

최고의 오늘을 꿈꿉니다.

**Anywhere,
Anything,
Adaptable**

**Server Developer 반영환
Portfolio**

SUITE

프로젝트 기간 : 2023.05 – 2023.12

진심을 보증합니다.
당신의 학습 파트너를
찾는 공간



 SUITE

Study With
Trust & Endeavor

Suite

스마트 컨트랙트를 이용한 스터디 보증금 계약서 기반 스터디 구인 플랫폼 dApp

담당 직무	백엔드 개발자 (블록체인 / 아키텍처 담당) (3인)
기술 스택	EKS, Kafka, Spring Boot, Express.js, Solidity, Ganache
핵심 구현사항	<ul style="list-style-type: none">• 마이크로 서비스 아키텍처를 위한 EKS 환경 구성• 서비스 간 비동기 통신과 보상 트랜잭션을 위한 Kafka 클러스터 구성• AWS 인프라 형상관리를 위한 Terraform 도입• Github Action을 사용한 CI 구축• Spring Boot를 사용해 API 서버를 구축하고 Express.js 의 Ethers.js 라이브러리를 사용해 블록체인 트랜잭션 API 서버 구축• Solidity를 사용한 스마트 컨트랙트 구현 및 Ganache와 remix를 활용한 로컬 환경 테스트 진행
깃허브 주소	https://github.com/orgs/SWM-TheDreaming/repositories

좋은 기획이, 좋은 개발을 만듭니다.

Design Thinking

Pain Point

- 보증금 사기성 스터디 그룹으로 금전적 피해를 입은 사람들이 존재한다
- 보증금 관리에 대해 부담감을 느끼는 사람들이 있다
- 체계적인 스터디 목표 관리를 위한 애플리케이션의 부재를 느끼는 사람들이 있다.

How Might We?

성공적인 스터디를 위해 보증금을 걷는 사람들의 불안과 부담을 줄이자

Solution

스마트 컨트랙트를 이용한 스터디 보증금 계약서를 사용자들에게 발급하고, 장부에 기록된 내용들을 바탕으로 보증금 지급 의무 시행을 보장하는 스터디 구인 플랫폼을 만들자.

Bussiness Model

- 보증금 관리에 대한 금융(이자) 수익
- 보증금 반환조건에 만족하지 못 한 사용자의 보증금에서 플랫폼 사용비 차감

Target(TAM, SAM, SOM)



Competitors

	공작소	얼퐁타	위스터디
스터디 형태	개인/모집/경쟁	개인/경쟁	그룹/모집
동기부여 여부	o (일일 인증)	o (통계)	x
UX 만족도	High	High	Low
타겟층	10대 학생	학생	대학생
스터디 목표 관리 도구	o (일일 인증)	x	o (채팅)

기획의 검증은 반드시 필요합니다.

Is the solution appropriate?

스마트 컨트랙트가 법적 효력을 가질 수 있는가?

- 전자 서명법 제 6조 1항
국가는 생체인증, 블록체인 등 다양한 전자서명수단의 이용 활성화를 위해 노력해야 한다.
- 전자 서명법 제 2조 2항
전자 서명이란 서명자의 신원과 서명한 사실이 적힌 전자적 형태의 정보를 말한다.

현금 흐름으로 인한 세금 문제가 없는가?

- 세무사 미팅을 통한 해결
보증금은 수익이 아닌 부채기 때문에 세금 대상이 아니다.
부가 수익의 경우 부가세 10%를 공제한 뒤 해당 금액을 수익으로 책정한다.
책정된 수익의 6.6% ~ 8.8%의 세금을 원천징수한다.
사용자에게 돌아가는 부가수익이 50,000원을 초과시 원천 징수가 발생하므로 주의.

말은 한 번만 해야 합니다.

산출물 목록

유저 스토리	필요 기능 식별 및 Jira Story와 맵핑
와이어 프레임	요청 단위 기준 서비스 대분류
서비스 정의서	서비스의 기능과 요구사항 작성
서비스 다이어그램	서비스별 Component, Operation, DI 식별 및 Event Storming Relationship 확인 및 API 정련
API First Design	API First Design을 통한 Front / Back 의 비동기적 작업 시작 및 API 정련
Sequence 다이어그램	서비스간 Event의 흐름과. 비동기 통신 요구 지점을 식별 및 API 정련
Kafka 다이어그램	SAGA 패턴의 Choreography 방식으로 보상 트랜잭션을 구상하며 Topic Name 및 Producer / Consumer 관계 구성

