

커뮤니티 중심 클라우드 펀딩 사이트

펀딩할래!

JavaChip First Project

자바(JAVA)활용 데이터 플랫폼 구축 풀스택 개발자 과정

2024.08.14 - 2024.08.29

진영신 · 박예슬 · 전영준 · 장희진 · 정찬희

팀 소개



진영신

팀장

Funding Page

리워드 페이지

커뮤니티 페이지

창작자 업데이트

마이페이지



박예슬

문서 총괄

회의 기록

Funding Page

리워드 페이지

후기 페이지



전영준

로그인

소셜 로그인

회원가입

회원가입 DB

사용자 DB



장희진

메인 페이지 내

Header

Contents

Footer

프론트엔드 구현



정찬희

후원자 가이드

창작자 가이드

인기 프로젝트

홈 화면 팝업

Introduction

- 1) 기술 환경
- 2) 프로젝트 개요
- 3) 프로젝트 소개
- 4) 프로젝트 진행 순서
- 5) 프로젝트 진행 일정

1. 기술 부분

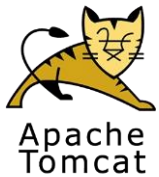
Back-End



Front-End



DataBase and Server



DBBeaver

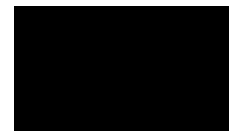
ORACLE

Open API

Kakao Login API

Naver Login API

Collaboration



2. 프로젝트 개요

배운 내용을
활용했습니다.

학원에서 배운 기능을 최대한 활용하자!



아직 배우지 못한 기능을 탐구하고 적용해 보자!



더 나은 서비스를 구현할 수 있을까?



3. 프로젝트 소개

커뮤니티 크라우드 펀딩

‘펀딩할래!’는

커뮤니티 중심 크라우드 펀딩 플랫폼입니다.

신뢰와 소속감을 기반으로, 자신이 속한 커뮤니티 내에서

프로젝트를 지원할 수 있는 기회를 제공합니다.

3. 프로젝트 소개

커뮤니티의
신뢰성과 소속감



커뮤니티 기반으로 운영되어
신뢰할 수 있고,

후원자들이 소속감을
느낄 수 있습니다.

이로 인해 펀딩 참여 의지를
높입니다!

프로젝트
맞춤형 지원



관심사에 맞는
프로젝트 지원으로

후원자들의 취향에 맞는 맞춤형
지원이 가능합니다.

높은 참여율과 성공적인 펀딩을
기대할 수 있습니다.

투명하고
직관적인 의사소통



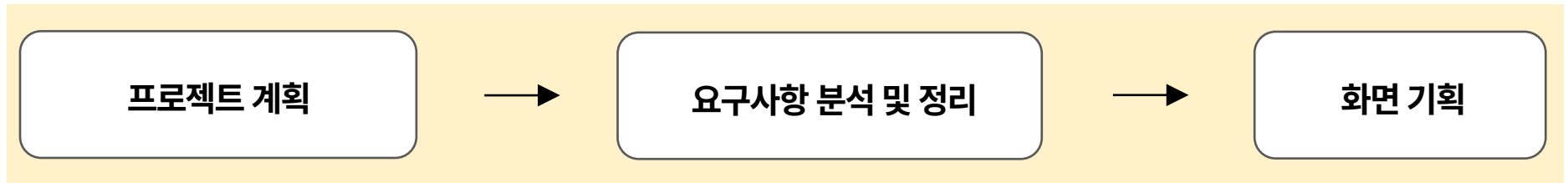
커뮤니티를 통해
투명하고 직관적인
소통으로

프로젝트의 진행 상황을 실시간으로
확인할 수 있기 때문에

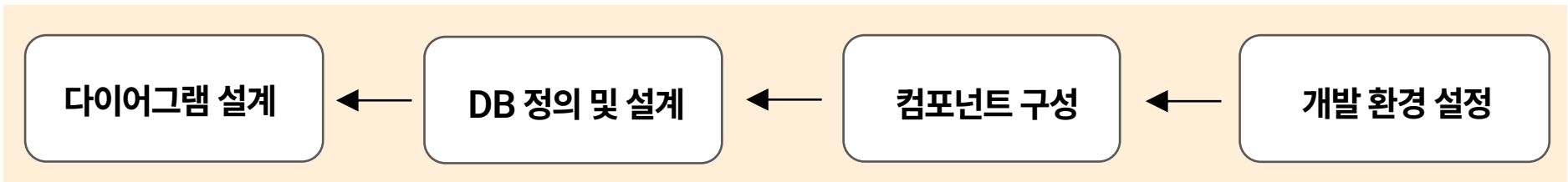
지속적인 참여를 유도합니다!

