



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2020-0077830
(43) 공개일자 2020년07월01일

(51) 국제특허분류(Int. Cl.)
G06F 11/36 (2006.01) H04L 29/08 (2006.01)
(52) CPC특허분류
G06F 11/3672 (2013.01)
H04L 67/104 (2013.01)
(21) 출원번호 10-2018-0167085
(22) 출원일자 2018년12월21일
심사청구일자 2018년12월21일

(71) 출원인
이화여자대학교 산학협력단
서울특별시 서대문구 이화여대길 52 (대현동, 이화여자대학교)
(72) 발명자
최병주
서울특별시 강남구 압구정로 201, 74동 1305호(압구정동, 현대아파트)
송인혜
경상남도 김해시 율하3로 76, 810동 1401호(율하동, 신리마을 중앙하이츠아파트)
(74) 대리인
특허법인(유한)아이시스

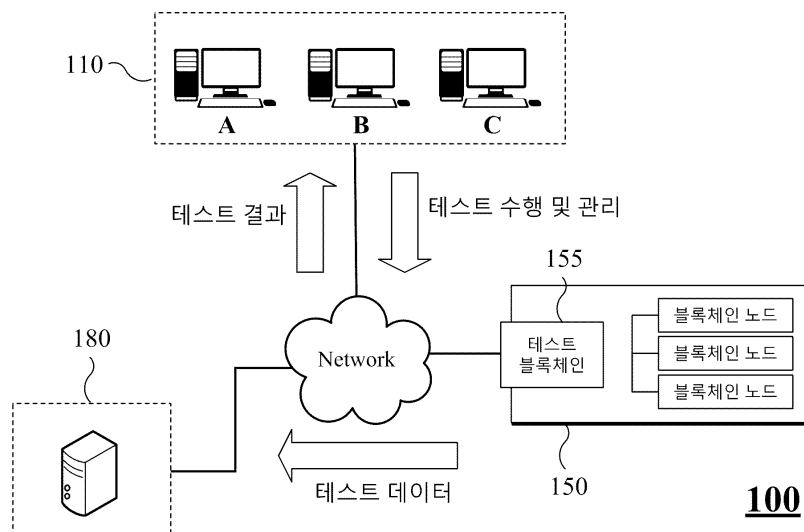
전체 청구항 수 : 총 13 항

(54) 발명의 명칭 **블록체인 기반 테스트 데이터 관리 방법 및 블록체인 노드 장치**

(57) 요약

블록체인 기반 테스트 데이터 관리 방법은 사용자가 테스트 데이터를 항목으로 포함하는 제1 스마트 계약을 상기 블록체인 네트워크에 배포하여 테스트 케이스를 생성하는 단계 및 상기 사용자가 제2 스마트 계약을 상기 블록체인 네트워크에 배포하여 상기 테스트 케이스를 테스트 시스템에 전달하는 단계를 포함한다. 상기 제1 스마트 계약이 사전에 정해진 항목을 포함하는 경우에만 상기 테스트 케이스가 생성된다.

대표도 - 도1



(52) CPC특허분류

H04L 67/1097 (2013.01)

이 발명을 지원한 국가연구개발사업

과제고유번호 1711070410

부처명 과학기술정보통신부

연구관리전문기관 정보통신기술진흥센터

연구사업명 정보통신기술인력양성(정보화)

연구과제명 적응형 블록체인 플랫폼 기술 개발 및 전문 인력 양성

기 여 율 1/1

주관기관 서강대학교 산학협력단

연구기간 2018.01.01 ~ 2018.12.31

명세서

청구범위

청구항 1

사용자가 테스트 데이터를 항목으로 포함하는 제1 스마트 계약을 상기 블록체인 네트워크에 배포하여 테스트 케이스를 생성하는 단계; 및

상기 사용자가 제2 스마트 계약을 상기 블록체인 네트워크에 배포하여 상기 테스트 케이스를 테스트 시스템에 전달하는 단계를 포함하되,

상기 제1 스마트 계약이 사전에 정해진 항목을 포함하는 경우에만 상기 테스트 케이스가 생성되는 블록체인 기반 테스트 데이터 관리 방법.

청구항 2

제1항에 있어서,

상기 제1 스마트 계약은 상기 테스트 케이스 식별자, 상기 테스트 케이스 저장 위치, 상기 테스트 데이터 및 상기 테스트 케이스에 대한 실행 절차를 포함하는 블록체인 기반 테스트 데이터 관리 방법.

청구항 3

제2항에 있어서,

상기 제1 스마트 계약은 상기 테스트 케이스의 사전 조건, 상기 테스트 케이스의 예상 결과 및 상기 테스트 케이스의 생성일 중 적어도 하나를 더 포함하는 블록체인 기반 테스트 데이터 관리 방법.

청구항 4

제1항에 있어서,

상기 제2 스마트 계약은 상기 테스트 케이스의 주소 및 상기 테스트 시스템의 위치를 포함하는 블록체인 기반 테스트 데이터 관리 방법.

청구항 5

제1항에 있어서,

상기 사용자가 프로젝트 식별자, 프로젝트 시작시점, 프로젝트 종료시점 및 프로젝트 설명을 포함하는 제3 스마트 계약을 상기 블록체인 네트워크에 배포하여 테스트 케이스에 대한 프로젝트를 생성하는 단계를 더 포함하는 블록체인 기반 테스트 데이터 관리 방법.

청구항 6

제1항에 있어서,

상기 사용자 또는 상기 테스트 시스템이 상기 테스트 케이스를 실행한 결과 데이터를 포함하는 제4 스마트 계약을 상기 블록체인 네트워크에 배포하여 상기 테스트 케이스에 대한 결함 정보를 생성하는 블록체인 기반 테스트 데이터 관리 방법.

청구항 7

제6항에 있어서,

상기 제4 스마트 계약은 프로젝트 식별자, 상기 사용자의 주소, 테스트 결과 데이터, 테스트 결과의 심각도 및 테스트 결과의 우선 순위를 포함하는 블록체인 기반 테스트 데이터 관리 방법.

