creative commons
COMMONS DEED

저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:

저작자표시. 귀하는 원저작자를 표시하여야 합니다.

비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.

변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 이용허락규약(Legal Code)을 이해하기 쉽게 요약한 것입니다.

Disclaimer

dCollection

공학석사 학위논문

신뢰성 있는 작업자 안전관리 프로세스를 제공하는 블록체인 기반 서비스

A Blockchain-based Service Supporting Reliable Worker Safety Management Processes

울 산 대 학 교 대 학 원

전기전자컴퓨터공학과
정보통신공학전공

조 연 정

신뢰성 있는 작업자 안전관리 프로세스를 제공하는 블록체인 기반 서비스

지 도 교 수 이 명 준

2022년 6월

울 산 대 학 교 대 학 원

전기전자컴퓨터공학과
정보통신공학전공

조 연 정

조 연 정의 공학석사 학위논문을 인준함

심사위원 윤 석 훈

심사위원 이 명 준

심사위원 조 동 식

울 산 대 학 교 대 학 원

2022년 6월

[감사의 글]

먼저 석사 과정 동안 부족한 저에게 따뜻한 격려와 조언을 아끼지 않으셨던 이명준 교수님, 박양수 교수님께 감사드립니다. 석사라는 길을 생각지도 못했던 저에게 새로운 기회에 도전하게 해주신 이명준 교수님께 다시 한번 감사드립니다. 교수님의 지도로 석사 과정 동안 수없이 마주친 터널들을 빠져나올 수 있었고 그 덕분에 많이 성장하여 무사히 졸업의 문을 열 수 있었습니다. 또한, 바쁘신 중에도 논문 심사를 맡아주시고 조언을 해주신 윤석훈 교수님, 조동식 교수님께도 정말 감사드립니다.

그 동안 저와 함께 연구실 생활을 하고 함께 연구했던 모든 선후배님께도 감사의 말씀을 드립니다. 졸업 작품 때부터 많은 도움을 받았고 석사의 길까지 도전해볼 수 있게 도와주신 진재환 선배님, 이우건 선배님, 갑작스레 랩장이 되어 힘들어 보였지만 항상 고민도 잘 들어주고 힘이 되었던 엄현민 오빠, 밤새며 연구하고 야식을 먹으며 함께 즐거웠던 윤여국 오빠, 밤새 서로의 연구를 도우며 같이 의지하고 힘든 것도 함께 이겨낸 룸메이트이고 친구이자 이젠 가족 같은 심채린, 나보다 늦게 들어왔지만 먼저 졸업했던 꼼꼼해서 물어볼 것이 많았던 권민호 오빠, 같이 연구실 생활은 못했지만 항상 함께여서 힘이 되었던 김혜원, 이영은 모두 정말 감사드립니다. 연구실에 처음 개인 자리가 생겼을 때부터, 제주학회, 홈커밍데이 등 프로그래밍 시스템 연구실에 들어와 겪었던 모든 경험들은 모두 잊지 못할 추억이 되었습니다.

그리고 내가 대학원간다고 했을 때 부담스러웠겠지만 나를 믿어주고 응원해주신 우리 부모님께 감사의 말씀을 드립니다. 항상 내가 가는 길을 지지해주고 내 얘기를 들으며 공감해주는 든든한 우리 아빠, 본집가면 맛있는 음식으로 따뜻하게 반겨주고 매번 먼저 연락해 내 걱정하는 우리 엄마, 아닌 척 하지만 항상 서로 의지하고 있는 내 동생 다정이, 우리 가족 언제나 건강하길 바라고 사랑합니다. 지금도 남들과는 조금 다른 길을 결정하여 졸업도 늦어졌고 불안정하여 걱정이 많으시겠지만, 꼭 늦지 않게 성공해서 사랑에 보답하겠습니다. 가족들에게 다시 한 번 감사하고 사랑한다는 말을 전합니다.

마지막으로 석사 과정을 무사히 마칠 수 있도록 도와주신 모든 분들게 다시 한 번 감사드리고 사랑합니다.

[국문요약]

신뢰성 있는 작업자 안전관리 프로세스를 제공하는 블록체인 기반 서비스

현장의 상황 정보와 작업자의 건강 상태가 작업자의 안전에 영향을 크게 미치는 작업 현장에서는 일반적으로 작업자 안전관리 시스템을 사용하는 것이 강조되고 있다. 또한, 이러한 정보를 바탕으로 관리감독자들이 확인하는 일련의 절차도 작업자 안전관리를 위하여 매우 중요한 요소이다. 하지만 기존의 작업자 안전관리 시스템에서는 작업자의 안전과 관련된 민감한 상황정보가 기업 내부의 단일 서버에서만 저장·관리되는 관계로 데이터의 위변조 및 유실 가능성이 높아 작업자로부터 신뢰를 받지 못하고 있는 실정이다. 따라서 작업 현장 및 작업자 상태 등의 정보와 안전관리의 절차도 신뢰성 있게 저장·관리되는 방식이 필요하다. 또한, 기업 및 작업자에게 이러한 정보와 절차가 투명하게 제공되도록 보장되는 것이 바람직하다.

본 논문에서는 작업자 안전관리 프로세스를 신뢰성 있게 제공하는 블록체인 기반 서비스인 FairSafe의 개발에 대하여 기술한다. FairSafe는 EOS 블록체인과의 통신을 통한 스마트 컨트랙트의 기능 수행을 지원하는 안전관리 플랫폼과 각 사용자의 기능을 수행하기 위한 3가지의 분산 애플리케이션으로 구성된다. 운영자 애플리케이션은 작업자 및 관리감독자에게 기능 수행에 적합한 어카운트 권한을 부여하고 작업 현장상태의 안전 기준치를 설정한다. 작업자 및 관리감독자 애플리케이션은 작업개시 요청 및 승인부터 종료까지의 모든 일련의 절차를 블록체인 트랜잭션으로 수행하기 위한 기능을 제공한다. 이들 어플리케이션을 통하여 수행되는 안전관리 프로세스와 작업장의 현장정보 및 작업자 상태 등의 정보는 안전관리 플랫폼을 통하여 EOS 스마트 컨트랙트에 명시적으로 저장되어 관리된다.

주요어 : 작업자 안전관리 시스템, 블록체인, EOS, 분산 애플리케이션, FairSafe

목 차

요약 ····· i
목차 ····· ii
그림 목차 ····· iv
표 목차 ····· v

1. 서 론 ····· 1
1.1. 연구의 배경 및 필요성 ····· 1
1.2. 연구의 개요 ····· 2
1.3. 논문의 구성 ····· 3

2. 관련 연구 ····· 4
2.1 EOS 블록체인 ····· 4
2.2 EOS 어카운트 ····· 5
2.3 EOS 스마트 컨트랙트 ····· 6
2.4 작업자 안전관리 시스템 ····· 7

3. FairSafe의 설계 ····· 8
3.1 FairSafe의 구조 ····· 8
3.2 작업자 및 관리감독자의 어카운트 권한 등록 프로세스 ····· 9
3.3 FairSafe에서의 투명한 작업자 안전관리 프로세스 ····· 11
3.3.1 작업자의 작업개시 및 종료 요청 ····· 11
3.3.2 관리감독자의 작업자 요청에 대한 승인 처리 ····· 13
3.3.3 요청 및 승인에 대한 취소 처리 ····· 15
3.4 작업자 안전관리 플랫폼 ····· 17

4. FairSafe의 구현 ····· 18
4.1 FairSafe의 스마트 컨트랙트 ····· 18
4.2 REST API 기반 안전관리 플랫폼 ····· 21
4.2.1 상황정보관리 API ····· 21
4.2.2 작업안전관리 API ····· 22
4.2.3 트랜잭션 전송 및 관리 API ····· 23
4.3 FairSafe에서 동작하는 운영자 애플리케이션 ····· 24
4.3.1 운영자 애플리케이션의 기능 ····· 24
4.3.2 운영자 애플리케이션의 동작화면 ····· 24
4.4 FairSafe에서 동작하는 작업자 및 관리감독자 애플리케이션 ····· 26
4.4.1 작업자 및 관리감독자 애플리케이션의 기능 ····· 26
4.4.2 작업자 및 관리감독자 애플리케이션의 동작화면 ····· 26
4.5 FairSafe 검증 ····· 29

5. 결론 ····· 32

[참고문헌] ····· 33

[Abstract] ····· 35

그 림 목 차

[그림 1] EOSIO의 주요 구성 컴포넌트와 이들 간의 관계 ····· 5
[그림 2] EOS 어카운트 권한 구조 ····· 7
[그림 3] FairSafe의 전체 구조 ····· 8
[그림 4] 작업자 및 관리감독자의 어카운트 권한 등록 프로세스 ····· 10
[그림 5] 신뢰성 있는 작업자의 작업 개시 요청에 대한 프로세스 ····· 12
[그림 6] 관리감독자의 작업자 요청에 대한 승인 처리 프로세스 ····· 13
[그림 7] FairSafe에서의 작업자 state 흐름도 ····· 14
[그림 8] 안전관리 플랫폼과 상호작용하는 구성 요소들 ····· 16
[그림 9] FairSafe의 스마트 컨트랙트 구조 ····· 17
[그림 10] FairSafe에서 블록체인으로 트랜잭션을 전송하는 프로세스 ····· 20
[그림 11] 작업장 및 섹션 관리 화면 ····· 21
[그림 12] 작업장 및 섹션 등록과 안전 기준치 설정 화면 ····· 24
[그림 13] 작업자와 관리감독자의 회원가입 승인 화면 및 회원 정보 확인 화면 ···· 26
[그림 14] 블록생성 제어 화면 ····· 30
[그림 15] 작업자 및 관리감독자의 개인키가 저장된 지갑 관리 화면과 회원가입 화면 ····· 31
[그림 16] 작업자가 작업 개시에 대한 요청을 진행하는 화면들 ····· 33
[그림 17] 작업장 및 섹션의 상태 관리 화면 ····· 34
[그림 18] 메인화면의 현재 상태 화면과 전체 기록 화면 ····· 34
[그림 19] 작업자 상태에 따른 리스트와 작업 요청 승인 화면 ····· 35
[그림 20] 검증 시나리오 중 데이터 일치성 확인 화면 ····· 36

표 목 차

[표 1] 작업현장 관리 관련 컨트랙트 및 주요 액션 ····· 22
[표 2] 투명한 작업자 안전관리 프로세스 관련 컨트랙트 및 주요 액션 ····· 23
[표 3] 컨트랙트의 코드 구조 ····· 25
[표 4] iotlog 컨트랙트에 저장된 데이터를 조회하는 주요 코드 ····· 26
[표 5] 작업자 안전관리 프로세스를 지원하기 위한 주요 API ····· 27
[표 6] 운영자 관련 주요 API ····· 28

1. 서론

1.1 연구의 배경 및 필요성

현장의 상황 정보가 작업자의 안전에 큰 영향을 미치는 작업 현장에서는 작업자의 안전을 확보하기 위하여 작업자 안전관리 시스템의 사용이 강조되고 있다[1-4]. 작업자 안전관리 시스템이란 현장 정보와 작업자의 생체 데이터를 실시간으로 확인함으로써 관리감독자가 작업자의 안전사고에 대한 예방적 조치 및 신속한 대처를 가능하게 하는 시스템을 말한다. 일반적으로 작업자 안전관리 시스템은 기업의 내부 단일 서버 및 데이터베이스를 기반으로 한 시스템으로 높은 비용이 요구되는 고성능 중앙 서버와 고가의 보안 장비가 필수적이다. 또한, 데이터의 위변조 및 유실 가능성이 높아 시스템의 신뢰도를 얻기 어려운 실정이다. 특히 기존의 시스템은 작업자가 직접 작업 안전과 관련된 데이터들을 확인하기 어려우며, 안전사고가 발생할 경우 데이터의 은폐 및 조작 등에 대한 작업자의 불신과 책임의 소재에 대한 논란을 야기할 수 있다.

작업자의 안전관리를 위하여 관리감독자가 현장 상태 데이터 및 작업자의 생체 데이터를 실시간으로 확인하는 것은 작업자들의 안전관리에 있어 중요한 프로세스이다. 이 뿐만 아니라, 작업자가 스스로 작업 현장 정보와 작업자 상태를 투명하게 확인하고, 작업 개시 및 종료에 대하여 관리감독자와 상호작용하는 프로세스 또한 매우 중요하다. 이를 위하여, 작업자의 안전을 관리하기 위한 프로세스와 이에 관한 데이터를 신뢰성 있게 제공 및 관리되는 방안과 이러한 절차를 포함한 모든 정보가 기업 및 작업자에게 투명하게 제공되는 방안이 필요한 실정이다.

최근 4차 산업혁명 기술이 빠르게 발전함에 따라 데이터의 투명성과 신뢰성을 위해 탈중앙화 저장 기술인 블록체인을 이용한 시스템이 많이 개발되고 있다[5-7]. 2세대 블록체인 기술인 이더리움의 한계점을 개선하기 위하여 등장한 3세대 블록체인인 EOS는 빠른 트랜잭션 처리 속도와 수수료가 없다는 큰 특징을 가진다[8,9]. 이는 수많은 데이터들을 다루는 사물인터넷 환경에 적합한 기술로, EOS 블록체인 기술을 사용하여 모든 정보와 절차를 투명하고 신뢰성 있게 제공하는 작업자 안전관리 시스템을 개발하는 것이 바람직하다.

