

TEAM 그릴그린

CorpCollector

201801628_오혜진

201801825_천세륜

201801825_오규진

목차

01

팀 구조

02

프로젝트 설명

03

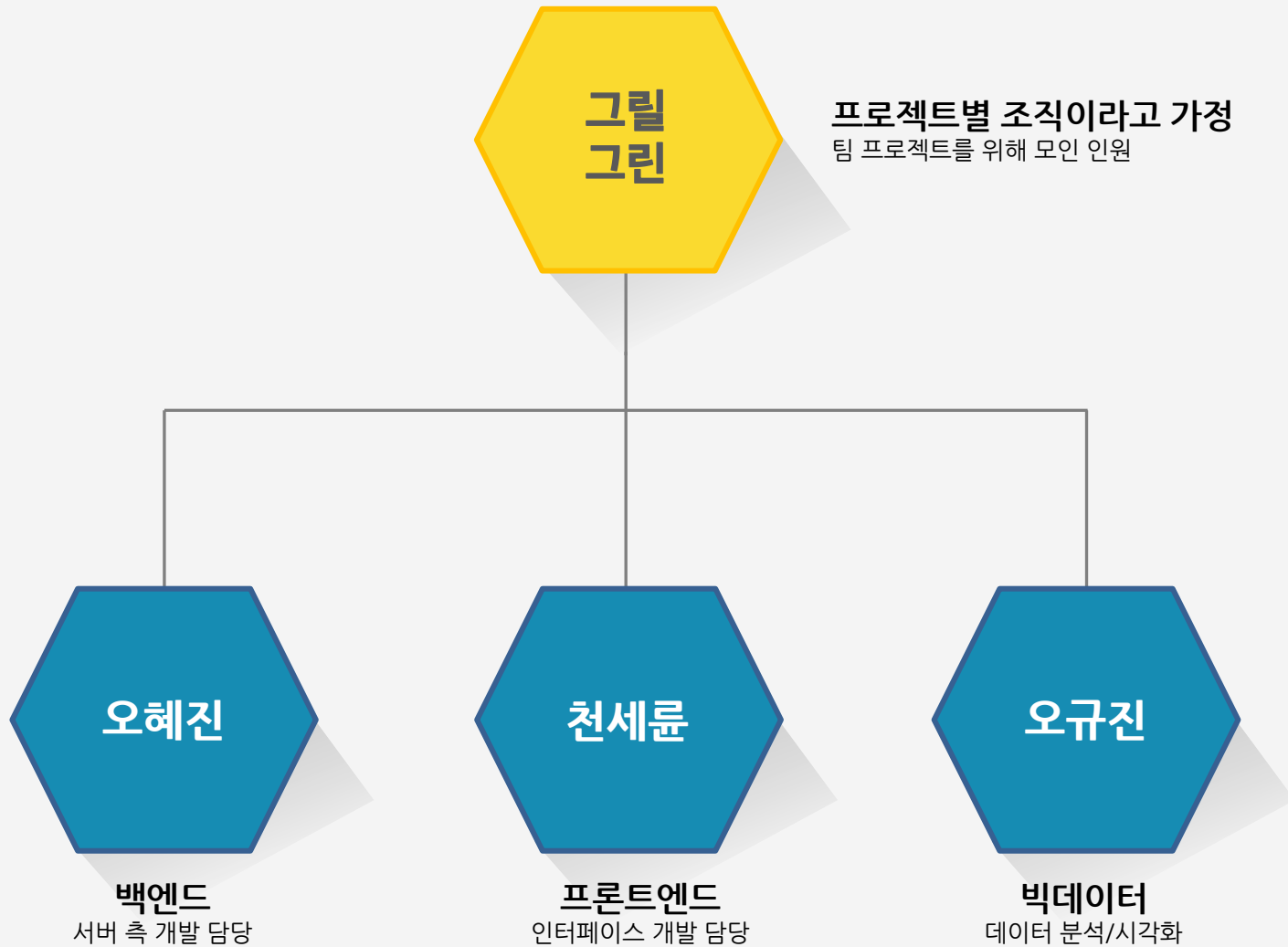
산출물 소개

04

현재 진행 상황

01. 팀 구조

팀 구조



02.프로젝트 설명

CorpCollector란?
프로젝트 기획 배경
사용한 HW, SW 목록
개발방법론

CorpCollector란?



