



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2019년02월19일
(11) 등록번호 10-1949337
(24) 등록일자 2019년02월12일

(51) 국제특허분류(Int. Cl.)
G06F 16/00 (2019.01)

(52) CPC특허분류
G06F 16/22 (2019.01)
G06F 16/176 (2019.01)

(21) 출원번호 10-2017-0150120

(22) 출원일자 2017년11월11일
심사청구일자 2017년11월11일

(56) 선행기술조사문헌
KR1020120015756 A*
(뒷면에 계속)

(73) 특허권자

(주)에스엠시스템

경기도 광주시 경안천로 159 ,101-1202(송정동, 금강펜테리움1단지)

(72) 발명자

김성민

경기도 광주시 경안천로 159, 101-1202

(74) 대리인

김영관

전체 청구항 수 : 총 1 항

심사관 : 박미정

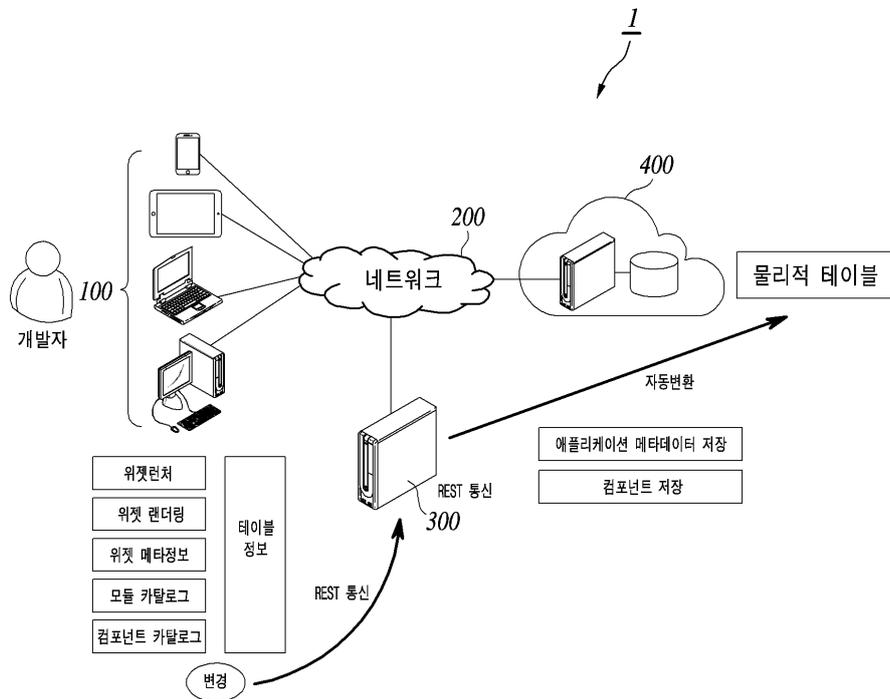
(54) 발명의 명칭 RIA를 이용한 클라우드 기반 aPaaS 제공 방법

(57) 요약

RIA를 이용한 클라우드 기반 aPaaS 제공 방법이 제공되며, 사용자 단말에서 신규 애플리케이션을 위한 RIA(Rich Internet Application)엔진을 요청하는 경우, 요청에 대응하여 RIA 엔진을 전송하는 단계, 사용자 단말에서 RIA 엔진을 실행하여 실행 환경을 생성하고, 실행 환경에서 개발 환경으로 변환하는 경우, 클라우드 서버로 테이블

(뒷면에 계속)

대표도 - 도1



생성을 요청하는 단계, 클라우드 서버에서 테이블 생성에 대응하는 물리적 테이블이 생성되면, 사용자 단말에서 신규 애플리케이션의 개발을 위한 적어도 하나의 모듈과 물리적 매칭 관계를 형성하도록 메타 데이터를 매칭시키는 단계, 사용자 단말에서 적어도 하나의 모듈로 기본 위젯을 생성하고, UI(User Interface)를 기본 위젯에 포함된 도구틀바로 생성하는 경우, 물리적 테이블의 컬럼이 요구되는지의 여부를 확인하는 단계, 및 물리적 테이블의 컬럼이 요구되는 경우, 클라우드 서버로 생성된 UI에 대응하는 테이블 컬럼을 생성하도록 요청하는 단계를 포함한다.

(52) CPC특허분류
G06F 16/1824 (2019.01)

(56) 선행기술조사문헌
 KR1020120045586 A*
 KR1020130050205 A
 KR1020120000595 A
 KR1020110090428 A
 KR101013810 B1
 KR101272867 B1
 *는 심사관에 의하여 인용된 문헌

이 발명을 지원한 국가연구개발사업
 과제고유번호 C0501-17-2019
 부처명 과학기술정보통신부
 연구관리전문기관 정보통신산업진흥원
 연구사업명 글로벌 SaaS 육성 프로젝트(GSIP) 아이디어형 SaaS 개발지원 2차(일반)
 연구과제명 기업형 App빌더 SaaS 개발
 기여율 1/1
 주관기관 (주)에스엠시스템
 연구기간 2017.08.01 ~ 2017.11.30

명세서

청구범위

청구항 1

클라우드 기반 aPaaS(Application Platform Services) 제공 서버에서 제공되는 클라우드 기반 aPaaS 제공 방법에 있어서,

사용자 단말에서 신규 애플리케이션을 위한 RIA(Rich Internet Application)엔진을 요청하는 경우, 상기 요청에 대응하여 RIA 엔진을 전송하는 단계;

상기 사용자 단말에서 상기 RIA 엔진을 실행하여 실행 환경을 생성하고, 상기 실행 환경에서 개발 환경으로 변환하는 경우, 클라우드 서버로 테이블 생성을 요청하는 단계;

상기 클라우드 서버에서 상기 테이블 생성에 대응하는 물리적 테이블이 생성되면, 상기 사용자 단말에서 상기 신규 애플리케이션의 개발을 위한 적어도 하나의 모듈과 물리적 매칭 관계를 형성하도록 메타 데이터를 매칭시키는 단계;

상기 사용자 단말에서 상기 적어도 하나의 모듈로 기본 위젯을 생성하고, UI(User Interface)를 상기 기본 위젯에 포함된 도구틀바로 생성하는 경우, 상기 물리적 테이블의 컬럼이 요구되는지의 여부를 확인하는 단계;

상기 물리적 테이블의 컬럼이 요구되는 경우, 상기 클라우드 서버로 상기 생성된 UI에 대응하는 테이블 컬럼을 생성하도록 요청하는 단계;

를 포함하고,

상기 물리적 테이블의 컬럼이 요구되는 경우, 상기 클라우드 서버로 상기 생성된 UI에 대응하는 테이블 컬럼을 생성하도록 요청하는 단계 이후에,

상기 기본 위젯을 변경할 때, 상기 사용자 단말에서 입력한 정보는, REST(Representational State Transfer)ful 통신규격으로 변환되어 수신되도록 제어하는 단계;

REST(Representational State Transfer) 통신규격의 리소스인 URI와 메소드(Method)인 HTTP에 기반하여 상기 클라우드 서버의 상기 물리적 테이블과 상기 사용자 단말을 통해 변경된 기본 위젯 간의 관계를 연결하는 단계;

를 더 포함하고,

상기 클라우드 기반 aPaaS 제공 서버는, 상기 HTTP와, 상기 기본 위젯을 변경하기 위해 상기 사용자 단말에서 입력한 정보에 대응하는 파라미터를 기반으로, 동적으로 SQL을 생성하여 상기 클라우드 서버의 상기 물리적 테이블과 상기 사용자 단말의 상기 기본 위젯 간의 관계를 설정함으로써, 상기 사용자 단말의 상기 기본 위젯이 변경되면 상기 클라우드 서버의 상기 물리적 테이블의 데이터 구조가 자동으로 완성되도록 하는

RIA를 이용한 클라우드 기반 aPaaS 제공 방법.

청구항 2

삭제

청구항 3

삭제

청구항 4

삭제

청구항 5

삭제

청구항 6

삭제

청구항 7

삭제

청구항 8

삭제

청구항 9

삭제

청구항 10

삭제

발명의 설명

기술 분야

[0001] 본 발명은 RIA를 이용한 클라우드 기반 aPaaS 제공 방법에 관한 것으로, 위젯과 물리적 테이블을 연동함으로써 위젯의 드래그 앤 드롭만으로도 데이터베이스로 접근 및 변경이 가능한 방법을 제공한다.

배경 기술

[0002] 클라우드 컴퓨팅에서 제공하는 서비스는 크게 SaaS, PaaS, IaaS 세 가지를 가장 대표적인 서비스로 분류한다. 구체적으로는, 적용 모델별 어떤 플랫폼과 기술 요소가 필요한지에 따라 구분될 수 있다. 구글 앱 엔진처럼 개발자들을 위한 API를 제공하는 플랫폼 모델이 있고, 유저를 위한 플랫폼이 존재하는데, PaaS(Platform as a Service)는 개발을 위한 플랫폼 구축을 할 필요없이 개발 요소를 웹에서 빌려쓸 수 있도록 하는 모델이다.