스마트 계약을 이용한 스터디 보증금 계약서 기반 스터디 구인 플랫폼 dApp

유저 스토리

ID	BUS-110 : 버퍼 오버플로
User Story	수신된 그램 데이터를 버퍼에 저장 한 후, 한꺼번에 데이터 그램을 출력하는데 버퍼가 넘치거나, 버퍼를 넘겨주는 버퍼 오버플로가 발생하지 않음.
Related Stories	
Acceptance Criteria	<p>AC1 버퍼를 넘겨주는 버퍼를 넘치는 경우 에러를 반환하여 처리에 반영되고,</p> <ul style="list-style-type: none"> - 버퍼 포인터 (bugsn0t@gmail.com) - 버퍼 크기 (2 바이트) - 버퍼를 넘겨 줄 때 (10000) <p>AC2 Guest 버퍼를 넘겨주는 버퍼를 넘치는 경우 에러를 반환하여 처리에 반영되고,</p> <p>AC3 Host 버퍼를 넘겨주는 버퍼를 넘치는 경우 에러를 반환하여 처리에 반영되고,</p>
Error Code	<div> <div>BUS-E001 / BUS-E002 / BUS-E003 / BUS-E004 / BUS-E005 / BUS-E006 / BUS-E007 / BUS-E008</div> </div>
Prototype or design	
Additional Details	

와이 어 프레임



서비스 정의서

[illegible]

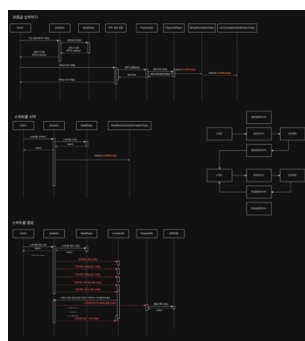
서비스 다이어그램



API First Design

Study API 항목				
API	URI	Request	Response	Description
자신에게 댓글 추가하기	study/detail/comments	댓글 ID 자신 ID	추가된 댓글	post
댓글 등록	study/comments	댓글 ID 자신 ID	댓글 등록 여부	post
댓글 삭제	study/comments/detail	댓글 ID 자신 ID (선택) 자신 ID	댓글 삭제 여부	post
댓글 제출	study/comments/submit	댓글 ID 자신 ID 댓글 ID	댓글 제출 여부	post
출석체크	study/detail/attendance	댓글 ID 자신 ID 자신 ID	출석체크	get
출석하기	study/attendance	댓글 ID 자신 ID 자신 ID	출석체크 여부	post
내 게시글 열람하기	study/my-boards	댓글 ID	내 게시글 열람	post
타인 게시글 열람하기	study/boards	댓글 ID	타인 게시글 열람	post
내 댓글 열람하기	study/my-attendance	댓글 ID	내 댓글 열람	post
타인 댓글 열람하기	study/comments/attendance	댓글 ID	타인 댓글 열람	post
게시글 등록	study/boards	댓글 ID	게시글 등록 여부	post
게시글 열람 목록	study/boards	댓글 ID	게시글 열람 목록	post
게시글 열람 상세	study/boards/detail/boards	댓글 ID	게시글 열람 상세	post
게시글 열람 상세	study/boards/detail/boards	댓글 ID	게시글 열람 상세	post
게시글 열람 상세	study/boards/detail/boards	댓글 ID	게시글 열람 상세	post

Service Flow



Kafka 다이어그램



서비스가 분리되면서 DB도 독립적으로 분리되었습니다.
스터디에는 가입되지만, 보증금 처리과정에서 **롤백**이 발생해
독립된 DB구조상 **스터디 참가 상태는 그대로 유지되는 경우**를 고려,
SAGA패턴의 Choreography 방식으로 보상 트랜잭션을 설계했습니다.
또한 Rollback 실패 시, Slack에 알림을 보내 개발자가
이를 수동 처리할 수 있도록 고려했습니다.

시스템 구성도



Infra as Code

VPC, Subnet, Routing Table, NAT, IGW, EFS, ECR의 형상 관리를 Terraform으로 처리.

MSA

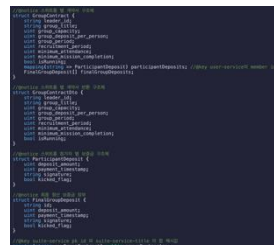
Managed K8S, EKS와 Kafka를 조합해 분산 서비스 환경을 구성

Continuous Integration

Github Action과 ECR을 사용해 CI 구성

블록체인

Contract Design



빠른 트랜잭션과 한 번의 트랜잭션 API 요청을 위한 DDD 설계

Mapping 위주의 데이터 구조 설계로
빠른 Read 처리 가능

기획의 Solution을 위한 기술적 도전

Network



트랜잭션 비용과 안정성을 위한
Polygon Test net, Main net사용

Trouble Shooting



One Contract, One Group 관계를 포기하고 mapping관계로 리팩토링하고 DDD 설계로 변화했습니다.

