 획득한 실시 예들이 적용될 수 있어, 상세한 설명을 생략한다.

- [114] 일 실시예에 있어서, 펌웨어 업데이트(또는 운영 체제 업데이트)를 통해 제1 DB(521)가 업데이트될 수 있다.
- [115]
- [116] 도 6은 일 실시 예에 따른, 전자 장치가 DB로부터 해결 방법을 획득하는 또 다른 예시를 설명하기 위한 도면이다.
- [117] 일 실시예에 있어서, 도 3을 통해 설명한 참조 데이터 셋은 서버(620)에 저장될 수 있다. 도 6의 서버(620)의 제1 DB(621)에 참조 데이터 셋이 기록되어 있을 수 있다.
- [118] 일 실시예에 있어서, 전자 장치(610)(예: 도 3의 전자 장치(300)) 내의 프로세서(611)(예: 도 3의 프로세서(320))는 제1 어플리케이션의 크래시가 발생하여 제1 어플리케이션이 강제 종료된 경우, 프레임워크(611-1)(예: 도 4의 프레임워크(410))를 통해 오류 정보를 획득할 수 있다. 프로세서(611)는 프레임워크(611-1)를 통해 획득된 오류 정보로부터 exception 타입과 오류의 원인을 파악할 수 있다.
- [119] 일 실시예에 있어서, 프로세서(611)는 제1 어플리케이션의 정보(예: 제1 어플리케이션의 패키지 이름과 버전 정보), 파악한 exception 타입, 및 파악한 오류의 원인을 전자 장치(610)의 통신 모듈(예: 도 1의 통신 모듈(190))을 통해 서버(620)에 전송할 수 있다. 서버(620)는 제1 어플리케이션에 발생한 exception 타입과 오류의 원인에 대한 해결 방법이 서버(620) 내의 제1 DB(621)에 있는지 확인 또는 결정할 수 있다. 서버(620)는 제1 어플리케이션에 발생한 exception 타입과 오류의 원인에 대한 하나 이상의 해결 방법이 서버(620) 내의 제1 DB(621)에 있는 경우 하나 이상의 해결 방법을 전자 장치(610)로 전송할 수 있다.
- [120] 일 실시예에 있어서, 프로세서(611)는 수신된 해결 방법의 적합성을 검증할 수 있고, 수신된 해결 방법이 적합한 것으로 검증되는 경우 수신된 해결 방법의 안내를 위한 팝업 창을 디스플레이(330)에 표시할 수 있다.
- [121] 일 실시예에 있어서, 제1 DB(621)에는 획득된 해결 방법의 제공 횟수가 기록되어 있을 수 있다. 프로세서(611)는 획득된 해결 방법을 사용자에게 제공한 경우, 제1 DB(621)에 기록되어 있는 제공 횟수를 업데이트할 수 있다. 일례로, 제1 DB(621)에 해결 방법 A의 제공 횟수 5회가 기록되어 있을 수 있다. 프로세서(611)는 해결 방법 A를 사용자에게 제공한 경우, 서버(620)를 통해 제1 DB(621)에 접근할 수 있고, 해결 방법 A의 제공 횟수를 5회에서 6회로 업데이트할 수 있다.
- [122] 일 실시예에 있어서, 프로세서(611)는 획득된 해결 방법을 사용자에게 제공한 경우, 획득된 해결 방법을 제1 어플리케이션의 현재 버전에서 제공한 것을 제1 DB(621)에 기록할 수 있다. 일례로, 오류가 발생한 제1 어플리케이션의 버전이

2.10이고, 프로세서(611)는 해결 방법 A를 사용자에게 제공할 수 있다.

프로세서(611)는 서버(620)를 통해 제1 DB(621)에 접근할 수 있고, 제1 어플리케이션의 버전 2.10에서 해결 방법 A를 제공한 것을 제1 DB(621)에 기록할 수 있다.

- [123] 일 실시예에 있어서, 프로세서(611)는 서버(620)로부터 해결 방법들을 수신할 수 있고, 수신된 해결 방법들의 리스트를 디스플레이(330)를 통해 사용자에게 제공할 수 있다. 일 실시예에 있어서, 프로세서(611)는 수신된 각 해결 방법을 요약할 수 있고, 요약된 각 해결 방법의 리스트를 디스플레이(330)에 표시하여 사용자에게 제공할 수 있다. 프로세서(611)는 사용자에게 제공된 해결 방법들(또는 디스플레이(330)에 표시된 해결 방법들) 중 사용자 선택이 있는 해결 방법을 수행할 수 있다.
- [124] 일 실시예에 있어서, 프로세서(611)는 서버(620)로부터 조회 결과 없음을 나타내는 응답을 수신할 수 있다. 프로세서(611)는 조회 결과 없음을 나타내는 응답을 수신하는 경우, 어플리케이션 마켓에 제1 어플리케이션의 업데이트 버전이 있는지 여부를 확인 또는 결정할 수 있다. 프로세서(611)는 제1 어플리케이션의 업데이트 버전이 어플리케이션 마켓에 있는 경우 제1 어플리케이션의 업데이트를 안내하기 위한 팝업 창을 디스플레이(330)에 표시할 수 있다. 프로세서(611)는 제1 어플리케이션의 업데이트 버전이 어플리케이션 마켓에 없는 경우 제1 어플리케이션의 오류에 대한 정보를 디스플레이(330)에 표시하여 사용자에게 제공할 수 있다. 프로세서(611)는 제공된 정보에 대한 사용자 입력이 있는 경우, 제1 어플리케이션의 오류 발생 이력과 제1 어플리케이션의 삭제를 위한 소프트 버튼을 포함하는 화면을 디스플레이(330)에 표시할 수 있다.
- [125] 일 실시예에 있어서, 프로세서(611)는 제1 어플리케이션의 오류 발생 시점과 제1 시점 사이의 시간 동안의 전자 장치(610)의 상태 정보(예: 전자 장치(610)의 부하 상태, 전자 장치(610)의 발열 상태, 및 전자 장치(610)와 기지국 사이의 전계 상태 중 하나 이상)를 분석할 수 있다. 전자 장치(610)의 상태 정보의 분석에 대한 실시예들은 도 4를 통해 설명한 전자 장치(300)의 상태 정보의 분석에 대한 실시예들이 적용될 수 있어, 상세한 설명을 생략한다.
- [126] 일 실시예에 있어서, 프로세서(611)는 크래시 정보로부터 exception 타입과 오류의 원인을 파악할 수 있고, 파악한 exception 타입, 파악한 오류의 원인, 및 전자 장치(610)의 상태 정보의 분석 결과를 이용하여 제1 어플리케이션에 발생한 오류를 해결하기 위한 하나 이상의 해결 방법을 획득할 수 있다. 전자 장치(610)의 상태 정보의 분석 결과를 이용하여 해결 방법을 획득하는 실시예들은 도 4를 통해 설명한 전자 장치(300)의 상태 정보의 분석 결과를 이용하여 해결 방법을 획득한 실시예들이 적용될 수 있어, 상세한 설명을 생략한다.
- [127]
- [128] 도 7은 일 실시예에 따른, 전자 장치의 어플리케이션의 크래시가 발생한 경우

해결 방법을 제공하는 일 예시를 설명하기 위한 흐름도이다.

