

(12) 특허협력조약에 의하여 공개된 국제출원

(19) 세계지식재산권기구
국제사무국



(10) 국제공개번호
WO 2015/041488 A1

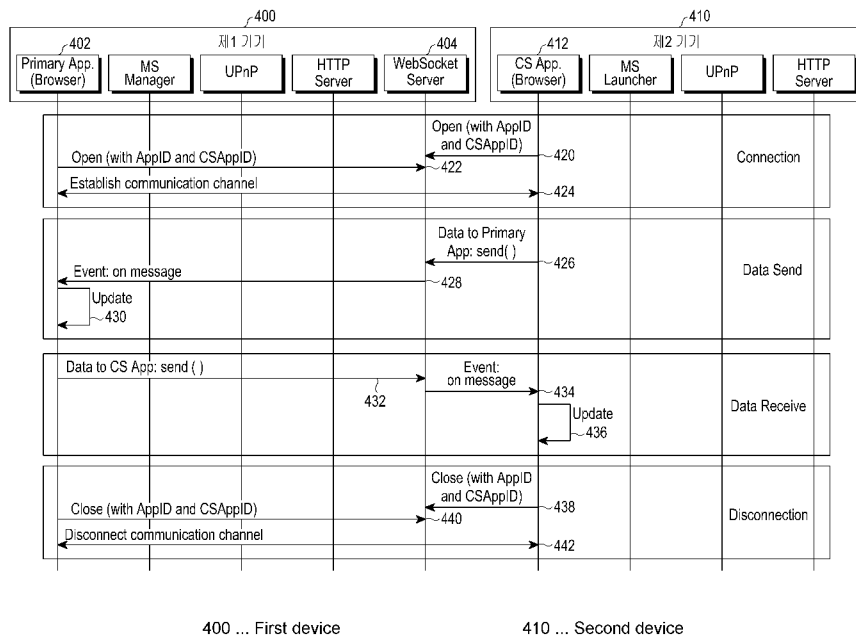
(43) 국제공개일
2015년 3월 26일 (26.03.2015)

- (51) 국제특허분류: *G06F 9/54* (2006.01)
- (21) 국제출원번호: PCT/KR2014/008770
- (22) 국제출원일: 2014년 9월 22일 (22.09.2014)
- (25) 출원언어: 한국어
- (26) 공개언어: 한국어
- (30) 우선권정보: 10-2013-0112794 2013년 9월 23일 (23.09.2013) KR
- (71) 출원인: 삼성전자 주식회사 (SAMSUNG ELECTRONICS CO., LTD.) [KR/KR]; 443-742 경기도 수원시 영통구 삼성로 129, Gyeonggi-do (KR).
- (72) 발명자: 류영선 (RYU, Young-Sun); 463-763 경기도 성남시 분당구 돌마로 486 번길 7 효자촌동아아파트 212동 902호, Gyeonggi-do (KR).
- (74) 대리인: 이견주 (LEE, Keon-Joo) 등; 110-524 서울시 중로구 대학로 9길 16 미화빌딩, Seoul (KR).
- (81) 지정국 (별도의 표시가 없는 한, 가능한 모든 종류의 국내 권리의 보호를 위하여): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.
- (84) 지정국 (별도의 표시가 없는 한, 가능한 모든 종류의 역내 권리의 보호를 위하여): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), 유라시아 (AM, AZ, BY, KG, KZ, RU, TJ, TM), 유럽 (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

[다음 쪽 계속]

(54) Title: METHOD AND APPARATUS FOR DEVICE-SPECIFIC COMMUNICATION BETWEEN APPLICATION PROGRAMMES

(54) 발명의 명칭 : 기기 별 응용 프로그램간 통신을 위한 장치 및 방법



(57) Abstract: The present invention relates to a method for providing communication between application programmes running via devices connected to the same network, and comprises: a step for connecting a first application programme, which is running in a first device via a web socket server in the first device, with a second application programme, which is running in a second device; and a step for transmitting and receiving data between the second application programme and the first application programme via the web socket server.

(57) 요약서: 본 발명은 동일망에 연결된 기들을 통해서 구동되는 응용프로그램들 간의 통신을 제공하는 방법에 있어서, 제 1 기기 내의 웹 소켓 서버를 통해서 상기 제 1 기기에서 구동되는 제 1 응용프로그램과, 제 2 기기에서 구동되는 제 2 응용프로그램을 연결하는 과정과, 상기 웹 소켓 서버를 통해서 상기 제 1 응용프로그램 및 제 2 응용프로그램 간의 데이터를 송수신하는 과정을 포함한다.

WO 2015/041488 A1

공개:

- 국제조사보고서와 함께 (조약 제 21 조(3))

명세서

발명의 명칭: 기기 별 응용 프로그램간 통신을 위한 장치 및 방법 기술분야

- [1] 본 발명은 동일망에 연결된 기기들을 통해서 구동되는 응용프로그램들 간의 통신 방법 및 장치에 관한 것이다.

배경기술

- [2] 무선 통신의 증대에 따라 실시간 양방향 데이터의 통신과 다수의 동시 접속자들을 수용 가능한 웹 소켓(websocket) 통신이 도입되었다. 이러한 웹소켓 통신은 서버와 클라이언트간의 통신을 기반으로 이루어진다.
- [3] 도 1은 일반적인 웹소켓 통신의 동작 방식의 일 예를 도시한 도면이다.
- [4] 도 1을 참조하면, 서버(102)는 웹 소켓 통신을 지원하는 서버이며, 클라이언트들 1 내지 4(104~110) 각각은 웹소켓 프로토콜(protocol)을 사용하여 원하는 상기 서버(102)와 연결하여 데이터를 주고 받는 경우를 가정하자. 여기서, 설명의 편의상 웹 소켓 통신을 지원하는 서버가 하나인 경우를 예시하였으나, 하나 이상의 웹 소켓 서버가 존재할 수 있다.
- [5] 이러한 웹소켓 통신은 기존의 통신 방법 중 하나인 HTTP(Hyper Text Transfer Protocol)가 가지고 있는 단점 즉, 단방향 통신을 극복하여 서버와 클라이언트 사이에 끊임없는 양방향 통신을 제공할 수 있다. 또한, 웹소켓 통신은 서버 측에서 복잡한 프로그래밍 없이 단순한 HTML(Hypertext Markup Language)5와의 연동이 쉽게 됨에 따라 기본적인 양방향 통신이 필요한 환경에서 다양하게 활용될 수 있을 것으로 기대된다.
- [6] 따라서 이러한 웹소켓 통신을 서로 다른 기기에서 구동되는 응용프로그램들간의 통신에 적용하는 방안이 연구되고 있다. 이에 따라, 서버와 클라이언트 사이의 양방향 통신을 위해 설계된 웹소켓 통신을 응용프로그램들간의 통신에 활용하기 위한 구체적 절차들이 요구된다.

발명의 상세한 설명

기술적 과제

- [7] 본 발명은 동일망에 연결된 기기들을 통해서 구동되는 응용프로그램간 통신을 제공할 수 있는 장치 및 방법을 제안한다.
- [8] 본 발명은, 제1기기에서 하나의 응용프로그램이 구동되고, 제2기기에서 상기 응용프로그램과 연동된 보조 응용프로그램이 구동되는 경우, 웹소켓 프로토콜을 사용하여 상기 응용 프로그램 및 보조프로그램간의 통신을 제공하기 위한 장치 및 방법을 제안한다.