[0003] 이때, 일반적인 PaaS는 클라우드를 기반으로 웹(Web)을 통하여 애플리케이션 개발 및 시험 과정을 거쳐 소스코드가 완성되며, 운영 서버로 배포하는 과정을 관리하는데, 개발도구를 웹에서 빌려쓰는 장점을 제외하고는, 전통적인 개발 방법이 그대로 적용되므로, 개발의 단계와 복잡도가 오히려 더 증가하며, 단계별로 전문화된 기술은 여전히 필요하게 되었다.

[0004] 이때, 애플리케이션의 통합 플랫폼을 제공하는 방법은 사용자 인터페이스를 제공하는 방법으로 이루어지고 있다. 이와 관련하여, 선행기술인 한국공개특허 제2014-0019623호(2014년02월17일 공개)에는, 개발자가 CSS나 HTML5에 관련된 전문적인 지식이 없더라도 웹 어플리케이션의 프로젝트 생성에서부터 테스트, 디버깅, 프로파일링, 빌드, 패키징, 배포에 이르는 과정을 심리스하게(Seamless) 일괄적으로 수행하는 구성을 개시하고, 선행기술인 한국공개특허 제2014-0069525호(2014년06월10일 공개)에는, 사용자인터페이스부를 통해 어플리케이션 실행 명령을 수신하면, 실행환경에 독립한 통합컴포넌트를 생성하기 위한 UI 화면제어 및 화면구성 스크립트 소스를 수신하여 실행환경에 독립한 통합컴포넌트를 생성하는 구성을 개시한다.

[0005] 다만, 전자의 경우, 웹 기반의 드래그 앤 드롭을 통한 애플리케이션의 UI 소스를 만드는 방법으로 UI 비즈니스 처리를 위해서는 반드시 스크립트를 구현해야 하는 방법이 필요하고, 소스를 별도로 운영서버에 빌더(Builder) 하고 배포하는 과정이 요구된다. 또한, 후자의 경우, 프로그램의 전반적인 영역을 관리하는 것이 아닌 개발자의 방법에서 UI를 구현하는 방법을 정의한 것으로 실행관점에서 서버 비즈로직이나 데이터 처리를 위한 자동화된 방법은 개시되어 있지 않다. 또한, 전자 및 후자는 물리적인 테이블의 구조에 기반하지 않았으므로, 테이블의 구조가 변경할 때에는 프로그램이 자동으로 대응하지 못하고, 전적으로 개발자의 스크립트에 의존해야 한다.

발명의 내용

해결하려는 과제

[0006] 본 발명의 일 실시예는, RIA 기술을 적용한 클라이언트 애플리케이션과, API 기반의 서버 서비스로 분리하여 aPaaS 플랫폼 서비스의 유지 보수의 방법을 제공하고, UI 디자인 시 개발 도구와 물리적인 필드 관계를 설정하여 물리적인 테이블의 변경을 자동화할 수 있고, RESTful 통신규격의 URI와 HTTP 메소드를 이용하여 객체 화 서비스에 접근하여 데이터베이스의 처리를 자동으로 구현함으로써 별도의 서버 프로그램이 없을지라도 물리적인 테이블을 변경하도록 하고, 단순한 UI 구성만을 통해서 애플리케이션의 개발을 가능케함으로써 저비용 및 단시간 내의 개발을 완성토록 하는, RIA를 이용한 클라우드 기반 aPaaS 제공 방법을 제공할 수 있다. 다만, 본 실시예가 이루고자 하는 기술적 과제는 상기된 바와 같은 기술적 과제로 한정되지 않으며, 또 다른 기술적 과제들이 존재할 수 있다.

과제의 해결 수단

[0007] 상술한 기술적 과제를 달성하기 위한 기술적 수단으로서, 본 발명의 일 실시예는, 사용자 단말에서 신규 애플리케이션을 위한 RIA(Rich Internet Application)엔진을 요청하는 경우, 요청에 대응하여 RIA 엔진을 전송하는 단계, 사용자 단말에서 RIA 엔진을 실행하여 실행 환경을 생성하고, 실행 환경에서 개발 환경으로 변환하는 경우, 클라우드 서버로 테이블 생성을 요청하는 단계, 클라우드 서버에서 테이블 생성에 대응하는 물리적 테이블이 생성되면, 사용자 단말에서 신규 애플리케이션의 개발을 위한 적어도 하나의 모듈과 물리적 매칭 관계를 형성하도록 메타 데이터를 매칭시키는 단계, 사용자 단말에서 적어도 하나의 모듈로 기본 위젯을 생성하고, UI(User Interface)를 기본 위젯에 포함된 도구툴바로 생성하는 경우, 물리적 테이블의 컬럼이 요구되는지의 여부를 확인하는 단계, 및 물리적 테이블의 컬럼이 요구되는 경우, 클라우드 서버로 생성된 UI에 대응하는 테이블 컬럼을 생성하도록 요청하는 단계를 포함한다.

발명의 효과

[0008] 전술한 본 발명의 과제 해결 수단 중 어느 하나에 의하면, RIA 기술을 적용한 클라이언트 애플리케이션과, API 기반의 서버 서비스로 분리하여 aPaaS 플랫폼 서비스의 유지 보수의 방법을 제공하고, UI 디자인 시 개발 도구와 물리적인 필드 관계를 설정하여 물리적인 테이블의 변경을 자동화할 수 있고, RESTful 통신규격의 URI와 HTTP 메소드를 이용하여 객체 화 서비스에 접근하여 데이터베이스의 처리를 자동으로 구현함으로써 별도의 서버 프로그램이 없을지라도 물리적인 테이블을 변경하도록 하고, 단순한 UI 구성만을 통해서 애플리케이션의 개발을 가능케함으로써 저비용 및 단시간 내의 개발을 완성토록 한다.

도면의 간단한 설명

[0009] 도 1은 본 발명의 일 실시예에 따른 RIA를 이용한 클라우드 기반 aPaaS 제공 시스템을 설명하기 위한 도면이다.
 도 2는 도 1의 시스템에 포함된 클라우드 기반 aPaaS 제공 서버와 사용자 단말의 RIA 엔진을 설명하기 위한 설명하기 위한 블록 구성도이다.
 도 3은 본 발명의 일 실시예에 따른 RIA를 이용한 클라우드 기반 aPaaS 중 사용자 단말에서 생성된 기본 위젯의 모듈을 설명하기 위한 도면이다.
 도 4는 본 발명의 일 실시예에 따른 RIA를 이용한 클라우드 기반 aPaaS 중 사용자 단말에서 메타 데이터를 이용하여 애플리케이션 객체를 생성하는 관계를 설명하기 위한 도면이다.
 도 5는 본 발명의 일 실시예에 따른 RIA를 이용한 클라우드 기반 aPaaS 중 사용자 단말에서 위젯을 생성하기 위한 메타 데이터의 속성을 설명하기 위한 도면이다.
 도 6은 본 발명의 일 실시예에 따른 RIA를 이용한 클라우드 기반 aPaaS 중 사용자 단말에서 출력된 위젯과 위젯 런처 및 도구툴바를 설명하기 위한 도면이다.
 도 7은 본 발명의 일 실시예에 따른 RIA를 이용한 클라우드 기반 aPaaS 중 사용자 단말에서 애플리케이션을 개발하는 과정을 설명하기 위한 도면이다.
 도 8은 본 발명의 일 실시예에 따른 RIA를 이용한 클라우드 기반 aPaaS 중 메타 데이터와 물리적 테이블 간의 연동관계를 설명하기 위한 프로그램 화면이다.
 도 9는 본 발명의 일 실시예에 따른 RIA를 이용한 클라우드 기반 aPaaS 제공 방법을 설명하기 위한 동작 흐름도이다.