4. 프로젝트 진행 순서

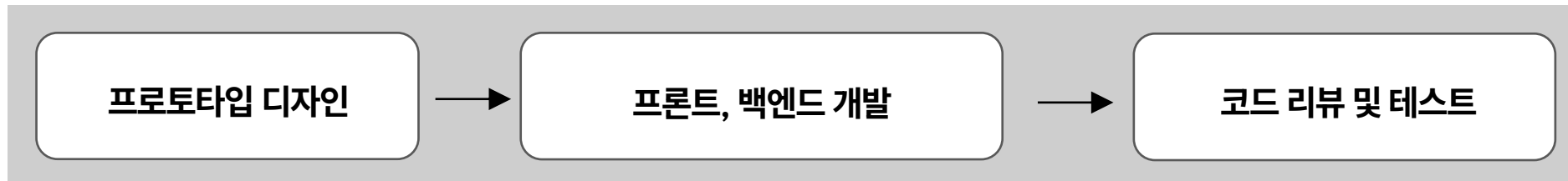
기획



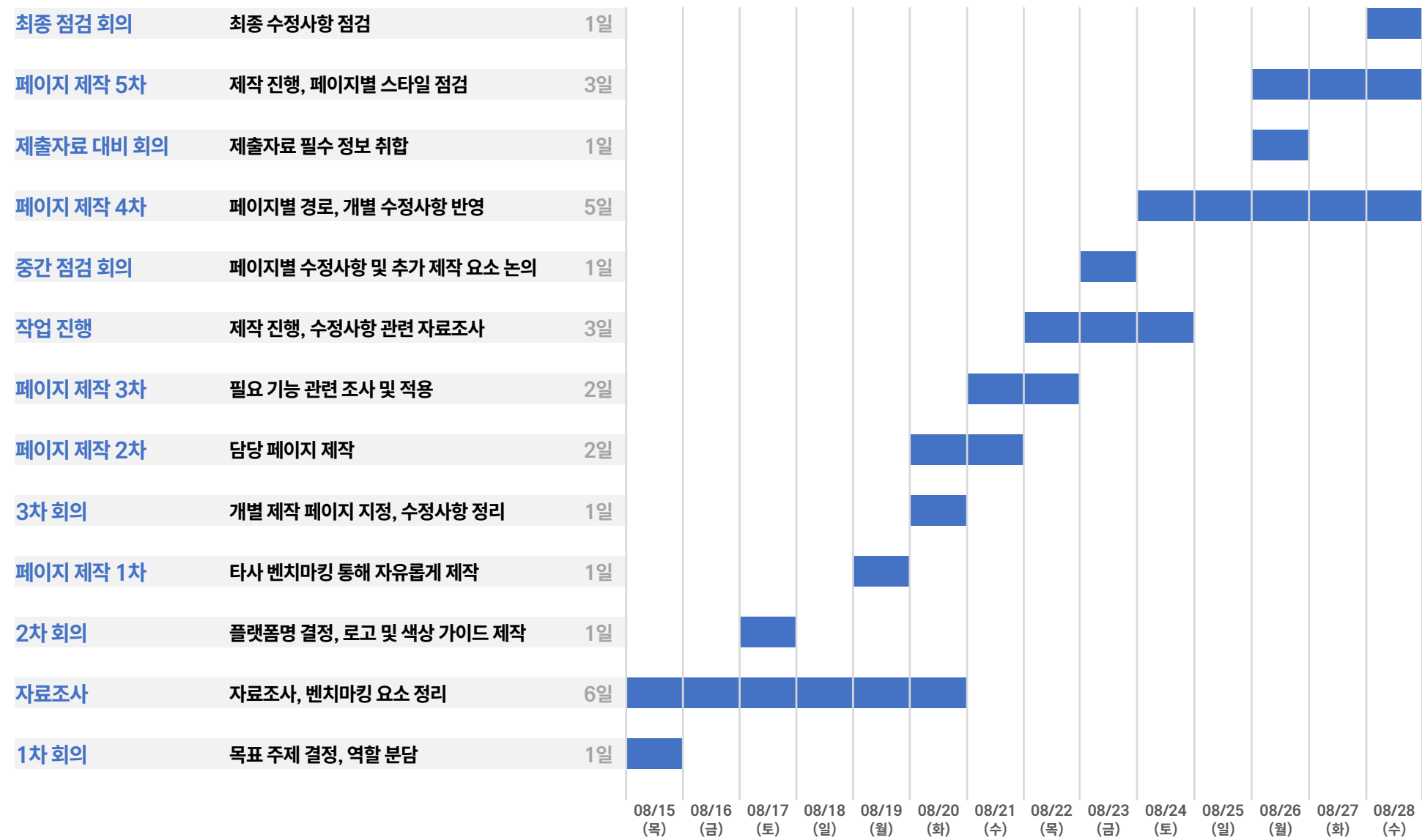
분석/설계



이행



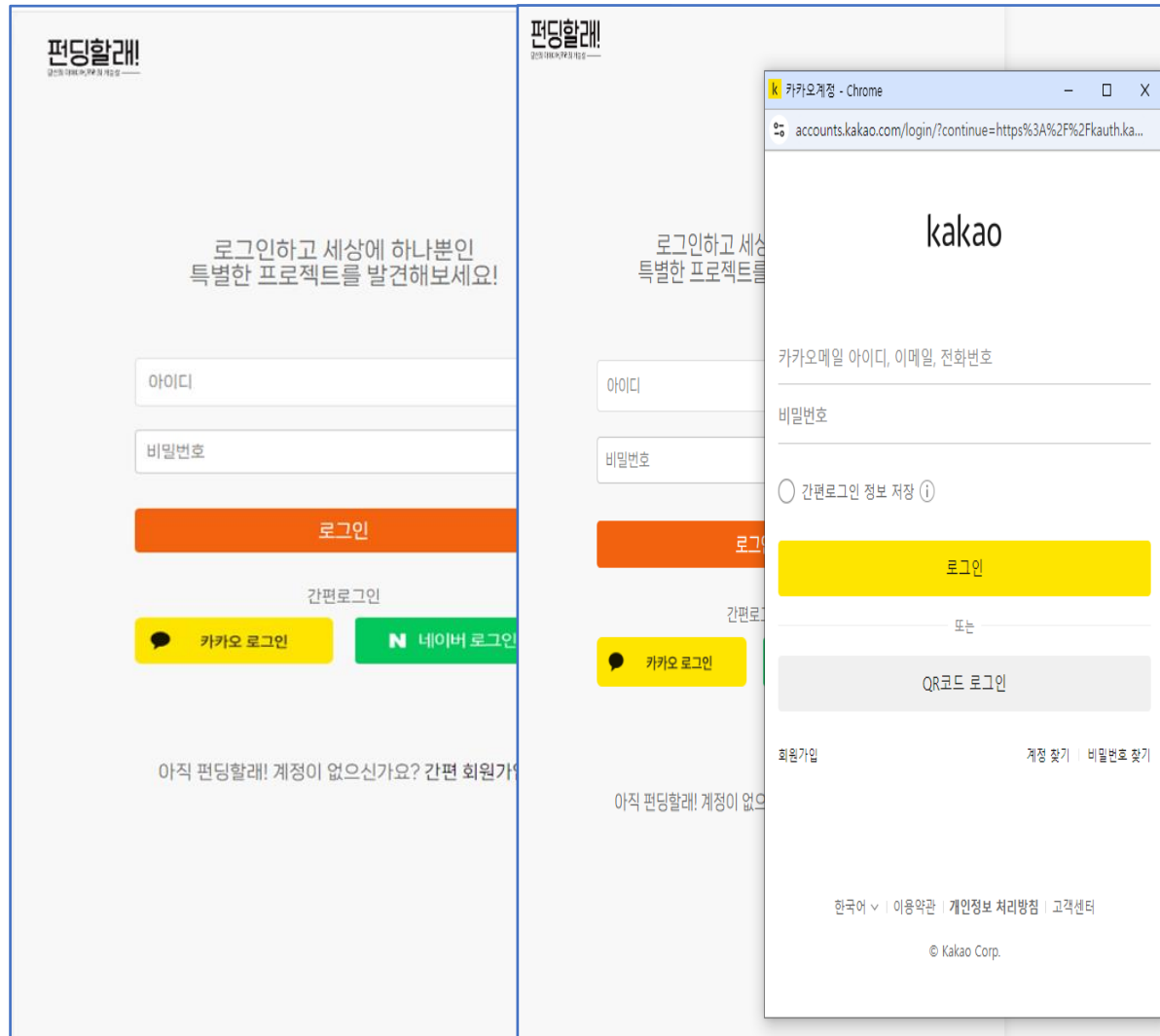
5. 프로젝트 진행 일정



Contents

- 1) 핵심 기능 소개
- 2) 각 페이지 구현 기능
- 3) ERD 설계
- 4) Class 다이어그램 설계
- 5) UseCase 다이어그램 설계

1. 핵심 기능 소개



회원 가입 및 로그인



인기 프로젝트(실시간 랭킹)

1. 핵심 기능 소개

편딩할래!

단편작가 커뮤니티와 창작자 지원 플랫폼

로그인/회원가입

프로젝트 만들기

프로젝트 계획

업데이트

커뮤니티

후기

창작자 업데이트

최종 작성 일자: 08-27 09:39

작성하기

창작자만 글을 쓸 수 있어요.

안녕하세요 ^^ 코난을 사랑해 주시는 모든 분들께 감사의 말씀 전합니다. 저 역시도 코난과 함께 나이를 들 어가고 있어요. 어린 시절 코난이 조금 무서웠던 기억이 있지만, 시간이 지나면서 코난 같은 애니메이션이 또 없었어요! 다들 마음에 품고 있는 어린 시절 애니메이션 하나씩은 있잖아요? 그것만으로 얼마나 삶의 위 안이 되는지 몰라요. 여기 게시는 많은 분들도 저랑 같은 마음이었죠!! 다들 이번 프로젝트로 인해 코난과 함

후원자만 글을 쓸 수 있어요.

작성하기

U 코난이 벌써 ..30주년이라니 남도일 돌아와~~~~

사용자 2024. 8. 27. 오전 9:36:55

U 코난 덕후는 여기서 쉬어갑시다!!!!

사용자 2024. 8. 27. 오전 9:36:55

U 코난 최고예요!

사용자 2024. 8. 27. 오전 9:36:55

프로젝트 내 커뮤니티

편딩할래!

단편작가 커뮤니티와 창작자 지원 플랫폼

로그인/회원가입

프로젝트 만들기

후기

리뷰를 작성하세요.

작성하기

코난 진짜..덕후.. 코난 굿즈 모은지 N년 차.. 이런 구성에 이런 해자 가격..해해.. 넘 예뻐요♥ 달력도 예쁘구.. 포토 카드도 원하는 게 나와요..지루하거나 심 심할 때 코난 보면서 시간을 보냈고, 코난 굿즈를 모으기 시작하면서 지금은 알아지지만 행복감은 배로 되는 것 같아요 🥰🥰 코난굿즈 모으는 재미가 쏠쏠 해유 퀄리티 있는 제품이라 더 만족스러워요♥

찬 2024. 8. 26. 오전 11:50:29

유용해요

장의적이에요

특별해요

퀄리티가 좋아요

의미 있어요

소통이 잘 돼요

너무 좋아요~🥰🥰🥰 코난은 저의 어린 시절 추억 그 자체인데.. 구성 전부 좋네요.기다린 보람이 있는 퀄리티예요! 좋은 프로젝트 감사합니다♥다음에는 더 알찬 구성으로 크라우드 펀딩 부탁드려요!!!!

예 2024. 8. 26. 오전 11:42:24

유용해요

장의적이에요

특별해요

퀄리티가 좋아요

의미 있어요

소통이 잘 돼요

드디어 너를 만났구나!!!!!!!!!!!!!!!설레는 마음이 가라앉지 않아서 이제서야 후기 씀니다ㅠㅠ♥♥가격이 좀 비싸긴 했지만 너무 예쁘네요 🥰 내 인생 최고의 애니메이션 코난.. 올해 마지막 달 최고의 선물이에요 🥰 감사합니다 🥰 완전 이빠요..! 이걸 아까워서 어떻게 쓰나.. 🥰♥ 달력 걸어놓고 한참을 바라봤네요..

희 2024. 8. 24. 오후 4:58:44

유용해요

장의적이에요

특별해요

퀄리티가 좋아요

의미 있어요

소통이 잘 돼요

말해줘..♥♥♥ 안산으면 당장 후회했을것.....포토카드와 키링까지..늦었지만 넘 잘 받았어요 코난 굿즈 다 예쁘더라고요! 랜덤 포토카드도 너무 갖고 싶던 게 나와서 진짜 넘 행복♥ 얼리버드 가격으로 구매할 수 있어서 대만족이에요 광명

영 2024. 8. 24. 오후 4:23:18

유용해요

장의적이에요

특별해요

퀄리티가 좋아요

의미 있어요

소통이 잘 돼요

후기 게시판

1. 핵심 기능 소개 - 회원가입 및 로그인

```
const handleKakaoLoginSuccess = async (kakaoAccessToken) => {
  // Kakao 로그인 성공 후 서버와 통신
  try {
    const response = await axios.post('http://localhost:9000/api/auth/kakao', {
      token: kakaoAccessToken,
    }, {
      headers: { 'Content-Type': 'application/json' }
    });

    console.log('서버 응답:', response.data);

    // 로그인 정보 기록 및 로그인 상태 업데이트
    const setIsLoggedIn = async (kakaoAccessToken) => {
      setIsLoggedIn(true); // 로그인 상태 업데이트
      localStorage.setItem('isLoggedIn', 'true'); // 로그인 상태를 로컬 스토리지에 저장
      navigate("/"); // 메인 페이지로 이동
    } catch (error) {
      console.error('카카오 인증 중 오류:', error);
      alert('카카오 로그인 처리 중 오류가 발생했습니다. 다시 시도해주세요.');
```

```
const fnGetKakaoUserInfo = async (kakaoAccessToken) => {
  // Kakao 사용자 정보 가져오기
  try {
    const res = await axios.get("https://kapi.kakao.com/v2/user/me", {
      headers: { "Authorization": `Bearer ${kakaoAccessToken}` }
    });

    const { id, kakao_account: { profile: { nickname } } } = res.data;
    await fnKakaoUserInfoCheck(id.toString(), nickname); // 서버로 사용자 정보 전송
    console.log(`카카오에서 온 아이디: ${id}, 닉네임: ${nickname}`);
  } catch (e) {
    console.log('카카오 사용자 정보 가져오기 중 오류:', e);
  }
};
```

```
const fnGetKakaoUserInfo = async (kakaoAccessToken) => {
  // Kakao 사용자 정보 가져오기
  try {
    const res = await axios.get("https://kapi.kakao.com/v2/user/me", {
      headers: { "Authorization": `Bearer ${kakaoAccessToken}` }
    });

    const { id, kakao_account: { profile: { nickname } } } = res.data;
    await fnKakaoUserInfoCheck(id.toString(), nickname); // 서버로 사용자 정보 전송
    console.log(`카카오에서 온 아이디: ${id}, 닉네임: ${nickname}`);
  } catch (e) {
    console.log('카카오 사용자 정보 가져오기 중 오류:', e);
  }
};
```

```
const fnKakaoUserInfoCheck = async (id, name) => {
  // Kakao 사용자 정보를 서버에 저장
  try {
    const response = await axios.post('http://localhost:9000/api/auth/register', {
      id,
      name
    }, {
      headers: { 'Content-Type': 'application/json' }
    });

    console.log('카카오 회원정보 추가 완료 응답:', response.data.result);
  } catch (error) {
    console.error('카카오 회원정보 추가 중 오류:', error);
  }
};
```