과제 해결 수단

- [9] 본 발명의 실시 예에 따른 방법은; 동일망에 연결된 기기들을 통해서 구동되는 응용프로그램들 간의 통신을 제공하는 방법에 있어서, 제1기기 내의 웹 소켓

서버를 통해서 상기 제1기기에서 구동되는 제1 응용프로그램과 제2기기에서 구동되는 제2응용프로그램을 연결하는 과정과, 상기 웹 소켓 서버를 통해서 상기 제1응용프로그램 및 제2응용프로그램 간의 데이터를 송수신하는 과정을 포함한다.

- [10] 본 발명의 실시 예에 따른 다른 방법은; 동일망에 연결된 기기들을 통해서 구동되는 응용프로그램들 간의 통신을 제공하는 장치에 있어서, 상기 장치에서 구동되는 제1응용프로그램과 제2기기에서 구동되는 제2응용프로그램의 연결을 제공하고, 상기 연결을 통해서 상기 제1응용프로그램 및 제2응용프로그램 간의 데이터를 송수신하도록 제어하는 웹 소켓 서버를 포함한다.

발명의 효과

- [11] 본 발명은, 서로 다른 기기에서 구동되는 응용프로그램들간의 통신 수단을 제공함으로써, 예를 들어, 제1기기의 응용프로그램과 연동된 제2기기의 응용프로그램을 작성할 수 있다. 이를 통해, 제2기기에서 제1기기를 제어할 수 있는 리모트 콘트롤 응용프로그램을 작성한다거나, 제1기기를 이용하여는 제2기기의 입력 응용프로그램을 조작하는등의 사용자 경험 등을 제공할 수 있는 효과가 있다.

도면의 간단한 설명

- [12] 도 1은 일반적인 웹소켓의 동작 방식의 일 예를 도시한 도면,
 [13] 도 2는 본 발명의 실시 예에 따른 응용 프로그램간 통신을 설명하기 위한 기본적인 구성도의 일 예,
 [14] 도 3은 본 발명의 실시 예에 따라 제1기기에서 웹소켓 서버를 구동 및 정지하는 동작 흐름도의 일 예,
 [15] 도 4는 본 발명의 실시 예에 따라 제2기기의 응용프로그램이 제1기기의 응용프로그램과 통신하는 동작 흐름도의 일 예,
 [16] 도 5는 본 발명의 실시 예에 따른 제1기기의 동작 흐름도의 일 예,
 [17] 도 6은 본 발명의 실시 예에 따른 제2기기의 동작 흐름도의 일 예,
 [18] 도 7은 본 발명이 실제 적용된 실시 예를 도시한 도면.

발명의 실시를 위한 최선의 형태

- [19] 이하 첨부된 도면을 참조하여 본 발명의 바람직한 실시 예에 대한 동작 원리를 상세히 설명한다. 도면상에 표시된 동일한 구성요소들에 대해서는 비록 다른 도면상에 표시되더라도 가능한 한 동일한 참조번호로 나타내었으며, 다음에서 본 발명을 설명함에 있어 관련된 공지 기능 또는 구성에 대한 구체적인 설명이 본 발명의 요지를 불필요하게 흐릴 수 있다고 판단되는 경우에는 그 상세한 설명을 생략할 것이다. 그리고 후술되는 용어들은 본 발명에서의 기능을 고려하여 정의된 용어들로서 이는 사용자, 운용자의 의도 또는 관례 등에 따라 달라질 수 있다. 그러므로 그 정의는 본 명세서 전반에 걸친 내용을 토대로 내려져야 할 것이다.

- [20] 본 발명의 실시 예에서는, 동일망에 연결된 기기들을 통해서 구동되는 응용프로그램(application)간 통신을 제공할 수 있는 장치 및 방법을 제안한다. 구체적인 실시 예에 따라, 본 발명은 동일한 네트워크(network) 예를 들어, 홈 네트워크에 복수의 기기들이 연결되어 있는 환경에서, 제1기기에서 구동되는 응용프로그램과, 제2기기에서 구동되는 상기 응용프로그램과 연동된 보조 응용프로그램간의 통신을 웹소켓 프로토콜(websocket protocol)을 사용하여 제공하기 위한 장치 및 방법을 제안한다.
- [21] 도 2는 본 발명의 실시 예에 따른 기기 별 응용 프로그램간의 통신을 설명하기 위한 기본적인 구성도의 일 예이다. 설명의 편의상, 도 2의 기기 구성들은 본 발명의 실시 예에 따른 구성들만 포함하는 개략적인 구성도이다. 따라서, 사업자의 의도나 상황에 따라 해당 구성들은 하나의 유닛(unit)으로 합쳐지거나, 기능에 따라 서브(sub) 유닛들로 분할되어 구성될 수도 있다.
- [22] 도 2를 참조하면, 설명의 편의상, 제1기기(200)와 제2기기(210)는 동일한 네트워크 일 예로, 홈 네트워크에 연결된 기기들 중 하나임을 가정하자. 도면에 도시하지는 않았으나, 상기 홈 네트워크에 연결된 기기들은 상기 제1기기(200) 및 제2기기(210) 이외의 기기들이 존재할 수 있다.
- [23] 상기 제1기기(200)는 임의의 응용 프로그램이 구동되는 주기기(master device)로 정의되며, 예를 들어, 디지털 TV(DTV: digital television) 또는 셋톱박스(STB: set top box) 등의 공용 단말을 포함할 수 있다. 본 발명의 실시 예에 따른 제1기기(200)는 일 예로, 웹 브라우저(204)와 웹소켓 서버(206)를 포함할 수 있다. 여기서, 상기 웹 브라우저(204)는 웹 응용프로그램을 구동할 수 있는 구성요소에 해당하며, 상기 웹소켓 서버(206)는 본 발명의 실시 예에 따라 다른 기기의 응용프로그램과, 상기 제1기기(200)의 응용프로그램간의 통신을 위해서 추가된 구성요소이다. 구체적인 예로, 상기 웹 소켓 서버(206)는 상기 제1기기(200)에서 구동되는 응용프로그램과, 상기 제2기기(210)에서 구동되는 응용 프로그램들간의 연결 및 통신을 제공할 수 있다.
- [24] 그리고, 제2기기(210)는 도면에 도시하지는 않았으나, 상기 홈 네트워크에 연결된 기기들 중 하나와 연동하여 응용 프로그램을 실행하는 보조(slave) 기기로 정의되며, 예를 들어, 이동 통신 단말, 태블릿(tablet) 및 스마트폰(smart phone) 등과 같은 개인 단말이 될 수 있다. 본 발명의 실시 예에 따른 상기 제2기기(210)는 일 예로, 상기 제1기기(200)와 마찬가지로 웹 응용프로그램을 구동할 수 있는 웹 브라우저(212)를 포함한다. 이하, 명세서에서는 웹 응용프로그램에 대해 응용프로그램간 통신을 예를 들어 설명하고 있다. 그러나, 본 발명에서 설명하는 절차들과 API(Application Program Interface)를 지원하는 응용프로그램일 경우, 어떠한 응용프로그램도 적용될 수 있다. 예를 들어, 안드로이드 OS(Operation System)나 아이폰의 OS(iOS)에서 동작하는 네이티브(native) 응용프로그램 등을 포함할 수 있다.
- [25] 여기서, 상기 제1기기(200)에서 구동되는 응용프로그램과 상기

제2기기(210)에서 구동되는 응용프로그램(웹 응용프로그램)은 직접 통신 할 수 없다. 그러므로, 본 발명의 실시 예에서는 상기 제1기기(200)에 존재하는 웹 소켓 서버(206)를 프락시(Proxy)처럼 사용하여 기기별 응용 프로그램들간의 연결 및 통신이 수행될 수 있다. 즉, 상기 웹 소켓 서버(206)는 기기별 응용 프로그램들간의 연결 및 통신을 위한 릴레이(relay)로서 동작할 수 있다. 구체적으로, 본 발명의 실시 예에서는 상기 제1기기(200)의 웹소켓 서버(206)와 상기 웹브라우저(204) 사이에 WS(WebSocket) 프로토콜을 사용하여 통신하고, 마찬가지로, 상기 웹 소켓 서버(206)와 상기 제2기기(210)의 웹브라우저(212 또는 네이티브 응용프로그램, 이하 생략) 사이에 상기 WS 프로토콜을 사용하여 통신한다.