- [129] 동작 701에서, 전자 장치(예: 도 3의 전자 장치(300))는 제1 어플리케이션에 크래시가 발생한 경우, 오류 정보를 획득할 수 있다. 획득된 오류 정보는, 예를 들어, 제1 어플리케이션의 process, 제1 어플리케이션의 process의 ID, 제1 어플리케이션의 정보, 및 크래시 정보를 포함할 수 있다.
- [130] 동작 703에서, 전자 장치(300)는 제1 어플리케이션이 서드 파티 어플리케이션인지 여부를 확인 또는 결정할 수 있다. 일 실시예에 있어서, 전자 장치(300)는 오류 정보(예: 제1 어플리케이션의 정보)를 이용하여 제1 어플리케이션이 서드 파티 어플리케이션인지 여부를 확인 또는 결정할 수 있다.
- [131] 전자 장치(300)는 제1 어플리케이션이 서드 파티 어플리케이션이 아닌 경우(동작 703-아니오), 동작 705에서 제1 어플리케이션의 실행 중단을 사용자에게 안내할 수 있다. 어떤 실시예에 있어서, 동작 703은 생략될 수 있다.
- [132] 전자 장치(300)는 제1 어플리케이션이 서드 파티 어플리케이션인 경우(동작 703-예), 동작 707에서 오류 정보 내의 크래시 정보에서 타겟 항목(예: exception 타입, 오류의 원인, 및 제1 어플리케이션의 정보 중 적어도 하나)을 파싱할 수 있다.
- [133] 동작 709에서, 전자 장치(300)는 파싱 결과(예: exception 타입, 오류의 원인, 및 제1 어플리케이션의 정보 중 적어도 하나)를 기초로 제1 DB(예: 도 4의 제1 DB(424), 도 5의 제1 DB(521), 또는 도 6의 제1 DB(621))를 조회할 수 있다. 일 실시예에 있어서, 전자 장치(300)는 파싱한 타겟 항목에 해당하는 해결 방법이 제1 DB에 있는지 조회할 수 있다.
- [134] 동작 711에서, 전자 장치(300)는 제1 DB로부터 반환받은 정보(또는 획득한 정보)(예: 제1 어플리케이션의 오류를 해결하기 위한 해결 방법)가 있는지 여부를 확인 또는 결정할 수 있다. 일례로, 전자 장치(300)는 파싱한 exception 타입이 제1 DB에 없는 경우 제1 DB로부터 조회 결과 없음을 나타내는 정보를 수신할 수 있다. 이 경우, 전자 장치(300)는 제1 DB로부터 반환받은 정보가 없는 것으로 확인할 수 있다.
- [135] 전자 장치(300)는 제1 DB로부터 반환받은 정보가 없는 경우(동작 711-아니오), 동작 721에서 어플리케이션 마켓에 제1 어플리케이션의 업데이트 버전이 있는지 여부를 확인 또는 결정할 수 있다.
- [136] 전자 장치(300)는 어플리케이션 마켓에 제1 어플리케이션의 업데이트 버전이 있는 경우(동작 721-예), 동작 723에서 제1 어플리케이션의 업데이트 안내를 사용자에게 제공할 수 있다.
- [137] 전자 장치(300)는 어플리케이션 마켓에 제1 어플리케이션의 업데이트 버전이 없는 경우(동작 721-아니오), 동작 725에서 제1 어플리케이션의 오류에 대한 정보를 사용자에게 제공할 수 있다. 전자 장치(300)는 제1 어플리케이션의 오류 발생 이력과 제1 어플리케이션의 삭제를 위한 버튼을 포함하는 화면을 디스플레이(예: 도 3의 디스플레이(330))에 표시할 수 있다.

- [138] 전자 장치(300)는 제1 DB로부터 제1 어플리케이션(또는 제1 어플리케이션의 현재 버전)에서 발생한 오류를 해결하기 위한 복수의 해결 방법들을 획득할 수 있다. 전자 장치(300)는 제1 DB로부터 복수의 해결 방법들을 획득하여 제1 DB로부터 반환받은 정보가 있는 것으로 확인한 경우(동작 711-예), 동작 713에서, 획득된 해결 방법들 중 제공될 해결 방법을 선택할 수 있다. 일례로, 전자 장치(300)는 제1 DB로부터 제1 어플리케이션의 오류를 해결하기 위한 해결 방법 #1과 해결 방법 #2를 획득할 수 있다. 해결 방법 #1과 해결 방법 #2 중에서 해결 방법 #1이 해결 방법 #2보다 먼저 제공되는 것으로 정해져 있을 수 있다. 해결 방법 #1과 해결 방법 #2 중에서 해결 방법 #1의 우선 순위가 해결 방법 #2의 우선 순위보다 높을 수 있다. 전자 장치(300)는 해결 방법 #1과 해결 방법 #2 중에서 최상위 순서인 해결 방법 #1을 선택할 수 있다. 일 실시예에 있어서, 해결 방법들의 순서는 변경될 수 있는데 이는 도 11 내지 도 12를 통해 후술한다.
- [139] 동작 715에서, 전자 장치(300)는 선택된 해결 방법이 수행되었는지 여부를 판단할 수 있다. 일례로, 제1 DB에는 전자 장치(300)는 제1 어플리케이션의 오류를 해결하기 위해 해결 방법 #1을 사용자에게 제공한 이력이 기록되어 있지 않을 수 있다. 전자 장치(300)는 제1 DB를 참조하여, 선택된 해결 방법 #1은 수행되지 않은 것으로 판단할 수 있다(동작 715-아니오).
- [140] 전자 장치(300)는 선택된 해결 방법 #1이 수행되지 않은 경우(동작 715-아니오), 동작 717에서, 선택된 해결 방법 #1이 적합한지 여부를 판단할 수 있다. 전자 장치(300)는 전자 장치(300)의 상태 정보(예: 메모리 상태, 통신 연결 상태)를 고려하여, 선택된 해결 방법 #1이 적합한지 여부를 판단할 수 있다. 전자 장치(300)는 선택된 해결 방법 #1이 적합하지 않은 경우(동작 717-아니오), 동작 713에서, 해결 방법 #2를 선택할 수 있고 선택된 해결 방법 #2가 적합한지 여부를 판단할 수 있다.
- [141] 전자 장치(300)는 선택된 해결 방법 #1이 적합한 경우(동작 717-예), 동작 719에서, 선택된 해결 방법 #1을 사용자에게 제공할 수 있다. 전자 장치(300)는 선택된 해결 방법 #1을 사용자에게 제공한 경우, 선택된 해결 방법 #1의 제공 횟수를 제1 DB에서 업데이트할 수 있고 선택된 해결 방법 #1을 제1 어플리케이션의 현재 버전에서 제공하였음을 제1 DB에 기록할 수 있다.
- [142] 선택된 해결 방법 #1이 수행되어도 제1 어플리케이션의 크래시가 재발생할 수 있다. 제1 어플리케이션의 크래시가 재발생한 경우, 전자 장치(300)는 동작 701, 동작 703, 동작 707, 동작 709, 및 동작 713을 다시 수행할 수 있다. 전자 장치(300)는 제1 어플리케이션의 크래시가 재발생한 경우, 동작 709에서 해결 방법 #1 및 해결 방법 #2를 제1 DB로부터 다시 획득할 수 있다. 동작 713에서, 전자 장치(300)는 해결 방법 #1 및 해결 방법 #2 중 최상위 순서인 해결 방법 #1을 선택할 수 있다.
- [143] 동작 715에서, 전자 장치(300)는 선택된 해결 방법 #1이 수행되었는지 여부를 판단할 수 있다. 제1 DB에는 전자 장치(300)가 제1 어플리케이션의 오류를

