(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2022-0166705
(43) 공개일자 2022년12월19일

(51) 국제특허분류(Int. Cl.)
*G06F 8/38* (2018.01) *G06F 16/958* (2019.01)
*G06F 8/34* (2018.01) *G06F 9/451* (2018.01)
*G06F 9/455* (2018.01) *G06F 9/54* (2018.01)
(52) CPC특허분류
*G06F 8/38* (2013.01)
*G06F 16/958* (2019.01)
(21) 출원번호 10-2021-7043185
(22) 출원일자(국제) 2021년10월28일
심사청구일자 2022년01월03일
(85) 번역문제출일자 2021년12월29일
(86) 국제출원번호 PCT/US2021/056933
(87) 국제공개번호 WO 2022/260695
국제공개일자 2022년12월15일
(30) 우선권주장
17/340,603 2021년06월07일 미국(US)
17/341,534 2021년06월08일 미국(US)

(71) 출원인
**유아이패스, 인크.**
미국 뉴욕주 10018 뉴욕 5번가 452 22층
(72) 발명자
**그리고레 미르체아**
미국 뉴욕주 10016 뉴욕 20번 플로어 파크 애비뉴 90
(74) 대리인
**김태홍, 김진희**

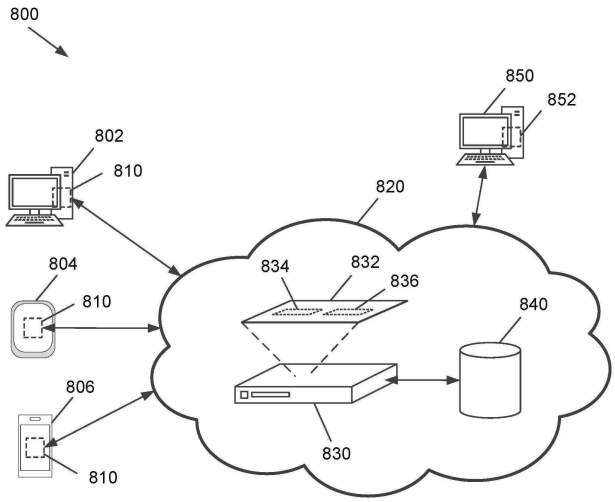
전체 청구항 수 : 총 30 항

(54) 발명의 명칭 **가상 머신들, 세션들, 및 컨테이너들에 대한 웹 기반 로보틱 프로세스 자동화 설계자 시스템들 및 자동화들**

## (57) 요 약

RPA 개발자들이 웹 서버리스 자동화들, UI(user interface) 자동화들, 및 다른 자동화들을 설계하고 구현하는 것을 가능하게 하는 웹 기반 RPA(robotic process automation) 설계자 시스템들이 개시된다. 그러한 웹 기반 RPA 설계자 시스템들은 개발자가 클라우드를 통해 사인 인하고 템플릿 프로젝트들, 개발자가 설계한 프로젝트들, 서비스들, 액티비티들 등의 리스트를 획득하는 것을 가능하게 할 수 있다. 따라서, RPA 개발이 중앙집중형 및 클라우드 기반일 수 있어서, 사용자의 컴퓨팅 시스템 상의 로컬 프로세싱 및 메모리 요건들을 감소시키고 RPA 설계자 기능을 중앙집중화하며, 더 나은 준수를 가능하게 한다. 웹 기반 RPA 설계자 시스템들에 의해 생성된 자동화들이 VM(virtual machine)들, 컨테이너들, 또는 운영 체제 세션들에서 배치되고 실행될 수 있다.

*대 표 도* - 도8



(52) CPC특허분류
*G06F 8/34* (2013.01)
*G06F 9/451* (2018.02)
*G06F 9/45504* (2013.01)
*G06F 9/547* (2013.01)

# 명 세 서

## 청구범위

### 청구항 1

시스템에 있어서,

웹 브라우저를 포함하는 개발자 컴퓨팅 시스템; 및

웹 기반(web-based) 로보틱 프로세스 자동화(robotic process automation; RPA) 설계자 애플리케이션을 제공하는 개발 서버를 포함하고,

상기 웹 기반 RPA 설계자 애플리케이션은 상기 웹 브라우저가,

상기 웹 기반 RPA 설계자 애플리케이션에 대한 코드를 다운로드하게 하도록,

상기 웹 브라우저에 상기 웹 기반 RPA 설계자 애플리케이션에 대한 웹 인터페이스를 디스플레이하게 하도록 - 상기 웹 인터페이스는 RPA 프로젝트들을 제작(create)하고 RPA 워크플로우들을 구성하는 기능을 제공하도록 구성됨 - ,

상기 웹 브라우저에, 구성된 RPA 워크플로우에 대한 자동화를 생성하게 하도록, 그리고

상기 웹 브라우저에서, 상기 생성된 자동화를 실행하고 검증하게 하도록 구성되는 것인, 시스템.

### 청구항 2

제 1 항에 있어서, 상기 생성된 자동화를 실행하고 검증하기 위한 코드가 상기 다운로드된 코드에 포함되는 것인, 시스템.

### 청구항 3

제 1 항에 있어서, 상기 코드는 상기 웹 기반 설계자 애플리케이션의 웹 인터페이스와 동일한 웹 브라우저의 탭의 일부로서 실행되는 것인, 시스템.

### 청구항 4

제 1 항에 있어서, 상기 생성된 자동화는 RPA 로봇 없이 런타임에 실행되도록 구성되는 것인, 시스템.

### 청구항 5

제 1 항에 있어서, 상기 생성된 자동화는 운영 체제 세션, 가상 머신(virtual machine; VM), 또는 컨테이너에서 원격으로 런타임에 생산 RPA 로봇에 의해 실행되도록 구성되는 것인, 시스템.

### 청구항 6

제 1 항에 있어서, 상기 생성된 자동화는 운영 체제 세션, 가상 머신(VM), 또는 컨테이너에서 원격으로 런타임에 생산 웹 브라우저에 의해 실행되도록 구성되는 것인, 시스템.

### 청구항 7

제 1 항에 있어서, 상기 웹 기반 RPA 설계자 애플리케이션은, 사용자 인터페이스(user interface; UI) 객체 저장소(object repository)로부터의 분류법(taxonomy)에 따라 체계화(organize)된 사용자 인터페이스(UI) 요소들을 리트리브(retrieve)하도록 구성되는 것인, 시스템.

### 청구항 8

제 7 항에 있어서, 상기 UI 객체 저장소는 애플리케이션들, 스크린들, UI 요소들, 및 UI 서술자(descriptor)들을, 상기 애플리케이션들이 스크린들을 그룹화하고, 상기 스크린들이 UI 요소들을 그룹화하며, 상기 UI 서술자들이 상기 UI 요소들의 위치 및 상기 UI 요소들과의 상호작용을 가능하게 하도록 포함하는 것인, 시스템.

**청구항 9**

제 1 항에 있어서, 상기 생성된 자동화는 서버리스 자동화(serverless automation)인 것인, 시스템.

**청구항 10**

제 1 항에 있어서, 상기 웹 기반 RPA 설계자 애플리케이션은, RPA 워크플로우들의 제작 및 수정을 용이하게 하는 캔버스를 제공하도록 구성되는 것인, 시스템.

**청구항 11**

제 1 항에 있어서, 상기 웹 기반 RPA 설계자 애플리케이션은, 자동화가 어떻게 시작할 것인지에 대한 옵션들을 제공하도록 구성되는 것인, 시스템.

**청구항 12**

제 1 항에 있어서, 상기 웹 기반 RPA 설계자 애플리케이션은, 비어 있는 액티비티(empty activity)를 디스플레이하고 사용자가 상기 비어 있는 액티비티를 클릭했을 때 액티비티 카테고리들 및 액션들을 제공하도록 구성되는 것인, 시스템.

**청구항 13**

제 1 항에 있어서, 상기 웹 기반 RPA 설계자 애플리케이션은, 액티비티의 유형에 기초하여 액티비티 구성 옵션들을 제공하도록 구성되는 것인, 시스템.

**청구항 14**

제 1 항에 있어서, 상기 웹 기판 RPA 설계자 애플리케이션은, 거버넌스 규칙(governance rule)들에 따라 사용자들이 정적 코드 분석 에러(static code analysis error)들을 갖는 자동화들을 게재(publish)하거나 러닝(running)하는 것을 방지하도록 구성되는 것인, 시스템.

**청구항 15**

제 1 항에 있어서, 상기 생성된 자동화가 검증에 실패했을 때, 상기 웹 기반 RPA 설계자 애플리케이션은, 상기 생성된 자동화를 생산 환경에 게재하기 전에 검증 에러들의 정정 또는 거버넌스 규칙들의 준수를 요하도록 구성되는 것인, 시스템.

**청구항 16**

컴퓨터 프로그램을 저장하는 비일시적 컴퓨터 판독가능 매체에 있어서, 상기 컴퓨터 프로그램은 상기 적어도 하나의 프로세서가,

웹 기반 로보틱 프로세스 자동화(RPA) 설계자 애플리케이션에 대한 코드를 웹 브라우저에 다운로드하게 하도록;

상기 웹 브라우저에 상기 웹 기반 RPA 설계자 애플리케이션에 대한 웹 인터페이스를 디스플레이하게 하도록 - 상기 웹 인터페이스는 RPA 프로젝트들을 제작하고 RPA 워크플로우들을 구성하는 기능을 제공하도록 구성됨 - ;

상기 웹 브라우저에, 구성된 RPA 워크플로우에 대한 자동화를 생성하게 하도록; 그리고

상기 웹 브라우저에서, 상기 생성된 자동화를 실행하고 검증하게 하도록 구성되는 것인, 컴퓨터 프로그램을 저장하는 비일시적 컴퓨터 판독가능 매체.

**청구항 17**

제 16 항에 있어서, 상기 생성된 자동화를 실행하고 검증하기 위한 코드가 상기 다운로드된 코드에 포함되는 것인, 비일시적 컴퓨터 판독가능 매체.

**청구항 18**

제 16 항에 있어서, 상기 코드는 상기 웹 기반 설계자 애플리케이션의 웹 인터페이스와 동일한 웹 브라우저의 탭의 일부로서 실행되는 것인, 비일시적 컴퓨터 판독가능 매체.

**청구항 19**

제 16 항에 있어서, 상기 생성된 자동화는 RPA 로봇 없이 런타임에 실행되도록 구성되는 것인, 비일시적 컴퓨터 판독가능 매체.

**청구항 20**

제 16 항에 있어서, 상기 생성된 자동화는 운영 체제 세션, 가상 머신(VM), 또는 컨테이너에서 원격으로 런타임에 생산 RPA 로봇에 의해 실행되도록 구성되는 것인, 비일시적 컴퓨터 판독가능 매체.

**청구항 21**

제 16 항에 있어서, 상기 생성된 자동화는 운영 체제 세션, 가상 머신(VM), 또는 컨테이너에서 원격으로 런타임에 생산 웹 브라우저에 의해 실행되도록 구성되는 것인, 비일시적 컴퓨터 판독가능 매체.

**청구항 22**

제 16 항에 있어서, 상기 웹 기판 RPA 설계자 애플리케이션은, 거버넌스 규칙들에 따라 사용자들이 정적 코드 분석 에러들을 갖는 자동화들을 게재하거나 러닝하는 것을 방지하도록 구성되는 것인, 비일시적 컴퓨터 판독가능 매체.

**청구항 23**

제 16 항에 있어서, 상기 생성된 자동화가 검증에 실패했을 때, 상기 웹 기반 RPA 설계자 애플리케이션은, 상기 생성된 자동화를 생산 환경에 게재하기 전에 검증 에러들의 정정 또는 거버넌스 규칙들의 준수를 요하도록 구성되는 것인, 비일시적 컴퓨터 판독가능 매체.

**청구항 24**

컴퓨터로 구현되는 방법(computer-implemented method)에 있어서,

컴퓨팅 시스템에 의해, 웹 기반 로보틱 프로세스 자동화(RPA) 설계자 애플리케이션에 대한 코드를 웹 브라우저에 다운로드하는 단계;

상기 컴퓨팅 시스템에 의해, 상기 웹 브라우저에 상기 웹 기반 RPA 설계자 애플리케이션에 대한 웹 인터페이스를 디스플레이하는 단계 - 상기 웹 인터페이스는 RPA 프로젝트들을 제작하고 RPA 워크플로우들을 구성하는 기능을 제공하도록 구성됨 - ;

상기 컴퓨팅 시스템에 의해, 상기 웹 브라우저에, 구성된 RPA 워크플로우에 대한 자동화를 생성하는 단계; 및

상기 컴퓨팅 시스템에 의해, 상기 웹 브라우저에서, 상기 생성된 자동화를 실행하고 검증하는 단계를 포함하고,

상기 생성된 자동화를 실행하고 검증하기 위한 코드가 상기 다운로드된 코드에 포함되는 것인, 컴퓨터로 구현되는 방법.

**청구항 25**

제 24 항에 있어서, 상기 코드는 상기 웹 기반 설계자 애플리케이션의 웹 인터페이스와 동일한 웹 브라우저의 탭의 일부로서 실행되는 것인, 컴퓨터로 구현되는 방법.

**청구항 26**

제 24 항에 있어서, 상기 생성된 자동화는 RPA 로봇 없이 런타임에 실행되도록 구성되는 것인, 컴퓨터로 구현되는 방법.

**청구항 27**

제 24 항에 있어서, 상기 생성된 자동화는 운영 체제 세션, 가상 머신(VM), 또는 컨테이너에서 원격으로 런타임에 생산 RPA 로봇에 의해 실행되도록 구성되는 것인, 컴퓨터로 구현되는 방법.

**청구항 28**

제 24 항에 있어서, 상기 생성된 자동화는 운영 체제 세션, 가상 머신(VM), 또는 컨테이너에서 원격으로 런타임에 생산 웹 브라우저에 의해 실행되도록 구성되는 것인, 컴퓨터로 구현되는 방법.

**청구항 29**

제 24 항에 있어서,

상기 웹 기반 RPA 설계자 애플리케이션에 의해, 거버넌스 규칙들에 따라 사용자들이 정적 코드 분석 에러들을 갖는 자동화들을 게재하거나 러닝하는 것을 방지하는 단계를 더 포함하는, 컴퓨터로 구현되는 방법.

**청구항 30**

제 24 항에 있어서,

상기 웹 기반 RPA 설계자 애플리케이션에 의해, 상기 생성된 자동화를 생산 환경에 게재하기 전에 검증 에러들의 정정 또는 거버넌스 규칙들의 준수를 요하는 단계를 더 포함하는, 컴퓨터로 구현되는 방법.

## *발명의 설명*

### *기 술 분 야*

[0001] 관련 출원들에 대한 상호참조

[0002] 본 출원은 2021년 6월 7일에 출원된 미국 특허 출원 제 17/340,603 호의 이익 및 이의 연속인 2021년 6월 8일에 출원된 미국 특허 출원 제 17/341,534 호의 이익을 주장한다. 이 이전에 출원된 출원들의 내용(subject matter)은 그 전체가 참조에 의해 본원에 포함된다.

[0003] 기술분야

[0004] 본 발명은 일반적으로 RPA(robotic process automation), 더 구체적으로 RPA 개발자들이 웹 서버리스 자동화(web serverless automation)들, UI(user interface) 자동화들, 및 다른 자동화들을 설계하고 구현하는 것을 가능하게 하는 웹 기반 RPA 설계자 시스템들에 관한 것이다.

### *배 경 기 술*

[0005] 많은 기구들의 정책들은 직원들에 의한 자신들 소유의 컴퓨팅 시스템들 상의 직접적인 추가 소프트웨어의 설치 또는 기존의 소프트웨어의 업그레이드를 허용하지 않는다. 그러한 정책들은 U.S. 및 다른 나라들에서, E.U. GDPR(General Data Protection Regulation), U.S. HIPAA(Health Insurance Portability and Accountability Act), 서드 파티 서비스 약관(third party terms of service) 등과 같은 법률 또는 합의를 준수하는 것을 보장하는 것을 시도하기 위해 시행될 수 있다. 현재의 RPA 개발 소프트웨어는, 새로운 버전 또는 증대(enhancement)가 사용자들에게 롤 아웃(roll out)될 수 있기 전에 IT에 의해 테스트되고 준법 감시인(compliance officers)에 의해 리뷰될 소프트웨어에 대한 새로운 버전들 또는 능력들을 요할 수 있는, 사용자의 컴퓨팅 시스템 상에 로컬로 상주하는 데스크탑 툴로서 존재한다. 또한, 많은 새로운 능력들 및 서비스들이 클라우드 기반이다. 따라서, RPA에의 향상되고/향상되거나 대안적인 접근법이 이로울 수 있다.

### *발명의 내용*

[0006] 본 발명의 특정 실시예들은 현재의 RPA 기술들에 의해 아직 완전히 식별되거나, 인식되거나, 또는 해결되지 않은 본 분야에서의 문제들 및 요구들에 대한 솔루션들을 제공할 수 있다. 예를 들어, 본 발명의 일부 실시예들은 RPA 개발자들이 웹 서버리스 자동화들, UI(user interface) 자동화들, 및 다른 자동화들을 설계하고 구현하는 것을 가능하게 하는 웹 기반 RPA 설계자 시스템들에 관한 것이다.

[0007] 실시예에서, 시스템은 웹 기반 RPA 설계자 애플리케이션을 제공하는 웹 브라우저 및 개발 서버를 포함하는 개발자 컴퓨팅 시스템을 포함한다. 웹 브라우저는 웹 기반 RPA 설계자 애플리케이션의 웹 인터페이스에 액세스하고 웹 기반 RPA 설계자 애플리케이션의 웹 인터페이스를 디스플레이하도록 구성된다. 웹 인터페이스는 RPA 프로젝트들을 제작(create)하고, RPA 워크플로우들을 구성하고, RPA 워크플로우 구성들을 웹 기반 설계자 애플리케이션에 제출(submit)하도록 구성된다. 웹 기반 RPA 설계자 애플리케이션은 웹 브라우저에 웹 인터페이스를 제공하고, 웹 브라우저로부터, 제출된 RPA 워크플로우 구성에 기초하여 자동화를 생성하고, 생성된 자동화를 RPA 로

봇을 통해 실행하고 생성된 자동화를 검증하며, RPA 로봇에 의한 실행 및 생성된 자동화의 검증의 결과들을 웹 브라우저에 제공하도록 구성된다. 생성된 자동화는 운영 체제 세션, VM, 또는 컨테이너에서 원격으로 RPA 로봇에 의해 실행되도록 구성된다.

[0008] 다른 실시예에서, 비일시적 컴퓨터 판독가능 매체가 웹 기반 RPA 설계자 애플리케이션용 컴퓨터 프로그램을 저장한다. 컴퓨터 프로그램은 적어도 하나의 프로세서가 컴퓨팅 시스템의 웹 브라우저에 웹 인터페이스를 제공하게 하도록 구성된다. 웹 인터페이스는 RPA 프로젝트들을 제작하고, RPA 워크플로우들을 구성하고, RPA 워크플로우 구성들을 웹 기반 설계자 애플리케이션에 제출하는 기능을 제공하도록 구성된다. 컴퓨터 프로그램은 또한, 적어도 하나의 프로세서가 웹 브라우저로부터, 제출된 RPA 워크플로우 구성에 기초하여 자동화를 생성하게 하고, 생성된 자동화를 RPA 로봇을 통해 실행하게 하며, 생성된 자동화를 검증하게 하도록 구성된다. 컴퓨터 프로그램은 또한, 적어도 하나의 프로세서가 RPA 로봇의 실행 및 생성된 자동화의 검증의 결과들을 웹 브라우저에 제공하게 하도록 구성된다. 생성된 자동화는 운영 체제 세션, VM, 또는 컨테이너에서 원격으로 런타임에 생산 RPA 로봇에 의해 실행되도록 구성된다.

[0009] 또 다른 실시예에서, 웹 기반 RPA를 위한 컴퓨터로 구현되는 방법(computer-implemented method)이 클라우드 기반 컴퓨팅 시스템에 의해, 웹 브라우저로부터, 제출된 RPA 워크플로우 구성에 기초하여 자동화를 생성하는 단계, 클라우드 기반 컴퓨팅 시스템에 의해, 생성된 자동화를 RPA 로봇을 통해 실행하고 생성된 자동화를 검증하는 단계를 포함한다. 컴퓨터로 구현되는 방법은 또한, 클라우드 기반 컴퓨팅 시스템에 의해, RPA 로봇의 실행 및 생성된 자동화의 검증의 결과들을 웹 브라우저에 제공하는 단계를 포함한다. 생성된 자동화는 운영 체제 세션, VM, 또는 컨테이너에서 원격으로 런타임에 생산 RPA 로봇에 의해 실행되도록 구성된다.

[0010] 다른 실시예에서, 클라우드 기반 시스템은 컴퓨터 프로그램 명령어들을 저장하는 메모리 및 컴퓨터 프로그램 명령어들을 실행하도록 구성된 적어도 하나의 프로세서를 포함한다. 컴퓨터 프로그램 명령어들은 적어도 하나의 프로세서가 복수의 RPA 로봇들을 런타임 서비스들로서 러닝(running)하게 하도록 구성된다. 복수의 RPA 로봇들은 클라우드 기반 시스템의 운영 체제 세션, VM, 또는 컨테이너에 위치된다. 컴퓨터 프로그램 명령어들은 또한, 적어도 하나의 프로세서가 복수의 RPA 로봇들에 의해 자동화들을 실행하게 하도록 구성된다. 복수의 자동화들은 복수의 RPA 로봇들에 의해 액세스가능하고 클라우드 기반 생산 서버의 운영 체제 세션, VM, 또는 컨테이너에 위치되거나 또는 클라우드 기반 시스템에 원격으로 위치된다.

[0011] 또 다른 실시예에서, 비일시적 컴퓨터 판독가능 매체가 컴퓨터 프로그램을 저장한다. 컴퓨터 프로그램은 적어도 하나의 프로세서가 RPA 로봇을 런타임 서비스로서 러닝하게 하도록 구성된다. RPA 로봇들은 운영 체제 세션, VM, 또는 컨테이너에 위치된다. 컴퓨터 프로그램은 또한, 적어도 하나의 프로세서가 RPA 로봇을 통해 자동화들을 실행하게 하도록 구성된다. 자동화는 RPA 로봇이 클라이언트 컴퓨팅 시스템의 웹 브라우저의 웹 확장부에 하나 이상의 명령을 전송하게 하도록 구성된다. 웹 확장부는 클라이언트 컴퓨팅 시스템의 웹 브라우저와 상호작용하거나, 클라이언트 컴퓨팅 시스템의 웹 브라우저의 하나 이상의 탭에서 하나 이상의 웹페이지로부터 정보를 획득하거나, 또는 이들 둘 다를 하도록 구성된다. 웹 확장부는 또한 RPA 로봇에, 요청된 상호작용이 발생했다는 확인을 제공하거나, 획득된 정보를 제공하거나, 또는 이들 둘 다를 하도록 구성된다.

