**캡스톤 디자인 I**

**종합설계 프로젝트**

|  |  |
| --- | --- |
| 프로젝트 명 | 여기모영 |
| 팀 명 | 꾹꾹 |
| 문서 제목 | 결과보고서 |

|  |  |
| --- | --- |
| **Version** | 1.1 |
| **Date** | 2022-05-16 |

|  |  |
| --- | --- |
| **팀원** | 안 성열 (조장) |
| 김 상윤 |
| 민 태식 |
| **지도교수** |  |

|  |
| --- |
| **CONFIDENTIALITY/SECURITY WARNING**  이 문서에 포함되어 있는 정보는 국민대학교 소프트웨어융합대학 소프트웨어학부 및 소프트웨어학부 개설 교과목 캡스톤 디자인I 수강 학생 중 프로젝트 “여기모영”을 수행하는 팀 “꾹꾹”의 팀원들의 자산입니다. 국민대학교 소프트웨어학부 및 팀 “꾹꾹”의 팀원들의 서면 허락없이 사용되거나, 재가공 될 수 없습니다. |

**문서 정보 / 수정 내역**

|  |  |
| --- | --- |
| **Filename** | 결과보고서-여기모영.doc |
| **원안작성자** | 김상윤, 민태식, 안성열 |
| **수정작업자** | 김상윤, 민태식, 안성열 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 수정날짜 | 대표수정자 | Revision | 추가/수정 항목 | 내 용 |
| 2022-05-04 | 안성열 | 1.0 | 최초 작성 | 목차 및 1번 내용 작성 |
| 2022-05-16 | 안성열 | 1.1 | 초안 작성 | 2, 3번 내용 작성 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

목차

[1 기존 프로젝트 4](#_Toc103681848)

[1.1 기존 프로젝트 폐기 이유 4](#_Toc103681849)

[2 프로젝트 개요 5](#_Toc103681850)

[2.1 프로젝트 추진 배경 5](#_Toc103681851)

[2.2 프로젝트 조직도 6](#_Toc103681852)

[3 프로젝트 세부 설명 6](#_Toc103681853)

[3.1 이미지 처리 6](#_Toc103681854)

[3.2 블록체인 6](#_Toc103681855)

[3.2.1 스마트 컨트랙트 7](#_Toc103681856)

[3.3 웹 어플리케이션 서버(Node.js) 8](#_Toc103681857)

[3.3.1 사용 모듈 8](#_Toc103681858)

[3.3.2 컴포넌트 9](#_Toc103681859)

[3.3.3 비즈니스 로직 9](#_Toc103681860)

[3.4 앱 어플리케이션(React-Native) 10](#_Toc103681861)

[3.5 데이터베이스 10](#_Toc103681862)

[3.6 AWS EC2 10](#_Toc103681863)

[4 기능별 흐름 설명 11](#_Toc103681864)

[4.1 회원가입 11](#_Toc103681865)

[4.2 로그인 12](#_Toc103681866)

[4.3 참가한 모임 조회 13](#_Toc103681867)

[4.4 모임 생성 14](#_Toc103681868)

[4.5 모임 참가 15](#_Toc103681869)

[4.6 모임 이동 16](#_Toc103681870)

[4.7 회비 사용 내역 조회 17](#_Toc103681871)

[4.8 총무 추가 17](#_Toc103681872)

[4.9 회비 추가 18](#_Toc103681873)

[4.10 영수증 추가 19](#_Toc103681874)

# 기존 프로젝트

기존 프로젝트명은 국민을 국회로 이다. 국민을 국회로는 입법예고제도를 통해 공시된 발의법안을 사용자에게 제공하고, 블록체인 기반의 투표를 통해 여론을 형성하는 모바일 어플리케이션 형태의 서비스이다.

입법예고제도란 법안의 발의되어 심사를 통해 최종적으로 공포되는 과정 동안 국민들의 의견을 수렴하기 위해 마련된 제도이다.

기존 프로젝트의 제작 배경은 다음과 같다.

* 새로 발의된 법안은 ‘국회 웹사이트’, ‘정부 웹사이트’ 두 가지의 방법으로 게시된다. 새로 발의된 법안을 하나의 웹 사이트에서 찾아볼 수 없다.
* 각 웹사이트에는 국민의 의견을 수렴하는 기능이 마련되어 있지만, 현재 찬성, 반대 등의 의견이 90% 이상을 차지한다.

기존 프로젝트에서는 다음과 같은 해결책을 마련했다.

* 국회 홈페이지, 정부 홈페이지의 게시된 법안을 크롤링하여 모바일 어플리케이션의 형태로 사용자에게 제공한다.
* ‘찬반 투표 기능’과 ‘세부 의견 작성’으로 기능을 분리한다. 블록체인을 사용하여 찬반 투표 과정에서 투표 결과와 비밀 투표를 가능하게 했다.

## 기존 프로젝트 폐기 이유

* 현실적인 문제 : ‘법’ 이라는 주제의 진입장벽이 높기 때문에 사람들의 관심을 끌기 어렵고 타겟 사용자의 수가 극히 적다. 또한, 현재 국회나 정부에서 국민들의 의견을 필수적으로 반영해야 한다는 조건이 없어 ‘입법예고제도’의 실효성에 의문이 있다.
* 기술적인 문제 : 블록체인 기술을 이용하여 투표 조작을 막을 수는 있지만, 부 계정을 이용한 여론 조작을 막을 수는 없다.