해결하기 위해 해결 방법 #1을 사용자에게 제공한 이력이 기록되어 있을 수 있어, 전자 장치(300)는 선택된 해결 방법 #1이 수행된 것으로 판단할 수 있다. 전자 장치(300)는 선택된 해결 방법 #1이 수행된 경우(동작 715-예), 동작 727에서, 선택된 해결 방법 #1이 마지막인지 여부를 판단할 수 있다. 전자 장치(300)는 해결 방법 #2가 남아있으므로 선택된 해결 방법 #1이 마지막이 아닌 것으로 판단할 수 있고(동작 727-아니오), 동작 713에서, 차순위 순서인 해결 방법 #2를 선택할 수 있다.

- [144] 동작 715에서, 전자 장치(300)는 선택된 해결 방법 #2가 수행되었는지 여부를 판단할 수 있다. 제1 DB에는 전자 장치(300)가 제1 어플리케이션의 오류를 해결하기 위해 해결 방법 #2를 사용자에게 제공한 이력이 기록되어 있지 않을 수 있어, 전자 장치(300)는 선택된 해결 방법 #2는 수행되지 않은 것으로 판단할 수 있다(동작 715-아니오).
- [145] 전자 장치(300)는 선택된 해결 방법 #2가 수행되지 않은 경우(동작 715-아니오), 선택된 해결 방법 #2가 적합한지 여부를 판단할 수 있다. 전자 장치(300)는 전자 장치(300)의 상태 정보(예: 메모리 상태, 통신 연결 상태)를 고려하여, 선택된 해결 방법 #2가 적합한지 여부를 판단할 수 있다.
- [146] 전자 장치(300)는 선택된 해결 방법 #2가 적합한 경우(동작 717-예), 동작 719에서, 선택된 해결 방법 #2를 사용자에게 제공할 수 있다. 전자 장치(300)는 선택된 해결 방법 #2를 사용자에게 제공한 경우, 선택된 해결 방법 #2의 제공 횟수를 제1 DB에서 업데이트할 수 있고 선택된 해결 방법 #2를 제1 어플리케이션의 현재 버전에서 제공하였음을 제1 DB에 기록할 수 있다.
- [147] 선택된 해결 방법 #2이 수행되어도 제1 어플리케이션의 크래시가 또 다시 발생할 수 있다. 제1 어플리케이션의 크래시가 또 다시 발생한 경우, 전자 장치(300)는 동작 701, 동작 703, 동작 707, 동작 709, 및 동작 713을 다시 수행할 수 있다. 전자 장치(300)는 제1 어플리케이션의 크래시가 재발생한 경우, 동작 709를 수행하여 해결 방법 #1 및 해결 방법 #2를 제1 DB로부터 다시 획득할 수 있다. 전자 장치(300)는 동작 713에서 해결 방법 #1 및 해결 방법 #2 중 최상위 순서인 해결 방법 #1을 선택할 수 있고, 동작 715에서, 선택된 해결 방법 #1이 수행되었는지 판단할 수 있다. 전자 장치(300)는 선택된 해결 방법 #1이 수행되었으므로(동작 715-아니오), 동작 727에서, 선택된 해결 방법 #1이 마지막인지 여부를 판단할 수 있다. 전자 장치(300)는 선택된 해결 방법 #1이 최상위 순서일 수 있어, 선택된 해결 방법 #1은 마지막이 아닌 것으로 판단할 수 있고(동작 727-아니오), 동작 713에서, 차순위 순서인 해결 방법 #2를 선택할 수 있다. 동작 715에서, 전자 장치(300)는 선택된 해결 방법 #2이 수행되었는지 여부를 판단할 수 있다. 전자 장치(300)는 선택된 해결 방법 #2이 수행되었으므로(동작 715-예), 동작 727에서, 선택된 해결 방법 #2이 마지막인지 여부를 판단할 수 있다. 선택된 해결 방법 #2는 마지막일 수 있어(동작 727-예), 전자 장치(300)는 동작 721을 수행할 수 있다. 전자 장치(300)는 어플리케이션

마켓에 제1 어플리케이션의 업데이트 버전이 있는 경우(동작 721-예), 동작 723을 수행할 수 있고, 어플리케이션 마켓에 제1 어플리케이션의 업데이트 버전이 없는 경우(동작 721-아니오), 동작 725를 수행할 수 있다.