[0012] 또 다른 실시예에서, 컴퓨터로 구현되는 방법은 클라우드 기반 컴퓨팅 시스템에 의해, RPA 로봇을 런타임 서비스로서 러닝하는 단계를 포함한다. RPA 로봇은 컴퓨팅 시스템의 운영 체제 세션, VM, 또는 컨테이너에 위치된다. 컴퓨터로 구현되는 방법은 또한, RPA 로봇에 의해 자동화를 실행하는 단계를 포함한다. 자동화는 RPA 로봇이 클라이언트 컴퓨팅 시스템의 로컬 RPA 확장 프로세스에 하나 이상의 요청을 전송하게 하도록 구성된다. 로컬 RPA 확장 프로세스는 클라이언트 컴퓨팅 시스템의 하나 이상의 애플리케이션 및/또는 프로세스와 상호작용하거나, 클라이언트 컴퓨팅 시스템 상에서 러닝되는 하나 이상의 애플리케이션 및/또는 프로세스로부터 정보를 획득하거나, 또는 이들 둘 다를 하도록 구성된다. 로컬 RPA 확장 프로세스는 또한 RPA 로봇에, 요청된 상호작용이 발생했다는 확인을 제공하거나, 획득된 정보를 제공하거나, 또는 이들 둘 다를 하도록 구성된다.

[0013] 다른 실시예에서, 시스템은 웹 기반 RPA 설계자 애플리케이션을 제공하는 웹 브라우저 및 개발 서버를 포함하는 개발자 컴퓨팅 시스템을 포함한다. 웹 기반 RPA 설계자 애플리케이션은 웹 브라우저가 웹 기반 RPA 설계자 애플리케이션에 대한 코드를 다운로드하게 하고 웹 브라우저에 웹 기반 RPA 설계자 애플리케이션용 웹 인터페이스를 디스플레이하게 하도록 구성된다. 웹 인터페이스는 RPA 프로젝트들을 제작하고 RPA 워크플로우들을 구성하는 기능을 제공하도록 구성된다. 웹 기반 RPA 설계자 애플리케이션은 또한, 웹 브라우저가 웹 브라우저에서, 구성된 RPA 워크플로우에 대한 자동화를 생성하게 하고, 웹 브라우저에서, 생성된 자동화를 실행하고 검증하게 하도록 구성된다.

[0014] 또 다른 실시예에서, 비일시적 컴퓨터 판독가능 매체가 컴퓨터 프로그램을 저장한다. 컴퓨터 프로그램은 적어도 하나의 프로세서가 웹 브라우저에 웹 기반 RPA 설계자 애플리케이션에 대한 코드를 다운로드하게 하고 웹 브라우저에 웹 기반 RPA 설계자 애플리케이션용 웹 인터페이스를 디스플레이하게 하도록 구성된다. 웹 인터페이스는 RPA 프로젝트들을 제작하고 RPA 워크플로우들을 구성하는 기능을 제공하도록 구성된다. 컴퓨터 프로그램은 또한, 적어도 하나의 프로세서가 웹 브라우저에서, 구성된 RPA 워크플로우에 대한 자동화를 생성하게 하고, 웹 브라우저에서, 생성된 자동화를 실행하고 검증하게 하도록 구성된다.

[0015] 또 다른 실시예에서, 컴퓨터로 구현되는 방법은 컴퓨팅 시스템에 의해 웹 브라우저에 웹 기반 RPA 설계자 애플리케이션에 대한 코드를 다운로드하는 단계를 포함한다. 컴퓨터로 구현되는 방법은 또한, 컴퓨팅 시스템에 의해, 웹 브라우저에 웹 기반 RPA 설계자 애플리케이션에 대한 웹 인터페이스를 디스플레이하는 단계를 포함한다. 웹 인터페이스는 RPA 프로젝트들을 제작하고 RPA 워크플로우들을 구성하는 기능을 제공하도록 구성된다. 컴퓨터로 구현되는 방법은 컴퓨팅 시스템에 의해 웹 브라우저에, 구성된 RPA 워크플로우에 대한 자동화를 생성하는 단계, 및 컴퓨팅 시스템에 의해 웹 브라우저에서, 생성된 자동화를 실행하고 검증하는 단계를 더 포함한다. 생성된 자동화를 실행하고 검증하기 위한 코드가 다운로드된 코드에 포함된다.

[0016] 다른 실시예에서, 클라우드 기반 시스템은 컴퓨터 프로그램 명령어들을 저장하는 메모리 및 컴퓨터 프로그램 명령어들을 실행하도록 구성된 적어도 하나의 프로세서를 포함한다. 컴퓨터 프로그램 명령어들은 적어도 하나의 프로세서가 런타임에 제 1 웹 브라우저에서 자동화를 러닝하게 하도록 구성된다. 제 1 웹 브라우저는 클라우드 기반 시스템의 운영 체제 세션, VM, 또는 컨테이너에 위치된다.

[0017] 또 다른 실시예에서, 비일시적 컴퓨터 판독가능 매체가 컴퓨터 프로그램을 저장한다. 컴퓨터 프로그램은 적어도 하나의 프로세서가 런타임에 제 1 웹 브라우저에서 자동화를 러닝하게 하도록 구성된다. 제 1 웹 브라우저는 클라우드 기반 시스템의 운영 체제 세션, VM, 또는 컨테이너에 위치된다. 자동화는 RPA 로봇이 클라이언트 컴퓨팅 시스템의 제 2 웹 브라우저의 웹 확장부에 하나 이상의 명령을 전송하게 하도록 구성된다. 웹 확장부는 클라이언트 컴퓨팅 시스템의 제 2 웹 브라우저와 상호작용하거나, 클라이언트 컴퓨팅 시스템의 제 2 웹 브라우저의 하나 이상의 탭에서 하나 이상의 웹페이지로부터 정보를 획득하거나, 또는 이들 둘 다를 하도록 구성된다. 웹 확장부는 또한 제 1 웹 브라우저에, 요청된 상호작용이 발생했다는 확인을 제공하거나, 획득된 정보를 제공하거나, 또는 이들 둘 다를 하도록 구성된다.

[0018] 또 다른 실시예에서, 컴퓨터로 구현되는 방법은 컴퓨팅 시스템에 의해 런타임에 웹 브라우저에서 자동화를 실행하는 단계를 포함한다. 웹 브라우저는 클라우드 기반 시스템의 운영 체제 세션, VM, 또는 컨테이너에 위치된다. 자동화는 웹 브라우저가 클라이언트 컴퓨팅 시스템의 로컬 RPA 확장 프로세스에 하나 이상의 요청을 전송하게 하도록 구성된다. 로컬 RPA 확장 프로세스는 클라이언트 컴퓨팅 시스템의 하나 이상의 애플리케이션 및/또는 프로세스와 상호작용하거나, 클라이언트 컴퓨팅 시스템 상에서 러닝되는 하나 이상의 애플리케이션 및/또는 프로세스로부터 정보를 획득하거나, 또는 이들 둘 다를 하도록 구성된다. 로컬 RPA 확장 프로세스는 또한 웹 브라우저에, 요청된 상호작용이 발생했다는 확인을 제공하거나, 획득된 정보를 제공하거나, 또는 이들 둘 다를 하도록 구성된다.

## *도면의 간단한 설명*

[0019] 본 발명의 특정 실시예들의 이점들이 쉽게 이해될 것을 위해, 위에서 간략히 설명된 본 발명의 더 구체적인 설명이 첨부된 도면들에 예시된 특정 실시예들을 참조하여 렌더링될 것이다. 이 도면들이 본 발명의 전형적인 실시예들만을 도시하고 따라서 본 발명의 범위를 제한하는 것으로 간주되어서는 안된다는 점이 이해되어야 하는 한편, 본 발명은 첨부한 도면들의 사용을 통해 추가적인 특이성 및 상세성으로 설명되고 해석(explain)될 것이다.

도 1은 본 발명의 실시예에 따른, 웹 기반 RPA 설계자 시스템을 예시하는 아키텍처도(architectural diagram)이다.

도 2는 본 발명의 실시예에 따른, 웹 기반 RPA 설계자 시스템 중 일부 또는 모두를 구현하도록 구성된 컴퓨팅 시스템을 예시하는 아키텍처도이다.

도 3a는 본 발명의 실시예에 따른, 웹 기반 RPA 시스템을 예시하는 아키텍처도이다.

도 3b는 본 발명의 실시예에 따른, 컨테이너, VM, 또는 운영 체제 세션에서 러닝되는 자동화들 및 RPA 로봇들을 예시한다.

도 4a는 본 발명의 실시예에 따른, 웹 기반 RPA 개발 애플리케이션 웹페이지의 홈 페이지 뷰를 예시한다.

도 4b는 본 발명의 실시예에 따른, 새로운 자동화 윈도우가 디스플레이되는 웹 기반 RPA 개발 애플리케이션 웹페이지의 홈 페이지 뷰를 예시한다.

도 4c는 본 발명의 실시예에 따른, 웹 기반 RPA 개발 애플리케이션 웹페이지의 캔버스 뷰를 예시한다.

도 4d는 본 발명의 실시예에 따른, 액티비티 카테고리(activity category)들 및 빠른 액션 윈도우가 있는 웹 기반 RPA 개발 애플리케이션 웹페이지의 캔버스 뷰를 예시한다.

도 4e는 본 발명의 실시예에 따른, Excel® 액티비티 윈도우가 있는 웹 기반 RPA 개발 애플리케이션 웹페이지의 캔버스 뷰를 예시한다.

도 4f는 본 발명의 실시예에 따른, 구성되지 않은 Excel® 파일 액티비티 사용이 있는 웹 기반 RPA 개발 애플리케이션 웹페이지의 캔버스 뷰를 예시한다.

도 4g는 본 발명의 실시예에 따른, Excel® 파일 액티비티 사용이 구성된 후의 웹 기반 RPA 개발 애플리케이션 웹페이지의 캔버스 뷰를 예시한다.

도 5는 본 발명의 실시예에 따른, 설계 시간에 웹 기반 RPA 개발을 수행하기 위한 프로세스를 예시하는 흐름도이다.

도 6은 본 발명의 실시예에 따른, 웹 확장부와 상호작용하는 런타임에 자동화를 실행하기 위한 프로세스를 예시하는 흐름도이다.

도 7은 본 발명의 실시예에 따른, VM 또는 컨테이너에서 러닝되는 런타임에 자동화를 실행하기 위한 프로세스를 예시하는 흐름도이다.

도 8은 본 발명의 실시예에 따른, 로컬 애플리케이션들과 상호작용하도록 구성된 웹 기반 RPA 시스템을 예시하는 아키텍처도이다.

도 9는 본 발명의 실시예에 따른, RPA 로봇과는 상이한 컴퓨팅 시스템 상에서 러닝되는 로컬 RPA 확장 프로세스와 상호작용하는 런타임에 자동화를 실행하기 위한 프로세스를 예시하는 흐름도이다.

도 10은 본 발명의 실시예에 따른, RPA 로봇이 없는 웹 브라우저를 통해 설계 시간에 웹 기반 RPA 개발을 수행하기 위한 프로세스를 예시하는 흐름도이다.

도 11은 본 발명의 실시예에 따른, 컨테이너, VM, 또는 운영 체제 세션의 웹 브라우저에서 런타임에 자동화를 실행하기 위한 프로세스를 예시하는 흐름도이다.

달리 표시되지 않는 한, 유사한 참조 부호들은 첨부된 도면들 전반에 걸쳐 일관적으로 대응하는 피처들을 나타낸다.

## *발명을 실시하기 위한 구체적인 내용*

[0020] 일부 실시예들은 RPA 개발자들이 웹 서버리스 자동화들, UI 자동화들, 및 다른 자동화들을 설계하고 구현하는 것을 가능하게 하는 웹 기반 RPA 설계자 시스템들에 관한 것이다. 일부 실시예들에서, 웹 기반 RPA 설계자 시스템, 시스템에 의해 생성된 자동화들, 또는 이들 둘 다가 예를 들어 클라우드 기반 환경에 제공된다. 그러한 실시예들은 데스크탑 설치를 요하지 않을 수 있고, 사용자가 클라우드를 통해 사인 인(sign in)하고 템블릿 프로젝트들, 개발자가 설계한 프로젝트들, 서비스들, 액티비티들의 리스트를 획득하는 것을 가능하게 할 수 있다. 따라서, RPA 개발이 일부 실시예들에서 중앙집중형 및 클라우드 기반일 수 있어서, 사용자의 컴퓨팅 시스템 상의 로컬 프로세싱 및 메모리 요건들을 감소시키고 RPA 설계자 기능을 중앙집중화하며, 더 나은 준수를 가능하게 한다. 그러한 실시예들은 또한, RPA 서비스 제공자가 그들 선에서(on their end) RPA 설계자 시스템을 유지하고 업그레이드하는 것을 가능하게 할 수 있어, 업그레이드와 연관된 메인터넌스 문제들 및 비용을 감소시키고 고객들에 의해 새로운 기능을 추가한다. 이는 또한 더 빠르고 더 정기적인 업데이트들, 새로운 피처들의 출시(rollout), 버그 수정(bug fixes) 등을 가능하게 할 수 있다.

[0021] 일부 실시예들에서, RPA 개발자는 웹 브라우저를 통해 클라우드 기반 RPA 설계자 시스템에 액세스할 수 있다. 누군가는 처음에, 그로부터 프로젝트들이 제작될 수 있고 사용자가 구축하기 원하는 것에 관한 정보가 제공될

수 있는 홈 페이지(예를 들어, 도 4a를 보라)를 맞이할 수 있다. 이어서 사용자가 원하는 RPA 자동화를 구축할 수 있는 "캔버스" 웹페이지가 나타날 수 있다(예를 들어, 도 4c를 보라). 예를 들어, 일부 실시예들에서, 사용자는 RPA 워크플로우에서 RPA 액티비티들을 구성할 수 있고 자동화가 어떻게 러닝될 것인지를 제어할 수 있다(예를 들어, 수동으로, 트리거에 기초하여, 스케줄대로 등). RPA 워크플로우가 의도한대로 설계된 후, 사용자가 일부 실시예들에서 RPA 워크플로우를 게재(publish)할 수 있고, 자동화가 원하는대로 배치(deploy)될 수 있다.

[0022] 일부 실시예들에서, 웹 기반 RPA 설계자 시스템은 사용자에게 서버측 워크플로우(server-side workflow)들, 서버리스 워크플로우들, 장기 러닝 워크플로우(long-running workflow)들(예를 들어, 동작을 재개하기 전에 특정 프로세스들이 완료되는 것 또는 특정 입력을 대기할 수 있는 워크플로우들), 지휘 프로세스(orchestration process)들, 헤드리스 워크플로우(headless workflow)들, 백그라운드 프로세스들, 비 UI 자동화 워크플로우들, 통상적으로 사용자 상호작용을 요하지 않는 워크플로우들을 설계하고, 편집(edit)하고 테스트하는 능력을 제공한다. 웹 기반 RPA 설계자 시스템에서 제작된 서버리스 워크플로우들은 예를 들어 클라우드에 저장될 수 있고 사용자들은 웹 브라우저를 통해 수동으로 서버측 워크플로우들을 러닝하고 디버깅할 수 있다. 패키지들이 클라우드(예를 들어, 서버측) 지휘자 및 보조자에게 게재될 수 있다. 프로젝트들이 패키지로서 게재될 때 컴파일될 수 있다.

[0023] 트리거들은 일부 실시예들에서 API(application programming interface) 트리거들이다. API 트리거들은, 애플리케이션 또는 OS(operating system)로부터 올 수 있는 프로그램 대 프로그램 호출(program-to-program call)들이며, 사용자가 사용자 인터페이스와 상호작용하거나(예를 들어, 버튼을 클릭하는 것) 특정 정보를 입력하는 것, 이메일이 수신된 통지, 그래픽적 요소가 스크린 상에 나타나는 것, 키가 눌리는 것, 파일이 변경되는 것, 요소 속성이 변경되는 것, OS-레벨 이벤트 등과 같은 다양한 사용자 또는 소프트웨어 이벤트로부터 초래할 수 있다. 트리거들은 이벤트들이 클래스들의 인스턴스들인 클래스들로서 구현될 수 있고, 이벤트들이 이벤트 리스너(event listener)들을 통해 검출될 수 있다.

[0024] 일부 실시예들의 웹 기반 RPA 설계자 시스템은 푸시 트리거(push trigger)들 및 풀 트리거(pull trigger)들과 같은 적어도 2개의 유형의 API 트리거들에 응답하도록 구성된 UI 자동화들을 개발할 수 있다. 풀 트리거들은 반복적으로(예를 들어, OS가 정보를 찾고 있는 프로세스들을 순환할 때마다, 1초마다, 1분마다) 러닝될 수 있고 새로운 이메일이 수신되었음을 표시하는 플래그, 텍스트가 변경되었는지의 여부를 보기 위해 텍스트 필드의 값을 체크하는 것과 같은 해당 트리거에 관련된 변경이 발생했는지의 여부를 체크할 수 있다. 푸시 트리거들은, 자신들이 파이어링(firing)될 때 리스너가 반응하게 하는 애플리케이션 생성 이벤트들[예를 들어, 웹 후크(web hook)들]이다. 이는 또한 서버로부터의 요청에 인한, 사용자가 버튼을 클릭하는 것 등에 인한, 새로운 이메일이 수신된 것에 응답하여 발생할 수 있다.

[0025] 웹 트리거들과 "데스크탑"(로컬) 트리거들 사이에 또한 차이가 있다. 데스크탑 트리거들은 사용자의 컴퓨팅 시스템 상에서 발생하고 있는 것에 기초하는 경향이 있다. 웹 트리거들은, 다른 한편으로, 웹 서버 상에서 발생하고 전역적일 수 있다. 다수의(예를 들어, 2개의, 10개의, 모든 등) 클라이언트들이 웹 트리거를 수신할 수 있다.

[0026] 그러나, 일부 실시예들에서, UI 자동화들이 웹 상에 존재하는 애플리케이션들 위에 웹 인터페이스로부터 구축될 수 있다. 예를 들어, Workday® 용 웹페이지 상에 탭이 있으면, RPA 개발자는 이 페이지와 상호작용하고 그 위에 자동화를 제작하기 위해 일부 실시예들의 클라우드 기반 RPA 설계자 시스템을 사용할 수 있다. 이 자동화는 서버측에서(예를 들어, 클라우드에서) 러닝될 수 있고 본원에서 이후에 더 상세히 논의되는 바와 같이, VM(virtual machine)에서 실행될 수 있다. 다수의 VM들이 단일 서버 상에서 실행될 수 있고, 각각의 VM은 전형적으로 자체 운영 체제 및 애플리케이션들을 갖는다.

[0027] 일부 실시예들에서, UI 자동화들에 대해, 사용자가 버튼들, 텍스트 필드들 등과 상호작용하는 것을 가능하게 하는 UI 요소들의 분류법(taxonomy)이 제공될 수 있다. 분류법은 계층적일 수 있고 어떤 요소들이 어떤 페이지 및/또는 페이지의 일부에 위치되는지와 같이 UI 요소들 간의 관계를 특정할 수 있다. 예를 들어, 사용자가 분류법으로부터 OK 버튼을 선택할 수 있고 자동화가 그 버튼을 클릭하게 할 수 있다.

[0028] 일부 실시예들에서, 웹 기반 UI 자동화들은 정보를 획득하고 UI 자동화를 로컬로 구현하기 위해 웹 브라우저 내의 탭들에 설치될 수 있고 이 탭들과 통신할 수 있는 웹 브라우저 확장부들(예를 들어, Google Chrome® 확장부, 다른 웹 브라우저용 플러그인, 또는 웹 브라우저의 기능을 확장시키기 위한 임의의 다른 적합한 메커니즘)에 의

해 지원된다. 예를 들어, 그러한 확장부들은 브라우저 상에서 클릭할 수 있고, 브라우저 요소들 등으로부터 텍스트를 얻을 수 있으며, 이 정보를 RPA 로봇에 제공할 수 있다. 일부 실시예들에서, 웹 브라우저 및 RPA 설계자 애플리케이션은 둘 다 RPA 개발자의 컴퓨팅 시스템 상에 위치될 수 있다. 그러나, 특정 실시예들에서, 웹 기반 RPA 설계자 애플리케이션이 원격으로 위치되고 RPA 개발자는 웹 브라우저를 통해 웹 기반 RPA 설계자 애플리케이션과 상호작용한다. 이는, 웹 기반 RPA 설계자 애플리케이션이 RPA 설계자 애플리케이션을 제작했고 애플리케이션을 최신으로 자동으로 유지하는 회사에 의해 유지되고 업그레이드되는 것을 가능하게 할 수 있다.

[0029] 런타임에, 확장부가 사용자의 컴퓨팅 시스템 상에 위치될 수 있고 자동화 코드가 원격으로(예를 들어, 서버 상에, 클라우드 기반 환경에 등) 위치될 수 있다. 웹 애플리케이션이 열리고 러닝되고 있는 사용자의 로컬 컴퓨팅 시스템 상에 운영 체제 세션이 존재할 수 있다. UI 자동화는, 웹 확장부를 통해, 웹 애플리케이션이 원하는 액션을 수행하도록 지시한다. 위에 따라, 일부 실시예들에서, 자동화들이 서버 상의 VM에서 러닝될 수 있고 자신의 기능을 구현하기 위해 사용자 컴퓨팅 시스템 상의 웹 브라우저 확장부와 상호작용할 필요가 없을 수 있다.

[0030] 일부 실시예들에서, Windows® 또는 일부 다른 운영 체제 상의 상호적 세션이 요구되지 않는 "헤드리스 브라우저"가 사용될 수 있다. 그러한 헤드리스 브라우저는 일부 실시예들에서, Linux® 컨테이너 내와 같이 임의의 OS 상에서 러닝될 수 있다. 컨테이너들은 이들이 러닝되는 호스트 시스템으로부터 이들이 격리되도록 애플리케이션들을 포함한다. 라이브러리들 및 자동화가 사용하는 다른 종속물들과 함께 자동화가 컨테이너에 패키징될 수 있다. 자동화가 웹 기반 RPA 설계자 시스템으로 설계될 수 있지만, API로부터 호출될 수 있고 호출이 서버리스 세션으로 변환될 수 있다. 이와 같이, 헤드리스 브라우저 자동화들은 예를 들어, 열려 있는 웹 브라우저, 상호적 세션, 및 Windows® VM을 요하지 않는다.

[0031] 일부 실시예들에서, 서버리스 세션은 예를 들어, 컨테이너, VM, 또는 Windows® 세션 내에 있을 수 있다. 컴퓨팅 시스템이 자신의 하드웨어 리소스들로 핸들링될 수 있는 많큼 많은 서버리스 세션들이 실행될 수 있다. 이 서버리스 워크플로우들은 예를 들어, 운영 체제에 러닝 프로세스들로서 나타날 수 있다.