또한 서명 알고리즘 라이브러리를 Web3.js 에서 Ethers.js로 변경해 비용과 시간을 대폭 감소시켰습니다.

Itemize

프로젝트 기간 : 2024.03.02 – 2024.11.10

서울과학기술대학 ITM 전공 Capstone Design Project



ITEMIZE



SEOULTECH

서울과학기술대학교

SEOUL NATIONAL UNIVERSITY OF SCIENCE & TECHNOLOGY

Itemize

스마트 컨트랙트를 이용한 증권 토큰 기반 클라우드 펀딩 거래 플랫폼

담당 직무	백엔드 개발자 (블록체인 / 아키텍처 담당) (2인)
기술 스택	On-premise K8s, Kafka, Spring Boot, Express.ts, Solidity
핵심 구현사항	<ul style="list-style-type: none">마이크로 서비스 아키텍처를 위한 On-premise Kubernetes Cluster 구성서비스 간 비동기 통신과 보상 트랜잭션을 위한 Kafka 클러스터 구성Solidity를 사용한 ERC-721 / ERC-20 스마트 컨트랙트 구현Github Action을 사용한 CI 구축 및 ArgoCD를 사용한 CD 구축Spring Boot를 사용해 API 서버를 구축하고 Express.ts의 Ethers.js 라이브러리를 사용해 블록체인 서버 구축팀원간 소통을 위한 다양한 문서 산출물 제작
깃허브 주소	https://github.com/Capstone-Design-2024

좋은 기획이, 좋은 개발을 만듭니다.

Design Thinking

Pain Point

기존 클라우드 펀딩 플랫폼에서는 투자 방식이 고정 금액 납부뿐이어서
균중 자금 조달이란 목표를 실현하기에 환경적 제약이 심하다.

펀딩 상품의 희귀성 및 가치 상승에 대한 생산자, 구매 권리자의 행동이 제한적이다.

How Might We?

클라우드 펀딩의 의미를 실현하고 펀딩 개설자와 투자자가 할 수 있는
행동의 제약을 줄여줄 방법은 무엇일까?

Solution

스마트 컨트랙트를 활용한 증권 토큰을 통해 구매 권리 증서를 발행하고
상품의 지분에 따른 거래를 ERC20 토큰을 사용해 자유롭게 이루어지게 할
수 있는 거래소를 만들자

Bussiness Model

증권 토큰 거래에 따른 수수료
기존 플랫폼의 BM과 동일한 판매 수수료

기획의 검증은 반드시 필요합니다.

Is the solution appropriate?

기대 효과 및 결과

군중에 의한 자금 조달의 목적을 가지는 클라우드 펀딩이 그 의미를 실현하기
위해서 플랫폼과 펀딩 참여자들 사이의 일방향적인 관계를 타파하고, 관계를 새로 이
정의함으로써 클라우드 펀딩 제품의 가격을 플랫폼이 아닌 시장의 가격으로
결정할 수 있게 될 것입니다. 이로써 사회 총잉여는 증가합니다.

펀딩 개설자는 좋은 상품을 시장에 공개하고, 이를 클라우드 펀딩 투자를 받고,
시장에서 펀딩 참여자들 사이에 거래가 활성화되고, 가치가 상승할 때마다 추가적인
수익을 창출할 수 있게 됩니다.

이로써 더 양질의 제품이 시장에 공개될 수 있는 동기가 마련됩니다.

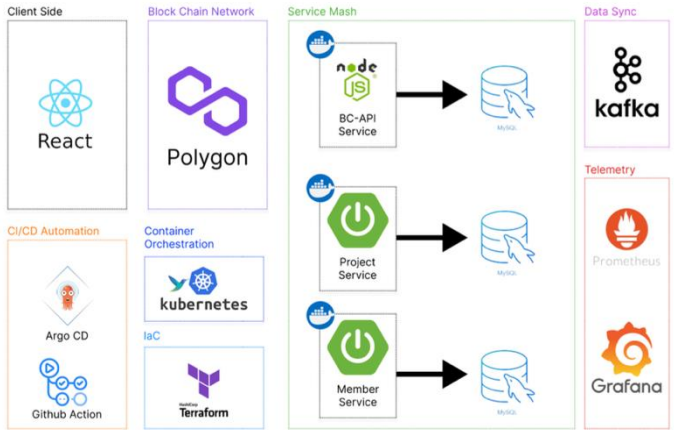
펀딩 참여자는 펀딩이 마감한 제품에 대해서 플랫폼에 종속되지 않고,
시장 가격에 형성된 클라우드 펀딩 티켓을 구매해 자신의 욕구를 충족 시킬 수
있습니다.