1.2 연구의 개요

본 논문에서는 투명하고 신뢰성 있는 작업자 안전관리 프로세스를 제공하는 블록체인 기반 서비스인 FairSafe의 개발에 대하여 기술한다. FairSafe는 작업 현장 내의 센서를 통해 받은 상황정보 데이터와 작업자의 신체 데이터를 블록체인 네트워크에 저장하고, 이를 기반으로 작업자들의 안전을 관리한다. 또한, 작업 현장에서 작업 개시 및 종료를 위하여 작업자와 관리감독자가 진행하는 프로세스에 대한 기록도 블록체인에 저장하여 관리한다. 본 논문에서는 FairSafe의 설계부터 EOS 블록체인과의 통신을 통한 스마트 컨트랙트의 기능 수행을 지원하는 안전관리 플랫폼과 각 사용자의 기능을 수행하기 위한 3가지의 분산 애플리케이션의 개발에 대하여 기술한다.

구현된 FairSafe는 작업 현장에 설치된 센서 플랫폼을 통하여 받은 작업 현장의 상황정보 데이터와 작업자의 상태정보 등의 작업자의 안전관리에 대한 데이터들을 EOS 스마트 컨트랙트를 이용하여 저장 및 관리한다. 이를 위하여, 본 논문에서는 작업자와 관리감독자가 신뢰성 있는 작업개시 및 종료를 진행하기 위한 프로세스와 이를 진행하는 작업자와 관리감독자의 명시적인 인증을 위하여 EOS 어카운트 관리 도구인 AccountEOS를[10] 통해 운영자로부터 어카운트의 권한을 등록하는 기법을 제안한다. 또한, 스마트 팩토리에서 작업 현장의 상황정보 데이터를 블록체인에 기록하기 위한 기법을 제안한다. FairSafe는 작업진행 프로세스 전 과정이 블록체인에 저장되어 신뢰성을 보장할 수 있으며, 안전사고가 발생할 경우 트랜잭션 히스토리를 통하여 책임소재를 분명히 할 수 있다는 장점을 가진다.

이와 더불어, 구현된 FairSafe는 권한을 관리하는 운영자 및 제안된 프로세스를 진행하는 작업자와 관리감독자의 각 역할에 따라 기능을 수행하는 3가지의 분산 애플리케이션으로 구성된다. 운영자 애플리케이션은 어카운트 권한을 부여하고 작업자의 안전관리에 대한 모든 데이터들을 안전 기준치에 따라 상, 하한선을 설정한다. 권한이 등록된 작업자와 관리감독자의 어카운트로 작업자 및 관리감독자 애플리케이션이 동작하고, 작업자 애플리케이션은 작업 개시 및 종료를 위한 요청을 하고 관리감독자 애플리케이션은 이에 따른 요청 승인 작업을 진행한다. 이를 통하여 관리감독자 측에서만 정보를 확인할 수 있는 것이 아니라 작업자도 애플리케이션을 통하여 진행사항을 투명하게 모니터링 할 수 있는 기능을 제공한다.

본 논문에서는 3개의 노드로 EOS 프라이빗 블록체인 네트워크를 구성하고, 센서플랫폼과 안전관리 플랫폼 간 데이터 일치성 및 분산 애플리케이션의 UI별 기능과 블록체인 네트워크에 저장된 데이터 일치성을 통해 FairSafe를 검증한다.

1.3 논문의 구성

논문의 구성은 다음과 같다. 다음 장에서는 배경지식으로 EOS 블록체인 분산 애플리케이션과 작업자 안전관리 시스템에 대해서 다루고 3장에서는 FairSafe의 설계에 대하여 설명한다. 4장에서는 FairSafe의 구현에 대하여 설명하고 이를 기반으로 동작하는 3가지의 분산 애플리케이션에 대하여 기술한다. 마지막으로 5장에서는 결론을 제시한다.

2. 관련 연구

본 장에서는 EOS 블록체인을 설명하기 위하여 개념과 EOS 어카운트 및 스마트 컨트랙트에 대하여 기술한다. 또한, 일반적으로 사용되는 작업자 안전관리 시스템에 대하여 기술한다.

2.1 EOS 블록체인

EOS는 2세대 블록체인 기술인 이더리움의 한계점을 개선하기 위하여 등장한 3세대 블록체인 기술로서, DPoS(Delegated Proof Of Stake) 합의 알고리즘을[11,12] 기반으로 하여 분산 애플리케이션(Decentralized Applicaion, DApp) 구동에 특화된 고성능 플랫폼이다. DPoS(Delegated Proof of Stake)는 기존 PoS(Proof of Stake) 합의 알고리즘에서 합의를 위한 권한을 대표자에게 위임할 수 있는 알고리즘으로, EOS에서는 21명의 BP(슈퍼노드)를 선출하여 블록 생성 및 네트워크에 대한 영향력을 행사할 수 있게 한다. EOS는 블록체인 오픈소스 소프트웨어인 EOSIO를 사용하여 전 세계 커뮤니티들에 의해서 구축된 최초의 EOSIO[13] 블록체인 네트워크로, DPoS 합의 알고리즘을 활용하는 EOSIO 블록체인 네트워크 중 가장 대표적이고 성능이 뛰어난 퍼블릭 블록체인이다.

EOSIO는 블록체인 오픈소스 소프트웨어로, EOSIO 기반 블록체인들은 서로 호환성과 유연성을 가진다. EOSIO는 블록체인 노드를 운영하고 블록체인 노드와 상호 작용하고 스마트 컨트랙트를 배포하는 등의 구성 요소를 라이브러리로 제공한다. 그림 1은 EOSIO의 주요 구성 컴포넌트와 이들 간의 관계를 나타낸다.

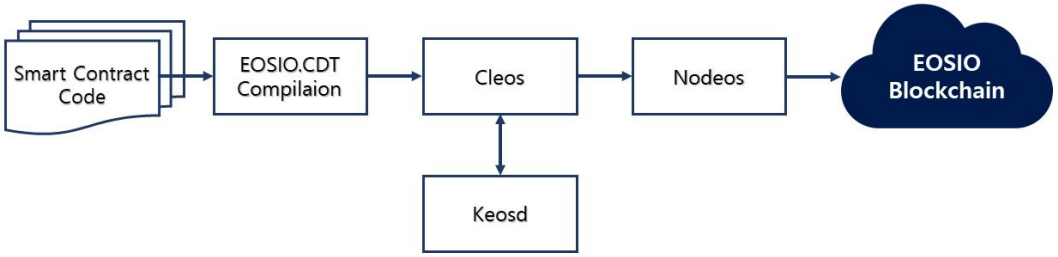

그림 1 EOSIO의 주요 구성 컴포넌트와 이들 간의 관계

EOSIO기반 체인을 운영하기 위한 핵심 구성 컴포넌트는 Nodeos, Keosd, Cleos로 구성된다. Nodeos는 코어 데몬으로 각종 설정을 플러그인 방식으로 구성하여 블록체

인 노드를 원하는 목적에 맞게 운영할 수 있다. Nodeos 데몬이 정상적으로 실행되면, 설정된 내용에 따라 블록을 생성하거나 트랜잭션을 처리하는 등의 작업을 수행한다. Keosd는 개인키를 저장하고 관리하는 키 관리용 데몬이다. Keosd는 키가 안전하게 암호화되어 지갑 파일에 저장될 수 있도록 하는 키 저장용 매개체 역할을 한다. Cleos는 Nodeos에서 제공하는 REST API와 상호작용하기 위한 커맨드라인 인터페이스 도구이다. Cleos를 통하여 Nodeos에 명령을 전달하거나 EOSIO 스마트 컨트랙트를 배포하고 테스트하는 등의 작업을 할 수 있고 Keosd와도 상호작용하여 동작한다. EOSIO 스마트 컨트랙트 개발에 필요한 컴포넌트는 EOSIO.CDT이다. EOSIO.CDT는 Web Assembly용 개발 도구이며, EOSIO를 기반으로 하는 스마트 컨트랙트를 쉽게 작성할 수 있도록 하는 도구들의 모음이다.

2.2 EOS 어카운트

EOS 어카운트는 EOSIO 기반 블록체인에서 스마트 컨트랙트의 배포, 액션 실행 등의 트랜잭션 전송을 위해 사용된다. 기존 2세대 블록체인인 이더리움은 숫자와 문자의 조합인 주소라는 개념을 가지지만, EOS는 사람이 읽을 수 있는 문자열 형태인 계정이라는 개념을 가진다. 사용자가 알파벳 A-Z 및 숫자 1-5를 조합하여 12글자 범위 내에서 원하는 대로 어카운트명을 가질 수 있고, 이는 일반 사용자에게 있어 사용성이 더 편리하다.

또한, EOS 어카운트는 강력한 사용자 권한 관리 시스템을 제공한다. 다중서명 및 키 별 가중치를 자유자재로 유연하게 설정하여 블록체인 계정의 권한을 여러 사람들이 서로 권한을 부여받아 관리할 수 있다. 각 어카운트는 기본으로 owner 권한과 active 권한을 가지게 되며, 이외에도 다양한 권한 레벨을 지정하여 계층적인 구조를 가지도록 한다. 일반적으로 어카운트에 추가된 권한 레벨들은 active 권한의 하위에 파생되며 추가된 권한 레벨 하위에 다시 새로운 레벨을 파생시킬 수 있다. 또한, 가중치와 임계값을 이용하여 다중 사용자에 의한 액션 실행 제어인 임계값 다중서명 확인(threshold multi-signature check)를 지원한다. 그림 2는 EOS 어카운트와 권한의 관계성에 대한 다이어그램으로 alice라는 어카운트에 정의된 권한과 그 중 active의 권한 테이블을 나타낸다.

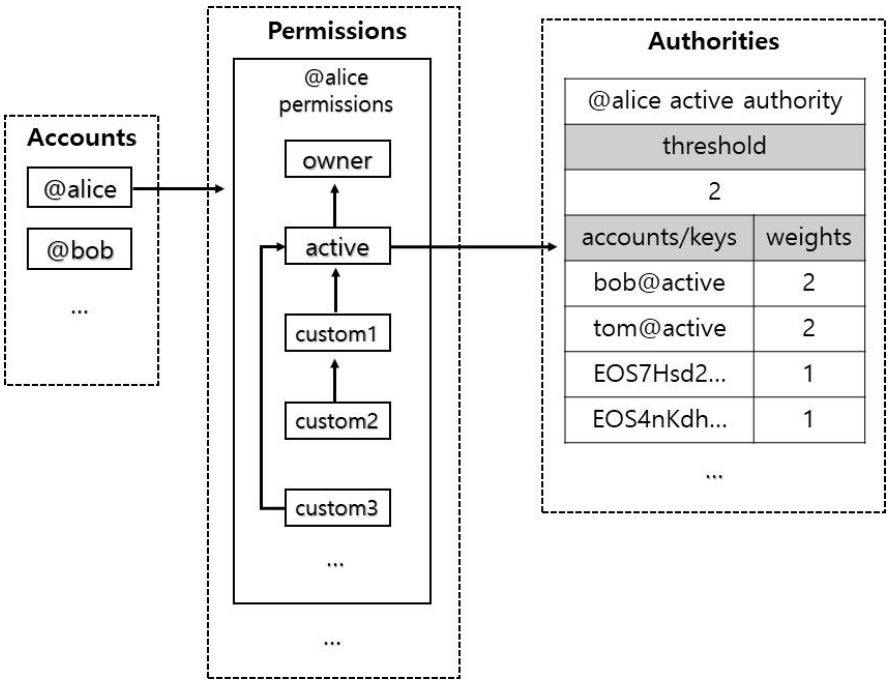

그림 2 EOS 어카운트 권한 구조

이에 따라 bob@active나 tom@active의 사용자는 가중치가 임계값에 도달하므로 해당 액션을 실행할 수 있다. 하지만 EOS7H...와 EOS4n...의 어카운트를 소지한 사용자는 다른 사용자의 추가적인 실행이 있어야지 액션 수행이 가능하다.

2.3 EOS 스마트 컨트랙트

스마트 컨트랙트는 블록체인이 1세대에서 2세대로 넘어갈 수 있게 된 가장 중요한 계기 중 하나로, 특정 조건이 달성되면 자동으로 계약 내용이 실행되도록 하는 시스템을 말한다. EOS 스마트 컨트랙트는 c++ 언어로 개발되며, 스마트 컨트랙트를 개발하기 위하여 새로운 프로그래밍 언어를 배울 필요가 없다는 특징을 가진다. 일반적인 스마트 컨트랙트는 개발자가 아닌 이상 일반 사람들이 내용을 쉽게 파악하기 어렵다는 한계점이 있다. 이러한 문제를 해결하기 위해 EOS는 리카르디안 컨트랙트를[14] 도입하여 사용한다. 리카르디안 컨트랙트란 프로그래밍 코드를 일반 사람들이 읽을 수 있는 언어로 표시하기 위한 표준 규약이다. 이를 통해 EOS는 사람과 프로그램 모두가 쉽게 읽을 수 있으며, 발생할 수 있는 계약에 대한 상세한 내용을 당사자들이 명확하게 읽고 이해할 수 있도록 도움을 준다[15]. 또한, 기존 스마트 컨트랙트는 코드 배포시 변경 또는 수정이 불가능하여 스마트 컨트랙트의 업그레이드나 약점을 뒤

늦게 발견한 경우 후속조치가 어렵다. 하지만 EOS는 스마트 컨트랙트의 변경 및 수정이 가능하다는 큰 장점을 가지고 있다. 이러한 특징은 특정 개발자가 임의로 코드를 변경하는 경우 위험할 수 있지만, 다중 서명 기능을 통해 수정이 가능하게 설정한 경우 이러한 위험을 방지할 수 있다.