[0032] 헤드리스 브라우저용 UI 자동화를 설계하기 위해, 데스크탑 RPA 설계자 애플리케이션과 유사한 설계 경험이 제공될 수 있고, 여기서 웹 브라우저와의 사용자 상호작용들이 기록될 수 있고/있거나 스크린 기능 상의 표시, 요소 기능을 캡처하는 것 등이 제공될 수 있다. 사용자가 상호작용하고 있는 현재 브라우저 탭 위의 한 레벨에 레코더가 위치될 수 있다. 이는, Chrome® 확장부 또는 브라우저 자체에 대한 액세스를 주는 일부 다른 브라우저 확장부를 통해 구현될 수 있다. 확장부는 설계 시간에, 로딩된 페이지에 대해 UI 자동화를 설계하기 위해 열려 있는 탭들과 상화작용할 수 있다(예를 들어, 요소들을 클릭하는 것, 테이블들을 제작하는 것 등). 게재되거나 달리 런타임 사용에 이용가능하게 만들어지면, 헤드리스 브라우저 UI 자동화가 VM 세션에서 보다는 예를 들어, Linux® 컨테이너에서 실행되도록 스케줄링될 수 있다. 이것이 어떻게 설계되었는지를 고려하면 본원에서 "UI 자동화"로 불리우는 반면, 헤드리스 브라우저 UI 자동화가 러닝될 때 사용자에게 실제로 UI가 디스플레이되지 않는다.

[0033] 일부 실시예들의 웹 기반 RPA 설계자 시스템은 사용자들이 UI 자동화들 및 다른 자동화들을 구축하는 것을 가능하게 한다. 본원에서 사용되는 바와 같이, "자동화들"은 일부 실시예들의 웹 기반 RPA 설계자 시스템에 의해 개발되는 것과 같은, RPA 워크플로우 논리를 구현하는 코드이다. 자동화가 러닝되는 환경은 자동화가 행하는 것에 의존할 수 있다. 자동화가 UI 상호작용들을 수행하는 것을 보여 줄 수 있도록 의도되고/의도되거나 사용자에게 입력을 요청하는 유인 자동화이면, UI 자동화가 성공적으로 러닝될 수 있도록 Windows®이 설치된 VM이 사용될 수 있다. 그러나, UI 자동화가 헤드리스 브라우저 자동화와 같이 보여 줄 수 있을 또는 사용자 입력을 요청할 필요가 없으면, 그러한 자동화는 예를 들어, Linux® 컨테이너에서 러닝될 수 있다. 웹 확장부는 최종 사용자(end user)의 컴퓨팅 시스템으로부터 정보가 필요되고 UI 자동화가 헤드리스인 웹 기반 애플리케이션용으로 사용될 수 있다. 그렇지 않고, 원한다면, 이 기능이 VM을 스피닝 업(spinning up)함으로써 구현될 수 있다.

[0034] VM들 또는 Linux® 컨테이너들은, RPA 개발자가 UI 자동화 또는 다른 자동화를 제작하고 자동화를 사용자의 컴퓨팅 시스템 상에 로컬로 설치하고 러닝하지 않고 직접적으로 이를 러닝하고자 할 때 일부 실시예들에서 사용된다. 그렇지 않으면 사용자의 컴퓨팅 시스템 상에서 발생했을 액션들이 VM 또는 컨테이너에서 대신 일어난다. 그러한 VM들 또는 컨테이너들을 관리하기 위해, 자동으로 또는 수동으로 프로비저닝된 풀(provisioned pool)들이 사용될 수 있다. 자동으로 프로비저닝된 풀은 탄력적(elastic)일 수 있고 풀 내의 VM들 또는 컨테이너들의

수가 요구에 따라 증가되거나 감소될 수 있다. 수동을 프로비저닝된 풀은, 다른 한편으로, 인간(예를 들어, 관리자)에 의해 특정된 수의 VM들 또는 컨테이너들을 포함할 수 있다.

[0035] "서버리스"(무인) 자동화들은 러닝되기 위해 전형적 서버를 필요로 하지 않고, 예를 들어 Linux® 컨테이너에서 러닝될 수 있다. 일부 실시예들의 웹 기반 RPA 설계자 시스템을 사용하여 개발된 서버리스 자동화들은 특정 시간에 러닝되도록 스케줄링되거나 그렇지 않으면 사용자 상호작용 없이 러닝될 수 있다. 예를 들어, 특정 이메일이 수신되었을 때, Slack®을 통해 메시지가 전송될 수 있다. 지휘자 애플리케이션 상에서 일(job)이 실패했을 때, 일이 재러닝될 수 있거나, 또는 일이 특정 에러와 함께 실패하면, 일을 재러닝하는 대신 관리자에게 이메일이 전송될 수 있다. 본 발명의 범위로부터 벗어나지 않고 임의의 적합한 자동화들이 개발되고 구현될 수 있다. UI 자동화들은 또한 특정 시간에 또는 사용자 요청들 또는 액션들에 응답하여 러닝될 수 있다.

[0036] 일부 실시예들에서, 웹 기능을 자동화하기 위해 UI 액티비티들의 전용 세트가 제공된다. 예를 들어, 웹 컴포넌트들을 인식하고 웹 컴포넌트들과 상호작용하는 액티비티들은, 잠재적으로 브라우저들 간의 시각적 차이들을 고려하여, 웹 애플리케이션들에 존재하는 경향이 있는 형태들 및 레이아웃들을 더 정확히 검출하도록 테일러링(tailoring)될 수 있다.

[0037] 일부 실시예들에서, 웹 기반 RPA 설계자 애플리케이션과 데스크탑 기반 RPA 설계자 애플리케이션 사이에 통합이 제공될 수 있다. 예를 들어, 주어진 UI 자동화에 대한 프로젝트 데이터가 XAML과 같이 설계자 애플리케이션 버전들 둘 다가 판독할 수 있는 공통 포맷으로 저장될 수 있다. 이는 RPA 개발자가 트래블링(traveling) 동안 누군가의 랩탑 상에서 오프라인으로 UI 자동화를 작업하고, 프로젝트를 인터넷 액세스가 이용가능할 때 클라우드에 업로드하며, 웹 브라우저를 사용하여 UI 자동화를 작업하는 것을 계속하는 것을 가능하게 할 수 있다. 데이터가 또한 일부 실시예들에서 설계자 애플리케이션의 웹 버전과 데스크탑 버전 사이에서 동기화될 수 있다. 또한, 웹 기반 설계자 애플리케이션의 사용은 각각의 사용자에 대한 로컬 메모리 풋프린트(local memory footprint) 및 프로세싱 요건들을 감소시킬 수 있을 뿐만 아니라 다수의 웹 사용자들 간의 설계 협업(design collaboration)을 용이하게 할 수 있다.

[0038] 일부 실시예들에서, 자동화들은, 특정 플랫폼에 대해 자동화 패키지를 수정하지 않고 VM, 컨테이너, 또는 운영체제 세션에서 실행되는 RPA 로봇들에 의해 실행되도록 설계되고 패키징된다. 환언하면, 예를 들어 패키지를 수정하지 않고 VMware® VM, Linux® 컨테이너, 또는 Windows® 세션에서 동일한 패키지가 러닝될 수 있다. RPA 로봇들은 특정 타겟 플랫폼용으로 설계될 수 있고 특정 타겟 플랫폼 상에 배치될 수 있지만, 자동화들을 실행하기 위해 공통 포맷이 사용될 수 있다. 예를 들어, XAML 또는 각각의 플랫폼 상의 RPA 로봇들이 컴파일된 머신 코드[예를 들어, DLL(digital link library)들]에서 판독되고 실행되도록 설계될 수 있는 일부 다른 포맷으로 워크플로우가 설명될 수 있다. 이는 코드 난독화(code obfuscation) 및 보안을 제공할 수 있을 뿐만 아니라, 잠재적으로 자동화를 더 빠르게 러닝되게 할 수 있다. 그러나, 특정 실시예들에서, 자동화가 XAML, XML, 일반 텍스트(plain text), 또는 일부 다른 적합한 포맷의 스크립트일 수 있다.

[0039] 도 1은 본 발명의 실시예에 따른, 웹 기반 RPA 설계자 시스템(100)을 예시하는 아키텍처도이다. RPA 시스템(100)은, 개발자가 클라우드 기반 웹 애플리케이션(120)을 통해 RPA 워크플로우들을 설계하고 구현하는 것을 가능하게 하는 (예를 들어, RPA 개발자의 컴퓨팅 시스템 상에서 러닝되는) 웹 브라우저(110)를 포함한다. 웹 브라우저(110)는 웹 애플리케이션(120)을 통해, 애플리케이션 통합을 위한 솔루션을 제공할 수 있을 뿐만 아니라, 서드 파티 애플리케이션들, 관리 IT(Information Technology) 작업들, 및 비지니스 IT 프로세스들을 자동화한다. 웹 브라우저(110) 및 웹 애플리케이션(120)은, 프로세스의 그래픽적 표현을 포함하는 자동화 프로젝트의 개발을 용이하게 할 수 있다. 간단히 말해서, 웹 브라우저(110) 및 웹 애플리케이션(120)은 서비스로서 러닝되는 RPA 로봇들에 의해 구현될 수 있는 RPA 워크플로우들의 개발 및 자동화들의 배치를 용이하게 한다.

[0040] 자동화 프로젝트는, "액티비티(activity)들"로서 본원에서 정의되는, 워크플로우에서 개발된 단계들의 커스텀 세트(custom set) 간의 실행 순서 및 관계의 RPA 개발자 제어를 줌으로써 규칙 기반 프로세스들의 자동화를 가능하게 한다. 웹 애플리케이션(120)의 실시예의 한 상업적 예는 UiPath Studio Web™이다. 각각의 액티비티는 버튼을 클릭하는 것, 파일을 판독하는 것, 로그 패널에 기록하는 것 등과 같은 액션을 포함할 수 있다. 일부 실시예들에서, 워크플로우들이 네스팅되거나(nested) 임베딩될 수 있다(embedded).

[0041] 일부 유형들의 워크플로우들은 시퀀스들, 플로우차트들, FSM(Finite State Machine)들, 및/또는 전역 예외 처리기(global exception handler)들을 포함할 수 있지만, 이에 제한되는 것은 아니다. 시퀀스들은 선형 프로세스들에 특히 적합할 수 있어, 워크플로우를 어지럽히지(clutter) 않고 한 액티비티로부터 다른 액티비티로의 플로

우를 가능하게 한다. 플로우차트들은 더 복잡한 비지니스 논리에 특히 적합할 수 있어, 다수의 브랜칭 로직 연산자(branching logic operator)들을 통해 더 다양한 방식으로 결정들의 통합 및 액티비티들의 연결을 가능하게 한다. FSM들은 큰 워크플로우들에 특히 적합할 수 있다. FSM들은 조건(condition)[즉, 전환(transition)] 또는 액티비티에 의해 트리거되는 유한 수의 상태들을 이들의 실행에서 사용할 수 있다. 전역 예외 처리기들은, 실행 에러에 직면했을 때 그리고 디버깅 프로세스들에 대해 워크플로우 거동을 결정하기 위해 특히 적합할 수 있다.

[0042] 워크플로우가 웹 브라우저(110)에서 개발되면, 자동화의 실행이 웹 애플리케이션(120)을 사용하여 테스트될 수 있다. 그러나, 일부 실시예들에서, 웹 브라우저(110)가 웹 애플리케이션(120)의 자동화 생성 및 테스트 기능을 포함한다. 자동화가 적절히 기능하고 있고 생산 서버(140) 상의 배치를 위한 준비가 된 후, 자동화가 머신 판독가능 코드 또는 스크립트의 형태로 생산 서버(140)의 컨테이너, VM, 또는 서버 운영 체제 세션에 배치된다. 일부 실시예들에서, 로봇들(130)이 지휘자(150)에 의해 지휘된다. 이어서 RPA 로봇들(130)이 배치된 자동화들을 원하는대로 실행할 수 있다. 관리될 수 있는 로봇들(130)의 유형들은 유인 로봇들(132), 무인 로봇들(134), 개발 로봇들[무인 로봇들(134)과 유사하지만, 개발 및 테스트 목적들을 위해 사용됨], 비생산 로봇들[유인 로봇들(132)과 유사하지만, 개발 및 테스트 목적들을 위해 사용됨]을 포함하지만, 이에 제한되는 것은 아니다. 유인 로봇들(132)은 사용자 이벤트들에 의해 트리거되고 동일한 컴퓨팅 시스템 상의 인간과 함께 동작한다. 예를 들어, 유인 로봇들(132)은 UI 자동화를 수행할 수 있고, 채울 수 있는 양식들 등을 제공할 수 있다. 유인 로봇들(132)은 인간 사용자가 다양한 작업들을 달성하는 것을 도울 수 있고, 사용자 이벤트들에 의해 트리거될 수 있다. 특정 실시예들에서, 유인 로봇들(132)은 로봇 트레이(robot tray)로부터만 또는 명령 프롬프트로부터만 시작될 수 있다. 유인 로봇들(132)은 일부 실시예들에서 인간 감독 하에서 러닝되어야 한다.

[0043] 무인 로봇(134)은 가상 환경들, 컨테이너들, 또는 운영 체제 세션들에서 무인으로 러닝될 수 있고, 많은 프로세스들을 자동화할 수 있다. 무인 로봇들(134)은 예를 들어 원격 실행, 모니터링, 스케줄링, 및 작업 큐(work queue)들에 대한 지원을 제공하는 것을 담당할 수 있다. 모든 로봇 유형들에 대한 디버깅이 일부 실시예들에서 웹 브라우저(110) 및 웹 애플리케이션(120)에서 러닝될 수 있다. 유인 로봇들 및 무인 로봇들 둘 다는 메인프레임들, 웹 애플리케이션들, VM들, 컨테이너들, 엔터프라이즈 애플리케이션들(예를 들어, SAP®, SalesForce®, Oracle® 등에 의해 생산되는 것들), 및 컴퓨팅 시스템 애플리케이션들(예를 들어, 데스크탑 및 랩탑 애플리케이션들, 모바일 디바이스 애플리케이션들, 웨어러블 컴퓨터 애플리케이션들 등)을 포함하지만 이에 제한되는 것은 아닌 다양한 시스템들 및 애플리케이션들을 자동화할 수 있다.

[0044] 로봇들(130)은 웹 애플리케이션(120)을 통해 웹 브라우저(110)에 구축된 워크플로우들을 러닝하는 실행 에이전트들이다. 로봇(들)(130)의 일부 실시예들의 한 상업적 예는 UiPath Robots™이다. 일부 실시예들에서, 로봇(130)은 Microsoft Windows® Service Control Manager(SCM)로 관리되는 서비스를 디폴트로 설치한다. 결과로서, 그러한 로봇들(130)은 로컬 시스템 계정 하에 상호적 Windows® 세션들을 열 수 있고, Windows® 서비스의 권한(right)들을 가질 수 있다.

[0045] 일부 실시예들에서, 로봇들(130)은 사용자 모드로 설치될 수 있다. 그러한 로봇들(130)에 대해, 이는 로봇들(130)이 사용자[이 사용자 하에 주어진 로봇(130)이 설치됨]와 동일한 권한들을 갖는 것을 의미한다. 이 피처는 또한, 각각의 머신의 자신의 최대 잠재력에서의 전체 활용을 보장하는 HD(High Density) 로봇들에 이용가능할 수 있다. 일부 실시예들에서, 임의의 유형의 로봇(130)이 HD 환경에서 구성될 수 있다.

[0046] 실행자들은 Windows® 세션 하에서 주어진 일들을 러닝할 수 있다(즉, 실행자들이 워크플로우들을 실행할 수 있음). 실행자들은 모니터별(per-monitor) DPI(dots per inch) 설정들을 알고 있을 수 있다. 에이전트들은 이용가능한 일들을 시스템 트레이 윈도우에 디스플레이하는 WPF(Windows® Presentation Foundation) 애플리케이션들일 수 있다. 에이전트들은 서비스의 클라이언트일 수 있다. 에이전트들은 일들을 시작하거나 중지할 것을 요청할 수 있고 설정들을 변경할 수 있다. 명령 라인(command line)이 서비스의 클라이언트일 수 있다. 명령 라인은, 일들을 시작할 것을 요청할 수 있고 이들의 출력을 대기하는 콘솔 애플리케이션이다.

[0047] 로봇들(130)의 컴포넌트들을 위에서 해석된 바와 같이 분할되게 하는 것은 개발자들, 지원 사용자들, 및 컴퓨팅 시스템들이 각각의 컴포넌트가 실행하고 있는 것을 더 쉽게 러닝하고, 식별하고, 추적하는 것을 돕는다. 실행자 및 서비스에 대해 상이한 방화벽 규칙(firewall rule)들을 설정하는 것과 같이, 특정 거동들이 이 방식으로 컴포넌트별로 구성될 수 있다. 실행자는 일부 실시예들에서 모니터별 DPI 설정들을 항상 알고 있을 수 있다.

결과로서, 워크플로우들은 이들이 제작된 컴퓨팅 시스템의 구성에 관계없이, 어떤 DPI에서도 실행될 수 있다. 웹 브라우저(110) 및 웹 애플리케이션(120)으로부터의 프로젝트들이 또한 일부 실시예들에서 브라우저 줌 레벨과 독립적일 수 있다. DPI를 알고 있지 않거나 알고 있지 않는 것으로서 의도적으로 마킹된 애플리케이션들에 대해, 일부 실시예들에서 DPI가 디스에이블될 수 있다.

[0048] 일부 실시예들에서, 지휘자(150)는 프로비저닝, 배치, 구성, 큐잉(queueing), 모니터링, 로깅(logging), 및/또는 상호연결성을 제공하는 것을 포함하지만 이에 제한되는 것은 아닌 다양한 능력들을 가질 수 있다. 특정 실시예들에서, 지휘자(150)는 생산 서버(140)에 원격으로 배치될 수 있다. 프로비저닝은 (예를 들어, 지휘자 웹 애플리케이션을 통한) 로봇들(130)과 지휘자(150) 사이의 연결들의 제작 및 유지를 포함할 수 있다. 배치는 실행을 위한 할당된 로봇들(130)에의 패키지 버전들의 올바른 전달을 보증하는 것을 포함할 수 있다. 구성은 로봇 환경들 및 프로세스 구성들의 유지 및 전달을 포함할 수 있다. 큐잉은 큐들 및 큐 항목들의 관리를 제공하는 것을 포함할 수 있다. 모니터링은 로봇 식별 데이터의 추적을 유지하는 것 및 사용자 허가(user permission)들을 유지하는 것을 포함할 수 있다. 로깅은 데이터베이스(예를 들어, SQL 데이터베이스) 및/또는 다른 저장 메커니즘[예를 들어, 큰 데이터세트들을 저장하고 빠르게 질의(query)하는 능력을 제공하는 ElasticSearch®]에 로그들을 저장하고 인덱싱하는 것을 포함할 수 있다. 지휘자(150)는 서드파티 솔루션들 및/또는 애플리케이션들에 대한 통신의 중앙 지점으로서 역할함으로써 상호연결성을 제공할 수 있다. 지휘자(150)는 배치된 로봇들 및 배치된 로봇들이 동작하는 컴퓨팅 시스템들을 포함하는 인프라스트럭처를 관리할 수 있다.

[0049] 도 2는 본 발명의 실시예에 따른, 웹 기반 RPA 설계자 시스템 중 일부 또는 모두를 구현하도록 구성된 컴퓨팅 시스템(200)을 예시하는 아키텍처도이다. 일부 실시예들에서, 컴퓨팅 시스템(200)은 본원에서 묘사되고/묘사되거나 설명된 컴퓨팅 시스템들 중 하나 이상일 수 있다. 컴퓨팅 시스템(200)은 정보를 전달하기 위한 버스(205) 또는 다른 통신 메커니즘, 및 정보를 프로세싱하기 위해 버스(205)에 커플링된 프로세서(들)(210)를 포함한다. 프로세서(들)(210)는 CPU(Central Processing Unit), ASIC(Application Specific Integrated Circuit), FPGA(Field Programmable Gate Array), GPU(Graphics Processing Unit), 이들의 다중 인스턴스들, 및/또는 이들의 임의의 조합을 포함한 임의의 유형의 범용 또는 특수 목적 프로세서일 수 있다. 프로세서(들)(210)는 또한 다중 프로세싱 코어들을 가질 수 있고, 코어들 중 적어도 일부가 특정 기능들을 수행하도록 구성될 수 있다. 다중 병렬 프로세싱이 일부 실시예들에서 사용될 수 있다. 특정 실시예들에서, 프로세서(들)(210) 중 적어도 하나는 생물학적 뉴런들을 모방하는 프로세싱 요소들을 포함하는 뉴로모픽 회로(neuromorphic circuit)일 수 있다. 일부 실시예들에서, 뉴로모픽 회로들은 폰 노이만 컴퓨팅 아키텍처의 전형적인 컴포넌트들을 요하지 않을 수 있다.

[0050] 컴퓨팅 시스템(200)은 프로세서(들)(210)에 의해 실행될 명령어들 및 정보를 저장하기 위한 메모리(215)를 더 포함한다. 메모리(215)는 RAM(Random Access Memory), ROM(Read Only Memory), 플래시 메모리, 캐시, 자기 또는 광학 디스크와 같은 정적 스토리지의 임의의 조합, 또는 임의의 다른 유형들의 비일시적 컴퓨터 판독가능 매체 또는 이들의 조합들을 포함할 수 있다. 비일시적 컴퓨터 판독가능 매체는 프로세서(들)(210)에 의해 액세스될 수 있는 임의의 이용가능한 매체일 수 있고 휘발성 매체, 비휘발성 매체, 또는 둘 다를 포함할 수 있다. 매체는 또한 제거가능하거나, 비제거가능하거나, 또는 둘 다일 수 있다.

[0051] 추가적으로, 컴퓨팅 시스템(200)은 무선 및/또는 유선 연결을 통한 통신 네트워크에의 액세스를 제공하기 위한 송수신기(transceiver)와 같은 통신 디바이스(220)를 포함한다. 일부 실시예들에서, 통신 디바이스(220)는 FDMA(Frequency Division Multiple Access), SC-FDMA(Single Carrier FDMA), TDMA(Time Division Multiple Access), CDMA(Code Division Multiple Access), OFDM(Orthogonal Frequency Division Multiplexing), OFDMA(Orthogonal Frequency Division Multiple Access), GSM(Global System for Mobile) 통신들, GPRS(General Packet Radio Service), UMTS(Universal Mobile Telecommunications System), cdma2000, W-CDMA(Wideband CDMA), HSDPA(High-Speed Downlink Packet Access), HSUPA(High-Speed Uplink Packet Access), HSPA(High-Speed Packet Access), LTE(Long Term Evolution), LTE-A(LTE Advanced), 802.11x, Wi-Fi, Zigbee, UWB(Ultra-WideBand), 802.16x, 802.15, HnB(Home Node-B), Bluetooth, RFID(Radio Frequency Identification), IrDA(Infrared Data Association), NFC(Near-Field Communications), 5G(fifth generation), NR(New Radio), 이들의 임의의 조합, 및/또는 본 발명의 범위로부터 벗어나지 않는 다른 현재 존재하거나 미래에 구현될 통신 기준 및/또는 프로토콜을 사용하도록 구성될 수 있다. 일부 실시예들에서, 통신 디바이스(220)는 단일형, 어레이형, 위상형, 스위치형, 빔포밍, 빔스티어링, 이들의 조합, 및 또는 본 발명의

범위로부터 벗어나지 않는 임의의 다른 안테나 구성인 하나 이상의 안테나를 포함할 수 있다.

