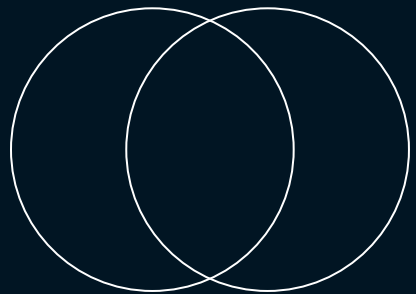


# View My Startup

Startup Information and Simulated Investment Service

1팀 초급 프로젝트



FS 2기 1팀 김영은 김재원 김태영 이강수

# Table of Contents

1. Overview	개요
2. Composition and Roles	구성 및 역할
3. Collaboration	협업
4. Procedures and Methods	수행 절차 및 방법
5. Results	수행 결과
6. Team Opinions	평가 의견
7. Q & A	질의응답

# 1. Overview

## 프로젝트 내용

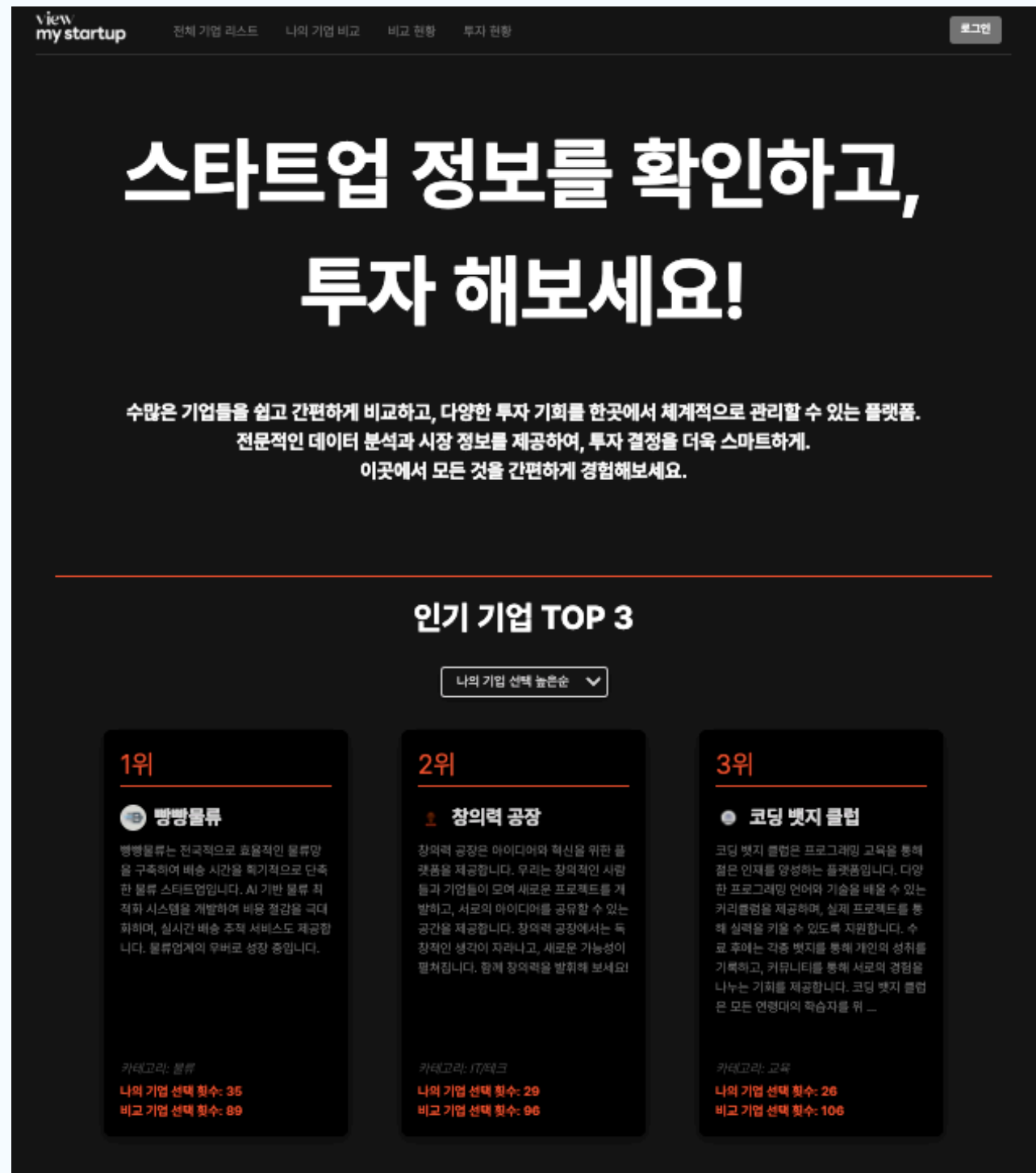
# View my startup

- **소개:** 스타트업 정보 확인 및 모의 투자 서비스를 제공하는 플랫폼.
- **배경:** 최근 개인 투자자들의 스타트업에 대한 관심이 증가.  
BUT, 스타트업에 대한 정보 접근성 부족.
- **목표:** 개인 투자자들이 스타트업을 선택하여 누적 투자 금액 및 매출 데이터를 확인하고 비교할 수 있는 모의 투자 환경을 제공하여, 접근성 개선.