청구항 8

클라이언트 장치로부터 테스트 데이터를 항목으로 포함하는 제1 스마트 계약을 수신하는 통신 장치;

상기 제1 스마트 계약이 사전에 정해진 항목을 포함하는 경우에만 상기 제1 스마트 계약을 실행하여 테스트 케이스를 생성하는 연산 장치; 및

상기 테스트 케이스를 저장하는 저장 장치를 포함하되,

상기 제1 스마트 계약은 상기 테스트 케이스 식별자, 상기 테스트 케이스 저장 위치, 상기 테스트 데이터 및 상기 테스트 케이스에 대한 실행 절차를 포함하는 테스트 데이터를 관리하는 블록체인의 노드 장치.

청구항 9

제8항에 있어서,

상기 통신 장치는 상기 클라이언트 장치로부터 제2 스마트 계약을 더 수신하고, 상기 테스트 케이스를 테스트 시스템에 송신하고,

상기 연산 장치는 상기 제2 스마트 계약이 사전에 정해진 항목을 포함하는 경우에만 상기 제2 스마트 계약을 실행하여 상기 테스트 케이스를 상기 테스트 시스템에 전달하도록 제어하되,

상기 제2 스마트 계약은 상기 테스트 케이스의 주소 및 상기 테스트 시스템의 위치를 포함하는 테스트 데이터를 관리하는 블록체인의 노드 장치.

청구항 10

제8항에 있어서,

상기 통신 장치는 상기 클라이언트 장치로부터 프로젝트 식별자, 프로젝트 시작시점, 프로젝트 종료시점 및 프로젝트 설명을 포함하는 제3 스마트 계약을 수신하고,

상기 연산 장치는 상기 제3 스마트 계약을 실행하여 테스트 케이스에 대한 프로젝트를 생성하고,

상기 저장 장치는 상기 프로젝트를 저장하는 테스트 데이터를 관리하는 블록체인의 노드 장치.

청구항 11

제8항에 있어서,

상기 통신 장치는 상기 테스트 시스템이 상기 테스트 케이스를 실행한 결과 데이터를 포함하는 제4 스마트 계약을 수신하고,

상기 연산 장치는 상기 제4 스마트 계약을 실행하여 테스트 케이스에 대한 결함 정보를 생성하고,

상기 저장 장치는 상기 결함 정보를 저장하는 테스트 데이터를 관리하는 블록체인의 노드 장치.

청구항 12

제11항에 있어서,

상기 제4 스마트 계약은 프로젝트 식별자, 상기 사용자의 주소, 테스트 결과 데이터, 테스트 결과의 심각도 및 테스트 결과의 우선 순위를 포함하는 테스트 데이터를 관리하는 블록체인의 노드 장치.

청구항 13

컴퓨터에서 제1항 내지 제7항 중 어느 하나의 항에 기재된 블록체인 기반 테스트 데이터 관리 방법을 실행하기 위한 프로그램을 기록한 컴퓨터로 읽을 수 있는 기록 매체.

발명의 설명

기술 분야

이하 설명하는 기술은 블록체인 기반한 테스트 데이터 관리 기법에 관한 것이다.

[0001]

배경 기술

- [0002] 소프트웨어 시스템은 4차 산업 혁명이 대두되면서 인공지능(AI), 빅데이터, 자율 주행, 가상현실 등에 이르기까지 생활의 많은 부분에서 폭넓게 사용되고 있다. 소프트웨어 시스템은 발생 가능한 문제를 최소화하기 위하여 소프트웨어 테스트가 필요하다. 소프트웨어 테스트는 결함을 발견하여 사전에 수정함으로써 품질을 높이기 위한 것이다.

선행기술문헌

특허문헌

- [0003] (특허문헌 0001) 한국등록특허 제10-1335912호

발명의 내용

해결하려는 과제

- [0004] 소프트웨어 테스트는 테스트 수행 과정에서 산출된 테스트 결과 및 결함 등에 대한 기록 관리를 지속적으로 수행해야 한다. 따라서 대다수 기업에서는 품질 향상을 위해 테스트 관리 도구를 활용하여 업무의 효율을 높이고 있다.
- [0005] 종래 테스트 관리 도구는 다음과 같은 한계가 있다. 첫째, 변경 이력에 대한 데이터가 존재하지 않는다. 종래 테스트 관리 도구는 데이터 항목에 대한 삭제가 가능하다. 하지만, 여러 사용자가 데이터를 등록 및 삭제하는 경우 이전 사용자들이 수행한 변경 이력을 확인하기가 힘들다. 둘째, 종래 테스트 관리 도구는 관리 도구를 제어하는 제 3자에 의해서 관리된다. 즉, 사용자(tester)가 데이터를 업로드 하기 위해서는 전체적인 동작을 제어하는 관리 주체를 통해 업로드 되어야 한다. 셋째, 테스트 데이터 및 관련 데이터는 클라우드 서버 하위에 여러 개의 데이터 베이스로 나뉘어 보관된다. 테스트 데이터를 저장하기 위해 테스트 설계, 테스트케이스, 실행실적 등 여러 개의 데이터 베이스로 나뉘어 저장된다. 이러한 저장 방식은 클라우드 서버를 따로 백업하지 않는 이상 중앙 서버에 대한 외부 공격에 취약하다.
- [0006] 이하 설명하는 기술은 블록체인 기반한 테스트 관리 기법을 제공하고자 한다. 구체적으로 이하 설명하는 기술은 스마트 계약(smart contract)을 이용하여 테스트 데이터 및 결과를 관리하는 기법을 제공하고자 한다.

과제의 해결 수단

- [0007] 블록체인 기반 테스트 데이터 관리 방법은 사용자가 테스트 데이터를 항목으로 포함하는 제1 스마트 계약을 상기 블록체인 네트워크에 배포하여 테스트 케이스를 생성하는 단계 및 상기 사용자가 제2 스마트 계약을 상기 블록체인 네트워크에 배포하여 상기 테스트 케이스를 테스트 시스템에 전달하는 단계를 포함한다. 상기 제1 스마트 계약이 사전에 정해진 항목을 포함하는 경우에만 상기 테스트 케이스가 생성된다.
- [0008] 테스트 데이터를 관리하는 블록체인의 노드 장치는 클라이언트 장치로부터 테스트 데이터를 항목으로 포함하는 제1 스마트 계약을 수신하는 통신 장치, 상기 제1 스마트 계약이 사전에 정해진 항목을 포함하는 경우에만 상기 제1 스마트 계약을 실행하여 테스트 케이스를 생성하는 연산 장치 및 상기 테스트 케이스를 저장하는 저장 장치를 포함한다. 상기 제1 스마트 계약은 상기 테스트 케이스 식별자, 상기 테스트 케이스 저장 위치, 상기 테스트 데이터 및 상기 테스트 케이스에 대한 실행 절차를 포함한다.