본 연구를 통해 개발된, 전자 지갑을 활용한 혁신적인 클라우드 펀딩 플랫폼은
기존의 문제 상황에서 발견한 인사이트를 탈중앙화 거래소 도입을 통해 해결하고자
하며 이를 통해 새로운 사용자 가치를 만들어낼 것입니다.

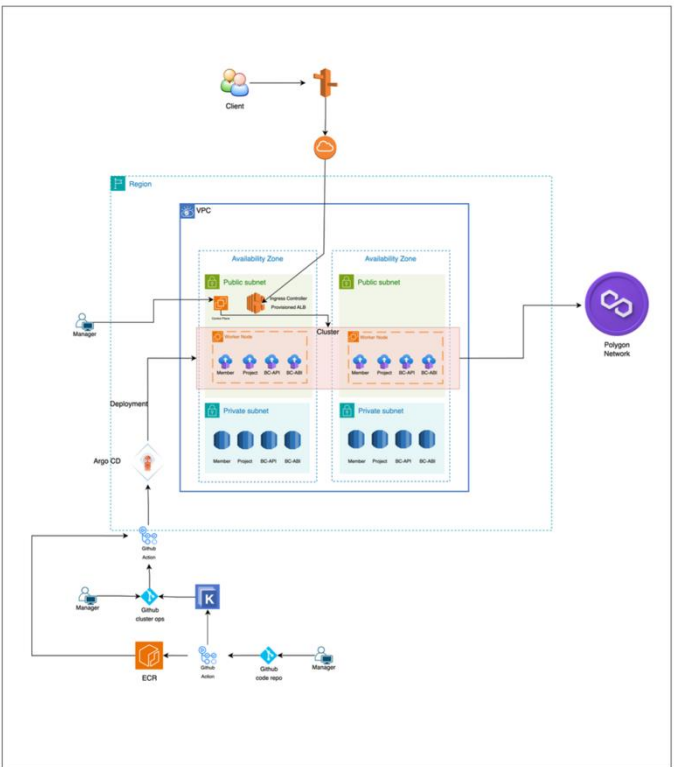
클라우드 펀딩 참여도가 높아지면서 생태계가 커지고, 수요와 공급에 맞추어
합리적으로 형성된 시장 가격으로 총 잉여가 향상돼 더욱 건강한 시장이 형성될
것으로 기대됩니다.

서비스 아키텍처

서비스 매쉬



서비스 아키텍처



지금 여기, AED

프로젝트 기간 : 2023.11.09 – 2023.11.10

대구시 ABB 블록체인 메이커톤 최우수상 수상작



**Team Moment,
Save Life in this moment**



**ABB
Hackathon**

지금 여기, AED

DID와 IoT를 이용한 실시간 AED 재고 조사와 관리자 dApp을 이용한 상태 관리표의 전자문서화

담당 직무	백엔드 개발자 (BlockChain DID API) (3인)
기술 스택	Spring Boot, Arduino, Kafka, IoT MQTT
핵심 구현사항	<ul style="list-style-type: none"> 대구 체인 API가 제공하는 DID 발급 서비스 (Issuer) 분석 및 추가적인 암호화 기능 구현 양방향 암호화를 통해 만든 사용자의 공개키를 블록체인상에 온체인 시켜 verifier 기능 간접 구현 IoT와 앱 프론트가 DID 서비스를 사용할 수 있도록 Spring boot를 이용한 API 개설
깃허브 주소	https://github.com/ABB-hack-moment

좋은 기획이, 좋은 개발을 만듭니다.

Design Thinking

Pain Point

시설물 관리를 종이로 할 때 관리에 어려움이 있다.

관리 대상이 외부에 노출돼 관계자 뿐만 아니라 다른 사람들이 점검 내용을 조작 할 수 있다.

현재 응급 의료 포털 사이트와 애플리케이션을 통해 AED의 위치와 상태를 확인 할 수 있지만 정보가 누락 되는 경우가 많다.

How Might We?

공개적이면서 안전하게 시설물 상태 관리를 할 수 있게 하면서 AED의 실시간 정보를 사용자와 관리자에게 정확하고 신속하게 보여줄 수 있는 해결책이 무엇일까?