#스타트업 선택

#투자 데이터 비교

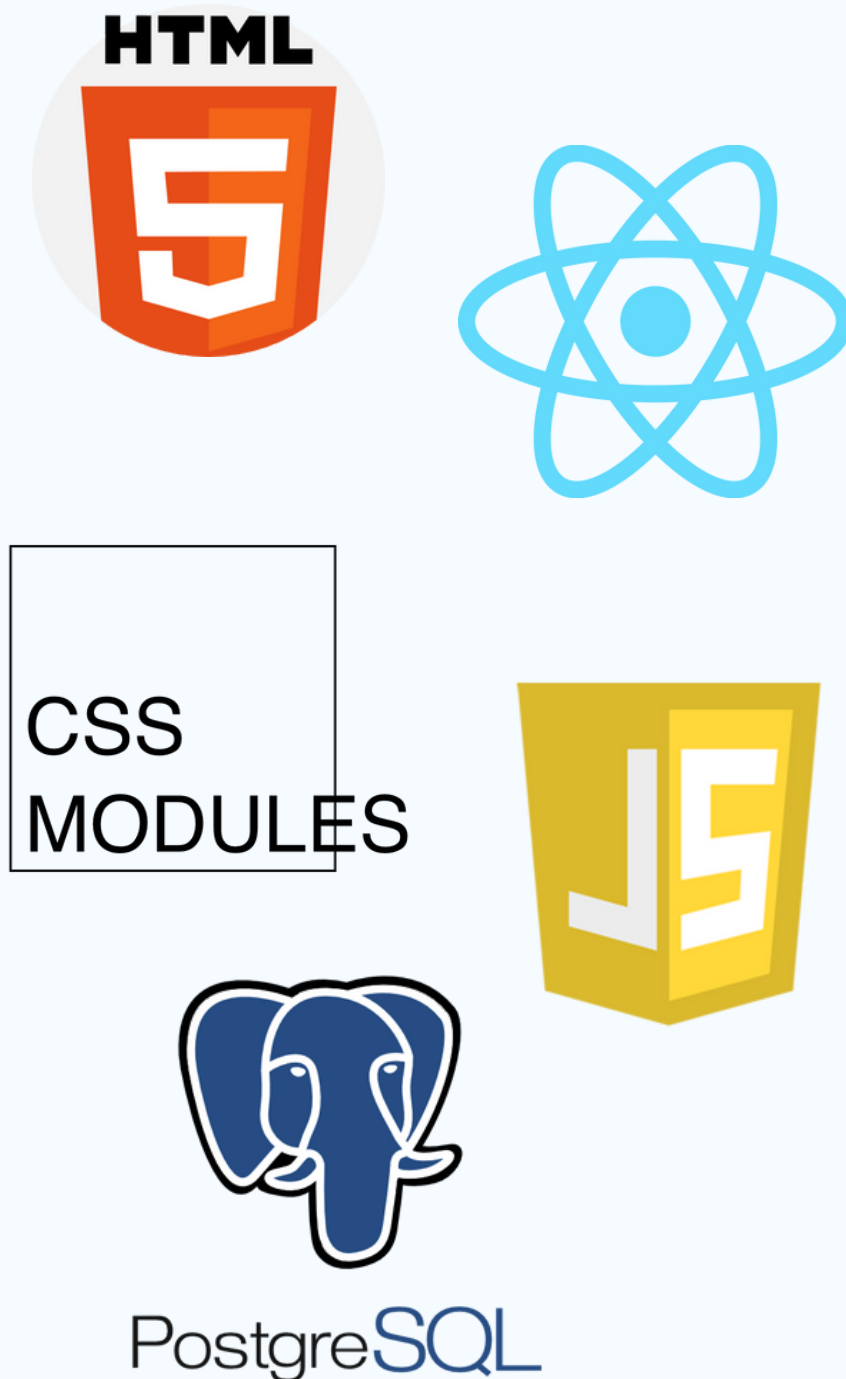
#모의 투자



# 1. Overview

## 개발 환경 및 사용 스택

- FE, BE, DB



- Productivity



- 배포



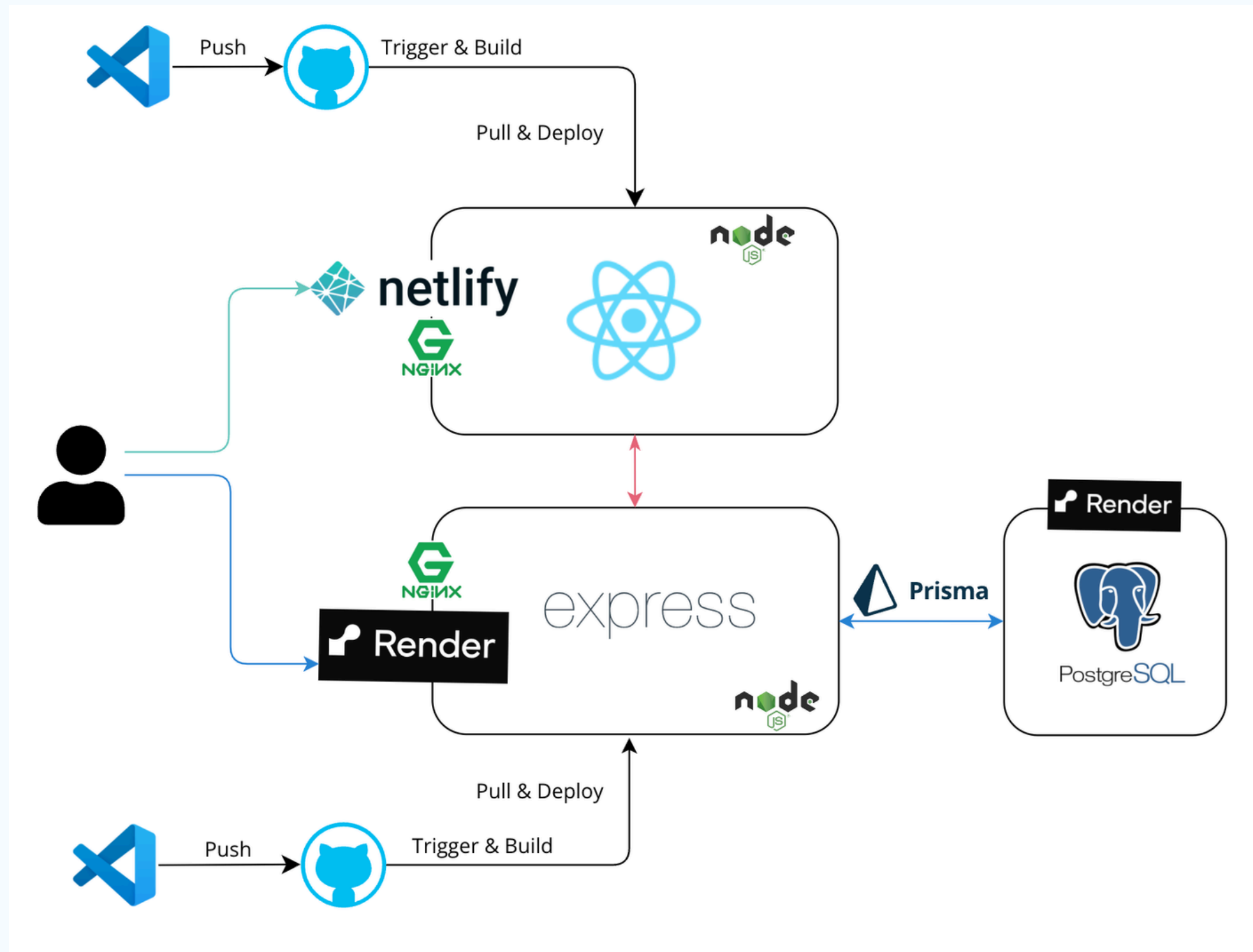
- 협업



# 1. Overview

- 아키텍처 다이어그램

## 프로젝트 구조

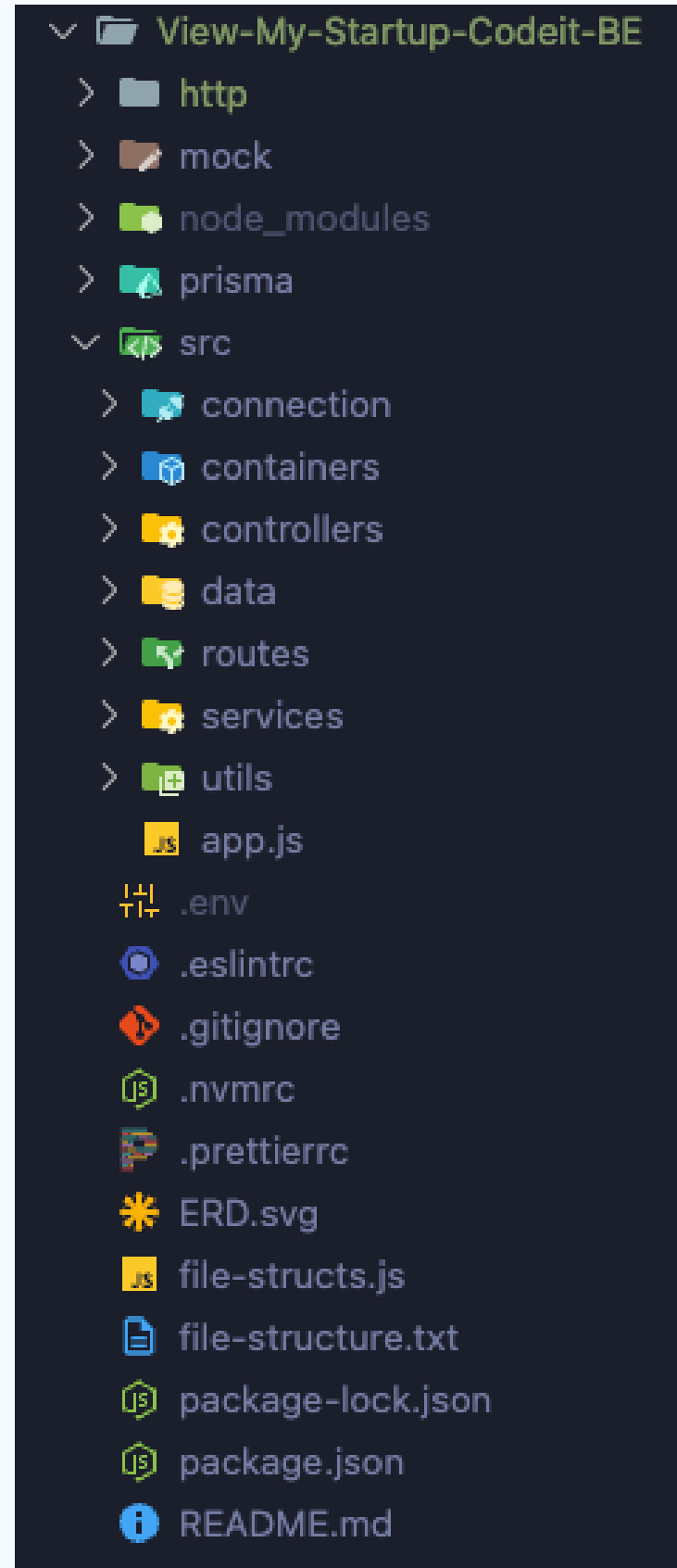
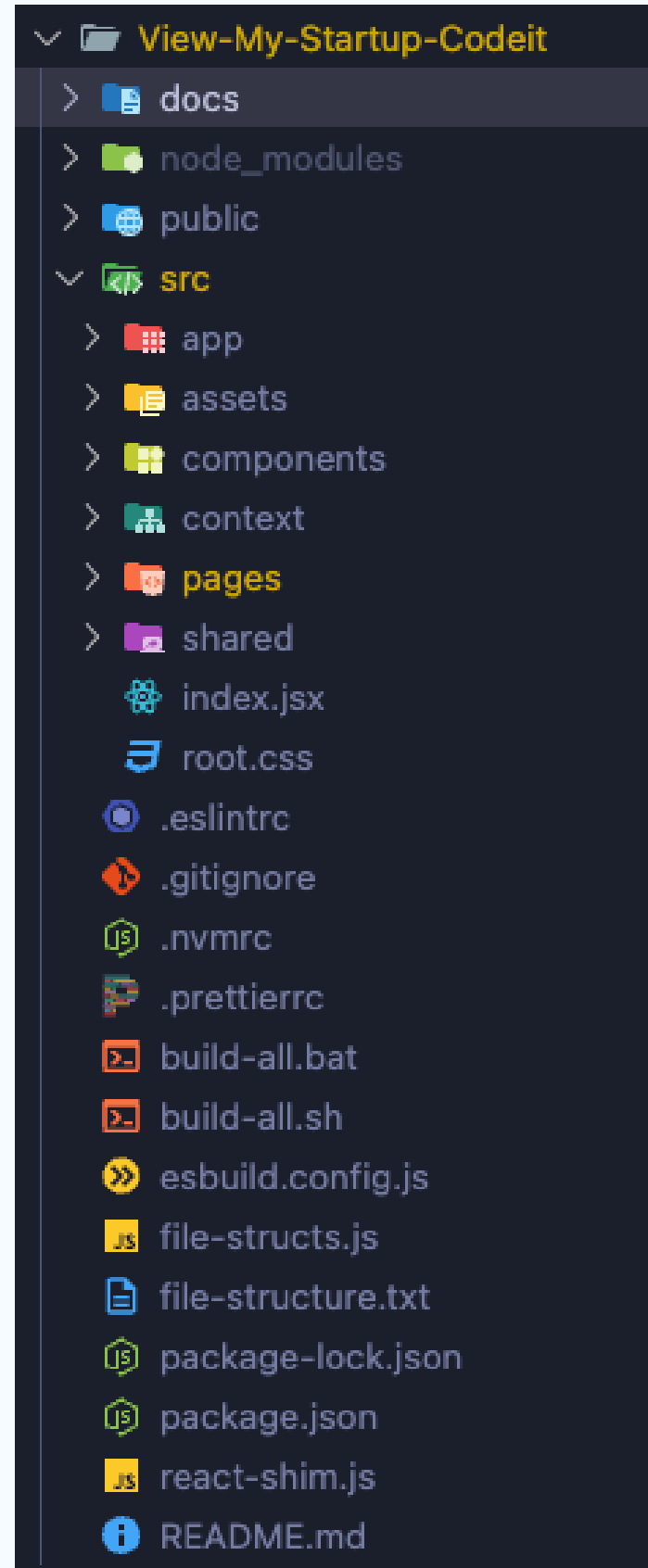