EOS 스마트 컨트랙트는 스마트 컨트랙트 코드에 [[eosio::contarct]] 등과 같은 속성을 명시하는 스마트 컨트랙트 인터페이스를 구현하며, multi_index 테이블이라는 컨테이너를 사용하여 데이터를 저장한다. multi_index 테이블은 테이블의 필드를 색인화해서 별도의 파일로 저장하고, 검색할 때 해당 테이블 데이터 전체를 검색하는 방식이 아니라 색인화 되어있는 인덱스 파일을 검색하여 검색 속도를 빠르게 한다. multi_index 테이블을 사용하여 구조체나 클래스로 구성된 데이터를 생성, 읽기, 수정, 삭제할 수 있으며 64비트 정수(uint64_t)로 된 기본 키로 색인할 수 있다. 이러한 데이터는 find, emplace 등과 같은 함수로 데이터를 조작한다.

2.4 작업자 안전관리 시스템

인공지능, 사물인터넷 등 4차 산업혁명 기술들의 수준이 향상되면서, 다양한 산업 현장은 이러한 기술들을 기반으로 공장 운영체계를 갖춘 스마트 팩토리로 빠르게 바뀌어 나아가고 있다[16,17]. 스마트 팩토리에서는 다양한 환경으로부터 발생되는 위험성과 작업자의 사소한 부주의로 인해 발생하는 안전사고를 막기 위하여 작업자 안전관리 시스템을 도입하여 사용하고 있다. 작업자 안전관리 시스템은 물류창고, 건설현장, 밀폐 작업장 등 산업 관련 위험 요소가 있는 사업장에서 일하는 작업자들의 안전을 관리하기 위한 안전 관리 솔루션이다[18]. 일반적으로 IoT 기술을 기반으로 작업자의 생체 데이터를 실시간으로 모니터링하고, 작업장의 유해가스 감지 등 다양한 유형의 안전사고에 대해 예방적 조치와 신속한 대처를 지원한다. 현재 다양한 작업자 안전관리 시스템들이 개발되었으며, 산업 별로 환경에 맞게 작업자의 안전을 관리하기 위하여 여러 기능들을 제공한다.

3. FairSafe의 설계

본 장에서는 투명하고 신뢰성 있는 작업자 안전관리 프로세스를 제공하는 블록체인 기반 서비스인 FairSafe에 대하여 설명한다. 또한, 작업자 안전관리를 위한 여러 프로세스와 FairSafe를 효율적으로 제공하기 위한 플랫폼에 대하여 기술한다.

3.1 FairSafe의 구조

기존의 작업자 안전관리 시스템은 중앙 서버 및 중앙 데이터베이스를 기반으로 한 시스템으로, 이는 작업자의 안전과 관련된 데이터들의 위변조 및 유실이 가능하여 시스템의 신뢰도를 높은 신뢰도를 얻기 어렵다. 특히 작업자는 작업 안전과 관련된 데이터들을 직접 확인하기 어렵고, 안전사고가 발생할 경우 데이터의 은폐 및 조작 등의 가능성이 있어 작업자에게 불신을 가져다 줄 수 있다. 따라서 본 절에서는 작업자와 관리감독자가 상호작용하는 투명한 작업자 안전관리 프로세스를 신뢰성 있게 제공하는 블록체인 기반의 서비스인 FairSafe에 대하여 설명한다. 그림 3은 FairSafe의 기본적인 구조에 대하여 나타낸다.

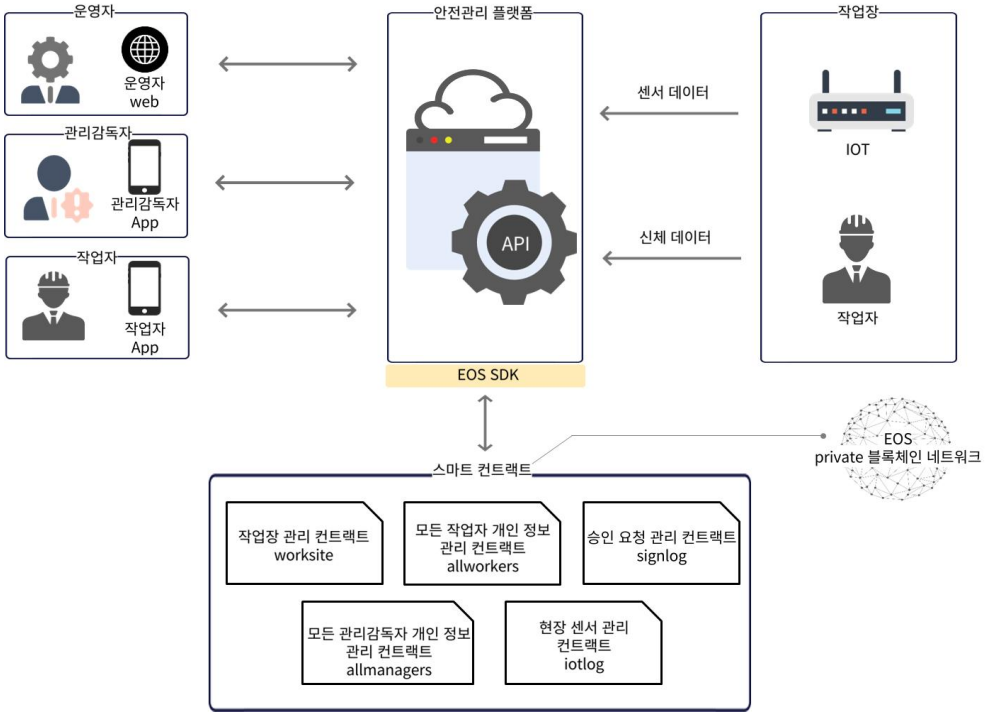

그림 3 FairSafe의 전체 구조

FairSafe는 작업자의 안전과 관련된 데이터의 신뢰성을 보장하기 위해 EOS 블록체인을 기반으로 작업자의 작업 개시와 종료 프로세스를 투명하게 지원하는 서비스이다. FairSafe의 전체적인 구조는 스마트 컨트랙트와 안전관리 플랫폼, 이를 활용하여 동작하는 3가지의 분산 애플리케이션으로 구성된다. 작업 현장의 데이터와 작업자의 신체 데이터 등에 대한 정보가 EOS 스마트 컨트랙트에 저장되고, 안전관리 플랫폼을 통해 컨트랙트의 기능을 수행할 수 있다. 또한, 3가지의 분산 애플리케이션은 각 사용자의 역할에 따라 기능을 제공한다.

FairSafe는 5개의 스마트 컨트랙트로 구성되어 있으며, 안전관리 플랫폼을 통하여 작업 현장의 데이터와 작업자의 신체 데이터 등이 스마트 컨트랙트에 저장된다. 또한, FairSafe의 사용자들은 역할에 따라 운영자용, 관리감독자용, 작업자용 분산 애플리케이션을 통해 기능을 수행한다. 운영자는 해당 애플리케이션을 통해 작업자와 관리감독자의 명시적인 인증을 위한 어카운트 권한을 등록하며, 작업 현장의 데이터의 관리를 위한 안전 기준치를 설정할 수 있다. 권한이 등록된 어카운트를 소지한 작업자 및 관리감독자는 애플리케이션을 통하여 작업자는 작업 개시와 종료에 대한 요청을 진행하고 관리감독자는 이에 대한 승인을 처리한다. 또한, 안전관리를 위한 모든 데이터는 애플리케이션으로 확인 할 수 있다.

3.2 작업자 및 관리감독자의 어카운트 권한 등록 프로세스

작업자와 관리감독자는 FairSafe에서 기능을 수행하기 위하여 스마트 컨트랙트의 액션을 실행하는 EOS 어카운트가 필요하다. 해당 어카운트는 액션을 실행할 스마트 컨트랙트의 권한에 등록되어 있어야 하며, 권한이 등록된 어카운트는 개발된 분산 애플리케이션을 통해 개인기기에 저장된 개인키를 이용하여 기능을 수행할 수 있다. 이를 위하여 작업자와 관리감독자의 어카운트가 FairSafe에서 동작하는 스마트 컨트랙트의 어카운트 권한에 등록되는 프로세스가 필요하다. 작업자 및 관리감독자는 프라이빗 EOS 블록체인을 통하여 어카운트를 생성하여 회원가입 절차를 진행한다. 운영자는 운영자 애플리케이션을 통해 작업자와 관리감독자의 회원가입을 요청을 확인하고, 해당 어카운트를 스마트 컨트랙트 어카운트의 권한에 등록한다. 그림 4는 작업자와 관리감독자가 어카운트를 요청하고 운영자가 승인하고 어카운트의 권한을 등록하는 과정을 나타낸다.

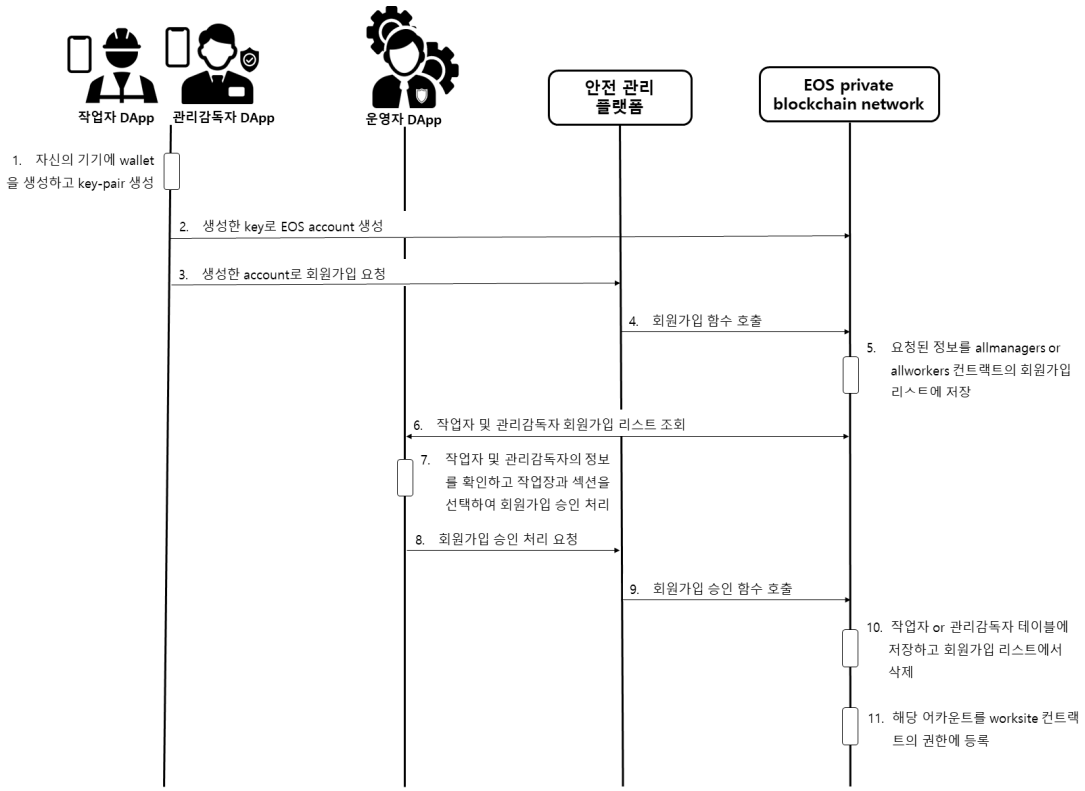

그림 4 작업자 및 관리감독자의 어카운트 권한 등록 프로세스

작업자 및 관리감독자는 기능을 수행하기 위하여 컨트랙트의 액션을 실행할 수 있는 권한을 위하여 어카운트를 등록해야 하며, 이는 운영자를 통하여 승인을 받아야 가능하다. 작업자 및 관리감독자가 컨트랙트 실행 권한을 위하여 운영자를 통해 권한을 등록하는 과정은 다음과 같다.

(1) 작업자 및 관리감독자는 자신의 기기에 wallet을 생성하고 어카운트를 발급하기 위한 key-pair를 생성한다.
(2) 생성한 key로 EOS account를 생성한다.
(3) 작업자 및 관리감독자는 자신의 정보를 입력하고 생성한 account로 회원가입을 요청한다.
(4) 안전관리 플랫폼은 회원가입 함수를 호출하여 작업자 및 관리감독자의 회원가입을 진행한다.
(5) 요청된 정보를 작업자는 allworkers 컨트랙트에, 관리감독자는 allmanagers 컨트랙트의 회원가입 리스트에 저장한다.
(6) 운영자 애플리케이션은 컨트랙트에 저장된 회원가입 리스트를 조회한다.
(7) 운영자는 애플리케이션에서 작업자 및 관리감독자의 회원가입의 정보를 확인하고

작업장과 섹션은 선택하여 배치한다.

(8) 작업장과 섹션의 배치가 끝난 작업자 및 관리감독자를 선택하여 회원가입 승인 처리를 진행한다.