[0052] 프로세서(들)(210)는 또한, 플라즈마 디스플레이, LCD(Liquid Crystal Display), LED(Light Emitting Diode) 디스플레이, FED(Field Emission Display), OLED(Organic Light Emitting Diode) 디스플레이, 플렉서블 OLED 디스플레이, 플렉서블 기판 디스플레이, 프로젝션 디스플레이, 4K 디스플레이, 고화질 디스플레이, Retina® 디스플레이, IPS(In-Plane Switching) 디스플레이, 또는 사용자에게 정보를 디스플레이하기 위한 임의의 다른 적합한 디스플레이와 같은 디스플레이(225)에 버스(205)를 통해 커플링된다. 디스플레이(225)는, 저항성, 용량성, SAW(surface-acoustic wave) 용량성, 적외선, 광학 이미징, 분산적 신호 기술(dispersive signal technology), 음향 펄스 인식(acoustic pulse recognition), 좌절형 내부 전반사(frustrated total internal reflection) 등을 사용하는 터치(햅틱) 디스플레이, 3D(three-dimensional) 터치 디스플레이, 다중 입력 터치 디스플레이, 다중 터치 디스플레이 등으로서 구성될 수 있다. 본 발명의 범위로부터 벗어나지 않는 임의의 적합한 디스플레이 디바이스 및 햅틱 I/O가 사용될 수 있다.

[0053] 컴퓨터 마우스, 터치패드 등과 같은 커서 제어 디바이스(235) 및 키보드(230)가 또한, 사용자가 컴퓨팅 시스템(200)과 인터페이싱하는 것을 가능하게 하기 위해 버스(205)에 커플링된다. 그러나, 특정 실시예들에서, 물리적 키보드 및 마우스가 존재하지 않을 수 있고, 사용자는 오로지 디스플레이(225) 및/또는 터치패드(미도시)를 통해 디바이스와 상호작용할 수 있다. 임의의 유형 및 조합의 입력 디바이스들이 설계 선택의 사안으로서 사용될 수 있다. 특정 실시예들에서, 물리적 입력 디바이스 및/또는 디스플레이가 존재하지 않는다. 예를 들어, 사용자가 통신 중인 다른 컴퓨팅 시스템을 통해 원격으로 컴퓨팅 시스템(200)과 상호작용할 수 있거나, 또는 컴퓨팅 시스템(200)이 자율적으로(autonomously) 동작할 수 있다.

[0054] 메모리(215)는, 프로세서(들)(210)에 의해 실행될 때 기능을 제공하는 소프트웨어 모듈들을 저장한다. 모듈들은 컴퓨팅 시스템(200)용 운영 체제(240)를 포함한다. 모듈들은 본원에서 설명된 프로세스들 또는 이들의 파생물들 중 일부 또는 모두를 수행하도록 구성된 웹 기반 RPA 모듈(245)을 더 포함한다. 컴퓨팅 시스템(200)은 추가 기능을 포함하는 하나 이상의 추가 기능적 모듈(250)을 포함할 수 있다.

[0055] 당업자는, "시스템"이 서버, 임베디드 컴퓨팅 시스템, 개인용 컴퓨터, 콘솔, PDA(personal digital assistant), 셀 폰, 태블릿 컴퓨팅 디바이스, 양자 컴퓨팅 시스템, 또는 임의의 다른 적합한 컴퓨팅 디바이스, 또는 본 발명의 범위로부터 벗어나지 않는 디바이스들의 조합으로서 구현될 수 있다는 점을 인식할 것이다. 위에서 설명된 기능들을 "시스템"에 의해 수행되는 것으로서 제시하는 것은 임의의 방식으로 본 발명의 범위를 제한하는 것으로 의도되는 것은 아니며, 본 발명의 많은 실시예들의 일례를 제공하는 것으로 의도된다. 실제로, 본원에서 개시된 방법들, 시스템들, 및 장치들은 클라우드 컴퓨팅 시스템들을 포함한 컴퓨팅 기술과 일치하는 로컬화되고 분산된 형태들로 구현될 수 있다. 컴퓨팅 시스템은 LAN(local area network), 모바일 통신 네트워크, 위성 통신 네트워크, 인터넷, 공개 또는 개인 클라우드, 하이브리드 클라우드, 서버 팜(server farm), 이들의 임의의 조합 등의 일부일 수 있거나 또는 그렇지 않으면 이들에 의해 액세스가능할 수 있다. 본 발명의 범위로부터 벗어나지 않는 임의의 로컬화되거나 분산된 아키텍처가 사용될 수 있다.

[0056] 본 명세서에서 설명된 시스템 피처들 중 일부가 이들의 구현 독립성을 더 구체적으로 강조하기 위해 모듈들로서 제시되었다는 점에 유념해야 한다. 예를 들어, 모듈은 커스텀 VLSI(very large scale integration) 회로들 또는 게이트 어레이들, 로직 칩들, 트랜지스터들과 같은 기성 반도체(off-the-shelf semiconductor)들, 또는 다른 개별 컴포넌트들을 포함하는 하드웨어 회로로서 구현될 수 있다. 모듈은 또한, 필드 프로그래밍가능 게이트 어레이들, 프로그래밍가능 어레이 로직, 프로그래밍가능 로직 디바이스들, 그래픽스 프로세싱 유닛들 등과 같은 프로그래밍가능 하드웨어 디바이스들에 구현될 수 있다.

[0057] 모듈은 또한 다양한 유형들의 프로세서들에 의한 실행을 위해 소프트웨어에 적어도 부분적으로 구현될 수 있다. 예를 들어, 실행파일 코드(executable code)의 식별된 유닛이, 예를 들어 객체, 프로시저, 또는 기능으로서 체계화(organize)될 수 있는 컴퓨터 명령어들의 하나 이상의 물리적 또는 논리적 블록을 포함할 수 있다. 그럼에도 불구하고, 식별된 모듈들의 실행파일들이 물리적으로 함께 위치될 필요는 없지만, 논리적으로 함께 결합될 때 모듈을 포함하고 모듈에 대한 명시된 목적을 달성하는 상이한 위치들에 저장된 이종 명령어(disparate instruction)들을 포함할 수 있다. 또한, 모듈들은, 예를 들어 하드 디스크 드라이브, 플래시 디바이스, RAM, 테이프, 및/또는 본 발명의 범위로부터 벗어나지 않는 데이터를 저장하는데 사용되는 임의의 다른 그러한 비일시적 컴퓨터 판독가능 매체일 수 있는 컴퓨터 판독가능 매체 상에 저장될 수 있다.

[0058] 실제로, 실행파일 코드의 모듈이 단일 명령어, 또는 많은 명령어들일 수 있고, 여러 상이한 코드 세그먼트들에 걸쳐, 상이한 프로그램들 사이에, 그리고 여러 메모리 디바이스들에 걸쳐 분산될 수도 있다. 유사하게, 동작

데이터가 모듈들 내에서 식별될 수 있고 본원에 예시될 수 있으며, 임의의 적합한 형태로 구현될 수 있고 임의의 적합한 유형의 데이터 구조 내에 체계화될 수 있다. 동작 데이터는 단일 데이터 세트로서 수집될 수 있거나, 또는 상이한 스토리지 디바이스들에 걸치는 것을 포함하여 상이한 위치들에 걸쳐 분산될 수 있으며, 시스템 또는 네트워크 상에 단지 전자 신호들로서 적어도 부분적으로 존재할 수 있다.

[0059] 도 3a는 본 발명의 실시예에 따른, 웹 기반 RPA 시스템(300)을 예시하는 아키텍처도이다. 시스템(300)은 데스크탑 컴퓨터(302), 태블릿(304), 및 스마트 폰(306)과 같은 사용자 컴퓨팅 시스템들을 포함한다. 그러나, 스마트 워치들, 랩탑 컴퓨터들, IoT(Internet-of-Things) 디바이스들, 차량 컴퓨팅 시스템들 등을 포함하지만 이에 제한되는 것은 아닌 본 발명의 범위로부터 벗어나지 않는 임의의 원하는 컴퓨팅 시스템이 사용될 수 있다.

[0060] 각각의 컴퓨팅 시스템(302, 304, 306)은 내부에 설치된 웹 브라우저(310)를 갖는다. 웹 브라우저(310)는 마우스 클릭들을 수행하는 것, 텍스트 필드들에 텍스트를 입력하는 것, 스크린샷들을 제공하는 것, 웹 페이지의 텍스트 필드들 또는 다른 컴포넌트들에 나타나는 정보를 제공하는 것 등과 같은 UI 자동화의 로컬 부분을 실행하는 확장부를 포함할 수 있다. 그러한 확장부는 예를 들어 유인 자동화들에 유용할 수 있다. 특정 실시예들에서, 컴퓨팅 시스템들(302, 304, 306)은 UI 객체 저장소(object repository)로부터 PowerPoint®, Outlook® 등에 대한 요소들을 리트리브(retrieve)하는 웹 기반 RPA 설계자 시스템을 사용하여 개발된 RPA 로봇들을 러닝할 수 있다. 예를 들어, 미국 특허 출원 제 16/922,289호를 보라. 타겟 애플리케이션에 대한 요소들[예를 들어, 애플리케이션들, 스크린들, 및 UI 요소들(애플리케이션이 스크린들을 그룹화하고 스크린들이 UI 요소들을 그룹화함)과 같은 객체 저장소 내의 항목들]이 (예를 들어, 요소들을 액티비티들로 드래그 앤 드롭함으로써) 워크플로우에 추가될 수 있다. 웹 브라우저(310)는 본 발명의 범위로부터 벗어나지 않는 원하는 유형의 웹 브라우저일 수 있다.

[0061] UI 객체 저장소는 애플리케이션들, 애플리케이션 버전들, 애플리케이션 스크린들, UI 요소들의 컬렉션들, 이들의 조합 등에 의해 그룹화될 수 있는 UI 객체들의 UI 객체 라이브러리들을 포함할 수 있다. 일부 실시예들에서의 UI 객체 라이브러리들의 UI 객체 저장소는 프로젝트에서 UI 서술자(descriptor)들의 신뢰도(reliability)를 관리하고, 재사용하고, 증가시키는 것을 용이하게 할 수 있다. UI 객체들을 재사용가능하게 하기 위해, 이들이 RPA 프로세스들에 의해 참조될 수 있는 UI 객체 라이브러리들로 추출될 수 있다. 선택자(selector)들 또는 다른 UI 서술자들이 예를 들어 새로운 버전의 애플리케이션으로 인해 수정될 때, 라이브러리가 수정된 UI 서술자들을 포함하도록 재제작(또는 재게재)될 수 있다. UI 객체 라이브러리를 사용하는 RPA 프로세스들이 이어서 수정된 UI 서술자 버전들을 호출할 수 있다.

[0062] UI 객체 라이브러리로부터의 UI 서술자들이 RPA 워크플로우 액티비티에 직접적으로 추가될 수 있어서, 그렇지 않으면 액티비티에 대한 커스텀 선택기를 제작하는데 요구될 수 있는 개발자 시간을 절약한다. 객체 브라우저는 제작된 선택기들을 UI 서술자들의 재사용을 가능하게 하기 위해 객체 라이브러리들 내에 저장하는 데이터베이스를 제공할 수 있다. 객체 라이브러리는 특정 버전의 애플리케이션으로부터의 하나 이상의 스크린에 대응하는 UI 서술자들의 컬렉션으로서 본원에서 정의된다. UI 서술자는 UI 요소를 찾기 위한 명령어들의 세트이다.

[0063] 웹 브라우저(310)는 웹 애플리케이션들을 러닝하고 이 웹 애플리케이션들에 관한 정보를 클라우드 환경 내의 서버(330)에 네트워크(320)[예를 들어, LAN(local area network), 모바일 통신 네트워크, 위성 통신 네트워크, 인터넷, 이들의 임의의 조합 등]를 통해 제공한다. 서버(330)는, 확장부들을 통해 웹 브라우저들(310)과 상호작용하고/상호작용하거나 웹 브라우저들(310)로부터 분리된 자체 프로세스(들)를 구현하는 자동화들(334) 및 RPA 로봇들(336)을 포함하는 호스팅 환경(332)(예를 들어, 컨테이너, VM, 또는 운영 체제 세션)을 러닝한다. 도 3b를 보라. 예를 들어, 자동화들(334) 및 RPA 로봇들(336)은 VM 또는 컨테이너 내에서 러닝될 수 있고 일부 실시예들에서 헤드리스일 수 있다. 서버(330)는 일부 실시예들에서 데이터베이스(340)에 자동화들(334)[예를 들어, 상이한 버전들의 자동화들(334), 객체 저장소 등]에 대한 데이터를 저장할 수 있고 이들을 로딩할 수 있거나 그렇지 않으면 이들을 RPA 로봇들(336)에 의한 실행을 위해 이용가능하게 할 수 있다.

[0064] RPA 로봇들(336)은, RPA 로봇들(336)이 서버(330)와 같은 주어진 컴퓨팅 시스템 상에 설치되고 러닝되는 머신측 서비스(machine-side service)들이라는 점에서 자동화들(334)과 상이하다. RPA 로봇들(336)은 지휘자 애플리케이션(예를 들어, UiPath Orchestrator™), RPA 설계자 애플리케이션 등으로부터의 요청들을 대기할 수 있고 이어서 요청의 수신 후 요청된 자동화를 러닝할 수 있다. 각자의 RPA 로봇(336)이 머신 판독가능 코드 또는 스크립트의 형태로 자동화(334)의 정의를 수신하고 이를 러닝한다. 자동화들(334)은, 다른 한편으로, 예를 들어 패키지의 형태에 있는 워크플로우들의 실시예들이다. 패키지들은 일부 실시예들에서 RPA 로봇들(336)에 의해 사

용되고 러닝된다.

[0065] 자동화들(334)은 컴퓨팅 시스템(350) 상에서 러닝되는 웹 기반 RPA 설계자 애플리케이션(352)을 사용하여 RPA 개발자에 의해 설계될 수 있다. 그러나, 일부 실시예들에서, 웹 기반 RPA 설계자 애플리케이션이 컴퓨팅 시스템(350)에 원격에 있고 웹 브라우저를 통해 액세스된다. RPA 개발자는 프로젝트들을 제작하거나 선택할 수 있고, RPA 워크플로우들을 제작하거나 수정할 수 있으며, RPA 워크플로우 액티비티들을 추가/수정/제거할 수 있는 등을 할 수 있다. 일부 실시예들에서, 웹 기반 RPA 설계자 애플리케이션(352)은 RPA 개발자가 UI 객체 저장소로부터 요소들을 선택하는 것 및 이 요소들을 각자의 UI 서술자들을 통해 RPA 워크플로우들의 액티비티들에 사용하는 것을 가능하게 한다. RPA 개발자는 또한 자동화(334)가 서버(330) 상에 생성되게 할 수 있고, 테스트를 수행할 수 있으며, 최종 사용자들에 의해 사용될 준비가 되면, 서버(330)에 의한 풀의 일부로서 사용되고, 컨테이너들, VM들, 또는 운영 체제 세션들(332) 등에 배치되는 자동화들(334)을 배치할 수 있다. 일부 실시예들에서, 자동화들은 컴퓨팅 시스템(350)의 웹 브라우저에 로컬로 생성되고, 원격으로 러닝되기 보다는 그곳에서 테스트되고 검증된다.

[0066] 도 4a는 본 발명의 실시예에 따른, 웹 기반 RPA 개발 애플리케이션 웹페이지(400)의 홈 페이지 뷰를 예시하는 스크린샷이다. 다양한 샘플 프로젝트들이 이들이 제작되면 사용자에게 디스플레이될 수 있다. RPA 개발자는 새로운 자동화 버튼(410)을 클릭함으로써 새로운 자동화를 제작할 수 있다. 일부 실시예들에서, 프로세스, 라이브러리 프로젝트, 테스트 자동화 프로젝트, 트리거 기반 프로세스 등을 포함할 수 있지만 이에 제한되는 것은 아닌 프로젝트 유형들의 세트와 같은 다른 옵션들이 제시될 수 있다. 사용자는 또한 이미 제작된 프로젝트들을 다시 열고, 편집하고, 재게재할 수 있다.

[0067] RPA 개발자가 새로운 자동화 버튼(410)을 클릭했을 때, 새로운 자동화 윈도우(420)가 나타난다. 도 4b를 보라. 사용자는 자동화에 대한 명칭 및 설명을 입력할 수 있고 제작 버튼을 클릭함으로써 새로운 프로젝트를 제작할 수 있다. 그러나, 개발자(들), 타겟 회사 등과 같은, 본 발명의 범위로부터 벗어나지 않는 임의의 다른 원하는 정보가 제공될 수 있다.

[0068] RPA 개발자가 새로운 자동화 프로젝트를 제작한 후, 자동화가 설계될 수 있는 캔버스(430)가 나타난다. 도 4c를 보라. 옵션 지정 시작(432)이 사용자가 자동화가 시작할 방법을 표시하는 것을 가능하게 한다. 여기서, 자동화는 수동으로 시작하도록 구성된다. 그러나, 사용자는 트리거(예를 들어, 사용자가 웹페이지 상의 버튼을 클릭하는 것, 특정 웹페이지가 열리는 것, 웹 양식이 제출되는 것 등)에 기초하는 것, 원하는 시작 시간에 기초하는 것, 원하는 주기적 런 타임에 기초하는 것(예를 들어, 매월 1일 자정에) 등과 같이 시작 조건을 다른 옵션으로 변경할 수 있다. 비어 있는 액티비티(empty activity)(434)를 클릭함으로써 사용자에 의해 액티비티들이 추가될 수 있다.

[0069] 사용자가 비어 있는 액티비티(434)를 클릭했을 때, 액티비티 카테고리들 및 빠른 액션 윈도우(440)가 나타난다. 도 4d를 보라. 사용자가 카테고리들을 서치할 수 있거나 목록화된 카테고리들로부터 선택할 수 있다. 예를 들어, 사용자가 Excel®을 클릭하면, Excel® 액티비티 윈도우(450)가 Excel® 액티비티들에 대한 옵션들과 함께 나타난다. 도 4e를 보라.

[0070] 사용자가 Excel® 파일 사용을 선택하면, 사용자가 지금 연결 버튼(461)을 클릭한 후 연결 정보를 제공하면 구성될 수 있는 Excel® 파일 액티비티 사용(460)이 제작된다. 특정 애플리케이션들에 대해, 웹 기반 RPA 설계자 애플리케이션, 로봇 트레이 프로세스(예를 들어, UiPath Assistant™), 또는 애플리케이션들 및 다른 프로세스들에 액세스하기 위한 다른 서비스들에 허가들이 부여될 필요가 있다. 자격증명(credential)들이 저장될 수 있고 향후 자동화들을 위해 재사용될 수 있다. RPA 로봇이 자동화를 실행했을 때, RPA 로봇은 자동화가 필요로 하는 연결들을 살필(look at) 수 있고 이 자격증명들 하에서 자동화가 러닝될 것이다. RPA 로봇은 기저 워크플로우 논리(underlying workflow logic)를 변경하지 않고 이 자격증명들에 액세스하도록 구성될 수 있고, 자격증명들을 찾을 수 없거나 자격증명들이 적절히 작동하지 않으면 사용자에게 자격증명들을 입력할 것을 요청할 수 있다. 파일들이 클라우드에서 호스팅될 수 있고, 원격 데이터베이스 등에 저장될 수 있다. 액티비티의 종료가 Excel® 파일 사용의 종료 표시자(462)에 의해 지정된다. 도 4f를 보라.

[0071] 다른 액티비티들, 조건부 논리(conditional logic) 등이 비어 있는 액티비티(464)를 클릭함으로써 Excel® 파일 액티비티 사용(460) 내에 추가되고 구성될 수 있다. 이는 도 4d의 윈도우(440)와 동일하거나 유사한 윈도우를 띄울(bring up) 수 있다. 테스트 자동화를 러닝하기 위한 옵션(436) 및 UI 자동화 프로젝트를 공유하기 위한

옵션(438)이 또한 나타난다. 이는 사용자가 워크플로우를 이들이 제작되었을 때 검증하는 것 뿐만 아니라 이들의 작업을 다른이들과 공유하는 것을 가능하게 할 수 있다.

[0072] 사용자가 액티비티를 구성할 때, 다양한 옵션들이 나타난다. 예를 들어, 사용자가 지금 연결 버튼(461)을 클릭하고 원격 파일 및/또는 애플리케이션에 액세스하기 위한 로그인 정보를 제공하면, 우측에 옵션 창(470)에 옵션들이 나타난다. 도 4g를 보라. 이 경우, 사용자가 파일 선택 메뉴(471)로부터 Excel® 파일 "Apollo components list.xlsx"를 선택한다. 파일 명칭이 파일 명칭 리스트(472)에 나타난다. 사용자는 테스트 액티비티 메뉴(473)를 사용하여 액티비티의 구성을 테스트할 수 있을 뿐만 아니라, 주어진 액티비티에 특유한 특성들의 그룹들인 일반 옵션들(474) 및 기본 옵션들(475)을 구성할 수 있다. 더 구체적인 옵션들이 또한 구성될 수 있다. 예를 들어, 사용자는 각각의 체크박스(476)에 반복을 선택할 수 있고, 이는 사용자가 액티비티가 반복될 스프레드시트에서 요소를 선택하는 것을 가능하게 한다. 여기서, 사용자는 옵션(477)으로 "Row"를 옵션(478)으로 스프레드시트의 "Sheet 1"을 선택했다.

[0073] 일부 실시예들에서, 사용자들은 거버넌스 규칙(governance rule)들에 따라, 분석 에러들이 있는 자동화들을 게재하거나 러닝하는 것이 허용될 수 있거나 허용되지 않을 수 있다. 검증 에러들이 일부 실시예들에서 자동화들을 러닝하고 게재하는 것에 대한 전제조건(prerequisite)들이며, 모든 사용자들에 대해 허용될 수 있다. 그러나, 정적 코드 분석 에러들이 허용되지 않을 수 있고, 그와 같이 식별될 수 있다.