발명을 실시하기 위한 구체적인 내용

- [0010] 아래에서는 첨부한 도면을 참조하여 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자가 용이하게 실시할 수 있도록 본 발명의 실시예를 상세히 설명한다. 그러나 본 발명은 여러 가지 상이한 형태로 구현될 수 있으며 여기에서 설명하는 실시예에 한정되지 않는다. 그리고 도면에서 본 발명을 명확하게 설명하기 위해서 설명과 관계없는 부분은 생략하였으며, 명세서 전체를 통하여 유사한 부분에 대해서는 유사한 도면 부호를 붙였다.
- [0011] 명세서 전체에서, 어떤 부분이 다른 부분과 "연결"되어 있다고 할 때, 이는 "직접적으로 연결"되어 있는 경우뿐 아니라, 그 중간에 다른 소자를 사이에 두고 "전기적으로 연결"되어 있는 경우도 포함한다. 또한 어떤 부분이 어떤 구성요소를 "포함"한다고 할 때, 이는 특별히 반대되는 기재가 없는 한 다른 구성요소를 제외하는 것이 아니라 다른 구성요소를 더 포함할 수 있는 것을 의미하며, 하나 또는 그 이상의 다른 특징이나 숫자, 단계, 동작, 구성요소, 부분품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.
- [0012] 명세서 전체에서 사용되는 정도의 용어 "약", "실질적으로" 등은 언급된 의미에 고유한 제조 및 물질 허용오차가 제시될 때 그 수치에서 또는 그 수치에 근접한 의미로 사용되고, 본 발명의 이해를 돕기 위해 정확하거나 절대적인 수치가 언급된 개시 내용을 비양심적인 침해자가 부당하게 이용하는 것을 방지하기 위해 사용된다. 본 발명의 명세서 전체에서 사용되는 정도의 용어 "~(하는) 단계" 또는 "~의 단계"는 "~를 위한 단계"를 의미하지 않는다.
- [0013] 본 명세서에 있어서 '부(部)'란, 하드웨어에 의해 실현되는 유닛(unit), 소프트웨어에 의해 실현되는 유닛, 양방을 이용하여 실현되는 유닛을 포함한다. 또한, 1개의 유닛이 2개 이상의 하드웨어를 이용하여 실현되어도 되고, 2개 이상의 유닛이 1개의 하드웨어에 의해 실현되어도 된다.
- [0014] 본 명세서에 있어서 단말, 장치 또는 디바이스가 수행하는 것으로 기술된 동작이나 기능 중 일부는 해당 단말, 장치 또는 디바이스와 연결된 서버에서 대신 수행될 수도 있다. 이와 마찬가지로, 서버가 수행하는 것으로 기술된 동작이나 기능 중 일부도 해당 서버와 연결된 단말, 장치 또는 디바이스에서 수행될 수도 있다.
- [0015] 본 명세서에서 있어서, 단말과 매핑(Mapping) 또는 매칭(Matching)으로 기술된 동작이나 기능 중 일부는, 단말의 식별 정보(Identifying Data)인 단말기의 고유번호나 개인의 식별정보를 매핑 또는 매칭한다는 의미로 해석될 수 있다.
- [0016] 이하 첨부된 도면을 참고하여 본 발명을 상세히 설명하기로 한다.
- [0017] 도 1은 본 발명의 일 실시예에 따른 RIA를 이용한 클라우드 기반 aPaaS 제공 시스템을 설명하기 위한 도면이다. 도 1을 참조하면, RIA를 이용한 클라우드 기반 aPaaS 제공 시스템(1)은, 사용자 단말(100), 클라우드 기반 aPaaS 제공 서버(300), 클라우드 서버(400)를 포함할 수 있다. 다만, 이러한 도 1의 RIA를 이용한 클라우드 기반 aPaaS 제공 시스템(1)은, 본 발명의 일 실시예에 불과하므로, 도 1을 통하여 본 발명이 한정 해석되는 것은 아니다.
- [0018] 이때, 도 1의 각 구성요소들은 일반적으로 네트워크(network, 200)를 통해 연결된다. 예를 들어, 도 1에 도시된 바와 같이, 사용자 단말(100)은 네트워크(200)를 통하여 클라우드 서버(400), 클라우드 기반 aPaaS 제공 서버(300)와 연결될 수 있다. 그리고, 클라우드 기반 aPaaS 제공 서버(300)는, 네트워크(200)를 통하여 사용자 단말(100), 클라우드 서버(400)와 연결될 수 있다. 또한, 클라우드 서버(400)는, 네트워크(200)를 통하여 사용자 단말(100), 클라우드 기반 aPaaS 제공 서버(300)와 연결될 수 있다.
- [0019] 본 발명의 일 실시예에 따른 출차 대기 단축 서비스 제공 시스템(1)에 포함된 사용자 단말(100), 클라우드 기반 aPaaS 제공 서버(300) 및 클라우드 서버(400)는 적어도 하나로 구성될 수 있으며, 단수 또는 복수일 수 있다. 따라서, 이하에서 단수로 명명하더라도 복수를 의미할 수도 있다.
- [0020] 여기서, 네트워크는, 복수의 단말 및 서버들과 같은 각각의 노드 상호 간에 정보 교환이 가능한 연결 구조를 의미하는 것으로, 이러한 네트워크의 일 예에는 RF, 3GPP(3rd Generation Partnership Project) 네트워크, LTE(Long Term Evolution) 네트워크, 5GPP(5rd Generation Partnership Project) 네트워크, WIMAX(World Interoperability for Microwave Access) 네트워크, 인터넷(Internet), LAN(Local Area Network), Wireless LAN(Wireless Local Area Network), WAN(Wide Area Network), PAN(Personal Area Network), 블루투스(Bluetooth) 네트워크, NFC 네트워크, 위성 방송 네트워크, 아날로그 방송 네트워크, DMB(Digital Multimedia

Broadcasting) 네트워크 등이 포함되나 이에 한정되지는 않는다.

[0021] 사용자 단말(100)은, RIA를 이용한 클라우드 기반 aPaaS 관련 웹 페이지, 앱 페이지, 프로그램 또는 애플리케이션을 이용하여 애플리케이션을 개발하고자 하는 개발자의 단말일 수 있다. 그리고, 사용자 단말(100)은, RIA(Rich Internet Application) 엔진을 실행 환경을 구현하고, 개발 환경으로 전환한 다음, 개발 환경에서 생성한 위젯(Widget)의 정보를 메타 데이터(메타정보)로 추출하여 REST(Representational State Transfer)ful 통신규격으로 클라우드 기반 aPaaS 제공 서버(300)로 전송하는 단말일 수 있다. 그리고, 사용자 단말(100)은, 클라우드 서버(400)에 생성된 위젯에 대응하는 물리적 테이블 내의 속성이나 컬럼(Column)을 위젯을 변경함으로써 자동변경되도록 하는 단말일 수 있다. 이를 통하여, 사용자 단말(100)을 이용하는 개발자는, 클라우드 서버(400) 내의 물리적 계층까지 접근하지 않아도 되며, 데이터베이스나 SQL(Structured Query Language)에 능숙하지 않을지라도 위젯의 드래그 앤 드롭으로 손쉽게 애플리케이션을 개발할 수 있고, 개발자의 능력에 따라 상이한 수준의 애플리케이션이 개발되는 것을 방지할 수 있다.

[0022] 여기서, 사용자 단말(100)은, 네트워크를 통하여 원격지의 서버나 단말에 접속할 수 있는 컴퓨터로 구현될 수 있다. 여기서, 컴퓨터는 예를 들어, 네비게이션, 웹 브라우저(WEB Browser)가 탑재된 노트북, 데스크톱(Desktop), 랩톱(Laptop) 등을 포함할 수 있다. 이때, 사용자 단말(100)은, 네트워크를 통해 원격지의 서버나 단말에 접속할 수 있는 단말로 구현될 수 있다. 사용자 단말(100)은, 예를 들어, 휴대성과 이동성이 보장되는 무선 통신 장치로서, 네비게이션, PCS(Personal Communication System), GSM(Global System for Mobile communications), PDC(Personal Digital Cellular), PHS(Personal Handyphone System), PDA(Personal Digital Assistant), IMT(International Mobile Telecommunication)-2000, CDMA(Code Division Multiple Access)-2000, W-CDMA(W-Code Division Multiple Access), Wibro(Wireless Broadband Internet) 단말, 스마트폰(smartphone), 스마트 패드(smartpad), 타블렛 PC(Tablet PC) 등과 같은 모든 종류의 핸드헬드(Handheld) 기반의 무선 통신 장치를 포함할 수 있다.

[0023] 클라우드 기반 aPaaS 제공 서버(300)는, RIA를 이용한 클라우드 기반 aPaaS 웹 페이지, 앱 페이지, 프로그램 또는 애플리케이션을 제공하는 서버일 수 있다. 그리고, 클라우드 기반 aPaaS 제공 서버(300)는, 개발자인 사용자가 사용자 단말(100)에서 애플리케이션을 포함한 소프트웨어를 개발할 수 있는 토대(플랫폼)를 제공해 주는 서버일 수 있다. 그리고, 클라우드 기반 aPaaS 제공 서버(300)는, 사용자 단말(100)로 RIA 엔진을 제공하고, 사용자 단말(100)에서 위젯, 컴포넌트를 이용하여 UI(User Interface)를 생성할 수 있도록 하며, 생성된 애플리케이션 객체는 메타 데이터와 클라우드 서버(400)의 물리적 테이블과 연동되어 사용자 단말(100) 단에서 위젯 등을 변경하는 간단한 방법으로, 클라우드 서버(400)의 물리적 테이블을 자동으로 변경하도록 하는 서버일 수 있다. 이에 따라, 사용자 단말(100)의 개발자는, 데이터베이스나, 클라우드 서비스에 대한 지식이 없을지라도 클라우드 기반으로 애플리케이션을 개발할 수 있도록 지원받을 수 있다. 또한, 클라우드 기반 aPaaS 제공 서버(300)는, PaaS(Platform as a Service)를 통해 서비스 구성 컴포넌트 및 호환성 제공 서비스를 지원하는 서버일 수 있다. 그리고, 클라우드 기반 aPaaS 제공 서버(300)는, 컴파일 언어, 웹 프로그램, 제작 툴, 데이터베이스 인터페이스, 과금 모듈, 사용자 관리모듈 등을 포함할 수 있다. 이에 따라, 사용자 단말(100)인 응용 서비스 개발자는, 이러한 플랫폼 상에서 데이터베이스와 애플리케이션 서버, 파일시스템과 관련한 솔루션 등 미들웨어까지 확장된 IT 자원을 활용하여 새로운 애플리케이션을 만들어 사용할 수 있다. 이때, 클라우드 기반 aPaaS 제공 서버(300)는, 클라우드 서버(400)와 일체로 형성되는 경우, 클라우드 서버(400)의 기능은 모두 클라우드 기반 aPaaS 제공 서버(300)로 포함될 수 있고, 다양한 구현 실시예에 따라 달라질 수 있음은 자명하다 할 것이다.