발명의 효과

- [0009] 이하 설명하는 기술은 블록체인을 이용하여 테스트 데이터 변경 이력을 확인할 수 있다. 또 이하 설명하는 기술은 스마트 계약을 사용하여 제3자 관여 없이 사용자가 테스트 데이터를 관리할 수 있다. 나아가 이하 설명하는 기술은 테스트 데이터 및 테스트 결과를 분산 저장하여 외부 공격에 강인하다.

도면의 간단한 설명

- [0010] 도 1은 블록체인 기반 테스트 관리 시스템에 대한 예이다.

도 2는 테스트 데이터 관리 장치의 구성에 대한 예이다.

도 3은 테스트 데이터 관리 장치에서 테스트 데이터를 전달하는 과정에 대한 예이다.

도 4는 블록체인에서 관리하는 구조체 배열에 대한 예이다.

도 5는 블록체인 노드 장치의 구성에 대한 예이다.

발명을 실시하기 위한 구체적인 내용

- [0011] 이하 설명하는 기술은 다양한 변경을 가할 수 있고 여러 가지 실시례를 가질 수 있는 바, 특정 실시례들을 도면에 예시하고 상세하게 설명하고자 한다. 그러나, 이는 이하 설명하는 기술을 특정한 실시 형태에 대해 한정하려는 것이 아니며, 이하 설명하는 기술의 사상 및 기술 범위에 포함되는 모든 변경, 균등물 내지 대체물을 포함하는 것으로 이해되어야 한다.
- [0012] 제1, 제2, A, B 등의 용어는 다양한 구성요소들을 설명하는데 사용될 수 있지만, 해당 구성요소들은 상기 용어들에 의해 한정되지는 않으며, 단지 하나의 구성요소를 다른 구성요소로부터 구별하는 목적으로만 사용된다. 예를 들어, 이하 설명하는 기술의 권리 범위를 벗어나지 않으면서 제1 구성요소는 제2 구성요소로 명명될 수 있고, 유사하게 제2 구성요소도 제1 구성요소로 명명될 수 있다. 및/또는 이라는 용어는 복수의 관련된 기재된 항목들의 조합 또는 복수의 관련된 기재된 항목들 중의 어느 항목을 포함한다.
- [0013] 본 명세서에서 사용되는 용어에서 단수의 표현은 문맥상 명백하게 다르게 해석되지 않는 한 복수의 표현을 포함하는 것으로 이해되어야 하고, "포함한다" 등의 용어는 설시된 특징, 개수, 단계, 동작, 구성요소, 부분품 또는 이들을 조합한 것이 존재함을 의미하는 것이지, 하나 또는 그 이상의 다른 특징들이나 개수, 단계, 동작, 구성요소, 부분품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 배제하지 않는 것으로 이해되어야 한다.
- [0014] 도면에 대한 상세한 설명을 하기에 앞서, 본 명세서에서의 구성부들에 대한 구분은 각 구성부가 담당하는 주기능 별로 구분한 것에 불과함을 명확히 하고자 한다. 즉, 이하에서 설명할 2개 이상의 구성부가 하나의 구성부로 합쳐지거나 또는 하나의 구성부가 보다 세분화된 기능별로 2개 이상으로 분화되어 구비될 수도 있다. 그리고 이하에서 설명할 구성부 각각은 자신이 담당하는 주기능 이외에도 다른 구성부가 담당하는 기능 중 일부 또는 전부의 기능을 추가적으로 수행할 수도 있으며, 구성부 각각이 담당하는 주기능 중 일부 기능이 다른 구성부에 의해 전담되어 수행될 수도 있음은 물론이다.
- [0015] 또, 방법 또는 동작 방법을 수행함에 있어서, 상기 방법을 이루는 각 과정들은 문맥상 명백하게 특정 순서를 기재하지 않은 이상 명기된 순서와 다르게 일어날 수 있다. 즉, 각 과정들은 명기된 순서와 동일하게 일어날 수도 있고 실질적으로 동시에 수행될 수도 있으며 반대의 순서대로 수행될 수도 있다.
- [0016] 이하 설명에서 사용하는 용어에 대하여 먼저 정의한다.
- [0017] 소프트웨어 테스트 또는 소프트웨어 테스트는 소프트웨어에 대한 결함 여부를 확인하는 과정을 의미한다.
- [0018] 사용자는 소프트웨어 테스트를 수행하는 주체이다. 사용자는 결국 테스터(tester)이다.
- [0019] 클라이언트 장치는 사용자가 사용하는 장치이다. 클라이언트 장치는 콘솔장치, PC, 스마트 기기 등과 같은 장치로 구현될 수 있다.
- [0020] 테스트 데이터 관리 장치는 테스트 데이터를 관리하는 장치이다. 이하 설명하는 테스트 데이터 관리 장치는 블록체인으로 구현된다.
- [0021] 테스트 시스템은 테스트 데이터를 입력받아 특정 소프트웨어를 테스트하는 객체이다. 테스트 시스템은 오픈 소스로 구현된 시스템일 수도 있다. 테스트 시스템은 소프트웨어 종류, 코드 종류, 인터페이스 등에 따라 다양한 시스템이 있을 수 있다.
- [0022] 테스트 데이터는 소프트웨어 테스트를 위해 사용하는 데이터이다. 테스트 결과 데이터는 소프트웨어 테스트 결과에 대한 데이터이다. 테스트 결과 데이터는 결함 정보를 포함할 수 있다. 결함 정보는 테스트 수행결과 일정한 결함이 나타난 경우 해당 결함에 대한 정보를 의미한다.
- [0023] 프로젝트는 사용자가 특정 소프트웨어 또는 복수의 소프트웨어를 대상으로 테스트를 수행하는 작업 단위를 의미한다.
- [0024] 퍼블릭 블록체인(Public Blockchain)은 네트워크 상의 모든 컴퓨팅 장치들을 데이터베이스로서 활용하는 블록체

인을 의미한다. 프라이빗 블록체인(Private Blockchain)은 접근이 허용된 일부 사용자(장치)들을 데이터베이스로서 활용하는 블록체인을 의미한다. 프라이빗 블록체인은 인증 등을 수행하는 별도의 서버를 포함할 수 있다. 컨소시움 블록체인(Consortium Blockchain)은 여러 주체(기관)들이 컨소시움을 이뤄 구성하는 블록체인이다. 컨소시움 블록체인도 프라이빗 블록체인과 같이 접근이 허용된 기관만 네트워크에 참여할 수 있다. 테스트 데이터 관리 장치는 퍼블릭 블록체인, 프라이빗 블록체인 또는 컨소시움 블록체인 중 어느 하나로 구현될 수 있다.