Solution

DID 서비스 구조를 사용해 AED 점검에 대한 정확성과 무결성을 보장하고, IOT + MQTT 와 Kafka를 활용해 AED의 존재 유무를 실시간으로 사용자들에게 보여주는 기술을 만들자.

기획의 검증은 반드시 필요합니다.

Problem is the Problem?

AED의 관리가 정말 부실한가?

전체 기기의 33% 작동 불량

출처 : 국민안전처
"아파트 등 자동심장충격기 관리부실 33%...155곳 761대 작동불량"

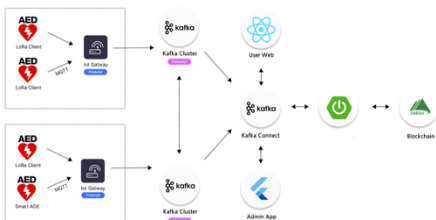
뉴스룸 : 국민안전처
"자동심장충격기 설치된 경기도 아파트 3곳 중 1곳 '작동 불량'"

위급 상황에 쓰이지 못하는 AED가 다수 존재

"안 보이고, 숨겨 놓고"...AED 관리부실 '도마뱀'
생명 소생 '자동심장충격기' 방치...위급 상황 '유명무실'
2022년 11월 03일(목) 20:00

AED는 위급 상황시 환자를 살릴 수도 죽일 수도 있는 아주 중요한 시설물이기에 반드시 잘 관리돼야 한다.

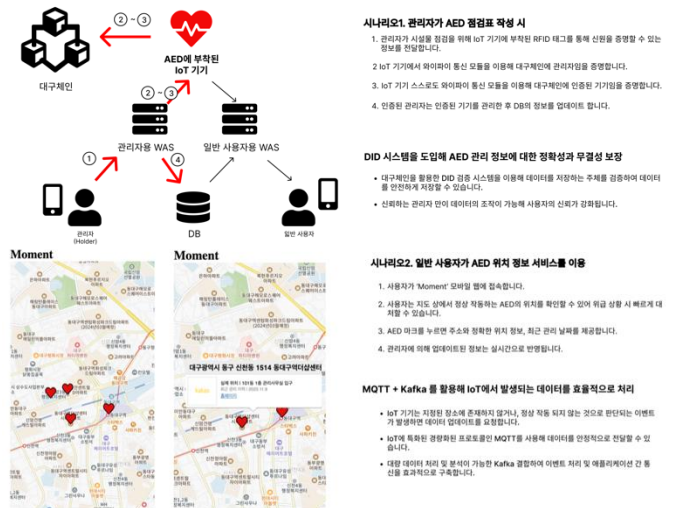
시스템 구성도



말은 한 번만 해야 합니다.

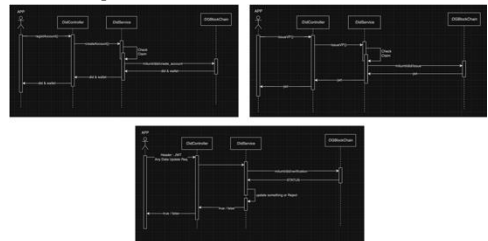
시나리오 작성

AED 관리 웹 서비스 서비스 흐름도

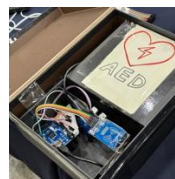


짧은 시간 내에 진행되는 메이커톤 특성상 아이디어를 모두가 공유할 필요성을 느꼈습니다. 따라서 저희 팀은 시나리오를 작성해 모두가 하나의 주제를 바라보도록 해 개발 진행 효율을 높였습니다.

대구 체인 Sequence



프로젝트 결과



프로토 타입을 통한 사용자 설득



블록 체인을 활용한
안전 시설물 통합 서비스 제공



기획에 대한 믿음과 빠른 판단에 따른 개발 시작으로 만족스러운 결과를 얻어냈습니다.

Keep Your Endeavor

너의 노력을 지켜라

프로젝트 기간 : 2022.09. – 2023.12.