(9) 안전관리 플랫폼은 회원가입 승인 함수를 호출하여 작업자 및 관리감독자의 회원가입 승인을 진행한다.

(10) 작업자 및 관리감독자의 정보를 각 컨트랙트에 저장하고 회원가입 리스트에서 삭제한다.

(11) 회원가입을 진행한 작업자 및 관리감독자의 어카운트를 배포된 컨트랙트의 어카운트 권한에 등록한다.

작업자의 어카운트는 worksite, signlog, allworkers 컨트랙트의 어카운트 권한에 등록되고, 관리감독자의 어카운트는 worksite signlog, allmanagers 컨트랙트의 어카운트 권한에 등록된다. 어카운트 권한에 등록된 작업자 및 관리감독자는 해당 컨트랙트의 액션을 실행할 수 있고, 이를 통하여 FairSafe의 기능을 신뢰성 있게 수행할 수 있다. 또한, 권한이 등록된 어카운트로 수행하는 트랜잭션은 작업자 및 관리감독자의 명시적 인증을 나타낸다.

3.3 FairSafe에서의 투명한 작업자 안전관리 프로세스

본 절에서는 FairSafe에서 작업자의 안전한 작업 개시와 종료를 진행하기 위하여 신뢰성 있게 작업자와 관리감독자가 상호작용하는 기법에 대하여 설명한다. 일반적으로 작업자의 안전을 관리하기 위한 방안은 시스템을 통하여 작업자의 생체 데이터를 실시간으로 확인하여 관리감독자가 작업자의 안전사고에 대한 예방적 조치나 신속한 대처를 진행하는 방식이다. 작업자의 안전관리를 위하여 작업 현장 정보와 작업자 상태 그리고 이 정보를 바탕으로 관리감독자들이 확인하는 프로세스가 매우 중요하게 작용하지만, 일반적인 프로세스는 관리감독자만이 현장 정보를 확인하고 작업 개시를 판단한다. 이는 작업자가 작업자의 안전에 관련된 각 정보의 확인과정을 진행하지 않아 신뢰성 보장에 한계점이 있으며, 작업자와 관리감독자 상호간의 신뢰와 안전을 함께 만족시키는 프로세스가 필요하다. 해당 프로세스는 사용자의 역할에 따라 애플리케이션으로 진행되고, 진행되는 모든 프로세스를 스마트 컨트랙트에 저장하여 신뢰성 있게 관리된다.

3.3.1 작업자의 작업 개시 및 종료 요청

본 항에서는 안전한 작업 개시 및 종료를 위하여 작업자와 관리감독자 상호간의 신뢰성 있는 프로세스에서 작업자의 작업 개시 및 종료를 위한 요청에 대하여 설명한다. 그림 5는 제안하는 신뢰성 있는 작업자의 작업 개시 요청에 대한 프로세스를 나타낸다.

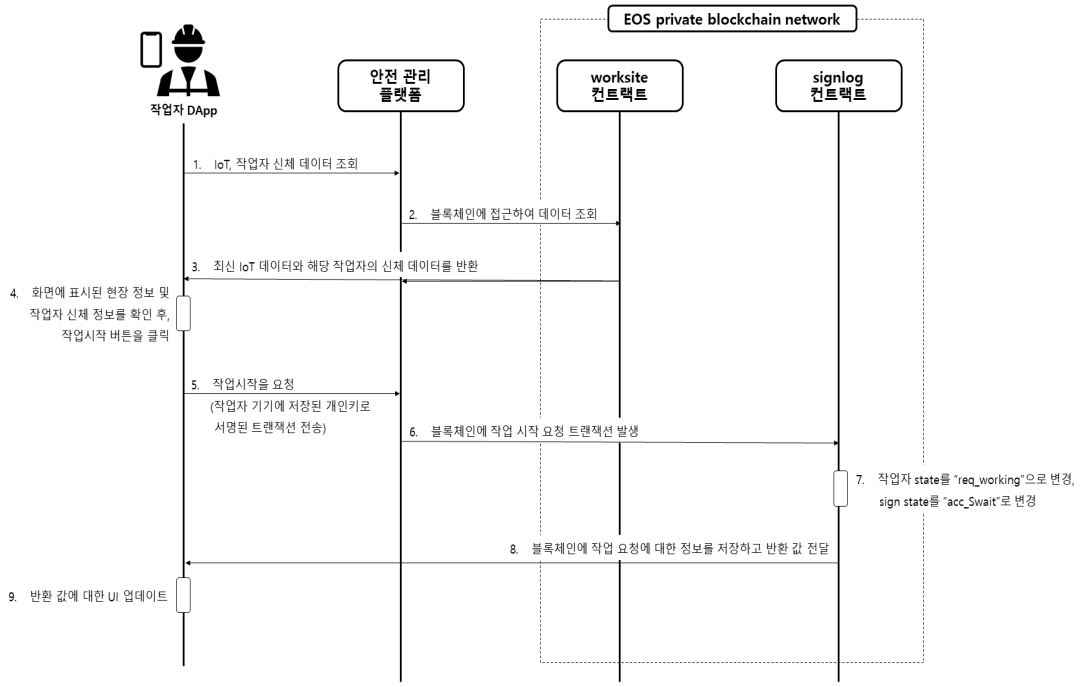

그림 5 신뢰성 있는 작업자의 작업 개시 요청에 대한 프로세스

작업자는 작업개시 전 작업 환경 및 안전관리에 대한 정확한 정보를 확인하여 작업 개시 또는 종료를 관리감독자에게 요청한다. 이를 위하여 작업자는 분산 애플리케이션을 사용하여 작업 개시 및 종료의 요청 프로세스를 진행하며 이는 스마트 컨트랙트에 저장되어 신뢰성 있게 관리된다. 작업자 애플리케이션은 관리감독자에게 작업 개시를 요청하기 위하여 다음과 같은 순서로 동작한다.

(1) 작업자 애플리케이션은 작업현장정보 데이터와 작업자 신체 데이터를 안전관리 플랫폼을 통하여 조회를 요청한다.
(2) 안전관리 플랫폼은 worksite 컨트랙트에 저장된 데이터를 조회한다.
(3) 최신의 작업현장정보 데이터와 작업자 신체 데이터를 반환한다.
(4) 작업자는 작업자 애플리케이션 화면에 표시된 작업 현장 정보 및 작업자 신체

정보를 확인하고 작업시작 버튼을 누른다.

(5) 작업자는 애플리케이션을 통하여 자신의 안전을 보장할 수 있는 정보를 직접 확인하고 작업 개시을 요청한다. 작업 개시 요청 트랜잭션을 발생하기 위하여 작업자 기기에 저장된 개인키로 트랜잭션을 서명하여 전달한다.

(6) 안전관리 플랫폼을 통하여 작업개시요청 트랜잭션이 블록체인으로 전송된다.

(7) signlog 컨트랙트에 작업개시에 대한 정보를 저장하고 작업자 state를 "작업준비완료(req_working)"로, 승인 state를 "작업시작대기(acc_Swait)"로 변경한다. 저장 및 변경을 완료하면 반환 값을 전달한다.

(8) 반환 값을 작업자 애플리케이션으로 전달한다.

(9) 반환 값을 작업자에게 표시해주며 작업개시 요청이 되었음을 보여준다.

작업종료에 대한 요청도 같은 순서로 동작하며, 변경되는 state만 달라진다. 작업종료 요청을 한 경우에는 작업자 state가 “작업종료준비완료(req_finish)”로 변경되고 승인 state는 “작업종료대기(acc_Ewait)”로 변경되어 저장된다. 이를 통하여 작업자는 작업 현장의 정보를 확인하여 인지한 상태에서 작업 개시 및 종료가 가능하고, 이는 후에 안전사고가 발생하여도 책임소재를 분명히 할 수 있다.

3.3.2 관리감독자의 작업자 요청에 대한 승인 처리

본 항에서는 작업자의 작업 개시 및 종료 요청에 대하여 신뢰성 있게 관리감독자가 승인을 처리하는 프로세스에 대하여 설명한다. 관리감독자는 분산 애플리케이션을 통하여 스마트 컨트랙트에 저장된 작업자의 요청에 대한 정보를 확인하고 안전 기준치에 따라 작업 개시 및 종료에 대한 승인을 처리한다. 그림 6은 관리감독자 애플리케이션을 통한 작업자의 작업 개시 요청에 대하여 승인을 처리하는 프로세스를 나타낸다.

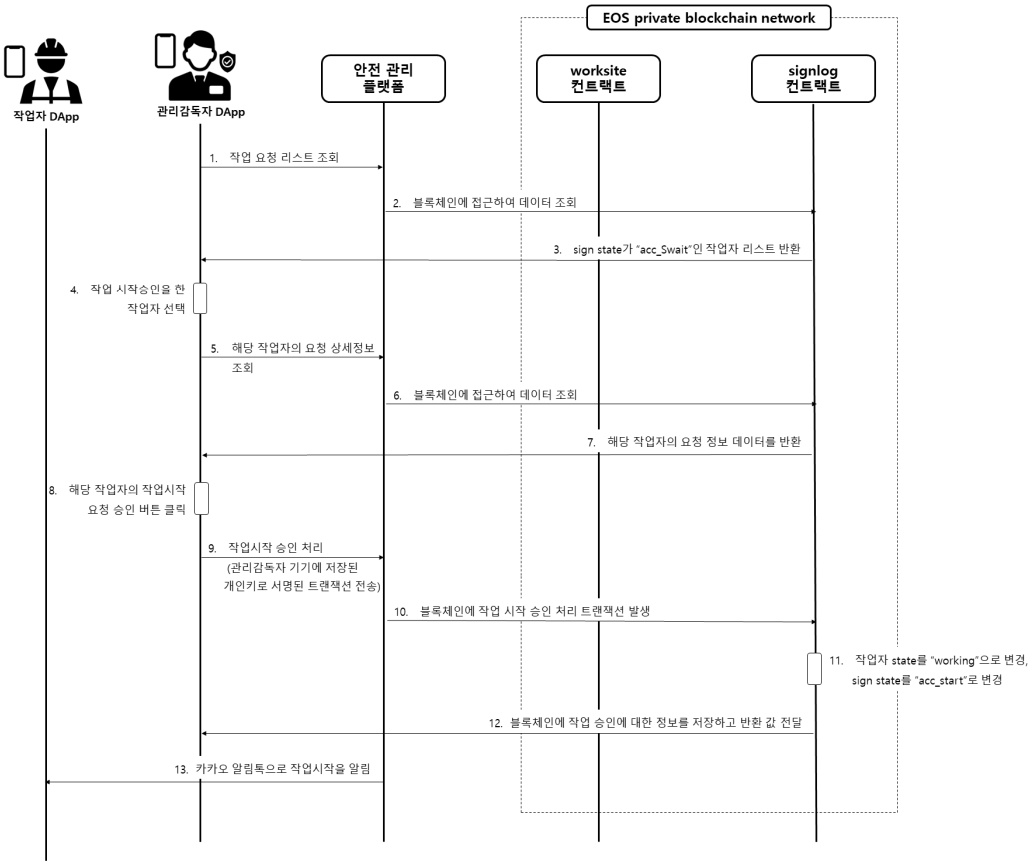

그림 6 관리감독자의 작업자 요청에 대한 승인 처리 프로세스

(1) 관리감독자 애플리케이션은 작업자의 작업 요청에 대한 정보를 안전관리 플랫폼을 통하여 조회한다.
(2) 안전관리 플랫폼은 signlog 컨트랙트에 저장된 데이터를 조회한다.
(3) 작업자 state가 "작업준비완료(acc_Swait)"인 작업자 리스트를 반환한다.
(4) 반환 값을 관리감독자 애플리케이션으로 확인한 관리감독자는 작업 시작을 승인할 작업자를 선택한다.
(5) 관리감독자는 작업 시작을 승인 처리하려고 선택한 작업자의 요청 상세 정보를 조회한다.
(6) 안전관리 플랫폼은 signlog 컨트랙트에 저장된 데이터를 조회한다.
(7) 해당 작업자의 작업개시 요청에 대한 정보 데이터를 반환한다.
(8) 관리감독자는 작업개시 요청에 대한 상세 정보를 확인하고 안전 기준치에 적합하면 작업시작 요청 승인 버튼을 누른다.
(9) 작업시작을 승인 처리하는 트랜잭션을 관리감독자 기기에 저장된 개인키로 서명하고 서명된 트랜잭션을 전송한다.
(10) 안전관리 플랫폼을 통하여 작업개시 승인 트랜잭션이 전송된다.

(11) signlog 컨트랙트에 작업개시 승인에 대한 정보를 저장하고 작업자 state를 "작업중(working)"로, 승인 state를 "작업시작승인(acc_start)"로 변경한다.
(12) 저장 및 변경을 완료하면 반환 값을 관리감독자 애플리케이션으로 전달한다.
(13) 안전관리 플랫폼은 작업개시에 대한 알림을 위하여 해당 작업자에게 카카오 알림톡을 전송한다.