# 1. Overview

## 파일 구조

- FE 구조

### FSD 아키텍처



- BE 구조

### 3 Tier 아키텍처

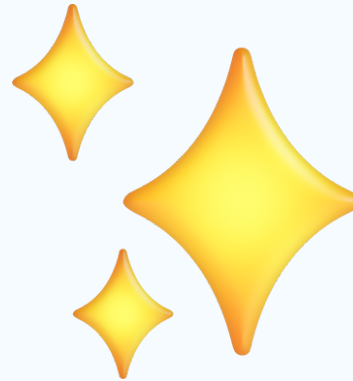
## 2. Composition and Roles

## 구성 및 역할



김영은

- 공통 컴포넌트 및 FE 라우트 세팅
- 나의 비교 페이지 FE
- 비교 완료 페이지 FE
- 기업 및 투자 관련 일부 BE



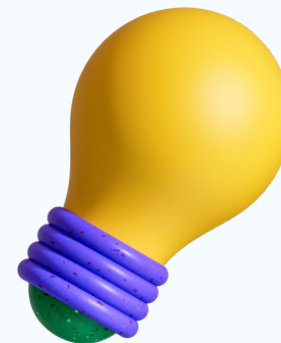
김태영

- 초기 컨벤션 세팅
- 아키텍처 세팅
- 기업 상세 페이지 FE/BE
- 해당 기업의 투자 목록 테이블 FE/BE
- 투자 정보 생성/수정/삭제 FE/BE



김재원

- 나의 비교 페이지 BE
- 비교 완료 페이지 BE
- 랜딩 페이지 FE/BE
- 발표자료 제작












이강수

- 전체 기업 리스트 FE
- 비교, 투자 현황 FE
- 로그인, 가입하기 FE/BE
- Social 로그인/가입하기 FE/BE
- 팀장

### 3. Collaboration

## 협업 방식

<input type="checkbox"/>  <b>Dev to main and Deploy through Netlify</b> #85 by kipid was merged yesterday		
<input type="checkbox"/>  <b>Feat: Add investment</b> #84 by Maybeiley was merged 2 days ago	<b> 들여쓰기</b> <ul style="list-style-type: none"> <li>1탭 = 2스페이스로 설정해 사용</li> </ul>	
<input type="checkbox"/>  <b>Fix: resolve css issue</b> #83 by Maybeiley was merged 2 days ago	<b>블록 구문</b> <ul style="list-style-type: none"> <li>한 줄짜리 블록일 경우라도 {}를 생략하지 않고, 명확히 줄 바꿈 하여 사용한다</li> </ul>	<pre>// bad if(true) return 'hello'  // good if(true){   return 'hello' }</pre>
<input type="checkbox"/>  <b>Fix: Add unauthoriz</b> #81 by csbizz was merged 3 days ago		
<input type="checkbox"/>  <b>Dev to main and Dep</b> #80 by kipid was merged 3 days ago		
<input type="checkbox"/>  <b>feat: LandingPage co</b> #79 by galaxy-78 was merged 4 days ago	<b>함수</b> <ul style="list-style-type: none"> <li>함수는 함수 표현식을 사용하며, 화살표 함수를 사용한다.</li> </ul>	
<input type="checkbox"/>  <b>Design: Refine design</b> #78 by Maybeiley was merged 4 days ago		<pre>// Bad function fnName() {}; [1,2,3].map(function(x){   retrun {}; })  // Good const fnName = () =&gt; {}; [1,2,3].map(x =&gt; x);</pre>
<input type="checkbox"/>  <b>Feat: Add complete r</b> #77 by Maybeiley was merged 5 days ago		
<input type="checkbox"/>  <b>Fix/post modal</b> #76 by csbizz was merged 5 days ago	<ul style="list-style-type: none"> <li>바로 return 하는 경우</li> </ul>	<pre>// bad const foo = () =&gt; { return "bar"; }  // good const foo = () =&gt; "bar";</pre>

## 1 PR 리뷰 및 개발 컨벤션

- Naming
- Commit
- Code

## 2 일정 관리

- AM 09:30 ~ 09:45 - 주강사님과의 데일리 스크럼 활용하여 각자 진행 상황 공유
- 주기적인 ZOOM 회의를 통한 체크



# 3. Collaboration

# 협업 방식

- BigInt 이슈

```
model Company {
  // TODO 가상 투자 금액 총계를 직접 저장할 것인가, 매번 계산할 것인가?
  // 당장은 FE 단에서 계산하고 있음.
  id      String @id @default(uuid())
  name    String @unique
  description String
  category String // 회사 카테고리. 입력단에서 검증해야함.
  accumulInvest BigInt // NOTE 누적 투자 금액(실제)
  revenue  BigInt // NOTE 매출액
  employees Int // NOTE 고용 인원수
  logo     String? // NOTE 로고 이미지

  createdAt DateTime @default(now())
  updatedAt DateTime @updatedAt

  investments Investment[]
  comparisons Comparison[]
  watcherList Watch[]
}

const getScaledNumber = number => {
  if (!number) return null;

  const scaler = 10000;
  let scale = 0;
  let rest = Number(number);
  const dic = ['', '만', '억', '조', '경', '해', '자', '양', '구', '간'];


  while (rest >= scaler && scale < dic.length) {
    rest /= scaler;
    scale += 1;
  }
  const scaled = `${parseFloat(rest?.toFixed(2))}${dic[scale]}`;

  return scaled;
};

export default getScaledNumber;
```


## 3 이슈 트래킹

- 이슈 발생 -> 디스코드 상황 공유 -> 분석 -> 해결책
- 멘토를 통한 보완 -> PR 업로드


 FS\_2기\_이강수 10/11/2024 5:31 PM

```
{
  "id": "2d6ecf38-35bd-4098-98bd-46d2ed799167",
  "name": "kipid",
  "amount": "1",
  "comment": "!!!!!!!",
  "password": "123123",
  "createdAt": "2024-10-11T08:14:03.564Z",
  "updatedAt": "2024-10-11T08:14:03.564Z",
  "userId": "5e48d63a-572d-44fb-9aee-9e679e7983f8",
  "companyId": "2c1af571-b2eb-479a-b8c0-87ad911421f4"
}
```


로 응답은 오긴 하는데... 왜 에러날까요? =ㅂ=

 FS\_2기\_김태영 10/11/2024 5:32 PM


응답에 문제가 없다면 페이지네이션 에러의 일종일까 싶네요. map으로 테이블에 뿌리는 과정에서 에러가 났을것 같아요

 FS\_2기\_이강수 10/11/2024 5:40 PM

1000원을 입력해도 에러나네요.

 FS\_2기\_이강수 10/11/2024 5:43 PM

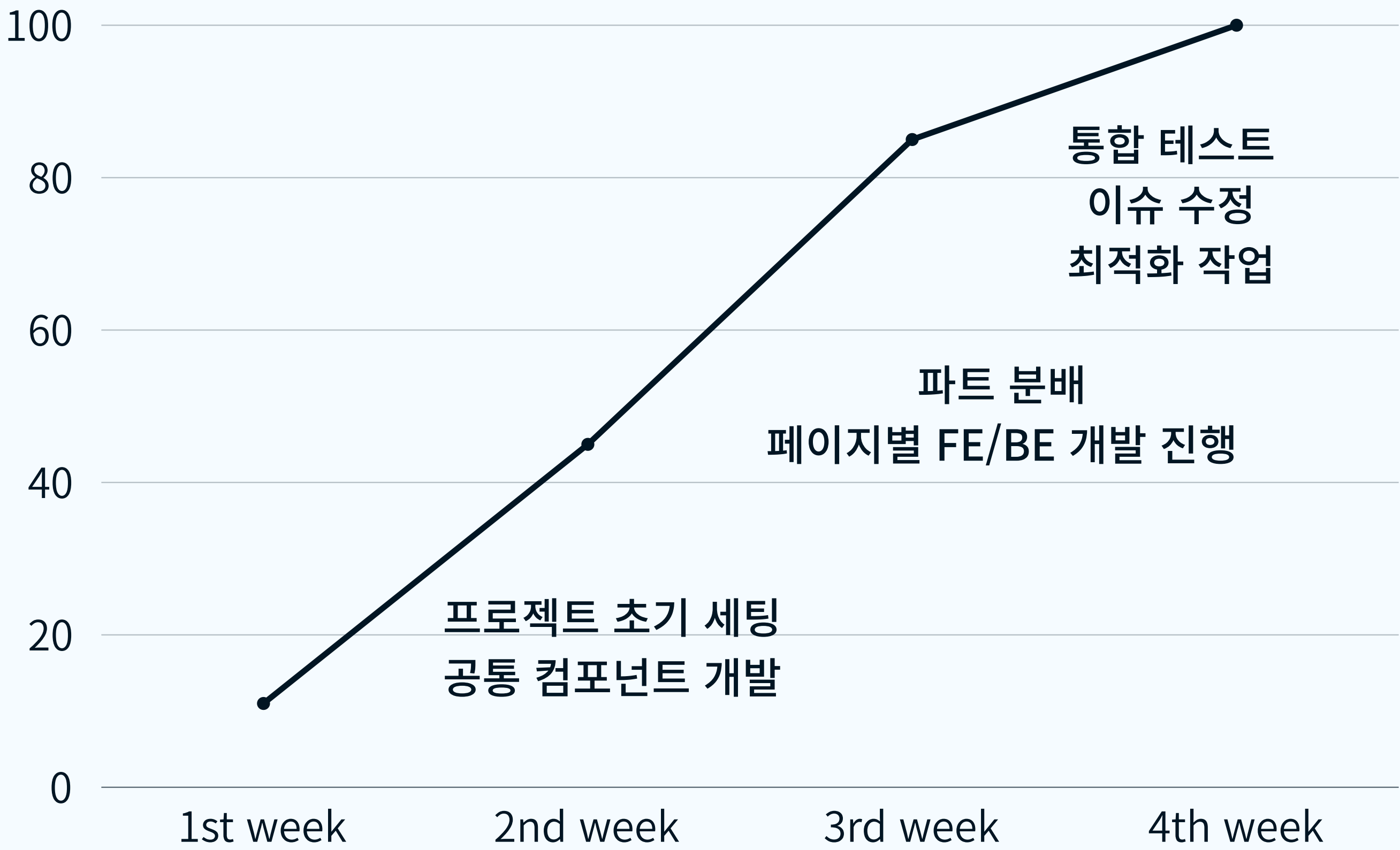
getScaledNumber 에서 rest = number 를 rest = Number(number) 로 바꿔볼게요.