[0024] 여기서, 클라우드 기반 aPaaS 제공 서버(300)는, 네트워크를 통하여 원격지의 서버나 단말에 접속할 수 있는 컴퓨터로 구현될 수 있다. 여기서, 컴퓨터는 예를 들어, 네비게이션, 웹 브라우저(WEB Browser)가 탑재된 노트북, 데스크톱(Desktop), 랩톱(Laptop) 등을 포함할 수 있다.

[0025] 클라우드 서버(400)는, 사용자 단말(100)에서 위젯을 생성하고, 모듈이나 컴포넌트를 제작할 때, 기본 위젯에 대한 메타 데이터를 클라우드 기반 aPaaS 제공 서버(300)를 통하여 수신하고, 물리적 테이블을 생성하는 서버일 수 있다. 또한, 클라우드 서버(400)는, 물리적 테이블에 대응하는 위젯, 컴포넌트, 모듈이 변경되는 경우, 변경에 대응하도록 물리적 테이블의 컬럼을 생성하거나, 속성을 저장하는 서버일 수 있다. 또한, 클라우드 서버(400)는 사용자 단말(100)로부터 클라우드 기반 aPaaS 제공 서버(300)를 통하여 수신한 위젯에 입력된 정보에 기반하여, 동적으로 SQL을 생성하여 데이터베이스에 자동으로 처리될 수 있도록 하는 서버일 수 있다.

[0026] 여기서, 클라우드 서버(400)는, 네트워크를 통하여 원격지의 서버나 단말에 접속할 수 있는 컴퓨터로 구현될 수

있다. 여기서, 컴퓨터는 예를 들어, 네비게이션, 웹 브라우저(WEB Browser)가 탑재된 노트북, 데스크톱(Desktop), 랩톱(Laptop) 등을 포함할 수 있다.

[0027] 도 2는 도 1의 시스템에 포함된 클라우드 기반 aPaaS 제공 서버와 사용자 단말의 RIA 엔진을 설명하기 위한 설명하기 위한 블록 구성도이고, 도 3 내지 도 8은 본 발명의 일 실시예에 따른 RIA를 이용한 클라우드 기반 aPaaS가 구현되는 일 실시예를 설명하기 위한 도면이다.

[0028] 도 2를 참조하면, 사용자 단말(100)의 RIA 엔진은, 위젯 런처부(110), 위젯 렌더링부(120), 위젯 메타정보부(130), 모듈 카탈로그부(140), 컴포넌트 카탈로그부(150), 통신부(160) 및 테이블 정보부(170)를 포함할 수 있다. 그리고, 클라우드 기반 aPaaS 제공 서버(300)는, 전송부(310), 요청부(311), 매칭부(320), 확인부(321), 동기화부(330), 자동 생성부(340), 연결부(350), 애플리케이션 메타 데이터 저장부(360) 및 컴포넌트 저장부(370)를 포함할 수 있다. 또한, 클라우드 서버(400)는 물리적 테이블(410)을 포함할 수 있다.

[0029] 본 발명의 일 실시예에 따른 클라우드 기반 aPaaS 제공 서버(300)나 연동되어 동작하는 다른 서버(미도시)가 사용자 단말(100)로 RIA를 이용한 클라우드 기반 aPaaS 애플리케이션, 프로그램, 앱 페이지, 웹 페이지 등을 전송하는 경우, 사용자 단말(100)은, RIA를 이용한 클라우드 기반 aPaaS 애플리케이션, 프로그램, 앱 페이지, 웹 페이지 등을 설치하거나 열 수 있다. 또한, 웹 브라우저에서 실행되는 스크립트를 이용하여 서비스 프로그램이 사용자 단말(100)에서 구동될 수도 있다. 여기서, 웹 브라우저는 웹(WWW: world wide web) 서비스를 이용할 수 있게 하는 프로그램으로 HTML(hyper text mark-up language)로 서술된 하이퍼 텍스트를 받아서 보여주는 프로그램을 의미하며, 예를 들어 넷스케이프(Netscape), 익스플로러(Explorer), 크롬(chrome) 등을 포함한다. 또한, 애플리케이션은 단말 상의 응용 프로그램(application)을 의미하며, 예를 들어, 모바일 단말(스마트폰)에서 실행되는 앱(app)을 포함한다.

[0030] 본 발명의 일 실시예에서 사용하는 용어인, aPaaS(Application Platform Services)는, 사용자들이 소프트웨어를 스스로 만들어 쓰게끔 웹 기반의 환경을 제공하는 서비스라 할 수 있다. PaaS(Platform as a Service)는 개발 플랫폼을 제공하는 서비스, SaaS(Software as a Service)는 엔드유저를 위해 바로 사용 가능한 소프트웨어 형태의 서비스인데 이 둘이 결합된 중간 형태이면서 개발의 주체가 실제 사용자인 소비자가 되는 것이 aPaaS라고 정의하기로 한다. 다만, 일반적인 aPaaS의 정의를 배제하는 것은 아님은 자명하다 할 것이다.

[0031] 본 발명의 일 실시예에서 사용하는 용어인, PaaS(Platform as a Service)는 클라우드 컴퓨팅 서비스의 분류의 하나로서, 일반적으로 앱의 개발 및 시작과 관련된 인프라를 만들고 유지 보수하는 복잡함없이 고객이 애플리케이션을 개발, 실행, 관리할 수 있게 하는 플랫폼을 제공한다. SaaS의 개념을 개발 플랫폼에도 확장한 방식으로, 개발을 위한 플랫폼 구축을 할 필요 없이 필요한 개발 요소들을 웹에서 쉽게 빌려쓸 수 있게 하는 모델로 정의한다. 다만, 일반적인 PaaS의 정의를 배제하는 것은 아님은 자명하다 할 것이다.

[0032] 도 2를 참조하면, 전송부(310)는, 사용자 단말(100)에서 신규 애플리케이션을 위한 RIA(Rich Internet Application)엔진을 요청하는 경우, 요청에 대응하여 RIA 엔진을 전송할 수 있다. 이때, RIA(Rich Internet Application; RIA)는, 웹 애플리케이션의 장점은 유지하면서 기존 웹 브라우저 기반 인터페이스의 단점인 늦은 응답 속도, 데스크톱 애플리케이션에 비해 떨어지는 조작성 등을 개선하기 위한 기술의 통칭이다. 즉, 별도의 설치가 필요 없는 웹 브라우저 기반의 애플리케이션 배포 장점과 서버 측 웹 서비스와의 연동, 마크업 언어 기반의 선언적 애플리케이션 구성 등은 유지하면서 데스크톱 애플리케이션과 대등한 사용자 경험을 주는 것을 목표로 하는 기술이다.

[0033] 요청부(311)는, 사용자 단말(100)에서 RIA 엔진을 실행하여 실행 환경을 생성하고, 실행 환경에서 개발 환경으로 변환하는 경우, 클라우드 서버(400)로 테이블 생성을 요청할 수 있다. 이때, 사용자 단말(100)의 RIA 엔진을 설명하면, 사용자 단말(100)은, 본 발명의 일 실시예에 따른 aPaaS서비스에 접속하게 되면 사용자 단말(100)에 맞는 RIA기반의 실행 엔진을 다운로드받는다. 이때, RIA 엔진은 위젯런처부(110)와 위젯렌더링부(120), 위젯 메타정보부(130), 모듈 카탈로그부(140), 컴포넌트 카탈로그부(150), 통신부(160) 및 테이블 정보부(170)로 구성될 수 있으나, 이에 한정되는 것은 아니다. 이때, 각각의 모듈, 위젯, 메타 정보, 컴포넌트 등은 사용자의 요청으로 클라우드 기반 aPaaS 제공 서버(300)로부터 각각 수신되어 저장될 수 있고, 테이블 정보부(170)는 이후에 기본 애플리케이션의 위젯이 생성되고, 각 컴포넌트와 모듈이 설정되면, 설정된 각각의 구성요소에 대응하는 데이터베이스 계층의 물리적 테이블(410), 즉 클라우드 서버(400)로의 접근을 하지 않고도, 각 구성요소가 변경되는 경우 자동으로 변경되기 위해 매핑된 정보를 저장할 수 있다. 그리고, 통신부(160)는, REST(Representational State Transfer) 통신 규격을 이용하여 메타 데이터를 전송하게 되는데, 리소스인 URI와 메소드인 HTTP를 이용하여 객체화된 서비스에 접근하기 위해 구성된다. 즉, 통신부(160)는, 직관적인 URI를

이용해서 객체의 멤버를 추적하는 것처럼 리소스에 접근하여 HTTP 메소드를 이용해 그것을 조작하도록 한다.