[0025] 블록체인 기술에 대하여 간략하게 설명한다.

[0026] 블록체인은 데이터를 거래할 때 중앙집중형 서버에 기록을 보관하는 기존 방식과 달리 거래 참가자 모두에게 내용을 공개하는 분산형 디지털 장부를 말한다. 블록체인에 참여한 모든 구성원이 네트워크를 통해 서로 데이터를 검증하고 저장함으로써 특정인의 임의적인 조작이 어렵도록 설계된 저장 플랫폼이라 할 수 있다. 이러한 상호분산원장(mutual distributed ledger)을 통하여 기존 중앙 집중형 네트워크 기반의 인프라를 뛰어넘는 높은 보안성, 확장성 및 투명성 등을 보장한다.

[0027] 거래 기록이 '블록'이라는 단위로 정리돼 시간별로 이어져 있는 것이 블록체인의 특징이다. 한 블록에는 앞의 블록과 뒤의 블록과 연결되는 연결 정보가 포함돼 있으며, 앞 블록의 내용을 변경하면 뒤에 이어지는 모든 블록을 다시 생성해야 한다. 따라서 과거 블록의 내용을 조작하는 것은 어렵다.

[0028] 블록체인은 분산형 원장 구조이며, 그 블록체인 네트워크에 참가한 모든 사람이 모든 거래내역을 기록한 원장을 소유한다. 그렇기 때문에 블록체인 기술을 활용하면 별도의 거래관리 기관 없이 분산화된 거래장부인 블록체인에 의해 작동되기 때문에 시스템 유지비용이 적고 해킹을 원천 차단할 수 있다는 장점이 있다. 분산 원장 환경에서는 사용자가 일정한 트랜잭션을 요청하면, 거래 정보가 기록된 하나의 블록을 생성하여 네트워크 상의 모든 참여자에게 블록을 전송한다. 이때 각 참여자가 전송된 블록을 승인하게 되면 기존 블록체인에 거래 기록이 추가되면서 거래가 완료된다. 예컨대, 기존 블록체인에 담겨 있는 거래정보를 수정하기 위해서는 전체 블록체인 네트워크 참여자의 과반수가 동일한 정보임을 확인해 주어야 하기 때문에 해커가 전 세계 네트워크 참여자의 블록체인을 동시에 해킹하는 것은 사실상 불가능하다.

[0029] 블록체인은 각 노드에서 만든 블록의 정당성을 검토하고 네트워크 전체에서 공유하는 블록체인에 반영하기 위해 합의 알고리즘을 사용한다. 블록체인에서 사용되고 있는 대표적인 합의 알고리즘은 PoW(Proof of Work), PoS(Proof of Stake), PBFT(Practical Byzantine Fault Tolerance) 등이 있다.

[0030] 블록체인은 결과적으로 거래로 이루어지는 변경이 불가능한 블록으로 형성된 분산, 분배 컴퓨터 시스템이다. 각 블록은 이전 블록의 해시를 포함하고, 따라서 블록은 그 시작부터 블록체인에 기록된 모든 거래의 기록을 생성하기 위해 서로 연결된다. 거래는 입력 및 출력에 포함된 스크립트로 알려진 작은 프로그램을 포함하며, 이는 어떻게 또는 누구에 의해 거래의 출력이 접근될 수 있는지를 특정한다.

[0031] 블록체인 1.0은 디지털 통화(Digital Currency)의 발행, 유통 및 거래가 주요 기능이였다. 블록체인 2.0은 기존 비트코인의 한계를 극복하고 다양한 영역으로의 확장을 목표로 하여 발전하고 있다. 블록체인 2.0의 대표적인 기술로는 이더리움(Ethereum)이 있으며, 디지털통화의 기능과 더불어 비트코인의 거래 스크립트를 다양한 형태로 프로그램 가능하게 만든 스마트 계약을 구현한다. 이하 설명하는 기술도 스마트 계약을 이용한다. 따라서 블록체인 구조는 이더리움에 기반한 것일 수 있다. 다만 이하 설명하는 기술에서 블록체인의 종류를 제한하지 않고, 스마트 계약 실행이 가능한 블록체인이면 충분하다.

[0033] 도 1은 블록체인 기반 테스트 관리 시스템(100)에 대한 예이다. 블록체인 기반 테스트 관리 시스템(100)은 클라이언트 장치(110), 테스트 데이터 관리 장치(150) 및 테스트 시스템(180)을 포함한다.

[0034] 도 1에서 클라이언트 장치(110)는 3개의 장치(A, B 및 C)를 예로 도시하였다. 클라이언트 장치(110)는 사용자가 사용하는 장치이다. 클라이언트 장치(110)는 테스트 데이터를 테스트 데이터 관리 장치(150)에 배포할 수 있다. 클라이언트 장치(110)는 테스트 데이터를 포함하는 프로젝트를 테스트 데이터 관리 장치(150)에 생성할 수 있다. 클라이언트 장치(110)는 테스트 데이터를 테스트 시스템(180)에 전달할 수 있다. 클라이언트 장치(110)는 테스트 결과 데이터를 테스트 시스템(180)으로부터 수신할 수 있다. 클라이언트 장치(110)는 결합 정보를 포함하는 테스트 결과 데이터를 테스트 데이터 관리 장치(150)에 배포할 수 있다.

[0035] 클라이언트 장치(110)는 스마트 계약을 이용하여 테스트 수행 및 관리를 할 수 있다. 예컨대, 클라이언트 장치(110)는 스마트 계약을 이용하여 테스트 데이터 배포, 프로젝트 생성, 테스트 데이터 전달, 테스트 데이터 관리

및 테스트 결과 데이터 배포 중 적어도 하나를 수행할 수 있다.