# 프로젝트 개요

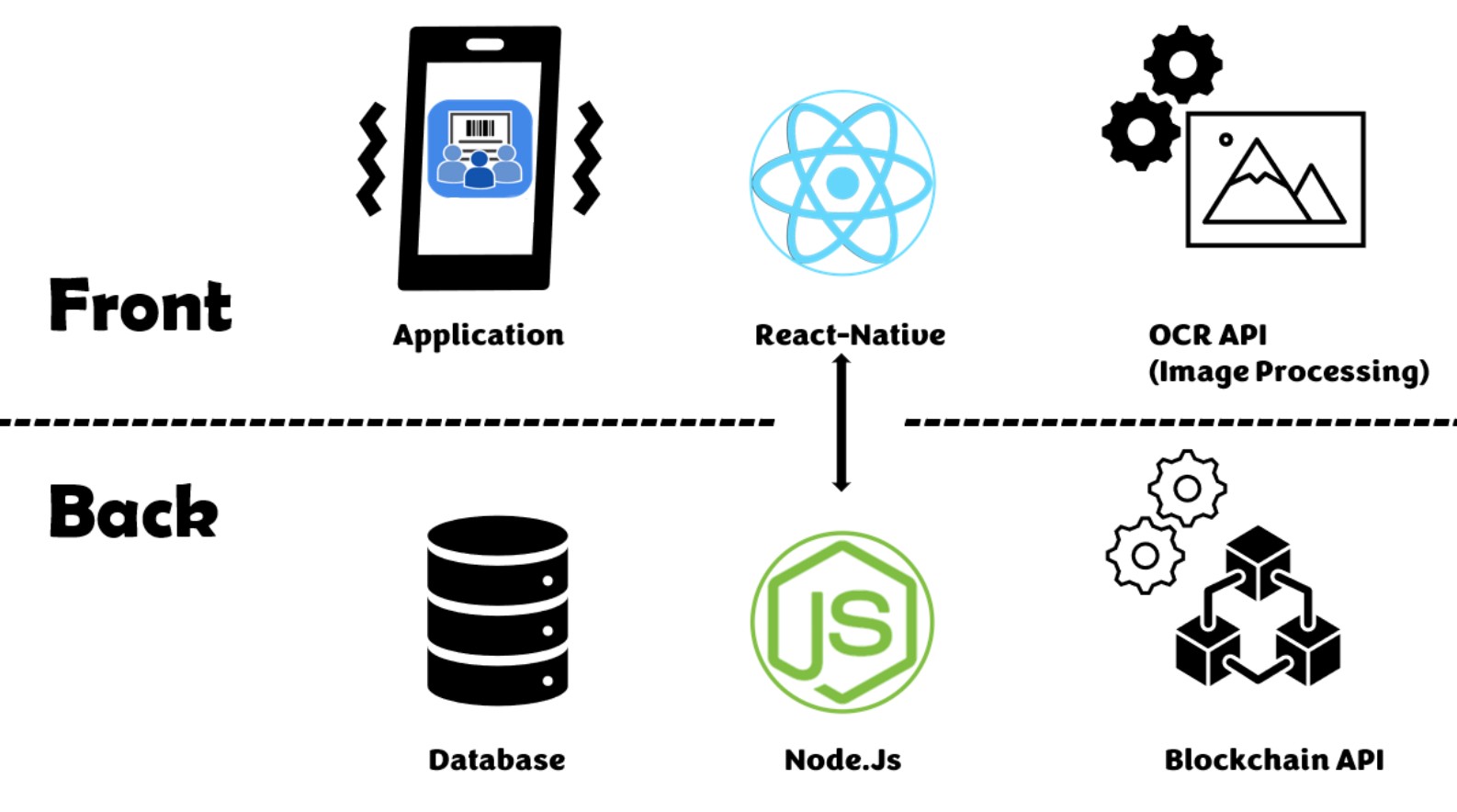
새로운 프로젝트명은 ‘모두모영’ 이다. ‘모두모영’은 모임의 회비 내역을 관리하는 모바일 어플리케이션이다. 휴대폰의 이미지와 카메라를 이용하여 회비 추가, 회비 결제내역을 업데이트하고 모임의 모든 참가자들이 이를 조회할 수 있다. 또한, 모임의 회비 내역을 저장하는 방식이 일반 DB와 블록체인 두 가지로 구분되어 있다.

## 프로젝트 추진 배경

매년 모든 대학교의 학생회에서 학생 회비와 관련된 잡음이 지속되고 있다. 학생회의 목적에 맞지 않게 회비를 사용하거나, 회비 사용 내역에 조작이 의심되는 경우가 다수 발견된다. 해당 문제점은 총무가 회비 사용 내역을 분실하거나, 내역 집계 중에 실수를 저지르거나, 현실적으로 회비 사용 내역을 모두에게 공개하기 어렵기 때문에 발생한다.

본 프로젝트는 총무가 모임의 회비를 편하게 관리하고, 모임 참여자가 회비 내역을 모두 조회하여 서로 간의 갈등을 해결하기 위해 제작되었다.

## 프로젝트 조직도



# 프로젝트 세부 설명

## 이미지 처리

이미지 처리에 사용되는 방식 설명

어떤 이미지 처리 API를 사용했고 그 이유?

이미지 처리 요청과 결과 반환이 어디와 이어지는지

## 블록체인

블록체인은 P2P 기반의 데이터 저장 방식이다. 데이터는 블록에 기록되고 해당 블록의 해쉬값이 다음 블록에 저장된다. 누군가 임의적으로 블록의 데이터를 변경하면 이어지는 블록들의 해쉬값이 변경된다. 블록체인 네트워크의 참가자들은 모두 동일한 데이터를 가지고 있기 때문에, 이를 파악하고 승인하지 않는다. 결과적으로, 블록체인에 저장된 데이터를 조작할 수 없다.

본 프로젝트에서는 총무가 회비를 데이터를 고의적으로 조작하는 것을 방지하기 위해 블록체인을 도입했다.

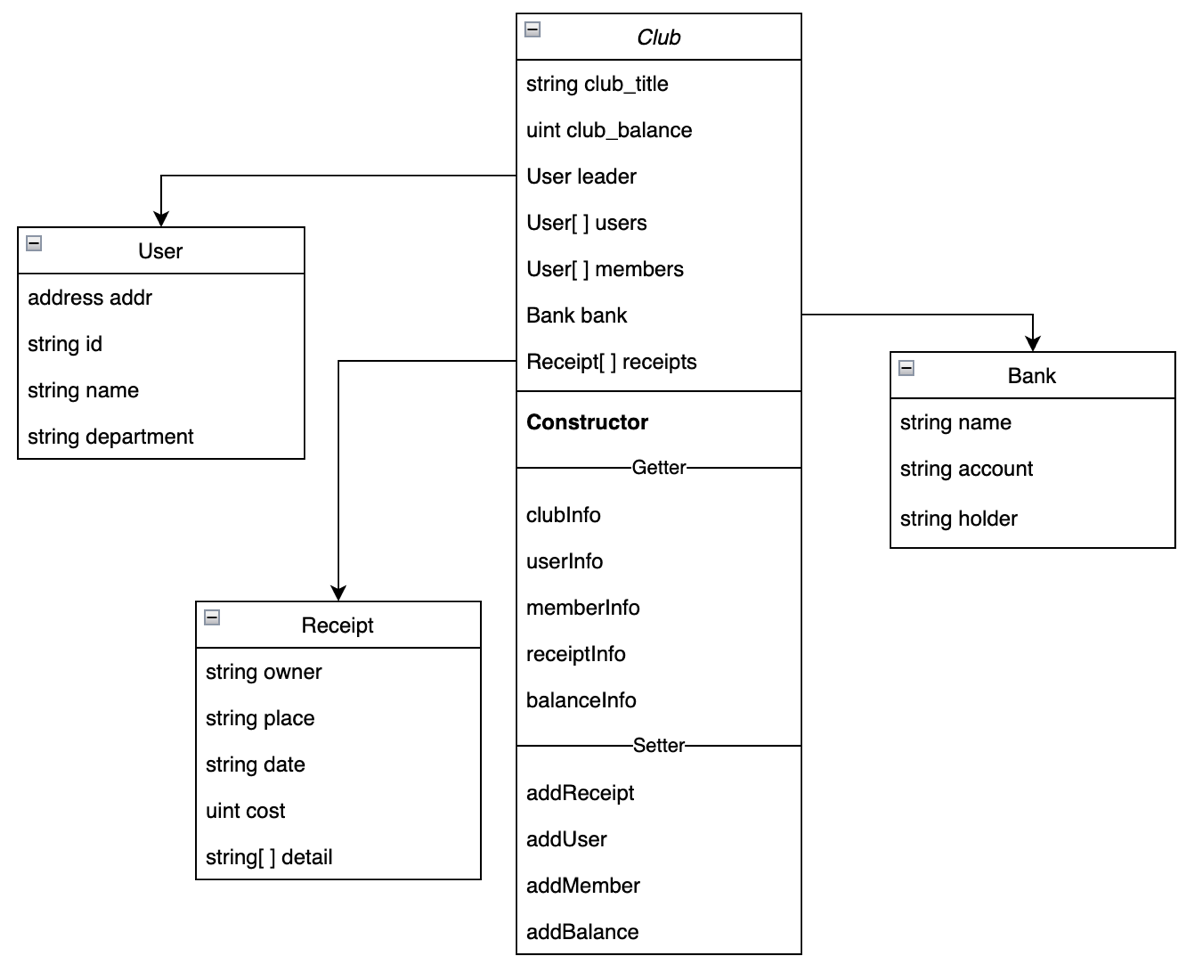
블록체인 네트워크는 이더리움 방식 중 하나인 Go-ethereum(이하 ‘Geth’) 클라이언트를 사용했다. Geth는 Hyperledger 보다 많은 정보가 있어 비교적 학습이 빠른 이유로 채택했다.