소비자 입장에서 보는 **선한 기업** 정보를 제공하는 웹

1. 동적 웹 프로젝트로 구성해서 접근성을 높인다.
2. 기업정보 관련 공공데이터 파일/API를 이용하되, 사용자가 쉽게 정보를 이해하고 파악할 수 있도록 UI를 구성한다.
3. 단순 기업 리스트만이 아닌 데이터 분석을 통해 얻을 수 있는 자료의 질을 높인다.

프로젝트 기획 배경



▲ 논란이 있는 기업 제품 대신 다른 기업의 대체재를 소비한 결과

<https://www.jobplanet.co.kr/contents/news-994>

사용한 HW, SW 목록

1

하드웨어

개인용 개발 PC 3대

2

소프트웨어 - 개발 툴

DBMS(MySQL), Eclipse IDE, Excel, Github, Colab, Visual Studio Code, Tableau

3

WAS

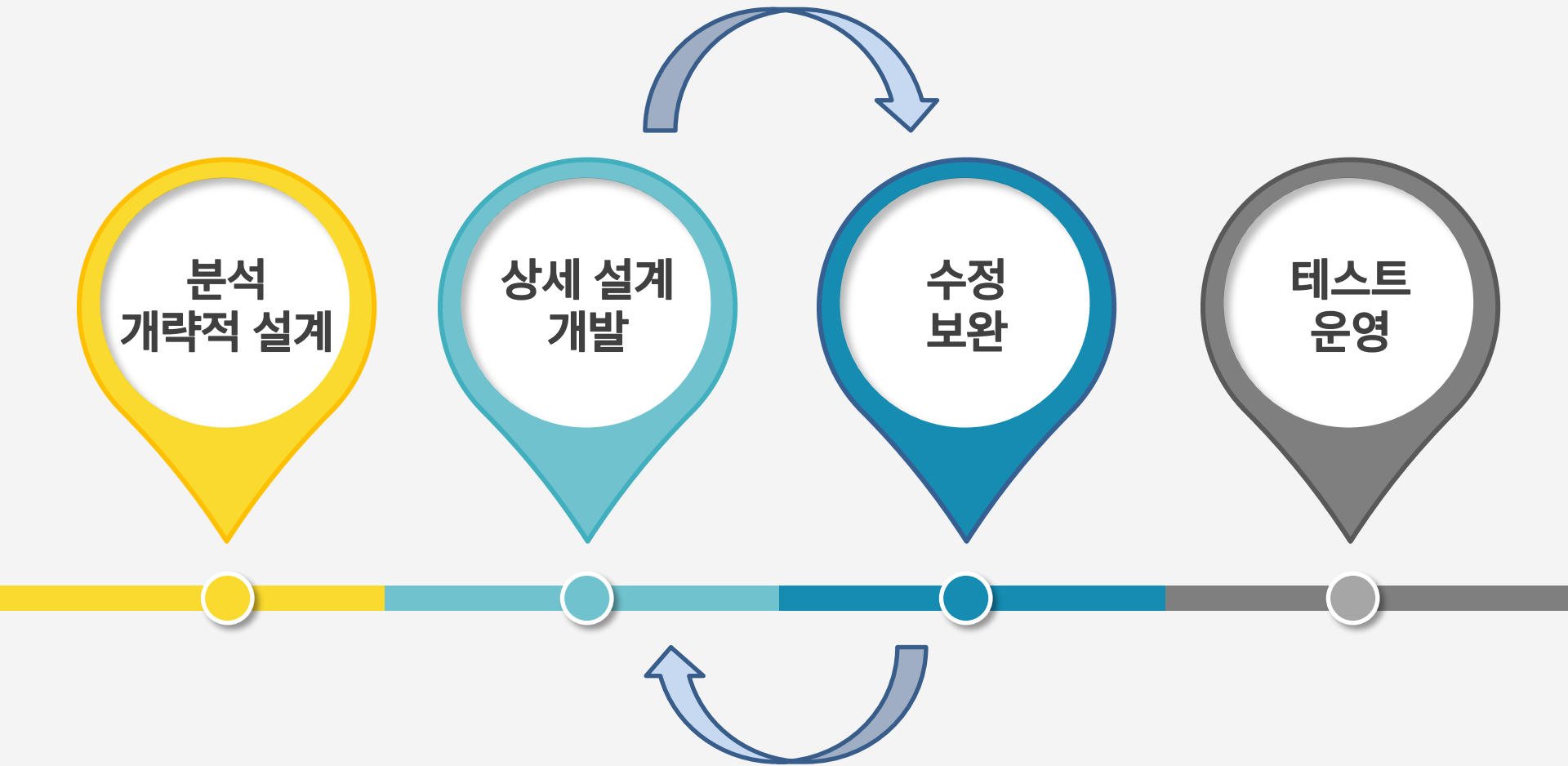
아파치 톰캣

4

클라우드 컴퓨팅 서비스

아마존 웹 서비스(AWS)