- [0036] 테스트 데이터 관리 장치(150)는 복수의 블록체인 노드로 구성된다. 복수의 블록체인 노드는 테스트 블록체인(155)을 구성한다. 테스트 블록체인(155)은 테스트 데이터, 프로젝트, 테스트 결과 데이터 등을 보유하고 관리한다. 블록체인 노드는 네트워크상의 객체이다. 예컨대, 블록체인 노드는 클라이언트 장치, PC, 서버 등과 같은 장치일 수 있다.
- [0038] 도 2는 테스트 데이터 관리 장치(200)의 구성에 대한 예이다. 테스트 데이터 관리 장치(200)는 도 1의 테스트 데이터 관리 장치(150)에 해당한다. 테스트 데이터 관리 장치(200)는 복수의 블록을 관리하는 가상 머신을 포함할 수 있다. 블록체인 네트워크의 노드들이 공유하는 하나의 가상머신은 일정한 합의 알고리즘에 따라 일정한 동작(상태 변경)을 수행할 수 있다. 예컨대, 이더리움 경우 블록체인이 이더리움 가상머신(Ethereum Virtual Machine, EVM)을 구성한다. EVM은 이더리움 스마트 계약의 런타임 환경이라고 할 수 있다. EVM은 블록이 보유한 스마트 계약을 실행할 수 있다. 스마트 계약은 테스트 데이터 관리 및 테스트 수행을 담당한다. 클라이언트 장치가 스마트 계약 실행을 위한 입력 데이터(변수)를 블록에 배포하면, EVM은 입력 데이터를 함수 형태의 스마트 계약에 입력하여 스마트 계약을 실행한다. 구체적인 스마트 계약 및 프로세스에 대해서는 후술한다.
- [0039] 다양한 소스 코드 및 애플리케이션을 통하여 블록 체인을 구성할 수 있다. 예컨대, 프론트 엔드(Front-end)는 HTML, CSS, Javascript를 이용하여 인터페이스(220)를 구현할 수 있다. 백 엔드(Back-end)는 jQuery, Mist, geth(go-ethereum), brower-solidity를 활용할 수 있다. 도 2는 클라이언트 장치가 웹 브라우저를 통해 필요한 데이터 및 명령을 블록체인 내지 EVM에 전달할 수 있다. 클라이언트 장치는 자바스크립트로 구현된 Node.js console을 통해 일정한 명령 및 데이터를 블록체인 내지 EVM에 전달할 수도 있다. 블록체인 시스템의 연동은 Web3.js 모듈로 구성될 수 있다.
- [0041] 도 3은 테스트 데이터 관리 장치(200)에서 테스트 데이터를 전달하는 과정에 대한 예이다. 클라이언트 장치는 스마트 계약을 통해 테스트 데이터를 테스트 시스템(180)에 전달할 수 있다. 예컨대, 블록체인 기반의 테스트 데이터 관리 장치는 testRPC 기반의 테스트 넷 환경에서 제어될 수 있다. 블록체인 시스템의 연동은 Web3.js 모듈로 구현될 수 있다. 예컨대, 블록체인 내지 EVM(210)은 테스트 데이터를 데이터 서버(250)에 전송하고, 이후 URL 형식으로 테스트 시스템(180)에 테스트 데이터를 전달할 수 있다.
- [0042] 가상 머신 EVM은 블록들을 관리하며 데이터 처리를 위해 Solidity언어로 작성된 스마트 계약을 활용할 수 있다. 스마트 계약은 프로젝트, 결함, 테스트케이스 기능을 수행하기 위해 각각의 계약이 존재하며 테스트 데이터를 저장하기 위한 구조체와 구조체를 배열을 통해 관리하기 위해 구조체 배열이 선언된다.
- [0043] 계약은 테스트 데이터를 저장과 호출을 하기 위한 기능을 구현한다. 사용자가 프로젝트, 결함, 테스트 케이스 데이터를 추가할 경우 트랜잭션 생성 코드가 실행되면서 저장하고자 하는 테스트 데이터의 트랜잭션이 생성된다. 생성된 트랜잭션은 사용자의 채굴을 통해 하나의 블록으로 생성되며 생성된 블록의 헤더(Header)의 속성인 이전블록해시(Prev Block Hash)에 이전 블록의 해시 값이 저장되어 현재 채굴된 블록과 이전 블록이 연결된다. 블록 내의 트랜잭션으로 저장된 데이터를 조작할 경우 헤더의 머클해시(Merkle Hash)값이 변경되어 블록 해시 값에도 영향으로 주어 이전 데이터에 대한 위/변조를 방지할 수 있다. 생성된 트랜잭션은 사용자의 채굴에 의해 블록으로 채굴되어 프로젝트에 참여하는 사용자들에게 전송되어 관리된다.
- [0045] 도 4는 블록체인에서 관리하는 구조체 배열에 대한 예이다. 사용자는 스마트 계약을 통해 프로젝트(Project) 생성, 테스트 케이스(Testcase, TC) 생성 및 결함 정보(Defect)를 블록체인에 생성(배포)할 수 있다. 프로젝트 생성, 테스트 케이스 생성 및 결함 정보 생성은 각각의 스마트 계약을 통해 수행될 수 있다.
- [0046] 프로젝트 구조체 배열(310)은 프로젝트 명 내지 식별자(pj_title), 시작시점(pj_startDate), 종료시점(pj_endDate) 및 설명(pj_text) 중 적어도 하나를 포함할 수 있다. 클라이언트 장치는 해당 구조체 배열의 항목(변수)을 블록체인에 전달하여 스마트 계약을 실행하고, 프로젝트를 블록체인에 생성한다. 한편 프로젝트 추가도 가능하다. 프로젝트 추가를 위한 별도의 스마트 계약이 존재할 수 있다.
- [0047] 클라이언트 장치는 프로젝트에 대한 블록 데이터를 통해 프로젝트 리스트 확인이 가능하다. 클라이언트 장치는 프로젝트 리스트에 프로젝트 명 내지 식별자, 시작일, 종료일, 설명이 표시, 프로젝트 추가 시 프로젝트 명, 시

작일, 종료일, 설명 추가가 가능하다.