작업 종료 요청에 대한 승인 처리도 조회되는 state와 변경되는 state만 다르고 똑같은 순서로 동작한다. 작업 종료 요청에 대한 승인 처리를 위하여 관리감독자 애플리케이션은 작업자 state가 "작업종료준비완료(acc_Ewait)"인 작업자 리스트를 조회하여 반환받는다. 또한 작업 종료 요청에 대하여 승인 처리를 진행하면 작업자 state는 “작업완료(finish)”로, 승인 state는 “작업종료승인(acc_finish)”로 변경된다. 3.2.1과 3.2.2를 통하여 제안한 프로세스는 신뢰성 있는 작업개시 및 종료에 대한 프로세스로 작업자와 관리감독자 상호간의 신뢰관계를 형성하며 안전을 함께 만족시킬 수 있다. 밀폐공간인 작업 현장이나 작업현장정보나 작업자의 상태정보가 민감하게 작용하는 작업장에서는 투명하고 신뢰성 있는 작업개시 및 종료를 위하여 제안한 프로세스에 따라 진행하는 것이 효율적이다.

3.3.3 요청 및 승인에 대한 취소 처리

작업자는 작업 개시 및 종료를 요청하고 어떠한 이유로 요청을 철회해야 하는 경우가 발생할 수 있으며, 이를 위하여 본 항에서는 요청한 작업 개시 및 종료에 대하여 취소 처리를 진행하는 프로세스에 대하여 설명한다. 작업자의 요청에 대한 취소 처리도 관리감독자의 승인이 필요하며, 이 과정도 작업자와 관리감독자가 상호작용하여 동작한다.

FairSafe는 작업자와 관리감독자의 작업의 흐름에 따른 요청과 승인에 대한 작업자의 state 흐름이 존재한다. 변경되는 작업자의 state는 총 2가지로, 작업자의 작업 개시 및 종료에 대한 상태를 나타내는 workerState와 작업자 및 관리감독자가 상호작용하며 동작하는 작업자 안전관리 프로세스에 대한 상태를 나타내는 signState로 나뉜다. workerState는 ‘작업대기’ 상태에서 시작하며, signState는 작업자의 작업 개시 요청으로 인하여 상태가 시작된다. 그림 7은 FairSafe에서의 작업자의 state 변경 흐름을 나타낸다.

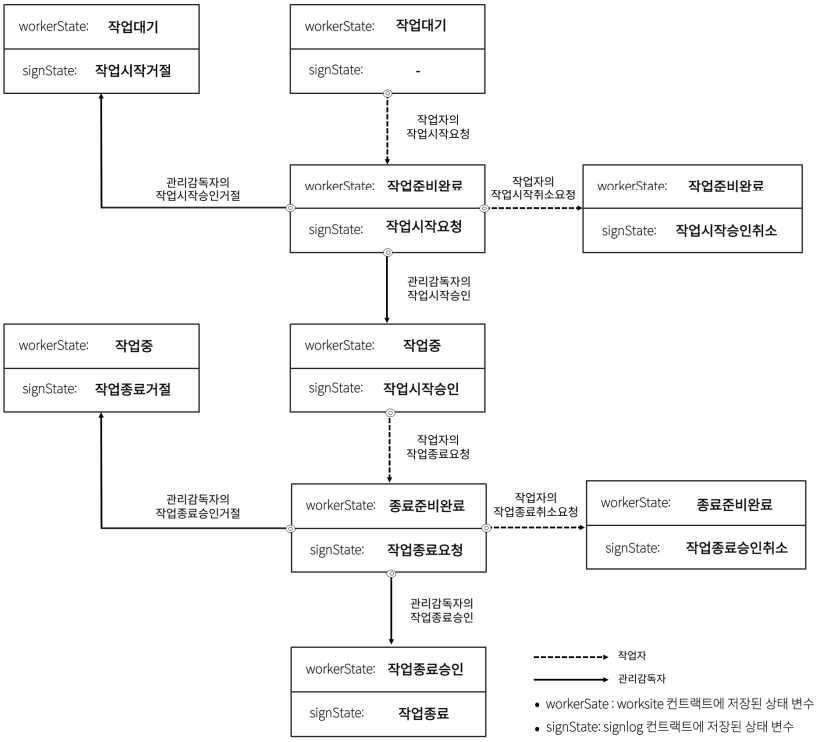

그림 7 FairSafe에서의 작업자 state 흐름도

작업 흐름에 따라 작업자의 state 변경과 작업자의 요청에 대한 취소처리 진행과정에 대하여 설명한다.

(1) 작업자는 pending 상태인 "작업대기" 상태로 시작한다.

(2) 이후 작업자가 작업개시를 요청하면 workerstate가 req_working 상태인 "작업준비완료" 상태로 변경되며 signstate는 req_start 상태인 "작업시작요청" 상태로 변경된다.

(2-1) 작업자가 어떠한 상황으로 작업개시 요청에 대한 취소를 진행하면 signstate가 cancle_start 상태인 "작업시작취소" 상태로 변경되고, workerstate는 이전 상태인 “작업대기” 상태로 변경된다.

(3) 관리감독자가 작업개시에 대하여 승인을 하면 signstate가 acc_start 상태인 "작업시작승인" 상태로 변경되며 workerstate가 working 상태인 "작업중" 상태로 변경된다.

(4) 작업이 끝난 이후 작업자는 작업종료에 대한 요청을 진행한다. workerstate는 req_finish 상태인 "종료준비완료" 상태로 변경되며 signstate는 req_end 상태인 "작업종료요청" 상태로 변경된다.

(4-1) 작업자가 어떠한 상황으로 작업개시 요청에 대한 취소를 진행하면 signstate

가 cancle_finish 상태인 "작업종료취소" 상태로 변경된다.

(5) 관리감독자가 작업종료에 대하여 승인을 하면 signstate가 acc_end 상태인 "작업종료승인" 상태로 변경되며 workerstate가 finish 상태인 "작업종료" 상태로 변경된다.

3.4 작업자 안전관리 플랫폼

본 절에서는 EOS 스마트 컨트랙트의 기능 수행을 지원하는 안전관리 플랫폼에 대하여 제안한다. 제안하는 안전관리 플랫폼은 작업자의 안전을 관리하기 위한 기능과 작업 개시 및 종료를 위하여 작업 현장의 상태와 작업자의 상태를 관리하는 기능을 제공한다. 또한, 문제가 발생하여 트랜잭션을 조회하여 확인해야하는 경우를 위한 트랜잭션 히스토리 관련 기능을 제공한다. 각 사용자의 역할에 따라 분산 애플리케이션은 안전관리 플랫폼을 통하여 EOS 스마트 컨트랙트의 기능을 수행하고, 작업 현장 내에 설치된 현장 센서도 안전관리 플랫폼을 통하여 EOS 스마트 컨트랙트에 데이터를 저장한다. 또한, 스마트 컨트랙트에 저장된 데이터들을 조회하기 위해서도 안전관리 플랫폼을 활용한다. 그림 8은 안전관리 플랫폼과 상호작용하는 구성 요소들을 나타낸 그림이다.

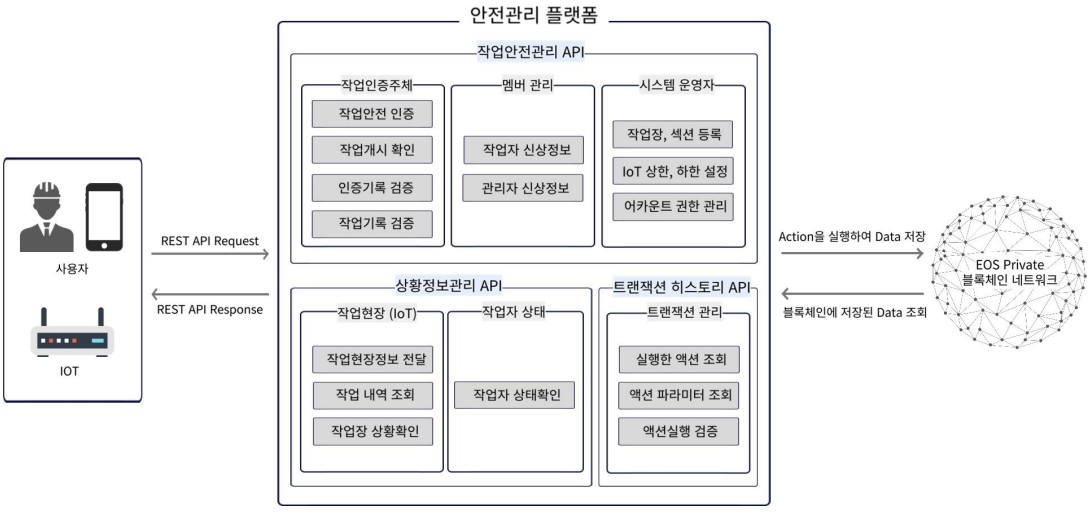

그림 8 안전관리 플랫폼과 상호작용하는 구성 요소들

안전관리 플랫폼은 기능별로 작업자의 안전을 관리하기 위하여 작업 개시 및 종료를 진행하는 프로세스, 작업장 등을 저장 및 관리하는 작업안전관리 API와 작업 현장과 작업자 상태를 관리하는 상황정보관리 API, 트랜잭션 히스토리를 위한 트랜잭션 히스토리 API로 구성된다.

4. FairSafe의 구현

본 장에서는 3장에서 설명한 FairSafe와 이를 위한 여러 기법들의 실질적인 구현에 대하여 설명한다. 또한, 구현된 FairSafe의 유효성을 검증한 내용에 대하여 기술한다.

4.1 FairSafe의 스마트 컨트랙트

FairSafe는 EOS 블록체인을 기반으로 기능을 제공하기 위하여 EOS 스마트 컨트랙트를 이용한다. 본 절에서는 FairSafe를 위한 EOS 스마트 컨트랙트의 구조 및 주요 함수에 대하여 설명한다.

구현된 FairSafe는 작업자의 안전을 관리하기 위하여 EOS 스마트 컨트랙트를 통해 작업현장의 데이터와 작업자의 신체 데이터 등의 모든 작업안전을 위한 데이터들을 저장 및 관리를 지원한다. FairSafe의 스마트 컨트랙트는 작업장을 관리하기 위한 worksite 컨트랙트와 모든 작업자와 관리감독자의 정보를 관리하는 allworkers, allmanagers 컨트랙트, 투명한 작업자 의 과정을 저장하고 관리하는 signlog 컨트랙트, 그리고 현장 센서로 부터 실시간으로 받은 데이터를 저장하고 관리하는 iotlog 컨트랙트를 포함한 총 5가지로 구성된다. 그림 9는 FairSafe의 스마트 컨트랙트 구조를 나타낸다.

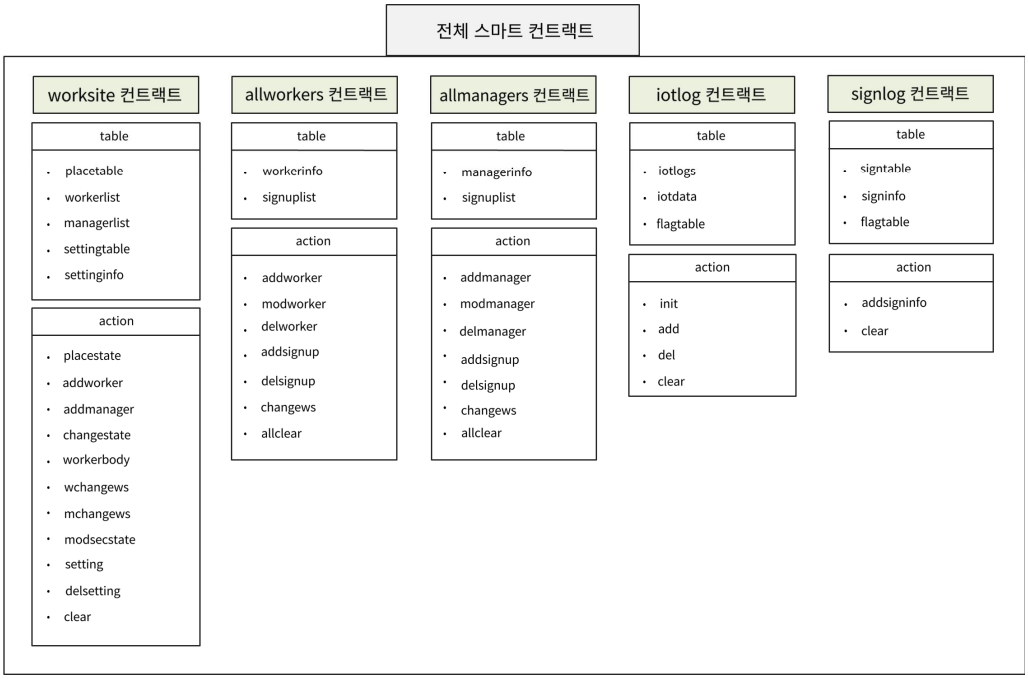

그림 9 FairSafe의 스마트 컨트랙트 구조

worksite 컨트랙트는 작업을 진행하고 있는 작업장의 관리를 위한 컨트랙트로서 최신의 작업현장정보 데이터와 작업자의 신체데이터를 저장 및 조회 등의 기능을 위한 함수를 제공한다. 모든 작업자와 관리감독자는 프로세스에 따라 진행하기 위하여 all workers, allmanagers 컨트랙트의 addworker, addmanager 액션을 통하여 자신의 정보를 등록하고 작업 개시 및 종료에 대한 요청과 승인을 worksite 컨트랙트의 changestate 액션과 signlog 컨트랙트의 addsigninfo 액션으로 진행할 수 있다. 작업현장에서 센서 플랫폼을 통하여 실시간으로 저장되는 데이터는 iotlog 컨트랙트에 저장되고 add 함수를 사용하여 저장된다. signlog 컨트랙트와 iotlog 컨트랙트는 시간과 프로세스대로 쉽게 보기위하여 로그형태의 테이블형식을 가진다. 표 1은 작업현장 관리 관련 컨트랙트 및 주요 액션에 대하여 나타낸다. EOS의 스마트 컨트랙트의 함수명은 소문자로만 명명할 수 있다.