과학기술정보통신부 주관 OSAM 오픈 소스 군 장병 해커톤 육군참모총장상 수상작



Keep Your Endeavor

blockchain-based military credential authentication service

2022

군장병 공개SW 역량강화 교육

“ 공개SW를 몰라?
오히려 좋아, 가보자고! ”



Keep Your Endeavor

IoT기기를 활용, 평가 내역을 자체 개발한 블록체인 네트워크를 통해 관리 되는 병 진급 시스템

담당 직무	백엔드 개발자 (API service & IoT) (2인)
기술 스택	Node.js, Redis, Docker, Arduino
핵심 구현사항	<ul style="list-style-type: none">블록체인 Node가 올라갈 수 있는 Docker 환경 구성Channel Node API에 접근할 수 있는 Node.js 기반 서버 구성병자적 인증 평가의 핵심 기능인 IoT 기기 설계 및 구성
깃허브 주소	https://github.com/osamhack2022-v2/CLOUD_APP_IOT_KeepYourEndeavor_Moment
깃 북 주소	https://moment-ky2.gitbook.io/keep_your_endeavor

좋은 기획이, 좋은 개발을 만듭니다.

Design Thinking

Pain Point

- 장병들의 자격 인증 평가의 결과가 올바르게 관리되고 있지 않다.
- 장병들은 자신의 자격 인증 평가 결과를 쉽게 확인할 수. 없다.
- 장병들의 자격 인증 평가가 부정행위와 청탁을 기반으로 이루어지고 있다.

How Might We?

장병들이 자신의 자격 인증 평가 결과를 모바일로 안전하게 확인하면서 제 3자에 의한 부정행위와 청탁으로 노력하지 않은 장병이 이익을 보지 않게 하고 중간 관리자없이 결과를 관리할 수 있는 방법이 없을까?

Solution

직접 구축한 KY2 체인을 통해 측정, 기록, 열람까지의 과정을 통합하고, 블록체인과 연계해 중간 과정에서 벌어질 수 있는 누락, 조작 사고를 예방하자!

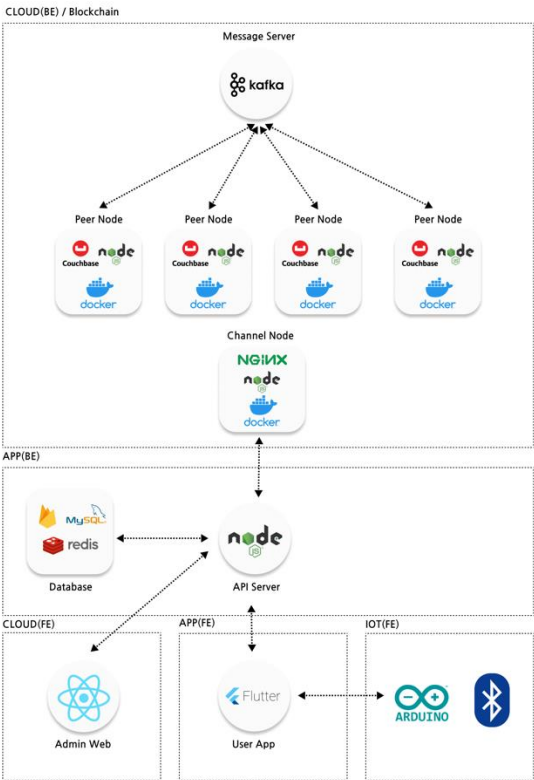
기획의 검증은 반드시 필요합니다.

Solution is adaptable?

병 자기능력 평가는 육군에 한정돼 있다. 시장이 너무 작지 않나?

육군에 한정된 도메인인 것을 인지, 솔루션 포인트를 진급 시험에서 자격 인증 평가로 확장, 군 내부적으로 상훈, 교육 관리에 활용될 수 있는 플랫폼 개발로 방향을 전환

시스템 구성도



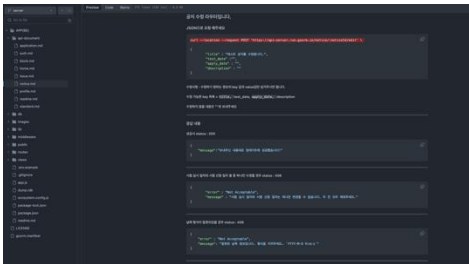
말은 한 번만 해야 합니다.

Git Book



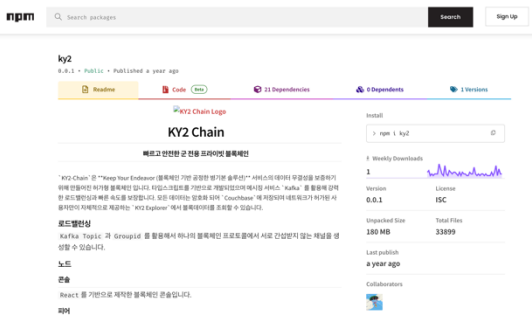
기획의 정리와 각 분야를 어떻게 사용할 것이며 기능/비기능 요구사항들을 Git Book을 통해 정리해 팀원과 공유했습니다. 새로운 팀원도 문서를 통해 빠르게 Follow up 할 수 있도록 상세히 작성했습니다.