[26] 이하, 설명의 편의상, 본 발명의 실시 예들에 따른 기기별 응용프로그램들 간의 통신을 지원하는 장치 및 동작들은 도 2의 구성 예를 기반으로 설명하기로 한다.

[27] 도 3은 본 발명의 실시 예에 따라 제1기기에서 웹소켓 서버를 구동 및 정지하는 동작 흐름도의 일 예이다. 설명의 편의상, 상기 제1기기는 홈 네트워크에 연결된 DTV이고, 제2 기기는 상기 홈 네트워크에 연결된 모바일(Mobile) 기기인 경우를 가정하여 설명하기로 한다. 그러나, 본 발명의 실시 예가 적용되는 제 1기기와 제 2기기는 해당 기술을 지원하는 어떠한 디바이스(device)도 될 수 있다.

[28] 먼저, 제1기기(310)에서 웹소켓 서버(304)의 구동(WS starting) 동작을 설명하기로 한다. 도 3을 참조하면, 322단계에 제1기기(310)의 웹 응용프로그램(이하, '프라이머리 APP'이라 칭함, 302)은 API를 통해 상기 제1기기(310)의 웹소켓 서버(304)에 전달하여, 상기 웹 소켓 서버(304)를 구동시킨다. 여기서, API는 하기 [표1]에 나타낸 바와 같이, start WS를 사용할 수 있다.

[29] 표 1

[Table 1]

API	설명	인자	리턴값
startWS(unit AppID)	기기의 웹소켓 서버를 구동한다.	uint AppID: 웹소켓 서버와 연동할 응용프로그램의 식별자	True: 구동 성공시 False: 구동 실패시
shutdownWS(uint AppID)	기기의 웹소켓 서버를 정지한다.	uint AppID: 웹소켓 서버와 연동할 응용프로그램의 식별자	True: 정지 성공시 False: 정지 실패시

[30] 이때, 상기 start WS는 상기 프라이머리 App(302)와의 연동을 위해 상기 응용프로그램의 ID(Identifier)인 AppID를 함께 제공한다. 여기서, 응용프로그램의 ID는, 실제 구현 시 해당 응용프로그램을 식별할 수 있는 ID나 URL(Uniform resource Locator) 등이 될 수 있다. 그러면, 웹소켓 서버(304)는 상기

AppID에 대응하는 상기 프라이머리 App(302)과 통신하기 위해서 자신을 구동시키고, 상기 구동의 완료를 ACK 신호로서 상기 프라이머리 App(302)에게 전송한다.

- [31] 다음으로, 상기 제1기기(300)에서 웹소켓 서버(304)를 정지(WS shutdown)하는 동작을 설명하기로 한다.
- [32] 326단계에서 상기 프라이머리 App(302)은 API를 상기 제1기기(310)의 웹소켓 서버(304)에게 전달하여, 구동중인 웹소켓 서버(304)를 정지시킨다. 이때, 상기 프라이머리 App(302)에서 호출하는 API는 상기 [표 1]과 같이 shutdown WS를 사용할 수 있다. 마찬가지로, 상기 shutdown WS 역시 상기 AppID를 함께 제공한다. 그러면, 웹소켓 서버(310)는 상기 AppID에 대응하는 상기 프라이머리 App와의 통신을 중단하고, 상기 중단의 완료를 ACK 신호로서 상기 제1기기(300)의 AppID(302)에게 전송한다.
- [33] 도 3의 실시 예에서는 제1기기 내의 특정 응용프로그램을 위한 웹소켓 서버를 제1기기가 직접 구동 및 중단시키는 방법에 대해 설명하였다. 다른 실시 예에서는 공용으로 사용할 수 있는 웹소켓 서버를 구동시킨 후, 다음의 절차에서 설명할 API들을 이용하여 제2기기의 응용프로그램이 상기 제1기기의 웹소켓 서버와 연동될 수 있다.
- [34] 도 4는 본 발명의 실시 예에 따라 제2기기의 응용프로그램이 제1기기의 응용프로그램과 통신하는 동작 흐름도의 일 예이다. 참고로, 하기 <표 2>는 기기별 응용 프로그램 간의 통신을 위한 API들의 예를 나타낸 표의 일 예이다.
- [35] 표 2

[Table 2]

API	설명	인자	리턴값
startWS(uint AppID)	웹소켓 서버를 구동시킨다.	uint AppID: 웹소켓 서버를 구동시킬 응용프로그램	True: 구동 성공 False: 구동 실패
ShutdownWS(uint AppID)	웹소켓 서버를 정지시킨다.	uint AppID: 웹소켓 서버를 정지시킬 응용프로그램	True: 정지 성공 False: 정지 실패
check_channel(AppID, CSAppID)	제1기기의 응용프로그램(AppID)와 제2기기의 응용프로그램(CSAppID)간 통신채널이 생성되었는지 확인한다.	AppID: 제1기기의 응용프로그램 ID CSAppID: 제2기기의 응용프로그램 ID	True: 특정채널이 생성되어 있음 False: 특정채널이 생성되어 있지 않음
getHbbTVAppID()	제1기기의 응용프로그램 ID를 구한다.	없음	uint HbbTVID: 제1기기의 응용프로그램 ID
getCSAppID()	제2기기의 응용프로그램 ID를 구한다.	없음	uint CSAppID: 제2기기의 응용프로그램 ID

- [36] 도 4를 참조하면, 420단계에서 제2기기(410)의 응용프로그램(이하, 'CS(Companion Screen) App'라 칭함)(412)은 웹소켓 객체를 생성하고, 제1기기(400)의 웹소켓 서버(404)에게 open() API를 호출하여 통신 채널의 생성을 요청한다. 이때, open() API는 프라이머리 App(402)의 ID(AppID)와, CS App(412)의 ID(CSAppID)를 전달할 수 있다. 그리고, 422단계에서, 상기 프라이머리 App(402)는 상기 AppID와 상기 CSAppID를 이용하여 웹소켓 객체를 생성하고, 웹 소켓 서버(404)에게 open() API를 호출하여 통신 채널의 생성을 요청한다. 마찬가지로, 상기 프라이머리 App(402)의 open() API역시 상기 AppID와 상기 CSAppID를 전달한다.
- [37] 그러면, 424단계에서 상기 제1기기(400)의 웹소켓 서버(404)는 AppID에 상응하는 상기 프라이머리 App(402)와 상기 CSAppID에 상응하는 상기 CS App(412) 간에 통신 채널을 생성한다. 상기한 바와 같은 과정으로 통신 채널이

생성되면, 상기 CS App(412)은 상기 통신 채널을 통해 프라이머리 App(402)과 통신할 수 있다. 마찬가지로, 상기 프라이머리 App(402) 역시 상기 통신 채널을 통해 상기 CSApp(412)과 통신할 수 있다. 구체적으로, 420단계 내지 424단계를 통해서 프라이머리 App(402)과 CSApp(412) 각각이 웹소켓 서버에 연결하는 방법은 일 예로, 하기 <표3>을 이용할 수 있다.