- [148] 도 1 내지 도 6을 통해 설명한 실시 예들은 도 7을 통해 설명한 실시 예들에 적용될 수 있어, 상세한 설명을 생략한다.
- [149]
- [150] 도 8 내지 도 10은 일 실시 예에 따른, 전자 장치의 팝업 창(예시)을 설명하기 위한 도면이다.
- [151] 도 8 내지 도 10에서, "%s"는 오류가 발생한 제1 어플리케이션의 이름을 나타낼 수 있다.
- [152] 일 실시예에 있어서, 도 8에 도시된 예에서, 전자 장치(예: 도 3의 전자 장치(300))는 제1 어플리케이션의 오류가 발생한 경우 오류 정보를 획득할 수 있다. 전자 장치(300)는 오류 정보로부터 제1 exception type(예: FileNotFoundException)을 추출할 수 있다. 전자 장치(300)는 제1 DB(예: 도 4의 제1 DB(424), 도 5의 제1 DB(521), 또는 도 6의 제1 DB(621))에 제1 exception type에 대한 해결 방법이 있는지 조회할 수 있고, 제1 DB로부터 캐시 정리(clear cache)의 해결 방법을 획득할 수 있다.
- [153] 일 실시예에 있어서, 전자 장치(300)는 캐시 정리의 해결 방법을 사용자에게 제공할 수 있다. 일례로, 도 8(박스 "(a)")에 도시된 예와 같이, 전자 장치(300)는 캐시 정리의 해결 방법을 안내하기 위한 팝업 창(810)을 디스플레이(예: 도 3의 디스플레이(330))에 표시할 수 있다. 전자 장치(300)는 팝업 창(810)의 캐시 정리(clear cache) 버튼(811)에 대한 사용자 입력(예: 터치 입력)이 있는 경우, 도 8(박스 "(b)")에 도시된 예와 같이 캐시를 정리 중임을 나타내는 시각적 피드백을 디스플레이(330)에 표시할 수 있다. 전자 장치(300)는 캐시 정리가 완료된 경우, 도 8(박스 "(c)")에 도시된 예와 같이, open 앱(app) 버튼(812)을 디스플레이(330)에 표시할 수 있다. 전자 장치(300)는 open app 버튼(812)에 대한 사용자 입력이 있는 경우 제1 어플리케이션을 실행할 수 있다.
- [154] 일 실시예에 있어서, 도 9에 도시된 예에서, 전자 장치(300)는 크래시가 발생한 제1 어플리케이션의 업데이트 안내를 사용자에게 제공할 수 있다. 일례로, 전자 장치(300)는 제1 DB로부터 반환받은 정보가 없는 경우 또는 제1 어플리케이션의 오류가 해결되지 않은 경우, 제1 어플리케이션의 업데이트 버전이 어플리케이션 마켓에 있는지 여부를 판단할 수 있다. 전자 장치(300)는 제1 어플리케이션의 업데이트 버전이 어플리케이션 마켓에 있는 경우 도 9에 도시된 예와 같이 제1 어플리케이션의 업데이트 안내를 위한 팝업 창(910)을 디스플레이(330)에 표시할 수 있다.
- [155] 일 실시예에 있어서, 도 9에 도시된 팝업 창(910)은 지금 업데이트(update) 버튼(911), 취소(cancel) 버튼(912), 및 제1 어플리케이션에 발생한 오류에 대한 정보(913)를 포함할 수 있다. 전자 장치(300)는 update 버튼(911)에 대한 사용자

입력이 있는 경우 제1 어플리케이션을 업데이트할 수 있다. 전자 장치(300)는 취소(cancel) 버튼(912)에 대한 사용자 입력이 있는 경우 제1 어플리케이션의 업데이트 제안내를 위한 팝업 창(920)을 디스플레이(330)에 표시할 수 있다.

- [156] 일 실시예에 있어서, 전자 장치(300)는 제1 어플리케이션의 업데이트 버전이 어플리케이션 마켓에 없는 경우, 도 10에 도시된 예와 같이 제1 어플리케이션의 오류에 대한 정보(1011)를 제공하는 팝업 창(1010)을 디스플레이(330)에 표시할 수 있다. 전자 장치(300)는 팝업 창(1010)의 got it 버튼(1012)에 대한 사용자 입력이 있는 경우, 팝업 창(1010)을 디스플레이(330)에서 사라지도록 할 수 있다. 전자 장치(300)는 제1 어플리케이션의 오류에 대한 정보(1011)에 사용자 입력이 있는 경우, 도 10의 (b)에 도시된 예와 같이, 제1 어플리케이션의 오류 발생 이력(예: 크래시 이력(1021))과 제1 어플리케이션의 삭제를 위한 버튼(1022)을 포함하는 화면(1020)을 디스플레이(330)에 표시할 수 있다. 오류 발생 이력(예: 크래시 이력(1021))은, 예를 들어, 발생한 exception 타입과 오류 발생 시간을 포함할 수 있다. 도 10에 도시된 예에서, 화면(1020)은 clear cache 버튼(1023)을 더 포함할 수 있다. 이는 예시적인 사항일 뿐, clear cache 버튼(1023)은 화면(1020)에서 생략될 수 있다.

[157]

- [158] 도 11 내지 도 12는 일 실시예에 따른, 전자 장치가 획득한 해결 방법들의 순서 변경의 예시들을 설명하기 위한 도면이다.

- [159] 일 실시예에 있어서, 전자 장치(예: 도 3의 전자 장치(300))는 제1 어플리케이션에 오류가 발생한 경우, 제1 DB(예: 도 4의 제1 DB(424), 도 5의 제1 DB(521), 또는 도 6의 제1 DB(621))로부터 오류를 해결하기 위한 복수의 해결 방법들을 획득할 수 있다. 일례로, 도 11에 도시된 예에서, 전자 장치(300)는 해결 방법 A, 해결 방법 B, 및 해결 방법 C를 획득할 수 있다. 해결 방법 A, 해결 방법 B, 및 해결 방법 C의 주어진 순서는, 예를 들어, A→B→C일 수 있다. 해결 방법 A, 해결 방법 B, 및 해결 방법 C 중에서 해결 방법 A는 최상위 순서일 수 있고 해결 방법 B는 차상위 순서일 수 있으며 해결 방법 C는 마지막 순서일 수 있다.

- [160] 일 실시예에 있어서, 전자 장치(300)는 획득된 해결 방법들 중에서, 연속되는 순서의 해결 방법들 사이의 제공 횟수의 차이를 결정할 수 있다. 일례로, 도 11에 도시된 예에서, 해결 방법 A의 제공 횟수는 10회일 수 있고, 해결 방법 B의 제공 횟수는 10회일 수 있으며, 따라서, 전자 장치(300)는 해결 방법 A의 제공 횟수와 해결 방법 B의 제공 횟수 사이의 차이를 0으로 결정할 수 있다. 해결 방법 B의 제공 횟수는 10회이고, 해결 방법 C의 제공 횟수는 5회이다. 따라서, 전자 장치(300)는 해결 방법 B의 제공 횟수와 해결 방법 C의 제공 횟수 사이의 차이 5를 결정할 수 있다.

- [161] 일 실시예에 있어서, 전자 장치(300)는 각 차이 중에서, 차상위 순서의 해결 방법의 제공 횟수와 차상위 순서보다 낮은 순서의 해결 방법의 제공 횟수 사이의 차이가 일정 수준 이상인 경우, 차상위 순서의 해결 방법이 변경된 순서에서

최상위 순서의 해결 방법이 되도록 주어진 순서를 변경할 수 있다. 일례로, 도 11에 도시된 예에서, 전자 장치(300)는 차상위 순서의 해결 방법 B의 제공 횟수와 차상위 순서보다 낮은 순서의 해결 방법 C의 제공 횟수의 차이 5는 일정 수준(예: 5) 이상일 수 있어, 해결 방법 B가 최상위 순서가 되도록 주어진 순서 $A \rightarrow B \rightarrow C$ 를 $B \rightarrow A \rightarrow C$ 로 변경할 수 있다.