API First Design



API 에러에 대한 능동적인 대처를 위해 모든 에러 상황에 대한 상세한 설명을 문서에 담아 서로 동시간 작업이 거의 불가능한 군대라는 환경속에서 작업 효율을 높이 고자 했습니다.

NPM Library Deployment



자체 개발한 KY2 Chain을 어디서나, 누구든지 활용했으면 하는 바람에 NPM에 라이브러리를 배포한 경험이 있습니다. 상세한 설명으로 그 누구든 블록체인 환경에 참여할 수 있습니다.

Keep Your Endeavor

IoT기기를 활용, 평가 내역을 자체 개발한 블록체인 네트워크를 통해 관리 되는 병 진급 시스템

API Server

권한 등급	이용 가능 서비스
개설자	자격 OPEN/CLOSE 등 자격 관련 모든 서비스까지 접근 가능
등록자	자격 관련 서비스를 제외한 공지 관련 서비스까지 접근 가능
일반	공지에 대한 신청과 자신의 자격에 대한 조회 권한

- JWT + Express.js middleware를 사용한 인증 방식
- Token을 Redis에서 저장해 관리, Expire Time과 Token Drop을 통한 사용자 세션관리

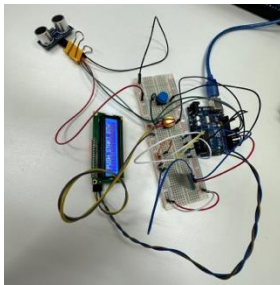
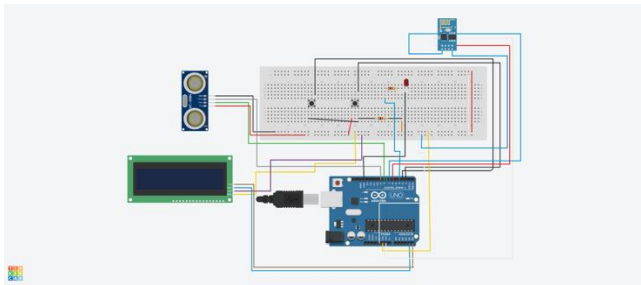
DB

- MySQL: 유저의 민감정보, Entity Meta Info 등을 저장하는데 사용
- Redis: Token 처리를 위해 사용했으며 Redis Cloud 사용
- Firebase: 등급을 매기는 기준표 저장소로 사용했으며 시험, 이슈마다 데이터 컬럼이 다르기 때문에 사용

KY2 Private block chain

- 자체 개발한 Private Block Chain Network
- Docker Compose를 사용해 1 User 1 Node 구현
- 군 부대 서버의 한계로 8개의 User까지 추가 가능
- Peer Node / Channel Node 로 노드의 역할을 분할해 외부 네트워크 노출을 최소화

KY2 IOT



- 초음파 센서를 이용한 기록 측정
- 블루투스 모듈을 이용한 서버 통신
- 사용자 APP을 거쳐 결과를 중간 관리자 없이 즉시 블록체인에 온체인

Result



군부대란 열악한 환경속에서 시간을 쪼개 공부를 하면서 처음 프로젝트를 진행, 우수한 멘토 진들의 도움과 원활한 업무 수행을 위해 끝없이 노력한 결과 육군 참모총장상이라는 우수한 성적을 거두게 되었습니다.

프레임워크를 사용한 서버 개발과 IoT 개발 등 모든 것이 처음이라 서툴었지만
어려운 것은 있어도 만들 수 없는 것은 없다. 라는 생각을 가지게 해준 값진 경험이었습니다.

Sgether

Study Together

프로젝트 기간 : 2023.01. – 2023.5.

Google주관 GDSC Solution Challenge **Global Top 100**

SGETHER
With AI



Google Developer Student Clubs

Sgether

비대면 AI 감독관 스터디 플랫폼

담당 직무	백엔드 개발자 (4인)
기술 스택	Node.js, Redis
핵심 구현사항	<ul style="list-style-type: none">Express.js를 사용한 API 서버 개발WebRTC를 위한 Socket.io 서버 개발
깃허브 주소	https://github.com/osamhack2022-v2/CLOUD_APP_IOT_KeepYourEndeavor_Moment

좋은 기획이, 좋은 개발을 만듭니다.