[38] 표 3

[Table 3]

웹소켓 서버의 주소가 ws://192.168.1.52일 경우 웹소켓 연결 방법 (제1기기와 제2기기)
ws://192.168.1.52/AppID/CSAppID

[39] 앞서 설명한 바와 같이, 상기 웹 소켓 서버(404)가 상기 프라이머리 App(402)의 open() API로부터 획득한 정보와, 상기 CS App(412)으로부터 획득한 정보가 동일함을 확인한다. 도 4의 실시 예에서는 상기 프라이머리 App(402) 및 CS App(412) 각각이 AppID 및 CSAppID를 상기 통신 채널을 위한 동일 정보로서 전송하는 경우를 일 예로서 설명하였다. 그러면, 상기 웹 소켓 서버(404)는 상기 프라이머리 App(402)와의 연결과, 자신과 상기 CS App(412)과의 연결을 중계(relay)하여 상기 프라이머리 App(402) 및 CS App(412)간의 통신 채널을 설정한다. 이에 따라, 상기 프라이머리 App(402)와 상기 웹 소켓 서버(404)간의 연결, 및 상기 CS App(412)과 상기 웹 소켓 서버(404) 간의 연결 각각은 동일한 채널 ID가 설정된다. 상기 채널 ID는 예를 들어, 상기 <표 3>에 도시한 바와 같이 웹 소켓 서버의 주소와, 상기 프라이머리 App(402)의 AppID 및 상기 CS App(412)의 CSAppID를 이용하여 고유한 값으로 생성될 수 있다.

[40] 이후, 상기 CS App(412)에서 상기 프라이머리 App(402)으로 전송할 데이터가 발생한 경우, 426단계에서 상기 CS App(412)은 send() API를 호출하여 상기 발생한 데이터를 상기 웹소켓 서버(404)에게 전송한다. 그러면, 428단계에서 상기 웹소켓 서버(404)는 Event(on message)를 호출하여 상기 프라이머리 App(402)에게 수신된 데이터가 있음을 알린다. 그러면, 430단계에서 상기 프라이머리 App(402)은 상기 수신된 데이터를 응용프로그램에 반영한다. 일 예로, 상기 프라이머리 App(402)이 웹 응용프로그램인 경우, 430단계의 과정은 DOM(Document Object Model) Update를 통해 이루어질 수 있다.

[41] 한편, 상기 프라이머리 App(402)에서 변경된 사항이 있는 경우, 432단계에서 send() API를 호출하여, 변경된 사항에 대한 정보를 상기 웹 소켓 서버(404)로 전달한다. 그러면, 434단계에서 상기 1기기(400)의 웹소켓 서버(404)는, 앞서 설명한 과정을 통해서 생성된 상기 프라이머리 App(402)와 상기 CS App(412)간의 상기 통신채널을 통해 상기 정보를 상기 CS App(412)에게 전송한다. 이때, 상기 정보는 Event(on message) API를 통해서 전달된다. 436단계에서 상기 CS App(412)는 상기 정보를 수신하여, 430단계와 유사하게

자신의 응용프로그램에 반영한다.

- [42] 이후, 모든 통신이 완료된 경우, 438단계에서 상기 CS App(412)은 close() API를 호출하여 상기 프라이머리 App(402)과의 통신채널의 종료 요청을 상기 웹소켓 서버(404)에게 전송한다. 마찬가지로, 440단계에서 상기 프라이머리 App(410) 역시 close() API를 호출하여 제1기기의 응용프로그램과의 통신채널의 종료 요청을 상기 웹소켓 서버(404)에게 전송한다. 그러면 상기 웹소켓 서버(404)는 상기 프라이머리 App(402)와 상기 CS App(412) 간의 연결을 종료한다. 이에 따라 442단계에서 상기 프라이머리 App(402)와 상기 CS App(412)간의 연결이 종료된다.
- [43] 도 5는 본 발명의 실시 예에 따른 제1기기의 동작 흐름도의 일 예이다.
- [44] 도 5를 참조하면, 500단계에서 제1기기는 HTML 페이지를 수신한다. 상기 HTML 페이지는 상기 제1기기의 프라이머리 App을 지시하는 AppID를 포함한다. 그러면, 502단계에서 상기 제1기기는 상기 HTML 페이지의 API 호출에 따라 상기 AppID를 이용하여 상기 웹소켓 서버를 구동시킨다.
- [45] 그리고, 504단계에서 상기 프라이머리 App은 상기 제1기기와 연결된 네트워크와 연결된 기기들 중 가용한 제2기기를 검색한다. 그리고, 506단계에서 상기 프라이머리 App은, 상기 검색된 제2기기로 상기 제2기기의 CS App의 구동 정보를 송신한 후, 상기 제1기기는, 508단계에서 사용자 입력을 요구하는 HTML 페이지를 상기 제1기기의 디스플레이 화면에 표시한다. 510단계에서 상기 제1기기는 사용자의 입력을 수신하면, 512단계에서 상기 사용자의 입력을 통해서 상기 CS App으로 전송할 데이터의 발생 유무를 확인한다. 상기 확인 결과, 데이터가 발생하지 않은 경우, 516단계로 진행한다.
- [46] 상기 확인 결과, 상기 데이터가 발생한 경우, 514단계에서 상기 제1기기는 502단계에서 구동시킨 웹소켓 서버로 상기 데이터를 송신한다. 그리고, 516단계에서 상기 제1기기는 상기 웹소켓 서버를 통해서 수신할 데이터가 있는지 여부를 확인한다. 상기 확인 결과, 수신할 데이터가 존재하지 않는 경우, 508단계로 진행하여, 다른 사용자 입력을 대기할 수 있다.
- [47] 상기 확인 결과, 수신할 데이터가 발생한 경우, 518단계에서 상기 제1기기는 상기 웹소켓 서버를 통해서 데이터를 수신한다. 그리고, 520단계에서 상기 제1기기는 상기 수신된 데이터로부터 획득한 응용프로그램의 ID가 상기 제1기기의 AppID인지 판단한다. 상기 판단 결과, 상기 데이터로부터 획득한 응용프로그램 ID가 상기 AppID인 경우, 522단계에서 상기 제1기기는 DOM을 업데이트하여 상기 수신된 데이터를 반영한 HTML을 508단계에서 자신의 디스플레이 화면을 통해서 표시한다.
- [48] 도 6은 본 발명의 실시 예에 따른 제2기기의 동작 흐름도의 일 예이다.
- [49] 도 6을 참조하면, 일 예로, 상기 제2기기는 600단계에서 도5의 506단계에 의해서 상기 제1기기가 송신한 정보로서 HTML 페이지를 수신한 경우를 가정한다. 여기서는 설명의 편의상, 상기 제2기기의 CSApp을 웹

응용프로그램으로 가정하여 HTML 페이지를 수신하는 것으로 설명하였다. 그러나, 다른 예로, 상기 제2기기가 600단계에서 상기 제1기기를 통해서 수신된 정보가 제2기기의 네이티브(native) 응용프로그램을 가리키는 경우, 해당 네이티브 응용프로그램을 구동하게 된다.