- [0034] 보다 상세하게는, REST(Representational State Transfer)는 월드 와이드 웹(World Wide Web: WWW)과 같은 분산 하이퍼미디어 시스템을 위한 소프트웨어 아키텍처의 한 형식이다. 또한, REST는 네트워크 아키텍처 원리의 모음이다. 여기서 네트워크 아키텍처 원리란 자원을 정의하고 자원에 대한 주소를 지정하는 방법 전반을 일컫는다. 또한, REST는, 웹 상의 자료를 HTTP위에서 SOAP이나 쿠키를 통한 세션 트래킹 같은 별도의 전송 계층 없이 전송하기 위한 아주 간단한 인터페이스를 의미할 수도 있다.
- [0035] 요청부(311)는, 사용자 단말(100)에서 RIA 엔진을 실행하여 실행 환경을 생성하고, 실행 환경에서 개발 환경으로 변환하는 경우, 클라우드 서버(400)로 테이블 생성을 요청할 때, 사용자 단말(100)이 RIA 엔진을 실행하여 실행 환경을 생성할 때, 자바스크립트 실행모듈로 실행 환경을 생성하고, 사용자 단말(100)은, 클라우드 기반 aPaaS 제공 서버(300)로부터 다운로드받은 컴포넌트를 이용하여 자바스크립트 실행모듈을 구성할 수 있도록 한다. 이때, 사용자 단말(100)은, 컴포넌트를 이용하여 실행 환경을 구성할 때, 클라우드 기반 aPaaS 제공 서버(300)에 기 저장된 적어도 하나의 컴포넌트 중 어느 하나 또는 적어도 하나의 조합을 요청하고, 요청에 대한 응답으로 수신된 컴포넌트를 저장하여 실행 환경을 구성할 수 있다.
- [0036] 매칭부(320)는, 클라우드 서버(400)에서 테이블 생성에 대응하는 물리적 테이블이 생성되면, 사용자 단말(100)에서 신규 애플리케이션의 개발을 위한 적어도 하나의 모듈과 물리적 매칭 관계를 형성하도록 메타 데이터를 매칭시킬 수 있다. 즉, 사용자인 개발자는, 신규 애플리케이션을 개발할 수 있고, 애플리케이션 템플릿을 이용하여 필요한 애플리케이션을 개발하게 되는데, 사용자가 애플리케이션을 개발하려고 하면 템플릿은 자신의 템플릿에서 사용자가 사용할수 있는 개발도구를 표시하고 사용자는 개발도구를 이용하여 화면을 디자인하게 한다. 이때, 사용자가 디자인하는 과정에서 새로운 데이터항목이 필요하다고 판단되는 경우, 클라우드 기반 aPaaS 제공 서버(300)에 필드 항목을 생성을 요청하게 되고, 클라우드 기반 aPaaS 제공 서버(300)는, 클라우드 서버(400)에 이를 통보하고 클라우드 서버(400)는 신규 애플리케이션으로 기 생성된 물리적 테이블에 하나의 구성인 물리적인 컬럼을 생성하게 된다.
- [0037] 그리고 나서, 사용자 단말(100)의 개발자는 템플릿의 개발도구를 이용하여 새로운 화면을 완성하게 되고 이때 자동으로 그와 관련된 물리적인 데이터 구조는 자동으로 완성이 되는 구조이므로, 클라우드 기반 aPaaS 제공 서버(300)에서 해당 구조를 자동으로 처리하게 됨으로 실질적인 데이터베이스의 물리적인 구조를 이해할 필요가 없다.
- [0038] 확인부(321)는, 사용자 단말(100)에서 적어도 하나의 모듈로 기본 위젯을 생성하고, UI(User Interface)를 기본 위젯에 포함된 도구툴바로 생성하는 경우, 물리적 테이블의 컬럼이 요구되는지의 여부를 확인할 수 있다.
- [0039] 동기화부(330)는, 물리적 테이블의 컬럼이 요구되는 경우, 클라우드 서버(400)로 생성된 UI에 대응하는 테이블 컬럼을 생성하도록 요청할 수 있다.
- [0040] 자동 생성부(340)는, 동기화부(330)에서 물리적 테이블의 컬럼이 요구되는 경우, 클라우드 서버(400)로 생성된 UI에 대응하는 테이블 컬럼을 생성하도록 요청한 후, 물리적 테이블에 추가된 테이블 컬럼과 UI의 물리적 매칭 관계를 형성하도록 메타 데이터를 매칭시킬 수 있고, 매칭된 UI, 추가된 테이블 및 메타 데이터를 사용자 단말(100)의 위젯 정보에 저장하도록 할 수 있다.
- [0041] 이때, 메타 데이터는, 적어도 하나의 모듈의 표시속성, 데이터 속성 및 처리함수를 포함하고, 클라우드 기반 aPaaS 제공 서버(300)와 사용자 단말(100)은 메타 데이터를 저장할 수 있다. 이는 각각, 위젯 메타 정보부(130)와 애플리케이션 메타데이터 저장부(360)에 저장되게 된다.
- [0042] 그리고, 적어도 하나의 모듈은 신규 애플리케이션의 객체를 생성하는 모듈이고, 객체의 클래스를 설정하는 클래스 모듈과 메타 데이터와 연동될 수 있다.
- [0043] 연결부(350)는, 요청부(311)에서 물리적 테이블의 컬럼이 요구되는 경우, 클라우드 서버(400)로 생성된 UI에 대응하는 테이블 컬럼을 생성하도록 요청한 후, 기본 위젯을 변경할 때, 사용자 단말(100)에서 입력한 정보는, REST(Representational State Transfer)ful 통신규격으로 변환되어 수신되도록 제어할 수 있다. 이때, REST(Representational State Transfer) 통신규격의 리소스인 URI와 메소드(Method)인 HTTP에 기반하여 클라우드 서버(400)의 물리적 테이블과 변경된 기본 위젯 간의 관계를 연결할 수 있다. 이때, 클라우드 기반 aPaaS 제공 서버(300)는, HTTP와, 기본 위젯을 변경하기 위해 사용자 단말(100)에서 입력한 정보에 대응하는 파라미터를 기반으로, 동적으로 SQL을 생성하여 물리적 테이블과 기본 위젯 간의 관계를 설정할 수 있다.