UN SDGs

4. Quality Education & 10. Reduced Inequality

Target

특정 지리적 또는 환경적 제약으로 인해 특정 수준의 교육을 받을 수 없는 어린이들

Problem

대면 수업을 진행하기 어려워 원격 수업을 진행해야 한다.
그러나 학생들의 집중도를 모니터링하기 어렵다.

Solution

AI 감독관을 도입한 가상 스터디 플랫폼을 통해서 학생들의 집중도를 확인하고 통계를 내 관리감독에 도움을 주는 서비스를 만들자

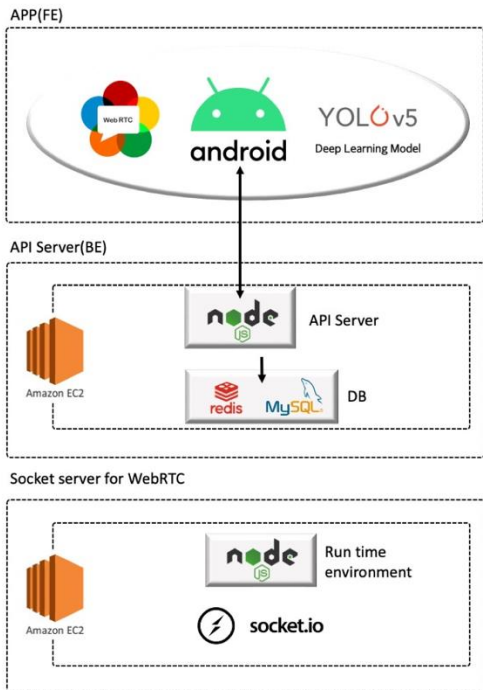
기획의 검증은 반드시 필요합니다.

Solution is adaptable?

아이들의 수업 집중도와 교육의 질을 플랫폼이 높여줄 수 있나?

어린이들은 Sgether 플랫폼을 사용할 때 서로 연결되고 동기부여를 느낄 수 있다. 따라서 학교와 떨어져 있는 감정을 줄여 하나의 소속감을 느끼게 할 수 있다. 또한 AI 학습 감독을 통해 자신의 학습 상태를 객관적으로 파악해 수업 집중도를 높일 수 있다.

시스템 구성도



말은 한 번만 해야 합니다.

API First Design

Group	
GET	/group/ 스터디 그룹 전체 불러오기
POST	/group/ 스터디 그룹 생성
PUT	/group/{roomId} 스터디 그룹 수정
DELETE	/group/{roomId} 스터디 그룹 삭제
GET	/group/{roomId} 특정 스터디 그룹 불러오기

Swagger와 Express.js를 연동해 협업하는 관계에서 빠른 피드백을 주고 받을 수 있게 문서화하였습니다.

API Server

- JWT + Express.js middleware를 사용한 인증 방식
- Token을 Redis에서 저장해 관리, Expire Time과 Token Drop을 통한 사용자 세션관리

DB

- MySQL: 유저의 민감정보, Entity Meta Info 등을 저장
- Redis Cloud: Token Life Cycle을 위해 사용

Certified

Solution Challenge

Google Developer Student Clubs

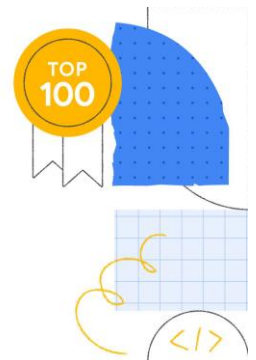
Top 100 | Global

Younghwan Phan

is hereby awarded this Certificate for successfully becoming a Top 100 Finalist in the 2023 Solution Challenge.

Rachel Francois

Google Developer Student Clubs, Global Program Lead



개발을 하면서 지속적인 수정과 리팩토링을 진행하며 다시 한 번 디자인 패턴의 중요성에 대해 생각하게 되는 계기가 되었습니다. Express.js 에 MVC 패턴의 도입을 고민하면서 코드의 역할 분할과 모듈화의 중요성을 몸소 깨달았습니다.

비록 GDSC 멤버는 아니었지만 좋은 기회를 얻어 Solution Challenge에 참가하게 돼 좋은 결과를 얻었습니다.

항상 제 주변의 문제에 대해서만 생각해오다가 처음으로 제 3국가의 문제점에 대해 관심을 가지고 자료를 조사하며 더불어 살아가는 세상에 대한 고민을 하게 된 계기가 되었습니다.

AI 도메인에 대해서 무지하였으나 AI 개발 팀원과 함께 개발을 진행하면서 새로운 도메인을 알아가는 재미도 있었습니다.