1. 핵심 기능 소개 - 인기 프로젝트

```
const RankingList = () => {
  // 1. 데이터 불러오기 및 상태 관리:
  const [ranks, setRanks] = useState([]); // 랭킹 데이터를 저장하는 상태
  const [loading, setLoading] = useState(true); // 데이터 로딩 상태 관리
  const [error, setError] = useState(null); // 에러 상태 관리

  useEffect(() => {
    const fetchData = async () => {
      try {
        const data = await fetchRankingList(); // API 호출을 통해 랭킹 데이터
        console.log(data); // 데이터 확인용 로그
        if (Array.isArray(data)) {
          setRanks(data); // 데이터가 배열일 경우 상태 업데이트
        } else {
          setRanks([]); // 데이터가 올바른 형식이 아닐 경우 빈 배열로 설정
          setError('Invalid data format received from API'); // 에러 메시지
        }
      } catch (err) {
        setError(err.message); // API 호출 실패 시 에러 상태 업데이트
      } finally {
        setLoading(false); // 로딩 완료
      }
    };

    fetchData(); // 컴포넌트 마운트 시 데이터 불러오기
  }, []); // 빈 배열을 의존성으로 설정해 컴포넌트가 처음 마운트될 때 한 번만 실행
```

// 2. 좋아요 기능 구현:

```
const handleLike = async (rankId) => {
  try {
    const encodedRankId = encodeURIComponent(rankId); // 랭킹 ID를 인코딩

    // 현재 랭킹 리스트에서 해당 ID를 가진 항목의 좋아요 수를 증가시킴
    const updatedRanks = ranks.map(rank => {
      if (rank.rid === rankId) {
        return { ...rank, rlike: parseInt(rank.rlike) + 1 }; // 좋아요 수 증가
      }
      return rank;
    });

    // 좋아요 수에 따라 랭킹을 다시 정렬
    const sortedRanks = updatedRanks.sort((a, b) => b.rlike - a.rlike);

    setRanks(sortedRanks); // 정렬된 랭킹 리스트로 상태 업데이트
    await updateRankingLike(encodedRankId); // 서버에 좋아요 업데이트 요청
  } catch (err) {
    setError(err.message); // 에러 발생 시 상태 업데이트
  }
};
```

1. 핵심 기능 소개 - 커뮤니티

```
74 // 창작자 업데이트 가져오기
75 export function fetchCreatorUpdate(gubun) {
76   return call(`/creator_updates/latest/${gubun}`, "GET", null);
77 }
78
79 // 창작자 업데이트 추가
80 export function addOrUpdateCreatorUpdate(update, gubun) {
81   const updateGubun = { ...update, gubun };
82   return call(`/creator_updates/${gubun}`, "POST", updateGubun);
83 }
84
85 // 창작자 게시물 작성 날짜 (JSON형식)
86 export function fetchLatestUpdateDate(gubun) {
87   return call(`/creator_updates/latest-date/${gubun}`, "GET", null).then(response => {
88     if (response.date) {
89       return response.date;
90     }
91     throw new Error("데이터를 찾을 수 없습니다.");
92   });
93 }
94
```

```
7 <!-- 창작자 업데이트 삽입 -->
8 <insert id="insertComment" parameterType="com.example.demo.dto.CreatorCommentDTO">
9   INSERT INTO creator_updates (username, text, created_at, updated_at, gubun)
10   VALUES ({username}, #{text}, CURRENT_TIMESTAMP, CURRENT_TIMESTAMP, #{gubun})
11 </insert>
12
13 <!-- 특정 gubun에 해당하는 창작자 업데이트 한 줄만 조회 -->
14 <select id="getCommentByGubun" parameterType="int" resultType="com.example.demo.dto.CreatorCommentDTO">
15   SELECT id, username, text, created_at AS createdAt, updated_at AS updatedAt
16   FROM (
17     SELECT id, username, text, created_at, updated_at
18     FROM creator_updates
19     WHERE gubun = #{gubun}
20     ORDER BY id DESC
21   )
22   WHERE ROWNUM = 1
23 </select>
24
25 <!-- 창작자 업데이트 수정 -->
26 <update id="updateComment" parameterType="com.example.demo.dto.CreatorCommentDTO">
27   UPDATE creator_updates
28   SET text = #{text}, updated_at = CURRENT_TIMESTAMP
29   WHERE id = #{id} AND gubun = #{gubun}
30 </update>
31
32 <!-- 창작자 업데이트 삭제 -->
33 <delete id="deleteComment" parameterType="map">
34   DELETE FROM creator_updates
35   WHERE id = #{id} AND gubun = #{gubun}
36 </delete>
37
```

1. 핵심 기능 소개 - 후기

```
const ReviewList = ({ gubun }) => { // ReviewList 컴포넌트 정의
  const [reviews, setReviews] = useState([]); // 리뷰 목록 상태 관리
  const [newComment, setNewComment] = useState(''); // 입력된 새로운 댓글 상태 관리
  const [editComment, setEditComment] = useState(''); // 수정 중인 댓글 상태 관리
  const [isEditing, setIsEditing] = useState(false); // 현재 편집 모드 여부 상태 관리
  const [currentReview, setCurrentReview] = useState(null); // 현재 편집 중인 리뷰 상태 관리
  const [isDialogOpen, setIsDialogOpen] = useState(false); // 다이얼로그 열림 상태 관리
  const [selectedTags, setSelectedTags] = useState([]); // 선택된 태그 상태 관리
  const [sparkleId, setSparkleId] = useState(null); // Sparkle 애니메이션을 적용할 리뷰 ID 상태 관리
  const [heartId, setHeartId] = useState(null); // Heart 애니메이션을 적용할 리뷰 ID 상태 관리

  useEffect(() => { // 컴포넌트가 마운트될 때와 gubun이 변경될 때 리뷰 데이터를 로드
    loadReviews(); // 리뷰 로드 함수 호출
  }, [gubun]); // gubun이 변경될 때마다 useEffect 재실행

  const loadReviews = () => { // 리뷰 데이터를 불러오는 함수
    fetchReviews(gubun).then(response => { // fetchReviews 함수로 리뷰 데이터 가져오기
      if (response) { // 응답이 있으면
        const reviewsWithTags = response.map(review => ({ // 각 리뷰에 태그 상태 추가
          ...review, // 기존 리뷰 데이터 유지
          selectedTags: Array.isArray(review.selectedTags) ? review.selectedTags : [],
          selectedTags가 없거나 배열이 아니면 빈 배열로 초기화
        })));
        setReviews(reviewsWithTags); // 리뷰 상태 업데이트
      }
    }).catch(error => { // 에러 발생 시
      console.error("리뷰를 로드하는데 실패했습니다.: ", error); // 에러 로그 출력
    });
  };

  const handleNewCommentChange = (e) => { // 새로운 댓글 입력 시 상태 업데이트 함수
    setNewComment(e.target.value); // 입력된 댓글 값 업데이트
  };

  const handleEditCommentChange = (e) => { // 수정 중인 댓글 입력 시 상태 업데이트 함수
    setEditComment(e.target.value); // 입력된 댓글 값 업데이트
  };

  const handleKeyPress = (e) => { // 엔터 키 입력 시 호출되는 함수
```

```
<insert id="insertReviews" parameterType="com.example.demo.dto.ReviewsDTO">
  INSERT INTO reviews (id, username, text, gubun)
  VALUES (reviews_seq.NEXTVAL, #{username}, #{text}, #{gubun})
</insert>

<!-- 특정 gubun의 모든 리뷰 가져오기 -->
<select id="getAllReviews" parameterType="int" resultType="com.example.demo.dto.ReviewsDTO">
  SELECT id, username, text, created_at AS createdAt, gubun
  FROM reviews
  WHERE gubun = #{gubun}
  ORDER BY created_at DESC
</select>

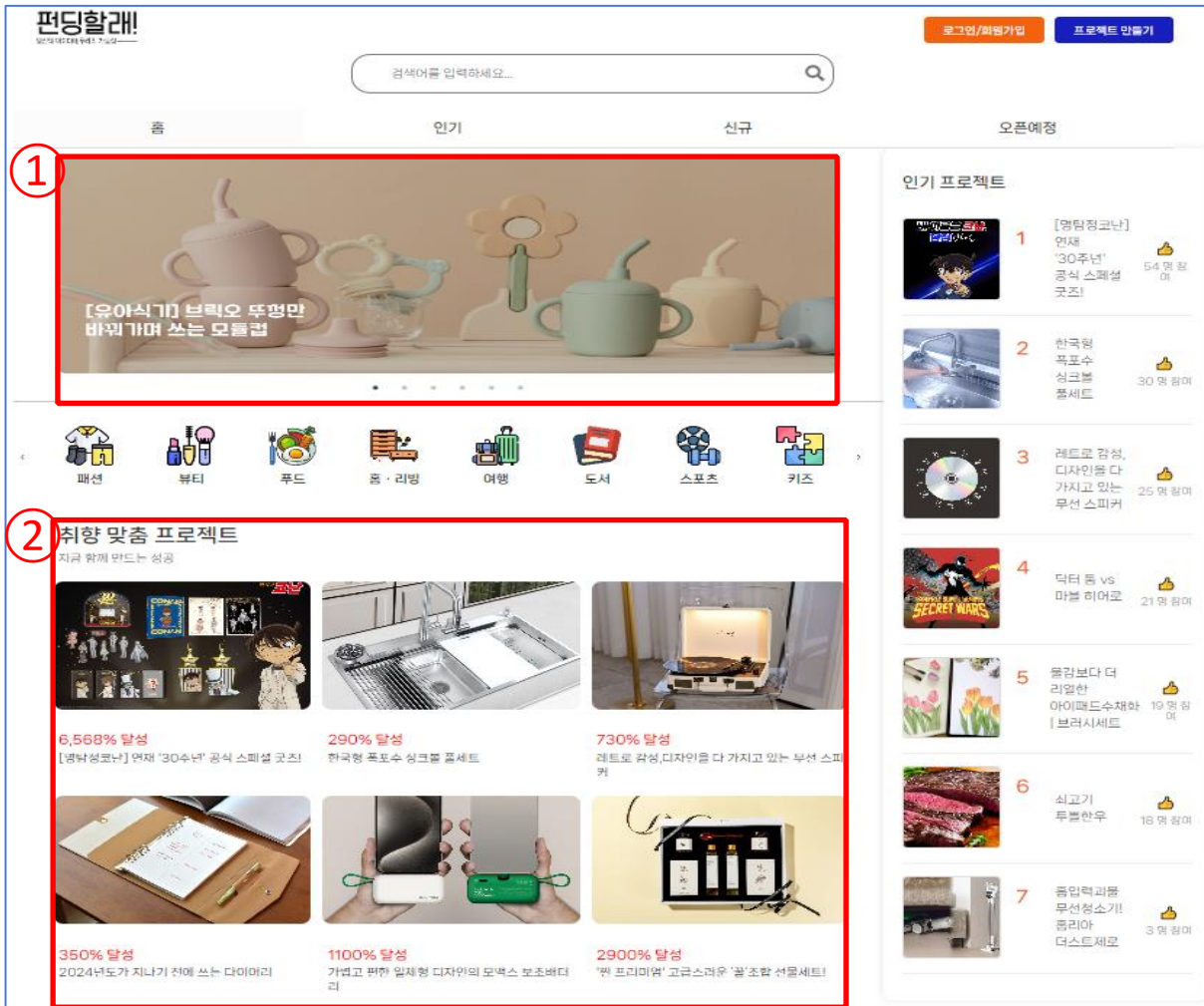
<!-- 특정 id와 gubun에 해당하는 리뷰 가져오기 -->
<select id="getReviewById" parameterType="map" resultType="com.example.demo.dto.ReviewsDTO">
  SELECT id, username, text, created_at AS createdAt, gubun
  FROM reviews
  WHERE id = #{id} AND gubun = #{gubun}
</select>

<!-- 댓글 수정 -->
<update id="updateReview" parameterType="com.example.demo.dto.ReviewsDTO">
  UPDATE reviews
  SET text = #{text}, created_at = CURRENT_TIMESTAMP
  WHERE id = #{id} AND gubun = #{gubun}
</update>

<!-- 댓글 삭제 -->
<delete id="deleteReview" parameterType="map">
  DELETE FROM reviews
  WHERE id = #{id} AND gubun = #{gubun}
</delete>
```