- [0048] 테스트 케이스 구조체 배열(320)은 프로젝트 명(tc_project), 프로젝트 번호(tc_num), 프로젝트 생성한 사용자 주소(tc_user), 테스트 케이스 경로(tc_path), 테스트 케이스 내용(tc_content), 테스트 데이터(tc_testdata), 테스트 사전 조건(tc_precondition), 테스트 과정(tc_process) 및 테스트 결과 내지 예상 결과(tc_result) 중 적어도 하나를 포함할 수 있다. 클라이언트 장치는 해당 구조체 배열의 항목(변수)을 블록체인에 전달하여 스마트 계약을 실행하고, 테스트 케이스를 블록체인에 생성한다.
- [0049] 클라이언트 장치는 테스트 데이터 또는 테스트 케이스를 테스트 시스템에 전달할 수 있다. 한편, 클라이언트 장치는 데이터 전달을 위한 별도의 스마트 계약을 사용하여 테스트 데이터 또는 테스트 케이스를 테스트 시스템에 전달할 수 있다.
- [0050] 클라이언트 장치는 테스트 데이터에 대한 블록 데이터를 통해 테스트 케이스 리스트 확인이 가능하다. 한편 테스트 케이스 추가도 가능하다. 테스트 케이스 추가를 위한 별도의 스마트 계약이 존재할 수 있다. 클라이언트 장치는 테스트 케이스 리스트에 테스트 케이스 명 내지 식별자, 테스트 케이스 경로, 테스트 케이스 내용, 테스트 데이터, 테스트 사전 조건, 테스트 과정 및 테스트 결과 내지 예상 결과 및 상태 중 적어도 하나가 추가 가능하다. 상태는 통과, 실패, N/A 또는 N/T 등의 상태가 있을 수 있다.
- [0051] 클라이언트 장치는 테스트 시스템으로부터 테스트 결과를 전달받을 수 있다. 결함 정보 구조체 배열(330)은 관련 프로젝트(df_projec), 사용자 주소(tc_user), 결함 정보 경로(df_path), 제목(title), 결함 내용(df_content), 심각도(df_severity), 우선순위(df_priority), 재현빈도(df_representation), 조건(df_condition), 결함 정보 생성시점(df_startDate) 및 결함 정조 생성 종료시점(df_endDate) 중 적어도 하나를 포함할 수 있다. 클라이언트 장치는 해당 구조체 배열의 항목(변수)을 블록체인에 전달하여 스마트 계약을 실행하고, 결함 정보를 블록체인에 생성한다.
- [0052] 클라이언트 장치는 결함 정보에 대한 블록 데이터를 통해 결함 정보 리스트 확인이 가능하다. 한편 결함 정보 추가도 가능하다. 결함 정보 추가를 위한 별도의 스마트 계약이 존재할 수 있다. 클라이언트 장치는 결함 정보 리스트에 프로젝트, 사용자 주소, 결함 정보 경로, 제목, 결함 내용, 심각도, 우선순위, 재현빈도, 조건, 결함 정보 생성시점 및 결함 정조 생성 종료시점 중 적어도 하나를 포함할 수 있다.
- [0053] 예컨대, 심각도는 치명적, 매우 심각, 심각, 보통 및 경미 중 어느 하나와 같이 결정될 수 있다. 우선순위는 즉시해결, 주의요망, 대기 및 낮은 순위 중 어느 하나로 결정될 수 있다. 재현빈도는 신규, 개선, 반려, 수정완료, 재개설 및 완료 중 어느 하나로 결정될 수 있다.
- [0055] 도 5는 블록체인 노드 장치(400)의 구성에 대한 예이다. 도 5에 도시한 각 구성은 아날로그 회로 또는 디지털 회로로 구현될 수 있다. 노드 장치(400)는 저장 장치(410), 메모리(420), 연산장치(430), 인터페이스 장치(440) 및 통신장치(450)를 포함한다.
- [0056] 통신장치(450)는 유선 또는 무선 네트워크를 통해 일정한 정보를 수신하고 전송하는 구성을 의미한다. 통신 장치(450)는 클라이언트 장치 또는 테스트 시스템으로부터 블록데이터 구성을 위한 정보를 수신할 수 있다. 통신 장치(450)는 클라이언트 장치로부터 테스트 데이터를 항목으로 포함하는 제1 스마트 계약을 수신할 수 있다. 제1 스마트 계약은 테스트 케이스 생성을 위한 스마트 계약이다. 테스트 케이스 생성을 위한 항목은 도 4에서 설명한 바와 같다. 연산장치(430)는 제1 스마트 계약이 사전에 정해진 항목을 포함하는 경우에만 제1 스마트 계약을 실행하여 테스트 케이스를 생성한다. 즉 블록체인 노드 장치(400) 또는 전송한 테스트 데이터 관리 장치는 정해진 포맷에 맞는 데이터가 전달되는 경우에만 블록체인에 데이터를 배포한다.
- [0057] 통신장치(450)는 클라이언트 장치로부터 제2 스마트 계약을 더 수신하고, 테스트 데이터 또는 테스트 케이스를 테스트 시스템에 송신할 수 있다. 제2 스마트 계약은 블록체인에 저장된 테스트 데이터 또는 테스트 케이스를 전달한다. 연산장치(430)는 제2 스마트 계약이 사전에 정해진 항목을 포함하는 경우에만 제2 스마트 계약을 실행하여 테스트 데이터 또는 테스트 케이스를 테스트 시스템에 전달한다.
- [0058] 통신 장치는 클라이언트 장치로부터 프로젝트 식별자, 프로젝트 시작시점, 프로젝트 종료시점 및 프로젝트 설명을 포함하는 제3 스마트 계약을 수신할 수 있다. 제3 스마트 계약은 프로젝트를 생성하는 계약이다. 연산장치(430)는 제3 스마트 계약을 실행하여 테스트 케이스에 대한 프로젝트를 생성한다. 연산장치(430)는 제3 스마트 계약이 사전에 정해진 항목을 포함하는 경우에만 제3 스마트 계약을 실행하여 프로젝트를 생성한다.