표 1 작업현장 관리 관련 컨트랙트 및 주요 액션

컨트랙트	액션명	기능 설명
iotlog	init	처음 IoT 데이터를 넣기 위하여 초기화하는 과정
	add	IoT 데이터 추가
worksite	addmanager	해당 작업장 및 섹션에 새로운 매니저 등록
	addworker	해당 작업장 및 섹션에 새로운 작업자 등록
	placestate	현재 작업 현장의 상태 정보 업데이트
	workerbody	현재 작업자의 신체 정보 업데이트
	setting	해당 작업장 및 섹션의 안전 기준치 설정

iotlog 컨트랙트는 작업 현장의 정보를 저장 및 관리하는 컨트랙트로, 데이터가 날짜별로 구별되어 들어가기 때문에 구별 flag를 초기화하는 init 함수와 IoT 데이터를 추가하기 위한 add 함수로 기능을 제공한다. worksite 컨트랙트는 작업장 및 섹션에 대한 정보를 저장 및 관리하는 컨트랙트로, 현재 작업장 및 섹션에 addmanager 함수와 addworker 함수를 통하여 관리감독자 작업자를 배치할 수 있다. 또한, setting 함수로 현재 작업장의 실시간 정보와 작업자의 안전 기준치를 저장 및 관리하며, placestate 함수와 workerbody 함수로 현재 작업 현장의 정보와 작업자의 신체 정보를 실시간으로 기록한다. 표 2는 투명한 작업자 안전관리 프로세스 관련 컨트랙트 및 주요 액션에 대하여 나타낸다.

표 2 투명한 작업자 안전관리 프로세스 관련 컨트랙트 및 주요 액션

컨트랙트	액션명	기능 설명
signlog	addsigninfo	작업 개시 및 종료에 대한 요청과 이에 대한 승인 처리를 하는 프로세스를 저장
worksite	changestate	작업 개시 및 종료에 따른 작업자의 작업 상태 변경

signlog 컨트랙트는 작업자의 작업 개시 및 종료에 대한 요청과 이에 대한 관리감독자의 승인 처리를 포함하는 프로세스를 addsigninfo 함수로 저장 및 관리한다. 프로세스에 따라 worksite 컨트랙트의 changestate 함수를 통해 작업자의 state가 변경된다. 표 3은 개발한 컨트랙트의 코드 구조를 나타낸다.

표 3 컨트랙트의 코드 구조

```
/* 필요한 헤더 파일들 */
#include <eosiolib/eosio.hpp>
using namespace eosio;
clasee [[eosio::contract]] contract_name : public eosio::contract {
  public:
    using contract::contract;
    /* 생성자 */

    [[eosio::action]] /* 사용할 액션 선언 */
    void action_name(name account, /* parameter */) {
      require_auth(account); /* 액션 실행을 위한 권한 등록 */
      ...
      /* 실행할 액션 내용 */
      auto iterator = table_name.find(account.value);
      if( iterator == table_name.end() ) {
          table_name.emplace(account, [&](auto& row) {
            ...
          }); }
    }

  private:
    [[eosio::table]] /*사용할 테이블의 구조체 선언*/

EOSIO_DISPATCH( contract_name, (선언한 액션 이름) )
```

4.2 REST API 기반 안전관리 플랫폼

본 절에서는 EOS 스마트 컨트랙트의 기능을 수행하기 위하여 개발된 안전관리 플랫폼에 대하여 기술한다. 안전관리 플랫폼은 RESTful 웹서비스로 Node.js를[19] 기반으로 구현되었으며, 작업장의 현장 상태를 관리하는 상황정보관리 API와 작업자의 안전을 관리하는 작업안전관리 API, 주요 기능을 제외한 나머지의 기능을 관리하는 운영자 API로 구성된다.

4.2.1 상황정보관리 API

상황정보관리 API는 작업장의 현장 상태와 작업자의 신체 상태 등과 관련한 전체 상황정보를 관리하기 위한 기능을 제공한다. 제공하는 주요 기능에는 iotlog 컨트랙트에 저장된 작업 현장의 최신 정보를 조회하는 기능, worksite 컨트랙트에 저장된 작업자의 현재 신체 정보를 조회하는 기능이 있다. 이를 통해서 실시간 작업 현장의 정보와 작업자의 신체 정보를 알 수 있고 안전 기준치에 따라 작업자의 안전을 관리할 수 있다. 표 4는 컨트랙트에 저장된 데이터를 조회하는 주요 코드를 나타낸다.

표 4 iotlog 컨트랙트에 저장된 데이터 조회하는 주요 코드

```
eosService.getIoTData = (index, callback) => {
    var iotData= new Object();
    EOS.getTableRows({
      "json": true,
      "scope": IOTLOG, //컨트랙트 이름
      "code": IOTLOG, //컨트랙트를 배포한 어카운트
      "table": 'iotdata', //컨트랙트에 저장되는 테이블 명
      "lower_bound": index,
      "limit": 1
    }).then(function(result){
      //필요한 테이블 칼럼 조회
      var temper = result.rows[0].temper;
      ...
      callback(JSON.stringify(iotData));
    });
  };
```

컨트랙트에 저장된 데이터를 조회하기 위하여 eosjs 라이브러리에서[20] 제공하는 EOS.getTableRows 함수를 사용한다. 접근할 컨트랙트 이름과 이를 배포한 어카운트

명, 데이터가 저장되어 있는 테이블명을 입력해서 필요한 데이터를 조회할 수 있다.

4.2.2 작업안전관리 API

작업안전관리 API는 작업자 및 관리감독자의 정보를 관리하는 기능과 투명한 작업자 안전관리 프로세스를 지원하기 위한 기능들을 제공한다. 또한 작업장과 작업자 및 관리감독자의 어카운트 권한을 부여하는 운영자를 위한 기능들을 지원한다. 표 5는 작업자 안전관리 프로세스를 지원하기 위한 주요 API를 나타낸다.

표 5 작업자 안전관리 프로세스를 지원하기 위한 주요 API

구분	방식	API	기능 설명
투명한 작업자 안전관리 프로세스 관련	post	/worksite/{worksite-id}/section/{section-id}/worker/{worker-id}/startwork	작업 시작 요청
	delete	/worksite/{worksite-id}/section/{section-id}/worker/{worker-id}/startwork	작업 시작 요청 취소
	post	/worksite/{worksite-id}/section/{section-id}/worker/{worker-id}/endwork	작업 종료 요청
	delete	/worksite/{worksite-id}/section/{section-id}/worker/{worker-id}/endwork	작업 종료 요청 취소
	post	worksite/{worksite-id}/section/{section-id}/manager/request/{worker-id}	작업 요청에 대한 승인
	delete	worksite/{worksite-id}/section/{section-id}/manager/request/{worker-id}	작업 요청에 대한 승인 거절

작업자 및 관리감독자는 제안된 프로세스를 진행하기 위하여 분산 애플리케이션에서 표5에서 제시된 API들을 호출한다. 표 6은 운영자가 수행하는 주요 API 기능을 나타낸다.

표 6 운영자 관련 주요 API

구분	방식	API	기능 설명
운영자 관련	post	/member/signup/approval/{who}	회원가입 승인
	post	/workstie/section/setting	작업장 & 섹션 등록 및 안전 기준치 설정

운영자가 안전관리 플랫폼을 통하여 진행하는 주요 기능은 작업자 및 관리감독자를 회원가입시켜 어카운트에 권한을 부여하는 기능과 작업장 및 섹션을 등록하고 안전 기준치를 설정하는 기능으로 구성된다.

4.2.3 트랜잭션 전송 및 관리 API

트랜잭션 전송 및 관리 API는 작업자 및 관리감독자가 애플리케이션에서 자신의 개인기기에 저장된 개인키로 서명한 트랜잭션을 블록체인으로 전송하는 기능이 주요 기능이다. FairSafe는 안전관리 플랫폼에서 트랜잭션을 생성하고 분산 애플리케이션에서 개인키로 서명한 후, 이를 다시 안전관리 플랫폼에서 블록체인 네트워크로 전송하는 프로세스를 가진다. 자신의 기기에 저장된 개인키를 사용하여 트랜잭션에 서명하고 블록체인으로 전송하기 때문에 이는 작업자 및 관리감독자의 명시적 인증을 나타낼 수 있다. 그림 10은 FairSafe에서 블록체인으로 트랜잭션을 전송하는 프로세스에 대하여 나타낸다.

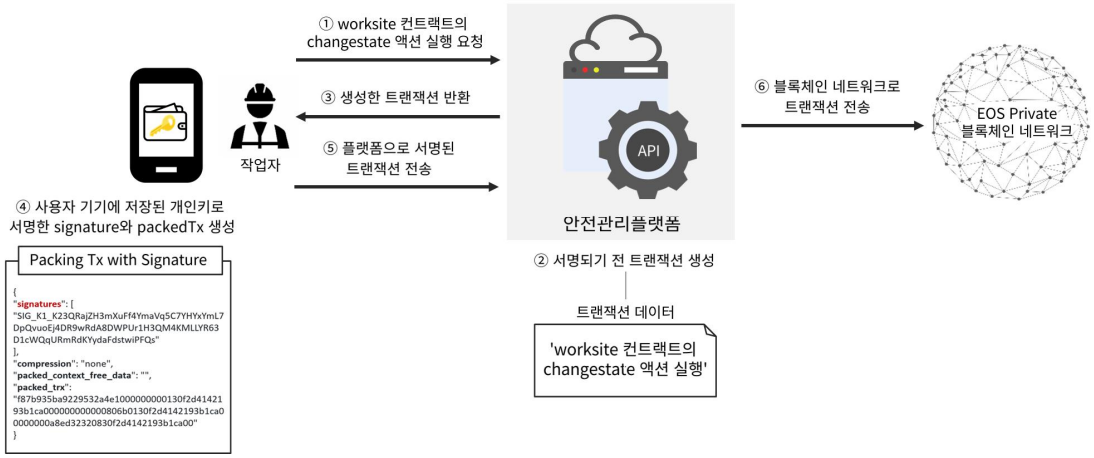

그림 10 FairSafe에서 블록체인으로 트랜잭션을 전송하는 프로세스

FairSafe에서 블록체인으로 트랜잭션을 전송하는 프로세스는 다음과 같이 동작한

다.

(1) 작업자 및 관리감독자 애플리케이션는 안전관리 플랫폼으로 컨트랙트의 액션 실행을 요청한다.
(2) 안전관리 플랫폼에서 해당 액션을 실행하는 트랜잭션을 생성한다.
(3) 생성한 트랜잭션을 작업자 및 관리감독자 애플리케이션으로 반환한다.
(4) 사용자 기기에 저장된 개인키로 반환된 트랜잭션에 서명하고 signature와 이를 포함한 트랜잭션인 packedTx를 생성한다.
(5) 안전관리 플랫폼으로 서명된 트랜잭션을 전송한다.
(6) 안전관리 플랫폼은 서명된 트랜잭션을 /push-transaction/{flag}의 API를 사용하여 블록체인 네트워크로 전송하여 액션을 실행시킨다.

4.3 FairSafe에서 동작하는 운영자 애플리케이션

4.3.1 운영자 애플리케이션의 기능

운영자 애플리케이션은 작업장 및 섹션을 등록하고 작업자와 관리감독자를 등록하여 어카운트의 권한을 부여하며 FairSafe 시스템을 운영하는 기능을 수행한다. FairSafe에서 먼저 진행해야하는 프로세스가 운영자 애플리케이션을 통하여 작업장 및 섹션을 등록하는 작업이다. 작업장 및 섹션을 등록하면서 해당 작업장에 맞는 안전 기준치를 함께 설정한다. 이때 설정된 안전 기준치에 따라서 관리감독자가 판단하여 작업의 진행을 결정할 수 있다. 또한, 작업자와 관리감독자를 작업장 및 섹션에 배치하는 기능을 수행한다. 회원가입을 요청한 작업자와 관리감독자를 올바른 작업장 및 섹션에 배치하며, 작업자와 관리감독자의 어카운트에 배치된 작업장 및 섹션에 대한 컨트랙트의 액션을 실행할 수 있는 권한을 부여받는다.

4.3.2 운영자 애플리케이션의 동작화면

FairSafe의 분산 애플리케이션 중 운영자 애플리케이션은 웹 기반의 애플리케이션으로 개발되어, 운영자가 웹 브라우저를 이용하여 각 기능을 수행할 수 있도록 인터페이스를 제공한다. 운영자는 작업장과 구별된 섹션을 등록하고 작업현장에 대한 안전기준치를 설정이 가능하다. 그림 11은 작업장 및 섹션을 관리 하는하는 화면들을 나타내고 그림 12는 작업장 및 섹션을 등록하고 안전 기준치를 설정하는 화면을 나타낸다.