- [162] 일 실시예에 있어서, 전자 장치(300)는 획득된 해결 방법들 중에서 연속되는 순서의 해결 방법들 사이의 제공 횟수의 차이를 계산할 수 있고, 계산된 각 차이가 서로 동일한 경우 주어진 순서에서의 최하위 순서의 해결 방법이 변경된 순서에서 최상위 순서의 해결 방법이 되도록 주어진 순서를 변경할 수 있다. 일례로, 도 12에 도시된 예에서, 전자 장치(300)는 연속되는 순서의 해결 방법 A와 해결 방법 B 각각의 제공 횟수의 차이 2를 계산할 수 있고, 연속되는 순서의 해결 방법 B와 해결 방법 C 각각의 제공 횟수의 차이 2를 계산할 수 있다. 전자 장치(300)는 계산된 각 차이가 동일할 수 있어, 주어진 순서 $A \rightarrow B \rightarrow C$ 에서의 최하위 순서의 해결 방법 C가 최상위 순서가 되도록 주어진 순서 $A \rightarrow B \rightarrow C$ 를 $C \rightarrow A \rightarrow B$ 로 변경할 수 있다.
- [163] 일 실시예에 있어서, 전자 장치(300)는 획득된 해결 방법들 각각의 제공 횟수에 정렬 알고리즘을 적용하여, 획득된 해결 방법들의 주어진 순서를 변경할 수 있다. 정렬 알고리즘은, 예를 들어, 오름 차순 및/또는 내림 차순을 포함할 수 있으나 이에 제한되지 않는다. 일례로, 도 12에 도시된 예에서, 전자 장치(300)는 해결 방법 A 내지 C 각각의 제공 횟수를 오름 차순으로 정렬하여 주어진 순서 $A \rightarrow B \rightarrow C$ 를 $C \rightarrow B \rightarrow A$ 로 변경할 수 있다.
- [164] 일 실시예에 있어서, 전자 장치(300)는 변경된 순서에서 최상위 순서의 해결 방법을 선택할 수 있다. 도 11에 도시된 예에서, 전자 장치(300)는 변경된 순서 $B \rightarrow A \rightarrow C$ 에서 해결 방법 B가 최상위 순서일 수 있어, 해결 방법 B를 선택할 수 있다. 도 12에 도시된 예에서, 전자 장치(300)는 변경된 순서 $C \rightarrow A \rightarrow B$ 에서 해결 방법 C가 최상위 순서일 수 있어, 해결 방법 C를 선택할 수 있다. 전자 장치(300)는 선택된 해결 방법의 적합성을 검증할 수 있고, 선택된 해결 방법이 적합한 것으로 검증되는 경우 선택된 해결 방법을 사용자에게 제공할 수 있다. 전자 장치(300)는 선택된 해결 방법이 적합하지 않은 것으로 검증되는 경우, 변경된 순서에서 차상위 순서의 해결 방법(예: 도 11의 변경된 순서의 경우 해결 방법 A, 도 12의 변경된 순서의 경우 해결 방법 A)을 선택할 수 있고, 차상위 순서의 해결 방법의 적합성을 검증할 수 있으며, 차상위 순서의 해결 방법이 적합한 경우 차상위 순서의 해결 방법을 사용자에게 제공할 수 있다.
- [165] 일 실시예에 있어서, 해결 방법들의 주어진 순서가 변경되었고 해결 방법들 중 어느 하나가 수행되어 제1 어플리케이션의 오류가 해결된 경우, 전자 장치(300)는 해결 방법들 각각의 제공 횟수를 초기화할 수 있다. 일례로, 도 11과 도 12에 도시된 예에서, 전자 장치(300)는 해결 방법 A 내지 해결 방법 C 각각의 제공 횟수를 0으로 초기화할 수 있다. 전자 장치(300)는 제1 어플리케이션에

발생했던 오류와 동일한 오류가 다른 어플리케이션에 발생한 경우, 제1 DB로부터 해결 방법 A 내지 해결 방법 C를 획득할 수 있다. 해결 방법 A 내지 해결 방법 C의 주어진 순서는 $A \rightarrow B \rightarrow C$ 일 수 있어, 전자 장치(300)는 해결 방법 A의 적합성을 가장 먼저 검증할 수 있고 해결 방법 A가 적합한 경우 해결 방법 A를 사용자에게 가장 먼저 제공할 수 있다. 다른 일례로, 전자 장치(300)는 해결 방법 A 내지 해결 방법 C의 제공 횟수 이력을 참조하여 해결 방법 C의 제공 횟수가 최소이었던 것을 확인할 수 있다. 전자 장치(300)는 해결 방법 A 내지 해결 방법 C의 주어진 순서 $A \rightarrow B \rightarrow C$ 를 $C \rightarrow A \rightarrow B$ 또는 $C \rightarrow B \rightarrow A$ 로 변경할 수 있고, 해결 방법 C의 적합성을 가장 먼저 검증할 수 있으며, 해결 방법 C가 적합한 경우 해결 방법 C를 사용자에게 가장 먼저 제공할 수 있다.

- [166] 일 실시예에 있어서, 전자 장치(300)는 오류 정보로부터 exception type "FileNotFoundException"이 발생하였음을 파악할 수 있고, 오류의 원인을 파악 불러 오기 실패로 파악할 수 있다. 전자 장치(300)는 파악한 exception type과 오류 발생 원인을 통해 제1 DB로부터 해결 방법 A 내지 C를 획득할 수 있다. 해결 방법 A 내지 C 중 해결 방법 B는 캐시 정리일 수 있고 해결 방법 C는 메모리 데이터 영역 정리일 수 있다. 해결 방법 A 내지 C의 주어진 순서는, 예를 들어, $A \rightarrow B \rightarrow C$ 일 수 있고, 해결 방법 A 내지 C의 주어진 순서를 변경할 수 있다. 전자 장치(300)는 발생한 exception type이 "FileNotFoundException"이어서, 해결 방법 C를 마지막 순서로 고정할 수 있다. 메모리(예: 도 3의 메모리(310))의 데이터 영역에는 중요한 데이터가 저장되어 있을 수 있어, 전자 장치(300)는 해결 방법 A와 해결 방법 B의 순서를 변경하여도 해결 방법 C의 순서를 변경하지 않을 수 있다.
- [167]
- [168] 도 13은 일 실시예에 따른, 전자 장치의 동작 방법의 일 예시를 설명하기 위한 흐름도이다.
- [169] 도 13에 도시된 동작 방법은 전자 장치(예: 도 3의 전자 장치(300))에 의해 수행될 수 있다.
- [170] 동작 1310에서, 전자 장치(300)는 실행 중인 제1 어플리케이션에 오류가 발생한 경우 오류 정보를 획득할 수 있다.
- [171] 동작 1320에서, 전자 장치(300)는 획득된 오류 정보에서 제1 어플리케이션에 발생한 오류의 타입과 원인을 파악할 수 있다.
- [172] 동작 1330에서, 전자 장치(300)는 파악된 타입 및 원인을 이용하여 참조 데이터 셋으로부터 제1 어플리케이션에 발생된 오류를 해결하기 위한 해결 방법을 획득할 수 있다.
- [173] 동작 1340에서, 전자 장치(300)는 전자 장치(300)의 상태 정보를 기초로, 획득된 해결 방법의 적합성을 검증할 수 있다.
- [174] 동작 1350에서, 전자 장치(300)는 획득된 해결 방법이 적합한 것으로 검증된 경우, 획득된 해결 방법에 대한 안내를 디스플레이(예: 도 3의

디스플레이(330))를 통해 사용자에게 제공할 수 있다.