이더리움 기반의 블록체인은 스마트 컨트랙트를 미리 작성하고, 서버에서 함수를 실행 요청할 수 있다. 함수를 실행하는 주체(서버)는 채굴자에게 지급되는 비용인 gas를 지불한다. 이때, 사용자는 채굴을 하지 않기 때문에 gas가 없다. 때문에 서버에서 생성한 계정을 사용하여 대리로 함수를 실행한다.

현재 블록체인은 사설 네트워크로 개설했다. 3개의 노드(작성자의 개인 머신 2개, AWS EC2 1개)로 구성되어 있다.

### 스마트 컨트랙트

사용자가 모임을 블록체인으로 생성한 경우 Club 스마트 컨트랙트가 발급된다. 해당 컨트랙트에서 데이터 변경이 일어나면 모두 블록체인에 기록된다. 일반 DB에 저장되는 데이터 구조와 가능한 동일하게 구성했다.



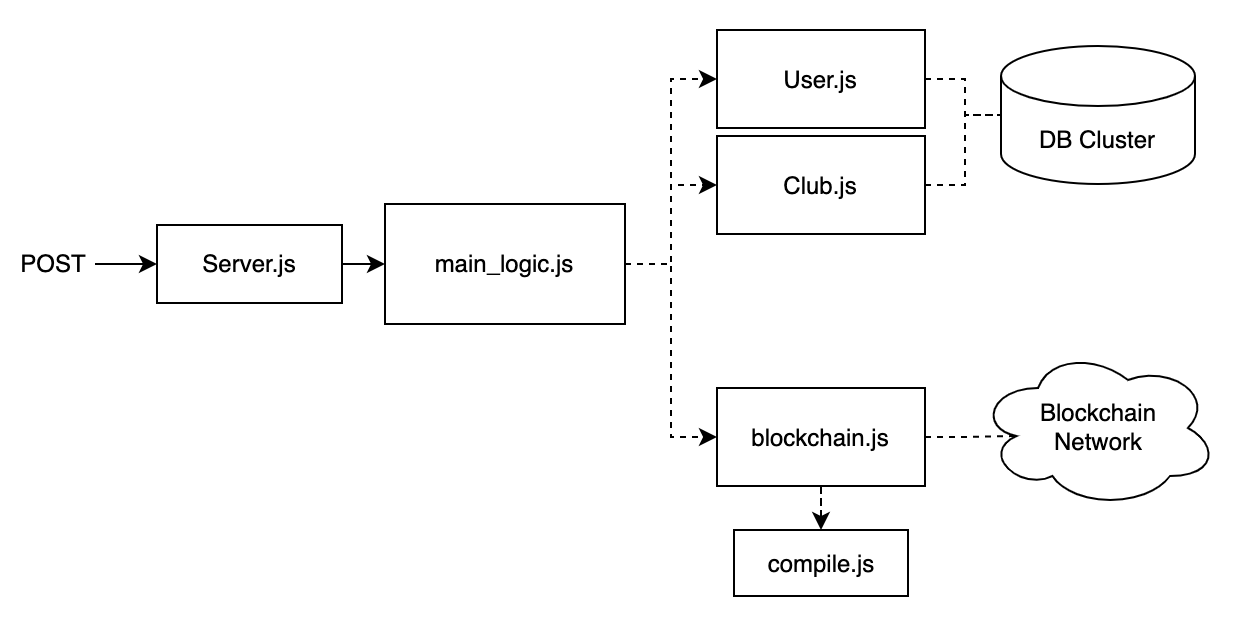
## 웹 어플리케이션 서버(Node.js)

Node.Js는 Web3 모듈을 통해 Geth와 통신하기 위한 가장 일반적인 플랫폼이다. 본 프로젝트는 적은 규모의 서비스를 다수의 사용자에게 모바일로 제공하기 때문에 Node.js 가 적합하다고 판단했다.

### 사용 모듈

* Solc : Solidity(스마트 컨트랙트 언어) 컴파일
* Web3 : Geth와 통신
* Express : 앱 서버와 POST 통신
* Mongoose : Mongo DB Cluster 통신
* Jsonwebtoken : (상윤 작성)
* Cookie-parser : (상윤 작성)
* Body-parser : (상윤 작성)
* Bcrypt : (상윤 작성)

### 컴포넌트



* server.js : 앱에서 요청한 기능을 비즈니스 로직(database.js)으로 연결하는 역할을 수행한다.
* Main\_logic.js : 모든 비즈니스 로직을 수행한다. 자세한 내용은 3.3.3 에서 설명한다.
* blockchain.js : Web3 모듈을 통해 블록체인 네트워크와 RPC(Remote Procedure Call) 통신을 수행한다.
* compile.js : Club.sol을 컴파일하고 bytecode 와 ABI code(Application Binary Interface)를 반환하는 역할을 수행한다.
* Club.js : 데이터베이스에 저장할 club의 데이터 구조가 정의되어 있다.
* User.js : 데이터베이스에 저장할 user의 데이터 구조가 정의되어 있다.

### 비즈니스 로직

* 회원가입 (register)
* 로그인 (login)
* 모임 생성 (createClub)
* 모임 참가 (joinClub)
* 사용자 모임 조회 (userClubInfo)
* 모임 페이지 이동 (gotoClub)
* 회비 추가 (addClubFee)
* 영수증 추가 (addClubReceipt)
* 총무 추가 (addClubMember)
* 모임 영수증 조회 (clubReceipt)

## 앱 어플리케이션(React-Native)

## 데이터베이스

어떤 데이터베이스를 선택했고 그 이유?