그림 11 작업장 및 섹션 관리 화면

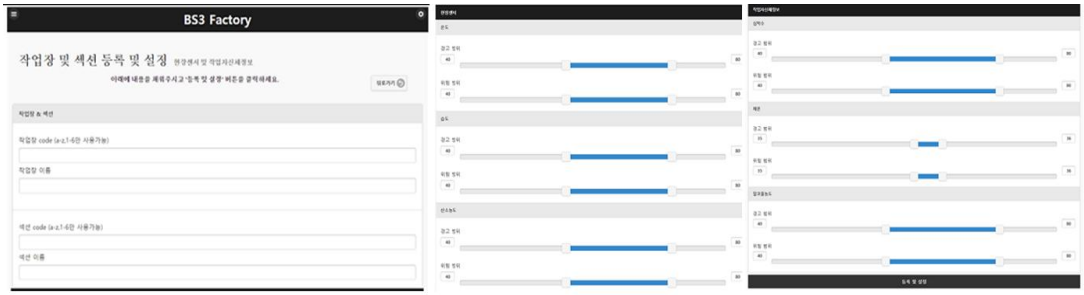

그림 12 작업장 및 섹션 등록과 안전 기준치 설정 화면

운영자는 작업을 시행할 작업장과 섹션과 안전 기준치를 worksite 컨트랙트의 placestate 함수를 통해 등록한다. 작업장의 등록이 완료되면 작업자와 관리감독자를 알맞은 작업장 및 섹션에 배치시킬 수 있다. 그림 13은 작업자와 관리감독자의 회원가입 승인을 위한 화면을 나타낸다.

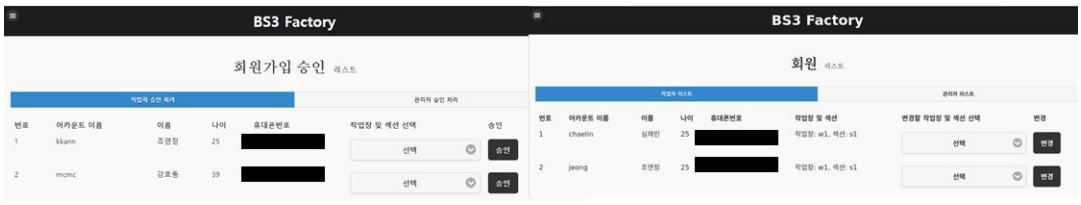

그림 13 작업자와 관리감독자의 회원가입 승인 화면 및 회원 정보 확인 화면

운영자는 작업자와 관리감독자를 작업장 및 섹션에 배치하면서 회원가입 승인을 처리한다. 승인이 처리되면서 함께 작업자와 관리감독자의 어카운트에 작업장 및 섹션에 해당하는 worksite 컨트랙트에 대한 권한이 부여된다. 회원가입 된 작업자와 관리감독자는 회원 리스트에서 정보를 확인할 수 있으며, 작업장 및 섹션 배치를 수정할 수 있다. 또한, 운영자는 EOS private 블록체인의 블록생성을 제어할 수 있다. 그림 14는 블록생성 제어 화면을 나타낸다.

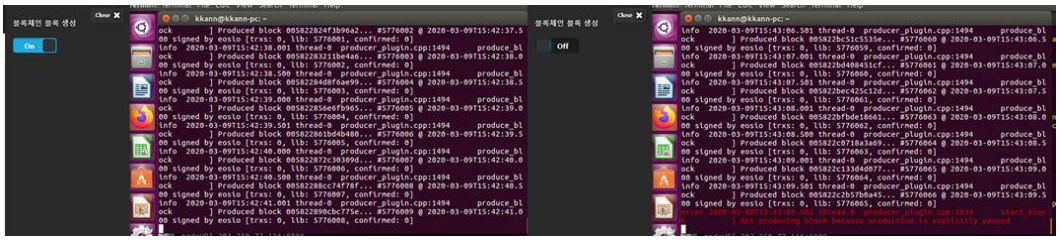

그림 14 블록생성 제어 화면

4.4 FairSafe에서 동작하는 작업자 및 관리감독자 애플리케이션

4.4.1 작업자 및 관리감독자 애플리케이션의 기능

작업자 및 관리감독자는 분산 애플리케이션을 통하여 작업 개시 및 종료에 대한 요청과 그에 따른 승인 처리를 진행하는 프로세스를 수행한다. 작업자 및 관리감독자 애플리케이션은 모바일 환경으로 개발되었고 작업자 및 관리감독자는 자신의 기기에 설치하여 사용할 수 있다. 이로 인하여 작업자 및 관리감독자의 개인키는 공유되지 않고 자신의 기기에만 저장되어 보안성이 높고, 해당 개인키로 생성한 어카운트를 이용하여 제안한 프로세스를 수행하면서 작업자 및 관리감독자의 명시적 인증을 나타낼 수 있다. 작업자는 작업자 애플리케이션의 메인화면을 통하여 작업 현장의 정보와 자신의 신체 정보를 확인할 수 있으며, 이로 인하여 신뢰적인 작업 개시 및 종료를 진행할 수 있다. 관리감독자는 메인화면에서 실시간 상황정보를 그래프로 확인할 수 있다. 이를 통하여 작업장의 상태를 실시간으로 확인 가능하며 수월한 작업장의 관리가 가능하다. 또한, 관리감독자는 애플리케이션을 통하여 작업자가 요청한 작업 개시 및 종료에 대하여 정보를 확인하고 승인하는 기능을 수행한다. 관리감독자는 요청한 정보에 대한 정보를 전부 확인할 수 있고 종료까지 모든 프로세스가 마무리되면 작업장의 상태도 비활성화로 바꾸어 모든 프로세스를 종료시킬 수 있다.

4.4.2 작업자 및 관리감독자 애플리케이션의 동작화면

FairSafe의 분산 애플리케이션 중 작업자 및 관리감독자 애플리케이션은 모바일 애플리케이션으로 개발되어, 작업자와 관리감독자는 자신의 모바일 기기를 이용하여 각 기능을 수행할 수 있도록 인터페이스를 제공한다. 작업자 및 관리감독자는 각 기능을 수행하기 위하여 어카운트를 발급하기 위한 지갑과 key-pair를 먼저 생성해야 하고, 해당 어카운트는 컨트랙트의 액션을 실행하기 위한 권한을 부여받아야 한다. 그림 15

는 사용자의 어카운트 발행 및 권한을 요청하는 화면들을 나타낸다.

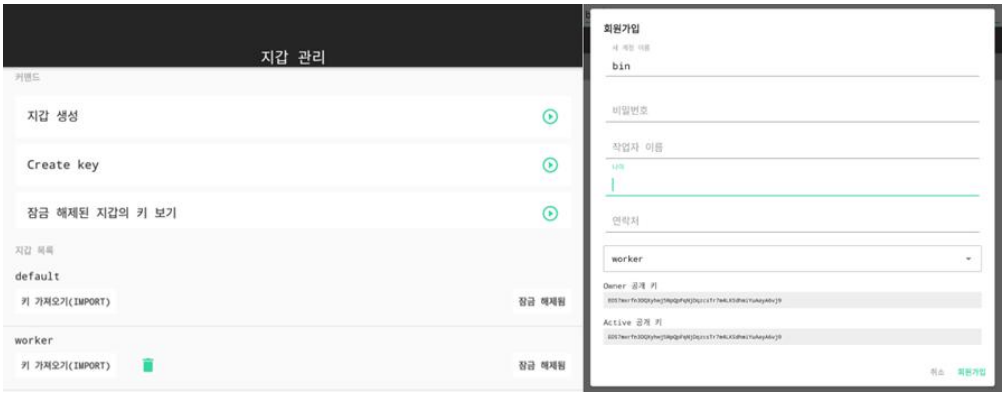

그림 15 작업자 및 관리감독자의 개인키가 저장된 지갑 관리 화면과 회원가입 화면

작업자 및 관리감독자는 자신의 기기에 지갑을 생성하고 해당 지갑에 key-pair를 애플리케이션을 통하여 생성한다. 생성이 완료되면, 해당 key-pair로 어카운트 발행을 요청한다. 발행된 어카운트로 작업자 및 관리감독자는 작업자 안전관리 프로세스에서 명시적 인증을 나타낼 수 있다. 권한을 가진 어카운트를 가진 작업자는 작업개시에 대하여 관리감독자에게 요청을 진행한다. 그림 16은 작업자가 작업 개시에 대한 요청을 진행하는 화면들을 나타낸다.

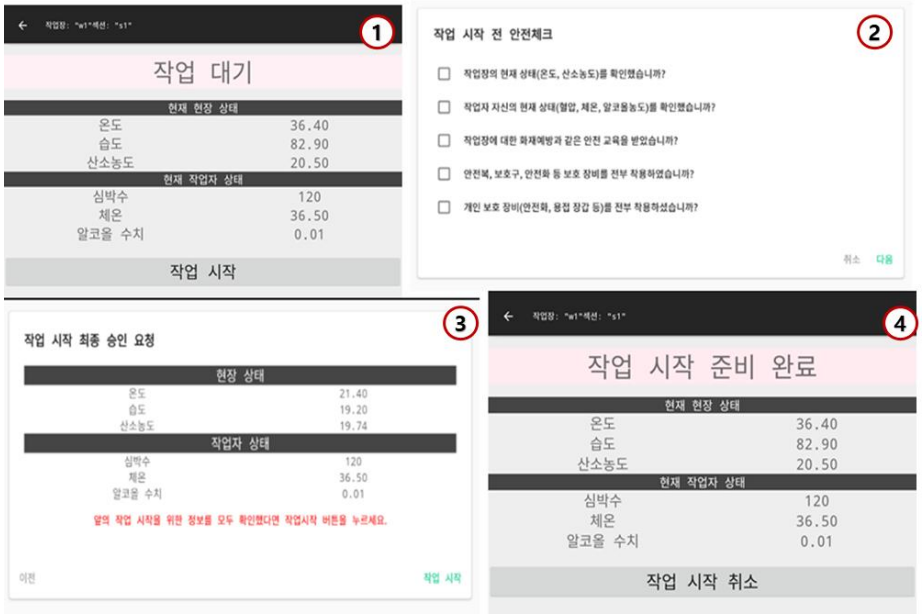

그림 16 작업자가 작업 개시에 대한 요청을 진행하는 화면들

작업자는 애플리케이션을 통하여 그림 16의 순서대로 작업 개시에 대한 요청을 진행한다. 작업자는 메인화면에서 작업 현장의 현재 상태와 작업자의 신체 정보를 확인하고 작업시작 버튼을 눌러 작업 개시를 요청한다. 몇 가지의 사전 안전체크를 스스로 확인하고 최종으로 전체 정보를 다시 확인한 후 작업 시작을 요청하게 된다. 이를 통하여 작업자도 작업 개시 및 종료를 위한 안전 정보를 전부 확인하여 진행할 수 있다. 작업 시작을 요청하게 되면 작업자의 변경된 state를 확인할 수 있다. 그림 17은 관리감독자 애플리케이션에서 작업장 및 섹션 관리 화면을 나타낸다.

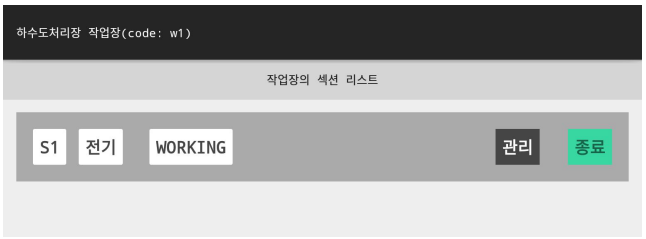

그림 17 작업장 및 섹션의 상태 관리 화면

관리감독자는 애플리케이션을 통하여 작업장 및 섹션의 state를 관리한다. 관리감독자는 작업장 및 섹션의 state를 작업자가 작업개시를 요청할 수 있게 활성화 시킨다. 작업자는 활성화된 작업장 및 섹션에서만 작업개시를 요청할 수 있다. 작업장 및 섹션의 상태는 ‘pending’, ‘working’, ‘finish’ 3가지 상태를 가진다. 그림 18은 관리감독자 애플리케이션의 메인화면들을 나타낸다.

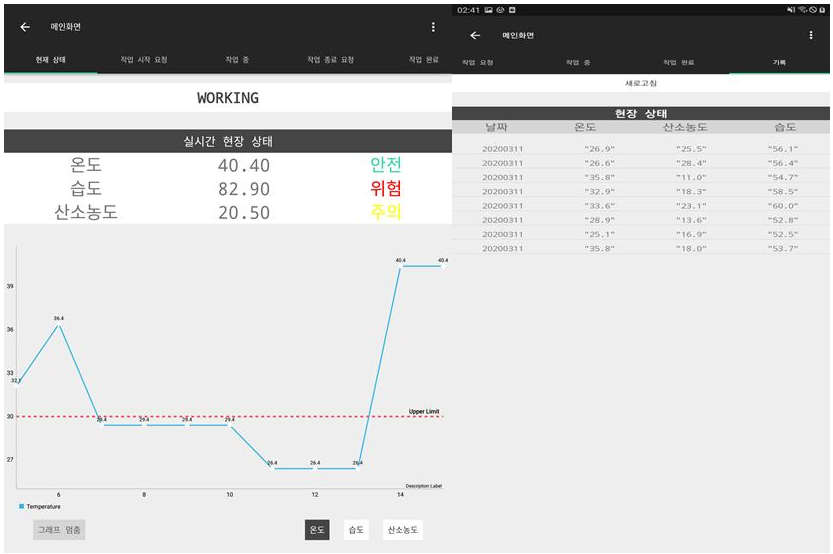

그림 18 메인화면의 현재 상태 화면과 전체 기록 화면

관리감독자는 메인화면에서 활성화된 작업장 및 섹션의 정보를 표와 그래프를 통하여 실시간으로 확인할 수 있다. 그림 18을 통해 안전 기준치에 따라 화면에 잘 나타나는지 테스트한 결과 화면을 볼 수 있다. 마지막 탭 화면에서는 작업 현장 상태의 전체 기록을 확인할 수 있다. 그림 19는 관리감독자 애플리케이션에서 작업자의 요청 리스트를 보여주는 화면과 이에 대한 승인을 진행하는 화면들을 나타낸다.