2. 페이지별 구현 기능

메인 홈페이지



장희진

- ① 배너 이미지 자동 슬라이드 기능 및 dot 버튼 클릭 시 이미지 넘어가도록 기능 추가
- ② 웹 사이트를 동적으로 만들기 위해 이미지 호버 효과 사용
- ③ 애니메이션 효과를 통해 웹 사이트의 시각적 효과 Up!

2. 페이지별 구현 기능

메인 홈페이지

스마트 추천 제품

이동자들이 안정한 성공 편당 집합삼



1500% 달성

캠핑리틀을 위한

캠핑장에서 필요한 모든 것 플세트



1300% 달성

아직도 노트북 수납을 고민 하시나요?

클래식 강성의 캔버스 레더 '아이보릭' 노트북 수납 가능



500% 달성

아직도 들고 다니세요?

두 손이 자유로운 슬림 블루투스 넥밴드 선풍기



1500% 달성

아이들과 여행다닐 때 필수템 유모차

유모차 중 가장 가벼운 4kg대의 조경량 휴대용 유모차

기획전



조금 다른 한끼 식사

외식이 부담지 않은 밀키트



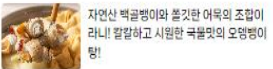
서포터가 주목하는

시선집중 스페셜 기획전



펫 위크

우리 집 댕냥이를 위한



자연산 백골뱅이와 쫄깃한 어묵의 조화가 라니! 깔끔하고 시원한 국물맛의 오랑뱅이 탕!



[밀키트]대구에서 가장 핫한 막창집! 막살가의 시그니처 메뉴를 집에서 즐기세요



[죽칼비] 오로지 국내산 대지갈비로만 막 담았습니디! 국내산 매운 갈비탕!



[까사무피] 핸드메이드 테이블웨어 브랜드 다양한 컬러와 디자인의 실리온 매트



[국내 런칭] 30년 수련 연구 개발 국산 베개, 전 세계 누적 판매 63억을 돌파 브레인 슬립



[무드등] 행랑적인 은하수 감성, 360도 어느 곳에서도 빛나는 오묘한 무드등



[진공미용기] 탈 날림 없이 깨끗하게 스마트 진공 미용기 고양이 강아지 애견바리깅



[자동차정리] 고양이 자동화정리 (WiFi 연결) 더 강력해져서 돌아왔다! 홈 IoT 자동



[방충망] 고양이 추락 탈출 방지 | 방충망 보다 강력한 철벽방이 초미세 흡출망

③ 오늘의 당신의 관심사는?

후원자들의 취향이 궁금했다면 주목!
모두의 시선을 사로잡는 프로젝트!



[단독혜택] 구버스의 먹작 세상에 없던 화판대, 하나로 두가지 감성을!



[아기 있는 집 디퓨저 뽀뽀뽀] 인공장르 0ml, 에센셜 오일 100%



[5분간편설치] 반려견의 안전을 지켜드립니다, 패스비 안전문



[편당할래 첫공개] 아견 어디서 살 수 있어요? 백 번 물어본 플레이트 세트

편당할래가 처음인 당신을 위해 꼭 알아야 할
창작자 가이드와 첫 후원 가이드를 확인해 보세요!

창작자 가이드

첫 후원 가이드

이용약관 | 개인정보처리방침 | 공지사항 | 서비스 소개 | 제휴문의 | 채팅 상담

회사명 | (주)편당할래

주소 | 충청남도 천안시 동남동 258, 1층 (A사원)

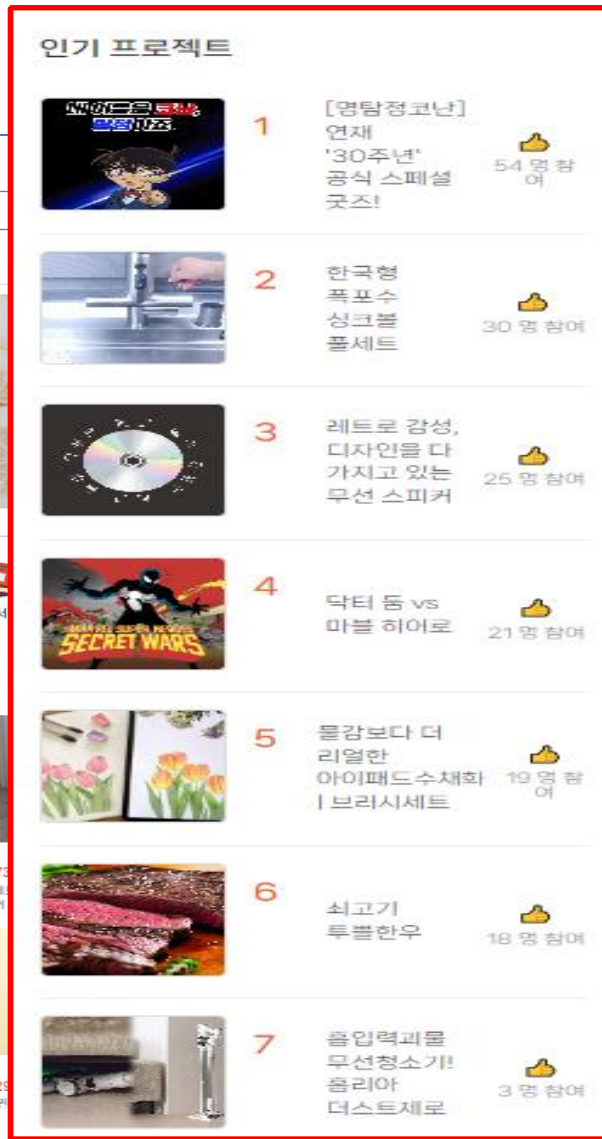
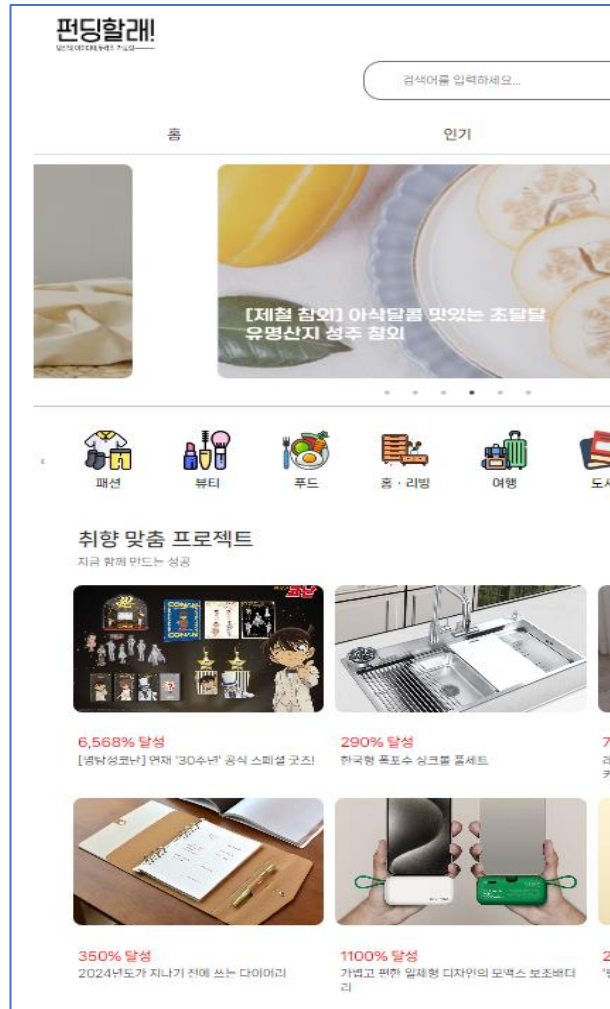
사업자등록번호 123-33-45678

이메일 상담 info@human.kr | 우선 상담 1588-0000 | © 편당할래! co. Ltd

일부 상품의 경우 편당할래는 동산판매중개자이며 동산판매 당사자가 아닙니다. 해당되는 상품의 경우 상품, 상품정보, 거래에 관한 의무와 책임은 판매자에게 있으므로, 각 상품 페이지에서 구체적인 내용을 확인하시기 바랍니다.

2. 페이지별 구현 기능

메인 홈페이지



- ① '좋아요' 버튼 클릭 시 참여자 수가 실시간으로 변동되면서 '인기 프로젝트' 순위 변동 (인기 프로젝트를 데이터베이스에 연동시켜 '좋아요' 수가 실시간으로 저장 되도록 기능 추가)

2. 페이지별 구현 기능

메인 홈페이지

오늘의 당신의 관심사는?

후원자들의 취향이 궁금했다면 주목!
모두의 시선을 사로잡는 프로젝트!

①



[단독혜택] 쿠팡의 역작 세상에 없
하나로 두가지 감성을!

편딩할래!

X

가이드 바로가기

후원자 가이드

창작자 가이드



어디서 살 수 있어요? 백
트



정찬희

① 메인 홈페이지 첫 진입 시

자동 팝업 생성 기능으로 후원자 및 창작자 페이지로
쉽게 이동 가능

② fullpage 라이브러리를 사용하여

React 애플리케이션에 전체 화면 스크롤링 기능 구현

편딩할래가 처음인 당신을 위해 꼭 알아야 할
창작자 가이드와 첫 후원 가이드를 확인해 보세요!