[0074] 도 4a 내지 도 4g에 도시된 것과 같은, 웹 기반 RPA 설계자로 개발된 자동화들을 테스트하기 위해, 개발자의 컴퓨팅 시스템이 아닌 서버측에서(예를 들어, 클라우드에서) 실행이 일어날 수 있다. 자동화를 설계하고 이를 실행하는 것 외에 추가 구성이 RPA 개발자에 의해 필요되지 않을 수 있다. 예를 들어 UiPath Orchestrator®와 같은 지휘자 애플리케이션에서 RPA 개발자에 의해 구성될 수 있거나 구성되지 않을 수 있는 원격 VM 상에서 실행이 일어날 수 있다. RPA 개발자는 웹 기반 RPA 설계자의 웹 브라우저 부분을 통해 성공적인 실행을 통지받을 수 있거나 실행 에러들 및/또는 거버넌스 규칙들의 위반들을 제공받을 수 있다. 특정 실시예들에서, RPA 워크플로우들은 자동화가 승인되고 생산 사용(즉, 런타임 사용)을 위해 게재되기 전에 적절한 허가들을 갖는 개인들에 의해 리뷰될 수 있다.

[0075] 도 4a 내지 도 4g가 예시에 의해서만 제공되며, 다양한 다른 액티비티들 및 조건들이 본 발명의 범위로부터 벗어나지 않고 구성될 수 있다는 점에 유념해야 한다. 예를 들어, RPA 개발자는 상이한 조건들이 존재하면 Excel® 파일에 대해 상이한 액션들을 수행하는 것을 선택할 수 있다(예를 들어, 컴포넌트에 대한 기한이 경과되었으면 경과되었음에 관하여 이메일을 전송하는 것 대(versus) 새로운 완성된 컴포넌트와 함께 이메일을 전송하는 것 및 컴포넌트가 기일이 지나지 않았으면 Excel®에서 상태를 업데이트하는 셀에 쓰기, 이메일들 및/또는 다른 이메일이 아닌 애플리케이션들의 메시지들을 전송하는 것 등). 자동화들은, 웹페이지와 상호작용할 수 있고, 웹페이지로부터 정보를 획득할 수 있으며, 사용자의 웹페이지와의 상호작용, 서버측 상에서 또는 사용자 컴퓨팅 시스템들 상에서 로컬로 다른 RPA 로봇들과 상호작용하는 것, 데이터베이스, 또는 본 발명의 범위로부터 벗어나지 않는 임의의 다른 RPA 프로세스로부터 정보를 리트리브하는 것을 포함할 수 있는 서버측 상에서 원하는 RPA 로봇 액션들을 수행할 수 있는 일부 실시예들의 웹 기반 RPA 설계자를 사용하여 설계될 수 있다. 위에 따라, 특정 실시예들에서, 자동화는 VM 또는 컨테이너를 통해 서버측 상에서 부분적으로 또는 전체적으로 수행될 수 있다.

[0076] 도 5는 본 발명의 실시예에 따른, 설계 시간에 웹 기반 RPA 개발을 수행하기 위한 프로세스를 예시하는 흐름도(500)이다. 510에서 프로세스는 웹 기반 설계자 애플리케이션의 웹 인터페이스에 액세스하고 이를 웹 브라우저 상에 디스플레이하는 것으로 시작한다. 웹 인터페이스는, 사용자가 PRA 프로젝트들을 제작하는 것, RPA 워크플로우들을 구성하는 것, 원격으로 자동화를 생성하고, 실행하고, 검증하기 위해 RPA 워크플로우 구성들을 서버측 웹 기반 RPA 설계자 애플리케이션에 제출하는 것을 가능하게 한다. 520에서 웹 인터페이스를 통해 웹 기반 설계자 애플리케이션에 의해 RPA 워크플로우가 설계되고 구성된다.

[0077] 530에서 사용자가 RPA 워크플로우에 만족한 후, 또는 사용자가 달리 테스트 및 검증을 수행하기를 원할 때, 사용자는 웹 인터페이스를 통해, 서버측 웹 기반 RPA 설계자 애플리케이션이 워크플로우를 구현하는 자동화를 생성하게 한다. 이어서 540에서 서버측 웹 기반 RPA 설계자 애플리케이션이 RPA 로봇을 통해 자동화를 실행하고 자동화를 검증한다. 550에서 검증이 실패하면, 520에서 사용자는 RPA 워크플로우를 설계하고 구성하는 것을 계속할 수 있다. 검증이 성공하면, 560에서 RPA 워크플로우/프로젝트가 생산(런타임) 사용을 위해 게재될 수 있다. 예를 들어, 이는 자동화 코드를 생성하는 것 및 자동화 코드를 VM, 컨테이너, 또는 운영 체제 세션에서

RPA 로봇을 통해 러닝하는 것을 포함할 수 있다. 일부 실시예들에서, 프로세스가 이어서 도 6, 도 7, 또는 도 9의 단계들로 진행될 수 있다.

[0078] 도 6은 본 발명의 실시예에 따른, 웹 확장부와 상호작용하는 런타임에 자동화를 실행하기 위한 프로세스(600)를 예시하는 흐름도이다. 610에서 프로세스는 RPA 로봇을 통해 서버측 자동화를 실행하는 것으로 시작한다. 서버측 자동화가 스케줄링될 수 있고, 수동으로 실행되게 될 수 있으며, 트리거 등에 기초하여 실행될 수 있다. 트리거는 애플리케이션 이벤트, 발생하는 웹페이지와의 사용자 상호작용, 특정 시간 경과 등일 수 있다. 620에서 RPA 로봇이 이어서 웹 브라우저와 상호작용하기 위해, 웹 브라우저의 하나 이상의 탭에서 하나 이상의 웹페이지로부터 정보를 획득하기 위해 또는 둘 다를 위해 (예를 들어, 클라이언트측 컴퓨팅 시스템 상에서, VM에서, 컨테이너에서 등) 웹 브라우저의 웹 확장부에 하나 이상의 명령을 전송한다. 630에서 웹 확장부가 이어서 RPA 로봇에 의해 요청된 대로, 웹 브라우저와 상호작용하거나, 웹 브라우저의 하나 이상의 탭에서 하나 이상의 웹페이지로부터 정보를 획득하거나, 또는 둘 다를 한다. 일부 실시예들에서, 웹 확장부는, 예를 들어 자동화가 컨테이너에서 러닝될 때 그리고 이 확장부로 정의된 결과로서 헤드리스로(headlessly) 실행될 수 있는 네이티브 기능(native functionality)을 제공한다. 그러한 기능은 데스크탑/랩탑 컴퓨터들, 신 클라이언트(thin client)들, 식 클라이언트(thick client)들 등에 대한 일부 실시예들에서 구현될 수 있다. 로컬 컴퓨팅 시스템으로부터 정보가 획득될 수 있고/있거나 브라우저 상호작용이 로컬로 수행될 수 있다.

[0079] 640에서 이어서 웹 확장부가 요청된 상호작용이 발생했다는 확인을 제공하거나, 요청된 정보를 제공하거나, 또는 RPA 로봇이 수신한 문제가 있으면 에러 메시지를 제공한다. 일부 실시예들에서, 이는 자동화가 달성하도록 설계된 작업일 수 있다. 650에서 그러나, 자동화가 데이터베이스 조회(database lookup)를 수행하는 것, 이메일을 전송하는 것, 파일에 기록하고 이를 저장하는 것, 웹 확장부로부터 추가 액션들 및/또는 정보를 요청하는 것 등과 같은 웹 확장부로부터의 정보에 기초하여 일부 추가 프로세싱 또는 다른 액션들을 수행하도록 설계되었으면, RPA 로봇이 이 액션들을 수행할 수 있다. 프로세스는 또한 하나 이상의 다른 명령에 대해 단계(620)로 리턴할 수 있다.

[0080] 도 7은 본 발명의 실시예에 따른, VM 또는 컨테이너에서 러닝되는 런타임에 자동화를 실행하기 위한 프로세스(700)를 예시하는 흐름도이다. 710에서 프로세스는 각자의 자동화들을 구현하는 VM들 및/또는 컨테이너들의 풀을 제작하는 것으로 시작한다. 풀은 예를 들어 탄력적으로 프로비저닝될 수 있거나 수동으로 프로비저닝될 수 있다. 720에서 이어서 서버측 자동화가 RPA 로봇에 의해 VM, 컨테이너, 또는 운영 체제 세션에서 실행된다. 서버측 자동화가 스케줄링될 수 있고, 수동으로 실행되게 될 수 있으며, 트리거 등에 기초하여 실행될 수 있다. 730에서 이어서 PRA 로봇이 자동화에 설계된 액션들을 수행한다.

[0081] 일부 실시예들에서, 웹 기반 설계자 시스템이 로컬 RPA 확장 프로세스를 통해 컴퓨팅 시스템 상에서 러닝되는 애플리케이션들의 로컬 인스턴스들과 상호작용하는 자동화들을 설계하는데 사용될 수 있다. 로컬 RPA 확장 프로세스는 컨테이너, VM, 또는 운영 체제 세션에서 원격으로 러닝되는 RPA 로봇들로부터 통신들을 수신할 수 있고, 로컬 애플리케이션들과 상호작용할 수 있으며, 요청하는 RPA 로봇들에 정보를 다시 전송할 수 있다. 도 8은 본 발명의 실시예에 따른, 로컬 애플리케이션들과 상호작용하도록 구성된 웹 기반 RPA 시스템(800)을 예시하는 아키텍처도이다. 도 3a 및 도 3b의 시스템(300)과 유사하게, 시스템(800)은 데스크탑 컴퓨터(802), 태블릿(804), 스마트 폰(806), 네트워크(820), 서버(830), 호스팅 환경(832), 데이터베이스(840), 개발자 컴퓨팅 시스템(850), 및 웹 기반 RPA 설계자 애플리케이션(852)을 포함한다. 또한 시스템(300)과 유사하게, 호스팅 환경(832)은 자동화들(834) 및 RPA 로봇들(836)을 포함한다.

[0082] 그러나, 시스템(800)에서, 데스크탑 컴퓨터(802), 태블릿(804), 및 스마트 폰(806)은, 데스크탑 컴퓨터(802), 태블릿(804), 및 스마트 폰(806) 상에서 러닝되는 애플리케이션들과 상호작용하도록 RPA 로봇들(836)의 기능을 확장시키는 로컬 RPA 확장 프로세스(810)를 러닝한다. 로컬 RPA 확장 프로세스(810)는 네트워크(820)를 통해 자동화(834)를 실행하는 RPA 로봇(836)으로부터 요청을 수신할 수 있다. 예를 들어, 로컬 RPA 확장 프로세스(810)는 통신들을 위해 포트 상에서 리스닝 중일 수 있다. 본 발명의 범위로부터 벗어나지 않는 임의의 적합한 통신 프로토콜[예를 들어, TCP(Transmission Control Protocol)/IP(Internet Protocol), FTP(File Transfer Protocol) 등]이 사용될 수 있다.

[0083] 로컬 RPA 확장 프로세스(810)는 RPA 로봇(836)으로부터의 요청을 파싱(parsing)할 수 있고 연관된 액션을 수행할 수 있고/있으며 요청된 정보를 리트리브할 수 있다. 예를 들어, RPA 로봇(836)으로부터의 요청은 용어 "인보이스"에 대한 사용자의 이메일을 서치하기 위해 Outlook®용 API를 사용하도록, 매칭하는 결과들을 리트리브하도록, 그리고 결과들을 RPA 로봇(836)에 다시 전송하도록 로컬 RPA 확장 프로세스(810)에 요청할 수 있다. 이

어서 RPA 로봇(836)은 자동화(834)의 논리를 실행하는 것을 계속하기 위해 이 정보를 사용할 수 있다.

[0084] 일부 실시예들에서, 로컬 RPA 확장 프로세스(810)는 IPC(Inter-Process Communication) 프로토콜들을 사용하여 각자의 컴퓨팅 시스템 상에서 러닝되는 다른 프로세스들과 통신할 수 있다. IPC 프로토콜들은, 운영 체제에서 러닝되는 프로세스들이 또한 상이한 세션들에서 잠재적으로 이 운영 체제 상에서 러닝되는 다른 프로세스들과 그에 의해 통신할 수 있는 메커니즘들이다. 이 프로토콜들은 네트워크(820), 파이프들, COM(Component Object Model), RPC(Remote Procedure Calls), 소켓들 등을 통한 통신을 촉진할 수 있다. IPC 프로토콜들은 러닝 중인 프로세스들 간에 정보, 요청들, 명령들 등을 전송하는데 사용될 수 있다.

[0085] 도 9는 본 발명의 실시예에 따른, RPA 로봇과는 상이한 컴퓨팅 시스템 상에서 러닝되는 로컬 RPA 확장 프로세스와 상호작용하는 런타임에 자동화를 실행하기 위한 프로세스(900)를 예시하는 흐름도이다. 910에서 프로세스는 RPA 로봇을 통해 서버측 자동화를 실행하는 것으로 시작한다. 서버측 자동화가 스케줄링될 수 있고, 수동으로 실행되게 될 수 있으며, 트리거 등에 기초하여 실행될 수 있다. 트리거는 애플리케이션 이벤트, 발생하는 웹페이지와의 사용자 상호작용, 특정 시간 경과 등일 수 있다. 920에서 이어서 RPA 로봇은, 로컬 RPA 확장 프로세스에 대해 로컬로 러닝되는 하나 이상의 애플리케이션 및/또는 프로세스(예를 들어, 운영 체제 프로세스, 애플리케이션들과 연관된 프로세스 등)와 상호작용하도록 원격 컴퓨팅 시스템 상에서(예를 들어, 클라이언트측 컴퓨팅 시스템 상에서, 원격 VM에서, 원격 컨테이너에서 등) 러닝되는 로컬 RPA 확장 프로세스에 하나 이상의 요청을 전송한다. 930에서 이어서 로컬 RPA 확장 프로세스는 애플리케이션(들)에서 원하는 동작들을 수행하기 위해, 컴퓨팅 시스템 상의 애플리케이션(들) 및/또는 다른 프로세스들로부터 정보를 획득하기 위해, 또는 둘 다를 위해 (예를 들어, API들을 통해) 하나 이상의 애플리케이션 및/또는 프로세스를 제어한다. 940에서 이어서 로컬 RPA 확장 애플리케이션이 애플리케이션(들)과의 요청된 상호작용(들)이 발생했다는 확인을 제공하거나, 요청된 정보를 제공하거나, 또는 둘 다를 한다. 일부 실시예들에서, RPA 로봇이 수신한 문제가 있으면 에러 메시지가 제공될 수 있다.

[0086] 일부 실시예들에서, 이는 자동화가 달성하도록 설계된 작업일 수 있다. 950에서 그러나, 자동화가 데이터베이스 조회를 수행하는 것, 이메일을 전송하는 것, 파일에 기록하고 이를 저장하는 것, 웹 확장부로부터 추가 액션들 및/또는 정보를 요청하는 것 등과 같은 웹 확장부로부터의 정보에 기초하여 일부 추가 프로세싱 또는 다른 액션들을 수행하도록 설계되었으면, RPA 로봇이 이 액션들을 수행할 수 있다. 프로세스는 또한 하나 이상의 다른 요청에 대해 단계(920)로 리턴할 수 있다.

[0087] 도 10은 본 발명의 실시예에 따른, RPA 로봇이 없는[즉, 로봇리스(robotless)] 웹 브라우저를 통해 설계 시간에 웹 기반 RPA 개발을 수행하기 위한 프로세스(1000)를 예시하는 흐름도이다. "로봇리스"는 본원에서 사용되는 바와 같이, RPA 로봇과 같은 기능이 웹 브라우저에 포함되어 있지만, RPA 로봇이 별도의 서비스로서 러닝되지 않는다는 것을 의미한다. 이와 같이, 웹 브라우저가 러닝되는 컴퓨팅 시스템 상에 로컬 클라이언트측 바이너리(binary)들을 설치할 필요가 없으며, 일부 실시예들에서 웹 브라우저만이 필요된다. 로봇과 같은 바이너리들이 탭 내에 포함될 수 있고 로드시 웹 서버로부터 서빙될 수 있다. HTML, JavaScript™ 등이 또한 이때 로딩될 수 있다.

[0088] 1010에서 프로세스는 웹 브라우저에 웹 기반 RPA 설계자 애플리케이션에 대한 코드를 다운로드하는 것으로 시작한다. 코드는 사용자가 자동화들을 제작하고 실행하는 것 둘 다를 가능하게 한다. 이 코드는 웹 브라우저에 대해 탭에 로딩되는 DDL과 같은 실행가능한 바이너리들을 포함할 수 있다. 이는, JavaScript™와 같은 다른 포맷들이 전형적으로 로딩되기 때문에 비전형적이다. 그러나, 본 발명의 범위로부터 벗어나지 않는 JavaScript™, Servlets™ 등과 같은 자동화 실행 기능을 포함하는 임의의 적합한 코드가 사용될 수 있다는 점에 유념해야 한다. 1020에서 웹 기반 설계자 애플리케이션에 대한 웹 인터페이스가 웹 브라우저에 디스플레이된다. 웹 인터페이스는 RPA 프로젝트들을 제작하고 RPA 워크플로우들을 구성하는 것 뿐만 아니라 그로부터 자동화들을 생성하고 실행하기 위한 기능을 제공하도록 구성된다. 1030에서 웹 인터페이스를 통해 웹 기반 설계자 애플리케이션에 의해 RPA 워크플로우가 설계되고 구성된다.

[0089] 사용자가 RPA 워크플로우에 만족한 후, 또는 사용자가 달리 테스트 및 검증을 수행하기 원할 때, 1040에서 웹 브라우저가 구성된 RPA 워크플로우에 대한 자동화를 생성한다. 이 자동화는 일반 브라우저 프로세스 대신 브라우저 작업자 프로세스로서 러닝될 수 있다. 1050에서 이어서 웹 브라우저가 생성된 자동화를 실행하고 검증한다. 1060에서 검증이 실패하면(예를 들어, 에러들이 있거나 거버넌스 규칙들이 위반됨), 1030에서 사용자는 RPA 워크플로우를 설계하고 구성하는 것을 계속할 수 있다. 검증이 성공하면, 1070에서 RPA 워크플로우/프로젝트가 생산(런타임) 사용을 위해 게재될 수 있다. 예를 들어, 이는 웹 브라우저에 대한 자동화 코드를 생성하는

것 및 웹 브라우저에 대한 자동화 코드를 VM, 컨테이너, 또는 운영 체제 세션에서 러닝하는 것을 포함할 수 있다. 일부 실시예들에서, 프로세스가 이어서 도 11의 단계들로 진행될 수 있다.

[0090] 자동화를 실행하고 디버깅하기 위한 메커니즘이 이 실시예에서 웹 브라우저 자체에 로딩된다. 이는 예를 들어, 원격 VM, 컨테이너, 또는 운영 체제 세션을 요하지 않는다. 환언하면, 사용자가 자동화를 러닝할 때, 코드가 원격 컨테이너, 원격 VM, 또는 원격 운영 체제 세션에서 실행되도록 이송되지 않는다. 대신, 자동화 설계 시간 및 런타임 기능이 웹 브라우저에, 가능하게는 일부 실시예들에서 동일한 탭에 있다. 자동화를 실행하기 위한 기능은, 웹 기반 RPA 설계자 애플리케이션에 대한 웹 페이지가 로딩될 때 웹 브라우저에 로딩될 수 있다. 이는 일부 실시예들에서 웹 브라우저 확장을 요하지 않는다. 따라서, 탭 자체가 기본적으로 설계 시간 RPA 설계자 및 런타임 RPA 로봇 둘 다가 되며, 사용자들이 자동화들을 제작하고, 실행하고, 디버깅할 수 있다. 일부 실시예들에서, 생산 RPA 로봇들에 의해 실행되기 보다는 생산 서버의 웹 브라우저에서 러닝되는 자동화들이 또한 설계될 수 있다.

[0091] 도 11은 본 발명의 실시예에 따른, 컨테이너, VM, 또는 운영 체제 세션의 웹 브라우저에서 런타임에 자동화를 실행하기 위한 프로세스(1100)를 예시하는 흐름도이다. 1110에서 프로세스는 서버의 컨테이너, VM, 또는 운영 세션에서 러닝되는 웹 브라우저를 통해 서버측 자동화를 실행하는 것으로 시작한다. 이 형태의 자동화만으로 신규하다. 그러나, 일부 실시예들에서, 각각 도 6 및 도 8의 프로세스들(600 및 800)과 유사하게, 1120에서 자동화가 원격 컴퓨팅 시스템 상에서 러닝되는 웹 확장 또는 로컬 RPA 확장 프로세스에 명령들을 전송한다. 이어서 1130에서 웹 확장 또는 로컬 RPA 확장 프로세스가 요청된 액션을 수행하고/수행하거나 요청된 정보를 획득하고 1140에서 서버의 컨테이너, VM, 또는 운영 세션에서 웹 브라우저에 확인 및/또는 정보를 제공한다. 이어서 1150에서 웹 브라우저가 자동화와 연관된 추가 액션들을 수행할 수 있다.

[0092] 도 5 내지 도 7 및 도 9 내지 도 11에서 수행된 프로세스 단계들은, 본 발명의 실시예들에 따라, 프로세서(들)가 도 5 내지 도 7 및 도 9 내지 도 11에서 설명된 프로세스(들)의 적어도 일부를 수행하도록 명령어들을 인코딩한, 컴퓨터 프로그램에 의해 수행될 수 있다. 컴퓨터 프로그램은 비일시적 컴퓨터 판독가능 매체 상에 구현될 수 있다. 컴퓨터 판독가능 매체는 하드 디스크 드라이브, 플래시 드라이브, RAM, 테이프, 및/또는 데이터를 저장하는데 사용되는 임의의 다른 그러한 매체 또는 매체의 조합일 수 있지만, 이에 제한되는 것은 아니다. 컴퓨터 프로그램은, 컴퓨터 판독가능 매체 상에 또한 저장될 수 있는, 도 5 내지 도 7 및 도 9 내지 도 11에서 설명된 프로세스 단계들 중 일부 또는 모두를 구현하도록 컴퓨팅 시스템의 프로세서(들)[예를 들어, 도 2의 컴퓨팅 시스템(200)의 프로세서(들)(210)]를 제어하기 위한 인코딩된 명령어들을 포함할 수 있다.

[0093] 컴퓨터 프로그램은 하드웨어에, 소프트웨어에, 또는 하이브리드 구현으로 구현될 수 있다. 컴퓨터 프로그램은, 서로 가동 통신하며, 디스플레이를 위해 정보 또는 명령어들을 패스(pass)하도록 설계된 모듈들로 구성될 수 있다. 컴퓨터 프로그램은 다목적 컴퓨터, ASIC, 또는 임의의 다른 적합한 디바이스 상에서 동작하도록 구성될 수 있다.