- [175] 도 1 내지 도 12를 통해 설명한 실시 예들은 도 13을 통해 설명한 실시 예들에 적용될 수 있어, 상세한 설명을 생략한다.
- [176]
- [177] 일 실시예에 따르면, 전자 장치(예: 도 3의 전자 장치(300))는 하나 이상의 어플리케이션을 저장하는 메모리(예: 도 3의 메모리(310)), 디스플레이(예: 도 3의 디스플레이(330)), 및 실행 중인 제1 어플리케이션에 오류가 발생한 경우 오류 정보를 획득하고, 상기 획득된 오류 정보에서 상기 발생한 오류의 타입과 원인을 파악하며, 상기 파악된 타입 및 원인을 이용하여 참조 데이터 셋으로부터 상기 발생한 오류를 해결하기 위한 해결 방법을 획득하고, 상기 전자 장치의 상태 정보를 기초로 상기 획득된 해결 방법의 적합성을 검증하고, 상기 획득된 해결 방법이 적합한 것으로 검증된 경우 상기 획득된 해결 방법에 대한 안내를 상기 디스플레이를 통해 사용자에게 제공하는 프로세서(예: 도 3의 프로세서(320))를 포함할 수 있다.
- [178] 상기 참조 데이터 셋은 하나 이상의 오류 타입, 하나 이상의 오류 발생 원인, 및 하나 이상의 오류 해결 방법을 포함할 수 있다.
- [179] 상기 프로세서는 상기 획득된 해결 방법에 의해서도 상기 발생한 오류가 해결되지 않는 경우 상기 제1 어플리케이션의 업데이트 버전이 있는지 여부를 판단하고, 상기 업데이트 버전이 있는 경우 상기 제1 어플리케이션의 업데이트에 대한 안내를 상기 디스플레이를 통해 상기 사용자에게 제공할 수 있다.
- [180] 상기 프로세서는 상기 업데이트 버전이 없는 경우 상기 발생한 오류에 대한 정보(예: 도 10의 정보(1011))를 상기 디스플레이를 통해 상기 사용자에게 제공하고, 상기 제공된 정보에 대한 사용자 입력이 있는 경우 상기 발생한 오류의 이력과 상기 제1 어플리케이션의 삭제를 위한 버튼을 포함하는 화면(예: 도 10의 화면(1020))을 상기 디스플레이에 표시할 수 있다.
- [181] 상기 프로세서는 상기 참조 데이터 셋에 상기 파악된 타입에 해당하는 오류 타입이 없는 경우, 상기 제1 어플리케이션의 업데이트 버전이 있는지 여부를 판단하고, 상기 업데이트 버전이 있는 경우 상기 제1 어플리케이션의 업데이트에 대한 안내를 상기 디스플레이를 통해 상기 사용자에게 제공할 수 있다.
- [182] 상기 프로세서는 상기 파악된 타입 및 원인을 이용하여 상기 참조 데이터 셋으로부터 상기 발생한 오류를 해결하기 위한 복수의 해결 방법들을 획득한 경우, 상기 획득된 해결 방법들의 주어진 순서(given order)에서 최상위 순서의 해결 방법을 선택하고, 상기 선택된 해결 방법의 적합성을 검증하며, 상기 선택된 해결 방법이 적합한 것으로 검증된 경우 상기 선택된 해결 방법에 대한 안내를 상기 디스플레이를 통해 상기 사용자에게 제공할 수 있다.
- [183] 상기 프로세서는 상기 파악된 타입 및 원인을 이용하여 상기 참조 데이터

셋으로부터 상기 발생한 오류를 해결하기 위한 복수의 해결 방법들을 획득한 경우, 상기 획득된 해결 방법들 각각의 제공 횟수를 기초로 상기 획득된 해결 방법들의 주어진 순서를 변경(또는 어레인지(arrange))하고, 상기 변경된 순서에서의 최상위 순서의 해결 방법을 선택하며, 상기 선택된 해결 방법의 적합성을 검증하고, 상기 선택된 해결 방법이 적합한 것으로 검증된 경우 상기 선택된 해결 방법에 대한 안내를 상기 디스플레이를 통해 상기 사용자에게 제공할 수 있다.

- [184] 상기 프로세서는 상기 획득된 해결 방법들 중에서 연속되는 순서의 해결 방법들 사이의 제공 횟수의 차이를 계산하고, 상기 계산된 각 차이가 서로 동일한 경우 상기 주어진 순서에서의 최하위 순서의 해결 방법(예: 도 12의 해결 방법 C)이 상기 변경된 순서에서 최상위 순서의 해결 방법이 되도록 상기 주어진 순서를 변경할 수 있다.
- [185] 상기 프로세서는 상기 계산된 각 차이 중에서, 상기 주어진 순서에서의 차상위 순서의 해결 방법의 제공 횟수와 상기 차상위 순서보다 낮은 순서의 해결 방법의 제공 횟수 사이의 차이가 일정 수준 이상인 경우, 상기 차상위 순서의 해결 방법(예: 도 11의 해결 방법 B)이 상기 변경된 순서에서 최상위 순서의 해결 방법이 되도록 상기 주어진 순서를 변경할 수 있다.
- [186] 상기 프로세서는 상기 주어진 순서가 변경된 경우 상기 획득된 해결 방법들 각각의 상기 제공 횟수를 초기화할 수 있다.
- [187] 상기 프로세서는 상기 파악된 타입 및 원인을 이용하여 상기 참조 데이터 셋으로부터 상기 발생한 오류를 해결하기 위한 복수의 해결 방법들을 획득한 경우, 상기 획득된 해결 방법들의 리스트를 상기 디스플레이를 통해 상기 사용자에게 제공하고, 상기 사용자의 선택이 있는 해결 방법을 수행할 수 있다.
- [188] 상기 참조 데이터 셋은 상기 메모리 및 서버(예: 도 6의 서버(620)) 중 적어도 하나에 저장될 수 있다.
- [189] 상기 프로세서는 상기 획득된 오류 정보를 기초로 상기 제1 어플리케이션이 서드 파티 어플리케이션인지 여부를 확인 또는 결정하고, 상기 제1 어플리케이션이 상기 서드 파티 어플리케이션인 경우 상기 타입과 원인을 파악하는 동작, 상기 해결 방법을 획득하는 동작, 상기 적합성을 검증하는 동작, 및 상기 안내를 상기 사용자에게 제공하는 동작을 수행할 수 있다.
- [190] 상기 프로세서는 운영 체제의 프레임워크(예: 도 4의 프레임워크(410))를 통해 또는 제2 어플리케이션(예: 도 4의 시스템 어플리케이션(420))을 통해 상기 안내를 위한 팝업 창을 상기 디스플레이에 표시할 수 있다.
- [191] 일 실시예에 따르면, 전자 장치(예: 도 3의 전자 장치(300))는 하나 이상의 오류 타입, 하나 이상의 오류 발생 원인, 및 하나 이상의 오류 해결 방법을 포함하는 참조 데이터 셋을 저장하고, 하나 이상의 어플리케이션을 저장하는 메모리(예: 도 3의 메모리 (310)), 디스플레이(예: 도 3의 디스플레이(330)), 및 실행 중인 제1 어플리케이션에 오류가 발생한 경우 오류 정보를 획득하고, 상기 획득된 오류