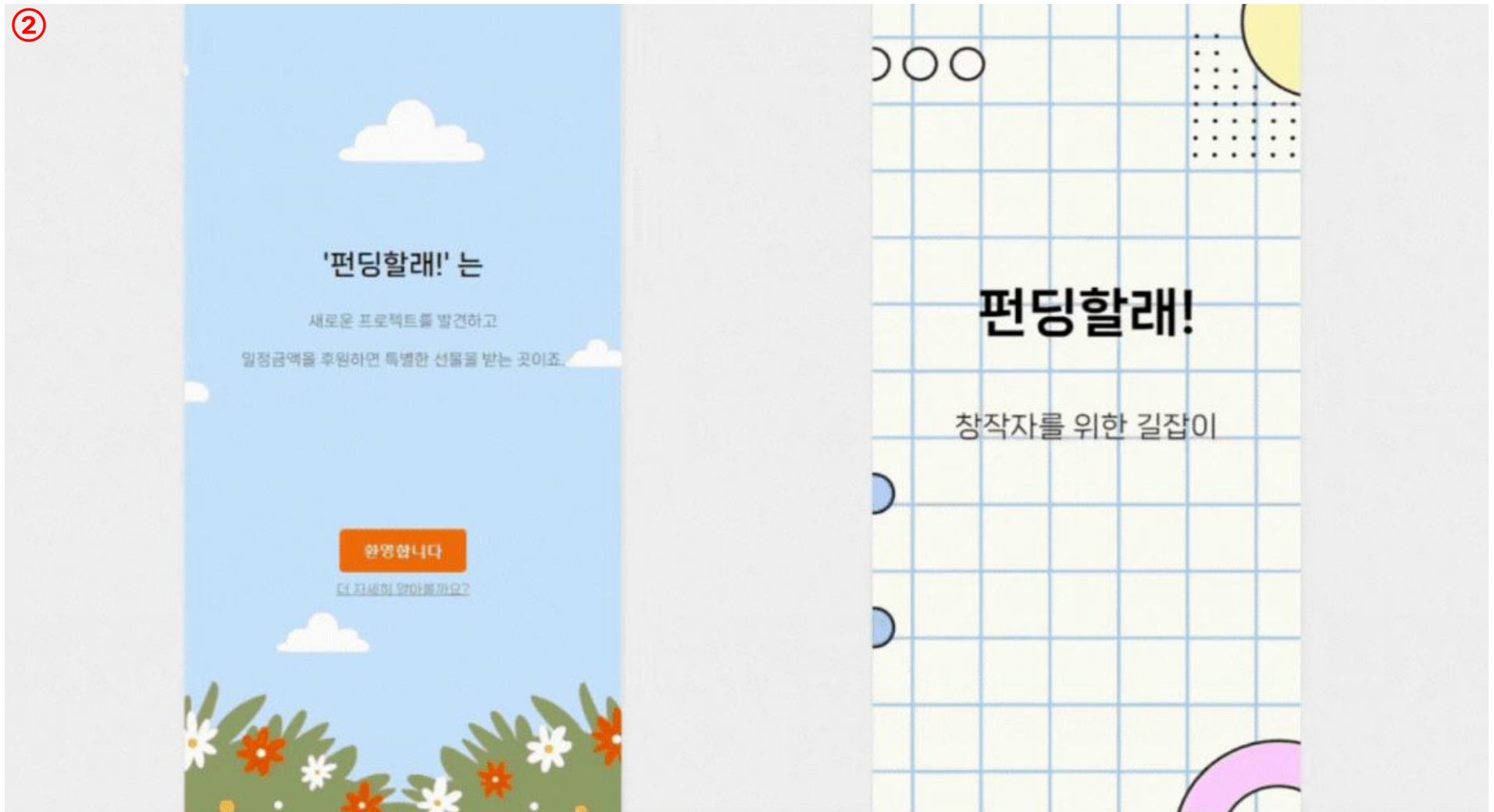
창작자 가이드

첫 후원 가이드

2. 페이지별 구현 기능

첫 후원, 창작자 가이드

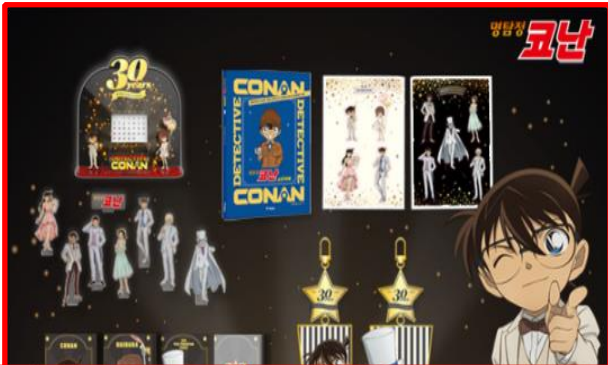
②



2. 페이지별 구현 기능

펀딩 페이지

①



예술 창작자의 첫 번째 팬이 되었습니다!

닫기

검색어를 입력하세요



[명탐정코난] 여대 '30주년' 축하 기념 TV파

창작자에게 문의하기

이름

이메일

문의 내용

문의하기

닫기

추가 기능이 있습니다

②

③

팔로우

창작자 문의

영신, 박예슬

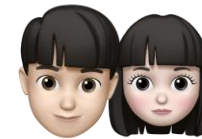
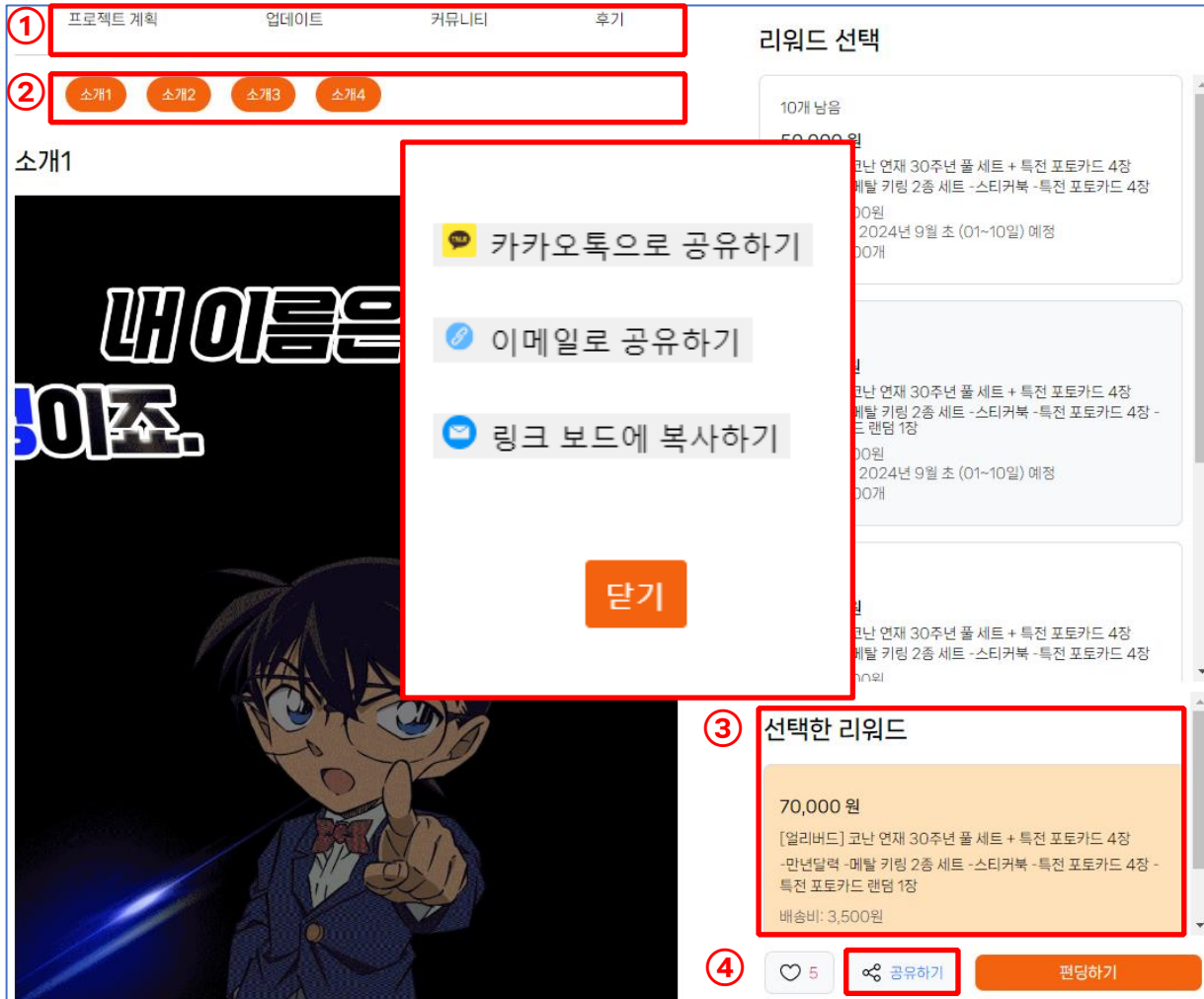
라이더 및 dot 기능 추가

클릭 시 팔로우 완료 팝업 노출

③ 창작자 문의 버튼 클릭 시 <창작자 문의> 팝업
형태로 노출

2. 페이지별 구현 기능

펀딩 페이지



진영신, 박예슬

- ① 각 카테고리 클릭 시 해당 페이지로 이동
- ② 버튼 클릭 시 해당 섹션으로 이동
- ③ 리워드 선택 시, 선택한 리워드 항목으로
추가 후 강조
- ④ '공유하기' 버튼 클릭 시
<이메일 및 링크보드에 복사> 팝업 노출

2. 페이지별 구현 기능

펀딩 페이지

펀딩할래!

달란의 아이디어, 무려의 가능성

로그인/회원가입

프로젝트 만들기

프로젝트 계획

업데이트

커뮤니티

후기

창작자 업데이트

최종 작성 일자: 08-27 09:39

작성하기

①

창작자만 글을 쓸 수 있어요.

안녕하세요 ^^ .코난을 사랑해 주시는 모든 분들께 감사의 말씀 전합니다. 저 역시도 코난과 함께 나이를 들어가고 있어요. 어린 시절 코난이 조금 무서웠던 기억이 있지만, 시간이 지나면서 코난 같은 애니메이션이 또 없었어요! 다들 마음에 품고 있는 어린 시절 애니메이션 하나씩은 있잖아요? 그것만으로 얼마나 삶의 위안이 되는지 몰라요. 여기 게시는 많은 분들도 저랑 같은 마음이겠죠!! 다들 이번 프로젝트로 인해 코난과 함

②

후원자만 글을 쓸 수 있어요.