- [0059] 나아가 통신장치(450)는 스마트 계약 실행을 위한 입력 변수를 다른 노드에 송신할 수 있다. 통신장치(450)는 생성한 스마트 계약을 다른 노드에 송신할 수 있다.
- [0060] 통신장치(450)는 테스트 시스템이 테스트 데이터 또는 테스트 케이스를 입력받아 실행한 결과 데이터를 포함하는 제4 스마트 계약을 수신할 수 있다. 연산장치(430)는 제4 스마트 계약을 실행하여 테스트 케이스에 대한 결함 정보를 생성한다.
- [0061] 저장 장치(410)는 노드 장치(400)의 동작을 제어하는 프로그램(코드)을 저장할 수 있다. 저장 장치(410)는 테스트 데이터, 테스트 케이스, 프로젝트 및 결함 정보 중 적어도 하나를 저장할 수 있다. 저장 장치(410)는 전술한 제1 스마트 계약, 제2 스마트 계약, 제3 스마트 계약 및 제4 스마트 계약 실행을 위한 코드(함수)를 저장할 수 있다.
- [0062] 메모리(420)는 노드 장치(400)가 수신한 정보 또는 연산 장치(430)의 동작에 따라 임시로 발생하는 데이터를 저장할 수 있다.
- [0063] 인터페이스 장치(440)는 외부 사용자로부터 일정한 명령을 입력받는 장치이다. 인터페이스 장치(440)는 물리적으로 연결된 입력 장치 또는 외부 저장 장치로부터 노드 장치(400) 동작에 기본적으로 필요한 프로그램 내지 데이터를 입력받을 수 있다. 인터페이스 장치(440)는 스마트 계약을 구성하는 코드를 입력받을 수 있다.
- [0064] 통신 장치(450) 내지 인터페이스 장치(440)는 외부로부터 일정한 데이터 내지 명령을 전달받는 장치이다. 통신 장치(450) 내지 인터페이스 장치(440)를 입력장치라고 명명할 수도 있다.
- [0065] 전술한 바와 같이 연산 장치(430)는 스마트 계약 내지 스마트 계약을 구성하는 항목을 입력받아 스마트 계약을 실행한다. 연산 장치(430)는 데이터를 처리하고, 일정한 연산을 처리하는 프로세서, AP, 프로그램이 임베디드된 칩과 같은 장치일 수 있다.
- [0067] 또한, 상술한 바와 같은 기반 테스트 데이터 관리 방법, 테스트 케이스 생성 방법 및 결함 정보 생성 방법 각각은 컴퓨터에서 실행될 수 있는 실행가능한 알고리즘을 포함하는 프로그램(또는 어플리케이션)으로 구현될 수 있다. 상기 프로그램은 비일시적 판독 가능 매체(non-transitory computer readable medium)에 저장되어 제공될 수 있다.
- [0068] 비일시적 판독 가능 매체란 레지스터, 캐쉬, 메모리 등과 같이 짧은 순간 동안 데이터를 저장하는 매체가 아니라 반영구적으로 데이터를 저장하며, 기기에 의해 판독(reading)이 가능한 매체를 의미한다. 구체적으로는, 상술한 다양한 어플리케이션 또는 프로그램들은 CD, DVD, 하드 디스크, 블루레이 디스크, USB, 메모리카드, ROM 등과 같은 비일시적 판독 가능 매체에 저장되어 제공될 수 있다.
- [0070] 본 실시례 및 본 명세서에 첨부된 도면은 전술한 기술에 포함되는 기술적 사상의 일부를 명확하게 나타내고 있는 것에 불과하며, 전술한 기술의 명세서 및 도면에 포함된 기술적 사상의 범위 내에서 당업자가 용이하게 유추할 수 있는 변형 예와 구체적인 실시례는 모두 전술한 기술의 권리범위에 포함되는 것이 자명하다고 할 것이다.

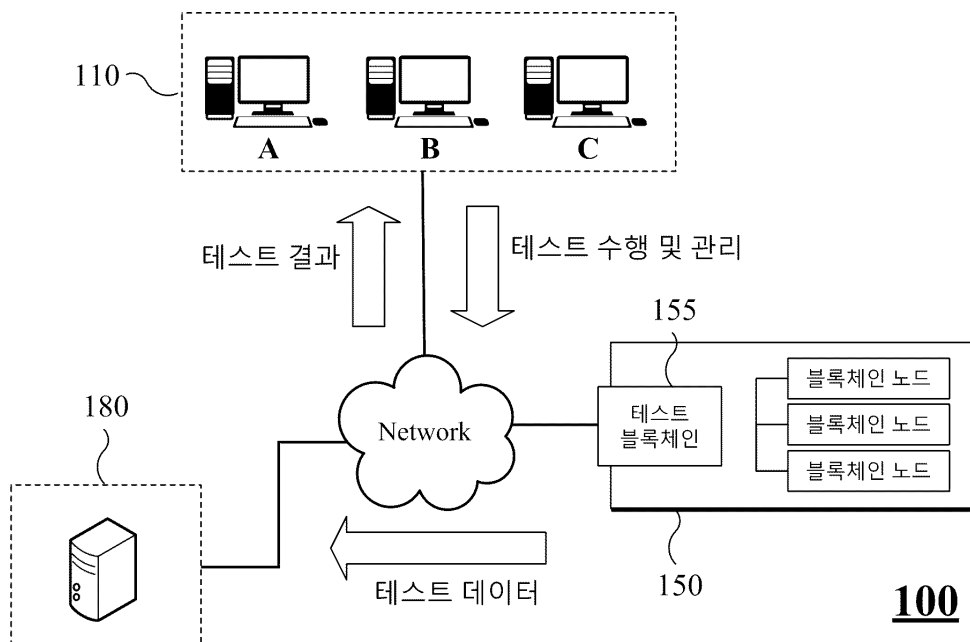
부호의 설명

- [0071] 100 : 블록체인 기반 테스트 관리 시스템
- 110 : 클라이언트 장치
- 150 : 테스트 데이터 관리 장치
- 154 : 테스트 블록체인
- 180 : 테스트 시스템
- 200 : 테스트 데이터 관리 장치
- 210 : 가상 머신
- 220 : 인터페이스부