- [0044] 이때, 사용자 단말(100)에서 신규로 생성한 애플리케이션의 애플리케이션 영역은 RIA 엔진의 애플리케이션 풀(Pool)에서 이를 관리하고 풀의 참조 주소에 의해서 애플리케이션의 영역간의 정보를 참조하게 된다. 이때, 사용자 단말(100)에서 애플리케이션의 자료를 변경하고, 클라우드 기반 aPaaS 제공 서버(300)에 저장하고자 할 경우에는, 애플리케이션 영역은 자동으로 사용자 단말(100)에서 발생하는 작업 사항을 인지하고, 이를 자동으로 RESTful 통신규격으로 클라우드 기반 aPaaS 제공 서버(300)로 전달하게 된다.
- [0045] 그리고, 클라우드 기반 aPaaS 제공 서버(300)는, RESTful 통신규격으로 전달된 요청을 받고, 사용자의 권한이 존재하는지의 여부를 확인하며, 이상이 없는 경우 실행되도록 한다. 이때, 클라우드 기반 aPaaS 제공 서버(300)는, 전달된 정보를 기반으로 클라우드 서버(400)의 데이터베이스에 전달하게 되고, 데이터베이스는 SQL로 변환하여 실행한다.
- [0046] 정리하면, 사용자 단말(100)의 사용자는, 자신이 사용자 단말(100)을 이용하여 화면에서 필요한 정보를 입력하고, 자료의 저장을 요청하면, 클라우드 기반 aPaaS 제공 서버(300)는, 사용자가 입력한 자료를 REST 통신규격 방식으로 수신하고, 클라우드 기반 aPaaS 제공 서버(300)는, 물리적인 SQL로 변환 및 데이터를 저장하는 과정을 수행하게 되는 것이다.
- [0047] 연결부(350)는, 요청부(311)에서 물리적 테이블의 컬럼이 요구되는 경우, 클라우드 서버(400)로 생성된 UI에 대응하는 테이블 컬럼을 생성하도록 요청한 후, 사용자 단말(100)에서 클라우드 서버(400)의 물리적 테이블을 수정, 변경 및 삭제를 포함하는 편집을 수행한 경우, 편집에 대응되도록 기본 위젯을 변경할 수 있다.
- [0048] 한편, 사용자 단말(100)에서 스크립트 편집기를 요청하는 경우, 클라우드 기반 aPaaS 제공 서버(300)는, 요청에 대응하도록 사용자 단말(100)로 기 저장된 스크립트 편집기를 제공하고, 사용자 단말(100)에서 실행되는 스크립트 편집기의 스크립트는, 사용자 단말(100) 및 클라우드 기반 aPaaS 제공 서버(300)에서 연동되도록 저장될 수 있다.
- [0049] 그리고, 적어도 하나의 모듈은, 클라우드 기반 aPaaS 제공 서버(300)에 동작패턴, 기능 및 속성이 기 정의되어 저장될 수 있고, 사용자 단말(100)에서 애플리케이션을 개발할 때 별도로 정의하는 경우, 적어도 하나의 모듈에 물리적 테이블에 삽입될 컬럼의 속성을 추가하는 경우, 클라우드 서버(400)의 물리적 테이블의 컬럼의 속성이 추가될 수 있다., 또한, 사용자 단말(100)에서 기본 위젯의 편집, 수정 및 삭제는 클라우드 기반 aPaaS 제공 서버(300)에서 제공된 컴포넌트 카탈로그의 드래그 앤 드롭으로 이루어질 수 있다. 이를 위하여, 클라우드 기반 aPaaS 제공 서버(300)는 컴포넌트 저장부(370)를 포함하고, 사용자 단말(100)은 선택적으로 요청한 컴포넌트를 컴포넌트 카탈로그부(150)에 저장하게 된다.
- [0050] 이를 통하여, 개발자가 신규 애플리케이션을 개발할 때, 데이터베이스, SQL, 물리계층에 대한 이해가 없을지라도 UI의 간단한 드래그 앤 드롭으로 애플리케이션을 제작할 수 있다.
- [0051] 이하, 상술한 도 2의 클라우드 기반 aPaaS 제공 서버의 구성에 따른 동작 과정을 도 3 내지 도 8을 예로 들어 상세히 설명하기로 한다. 다만, 실시에는 본 발명의 다양한 실시예 중 어느 하나일 뿐, 이에 한정되지 않음은 자명하다 할 것이다.
- [0052] 도 3은 본 발명의 일 실시예에 따른 RIA를 이용한 클라우드 기반 aPaaS 중 사용자 단말에서 생성된 기본 위젯의 모듈을 설명하기 위한 도면이다. 도 3을 참조하면, 모듈의 종류를 볼 수 있다. 즉, 모듈은, 그리드, 폼, 검색 팝업, 탭그리드, 팝업폼 등을 포함할 수 있다. 여기서, 기본적인 조회, 추가, 수정, 삭제, 검색필터의 기능 및 동작을 위한 메소드가 구현되어 있다.
- [0053] 애플리케이션을 개발할 때에는, 동작 패턴을 통일화시키는 모듈을 기반으로 위젯을 생성하게 되는데, 대부분의 애플리케이션은 그리드를 기반으로 조회, 수정을 하고, 폼을 기반으로, 입력, 수정, 삭제 및 작업 등의 동일한 동작 패턴을 가지게 된다. 이때, 업무성격에 따라 물리적인 테이블만 변경될 뿐, 데이터의 처리를 위한 동작은 동일하다. 모듈은, 그리드형, 폼형, 팝업검색형, 탭조회형 등의 동일한 동작패턴으로 정의되는데, 모듈에 관련된 동작 패턴을 미리 정의한 후, 개발시에는 모듈에 물리적 테이블의 컬럼 속성을 추가하여 완성하게 하는 것이다. 이는, 개발 과정의 동일한 작업에 대하여 별도의 동작 스크립트를 구현하지 않게 하여, 드래그 앤 드롭(Drag and Drop)만으로도 데이터 처리가 가능하고, 드래그 앤 드롭만으로도 데이터 처리가 가능한 완성된 애플리케이션을 포함한 소프트웨어를 개발할 수 있게 한다.
- [0054] 도 4는 본 발명의 일 실시예에 따른 RIA를 이용한 클라우드 기반 aPaaS 중 사용자 단말에서 메타 데이터를 이용하여 애플리케이션 객체를 생성하는 관계를 설명하기 위한 도면이고, 도 5는 본 발명의 일 실시예에 따른 RIA를

이용한 클라우드 기반 aPaaS 중 사용자 단말에서 위젯을 생성하기 위한 메타 데이터의 속성을 설명하기 위한 도면이고, 도 6은 본 발명의 일 실시예에 따른 RIA를 이용한 클라우드 기반 aPaaS 중 사용자 단말에서 출력된 위젯과 위젯 런처 및 도구툴바를 설명하기 위한 도면이다.

- [0055] 도 4를 참조하면, 위젯은 모듈 클래스와 메타정보를 이용하여 애플리케이션 객체로 생성하는 관계를 알 수 있다. 또한, 도 5를 참조하면, 위젯을 생성하기 위한 메타정보를 도시하는데, 표시속성, 데이터속성, 처리함수로 구성될 수 있다. 메타정보는 관리형태에 따라서 JSON, XML, TEXT 등의 포맷으로 관리될 수 있다. 데이터속성은, 필드에는 매칭되는 테이블의 컬럼의 정보와, 타입에는 화면표시를 위한 컴포넌트 종류를 포함한다. 이때, 필수입력여부, 숨김여부, 적용되는 공통코드의 정보를 포함할 수 있다. 이는, 위젯 렌더링 때, 매칭되는 컴포넌트와 그에 대한 동작의 정보를 나타낸다.
- [0056] 도 6은 위젯 런처가 실행되는 화면인데, 메타 데이터(메타정보)는 위젯 렌더링을 통하여 프로그램 형태로 변환이 되고, 변환된 결과물은 위젯런처(Launcher)를 통하여 사용자 단말(100)의 화면에 표시되게 된다. 이때, 메타정보는, HTML이나 자바스크립트(javascript)의 프로그램 형태로 변환될수 있다. 메타정보의 위젯속성에서 모듈타입을 지정할수 있으며, 모듈타입은, 상술한 바와 같이, 그리드, 폼, 검색팝업, 탭그리드, 팝업폼 등의 화면 구성 형태이다.
- [0057] 그리고, 메타정보의 컬럼속성은 모듈타입에 관한 상세 구성요소를 저장하는 것으로 필드명칭, 표시라벨, 화면컴포넌트, 크기, 위치, 읽기전용, 필수입력요소, 선택코드종류 등의 세부속성값을 가지며, 화면 컴포넌트 타입은 데이터처리를 위한 문자형, 숫자형, 지도형, 날짜형, 콤보형 등 화면디자인을 위한 메모형, 이미지형, html형 등으로 구성되며 컴포넌트 타입은 사용자 정의등을 통해서 추가될 수 있다.
- [0058] 또한, 위젯렌더링은 메타정보의 모듈타입에 따라서 타입렌더링(그리드,폼 등)을 실시하고 컬럼속성을 이용하여 컴포넌트 카다로그에 있는 컴포넌트 도구를 이용하여 해당 속성에 맞는 속성 렌더링을 실시한 후 기능처리함수를 병합하여 최종적인 실행 애플리케이션 위젯을 만들어낼 수 있다.
- [0059] 이때, 위젯 런처는, 사용자가 선택한 메뉴를 활성화하고, 활성화된 위젯에서 사용자가 입출력한 정보를 자동으로 인지함과 동시에, 변경, 추가,삭제 등의 작업에 따른 동적인 변수를 추출하고, 이를 RESTful 통신방식으로 전달하고, 그 이후의 과정은 상술한 바와 같으므로 생략한다.
- [0060] 도 7은 본 발명의 일 실시예에 따른 RIA를 이용한 클라우드 기반 aPaaS 중 사용자 단말에서 애플리케이션을 개발하는 과정을 설명하기 위한 도면이다. 도 7을 참조하여 정리하면, 클라우드 기반 aPaaS 제공 서버(300)는, 사용자 단말(100)에서 신규로 애플리케이션을 개발하고자 하면, 애플리케이션의 명칭을 기재하게 됨으로써 기본 위젯을 생성하게 된다. 이때, 기본 위젯은 상술한 바와 같이, 그리드, 폼, 검색팝업, 탭그리드, 팝업폼 등을 포함할 수 있다.
- [0061] 그리고 나서, 사용자 단말(100)은, 기본위젯 중에 폼 위젯을 렌더링해서 위젯 런처에 탑재하게 되고, 폼위젯에 해당되는 모듈의 개발도구(도구툴바)를 옆에 표시하여 추가적인 속성을 정의하게 한다. 이때, 개발도구는 컴포넌트 카달로그에 있는 도구 컴포넌트중에서 해당모듈에 적합한 도구들로 구성되며, 모듈에 따라서 구성형태가 달라질수 있다. 사용자 단말(100)에서 개발도구를 드래그하여 폼에 위치시키면, 해당 속성의 명칭을 입력하도록 하는 화면이 출력되고, 명칭이 입력되면 이를 클라우드 기반 aPaaS 제공 서버(300)에 전달하여 물리적인 DB의 테이블에 속성을 추가하게 한다.
- [0062] 이때, 위젯 런처는 사용자 단말(100)에서 선택한 애플리케이션에 따라 출력되는데, 사용자 단말(100)에서 위젯 도구(개발도구, 도구툴바)를 드래그하여 위치하도록 하면, 위젯도구는, 지정한 속성정의창을 팝업으로 표시하고, 사용자는 팝업된 속성창에 필드의 정보를 입력하게 된다. 이때, 사용자 단말(100)은, 상세속성창을 이용하여 필드의 속성을 추가할 수 있고, 위젯 메타정보로 관리되고, 사용자 단말(100)에서 애플리케이션의 구성을 완료하면, 위젯은 자동으로 위젯 메타정보를 클라우드 기반 aPaaS 제공 서버(300)로 전송한다.
- [0063] 이렇게 완료가 되면, 사용자 단말(100)은 위젯렌더링을 통하여 실제로 애플리케이션이 유저 단말에서 구동이 될 때 어떠한 화면이 되는지를 출력할 수 있고, 위젯 렌더링을 통하여 메타데이터를 읽어들이 HTML을 생성하게 된다. 이때, 사용자의 실행환경에 대응하도록 렌더링이 가능할 수 있는데, 사용자 단말(100)이 PC용 브라우저이면, HTML로 렌더링이되고, 사용자 단말(100)이 스마트 폰과 같은 모바일 브라우저이면, 모바일 브라우저용 HTML로 렌더링이 된다. 마찬가지로, 일반 스마트폰 애플리케이션인 경우에는, 애플리케이션으로 렌더링될 수 있다. 이렇게, 사용자 단말(100)의 특성에 맞도록 렌더링을 할 수 있고, 사용자 환경에 적합한 최적의 애플리케이션을 실행시킬 수 있다.