- [50] 그리고, 602단계에서 상기 제2기기의 CSApp은 상기 HTML 메시지에서 획득한 제1기기의 AppID와 제2기기의 CSAppID를 이용하여 상기 제1기기의 웹소켓 서버와의 연결을 호출한다.
- [51] 604단계에서 상기 제2기기는 상기 제1기기의 프라이머리 App과 연동할 사용자 입력을 요구하는 HTML 페이지를 상기 제2기기의 디스플레이 화면에 표시한다. 606단계에서 상기 제2기기는 사용자의 입력의 수신을 감지하면, 608단계에서 상기 사용자의 입력에 따라 제1기기의 프라이머리 App으로 전송할 데이터가 존재하는 지 여부를 확인한다. 상기 확인 결과, 전송할 데이터가 존재하지 않는 경우, 상기 제2기기는 612단계로 진행한다.
- [52] 상기 확인 결과, 상기 사용자의 입력에 따라 상기 제1기기로 전송할 데이터가 존재하는 경우, 610단계에서 상기 제2기기는 602단계에서 연결한 상기 제1기기의 웹소켓 서버를 통해서 상기 프라이머리 App으로 전송할 데이터를 송신한다.
- [53] 한편, 612단계에서 상기 제2기기는 상기 웹 소켓 서버를 통해서 수신할 데이터가 있는 경우, 614단계에서 해당 데이터를 수신한다. 그리고, 616단계에서 상기 제2기기는 상기 수신된 데이터를 반영하여 DOM을 업데이트한다. 이후, 604단계로 복귀하여 업데이트 결과를 기반으로 HTML을 재구성하여 자신의 디스플레이 화면에 표시한다.
- [54] 한편, 상기 612단계에서의 확인 결과, 상기 웹 소켓 서버를 통해서 수신할 데이터가 존재하지 않으면, 상기 제2기기는 상기 604단계로 복귀하여 다른 사용자 입력을 대기할 수 있다.
- [55] 도 7은 본 발명이 실제 적용된 실시 예를 도시한 도면이다.
- [56] 도 7을 참조하면, 일 예로, 홈 네트워크에 제1기기(700)와 제2기기(705)가 연결된 상황을 가정하자. 그리고, 사용자가 제1기기(700)로 방송 콘텐츠를 수신하면서, 상기 제2기기(705)를 통해 상기 제1기기를 제어하는 경우를 가정하자. 구체적인 예로, 상기 사용자는 사용자 입력을 위한 조작이 편리한 상기 제2기기(705)를 이용하여 상기 제1기기(700)에게 리모트 콘트롤, 문자 입력 등을 위한 사용자 입력을 수행할 수 있다. 또한, 상기 제1기기(700)의 방송 콘텐츠에 따라 디스플레이되는 화면을 선택하거나 방송 콘텐츠 별로 제공하는 해당 어플리케이션을 실행시키거나, 투표 등을 상기 제2기기(705)를 이용하여 실시하는 등의 다양한 실시 예가 가능하게 된다. 이때, 상기 제1기기(700)는 앞서 설명한 바와 같이 웹 소켓 서버를 구비하고, 상기 웹 소켓 서버를 통해서 상기 제1기기(700)에서 구동하는 어플리케이션과 제2기기(705)에서 상기 제1기기(700)의 조작을 위한 어플리케이션 간의 연결 및 통신을 제공할 수 있다.

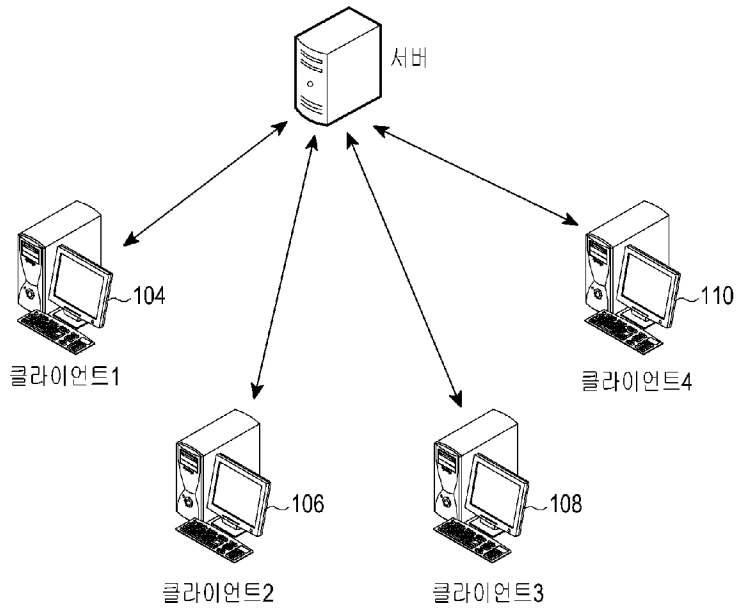
- [57] 한편 본 발명의 상세한 설명에서는 구체적인 실시 예에 관해 설명하였으나, 본 발명의 범위에서 벗어나지 않는 한도 내에서 여러 가지 변형이 가능함은 물론이다. 그러므로 본 발명의 범위는 설명된 실시 예에 국한되어 정해져서는 안되며 후술하는 특허 청구의 범위뿐만 아니라 이 특허 청구의 범위와 균등한 것들에 의해 정해져야 한다.

청구범위

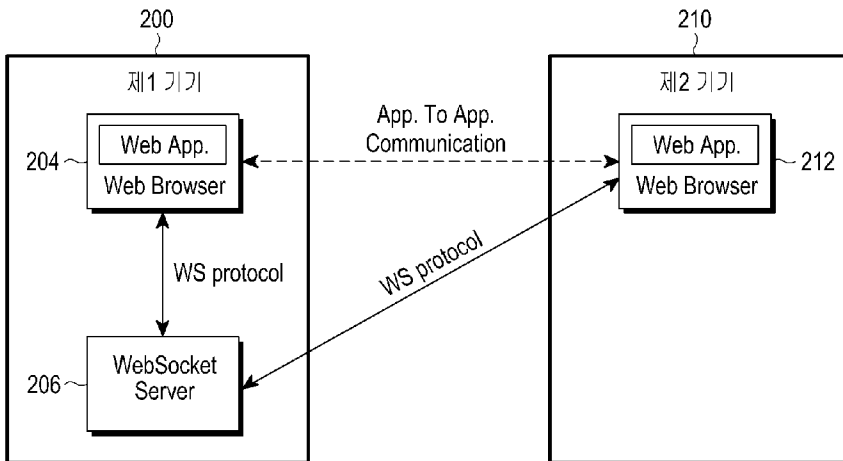
- [청구항 1] 동일망에 연결된 기기들을 통해서 구동되는 응용프로그램들 간의 통신을 제공하는 방법에 있어서,
제1기기 내의 웹 소켓 서버를 통해서 상기 제1기기에서 구동되는 제1응용프로그램과 제2기기에서 구동되는 제2응용프로그램을 연결하는 과정과,
상기 웹 소켓 서버를 통해서 상기 제1응용프로그램 및 제2응용프로그램 간의 데이터를 송수신하는 과정을 포함하는 방법.
- [청구항 2] 제1항에 있어서,
상기 연결하는 과정은,
상기 제1응용프로그램과 상기 제2응용프로그램 각각으로부터 연결 요청을 수신하는 과정을 포함하는 방법.
- [청구항 3] 제1항에 있어서,
상기 연결하는 과정은,
상기 웹 소켓 서버의 프로토콜을 사용하여 상기 웹 소켓 서버와 상기 제1응용프로그램을 제1연결하는 과정과,
상기 프로토콜을 사용하여 상기 웹소켓 서버와 상기 제2응용프로그램을 제2연결하는 과정을 포함하는 방법.
- [청구항 4] 제1항에 있어서,
상기 웹 소켓 서버를 통해서 상기 제1연결과 상기 제2연결을 중계하여 상기 제1응용프로그램 및 제2응용프로그램 간의 통신채널을 구성하는 과정을 포함하는 방법.
- [청구항 5] 제4항에 있어서,
상기 제1연결 및 제2연결 각각은 동일한 채널 식별자가 설정됨을 특징으로 하는 방법.
- [청구항 6] 제1항에 있어서,
상기 제1응용프로그램과 상기 제2응용프로그램 각각으로부터 연결 해제 요청을 수신하는 과정과,
상기 제1응용프로그램 및 제2응용프로그램간의 연결을 해제하는 과정을 더 포함하는 방법.
- [청구항 7] 동일망에 연결된 기기들을 통해서 구동되는 응용프로그램들 간의 통신을 제공하는 장치에 있어서,
상기 장치에서 구동되는 제1응용프로그램과 제2기기에서 구동되는 제2응용프로그램의 연결을 제공하고, 상기 연결을 통해서 상기 제1응용프로그램 및 제2응용프로그램 간의 데이터를 송수신하도록 제어하는 웹 소켓 서버를 포함함을 특징으로 하는