[0094] 본원의 도면들에서 일반적으로 설명되고 예시된 바와 같은, 본 발명의 다양한 실시예들의 컴포넌트들이 매우 다양한 상이한 구성들로 배열되고 설계될 수 있다는 점이 쉽게 이해될 것이다. 따라서, 첨부된 도면들에 나타내어진 바와 같은, 본 발명의 실시예들의 상세한 설명이 청구된 바와 같은 본 발명의 범위를 제한하도록 의도되는 것은 아니며, 본 발명의 선택된 실시예들을 나타낼 뿐이다.

[0095] 본 명세서 전반에 걸쳐 설명된 본 발명의 피처들, 구조들, 또는 특성들이 하나 이상의 실시예에서 임의의 적합한 방식으로 조합될 수 있다. 예를 들어, "특정 실시예들", "일부 실시예들", 또는 유사한 언어에 대한 본 명세서 전반에 걸친 참조는, 실시예와 관련하여 설명된 특정 피처, 구조, 또는 특성이 본 발명의 적어도 하나의 실시예에 포함된다는 것을 의미한다. 따라서, 본 명세서 전반에 걸친 문구들 "특정 실시예들에서", "일부 실시예에서", "다른 실시예들에서", 또는 유사한 언어의 출현이 반드시 모두 동일한 그룹의 실시예들을 지칭하는 것은 아니며 설명된 피처들, 구조들, 또는 특성들이 하나 이상의 실시예에서 임의의 적합한 방식으로 조합될 수 있다.

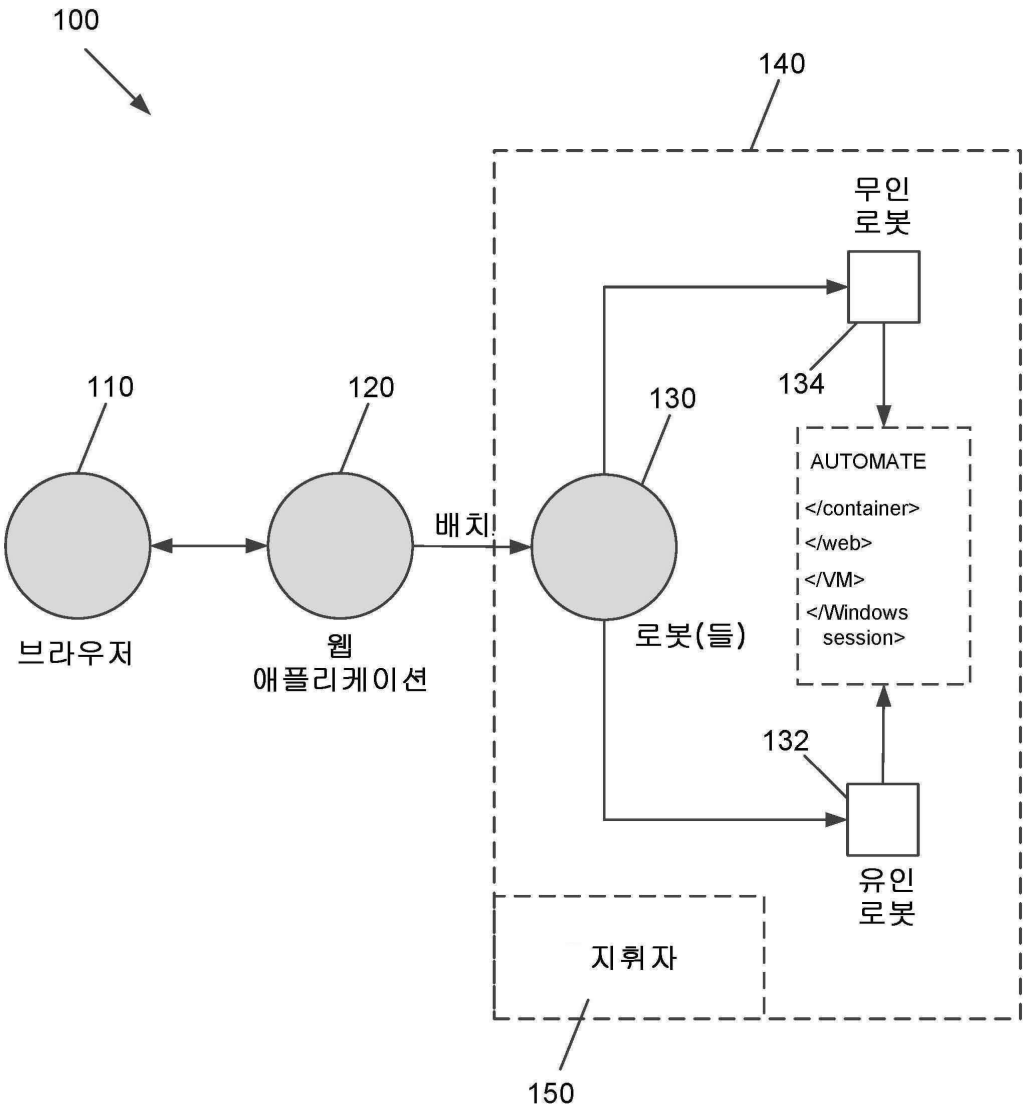
[0096] 피처들, 이점들, 또는 유사한 언어에 대한 본 명세서 전반에 걸친 참조가 본 발명으로 실현될 수 있는 피처들 및 이점들 모두가 본 발명의 임의의 단일 실시예에 있어야 하거나 있는 것을 의미하는 것은 아니라는 점에 유념해야 한다. 그 보다는, 피처들 및 이점들을 참조하는 언어는 실시예와 관련하여 설명된 특정 피처, 이점, 또는 특성이 본 발명의 적어도 하나의 실시예에 포함된다는 것을 의미하는 것으로 이해된다. 따라서, 본 명세서 전반에 걸친 피처들 및 이점들, 및 유사한 언어의 논의가 동일한 실시예를 참조할 수 있지만, 반드시 그런 것은 아니다.

[0097] 또한, 본 발명의 설명된 피처들, 이점들, 및 특성들이 하나 이상의 실시예에서 임의의 적합한 방식으로 조합될 수 있다. 당업자는, 본 발명이 특정 실시예의 특정 피처들 또는 이점들 중 하나 이상 없이 실시될 수 있다는 점을 인식할 것이다. 다른 경우들에서, 추가 피처들 및 이점들이 본 발명의 모든 실시예들에 존재하지 않을 수 있는 특정 실시예들에서 인식될 수 있다.

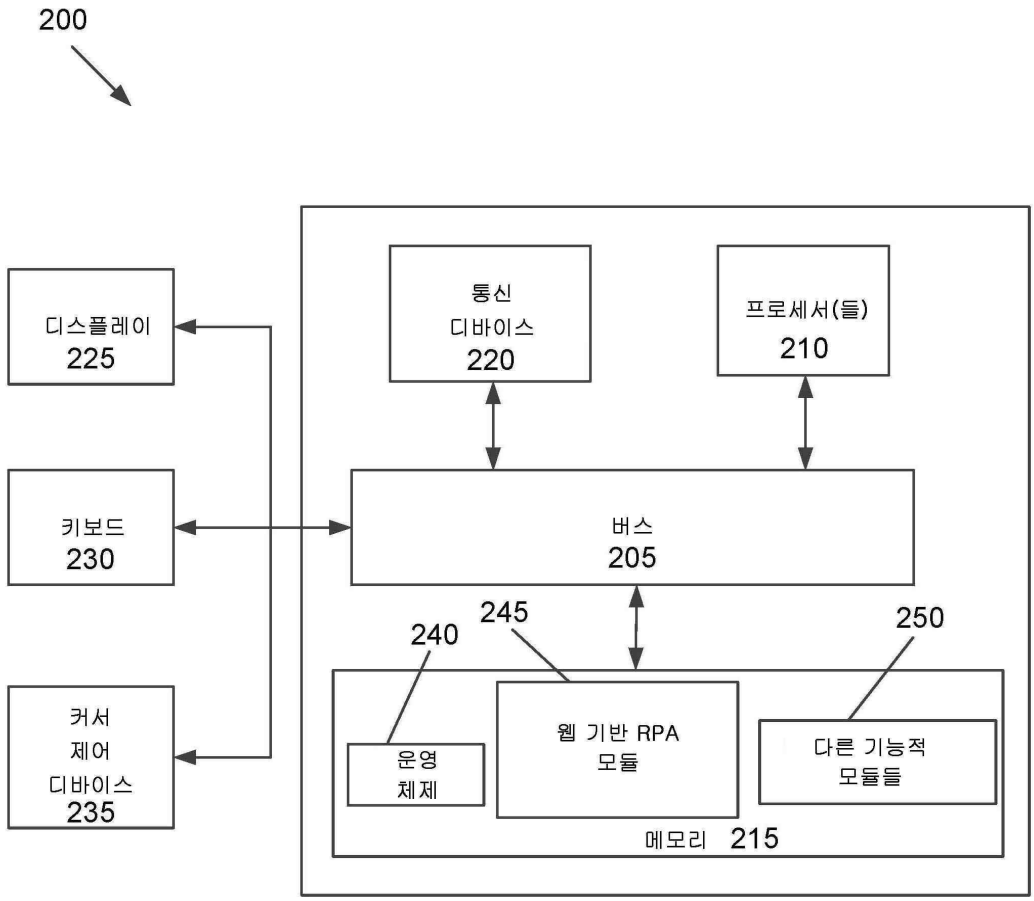
[0098] 당업자는, 위에서 논의된 바와 같은 본 발명이 상이한 순서에 있는 단계들로, 그리고/또는 개시된 것과는 상이한 구성들에 있는 하드웨어 요소들로 실시될 수 있다는 점을 쉽게 이해할 것이다. 따라서, 본 발명이 이 바람직한 실시예들에 기초하여 설명되었지만, 특정 수정들, 변형들, 및 대안적인 구성들이 본 발명의 사상 및 범위 내에 남아 있으면서 명백할 것이라는 점이 당업자에게 명백할 것이다. 본 발명의 한계 및 경계를 결정하기 위해, 따라서, 첨부된 청구범위에 대한 참조가 이루어져야 한다.

## 도면

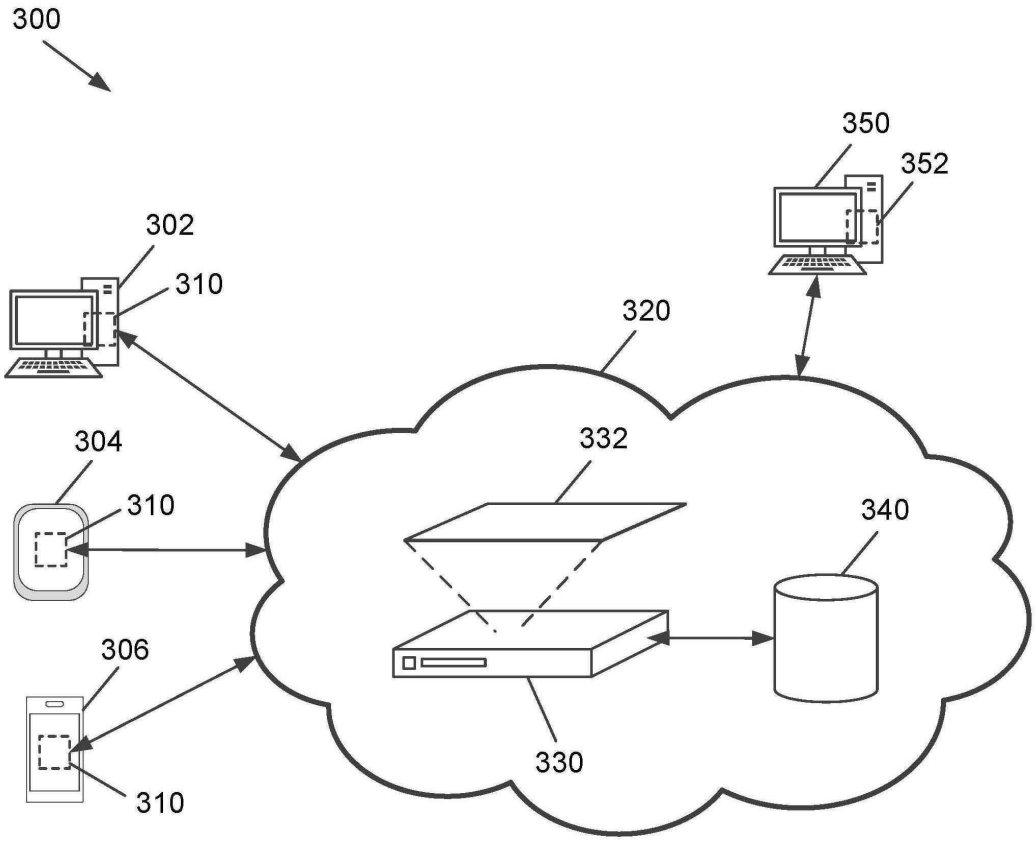
### 도면1



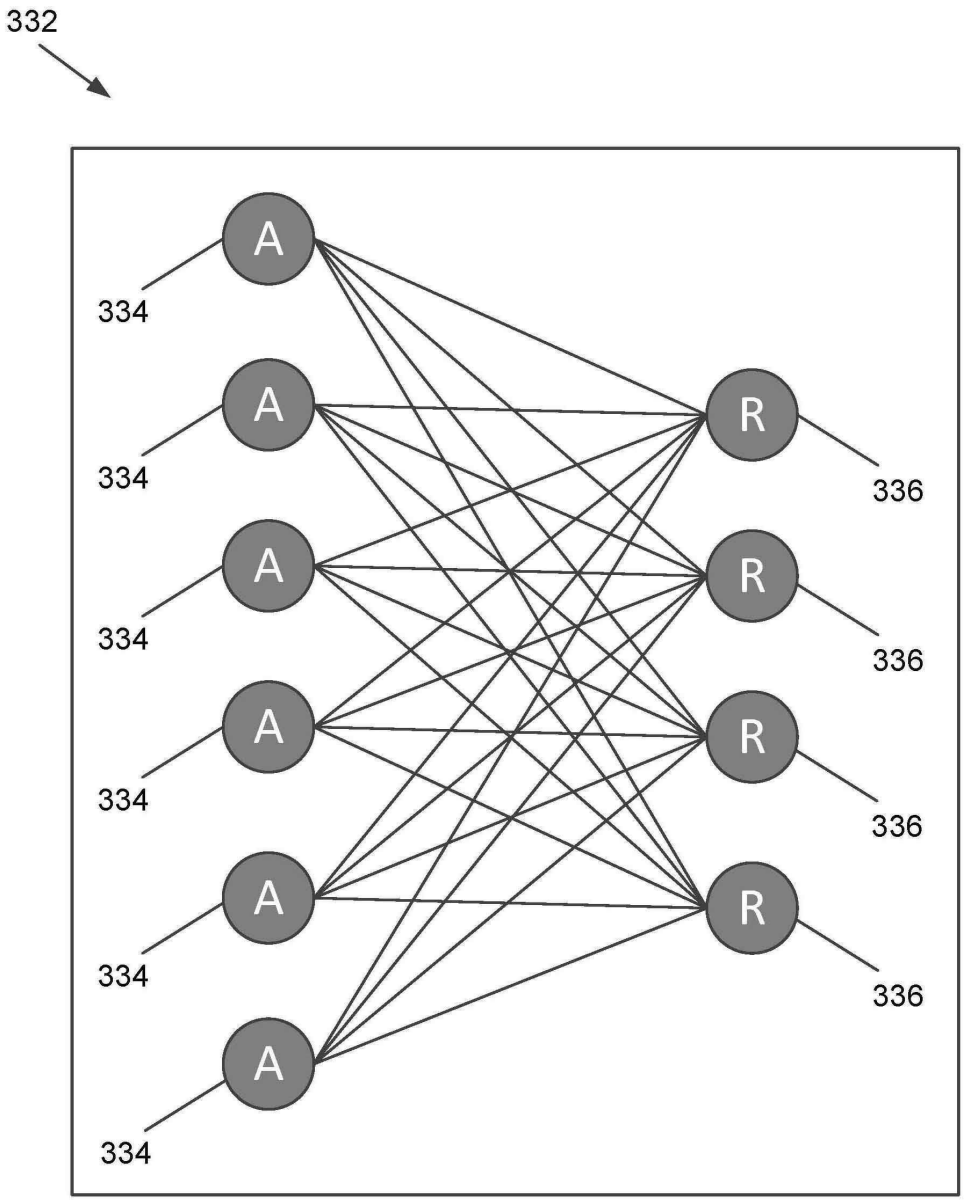
도면2


200
디스플레이
225
키보드
230
커서
제어
디바이스
235
통신
디바이스
220
프로세서(들)
210
버스
205
240
245
250
운영
체제
웹 기반 RPA
모듈
다른 기능적
모듈들
메모리 215