작성하기

③

U 코난이 벌써 ..30주년이라니 남도일 돌아와~~~~

사용자 2024. 8. 27. 오전 9:36:55

✎

🗑

U 코난 덕후는 여기서 쉬어갑니다!!!!

사용자 2024. 8. 27. 오전 9:36:55

✎

🗑

U 코난 최고예요!

사용자 2024. 8. 27. 오전 9:36:55

✎

🗑



진영신

① 창작자 업데이트란을 통해

창작자가 소식을 업데이트 할 수 있도록

시간 기록 및 게시물 작성 기능 추가

② 해당 프로젝트에 댓글 작성 기능 추가

③ 입력된 댓글 수정 및 삭제 가능

2. 페이지별 구현 기능

펀딩 페이지

펀딩할래!

당신의 아이디어가 세상을 바꾼다

로그인/회원가입

프로젝트 만들기

후기

1

리뷰를 작성하세요.

작성하기

찬

코난 진짜...덕후...코난 굿즈 모은지 N년 차.. 이런 구성에 이런 혜자 가격..헤헤.. 넘 예뻐요..♥달력도 예쁘구,, 포토 카드도 원하는 게 나왔어요,,지루하거나 심심할 때 코난 보면서 시간을 보냈고, 코난 굿즈를 모으기 시작하면서 지갑은 알아지지만 행복감은 배로 되는 것 같아요 😊💕😍 코난굿즈 모으는 재미가 쏠쏠해요 퀄리티 있는 제품이라 더 만족스러워요♥

찬화 2024. 8. 26. 오전 11:50:29

2

유용해요

창의적이에요

특별해요

퀄리티가 좋아요

의미 있어요

소통이 잘 돼요

3

4

✏️

🗑️

👍

👎

예

너무 좋아요~😊😍😍😍코난은 저의 어린 시절 추억 그 자체인데.. 구성 전부 좋네요.기다린 보람이 있는 퀄리티예요! 좋은 프로젝트 감사합니다♥다음에는 더 알찬 구성으로 크라우드 펀딩 부탁드려요!!!!

예슬 2024. 8. 26. 오전 11:42:24

유용해요

창의적이에요

특별해요

퀄리티가 좋아요

의미 있어요

소통이 잘 돼요

✏️

🗑️

👍

👎

희

드디어 너를 만났구나!!!!!!!!!!!!!!!설레는 마음이 가라앉지 않아서 이제서야 후기 씁니다ㅠㅠ♥♥♥가격이 좀 비싸긴 했지만 너무 예쁘네요 😊 내 인생 최고의 애니메이션 코난,, 올해 마지막 달 최고의 선물이에요 🥰 감사합니다 🥰완전 이빠요..! 이걸 아까워서 어떻게 쓰나...🥰♥달력 걸어놓고 한참을 바라봤네요..

희진 2024. 8. 24. 오후 4:58:14

유용해요

창의적이에요

특별해요

퀄리티가 좋아요

의미 있어요

소통이 잘 돼요

✏️

🗑️

👍

👎

영

말해줘해.....♥💕 안샀으면 당장 후회했을것.....포토카드와 키링까지..늦었지만 넘 잘 받았어요 코난 굿즈 다 예뻐더라고요! 랜덤 포토카드도 너무 갖고 싶던 게 나와서 진짜 넘 행복♥얼리버드 가격으로 구매할 수 있어서 대만족이에요 짱짱

영신 2024. 8. 24. 오후 4:23:18

유용해요

창의적이에요

특별해요

퀄리티가 좋아요

의미 있어요

소통이 잘 돼요

✏️

🗑️

👍

👎

박예슬

① 해당 리워드 후기 작성 기능

② TAG 기능을 통해 후기 가독성 향상 및 긍정적 리뷰 유도

③ 수정/삭제 아이콘 클릭 시 팝업 상태에서 수정/삭제 가능

④ 좋아요/싫어요 버튼 클릭 시 애니메이션 효과

2. 페이지별 구현 기능

회원가입

편당할래!
일인당 100원 이상 구매 시 할인

이메일 간편가입

①

이메일 계정

인증하기

유효한 이메일을 입력해 주세요.

②

비밀번호 입력

비밀번호는 8글자 이상, 대소문자, 숫자, 특수문자를 포함해야 합니다.

비밀번호 확인

☐ 약관에 동의합니다

③

간편가입 완료하기

편당할래!
일인당 100원 이상 구매 시 할인

이메일 간편가입

javachip@test.com

인증하기

javachip

.....

비밀번호는 8글자 이상, 대소문자, 숫자, 특수문자를 포함해야 합니다.
비밀번호 강도: 약함

....|

비밀번호가 일치하지 않습니다.

☐ 약관에 동의합니다

간편가입 완료하기



전영준

- ① 이메일 계정 '@'가 포함 되지 않으면 “유효한 이메일을 입력해 주세요.” 에러메세지 노출
가입 되어진 이메일은 “이미 사용 중인 이메일입니다.” 팝업 노출
- ② 비밀번호는 “8글자 이상 대소문자, 숫자, 특수 문자를 포함” 하도록 메시지 노출
- ③ 조건을 모두 만족하면
<간편가입 완료하기> 버튼이
활성화 되어 회원가입 완료

2. 페이지별 구현 기능

로그인

The image shows a screenshot of the Kakao login page in a Chrome browser window. The browser's address bar shows the URL: `accounts.kakao.com/login/?continue=https%3A%2F%2Fkauth.ka...`. The page has the Kakao logo at the top. Below it, there are input fields for '아이디' (ID) and '비밀번호' (Password). The ID field contains the text 'yesuleeee'. There are two callout boxes with red circles and numbers: ① points to the ID field with the text '이메일 주소에 '@'를 포함' (Include '@' in the email address), and ② points to a yellow button with a speech bubble icon and the text '카카오 로그인' (Kakao Login). Below the password field, there is a checkbox for '간편로그인 정보 저장' (Save convenient login information) and a yellow '로그인' (Login) button. Below the login button, there is a link '또는' (Or) and a grey button for 'QR코드 로그인' (QR code login). At the bottom, there are links for '회원가입' (Sign up), '계정 찾기' (Find account), and '비밀번호 찾기' (Find password). The footer contains links for '한국어' (Korean), '이용약관' (Terms of service), '개인정보 처리방침' (Privacy policy), and '고객센터' (Customer service), along with the copyright notice '© Kakao Corp.'.



전영준

① 아이디 입력란에 '@'가 포함되지 않으면 “유효한 이메일을 입력해 주세요.”

에러 메시지 노출

② 소셜 로그인 API를 통하여 사용자에게 편리하고

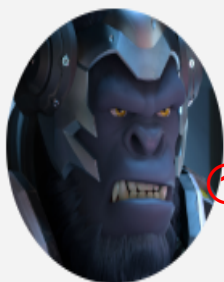
다양한 로그인 방식 제공,

이메일 및 SNS 회원가입을

데이터베이스 상에 구분 하여 회원 정보 관리 가능

2. 페이지별 구현 기능

마이 페이지



자바칩
2024-08-27 가입

①

수정하기

메인 페이지로

내 정보 수정

커뮤니티

후기

②

자기소개를 입력하세요

작성완료

안녕하세요. JAVACHIP입니다! 😊 저희의 첫 번째 프로젝트 '편딩할래!' 많은 관심과 응원 부탁드립니다. 모든 팀원이 처음 배우는 프로그래밍이었고, 첫 프로젝트였지만 서로 부족한 점을 보완해주기 위해 노력했습니다. 직접 배워가며 작은 실무를 경험할 수 있어 정말 감사한게 많은 프로젝트였습니다. 🥰
👍 앞으로도 더 좋은 모습 많이 보여드리는 자바칩이 되겠습니다. 🥰



진영신

① 수정하기 버튼 클릭 시 이름 수정 가능

② 자기소개 입력 시, 하단 자기소개 출력

ERD 다이어그램 설계

순위 RANKING

아이디	R_ID	번호	NUMBER
제목	R_TITLE	제목	VARCHAR2(300)
참여자 수	R_LIKE	참여자 수	NUMBER(*,0)
이미지	R_IMG	이미지	VARCHAR2(300)

창작자 업데이트 CREATOR_UPDATES

아이디	ID	번호	NUMBER
사용자 이름	USERNAME	사용자 이름	VARCHAR2(255)
본문	TEXT	본문	CLOB
작성시간	CREATED_AT	시간	TIMESTAMP (6)
업데이트	UPDATED_AT	시간	TIMESTAMP (6)
구분	GUBUN	구분	NUMBER(*,0)
회원 번호	MEMBER_ID	번호	NUMBER
회원 번호	MEMBER_ID2	번호	NUMBER

회원 MEMBERSHIP

회원 번호	MEMBER_ID	번호	NUMBER
이메일 계정	EMAIL	이메일 계정	VARCHAR2(100)
비밀번호	PASSWORD	비밀번호	VARCHAR2(20)
회원 이름	NAME	이름	VARCHAR2(30)
가입일	SIGNUP_DATE	가입일	DATE
마지막 업데이트	LAST_UPDATE	마지막 업데이트	DATE
회원상태	STATUS	상태	CHAR(1)
회원등급	GRADE	등급	NUMBER(1,0)

댓글 COMMENTS

아이디	ID	번호	NUMBER
이름	USERNAME	이름	VARCHAR2(255)
본문	TEXT	본문	CLOB
구분	GUBUN	구분	NUMBER(*,0)
작성시간	CREATED_AT	시간	TIMESTAMP (6)
회원 번호	MEMBER_ID	번호	NUMBER
회원 번호	MEMBER_ID2	번호	NUMBER