- 장치.
- [청구항 8] 제7항에 있어서,
상기 웹 소켓 서버는,
상기 제1응용프로그램과 상기 제2응용프로그램 각각으로부터
연결 요청을 수신하면, 상기 연결을 제공함을 특징으로 하는 장치.
- [청구항 9] 제7항에 있어서,
상기 웹 소켓 서버는,
상기 웹 소켓 서버의 프로토콜을 사용하여 상기 웹 소켓 서버와
상기 제1응용프로그램을 제1연결하고, 상기 프로토콜을 사용하여
상기 웹소켓 서버와 상기 제2응용프로그램을 제2연결함을
특징으로 하는 장치.
- [청구항 10] 제7항에 있어서,
상기 웹 소켓 서버는,
상기 제1연결과 상기 제2연결을 중계하여 상기 제1응용프로그램
및 제2응용프로그램 간의 통신채널을 구성함을 특징으로 장치.
- [청구항 11] 제10항에 있어서,
상기 제1연결 및 제2연결 각각은 동일한 채널 식별자가 설정됨을
특징으로 하는 장치.
- [청구항 12] 제7항에 있어서,
상기 웹 소켓 서버는,
상기 제1응용프로그램과 상기 제2응용프로그램 각각으로부터
연결 해제 요청을 수신하면, 상기 제1응용프로그램 및
제2응용프로그램간의 연결을 해제함을 특징으로 하는 장치.

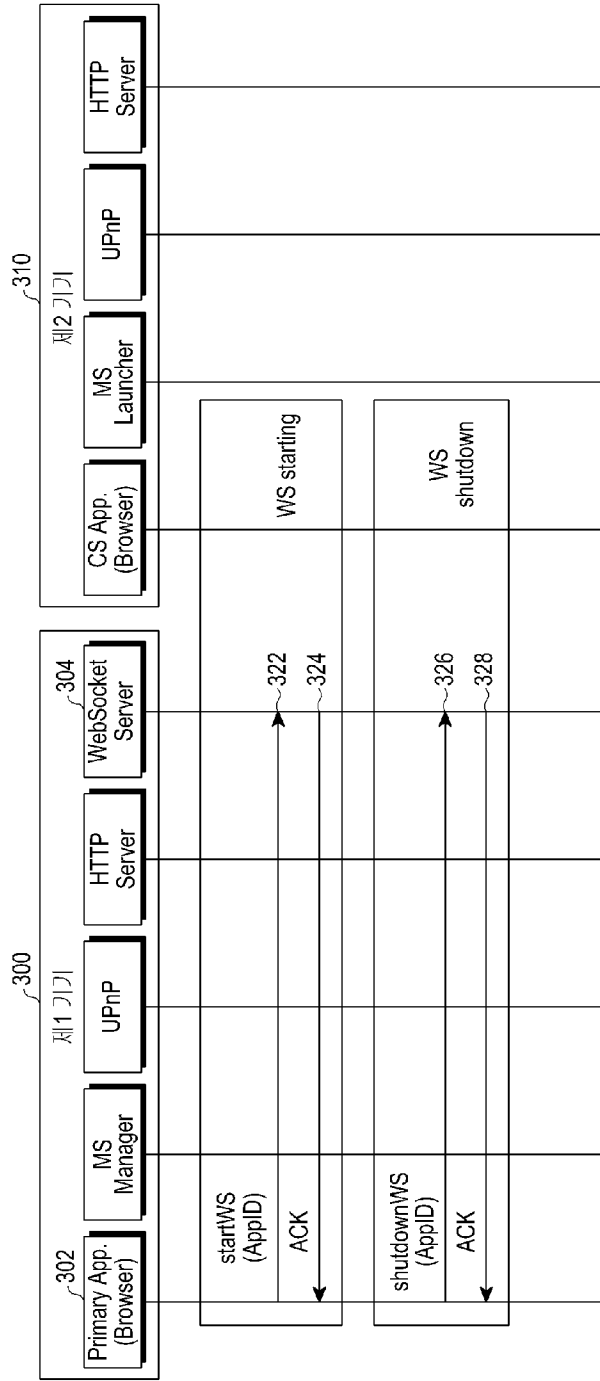
[Fig. 1]