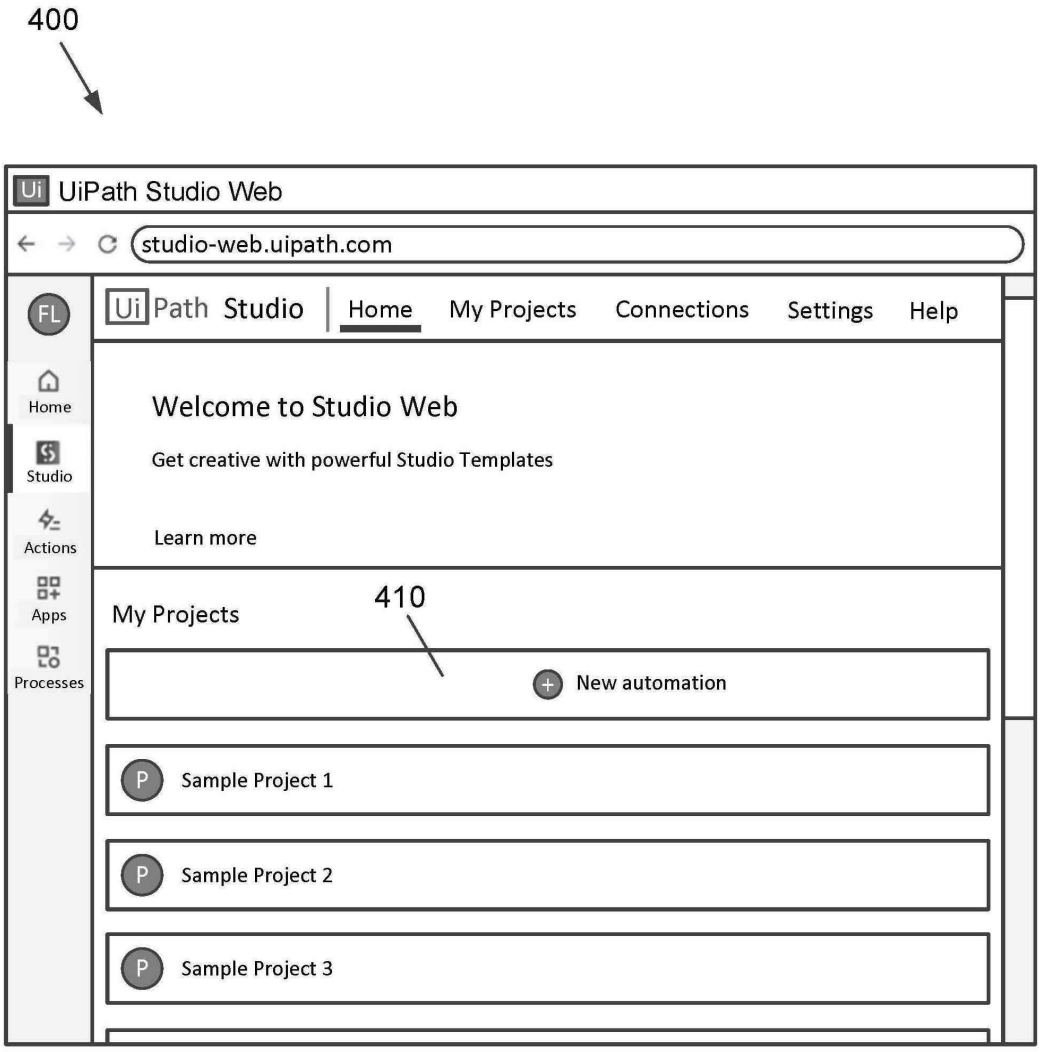
도면3a



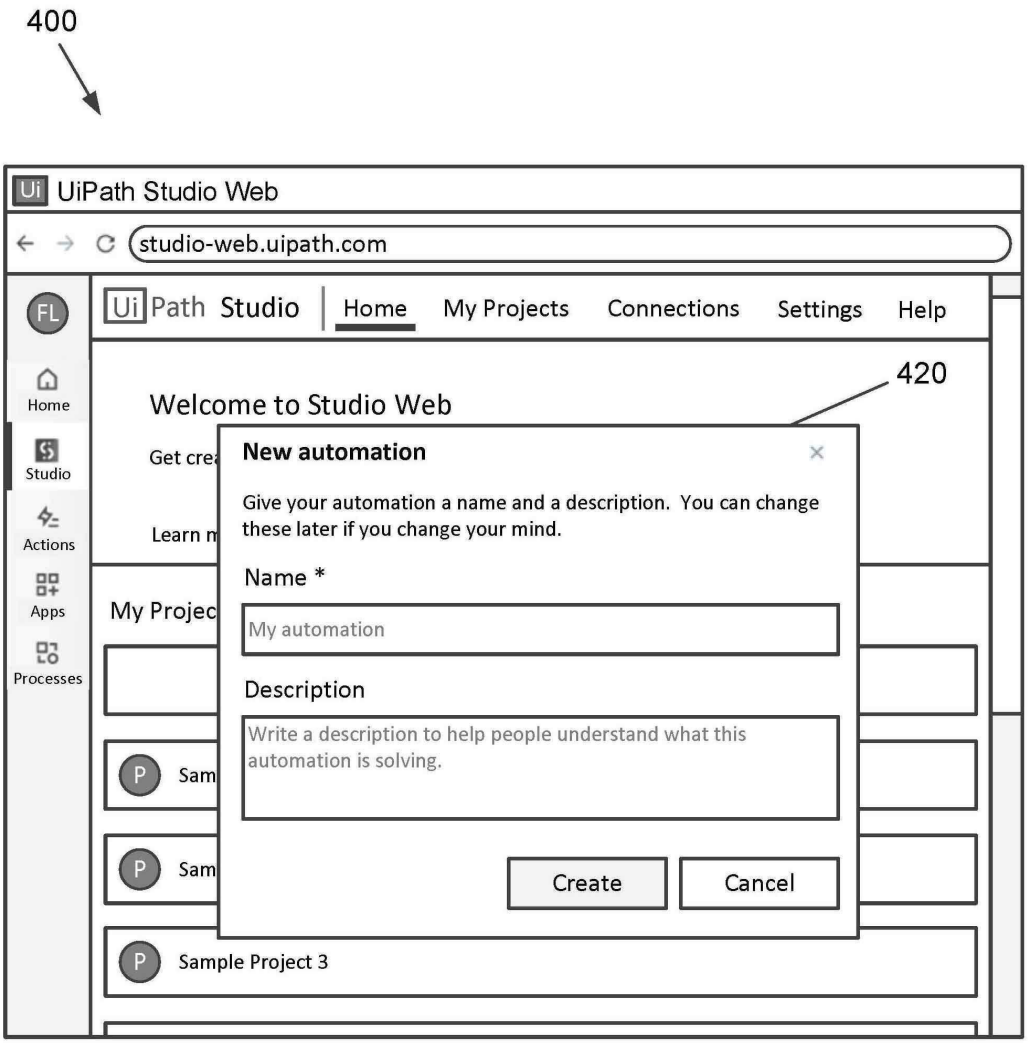
도면3b



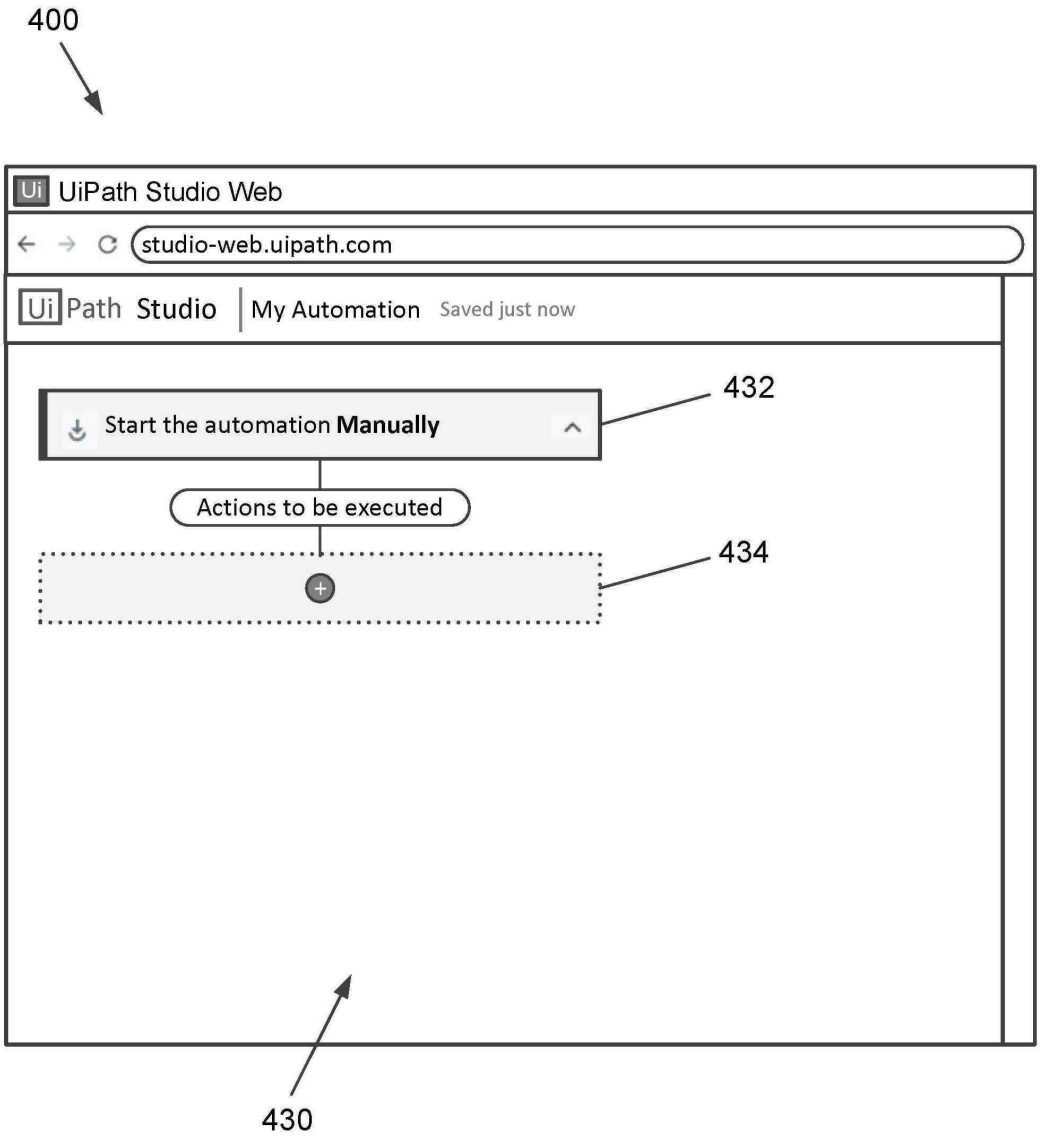
도면4a


400
UiPath Studio Web
studio-web.uipath.com
FL
UiPath Studio
Home
My Projects
Connections
Settings
Help
Home
Studio
Actions
Apps
Processes
Welcome to Studio Web
Get creative with powerful Studio Templates
Learn more
My Projects
410
New automation
Sample Project 1
Sample Project 2
Sample Project 3

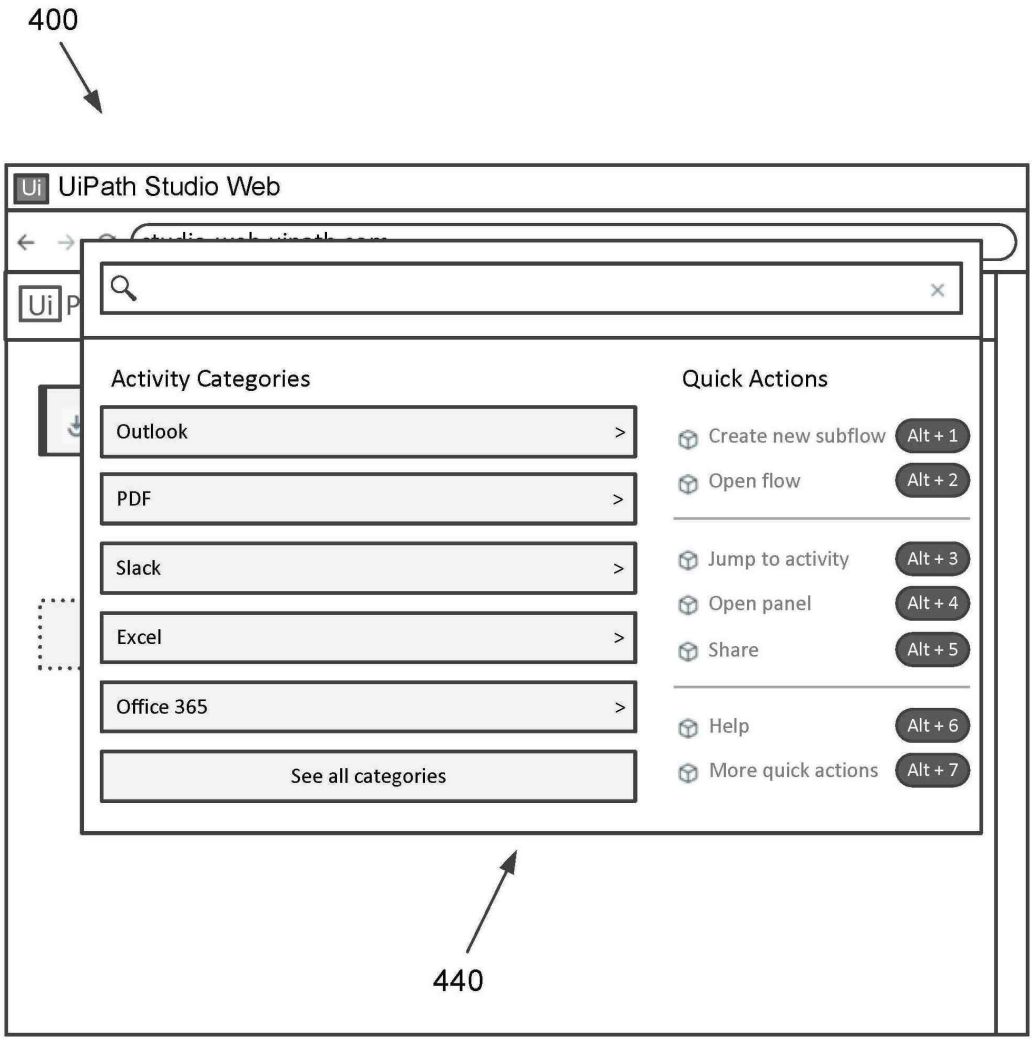
도면4b


400
UiPath Studio Web
studio-web.uipath.com
FL
UiPath Studio
Home
My Projects
Connections
Settings
Help
Home
Studio
Actions
Apps
Processes
Welcome to Studio Web
420
New automation
Give your automation a name and a description. You can change these later if you change your mind.
Name *
My automation
Description
Write a description to help people understand what this automation is solving.
Create
Cancel
Sample Project 3

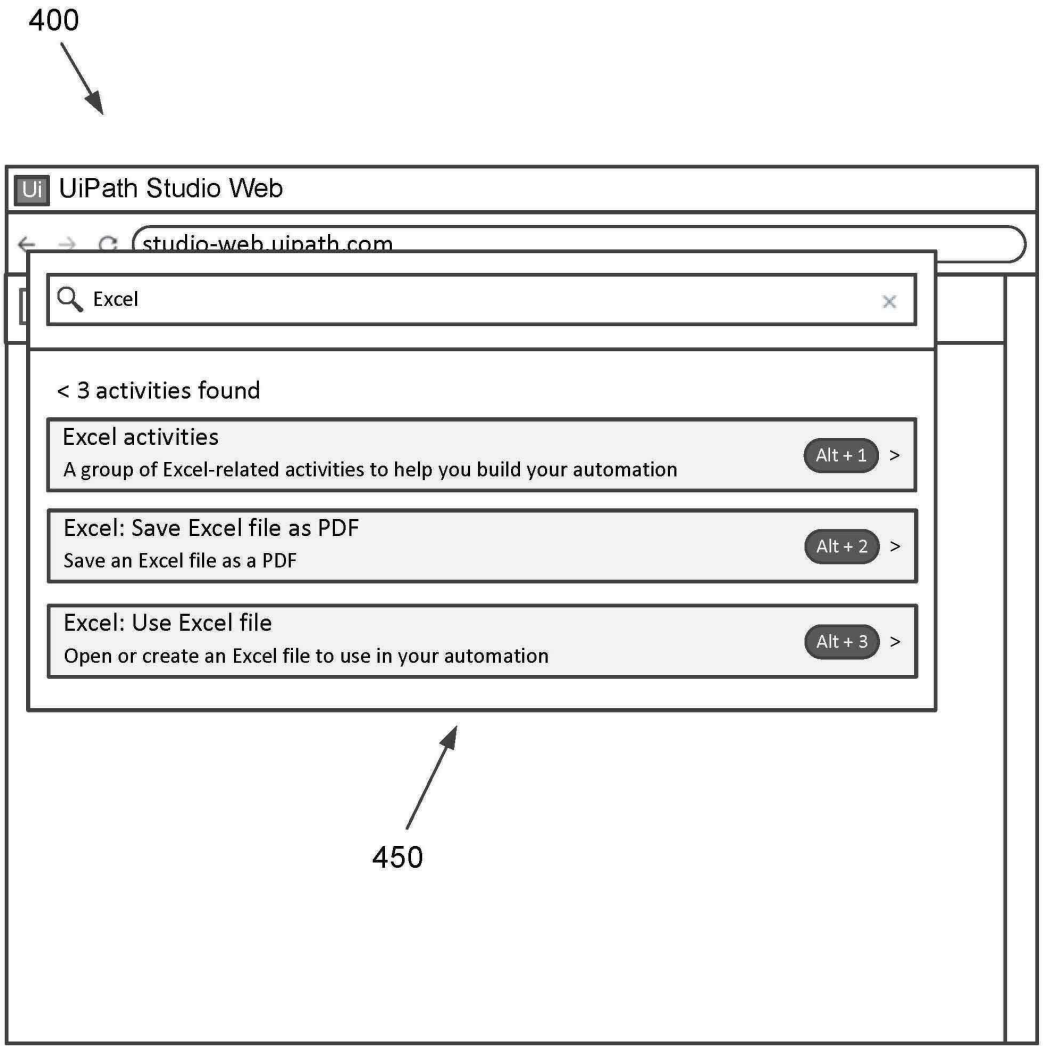
도면4c


400
UiPath Studio Web
studio-web.uipath.com
UiPath Studio
My Automation
Saved just now
Start the automation Manually
Actions to be executed
432
434
430

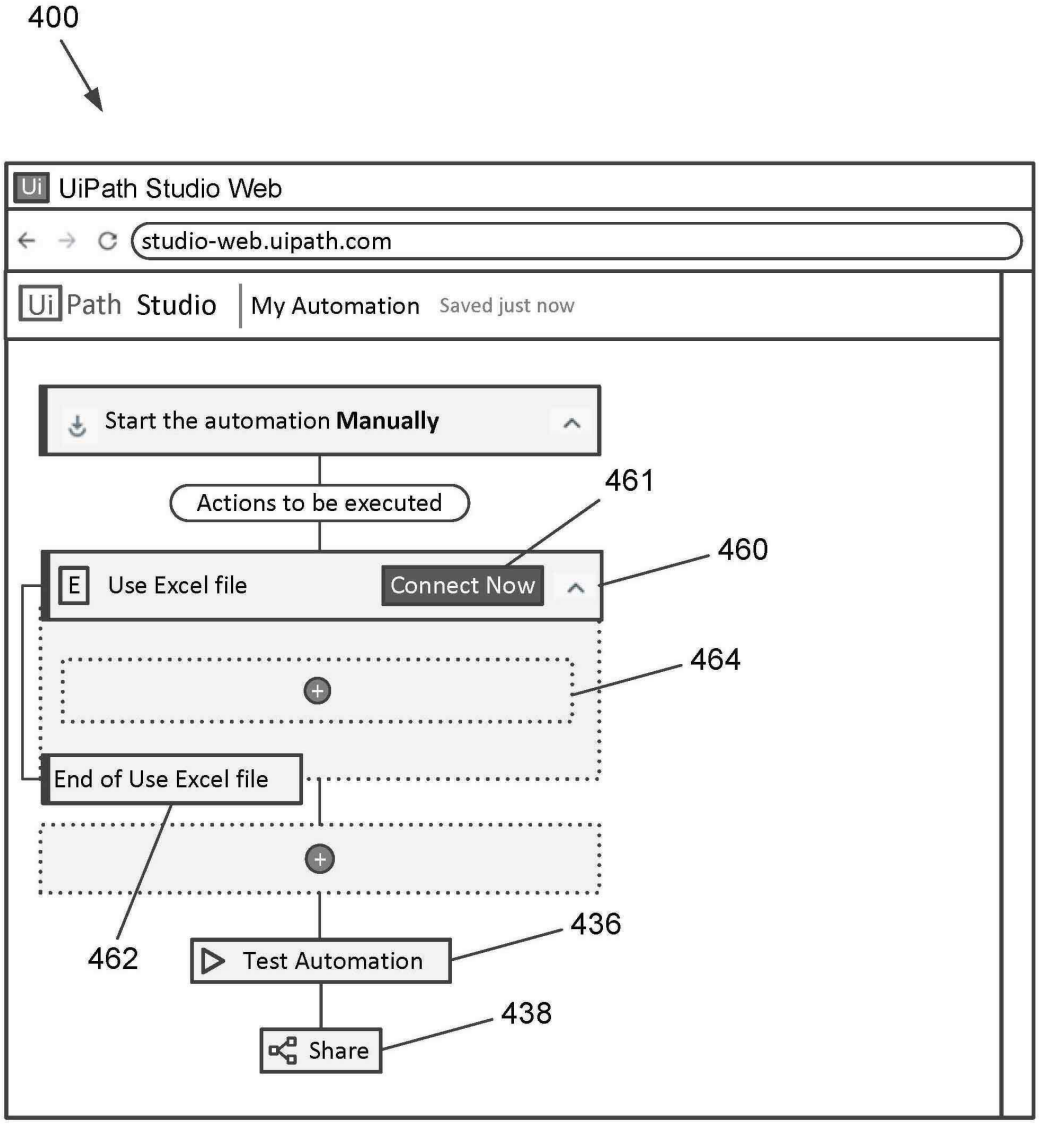
도면4d


400
UiPath Studio Web
Activity Categories
Outlook
PDF
Slack
Excel
Office 365
See all categories
Quick Actions
Create new subflow
Alt + 1
Open flow
Alt + 2
Jump to activity
Alt + 3
Open panel
Alt + 4
Share
Alt + 5
Help
Alt + 6
More quick actions
Alt + 7
440

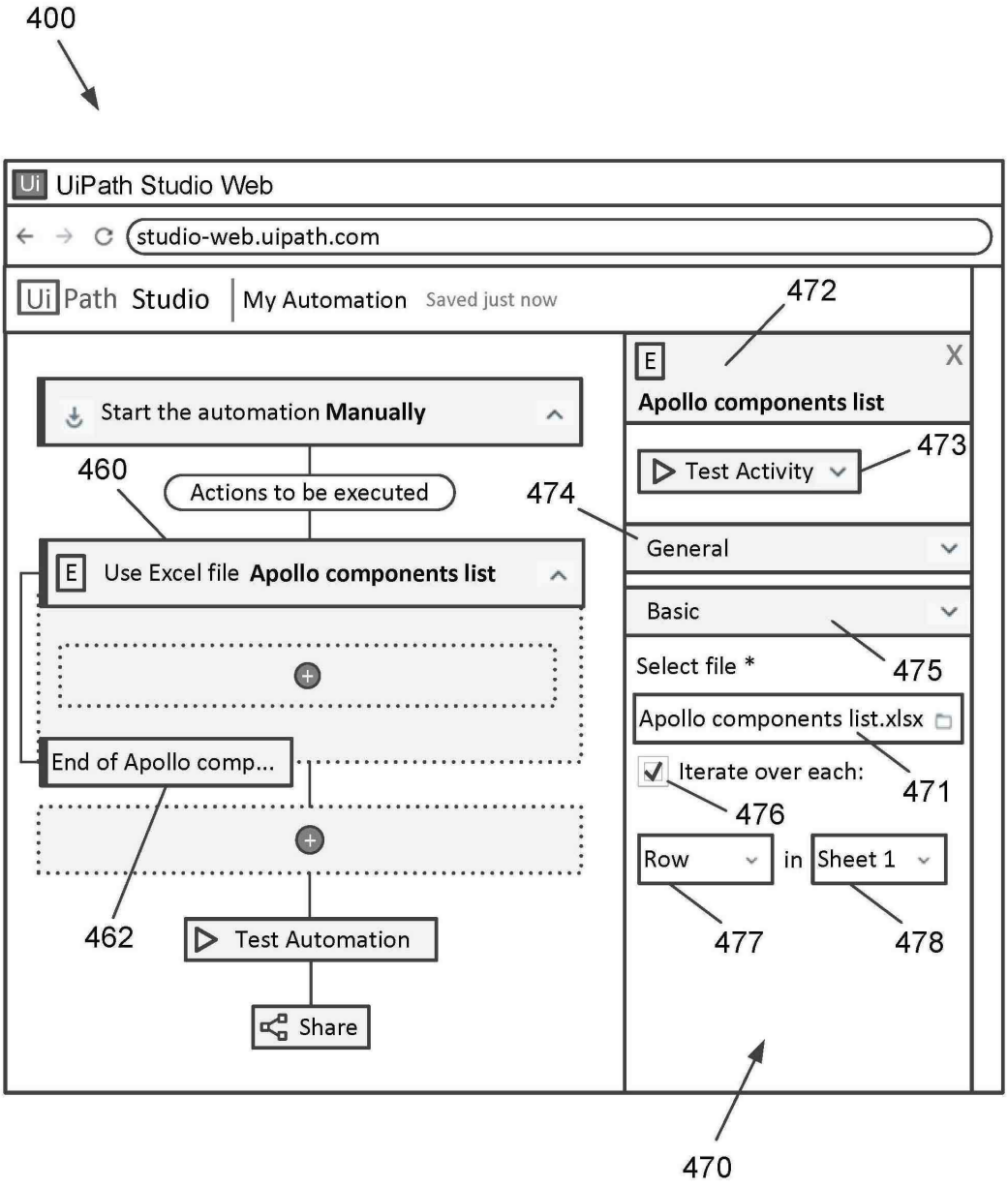
도면4e


400
UiPath Studio Web
studio-web.uipath.com
Excel
< 3 activities found
Excel activities
A group of Excel-related activities to help you build your automation
Alt + 1
Excel: Save Excel file as PDF
Save an Excel file as a PDF
Alt + 2
Excel: Use Excel file
Open or create an Excel file to use in your automation
Alt + 3
450

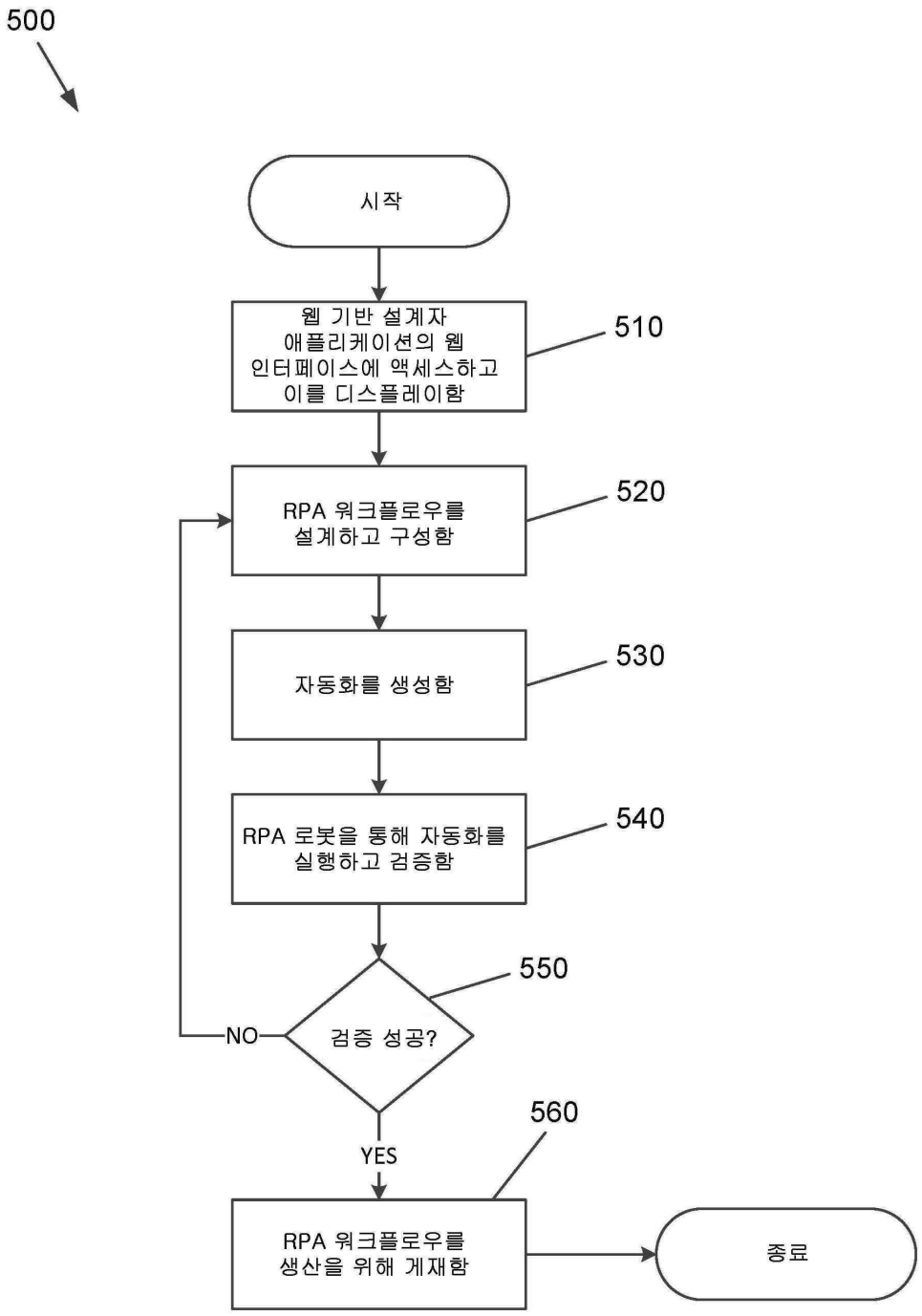
도면4f


400
UiPath Studio Web
studio-web.uipath.com
Path Studio
My Automation
Saved just now
Start the automation Manually
Actions to be executed
461
460
Use Excel file
Connect Now
464
End of Use Excel file
462
436
Test Automation
438
Share