데이터베이스의 데이터 저장 구조

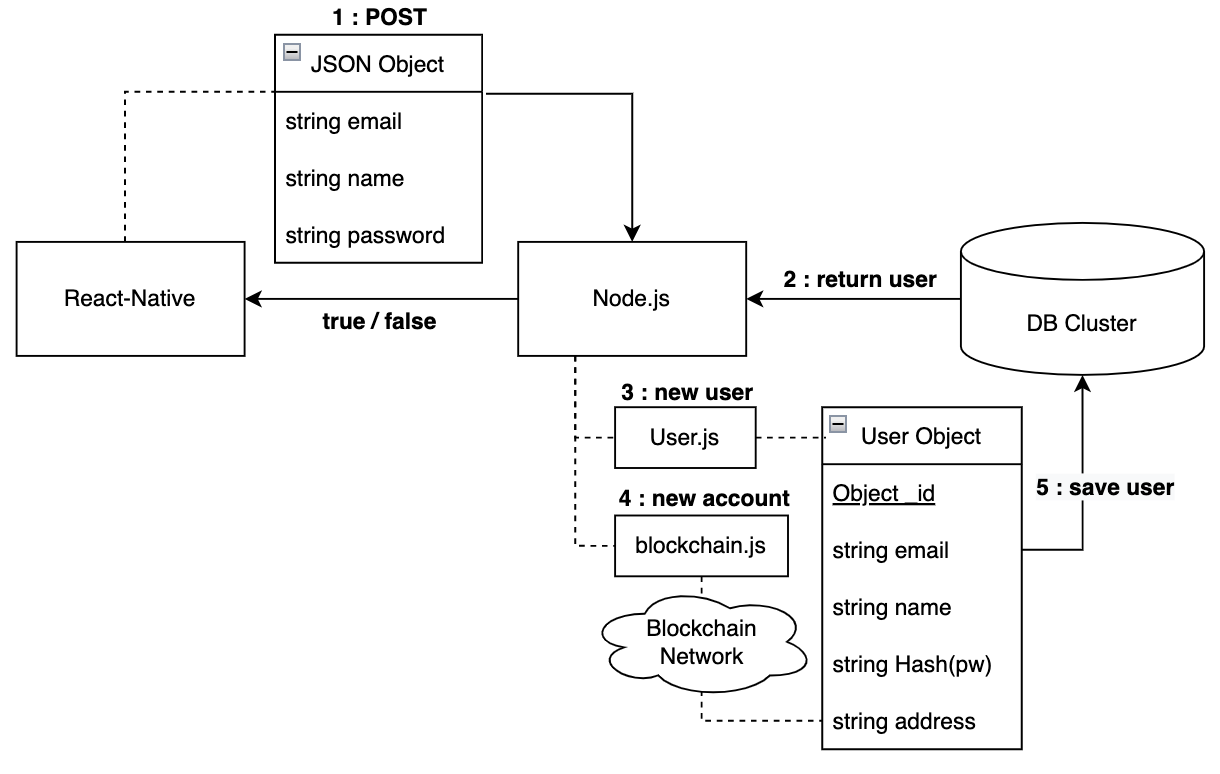
* 일반 DB 선택 시
* 블록체인 선택 시

## AWS EC2

AWS EC2는 서버를 개설하는 가장 일반적인 방법이다. EC2 인스턴스에서 Node.js와 Geth를 구동한다. Geth에서 블록을 채굴할 수 있는 최소 사양 조건과 지원받을 수 있는 금액 조건에 맞추어 t2.medium을 선택했다. React-Native의 연결 주소를 고정하기 위해 Elastic IP를 배정받았고 Port를 3001로 고정했다.