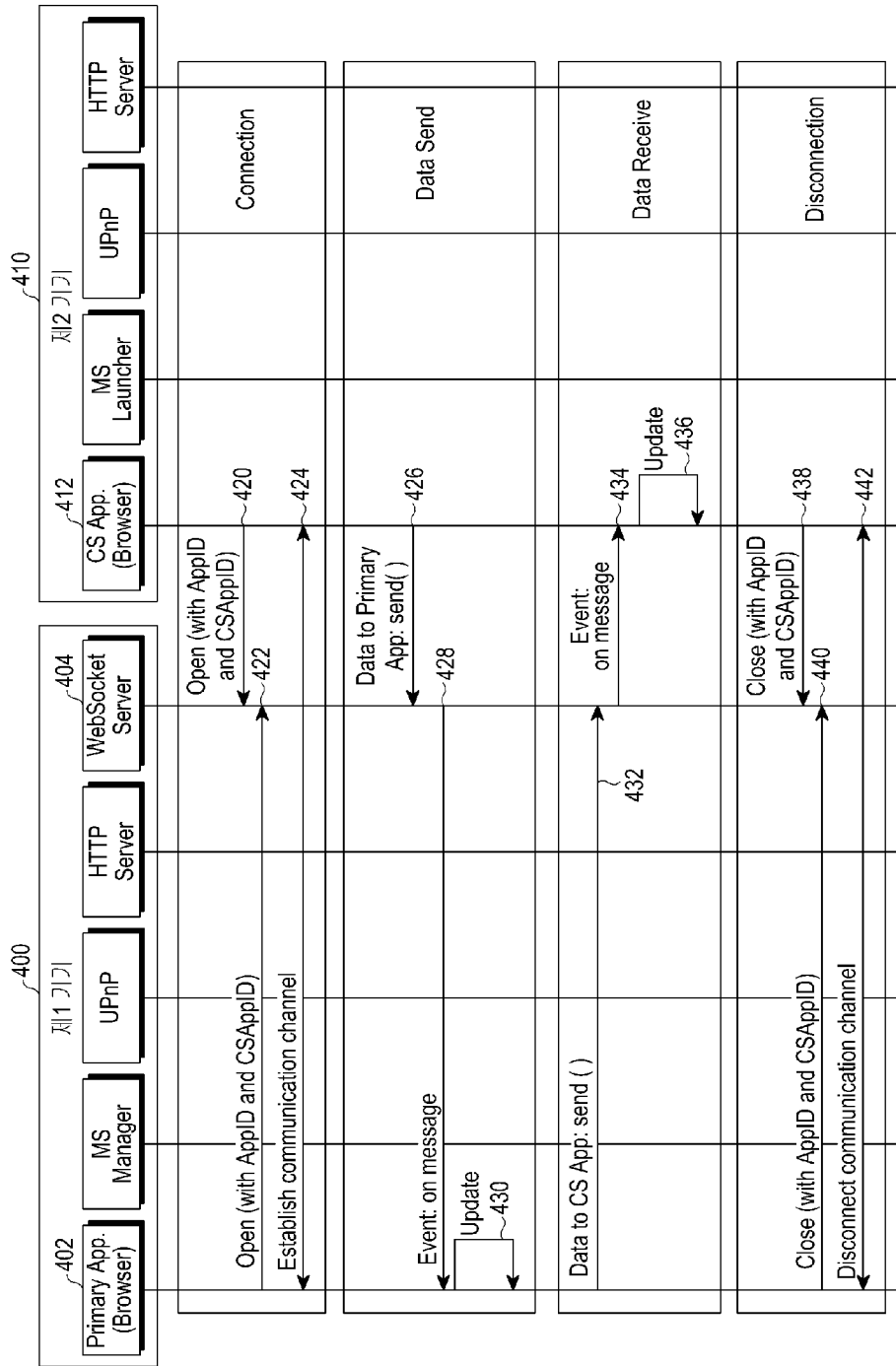
[Fig. 2]



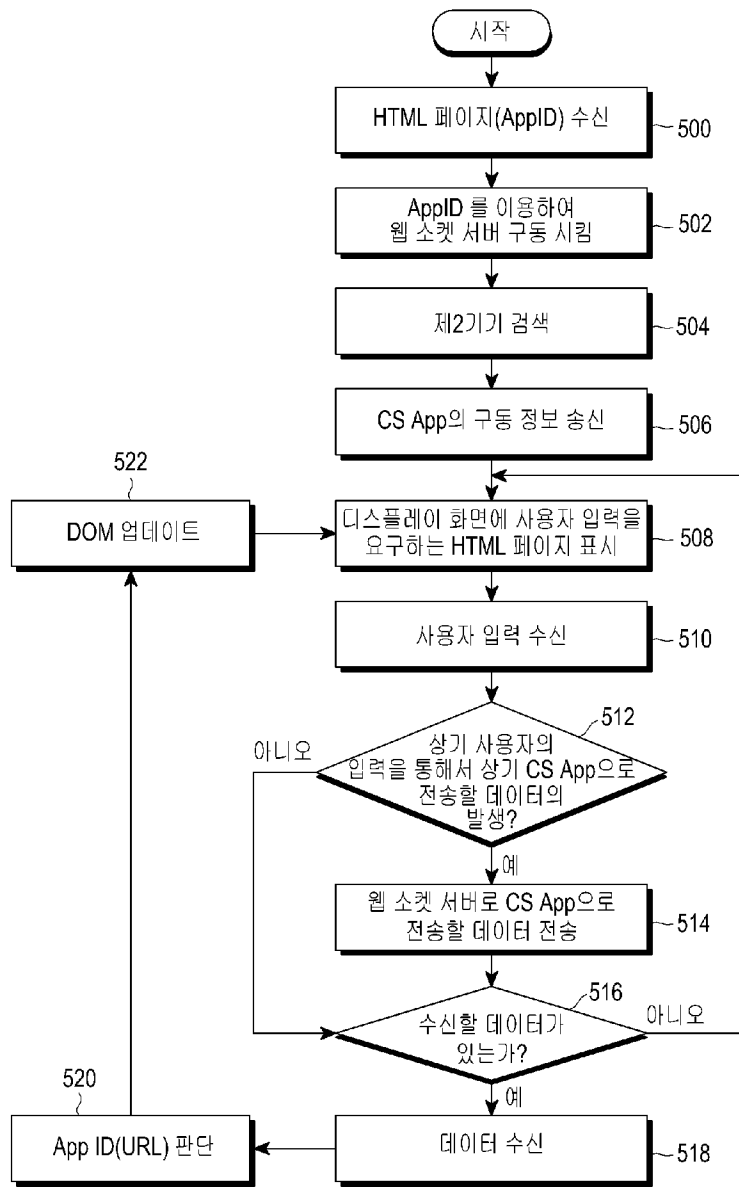
[Fig. 3]



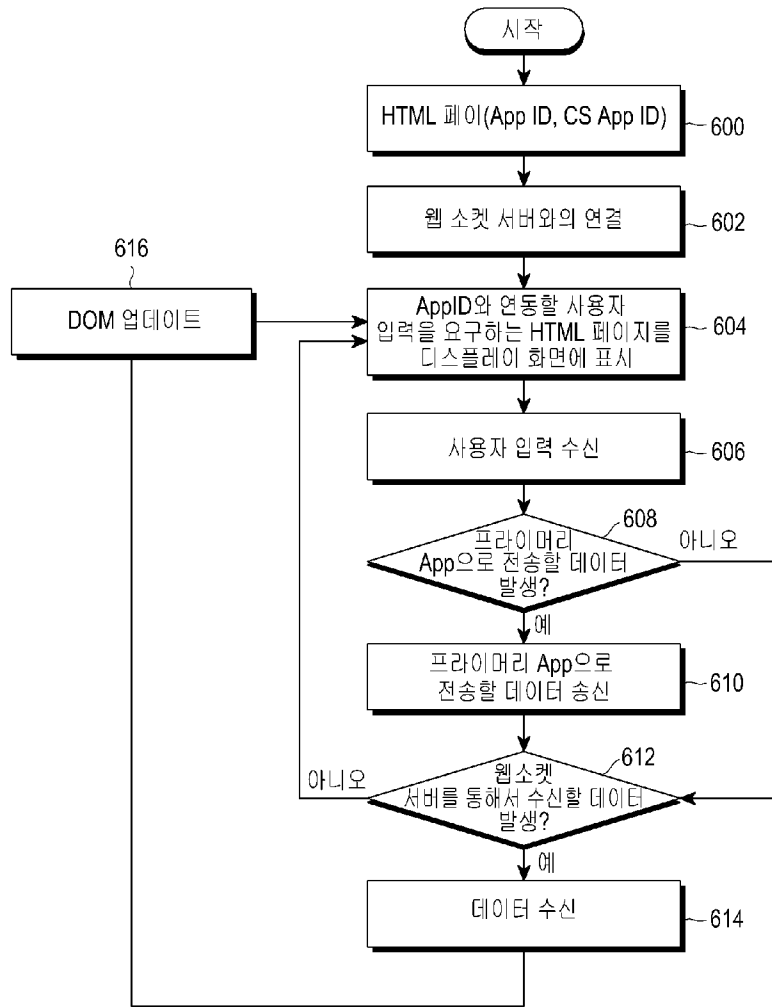
[Fig. 4]