- [0064] 본 발명의 일 실시예에 따른 서비스는, 하기의 서비스 대상에 적용할 수 있다. 즉, 적용가능한 대상은, API Management Services(apiPaaS), Application Development Services(adPaaS), Application Platform Services(aPaaS), Business Analytics Platform Services(baPaaS), Business Process Management Services(bpmPaaS), Business Rule Platform Services(brPaaS), Citizen Integrator Software as a Service(iSaaS), Communications Platform Services(cPaaS), Database Platform Services(dbPaaS), Data Broker Platform Services(dbrPaaS), Event-Processing Services(epPaaS), Function Platform Services(fPaaS), Enterprise Horizontal Portal Services(Portal PaaS), In-Memory Data Grid Services(imdgPaaS), Integration Platform Services(iPaaS), Internet of Things Platform Services(IoTPaaS), Managed File Transfer Services(mftPaaS), Master Data Management Services(mdmPaaS)를 포함할 수 있다. 다만, 이에 한정되는 것은 아니고, 실시예에 따라 PaaS 기반 모든 서비스에 적용될 수 있다.
- [0065] 도 8은 본 발명의 일 실시예에 따른 RIA를 이용한 클라우드 기반 aPaaS 중 메타 데이터와 물리적 테이블 간의 연동관계를 설명하기 위한 프로그램 화면이고, 물리적 테이블의 물리적 및 논리적인 정보가 매칭된 관계를 유지하도록 기재된 것을 알 수 있다.
- [0066] 정리하면, 본 발명의 일 실시예에 따른 서비스는, 사용자 단말(100)에서 신규 애플리케이션을 개발하게 되면 모듈 카탈로그에 등록된 모듈을 통해서 기본 애플리케이션의 위젯이 생성이 되고, 물리적인 테이블이 자동생성된다. 그리고, 사용자 단말(100)에서 컴포넌트 카탈로그를 통해 작업모듈에 맞는 개발도구가 표시되고, 사용자는 사용자 단말(100)에 표시된 개발도구를 드래그하여 위젯을 수정하게 되며, 개발도구에 매칭된 컬럼의 속성에 의해서 물리적인 컬럼이 생성되게 된다.
- [0067] 그리고, 사용자 단말(100)에서 위젯을 변경할 때에는, 클라우드 기반 aPaaS 제공 서버(300)에서 데이터베이스의 물리적인 컬럼과 관계를 구성하게 함으로써 위젯과 데이터베이스의 테이블은 매칭관계를 유지할 수 있다. 또한, 클라우드 기반 aPaaS 제공 서버(300)는, 매칭관계를 유지하게 함으로써 사용자 단말(100)에서 위젯에 입력한 정보를 REST 통신규격으로 변환하여 전달하면, REST의 URI와 HTTP를 이용하여 데이터베이스의 물리적 테이블 간의 관계를 연결할 수 있게 된다.
- [0068] 클라우드 기반 aPaaS 제공 서버(300)는, HTTP 메소드와 전달된 파라미터를 기반으로 동적으로 SQL을 생성하여 데이터베이스에 처리를 자동으로 할 수 있게 된다. 이때, 위젯은 데이터베이스의 물리적 테이블과 매칭된 관계를 유지하기 때문에 테이블의 변화에 대해서 설정된 메타가 자동으로 반영되게 된다. 위젯은 물리적, 논리적인 테이블의 정보를 가지고 있어야 하며 실행소스로 랜더링될 때 테이블 정보와 컬럼 정보를 파악하여 변화가 발생한 경우에 대한 처리가 가능하게 구성될 수 있다. 이는 사용자 단말(100)에서 사용자가 임의로 테이블을 변경한 경우 컬럼 정보를 파악하고 추가된 컬럼을 위젯에 반영해야 하기 때문이다. 또한, 비즈니스의 처리는 필요시에 스크립트로 구현이 가능할 수 있도록 스크립트편집기가 제공될 수 있고, 이를 통해서 비즈니스 로직이 구현될 수 있다. 이때 구현된 스크립트는 메타정보와 같이 저장될 수 있다.
- [0069] 이와 같은 도 2 내지 도 8의 RIA를 이용한 클라우드 기반 aPaaS 제공 방법에 대해서 설명되지 아니한 사항은 앞서 도 1을 통해 RIA를 이용한 클라우드 기반 aPaaS 제공 방법에 대하여 설명된 내용과 동일하거나 설명된 내용으로부터 용이하게 유추 가능하므로 이하 설명을 생략하도록 한다.
- [0070] 도 9는 본 발명의 일 실시예에 따른 도 1의 RIA를 이용한 클라우드 기반 aPaaS 제공 시스템에 포함된 각 구성들 상호 간에 데이터가 송수신되는 과정을 나타낸 도면이다. 이하, 도 9를 통해 각 구성들 상호간에 데이터가 송수신되는 과정의 일 예를 설명할 것이나, 이와 같은 실시예로 본원이 한정 해석되는 것은 아니며, 앞서 설명한 다양한 실시예들에 따라 도 9에 도시된 데이터가 송수신되는 과정이 변경될 수 있음은 기술분야에 속하는 당업자에게 자명하다.
- [0071] 도 9를 참조하면, 클라우드 기반 aPaaS 제공 서버는, 사용자 단말에서 신규 애플리케이션을 위한 RIA(Rich Internet Application)엔진을 요청하는 경우, 요청에 대응하여 RIA 엔진을 전송한다(S9100).
- [0072] 또한, 클라우드 기반 aPaaS 제공 서버는, 사용자 단말에서 RIA 엔진을 실행하여 실행 환경을 생성하고, 실행 환경에서 개발 환경으로 변환하는 경우, 클라우드 서버로 테이블 생성을 요청하고(S9200), 클라우드 서버에서 테이블 생성에 대응하는 물리적 테이블이 생성되면, 사용자 단말에서 신규 애플리케이션의 개발을 위한 적어도 하나의 모듈과 물리적 매칭 관계를 형성하도록 메타 데이터를 매칭시킨다(S9300).
- [0073] 그리고, 클라우드 기반 aPaaS 제공 서버는, 사용자 단말에서 적어도 하나의 모듈로 기본 위젯을 생성하고, UI(User Interface)를 기본 위젯에 포함된 도구툴바로 생성하는 경우, 물리적 테이블의 컬럼이 요구되는지의 여

부를 확인하고(S9400), 물리적 테이블의 컬럼이 요구되는 경우, 클라우드 서버로 생성된 UI에 대응하는 테이블 컬럼을 생성하도록 요청한다(S9500).

[0074] 상술한 단계들(S9100~S9500)간의 순서는 예시일 뿐, 이에 한정되지 않는다. 즉, 상술한 단계들(S9100~S9500)간의 순서는 상호 변동될 수 있으며, 이중 일부 단계들은 동시에 실행되거나 삭제될 수도 있다.

[0075] 이와 같은 도 9의 RIA를 이용한 클라우드 기반 aPaaS 제공 방법에 대해서 설명되지 아니한 사항은 앞서 도 1 내지 도 8을 통해 RIA를 이용한 클라우드 기반 aPaaS 제공 방법에 대하여 설명된 내용과 동일하거나 설명된 내용으로부터 용이하게 유추 가능하므로 이하 설명을 생략하도록 한다.

[0076] 도 9를 통해 설명된 일 실시예에 따른 RIA를 이용한 클라우드 기반 aPaaS 제공 방법은, 컴퓨터에 의해 실행되는 애플리케이션이나 프로그램 모듈과 같은 컴퓨터에 의해 실행가능한 명령어를 포함하는 기록 매체의 형태로도 구현될 수 있다. 컴퓨터 판독 가능 매체는 컴퓨터에 의해 액세스될 수 있는 임의의 가용 매체일 수 있고, 휘발성 및 비휘발성 매체, 분리형 및 비분리형 매체를 모두 포함한다. 또한, 컴퓨터 판독가능 매체는 컴퓨터 저장 매체를 모두 포함할 수 있다. 컴퓨터 저장 매체는 컴퓨터 판독가능 명령어, 데이터 구조, 프로그램 모듈 또는 기타 데이터와 같은 정보의 저장을 위한 임의의 방법 또는 기술로 구현된 휘발성 및 비휘발성, 분리형 및 비분리형 매체를 모두 포함한다.