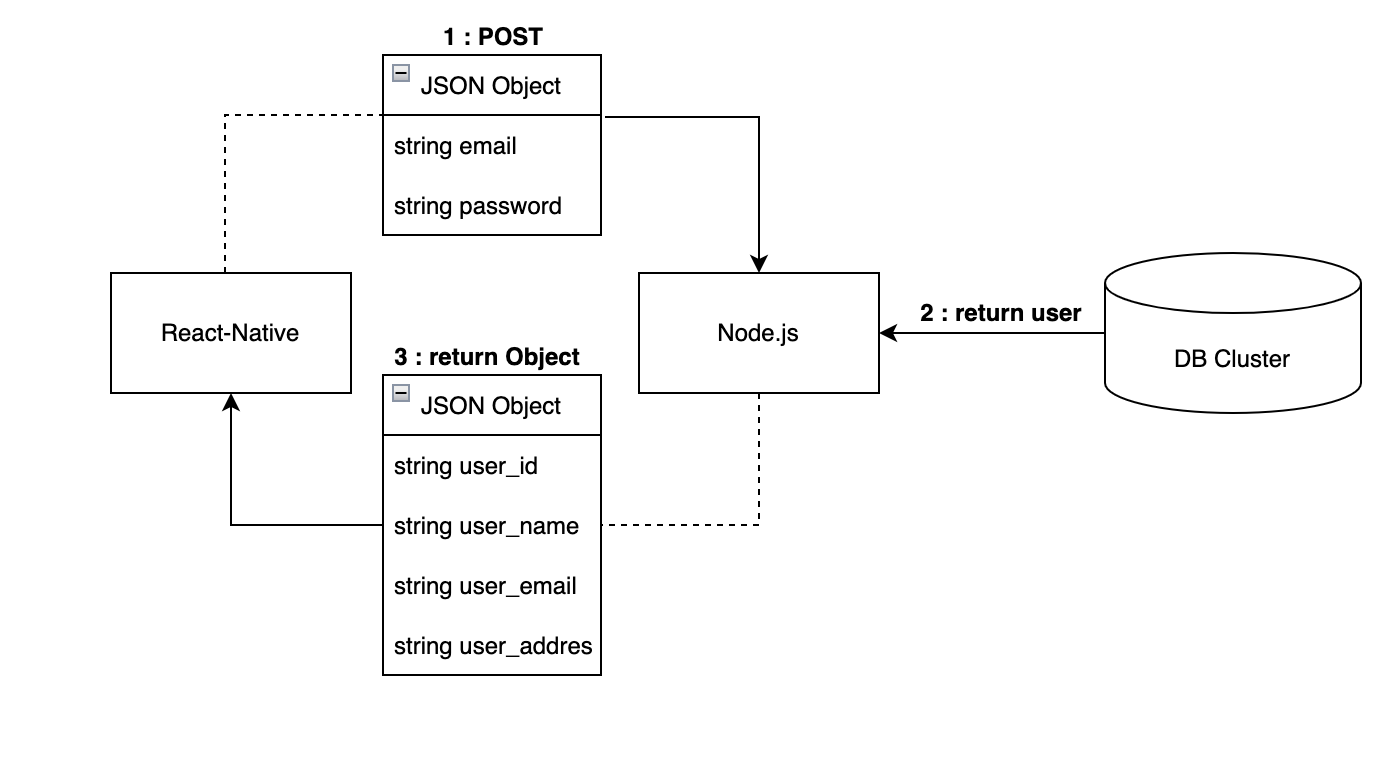
# 기능별 흐름 설명

## 회원가입



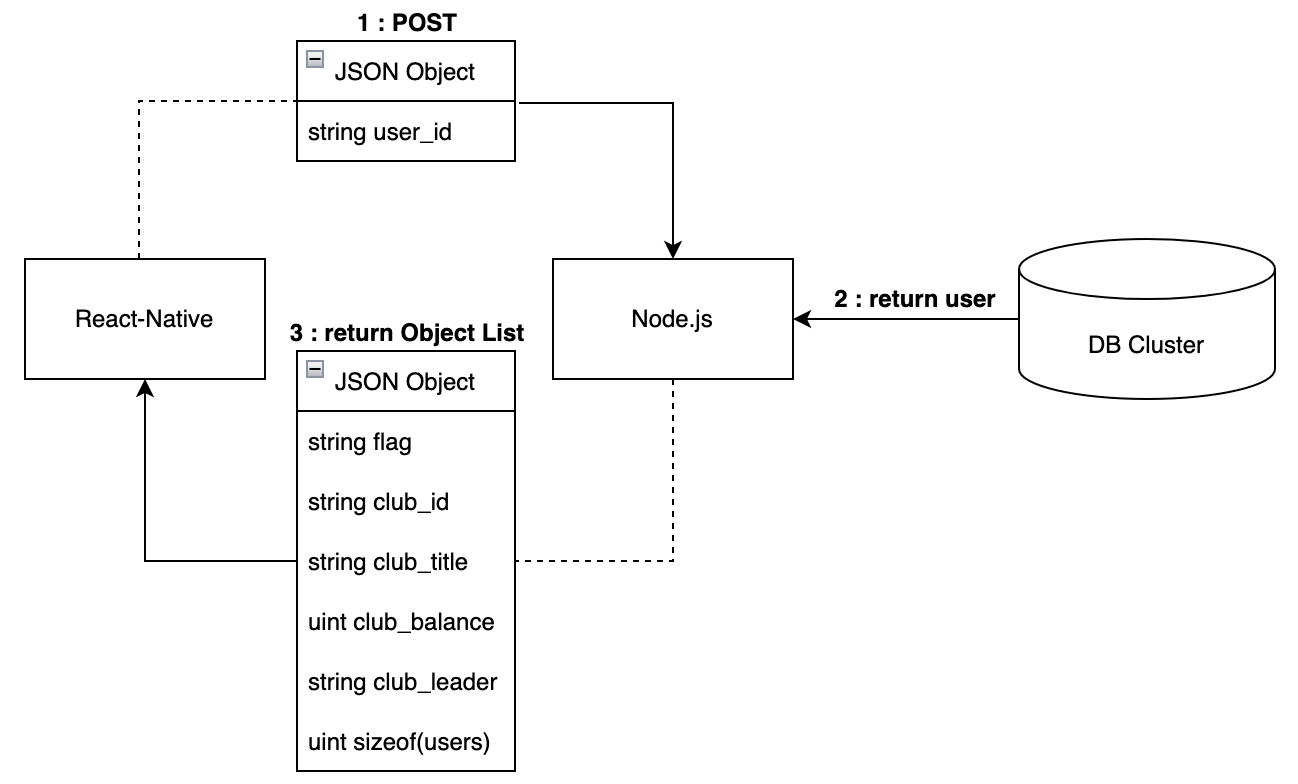
1. React-Native 에서 POST로 사용자가 입력한 데이터를 전달한다.
2. Node.js는 DB에서 email의 중복을 검사한다. 중복 시 false를 응답한다.
3. Node.js는 User.js에서 정의된 구조에 따라 새로운 User를 생성한다.
4. Node.js는 블록체인에서 새로운 account를 생성하여 User에 추가한다.
5. User를 DB에 저장하고 true를 응답한다.

## 로그인



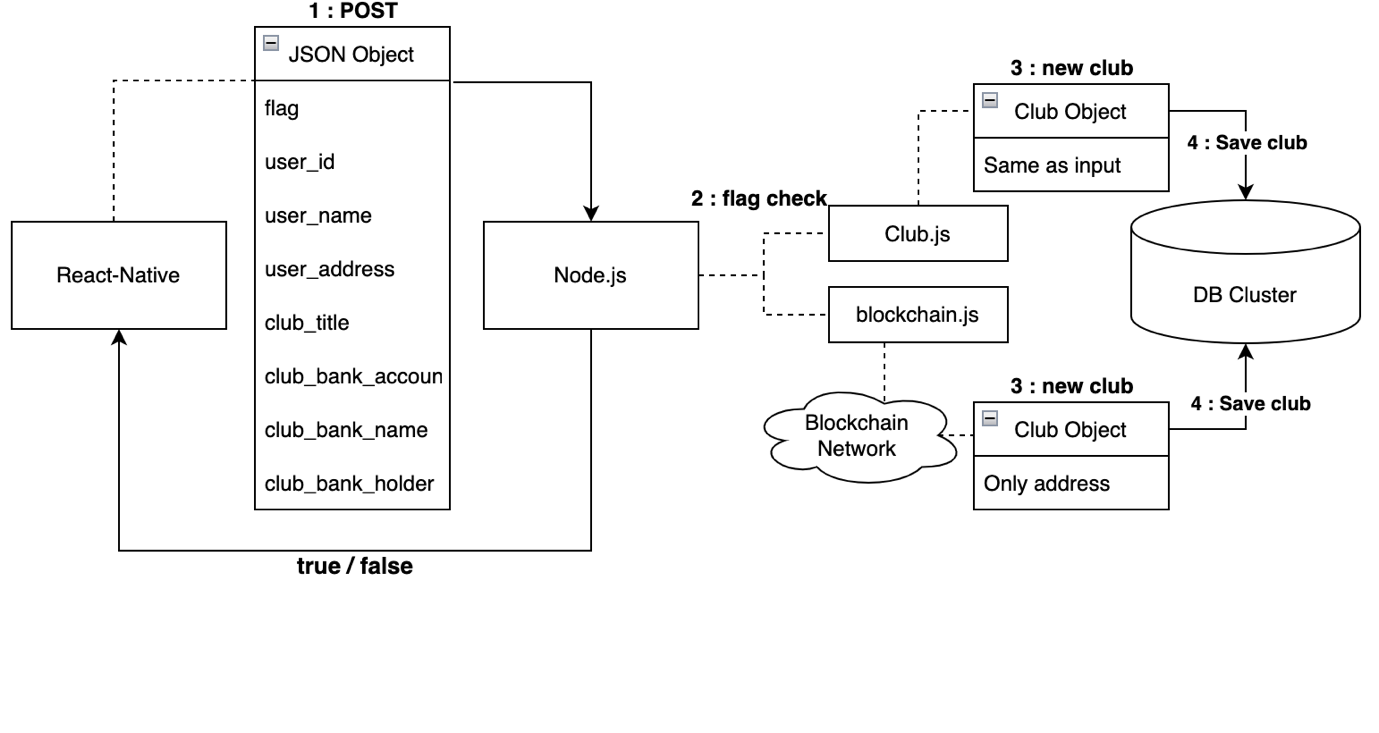
1. React-Native 에서 POST로 사용자가 입력한 데이터를 전달한다.
2. Node.js는 DB에서 user을 가져오고, password 일치 여부를 판단한다. 불일치 시 false를 응답한다.
3. User의 데이터 일부분을 가공하여 응답한다.