 FS\_2기\_김태영 10/11/2024 5:43 PM

네 그게 맞네요  
저도 확인해보니  
rest가 string으로 잡히더라고요  
방금 똑같이 수정하니까 괜찮아졌습니4다

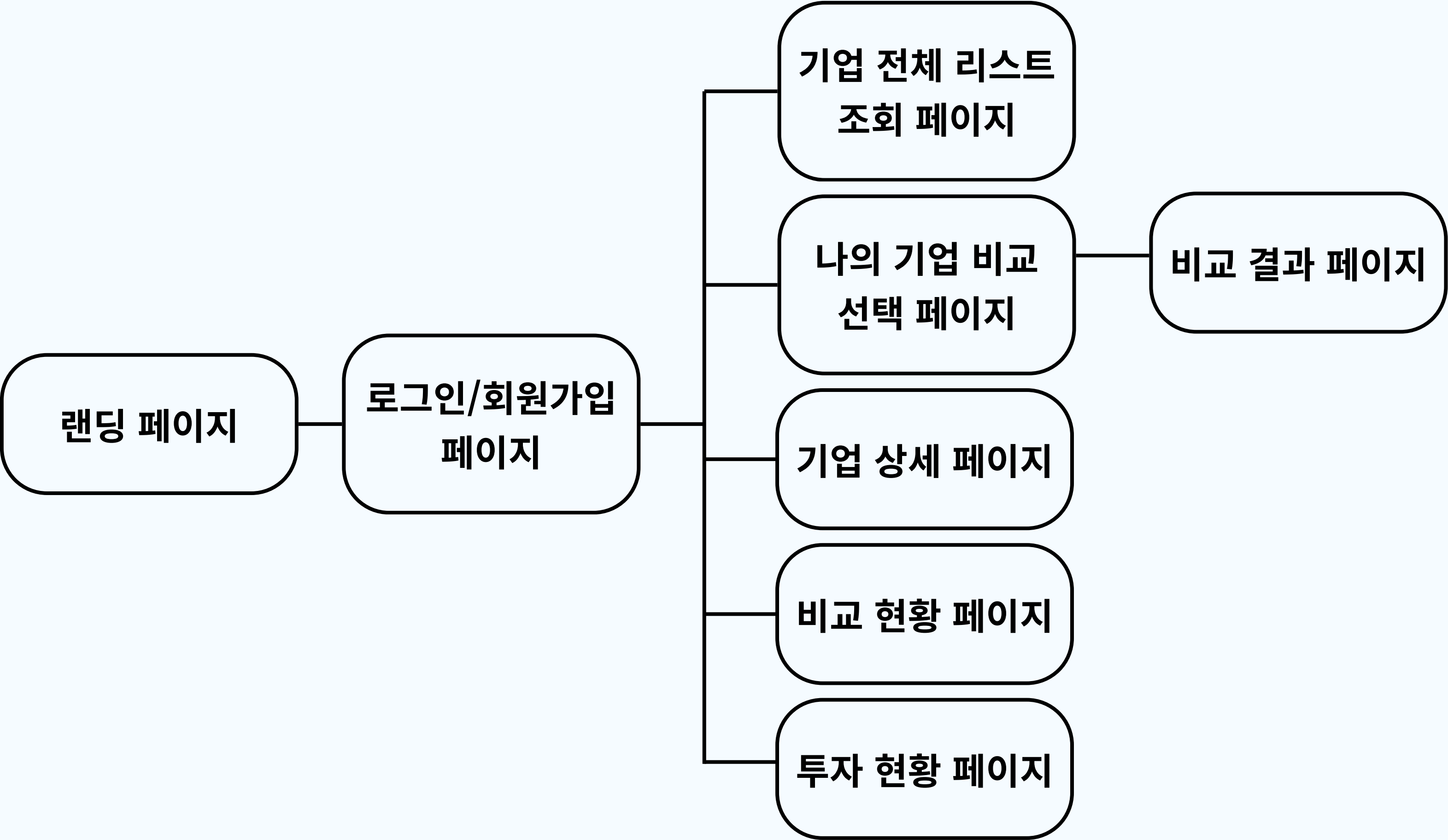
# 4. Procedures and Methods

프로젝트 일정



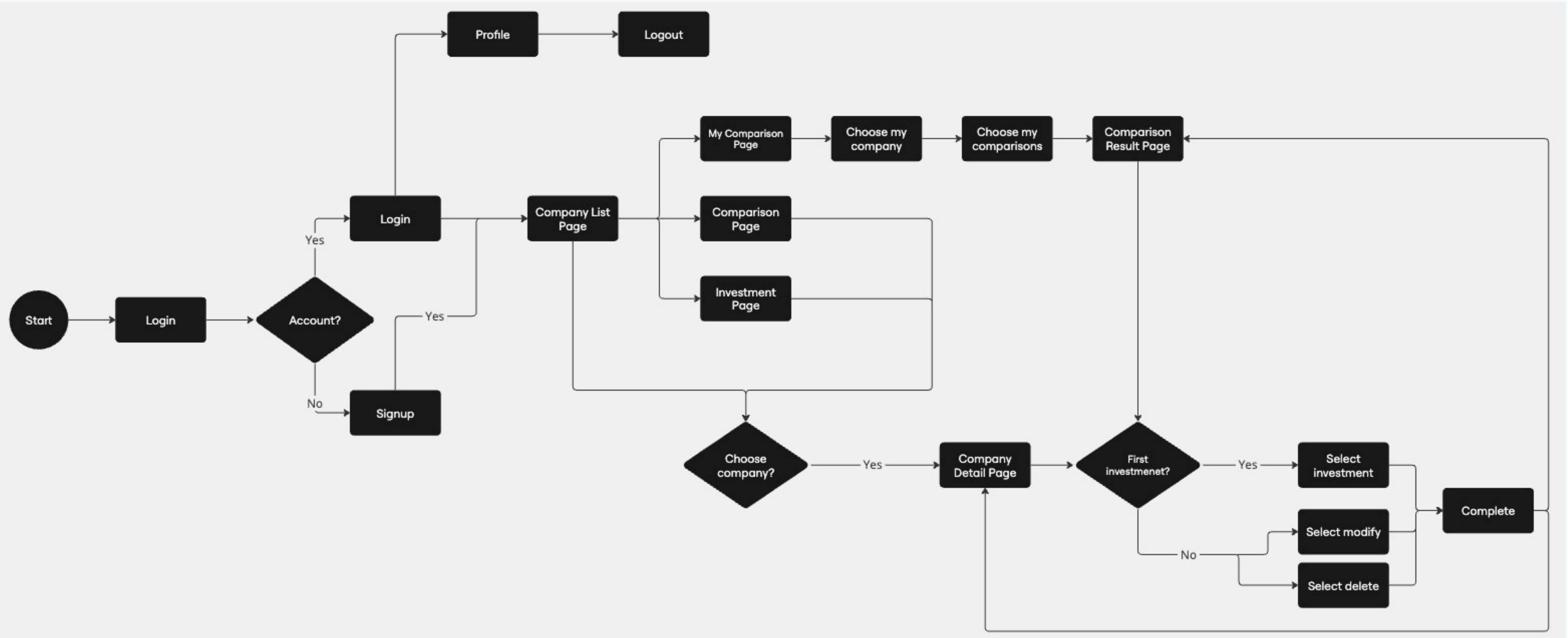
# 4. Procedures and Methods

페이지 플로우



## 4. Procedures and Methods

## 유저 플로우



# 4. Procedures and Methods

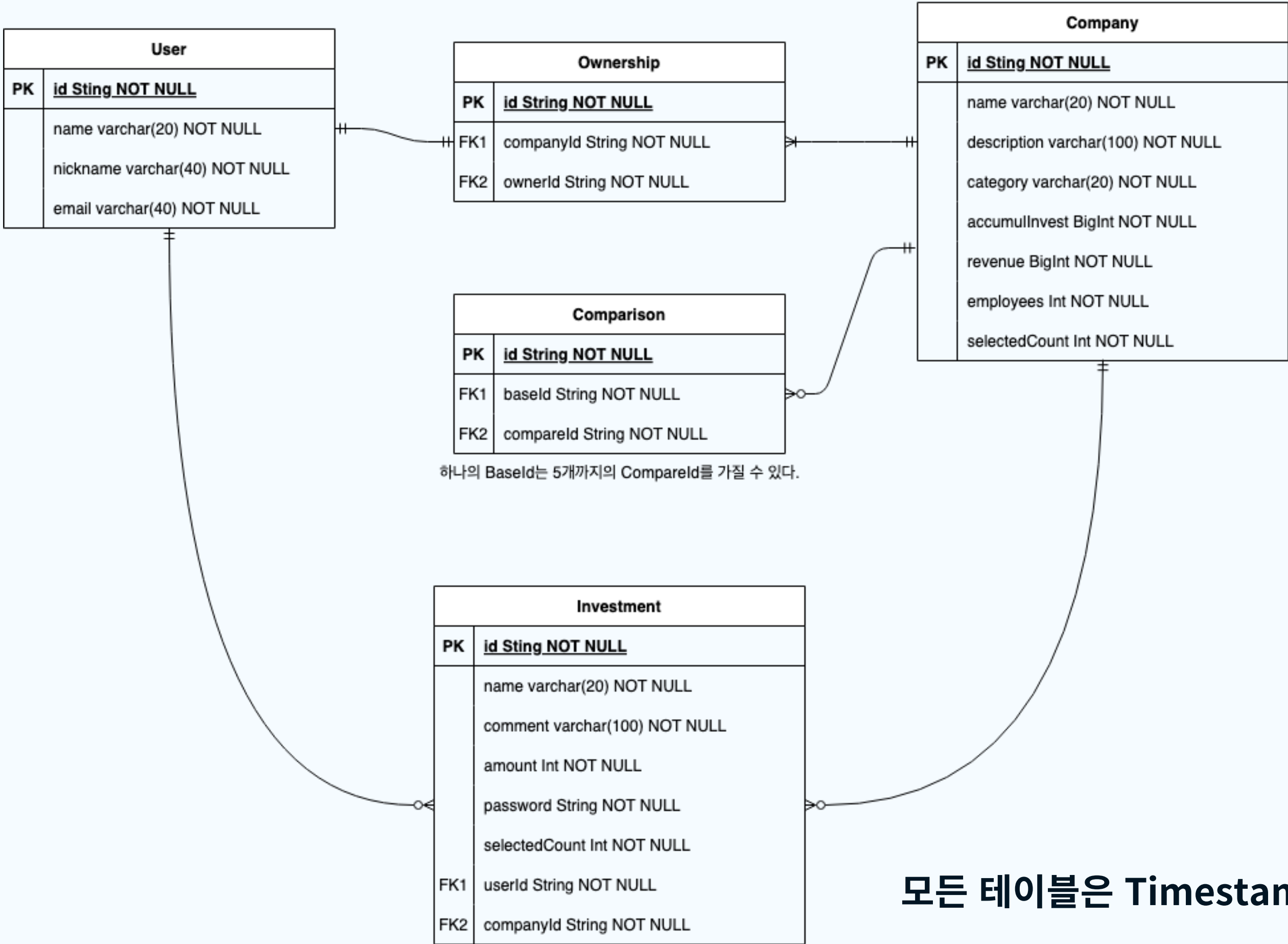
# API 명세

☯ 분류	≡ API Path	☯ HTTP 메서드	Aa 기능
company	/companies	GET	모든 기업 리스트 검색
company	/companies/:id	GET	특정 기업 정보 가져오기
comparison	/comparisons/select	POST	비교 기업 등록
comparison	/comparisons/select/:userId	GET	특정 유저가 선택한 비교 기업 리스트 가져오기
watch	/watches/select	POST	나의 기업 등록
watch	/watches/select/:userId	GET	특정 유저가 선택한 나의 기업 가져오기
investment	/investments	GET	투자 전체 정보 불러오기
investment	/investments/:companyId	GET	특정 기업에 해당하는 기업에 투자된 정보 불러오기
investment	/investments/:companyId/total	GET	기업에 투자된 투자들의 총 금액 불러오기
investment	/investments	POST	투자 등록
investment	/investments/:id	PATCH	특정 투자 수정
investment	/investments/:id	DELETE	특정 투자 삭제