정보를 기초로 상기 제1 어플리케이션이 서드 파티 어플리케이션인지 여부를 확인 또는 결정하며, 상기 제1 어플리케이션이 서드 파티 어플리케이션인 경우 상기 획득된 오류 정보에서 상기 발생한 오류의 타입과 원인을 파악하고, 상기 파악된 타입 및 원인을 이용하여 상기 참조 데이터 셋으로부터 상기 발생한 오류를 해결하기 위한 해결 방법을 획득하고, 상기 전자 장치의 상태 정보를 기초로 상기 획득된 해결 방법의 적합성을 검증하고, 상기 획득된 해결 방법이 적합한 것으로 검증된 경우 상기 획득된 해결 방법에 대한 안내를 위한 팝업 창을 상기 디스플레이에 표시하는 프로세서(예: 도 3의 프로세서(320))를 포함할 수 있다.

- [192] 상기 프로세서는 상기 획득된 해결 방법에 의해서도 상기 발생한 오류가 해결되지 않는 경우 상기 제1 어플리케이션의 업데이트 버전이 있는지 여부를 판단하고, 상기 업데이트 버전이 있는 경우 상기 제1 어플리케이션의 업데이트에 대한 안내를 위한 팝업 창을 상기 디스플레이에 표시할 수 있다.
- [193] 상기 프로세서는 상기 파악된 타입 및 원인을 이용하여 상기 참조 데이터 셋으로부터 상기 발생한 오류를 해결하기 위한 복수의 해결 방법들을 획득한 경우, 상기 획득된 해결 방법들의 주어진 순서에서 최상위 순서의 해결 방법을 선택하고, 상기 선택된 해결 방법의 적합성을 검증하며, 상기 선택된 해결 방법이 적합한 것으로 검증된 경우 상기 선택된 해결 방법에 대한 안내를 위한 팝업 창을 상기 디스플레이에 표시할 수 있다.
- [194] 상기 프로세서는 상기 파악된 타입 및 원인을 이용하여 상기 참조 데이터 셋으로부터 상기 발생한 오류를 해결하기 위한 복수의 해결 방법들을 획득한 경우, 상기 획득된 해결 방법들 각각의 제공 횟수를 기초로 상기 획득된 해결 방법들의 주어진 순서를 변경(또는 어레인지)하고, 상기 변경된 순서에서의 최상위 순서의 해결 방법을 선택하며, 상기 선택된 해결 방법의 적합성을 검증하고, 상기 선택된 해결 방법이 적합한 것으로 검증된 경우 상기 선택된 해결 방법에 대한 안내를 위한 팝업 창을 상기 디스플레이에 표시할 수 있다.
- [195] 상기 프로세서는 운영 체제의 프레임워크를 통해 또는 제2 어플리케이션을 통해 상기 팝업 창을 상기 디스플레이에 표시할 수 있다.
- [196] 일 실시예에 따르면, 전자 장치(예: 도 3의 전자 장치(300))의 동작 방법은 실행 중인 제1 어플리케이션에 오류가 발생한 경우 오류 정보를 획득하는 동작, 상기 획득된 오류 정보에서 상기 발생한 오류의 타입과 원인을 파악하는 동작, 상기 파악된 타입 및 원인을 이용하여 참조 데이터 셋으로부터 상기 발생한 오류를 해결하기 위한 해결 방법을 획득하는 동작, 상기 전자 장치의 상태 정보를 기초로 상기 획득된 해결 방법의 적합성을 검증하는 동작, 및 상기 획득된 해결 방법이 적합한 것으로 검증된 경우 상기 획득된 해결 방법에 대한 안내를 디스플레이를 통해 사용자에게 제공하는 동작을 포함할 수 있다.
- [197] 일 실시예에 있어서, 전자 장치의 동작 방법은 상기 획득된 해결 방법에 의해서도 상기 발생한 오류가 해결되지 않는 경우 상기 제1 어플리케이션의 업데이트

버전이 있는지 여부를 판단하는 동작 및 상기 업데이트 버전이 있는 경우 상기 제1 어플리케이션의 업데이트에 대한 안내를 상기 디스플레이를 통해 상기 사용자에게 제공하는 동작을 더 포함할 수 있다.

청구범위

- [청구항 1] 전자 장치에 있어서,
디스플레이;
프로세서; 및
상기 프로세서와 작동적으로 연결되고 상기 프로세서에 의해 실행되는
하나 이상의 명령어들을 저장하는 메모리
를 포함하고,
상기 명령어들이 실행될 때 상기 프로세서는,
상기 프로세서에 의해 실행되는 제1 어플리케이션에 발생한 오류를
기초로 오류 정보를 획득하고, 상기 획득된 오류 정보를 기초로 상기
오류의 타입과 상기 오류의 원인을 파악하며, 상기 오류의 상기 파악된
타입 및 상기 오류의 파악된 원인을 이용하여 참조 데이터 셋으로부터
상기 오류를 해결하기 위한 해결 방법을 획득하고, 상기 전자 장치의 상태
정보를 기초로 상기 획득된 해결 방법의 적합성을 검증하고, 상기 적합한
것으로 검증된 획득된 해결 방법을 기초로 상기 디스플레이를 통해 상기
획득된 해결 방법에 대한 안내를 사용자에게 제공하고,
상기 참조 데이터 셋은 하나 이상의 오류 타입, 하나 이상의 오류 발생
원인, 및 하나 이상의 오류 해결 방법을 포함하는,
전자 장치.
- [청구항 2] 제1항에 있어서,
상기 프로세서는,
상기 획득된 해결 방법에 의해서도 해결되지 않은 상기 오류에 기초하여,
상기 제1 어플리케이션의 업데이트 버전이 있는지 여부를 판단하고, 상기
업데이트 버전에 기초하여, 상기 제1 어플리케이션의 상기 업데이트
버전에 대한 안내를 상기 디스플레이를 통해 상기 사용자에게 제공하는,
전자 장치.
- [청구항 3] 제2항에 있어서,
상기 프로세서는,
상기 제1 어플리케이션의 상기 업데이트 버전이 없는 판단에 기초하여,
상기 오류에 대한 정보를 상기 디스플레이를 통해 상기 사용자에게
제공하고, 상기 제공된 정보에 대한 사용자 입력에 기초하여, 상기 오류의
이력과 상기 제1 어플리케이션의 삭제를 위한 버튼을 포함하는 화면을
상기 디스플레이에 표시하는,
전자 장치.
- [청구항 4] 제1항 내지 제3항 중 어느 하나에 있어서,
상기 프로세서는,
상기 참조 데이터 셋에 상기 파악된 타입에 해당하는 오류 타입이 없는