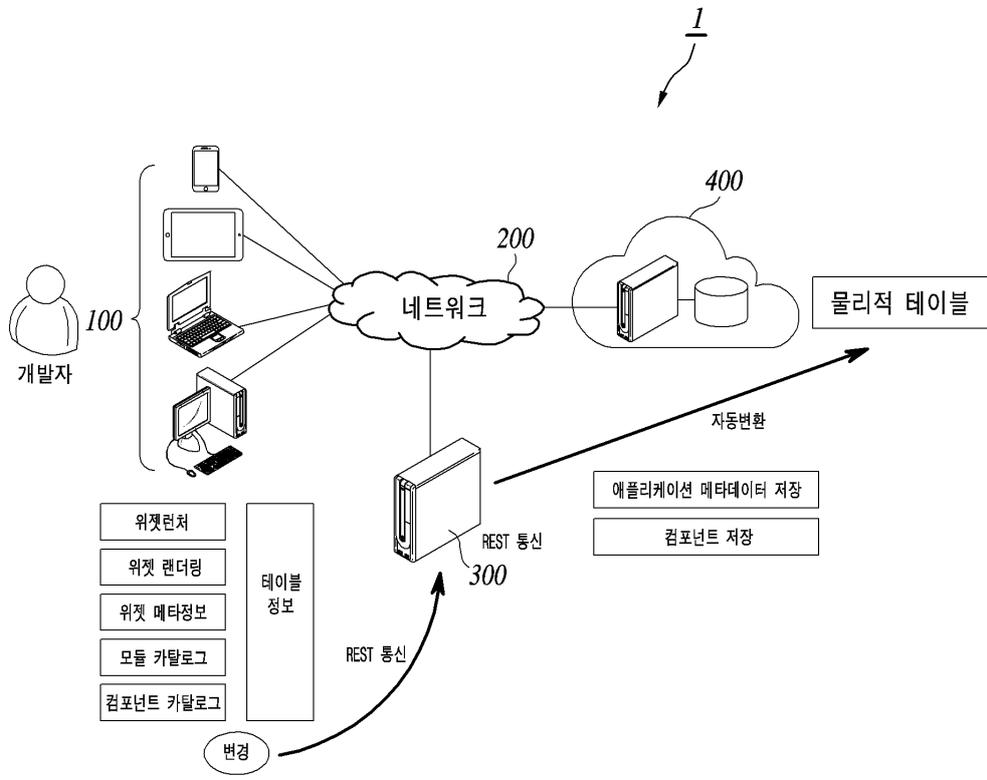
[0077] 전술한 본 발명의 일 실시예에 따른 RIA를 이용한 클라우드 기반 aPaaS 제공 방법은, 단말기에 기본적으로 설치된 애플리케이션(이는 단말기에 기본적으로 탑재된 플랫폼이나 운영체제 등에 포함된 프로그램을 포함할 수 있음)에 의해 실행될 수 있고, 사용자가 애플리케이션 스토어 서버, 애플리케이션 또는 해당 서비스와 관련된 웹 서버 등의 애플리케이션 제공 서버를 통해 마스터 단말기에 직접 설치한 애플리케이션(즉, 프로그램)에 의해 실행될 수도 있다. 이러한 의미에서, 전술한 본 발명의 일 실시예에 따른 RIA를 이용한 클라우드 기반 aPaaS 제공 방법은 단말기에 기본적으로 설치되거나 사용자에 의해 직접 설치된 애플리케이션(즉, 프로그램)으로 구현되고 단말기에 등의 컴퓨터로 읽을 수 있는 기록매체에 기록될 수 있다.

[0078] 전술한 본 발명의 설명은 예시를 위한 것이며, 본 발명이 속하는 기술분야의 통상의 지식을 가진 자는 본 발명의 기술적 사상이나 필수적인 특징을 변경하지 않고서 다른 구체적인 형태로 쉽게 변형이 가능하다는 것을 이해할 수 있을 것이다. 그러므로 이상에서 기술한 실시예들은 모든 면에서 예시적인 것이며 한정적이 아닌 것으로 이해해야만 한다. 예를 들어, 단일형으로 설명되어 있는 각 구성 요소는 분산되어 실시될 수도 있으며, 마찬가지로 분산된 것으로 설명되어 있는 구성 요소들도 결합된 형태로 실시될 수 있다.

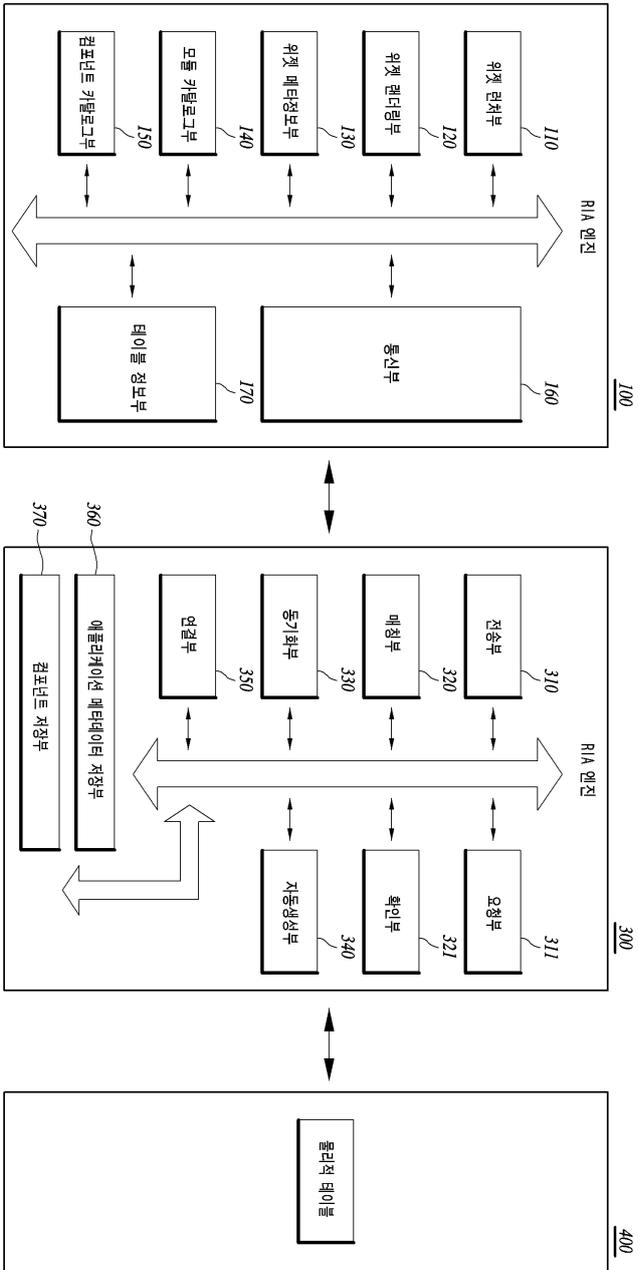
[0079] 본 발명의 범위는 상기 상세한 설명보다는 후술하는 특허청구범위에 의하여 나타내어지며, 특허청구범위의 의미 및 범위 그리고 그 균등 개념으로부터 도출되는 모든 변경 또는 변형된 형태가 본 발명의 범위에 포함되는 것으로 해석되어야 한다.

도면

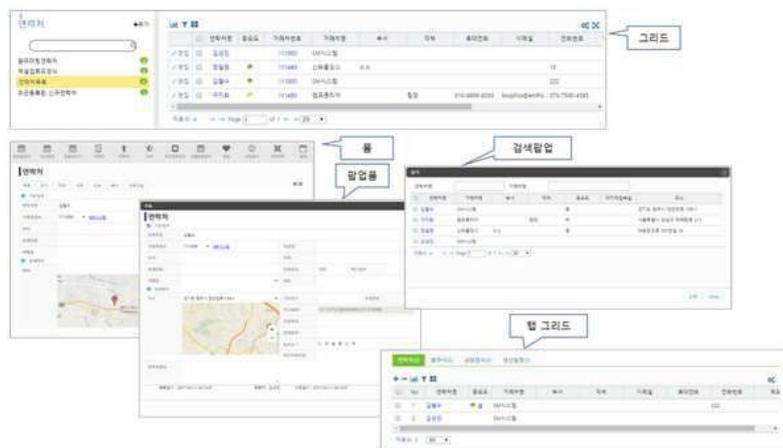
도면1



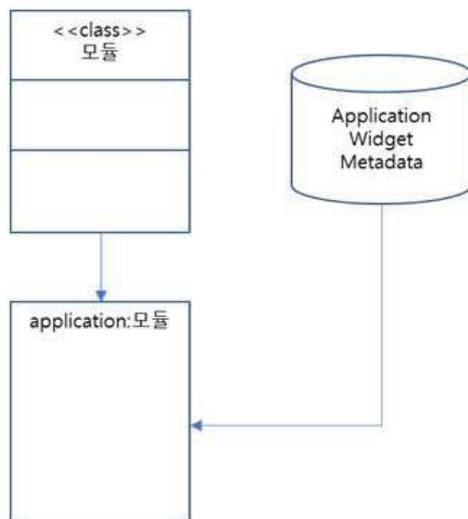
도면2



도면3



도면4



도면9