account	/account/iter	POST	중요한 데이터를 전송, 수정, 받기 할때마다 체크해야되기 때문에 이런 작업이 이루어질때마다 session-iter 를 전송받아 sessionPwd 를 iter 만큼 hash 를 돌리고 성공하면 iter 가 하나씩 줄어드는 형식
account	/account/session-iter	POST	hash 함수를 전송받은 개개인의 salt 와 함께 client 단에서 몇번 hash function 을 돌린 뒤 전송 할지를 알려주는 기능
account	/account/sessions	POST	세션 정보 저장
account	/account/log-in	POST	로그인 정보 저장
account	/account/log-out	POST	로그아웃 잘 되었는지 메세지
account	/account/log-out-from-all	POST	모든 세션이 잘 지워졌는지 메세지
account	/account/check	POST	이메일, 닉네임 확인
account	/account/sign-up	POST	회원가입 정보 저장
account	/account/log-in-with-social	POST	정보가 잘 저장되었는지 결과 (true/false) 저장
account	/account/log-in-with-social.do	POST	소셜 로그인 정보 저장
account	/account/users	GET	전체 user 의 데이터, 실제 개발 모드에서는 보안을 위해 삭제할 예정

# 4. Procedures and Methods

ERD



모든 테이블은 Timestamp를 가진다.

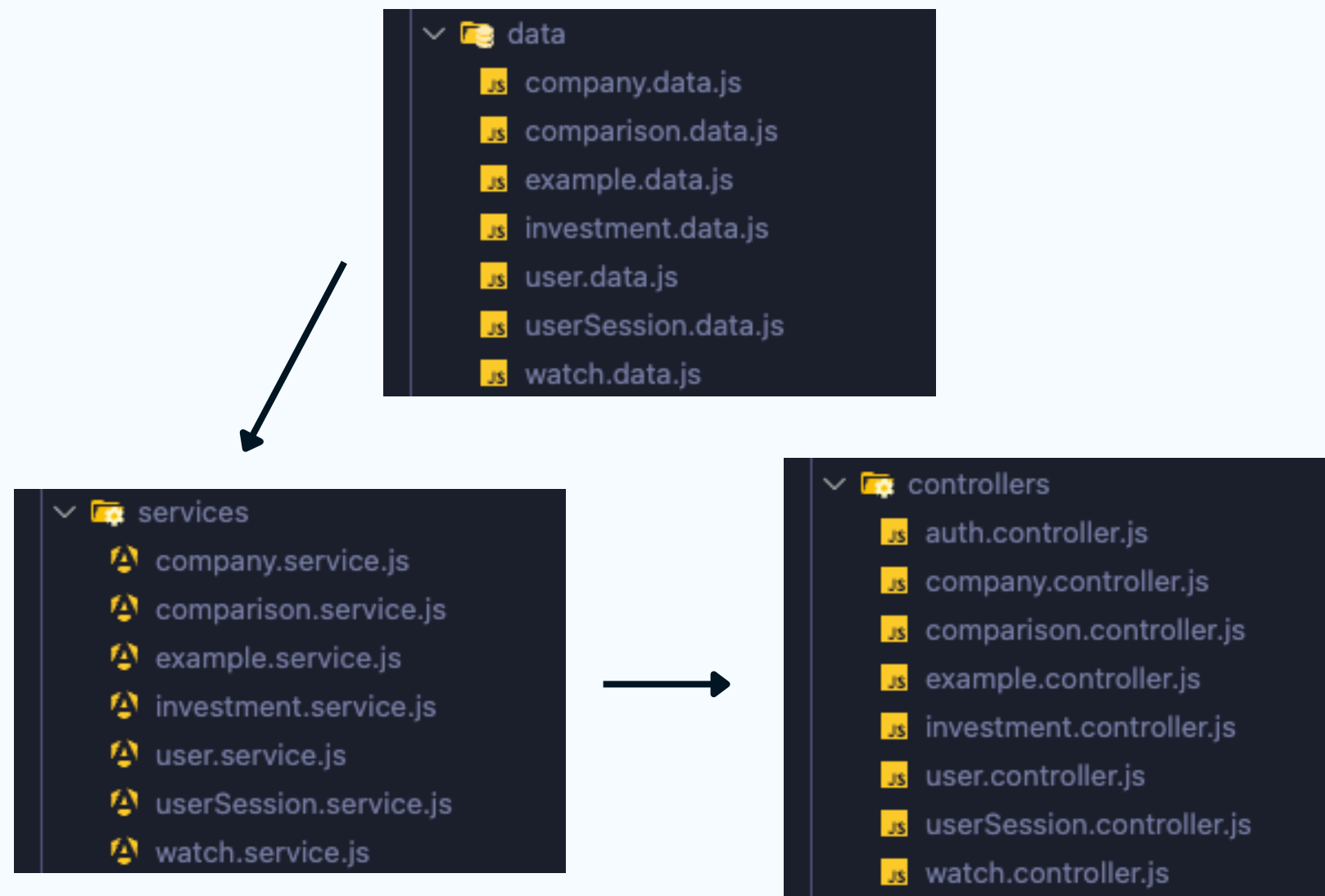


## 5. ✕Results

## 프로젝트 강점 1

### 3-tier 데이터베이스 아키텍처 (mvc)

3-tier 구조를 통해 사용자의 요청이 효과적으로 처리될 수 있도록 분산 아키텍처를 채택.  
프론트엔드는 사용자가 직접 상호작용하는 부분.  
비즈니스 로직은 요청을 처리하며 데이터베이스와 상호작용하여 필요한 데이터를 제공.