판단에 기초하여, 상기 제1 어플리케이션의 업데이트 버전이 있는지 여부를 판단하고, 상기 제1 어플리케이션의 상기 업데이트 버전이 있다는 판단에 기초하여, 상기 제1 어플리케이션의 상기 업데이트 버전에 대한 안내를 상기 디스플레이를 통해 상기 사용자에게 제공하는, 전자 장치.

[청구항 5] 제1항 내지 제4항 중 어느 하나에 있어서, 상기 프로세서는, 상기 파악된 타입 및 원인을 이용하여 상기 참조 데이터 셋으로부터 상기 오류를 해결하기 위한 복수의 해결 방법들을 획득하고, 상기 획득된 해결 방법들의 주어진 순서(given order)에서 최상위 순서의 해결 방법을 선택하며, 상기 선택된 해결 방법의 적합성을 검증하고, 적합한 것으로 검증된 상기 선택된 해결 방법에 기초하여, 상기 선택된 해결 방법에 대한 안내를 상기 디스플레이를 통해 상기 사용자에게 제공하는, 전자 장치.

[청구항 6] 제1항 내지 제5항 중 어느 하나에 있어서, 상기 프로세서는, 상기 파악된 타입 및 원인을 이용하여 상기 참조 데이터 셋으로부터 상기 오류를 해결하기 위한 복수의 해결 방법들을 획득하고, 상기 획득된 해결 방법들 각각의 제공 횟수를 기초로 상기 획득된 해결 방법들의 주어진 순서를 어레인지(arrange)하며, 상기 주어진 순서에서의 최상위 위치의 해결 방법을 선택하고, 상기 선택된 해결 방법의 적합성을 검증하고, 적합한 것으로 검증된 상기 선택된 해결 방법에 기초하여, 상기 선택된 해결 방법에 대한 안내를 상기 디스플레이를 통해 상기 사용자에게 제공하는, 전자 장치.

[청구항 7] 제6항에 있어서, 상기 프로세서는, 상기 획득된 해결 방법들의 제공 횟수 사이의 차이들을 결정하고, 상기 결정된 차이들을 기초로 상기 주어진 순서에서의 최하위 위치의 해결 방법이 상기 주어진 순서에서의 최상위 위치로 이동하도록 상기 주어진 순서를 변경하는, 전자 장치.

[청구항 8] 제7항에 있어서, 상기 프로세서는, 상기 주어진 순서에서의 차상위 위치의 제1 해결 방법의 제1 제공 횟수와 상기 제1 해결 방법보다 낮은 제2 해결 방법의 제2 제공 횟수 사이의 차이가 일정 수준 이상인지 결정하고, 상기 결정된 차이에 기초하여, 상기 차상위 위치의 제1 해결 방법이 상기 주어진 순서에서 최상위 위치로

- 이동하도록 상기 주어진 순서를 변경하는,
전자 장치.
- [청구항 9] 제6항에 있어서,
상기 프로세서는,
어레인지된 상기 주어진 순서에 기초하여, 상기 획득된 해결 방법들
각각의 상기 제공 횟수를 초기화하는,
전자 장치.
- [청구항 10] 제1항 내지 제9항 중 어느 하나에 있어서,
상기 프로세서는,
상기 파악된 타입 및 상기 파악된 원인을 이용하여 상기 참조 데이터
셋으로부터 상기 오류를 해결하기 위한 복수의 해결 방법들을 획득하고,
상기 획득된 해결 방법들의 리스트를 상기 디스플레이를 통해 상기
사용자에게 제공하며, 상기 제공된 해결 방법들에서 상기 사용자에게 의해
선택된 해결 방법을 수행하는,
전자 장치.
- [청구항 11] 제1항 내지 제10항 중 어느 하나에 있어서,
상기 참조 데이터 셋은,
상기 메모리 또는 서버 중 적어도 하나에 저장되는,
전자 장치.
- [청구항 12] 제1항 내지 제11항 중 어느 하나에 있어서,
상기 프로세서는,
제1 어플리케이션이 서드 파티(third party) 어플리케이션인지 여부를
결정하고, 상기 제1 어플리케이션이 상기 서드 파티 어플리케이션으로
결정된 것에 기초하여, 상기 오류의 상기 원인과 상기 오류의 상기 타입을
파악하는,
전자 장치.
- [청구항 13] 제1항 내지 제12항 중 어느 하나에 있어서,
상기 프로세서는,
운영 체제의 프레임워크를 통해 또는 제2 어플리케이션을 통해 상기
안내를 위한 팝업 창을 상기 디스플레이에 표시하는,
전자 장치.
- [청구항 14] 전자 장치의 동작 방법에 있어서,
실행되는 제1 어플리케이션에 발생한 오류를 기초로 오류 정보를
획득하는 동작;
상기 획득된 오류 정보에서 상기 오류의 타입과 상기 오류의 원인을
파악하는 동작;
상기 파악된 타입 및 상기 파악된 원인을 이용하여 참조 데이터
셋으로부터 상기 오류를 해결하기 위한 해결 방법을 획득하는 동작;

상기 전자 장치의 상태 정보를 기초로 상기 획득된 해결 방법의 적합성을 검증하는 동작; 및

적합한 것으로 검증된 상기 획득된 해결 방법에 기초하여, 상기 획득된 해결 방법에 대한 안내를 디스플레이를 통해 사용자에게 제공하는 동작을 포함하고,

상기 참조 데이터 셋은,

복수의 오류 타입들, 복수의 오류 발생 원인들, 및 복수의 해결 방법들을 포함하는,

전자 장치의 동작 방법.

[청구항 15]

제14항에 있어서,

상기 획득된 해결 방법에 의해서도 해결되지 않는 상기 오류에 기초하여 상기 제1 어플리케이션의 업데이트 버전이 있는지 여부를 판단하는 동작; 및

상기 업데이트 버전이 있는 판단에 기초하여, 상기 제1 어플리케이션의 상기 업데이트 버전에 대한 안내를 상기 디스플레이를 통해 상기 사용자에게 제공하는 동작

을 더 포함하는,

전자 장치의 동작 방법.