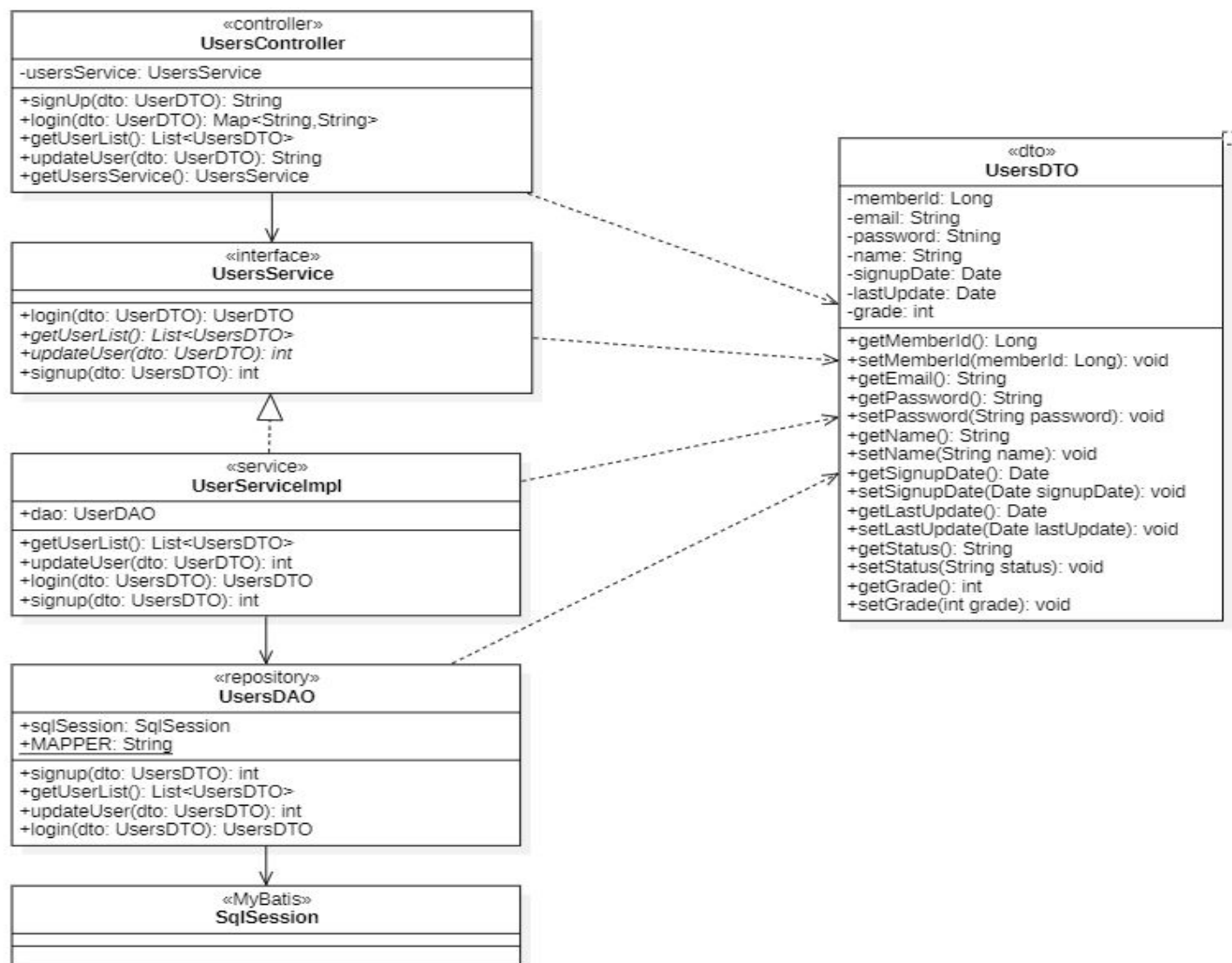
SNS회원 SNS_MEMBERSHIP

회원 번호	MEMBER_ID	번호	NUMBER
로그인 제공자	PROVIDER	제공자	VARCHAR2(20)
이메일	EMAIL	이메일	VARCHAR2(100)
SNS 아이디	ID	아이디	VARCHAR2(50)
회원 이름	NAME	회원 이름	VARCHAR2(30)
가입일	SIGNUP_DATE	가입일	DATE
마지막 업데이트	LAST_UPDATE	마지막 업데이트	DATE
회원 상태	STATUS	상태	CHAR(1)
회원 등급	GRADE	등급	NUMBER(1,0)

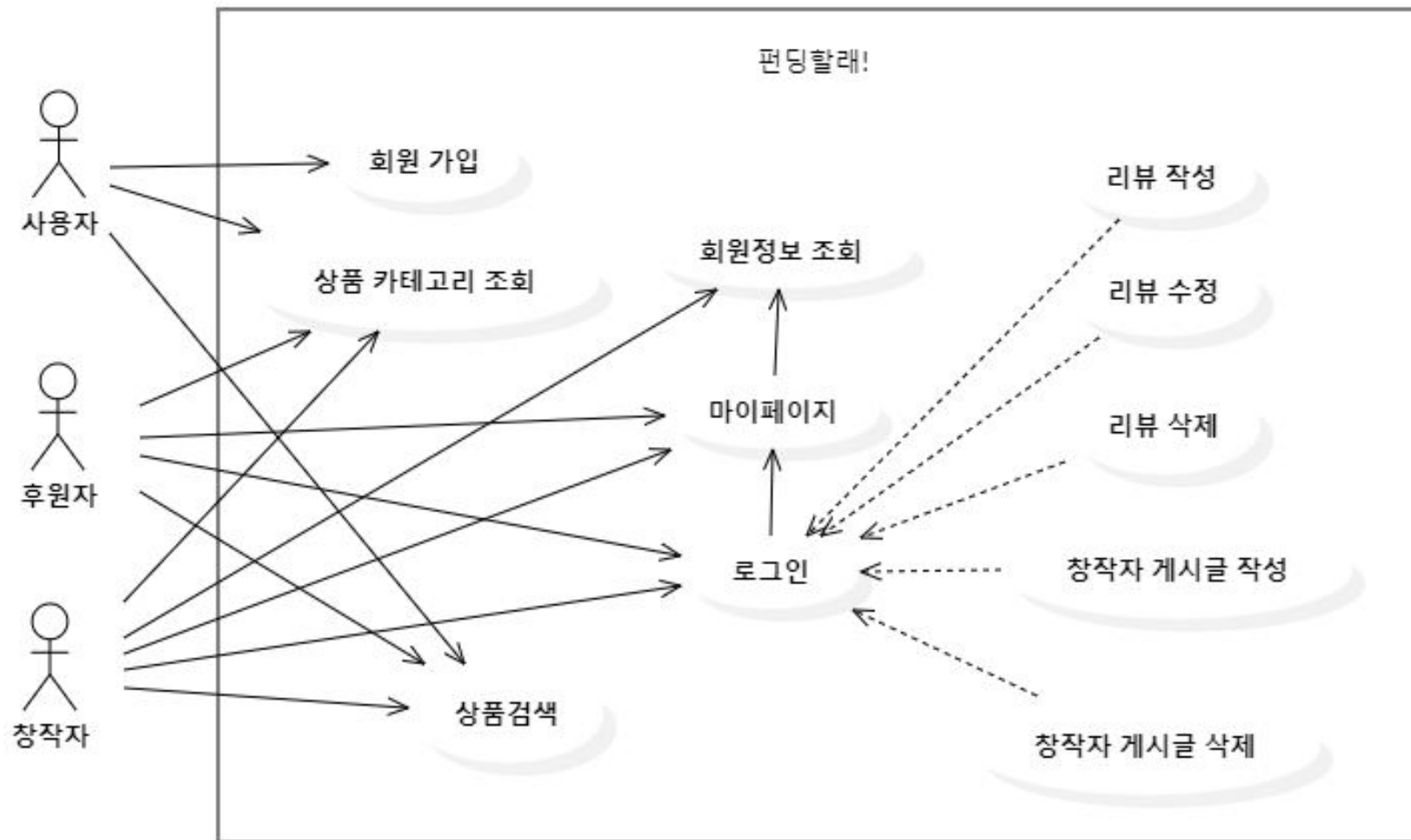
후기 REVIEWS

아이디	ID	번호	NUMBER
사용자 이름	USERNAME	사용자 이름	VARCHAR2(300)
본문	TEXT	본문	CLOB
작성시간	CREATED_AT	시간	TIMESTAMP (6)
구분	GUBUN	구분	NUMBER
회원 번호	MEMBER_ID	번호	NUMBER
회원 번호	MEMBER_ID2	번호	NUMBER

Class 다이어그램 설계



UseCase 다이어그램 설계



Project Review

- 1) 경험했던 문제점과 해결방안
- 2) 프로젝트 후기
- 3) 팀원 평가
- 4) Q&A
- 5) 마무리

1. 경험했던 문제점과 해결방안



진영신

문제점1	<u>화면을 스크롤해서 내릴 때 리워드 선택 부분이 고정되지 않고 사라지는 문제점</u>
해결방안1	리워드 선택 div에 <u>position: sticky</u> 를 주고 <u>top: 0;</u> 으로 고정으로 해결
문제점2	창작자 업데이트 부분의 <u>최종 작성 일자가 불러오지 않는 문제점</u>
해결방안2	call 함수에 데이터를 JSON 객체로 반환하게 했기 때문에 백엔드의 Controller 부분에서 JSON 형식으로 반환되기 위해 기존에 단순 문자열로 반환했던 <u>String 부분을 Map<String, String>으로 수정하여 JSON 형태인 키-값 형태로 수정 후 해결</u>

1. 경험했던 문제점과 해결방안



전영준

문제점1	리액트에서 네이버 API을 호출하여 회원정보를 요청하는 과정에서 CORS 문제가 발생 <u>CORS 문제로 인해 리액트에서 네이버API로부터 회원정보를 받아 올 수 없는 문제점</u>
해결방안1	스프링부트 서버에 프록시 컨트롤러를 구현하여 리액트에서 스프링부트 서버에 요청을 보내면, <u>스프링부트 서버가 네이버API와 통신하여 필요한 데이터를 받아 온 후 리액트로 반환하게 하여 해결</u>
문제점2	회원가입 페이지에서 <u>모든 입력란을 작성하면 form의 너비가 줄어드는 문제점</u>
해결방안2	form 태그에 className을 join으로 설정하여 <u>CSS에서 클래스 선택자를 활용해 form에 너비값을 설정하여 form이 container에 영향을 받지 않도록 해결</u>

1. 경험했던 문제점과 해결방안



장희진

문제점1	배너의 컨테이너의 <u>사이즈에 맞게 배너의 이미지가 적용이 안되는 문제점</u>
해결방안1	배너의 컨테이너의 <u>css의 position: relative; width: 100%; height: 300px; max-width: 1000px;의 속성의 값들을 변경을 한 후 이미지 조정으로 해결</u>
문제점2	‘오늘의 당신의 관심사는?’ 부분의 <u>애니메이션 효과가 반복으로 적용이 안되는 문제점</u>
해결방안2	오늘의 당신의 관심사의 JS부분의 <u>text.classList.add('text-visible');</u> <u>text.classList.remove('text-visible');</u> 코드를 추가 하여 해결