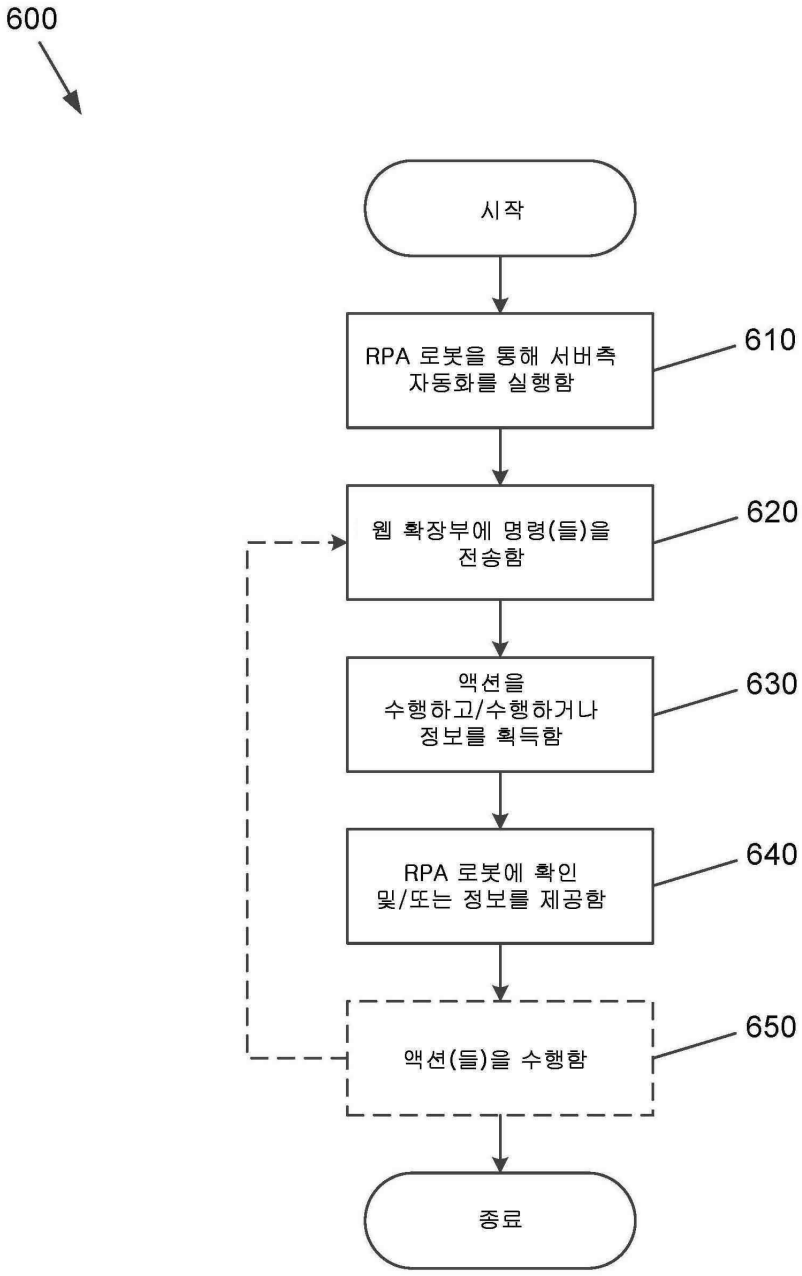
도면4g


400
UiPath Studio Web
studio-web.uipath.com
Path Studio
My Automation
Saved just now
472
Apollo components list
Start the automation Manually
460
Actions to be executed
474
Use Excel file Apollo components list
End of Apollo comp...
462
Test Automation
Share
Test Activity
473
General
Basic
Select file *
475
Apollo components list.xlsx
Iterate over each:
471
476
Row
in
Sheet 1
477
478
470

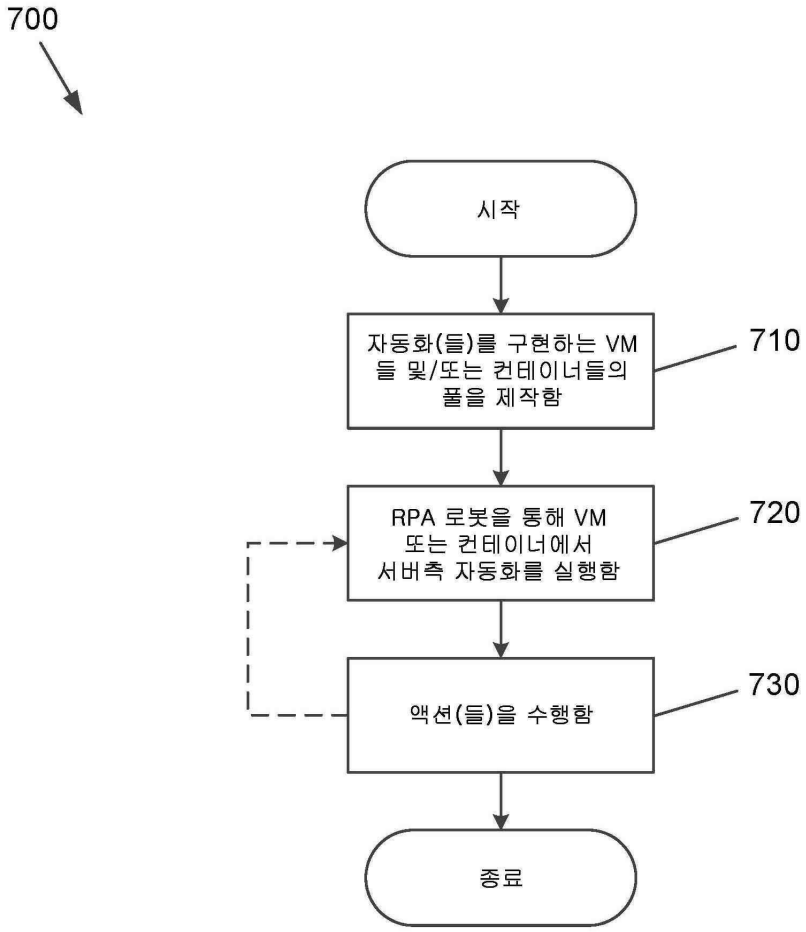
도면5


500
시작
웹 기반 설계자 애플리케이션의 웹 인터페이스에 액세스하고 이를 디스플레이함
510
RPA 워크플로우를 설계하고 구성함
520
자동화를 생성함
530
RPA 로봇을 통해 자동화를 실행하고 검증함
540
검증 성공?
550
NO
YES
RPA 워크플로우를 생산을 위해 게재함
560
종료

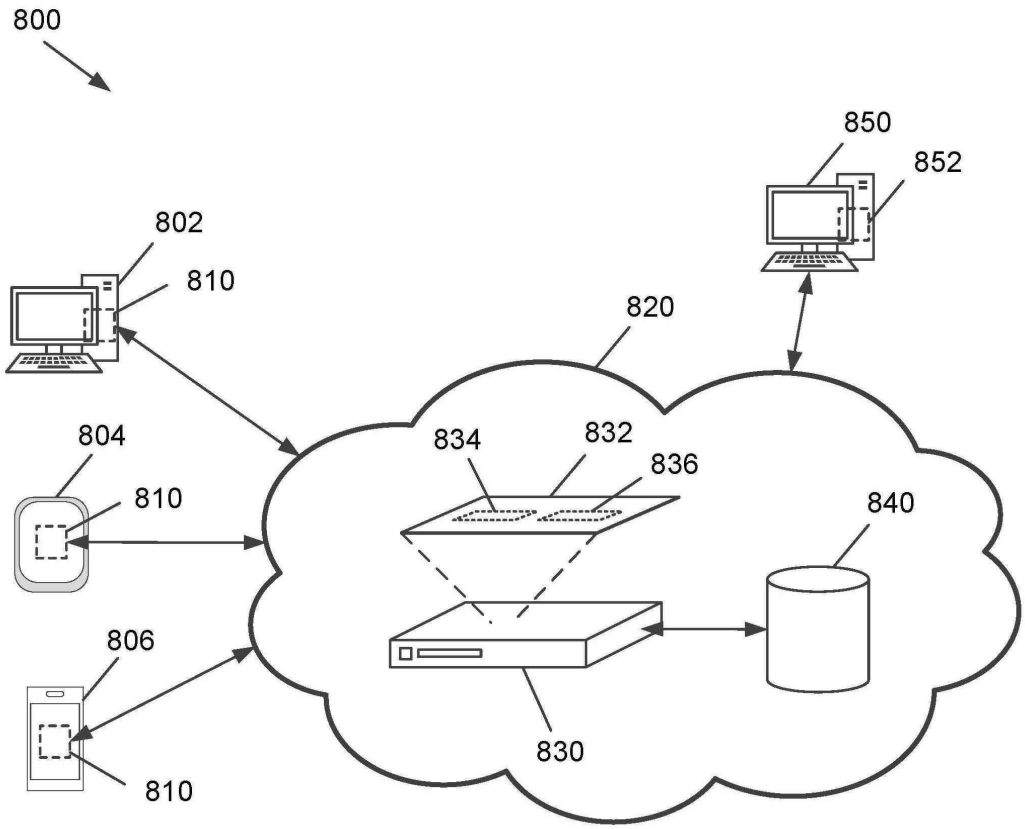
도면6


600
시작
RPA 로봇을 통해 서버측 자동화를 실행함
610
웹 확장부에 명령(들)을 전송함
620
액션을 수행하고/수행하거나 정보를 획득함
630
RPA 로봇에 확인 및/또는 정보를 제공함
640
액션(들)을 수행함
650
종료

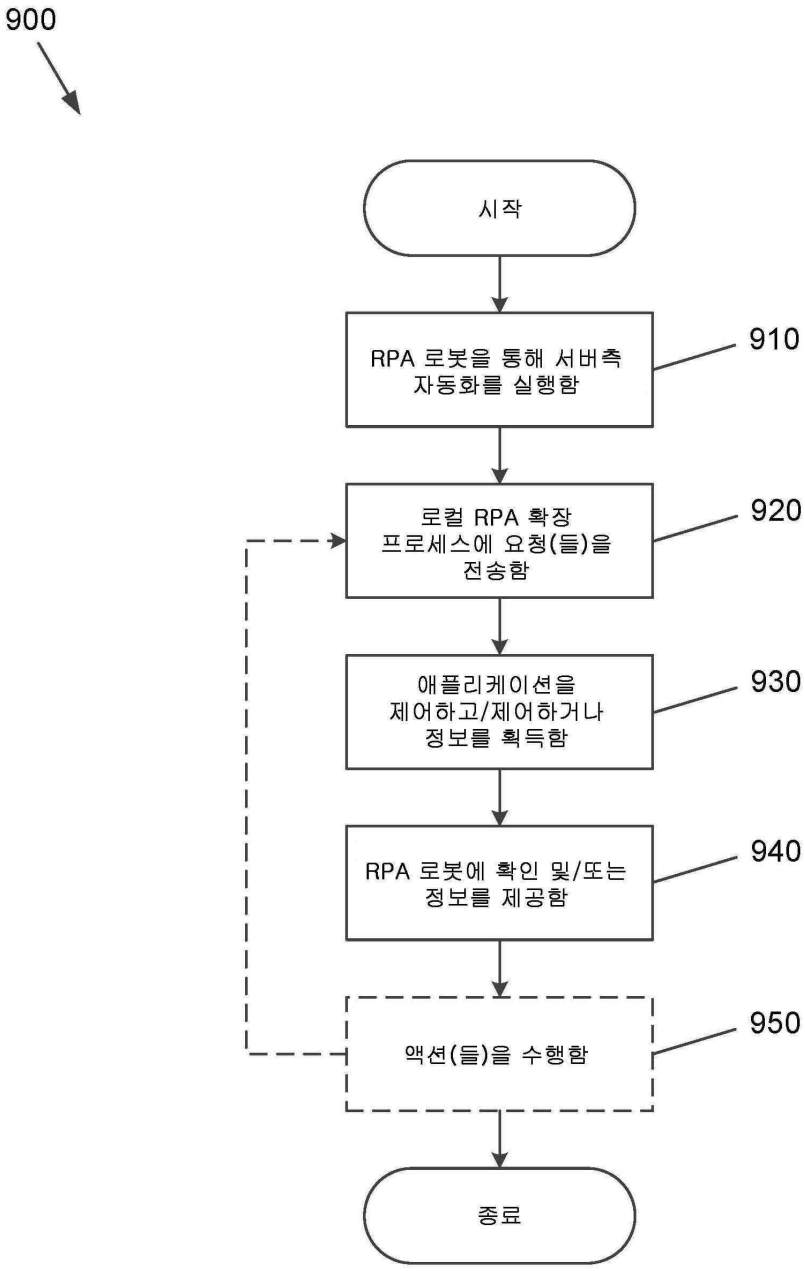
도면7


700
시작
자동화(들)를 구현하는 VM 들 및/또는 컨테이너들의 풀을 제작함
710
RPA 로봇을 통해 VM 또는 컨테이너에서 서버측 자동화를 실행함
720
액션(들)을 수행함
730
종료

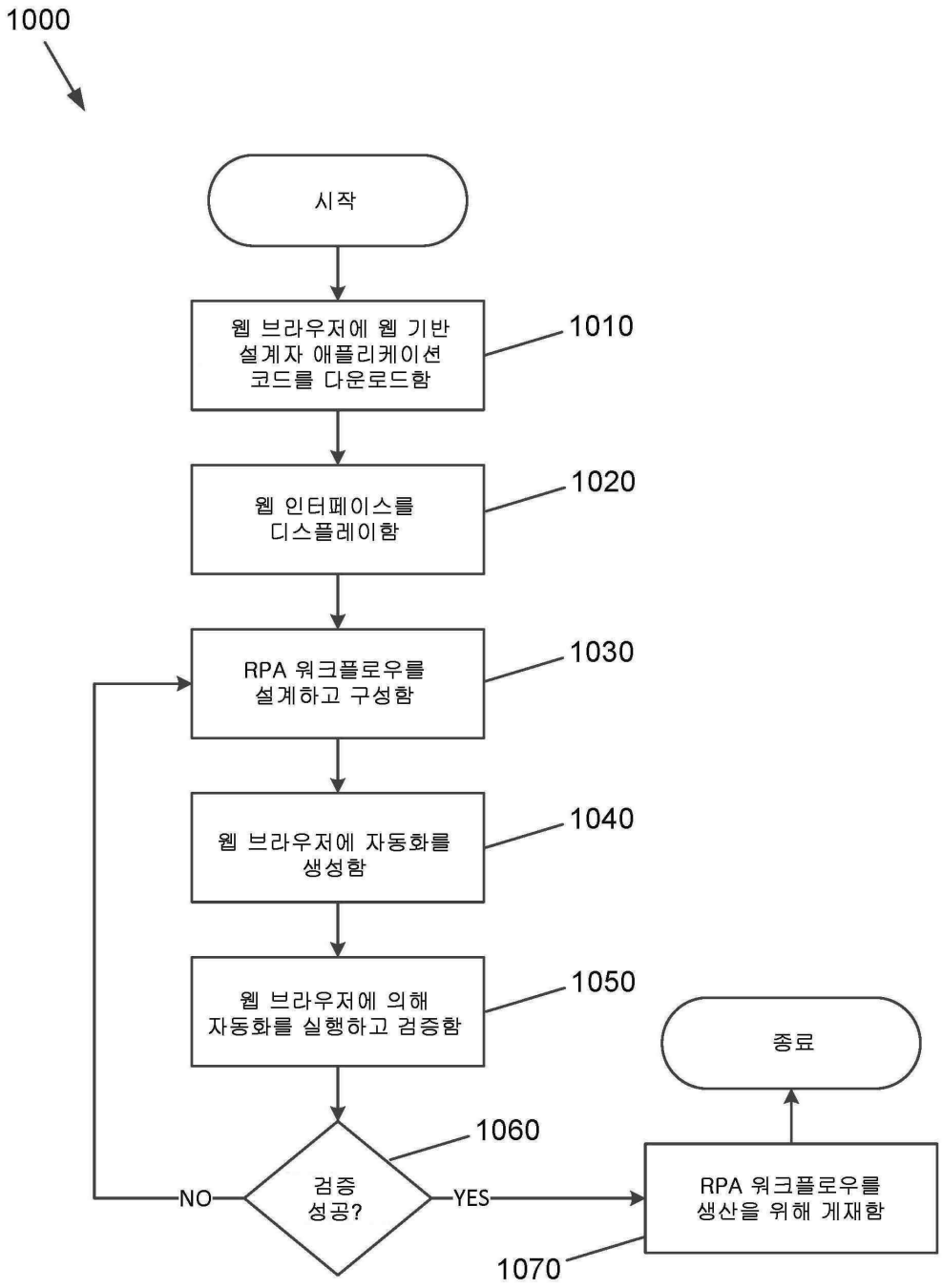
도면8



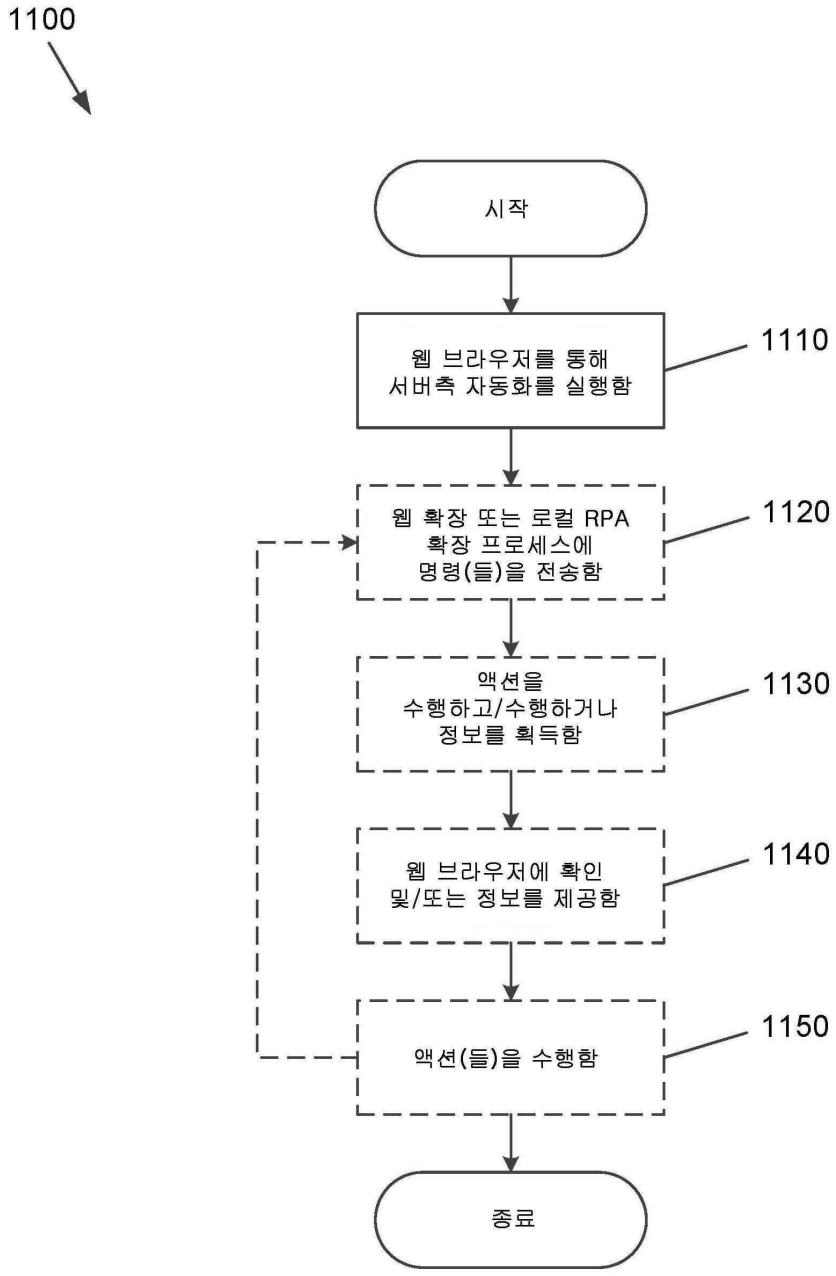
도면9


900
시작
RPA 로봇을 통해 서버측 자동화를 실행함
910
로컬 RPA 확장 프로세스에 요청(들)을 전송함
920
애플리케이션을 제어하고/제어하거나 정보를 획득함
930
RPA 로봇에 확인 및/또는 정보를 제공함
940
액션(들)을 수행함
950
종료

도면10


1000
시작
웹 브라우저에 웹 기반 설계자 애플리케이션 코드를 다운로드함
1010
웹 인터페이스를 디스플레이함
1020
RPA 워크플로우를 설계하고 구성함
1030
웹 브라우저에 자동화를 생성함
1040
웹 브라우저에 의해 자동화를 실행하고 검증함
1050
검증 성공?
1060
NO
YES
RPA 워크플로우를 생산을 위해 게재함
1070
종료

도면11


1100
시작
웹 브라우저를 통해
서버측 자동화를 실행함
1110
웹 확장 또는 로컬 RPA
확장 프로세스에
명령(들)을 전송함
1120
액션을
수행하고/수행하거나
정보를 획득함
1130
웹 브라우저에 확인
및/또는 정보를 제공함
1140
액션(들)을 수행함
1150
종료