### 1 선택 이유

- 3-tier 구조를 사용함으로써 각 계층을 독립적으로 업데이트할 수 있어, 특정 요구사항 변경이나 기능 추가 시 더 유연하게 대응 가능.

### 2 각 계층의 역할

- controllers - api로써 요청 및 응답(Client tier)
- services - 데이터를 가공하여 controller에 전달(Application Tier)
- data - DB와 직접 통신(Data tier)

### 3 구조의 장점

- 데이터베이스 업데이트를 통해 빠르게 변경 사항을 반영할 수 있고, 비즈니스 로직과 프론트엔드를 독립적으로 업데이트하여 새로운 요구사항에 즉각적으로 대응.
- 확장성과 유지보수 측면에서 기업이나 다른 개발자가 쉽게 해당 프로젝트를 관리 가능.
- 특정 계층이 다른 계층에 영향을 주지 않고 업데이트될 수 있고, 다른 계층이 기존 상태를 유지하면서도 독립적인 기능 개발이 가능.

## 5. ✖ Results

### session 기반 hash 암호화 방식 로그인 구현

```
const encrypt = (salt, pwd0, iter) => {
  const pwd = salt + pwd0;
  return encrypt0(
    salt,
    hash1(pwd) +
    hash2(pwd) +
    hash3(pwd) +
    hash4(pwd) +
    hash5(pwd) +
    hash6(pwd) +
    hash7(pwd) +
    hash8(pwd) +
    hash9(pwd) +
    hash10(pwd) +
    hash11(pwd) +
    hash12(pwd) +
    hash13(pwd),
    iter,
  );
};

export const encryptRest = (salt, pwd0, iter) => {
  return encrypt0(salt, pwd0, ITER_FULL - iter);
};

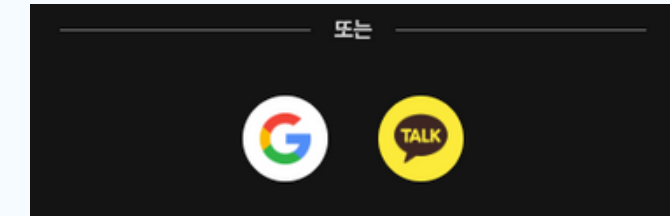
export const encryptSSNRest = (salt, pwd0, iter) => {
  return encrypt0(salt, pwd0, ITER_SSN_FULL - iter);
};

export function generateRandomHexString(length = 32) {
  const characters = '0123456789abcdef';
  let result = '';
  for (let i = 0; i < length; i += 1) {
    result += characters.charAt(Math.floor(Math.random() * characters.length));
  }
  return result;
};

const handleLogin = () => {
  if (validation.email && validation.pwd) {
    try {
      postPwdIter({ email })
        .then(async iter => {
          if (iter?.message) {
            setError(iter);
            return;
          }
          const pwdEncrypted = encrypt(iter.salt, pwd, iter.iter);
          setPwd('');
          const result = await postLogin({ email, pwdEncrypted });
          if (result) {
            setUser(result);
            localStorage.setItem('userUuid', result.userUuid);
            localStorage.setItem('nickname', result.nickname);
            localStorage.setItem('sessionPwd', result.sessionPwd);
            localStorage.setItem('createdAt', result.createdAt);
            return;
          }
        })
    }
  }
};
```

## 프로젝트 강점 2

### 소셜 간편 로그인 구현



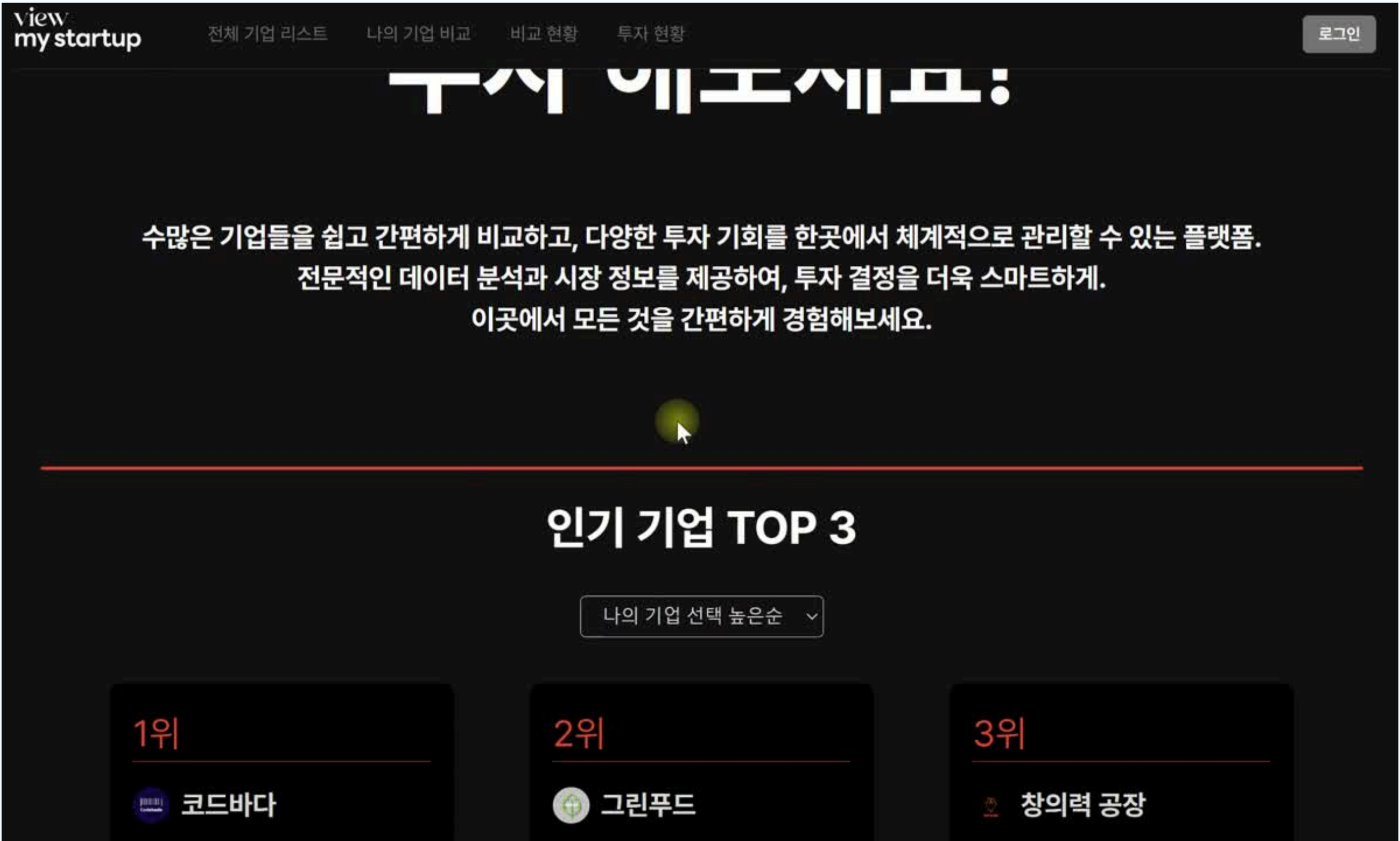
```
function LoginWithGoogle() {
  useEffect(() => {
    async function loginWithGoogle() {
      let { hash } = window.location;
      if (hash) {
        hash = hash.substring(1);
        const hashVars = hash.split('&');
        const hashMap = {};
        for (const hashVar of hashVars) {
          const keyNValue = hashVar.split('=');
          hashMap[decodeURIComponent(keyNValue[0])] = decodeURIComponent(keyNValue[1]);
        }
        if (hashMap.access_token && hashMap.token_type) {
          const resp = await getGoogle({ access_token: hashMap.access_token, token_type: hashMap.token_type });
          if (resp.verified_email && resp.email) {
            let sw;
            let sh;
            if (window.screenWidth > window.screenHeight) {
              sw = window.screenWidth;
              sh = window.screenHeight;
            } else {
              sw = window.screenHeight;
              sh = window.screenWidth;
            }
            const session = await postLoginWithGoogle({
              sw,
              sh,
              state: hashMap.state.val,
              email: resp.email,
            });
          }
        }
      }
    }
  });
}
```



5. Results

시연 영상

랜딩 페이지



5. Results

시연 영상

기업 상세 및 기업 투자하기








view  
my startup

[전체 기업 리스트](#)[나의 기업 비교](#)[비교 현황](#)[투자 현황](#)