1. 경험했던 문제점과 해결방안



정찬희

문제점1	<u>랭킹 리스트 불러오기에서 백엔드에서 설정이 잘 되었지만, 프론트에서 불러오지 못하는 문제점</u>
해결방안1	<u>VSCode에서 console.log를 추가 하고 API 호출할 때 에러가 발생한 응답 데이터를 콘솔에 출력하여 데이터가 제대로 오고 있는지 확인하고 순차적으로 디버깅 하여 해결</u>
문제점2	랭킹 리스트를 프론트에서 <u>호출은 성공했지만 null 값으로 호출하는 문제점</u>
해결방안2	<u>console.log를 추가 하고 오류를 확인하였지만, 오류가 발견되지 않아 DBeaver에 테이블을 다시 한 번 확인, Eclipse 설정이 잘 되어 있는지 확인, 마지막으로 VSCode에서 랭킹 리스트를 불러오는 코드에서 랭킹 테이블에 이름이 대소문자 차이로 못 불러오는 것을 확인하여 랭킹 리스트를 디버깅 하여 해결</u>

1. 경험했던 문제점과 해결방안



박예슬

문제점1	Review에서 태그 선택 시 <u>일괄적으로 모든 후기에 태그가 선택되는 문제점</u>
해결방안1	loadReviews함수에서 setReviews(response) 대신 <u>'setReviews(reviewsWithTags)'로 수정 후 handleTagClick에서 리뷰 별로 태그 상태를 관리하도록 review.id 태그를 모두 전달</u> <u>'review.selectedTags.includes(tag)'로 수정 후 태그 컴포넌트에서 selectedTags를 개별 리뷰로 전달 후 해결</u>
문제점2	Review에서 <u>수정 및 삭제가 제대로 작동하지 않는 문제점</u>
해결방안2	ReviewDAO에서 중복되는 메서드를 지우고 Mapper에서 누락된 ID값을 수정, ReviewServiceImpl 클래스의 생성자에서 오탈자 수정, <u>중복으로 인해 ReviewServiceImpl 클래스가 ReviewService 인터페이스에 상속된</u> <u>'getAllReviews(int)' 메소드가 정의되지 않아 새로 구현 후 해결</u>

협업 시 커뮤니케이션의 중요성



진영신

의사소통과 협업이 정말 중요하다는 걸 깨닫게 해준 프로젝트였습니다.

혼자서 하기 보다 같이 하면 더욱더 효율적이라는 걸 느꼈고, 의사 소통을 많이 하면서 프로젝트의 방향을 잘 전달하고 같이 소통하면서 진행하니까 제가 부족한 부분은 믿고 맡기게 되고 잘 하는 부분에 집중할 수 있었습니다.

처음 하는 프로젝트인 만큼 잘 해내고 싶었고, 열심히 한 것 같지만 처음 구상했던 것보다 완성도가 떨어진 거 같아서 아쉽기도 합니다.

1차 프로젝트를 진행하면서 배우고 느낀 것들을 잘 활용해서 2차 때는 더욱더 완성도 있게 프로젝트를 마무리하고 싶습니다!

2. 프로젝트 후기

낮섭에서 흥미진진 (순항 중인 돛단배)



박예슬

시작은 어려웠으나, 할수록 흥미를 느끼고 성취감을 찾을 수 있었던 첫 프로젝트였습니다.

처음 해보는 프로그래밍, 낮섭이 가득하고 어렵다고 느꼈던 수업 시간이었지만

그럼에도 꼭 잘 해내고 싶었습니다.

팀원들에게는 계속 함께 하고 싶은 팀원이 되고 싶었고,

스스로에게는 부끄럽지 않게 끝까지 최선을 다해서 유종의 미를 거두고 싶었습니다.

프로젝트 발표를 앞두고 시간을 되돌아 보았을 때 뿌듯함과 아쉬움 그럼에도 오랜만에 열심히 무언가를

해봤다는 성취감이 앞으로 2, 3번째 프로젝트의 큰 동기부여가 될 것만 같습니다!

2. 프로젝트 후기

서로의 도움으로 만족스러운 결과를



장희진

메인 페이지를 만들때 참고를 하기로 한 편딩 페이지들을 참고를 해서 일단 피그마를 통해서 디자인만 구현을 해보고 만들어 보기로 하였지만 저의 생각 만큼 css를 추가하고 위치를 잡는게 쉽지가 않았습니다.

기능들을 추가하고 애니메이션도 추가를 하는데 제가 구현을 하고 싶은 기능들이 있지만 쉽지가 않았고, 하지만 점점 기능들을 추가 하면서 자신감도 생기고 욕심도 생겼지만 저의 마음처럼 쉽지는 않고, 페이지를 다 완성을 하고 다 합치다 보니 혼자 했을때와는 다른 문제 점들도 생기고 하였지만 팀원들과의 소통을 하면서 의견을 말을 하고 협업도 중요하게 된다는 것을 느꼈습니다.

또한, 첫 프로젝트를 팀 프로젝트로 시작을 하였고, 팀원들과 소통을 하면서 서로서로 챙겨주고 도움을 주는 팀원들 덕분에 더 잘하고 싶다는 마음이 생기고 다 함께 열심히 첫 프로젝트를 잘 마무리를 했다고 생각을 합니다.

2. 프로젝트 후기

문제 원인 파악과 창의적이면서 실용적인 해결책을 찾아라!



전영준

이번 프로젝트에서 가장 크게 배운 것은 기술적 문제를 해결하는 과정에서의 체계적인 접근법이라고 생각합니다. CORS 문제와 같이 예상치 못한 문제가 발생했을 때, 문제의 원인을 정확하게 파악하고, 다양한 해결 방법을 모색하는 과정이 중요하다는 것을 느꼈습니다 특히 프록시 컨트롤러를 통해 백엔드 서버를 경유하게 함으로써 문제를 해결한 경험은 기술적 도전에 직면했을 때 창의적이면서 실용적인 해결책을 찾아야 한다는 점을 다시금 깨닫게 해주었습니다.

2. 프로젝트 후기

끈기 있고, 책임감 있게



정찬희

이번 프로젝트를 통해 처음으로 제대로 된 코딩을 접해보았습니다. 처음에는 프론트엔드 작업만 하려고 했지만, 랭킹 리스트 부분을 담당하게 되면서 백엔드 작업도 경험하게 되었습니다.

프론트엔드는 디자인 작업에서 창의성과 아이디어가 매우 중요하다는 것을 깨달았고, 백엔드에서는 초기 설정과 API 호출 부분이 핵심이라는 점을 배웠습니다.

코딩 과정에서 디버깅이 쉬운 일은 아니지만, 충분한 시간을 들이면 해결할 수 있다는 생각이 들었습니다.

3. 팀원평가

진영신	저희 팀 팀장으로 정말로 잘 알려주시고 열심히 해주셔서 감사합니다!
	조장으로서의 역할을 완벽하게 수행했습니다. 팀원들의 모든 문제들을 함께 고쳐주고 같이 바로 잡아 나가며, 팀의 밸런스를 잘 잡아주는 꼭 필요한 사람이었습니다. 다음에도 함께 하길 바라며 너무 고생 많으셨습니다.
	진영신 조장은 팀원들의 의견을 존중하면서 프로젝트의 중요한 부분을 이끌었습니다. 특히, 다양한 기능을 아우르는 페이지들을 성공적으로 구현하고 팀의 목표를 달성하는 데 큰 기여를 했습니다.
	팀장으로서 팀원들을 잘 격려해주시고, 전체적인 조율을 훌륭하게 해주셔서 감사합니다. 덕분에 프로젝트가 원활하게 진행될 수 있었습니다.
정찬희	프로젝트의 전반적인 사용자 흐름을 설계하고 구현했으며 인기 프로젝트 기능을 통해 사용자의 관심을 끌수 있는 요소를 성공적으로 구현했습니다.
	열심히 하려고 하는 모습이 돋보입니다. 잘 하지 못하는 부분이어도 굴하지 않고 꼭 해내는 모습이 인상 깊었습니다.
	항상 성실하게 프로젝트에 임하고 맡은 부분은 어떻게든 해내는 모습이 멋있었어요! 팝업 같은 부분도 신경 쓸게 많았을텐데 깔끔하게 잘 해준 것 같아요!
	묵묵히 잘 해주셔서 감사합니다.

3. 팀원평가

박예슬	펀딩 페이지, 리워드 페이지, 후기 페이지와 같은 중요한 기능을 담당하면서 문서관리와 회의 기록을 총괄하며, 프로젝트가 체계적으로 진행될 수 있도록 큰 기여를 했습니다.
	PPT 자료 잘 정리해 주시고 모르는거 하나하나 친절하게 알려주셔서 감사합니다.
	프론트면 프론트 백엔드면 백엔드 척척 잘 하는 거 보고 너무 멋있었어요!! PPT도 예쁘게 잘 만들어 주시고 못하는 게 없는 거 같아요! 같은 팀이여서 너무 든든했습니다. 다음에도 같이해주세요!
	항상 열심히 해주시고, 팀원들을 도와주셔서 정말 감사드립니다. 덕분에 팀워크가 더욱 단단해졌습니다.
장희진	희진 팀원은 항상 일찍와서 자기가 맡은 역할을 완수해 나갔고, 어렵고 모르는 부분은 친구들이나 선생님께 물어보면서 부족한 부분을 채워나갔습니다.
	메인 페이지가 가장 꾸밀 것도 많고 배치나 구성 등 체크해야 할 것들이 많은데 깔끔하고 세련되게 잘 만들어줘서 고마워요!
	팀원들과 원활한 협업과 적극적인 피드백을 반영하여 깔끔한 디자인을 통해 프로젝트의 완성도를 높였습니다.
	메인 페이지 디자인 작업을 맡아주셔서 정말 고생 많으셨습니다. 덕분에 프로젝트의 완성도가 높아졌습니다.

3. 팀원평가

전영준

로그인이랑 회원가입 페이지를 혼자서 완벽하게 완성 시키고 잘 해주셔서 감사합니다.

영준이는 자기가 맡은 부분을 최선을 다하고, 엄청난 집중력을 보여주는 팀원이었고 갈수록 든든하고 성장하는 모습이 보이는 팀원이었습니다.

로그인 회원가입 부분 중요한데 혼자 맡아서 하느라 고생했습니다.
척척 잘 해내고 API까지 구현한거 보고 멋있었습니다! 영준님한테 배우러 갈게요!

로그인 API 기능이 쉽지 않았을텐데, 잘 해결해 주셔서 감사합니다.

Q&A

Made in JavaChip

**Thank you for
Listening! :)**