## 참가한 모임 조회



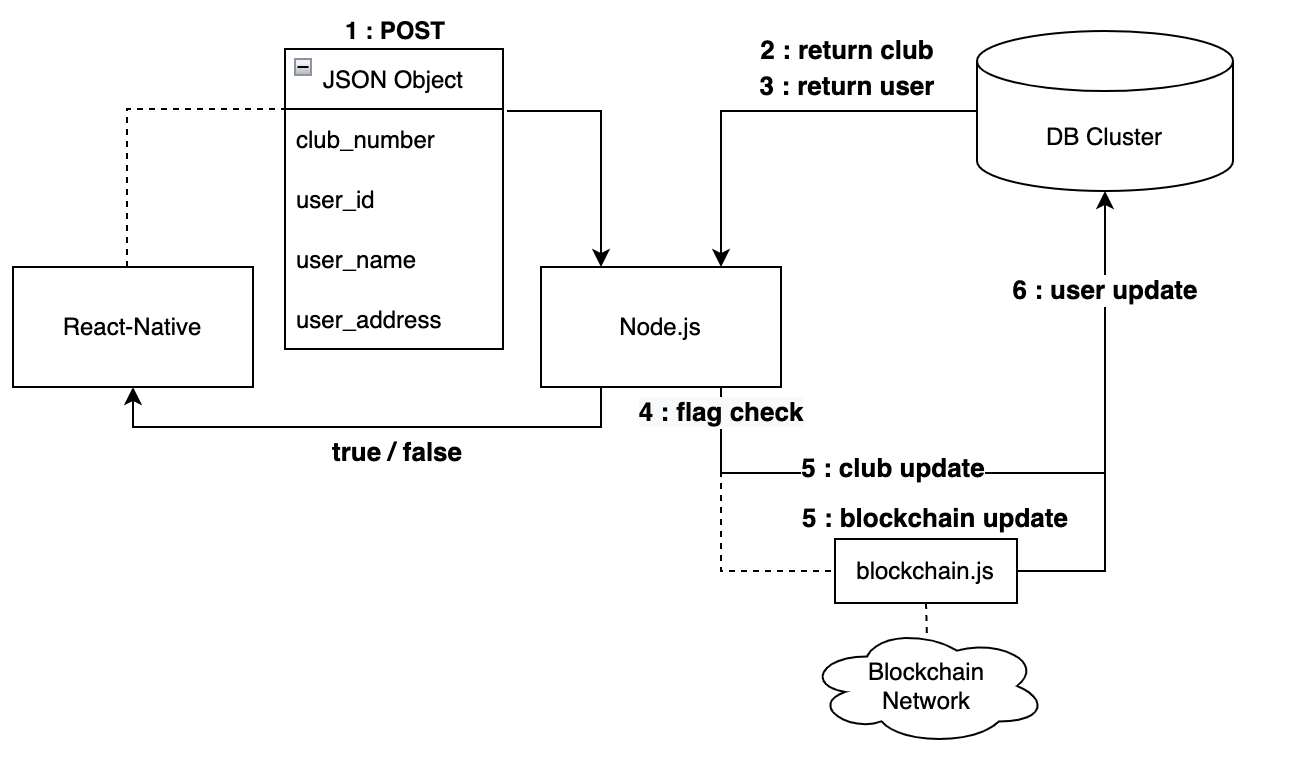
1. React-Native 에서 POST로 사용자가 입력한 데이터를 전달한다.
2. Node.js는 DB에서 user을 가져온다.
3. User의 데이터 일부분을 가공하여 응답한다.

## 모임 생성



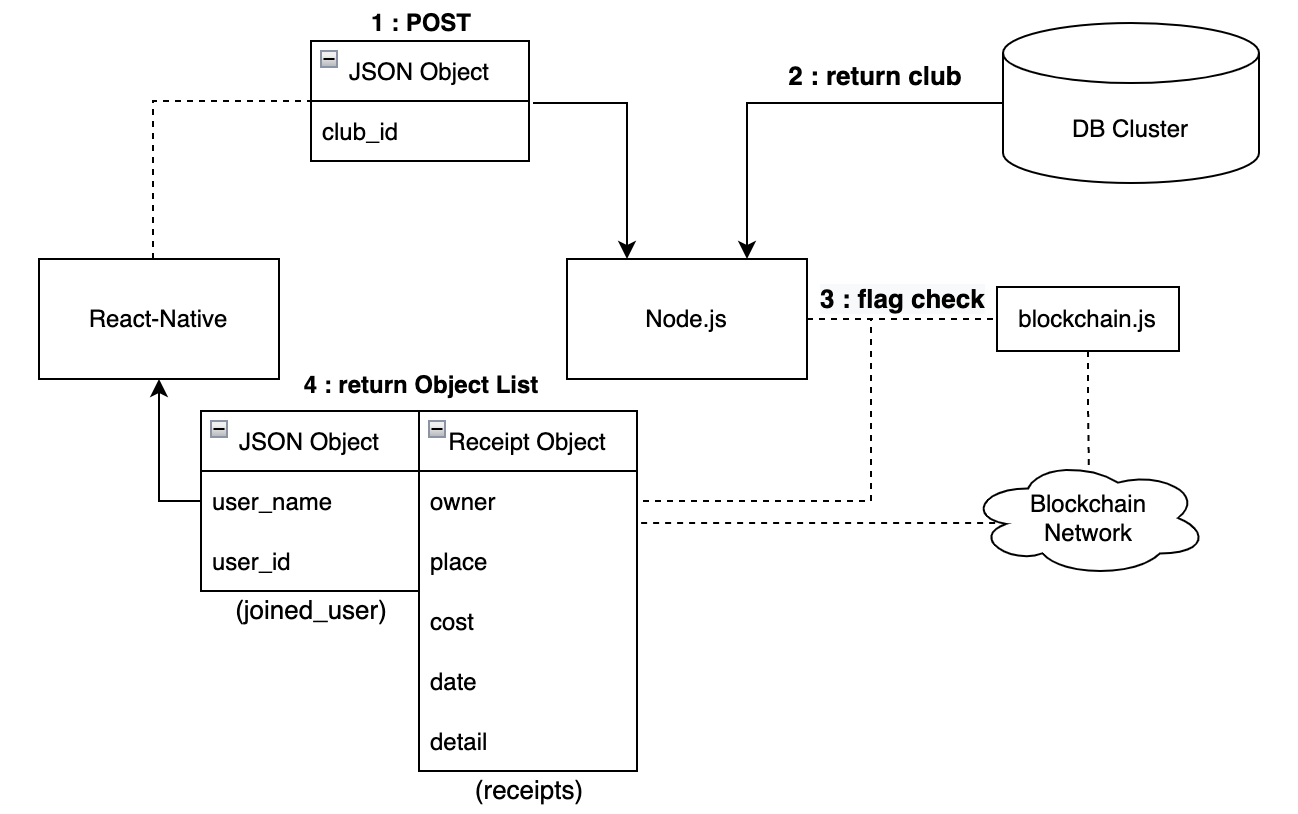
1. React-Native 에서 POST로 사용자가 입력한 데이터를 전달한다.
2. Node.js는 flag 여부를 판단한다.
3. 일반 DB / 블록체인 분기
   1. 일반 DB : Club.js에 정의된 데이터 구조에 맞게 새로운 Club을 생성한다.
   2. 블록체인 : 스마트 컨트랙트를 발급하고 해당 address를 담은 새로운 Club을 생성한다.
4. Club을 DB에 저장하고 true를 응답한다.