로그인

전체 스타트업 목록

최신순

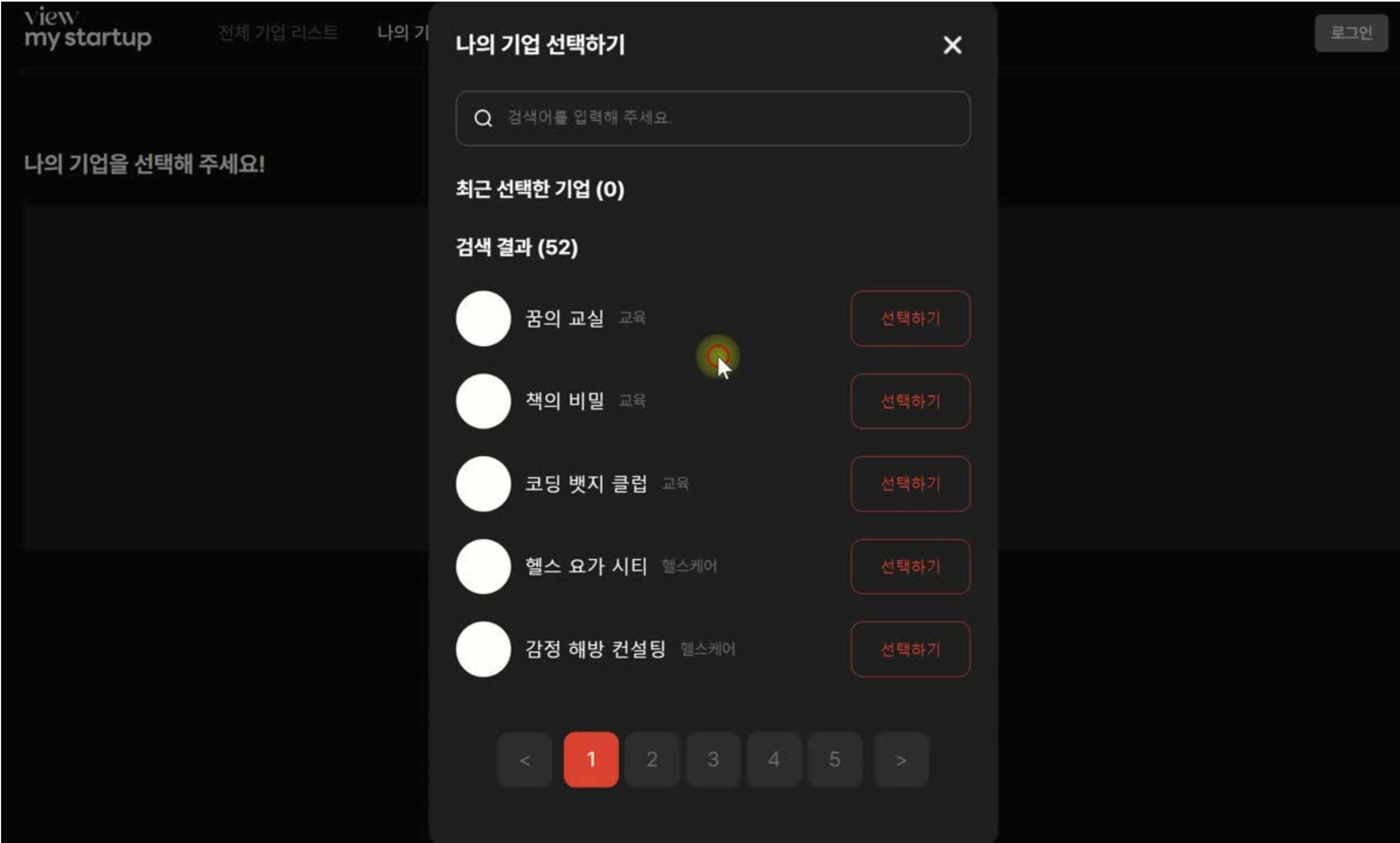
순위	기업 명	기업 소개	카테고리	누적 투자 금액	매출액	고용 인원
1위	 꿈의 교실	꿈의 교실은 모든 아이들이 꿈꾸는 미래를 실현할 수 있도록 돕는 교육 플랫폼입니다. 우리는 창의적이고 혁신적인 교육 방식을 ...	교육	55억원	220억원	35명
2위	 책의 비밀	책의 비밀은 독서와 문학을 사랑하는 이들을 위한 커뮤니티 플랫폼입니다. 우리는 독서 모임, 작가와의 대화, 그리고 문학 관련 ...	교육	45억원	170억원	15명
3위	 코딩 뱃지 클럽	코딩 뱃지 클럽은 프로그래밍 교육을 통해 젊은 인재를 양성하는 플랫폼입니다. 다양한 프로그래밍 언어와 기술을 배울 수 있는 ...	교육	45억원	180억원	20명
4위	 헬스 요가 시티	헬스 요가 시티는 전통 요가와 현대 운동을 접목한 헬스 프로그램을 제공하는 헬스케어 플랫폼입니다. 다양한 운동 수업과 건강 ...	헬스케어	35억원	145억원	15명
5위	 감정 해방 컨설팅	감정 해방 컨설팅은 개인의 정서적 건강을 위한 심리 상담과 워크숍을 제공하는 플랫폼입니다. 다양한 심리적 문제를 해결하기 ...	헬스케어	20억원	90억원	12명
6위	 녹색 에너지	녹색 에너지는 지속 가능한 에너지를 제공하는 스타트업으로, 태양광과 풍력 발전을 전문으로 하고 있습니다. 우리는 기업과 가...	IT/테크	45억원	150억원	30명
7위	 우주 탐험대	우주 탐험대는 우주와 관련된 교육과 체험 프로그램을 제공하는 스타트업입니다. 우리는 어린이와 청소년들이 우주에 대한 꿈을..	교육	30억원	90억원	12명