그림 19 작업자 상태에 따른 리스트와 작업 요청 승인 화면

관리감독자의 애플리케이션에서 작업자가 작업 개시 및 종료에 대한 요청을 한 리스트를 확인할 수 있다. '상세 정보 보기' 버튼을 누르면 해당 요청에 대한 상세 정보를 안전 기준치에 따른 결과와 함께 확인할 수 있다. 이를 확인하여 관리감독자는 작업 개시 및 종료에 대한 승인과 거절을 결정하여 진행한다. 나머지 탭에서는 작업을 진행 중인 작업자의 정보와 작업을 완료한 작업자의 리스트를 확인할 수 있다.

4.5 FairSafe의 검증

본 논문에서는 3개의 노드로 EOS 프라이빗 블록체인 네트워크를 구성하여 안전관리 플랫폼과 3개의 분산 애플리케이션을 통한 FairSafe를 검증한다. FairSafe의 검증을 진행한 환경은 다음과 같다.

· 3-Node (OS: Linux / ubuntu 18.04)
· EOSIO v2.0.0
· Android 기반 테블릿 2대 (작업자용, 관리감독자용)

Fair Safe의 검증은 센서 플랫폼과 안전관리 플랫폼 간 데이터 일치성을 확인하고 분산 애플리케이션의 UI별 기능과 블록체인 네트워크에 저장된 데이터의 일치성을 검증한다. FairSafe를 통하여 투명한 작업자 안전관리 프로세스를 검증한 시나리오는 다음과 같다.

(1) 관리감독자 애플리케이션을 로그인하여 작업장의 섹션 상태를 pending 상태에서 working 상태로 변경시킨다.
(2) 블록체인 데이터에서 working으로 변경되었음을 확인한다.
(3) 관리감독자 애플리케이션에서 관리하면으로 접속하여 작업장의 현장데이터를 확인한다.
(4) 블록체인 데이터에서 관리감독자 애플리케이션의 현장데이터와 일치함을 확인한다.
(5) 작업자 애플리케이션에서 현장 데이터와 신체데이터를 확인하고 작업시작을 요청한다.
(6) 블록체인 데이터에서 작업자의 신체데이터와 현장데이터의 일치함과 req_working 상태로 변경되었음을 확인한다.
(7) 작업자 애플리케이션에서 작업 시작 준비 완료 상태로 변경되었음을 확인하고, 관리감독자 애플리케이션으로 작업자 애플리케이션에서 작업 시작 준비 완료를 요청한 리스트를 확인한다.
(8) 관리감독자 애플리케이션에서 현장데이터와 작업자의 신체데이터를 확인하고 승인처리를 실행시킨다.
(9) 작업자 애플리케이션에서 작업 중 상태로 변경 및 블록체인 데이터에서 working 상태로 변경되었음을 확인한다.
(10) 관리감독자 애플리케이션에서 작업자의 신체데이터와 블록체인 데이터의 작업자 신체데이터의 일치함을 확인한다.
(11) 작업자 애플리케이션에서 작업 종료 요청을 실행하고, 블록체인 데이터에서 req_finish 상태로 변경되었음을 확인한다.
(12) 작업자 애플리케이션에서 작업 종료 준비 완료 상태로 변경되었음을 확인하고, 관리감독자 애플리케이션에서 작업 종료 요청 리스트 생성을 확인하고 승인처리를 실행시킨다.
(13) 작업자 애플리케이션에서 작업 종료 상태 변경 확인 및 블록체인 데이터에서 finish 상태로 변경되었음을 확인한다.
(14) 관리감독자 애플리케이션에서 종료된 작업자를 작업 대기상태로 변경함을 실행시키고, 블록체인 데이터에서 pending 상태로 변경되었음을 확인한다.
(15) 시험 횟수 : 5회

그림 20은 시나리오 중 9, 10번에 해당하는 데이터 일치성을 나타낸다.

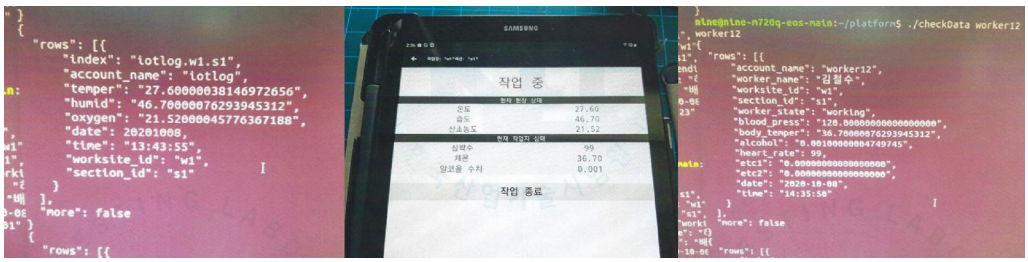

그림 20 검증 시나리오 중 데이터 일치성 확인 화면

5. 결론

본 논문에서는 밀폐 공간이나 건설 등의 작업 현장에서 작업자에게 투명하고 신뢰성 있는 작업자 안전관리 프로세스를 제공하는 블록체인 기반 안전관리 서비스인 FairSafe의 개발에 대하여 기술하였다. FairSafe는 수많은 데이터들을 다루는 사물인터넷 환경에 적합한 EOS 블록체인을 사용하여 개발하였고, 신뢰성 있는 작업 개시 및 종료 프로세스를 제안하고 이를 지원하기 위한 5개의 컨트랙트에 대하여 설명하였다. 제안된 투명한 작업자 안전관리 프로세스는 작업자가 작업을 시작하거나 종료를 하기 위하여 작업 현장의 상황정보와 자신의 신체정보를 확인하고 관리감독자에게 요청하는 과정과 이를 확인하고 승인하는 모든 과정을 블록체인을 통하여 신뢰성 있고 투명하게 제공한다.

또한 투명한 작업자 안전관리 프로세스를 진행하기 위하여 각 사용자의 역할에 따라 기능을 수행하는 3가지의 분산 애플리케이션에 대하여 설명하였다. 시스템 운영자 애플리케이션은 작업자와 관리감독자의 명시적인 인증을 위하여 어카운트에 권한을 부여하고 작업자의 안전관리에 관한 데이터들을 기준치에 따라 상,하한선을 설정한다. 권한을 가진 작업자와 관리감독자는 투명하고 신뢰성 있는 작업 개시 및 종료 프로세스를 작업자 및 관리감독자 애플리케이션으로 진행한다. 이를 통하여 각 사용자의 애플리케이션은 작업자 안전관리 프로세스 과정 및 상황 정보 데이터들을 모니터링 기능을 제공한다.

본 논문에서는 3개의 EOS 프라이빗 블록체인 네트워크를 구성하고 안전관리 플랫폼과 3개의 분산 애플리케이션으로 FairSafe의 유효성을 검증하였다. 검증을 위한 시나리오에 따라 센서 플랫폼과 안전관리 플랫폼 간 데이터 일치성을 확인하였고, 분산 애플리케이션의 UI별 기능과 블록체인 네트워크에 저장된 데이터의 일치성을 검증하였다.

본 논문에서 개발된 블록체인 기반 작업자 안전관리 서비스는 밀폐공간이나 건설 등의 상황정보에 따라 위험도가 높아지는 산업 현장에서 효과적으로 활용될 수 잇을 것이다. 특히, 작업의 흐름에 따른 데이터가 블록체인에 저장되어 상황정보나 작업자의 부주의 등의 다양한 이유로 발생할 수 있는 안전사고에 대하여 책임소재를 분명히 할 수 있을 것으로 기대된다. 또한, 안전사고와 관련된 상황정보들을 관리감독자만 확인할 수 있는 것이 아니라 작업자 또한 확인할 수 있기 때문에 서로 간의 신뢰관계가 향상되며, 개발된 기법은 여러 형태의 안전관리 시스템과 블록체인을 용이하게 결합하는 데 활용될 수 있다.

참 고 문 헌

[1] Min-Seung Jeong, Chang-Shin Lee, Woo-Hyeon Cho. (2019). Design of Integrated Safery System for Sealed Places.Journal of the Korea Institute of Information and Communication Engineering,23(1),97-102.

[2] Seung Woo Ha, Tai Woo You, Hwa Soo Kim, Dae Seong Lee, Sun Young Cho. (2019). Development of Safety Management System to Step Up Safeness of Furnace Scaffold Workers at Thermal Power Plant.Transactions of the KSME C Industrial Technology and Innovation,7(2),111-118.

[3] Kim SeungYong, In-Cheol Hwang, Kim DongSik, Moon Byungmoo, Oh Seytong. (2019). A Study on IoT/ICT Convergence Smart Safety Management System for Safety of High Risk Workers. Journal of the Korea Society of Disaster Information,15(1),39-48.

[4] Jung-ho Eom. (2020). An Architecture of a Smart Safety Management System to prevent safety Accidents in Workplace.Journal of Digital Contents Society,21(4),817-823.

[5] Kwang Hyoung Lee, Yong Hoon Jung. (2021). Blockchain-based access control and material management system.Journal of the Korea Academia-Industrial cooperation Society,22(11),442-448.

[6] Lee, Jung Jae. (2022). A Study on the Management Plan of Copyright Contracts Using Blockchain Technology : Blockchain-based copyright contracts centered on music copyrights.Journal of Korea Service Management Society,23(1),218-236.

[7] Kyeong-Dong Baek, Dong-Gue Park. (2022). Emergency RBAC System based on Blockchain.The Journal of Korean Institute of Information Technology,20(3),79-93.

[8] Ethereum, https://ethereum.org/ko/

[9] Eos, https://eos.io/eos-public-blockchain/

[10] H. M. Eom, K. H. Kim, J. H. Jin, H. S. Koo, and M. J. Lee. (2019). "Account Management Tool for DApp on the EOS Blockchain", In: Proceedings of Convergence Research Letter, Vol. 5, No. 1, pp. 311-314.

[11] D. Larimer. (2014). "Delegated proof-of-stake (dpos)", Bitshare whitepaper.

[12] Liu, Jieli, et al. (2022). "Understanding the Decentralization of DPoS: Perspectives From Data-Driven Analysis on EOSIO." arXiv preprint arXiv:2201.06187.

[13] EOSIO, https://eos.io/

[14] Grigg, Ian. "Financial cryptography in 7 layers." International Conference on Financial Cryptography. Springer, Berlin, Heidelberg, 2000.

[15] EOS Canada, https://www.eoscanada.com/

[16] LEE YONG GYU.(2021).A Study on the Factors Influencing Acceptance Intention and Acceptance Behavior of Technologies Related to the 4th Industrial Revolution and Smart Factory. Journal of Convergence for Information Technology, 11(4),1-18.

[17] Tae-Jong Kim, Dong-Yoon Lee. (2021). A Study on the Necessity of Smart Factory Application in Electronic Components Assembly Process . Journal of Convergence for Information Technology, 11(9), 138-144.

[18] Gwang-Bae Kim, Je-Dong Lee, Eui-Jeong Kim, Yongtae Shin. (2021). A Study on Architecture for Building Smart Safety Management Systems Based on Construction site IoT.Jounal of The Korea Society of Information Technology Policy & Management,13(1),2305-2311.

[19] Node.js, https://nodejs.org/

[20] Eos.js, https://developers.eos.io/manuals/eosjs/latest/index

ABSTRACT

A Blockchain-based Service Supporting Reliabe Worker Safety Mnanagement Processes

In general, the use of a worker safety management system is emphasized at work sites where context information on the sites and the health status of workers greatly affect the safety of workers. In addition, the series of processes that managers check based on such information are also very important factors for worker safety management. But, in the current worker safety management systems where sensitive context informations related to worker safety are stored and managed only on a single server inside the company, there is a high possibility of forgery and loss of data, leading to workers' distrust of the system. So, it is necessary to reliably store and manage information such as the contexts of work sites and health status of workers and safety management processes. It is also desirable to ensure that such information and processes are transparently provided to enterprises and workers.

This paper describes the development of FairSafe, a blockchain-based service that reliably provides worker safety management processes. FairSafe consists of a safety management platform that supports the execution of smart contract functions through communication with the EOS private blockchain, and three decentralized applications to perform the functions for each user. The administrator application grants managers and workers appropriate account privileges to perform their functions, and specifies safety standards for work site conditions. The worker application and the manager application cooperate with each other to provide the functionality to perform all series of processes from request and approval of work initiation to termination as blockchain transactions. Informations such as safety management processes and the contexts of work sites and health status of workers performed through these applications are explicitly stored and managed in EOS smart contracts through the safety management platform.

Key words : Worker Safety Management System, Blockchain, EOS, Decentralized Application, FairSafe