## 모임 참가



1. React-Native 에서 POST로 사용자가 입력한 데이터를 전달한다.
2. Node.js는 DB에서 club을 가져온다.
3. Node.js는 DB에서 user를 가져온다.
4. Node.js는 Club에 정의된 flag 여부를 판단한다.
5. 일반 DB / 블록체인 분기
   1. 일반 DB : Club의 joined\_user에 user\_id를 추가하고 DB를 업데이트 한다.
   2. 블록체인 : 스마트 컨트랙트 변수 joined\_user에 user를 수정한다.
6. User의 joined\_club에 club\_id를 추가하고 DB를 업데이트 한다.

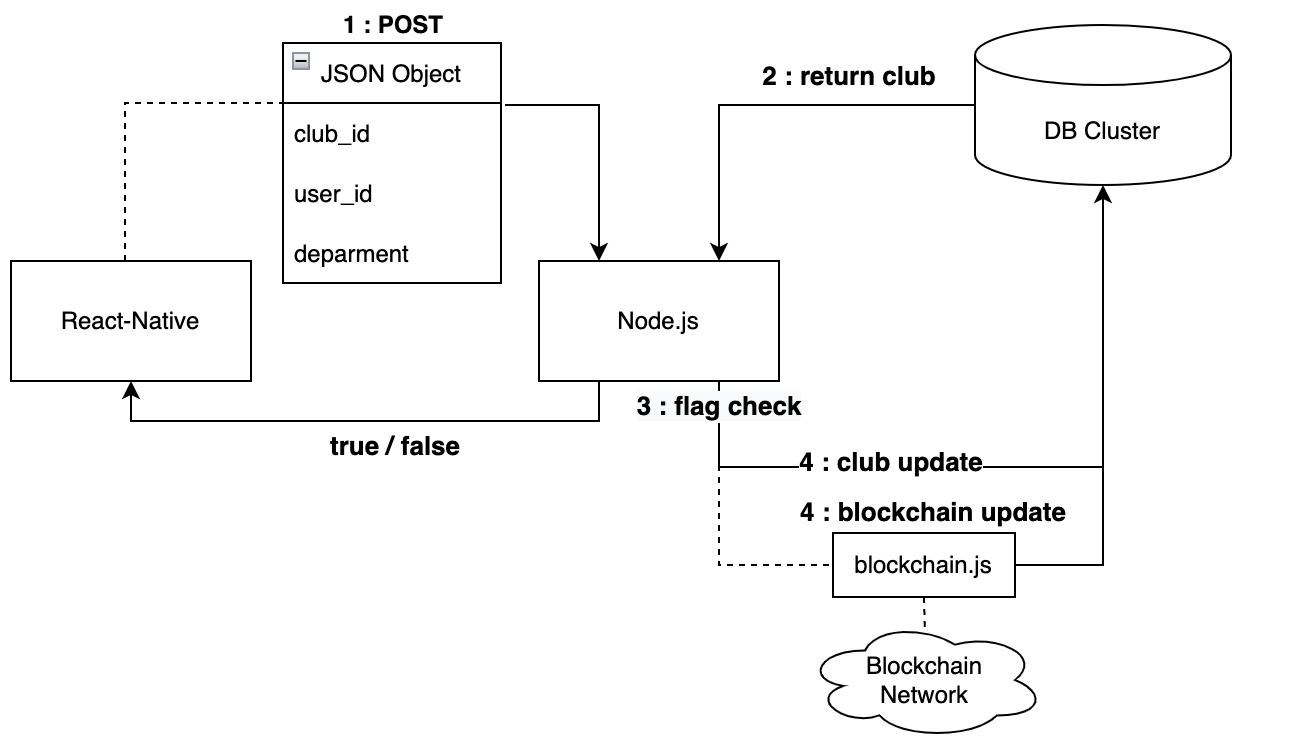
## 모임 이동



1. React-Native 에서 POST로 사용자가 입력한 데이터를 전달한다.
2. Node.js는 DB에서 club을 검색한다.
3. Node.js는 Club에 정의된 flag 여부를 판단한다.
4. 일반 DB / 블록체인 분기
   1. 일반 DB : club의 joined\_user와 receipts 응답한다.
   2. 블록체인 : 스마트 컨트랙트 변수 joined\_user와 receipts를 응답한다.