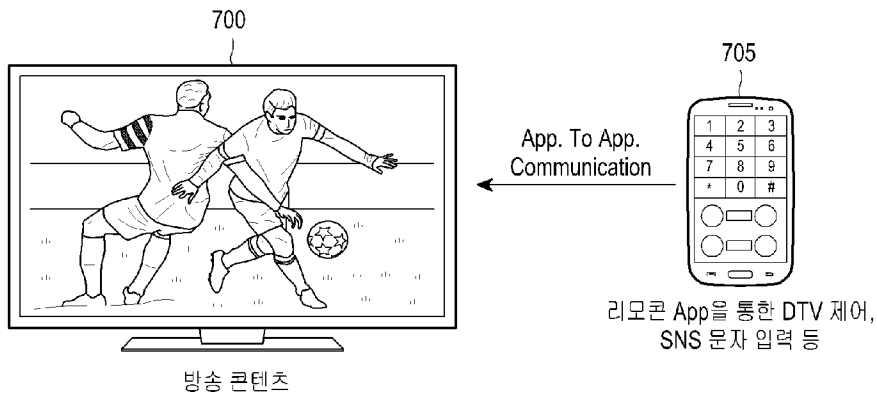
[Fig. 5]



[Fig. 6]



[Fig. 7]



INTERNATIONAL SEARCH REPORT

International application No.

PCT/KR2014/008770

A. CLASSIFICATION OF SUBJECT MATTER

G06F 9/54(2006.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F 9/54; G06F 15/16; H04W 76/02; H04W 88/18; H04Q 9/00; H04Q 9/04

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Korean Utility models and applications for Utility models: IPC as above

Japanese Utility models and applications for Utility models: IPC as above

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

eKOMPASS (KIPO internal) & Keywords: websocket, protocol, application program, communication

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	FUKAI et al. "Web Browser based GUI for TV." In: 2012 IEEE Global Conference on Consumer Electronics, 02-05 October 2012, pp. 579-580. See abstract, figure 1.	1,2,3,4-6,7,8,9 ,10-12
A	KR 10-2012-0080892 A (LOGICPLANT) 18 July 2012 See abstract, claims 1 and 6, figures 1 and 2.	1-12
A	KR 10-2007-0060957 A (ELECTRONICS AND TELECOMMUNICATIONS RESEARCH INSTITUTE) 13 June 2007 See abstract, paragraphs [0029]-[0041], figure 4.	1-12
A	KR 10-2006-0031913 A (SAMSUNG ELECTRONICS CO., LTD.) 14 April 2006 See abstract, claims 1 and 4, figure 2.	1-12



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"I" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family


Date of the actual completion of the international search

29 DECEMBER 2014 (29.12.2014)

Date of mailing of the international search report

30 DECEMBER 2014 (30.12.2014)

Name and mailing address of the ISA/KR


 Korean Intellectual Property Office
 Government Complex-Daejeon, 189 Seonsa-ro, Daejeon 302-701,
 Republic of Korea

Facsimile No. 82-42-472-7140

Authorized officer

Telephone No.

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.

PCT/KR2014/008770

Patent document cited in search report	Publication date	Patent family member	Publication date
KR 10-2012-0080892 A	18/07/2012	NONE	
KR 10-2007-0060957 A	13/06/2007	NONE	
KR 10-2006-0031913 A	14/04/2006	CN 1764187 A	26/04/2006
		CN 1764187 C0	26/04/2006
		US 2006-0080402 A1	13/04/2006

A. 발명이 속하는 기술분류(국제특허분류(IPC))
G06F 9/54(2006.01)i

B. 조사된 분야

조사된 최소문헌(국제특허분류를 기재)
G06F 9/54; G06F 15/16; H04W 76/02; H04W 88/18; H04Q 9/00; H04Q 9/04

조사된 기술분야에 속하는 최소문헌 이외의 문헌
한국등록실용신안공보 및 한국공개실용신안공보: 조사된 최소문헌란에 기재된 IPC
일본등록실용신안공보 및 일본공개실용신안공보: 조사된 최소문헌란에 기재된 IPC

국제조사에 이용된 전산 데이터베이스(데이터베이스의 명칭 및 검색어(해당하는 경우))
eKOMPASS(특허청 내부 검색시스템) & 키워드: 웹 소켓, 프로토콜, 응용 프로그램, 통신

C. 관련 문헌

카테고리*	인용문헌명 및 관련 구절(해당하는 경우)의 기재	관련 청구항
X	FUKAI 외 4명. 'Web Browser based GUI for TV.' In: 2012 IEEE Global Conference on Consumer Electronics, 2012.10.2-5., pp. 579-580. 요약, 도면 1 참조.	1,2,3,4-6,7,8,9,10-12
A	KR 10-2012-0080892 A (주식회사 로직플랜트) 2012.07.18 요약, 청구항 1 및 6, 도면 1 및 2 참조.	1-12
A	KR 10-2007-0060957 A (한국전자통신연구원) 2007.06.13 요약, 문단번호 [0029]-[0041], 도면 4 참조.	1-12
A	KR 10-2006-0031913 A (삼성전자주식회사) 2006.04.14 요약, 청구항 1 및 4, 도면 2 참조.	1-12

추가 문헌이 C(계속)에 기재되어 있습니다. 대응특허에 관한 별지를 참조하십시오.

* 인용된 문헌의 특별 카테고리:
 "A" 특별히 관련이 없는 것으로 보이는 일반적인 기술수준을 정의한 문헌 "T" 국제출원일 또는 우선일 후에 공개된 문헌으로, 출원과 상충하지 않으며 발명의 기초가 되는 원리나 이론을 이해하기 위해 인용된 문헌
 "E" 국제출원일보다 빠른 출원일 또는 우선일을 가지나 국제출원일 이후에 공개된 선출원 또는 특허문헌 "X" 특별한 관련이 있는 문헌. 해당 문헌 하나만으로 청구된 발명의 신규성 또는 진보성이 없는 것으로 본다.
 "L" 우선권 주장에 의문을 제기하는 문헌 또는 다른 인용문헌의 공개일 또는 다른 특별한 이유(이유를 명시)를 밝히기 위하여 인용된 문헌 "Y" 특별한 관련이 있는 문헌. 해당 문헌이 하나 이상의 다른 문헌과 조합하는 경우로 그 조합이 당업자에게 자명한 경우 청구된 발명은 진보성이 없는 것으로 본다.
 "O" 구두 개시, 사용, 전시 또는 기타 수단을 언급하고 있는 문헌 "&" 동일한 대응특허문헌에 속하는 문헌
 "P" 우선일 이후에 공개되었으나 국제출원일 이전에 공개된 문헌

국제조사의 실제 완료일 2014년 12월 29일 (29.12.2014)	국제조사보고서 발송일 2014년 12월 30일 (30.12.2014)
--	---

ISA/KR의 명칭 및 우편주소 대한민국 특허청 (302-701) 대전광역시 서구 청사로 189, 4동 (둔산동, 정부대전청사) 팩스 번호 +82-42-472-7140	심사관 유진태 전화번호 +82-42-481-8530
---	------------------------------------



국제조사보고서에서 인용된 특허문헌	공개일	대응특허문헌	공개일
KR 10-2012-0080892 A	2012/07/18	없음	
KR 10-2007-0060957 A	2007/06/13	없음	
KR 10-2006-0031913 A	2006/04/14	CN 1764187 A CN 1764187 CO US 2006-0080402 A1	2006/04/26 2006/04/26 2006/04/13