by-startup-by-team-1.netlify.app/companies

5.❌Results

시연 영상

기업 비교하기 및 비교 결과 확인





5.📊Results

시연 영상

기업 리스트, 비교·투자 현황 페이지 & 소셜 로그인 기능

view  
my startup

전체 기업 리스트

나의 기업 비교

비교 현황

투자 현황

로그인








전체 스타트업 목록

Q

검색어를 입력해주세요

최신순

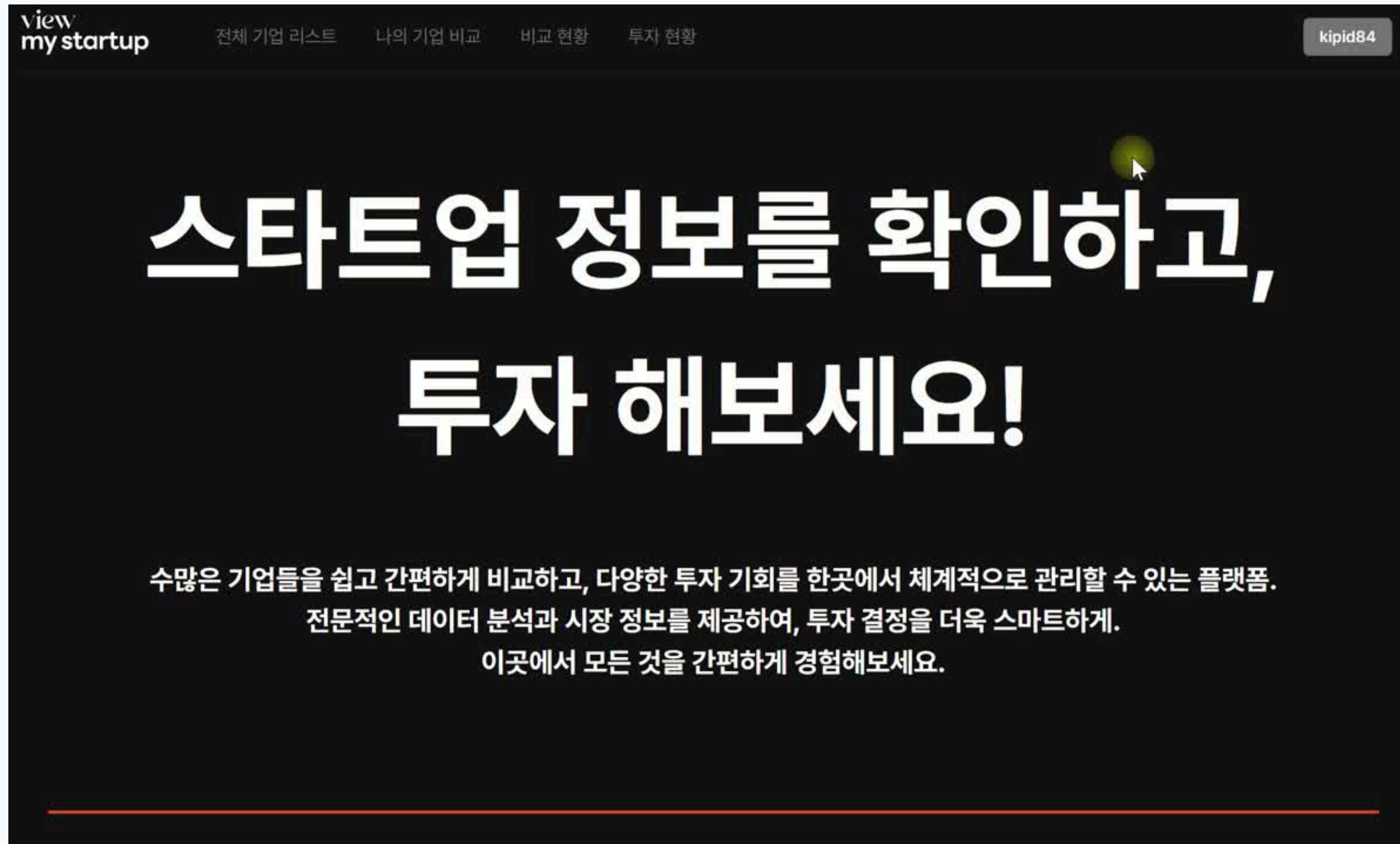
▼

순위	기업 명	기업 소개	카테고리	누적 투자 금액	매출액	
31위	 스마트 빌리지	스마트 빌리지는 농촌 지역에 최첨단 기술을 도입하여 농업 생산성을 높이는 스타트업입니다. 우리는 데이터 분석과 IoT 기술을..	농업	40억원	140억원	
32위	 지구를 지켜라	지구를 지켜라는 환경 보호를 위한 스타트업으로, 지속 가능한 제품을 개발하여 소비자에게 제공합니다. 우리는 일상생활에서 쉽..	IT/테크	50억원	150억원	22명
33위	 행복한 요리사	행복한 요리사는 식사 준비를 간편하게 만들어주는 식품 스타트업입니다. 우리는 신선한 재료를 기반으로 다양한 요리 키트를 ...	식품	30억원	100억원	15명
34위	 디지털 마케팅 허브	디지털 마케팅 허브는 중소기업을 위한 온라인 마케팅 솔루션을 제공하는 스타트업입니다. 우리는 다양한 마케팅 도구와 전략을..	IT/테크	55억원	160억원	20명
35위	 바삭한 소세지 마켓	바삭한 소세지 마켓은 신선하고 품질 좋은 소세지를 전문으로 판매하는 플랫폼입니다. 매일 아침 엄선된 재료를 사용하여 고객...	유통	30억원	125억원	25명
36위	 그린푸드	그린푸드는 건강한 먹거리를 제공하는 친환경 식품 유통 기업입니다. 그린푸드는 자체 농장에서 유기농으로 재배한 신선한 농...	유통	30억원	183억원	50명
37위	 드림테크	드림테크는 4차 산업혁명 기술을 선도하는 IT 기업으로, AI, 블록체인, IoT 등 혁신적인 기술을 기반으로 다양한 솔루션을 제공...	IT/테크	85억원	642억원	150명

## 5. Results

시연 영상

나의 투자 현황 추가



## 5.❏Results

서비스 배포 링크

[Netlify URL] <https://view-my-startup-by-team-1.netlify.app/>

[Render URL] <https://view-my-startup-codeit-be.onrender.com/>

[FE github URL] <https://github.com/kipid/View-My-Startup-Codeit>

[BE github URL] <https://github.com/kipid/View-My-Startup-Codeit-BE>

Netlify QR



## 6. Team Opinions

## 평가 의견



김영은

- 아쉬운 점

- 새로운 서비스를 처음부터 기획해서 만들어 보고 싶었는데, 주어진 프로젝트 컨셉이 명확해서 변경하기가 어려워서 그대로 제작했는데 그 점이 좀 아쉬움.
- 처음에 전체 구성에 대한 이해와 해석에 대해 전체 회의를 구체적으로 하지 않고 시작했는데, 각자 맡은 파트만 독립적으로 제작하다보니 중간중간 사소한 이슈가 발생하기도 함.

- 잘한 점

- 로그인 기능을 추가로 구현하여 실제 서비스를 배포 가능할 정도로 완성했다는 것.
- 모두 팀 작업은 처음이었는데, 초반 세팅 이후로는 큰 이슈 없이 협업한 것.
- 기존에 학습한 내용 외에 새로운 방식이나 기능을 잘 활용하고 적용한 것.

- 느낀 점

- 아직 기본적으로도 배울 게 많이 남아있지만, 실제 프로젝트마다 필요한 기술을 계속해서 배워야 한다는 것을 직접 느끼게 됨.
- 처음 만든 기본적인 컨벤션 외에는 각자 스타일로 작성하다 보니, 서로 영향을 주는 경우도 있어 공동 작업인 만큼 더 구체적으로 협의가 필요하다는 생각이 듦.

## 6. Team Opinions

## 평가 의견



김재원

- 아쉬운 점

- 팀원들에 비해서 많이 부족하여 초반에 잘 못따라가고, 오히려 아키텍처나 만들어 놓은 코드를 파악하고 공부하는 데에 시간을 많이 씀.
- 초반 세팅 외에 API 명세 등 정해야할 것들을 다 정해놓지 않고 협업하여, 약간의 혼동이 왔던 것 같음.
- 프로젝트 기간 동안 거의 백엔드 위주로 다루다 보니, 풀스택 역량 개발을 제대로 못해본 것 같음.

- 잘한 점

- 디자인이나 설계가 주어지지 않은 랜딩 페이지를 잘 만들어낸 것.
- MVC 패턴을 공부하면서 바로 적용시킨 점.
- 각자 충실히 팀플에 참여하며, 프로젝트를 완성시킨 것.

- 느낀 점

- 깃을 제대로 다루는 게 처음이었지만, 협업이 많이 어렵다는 것을 느끼고 내가 해야할 일 보다 개발 진행 방향과 중간 점검의 중요성을 알게 됨.
- 항상 배운 것에 대해 실습과 복습을 잘 해 놓아야 함을 느끼고, 다음 프로젝트에서는 좀 더 많은 것을 해보겠다고 다짐하게 됨.



## 6. Team Opinions

## 평가 의견



김태영

- 아쉬운 점

- 주어진 플로우나 디자인에서 요구사항과 어울리지 않는 것들이 있는데, 자연스럽게 변경하지 못했던 것 같음.
- 첫 팀 프로젝트라 초기에 정해야 할 것들을 제대로 정하지 못하고 진행했던 점.
- 프로젝트 분량을 제대로 고려하지 못해서 파트 배분을 제대로 하지 못한 점.
- 개인 팀원과 제대로 소통하지 못하고 감정에 맡겨 팀워크를 해친 점.

- 잘한 점

- 초기 컨벤션 세팅과 아키텍처 세팅 이후 이를 최대한 잘 지켜낸 점.

- 느낀 점

- 너무 자신의 감정에 매몰되면 안될 것 같다고 생각했음. 분명 상대도 프로젝트에 기여하려는 의도가 있다는걸 고려해야한다.
- css 간섭이 자주 일어났음. 겪어보니 최근 현업에서 css-in-js 방식이나 tailwind 같은 utility class 라이브러리를 사용하는 이유를 알게 됐음.

## 6. Team Opinions

## 평가 의견



이강수

- 아쉬운 점

- 협업이 거의 처음이다 보니, 팀과의 소통에 미흡한 점이 많았던 것 같음. 팀원들이 짠 코드를 내 맘대로 수정한 점이나 그 내용을 디스코드 등을 통해 알리지 않고 진행해서 팀원들을 곤란하게 만들었던 것 같음.
- 사소한 실수로 버그들이 자주 생겼는데, 처음 코딩할 때 신경을 좀 더 썼다면, 이런 버그들을 해결하는데 드는 시간을 엄청 아낄 수 있었을거란 생각이 듦. 버그들이 생기지 않게 코딩 습관을 잘 들일 필요가 있다고 생각 됨.

- 잘한 점

- 학습한 내용을 바탕으로 필수 기능들을 완성한 것.
- 학습했던 것 이상으로 MVC 패턴이나 로그인 구현 (소셜 로그인) 까지 해낸 것.

- 느낀 점

- 협업 개발을 할 때, 소통의 중요성 및 파트 분배의 중요성을 뼈저리게 깨달음.
- 버그들이 생기지 않게 코딩 습관을 잘 들이는 것과 어쩔 수 없이 버그들이 생겼을 때 빠르게 디버깅 할 수 있는 능력을 키워야겠다고 생각함.

## 6. Team Opinions

## 평가 의견

### 정리 ✨

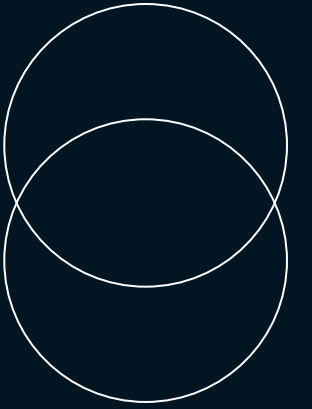
- 아쉬운 점

- 처음에 전체 구성에 대한 이해와 해석에 대해 전체 회의를 구체적으로 하지 않고 시작했는데, 각자 맡은 파트만 독립적으로 제작하다보니 중간중간 사소한 이슈가 발생하기도 함.
- 협업이 거의 처음이다 보니, 팀과의 소통에 미흡한 점이있어서 코드 이슈가 생겨 곤란한 상황이 생겼음. 그래도, 이후 팀원 전체가 열심히 소통하여 이슈가 생기지 않도록 노력해 프로젝트를 잘 완수해 낼 수 있었음.
- 주어진 플로우나 디자인을 자연스럽게 변경하지 못했던 것 같음. 그대로 제작했는데 그 점이 좀 아쉬움.

- 잘한 점

- 기존에 학습한 내용 외에 새로운 방식이나 기능을 잘 활용하고 적용한 것. MVC 패턴이나 로그인 구현 (소셜 로그인).
- 초기 컨벤션 세팅과 아키텍처 세팅 이후 이를 최대한 잘 지켜낸 점.
- 완성도가 높다는 점.





# Thank You

FS 2기 1팀 김영은 김재원 김태영 이강수

[풀스택 초급 프로젝트 가이드] <https://www.codeit.kr/topics/fs-project-guide/lessons/10484>

[풀스택 초급 프로젝트 주제 안내] <https://www.codeit.kr/topics/fs-basic-project-topic-guide/lessons/10493>

[디자인 Figma] <https://url.kr/ewqstb>

[페이지 및 유저 플로우 Figma] <https://url.kr/v2sm7s>

---