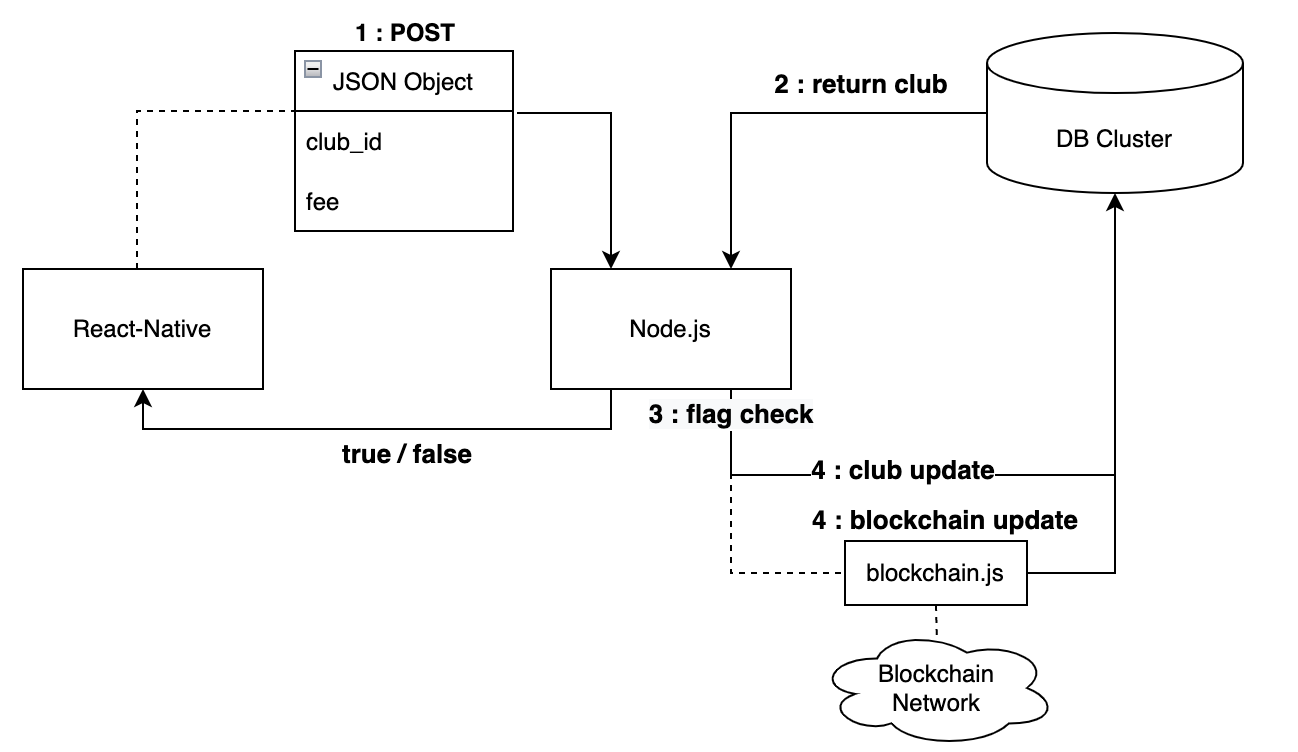
## 회비 사용 내역 조회

## 총무 추가



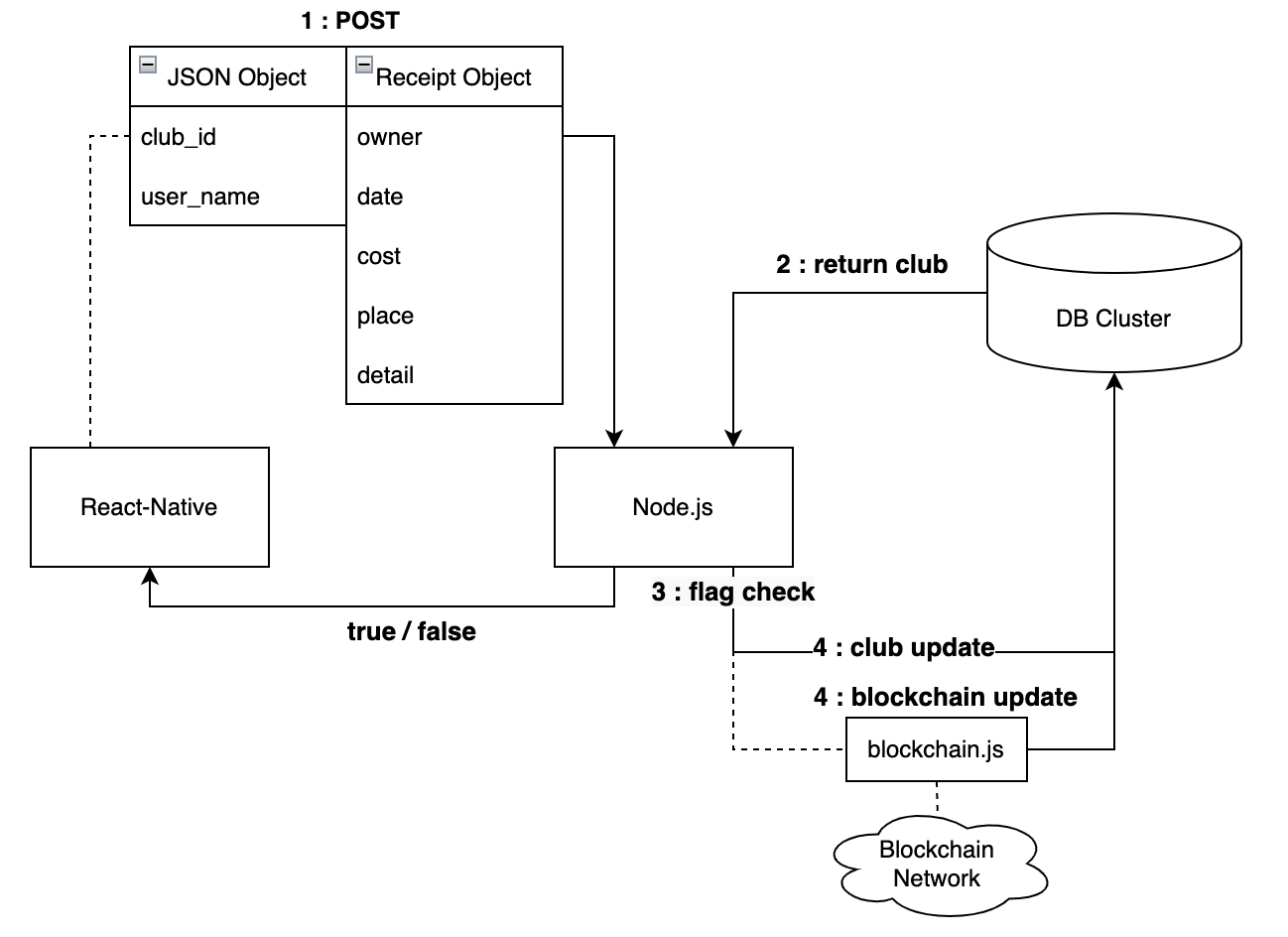
1. React-Native 에서 POST로 사용자가 입력한 데이터를 전달한다.
2. Node.js는 DB에서 club을 검색한다.
3. Node.js는 Club에 정의된 flag 여부를 판단한다.
4. 일반 DB / 블록체인 분기
   1. 일반 DB : Club의 joined\_member에 user\_id를 추가하고 DB를 업데이트 한다.
   2. 블록체인 : 스마트 컨트랙트 변수 joined\_member를 수정한다.

## 회비 추가



1. React-Native 에서 POST로 사용자가 입력한 데이터를 전달한다.
2. Node.js는 DB에서 club을 검색한다.
3. Node.js는 Club에 정의된 flag 여부를 판단한다.
4. 일반 DB / 블록체인 분기
   1. 일반 DB : Club의 club\_balance에 fee를 증가시키고 DB를 업데이트 한다.
   2. 블록체인 : 스마트 컨트랙트 변수 club\_balance를 수정한다.

## 영수증 추가



1. React-Native 에서 POST로 사용자가 입력한 데이터를 전달한다.
2. Node.js는 DB에서 club을 검색한다.
3. Node.js는 Club에 정의된 flag 여부를 판단한다.
4. 일반 DB / 블록체인 분기
   1. 일반 DB : Club의 receipts에 receipt를 추가하고 DB를 업데이트 한다.
   2. 블록체인 : 스마트 컨트랙트 변수 receipts를 수정한다.

# 자기평가

# 부록

## 사용자 메뉴얼

## 운영자 메뉴얼