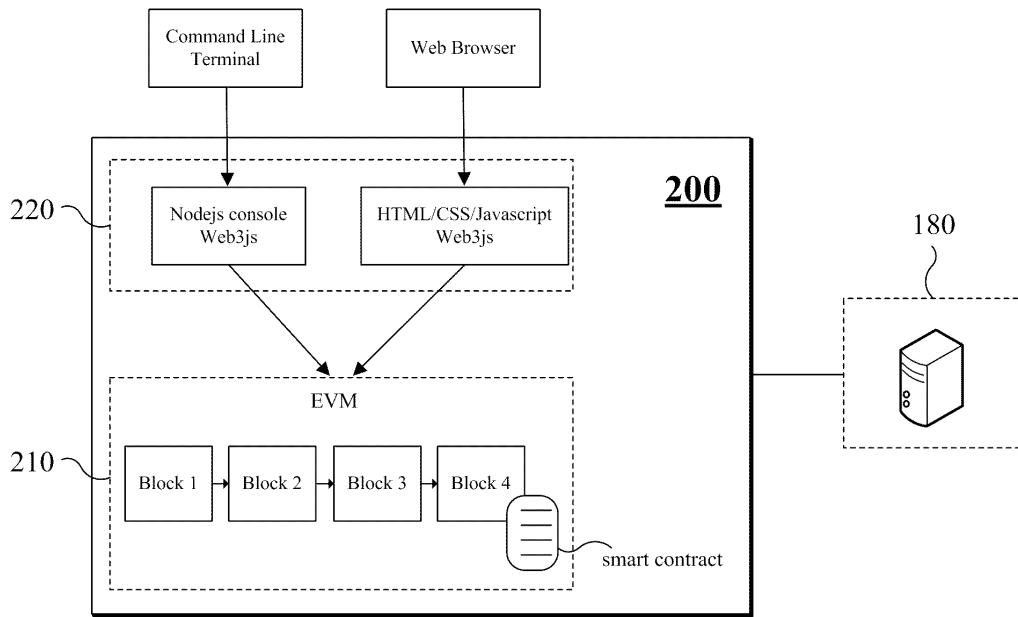
- 250 : 데이터 서버
- 400 : 테스트 데이터 관리 장치
- 410 : 저장장치
- 420 : 메모리
- 430 : 연산장치
- 440 : 인터페이스 장치
- 450 : 통신장치

도면

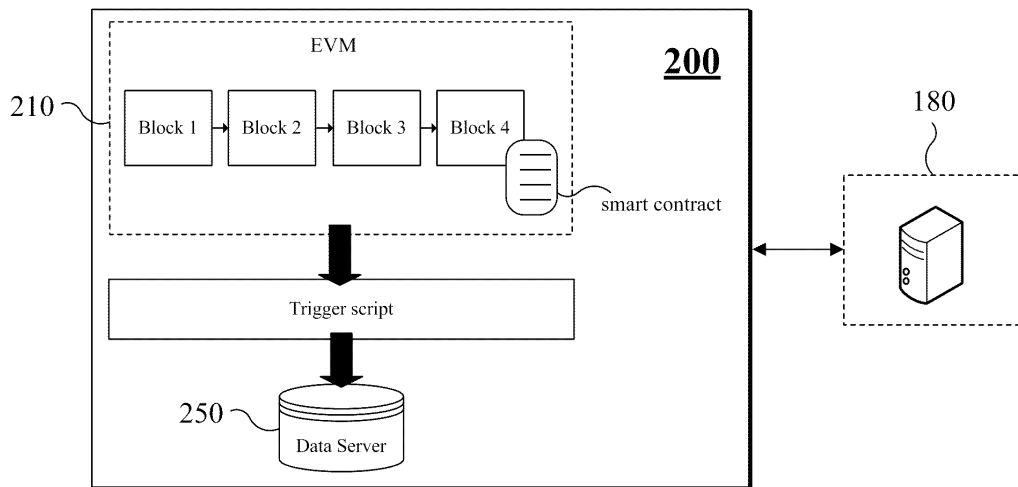
도면1



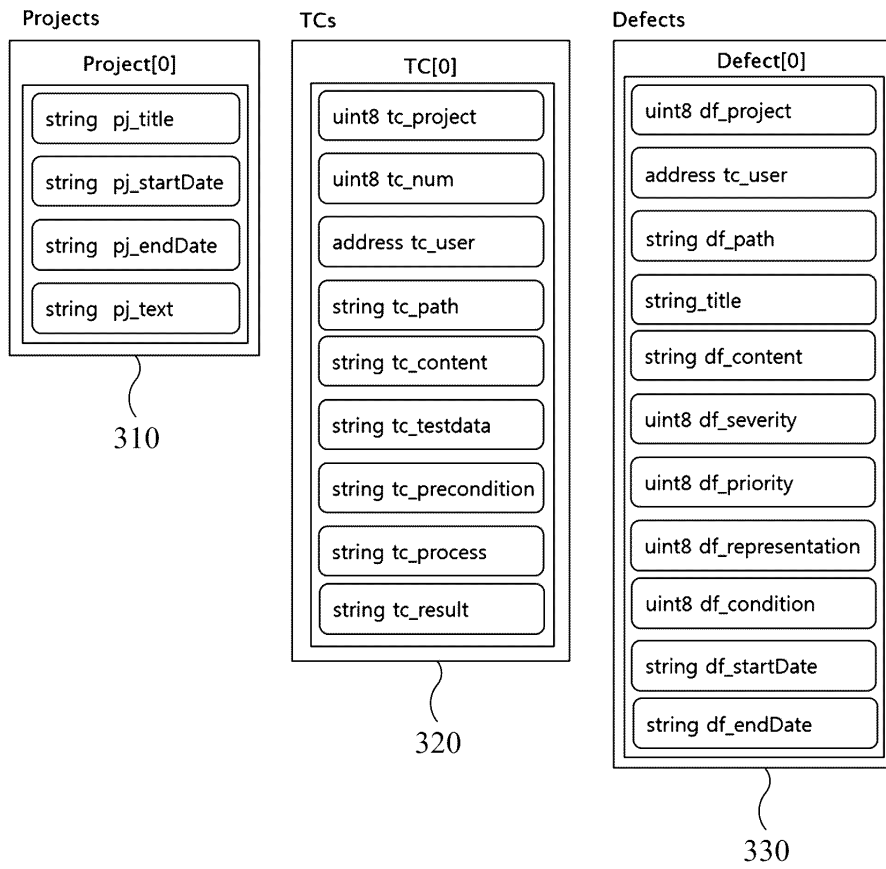
도면2



도면3



도면4



도면5