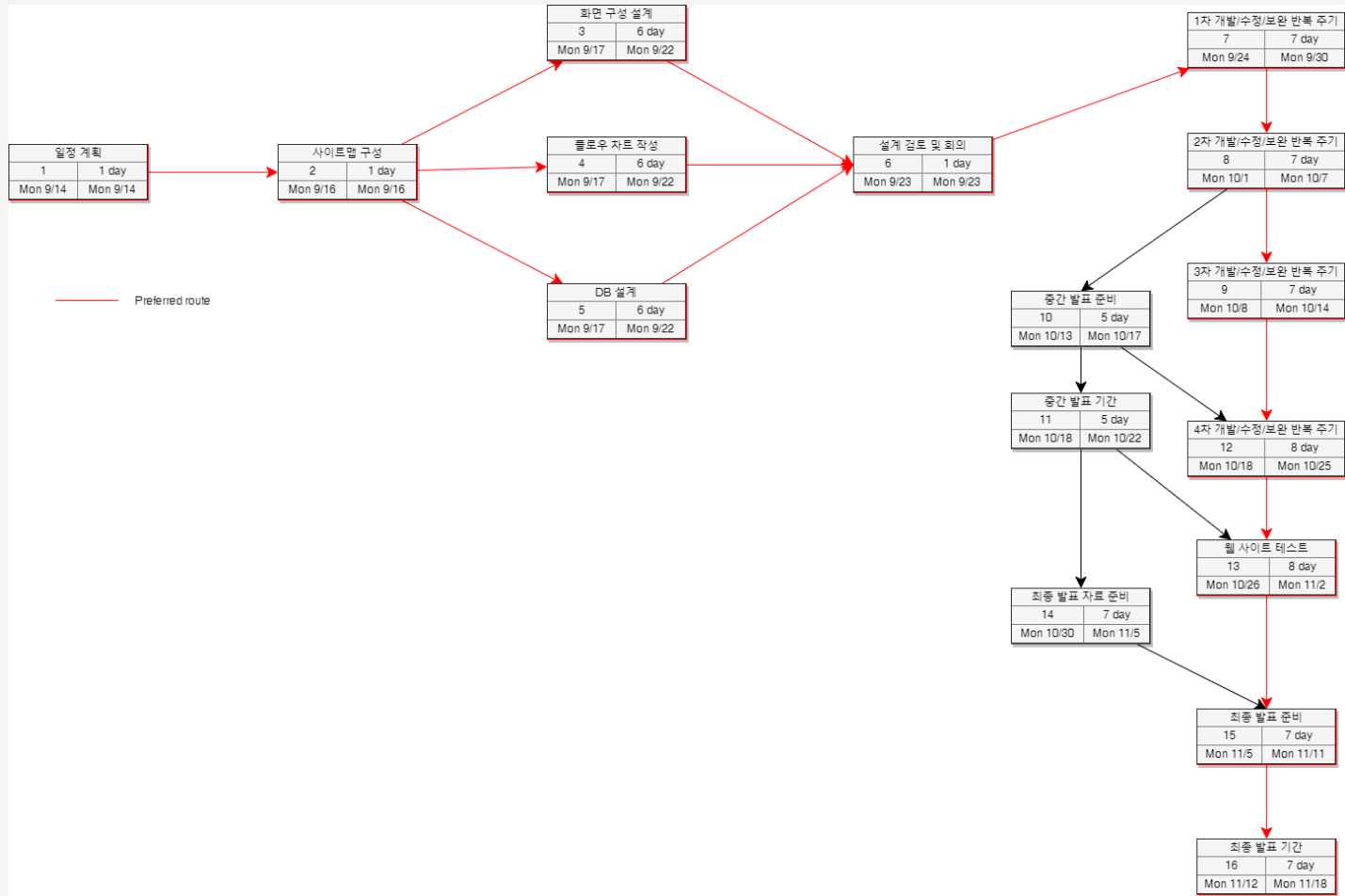
개발방법론 - RAD



03.산출물 공개

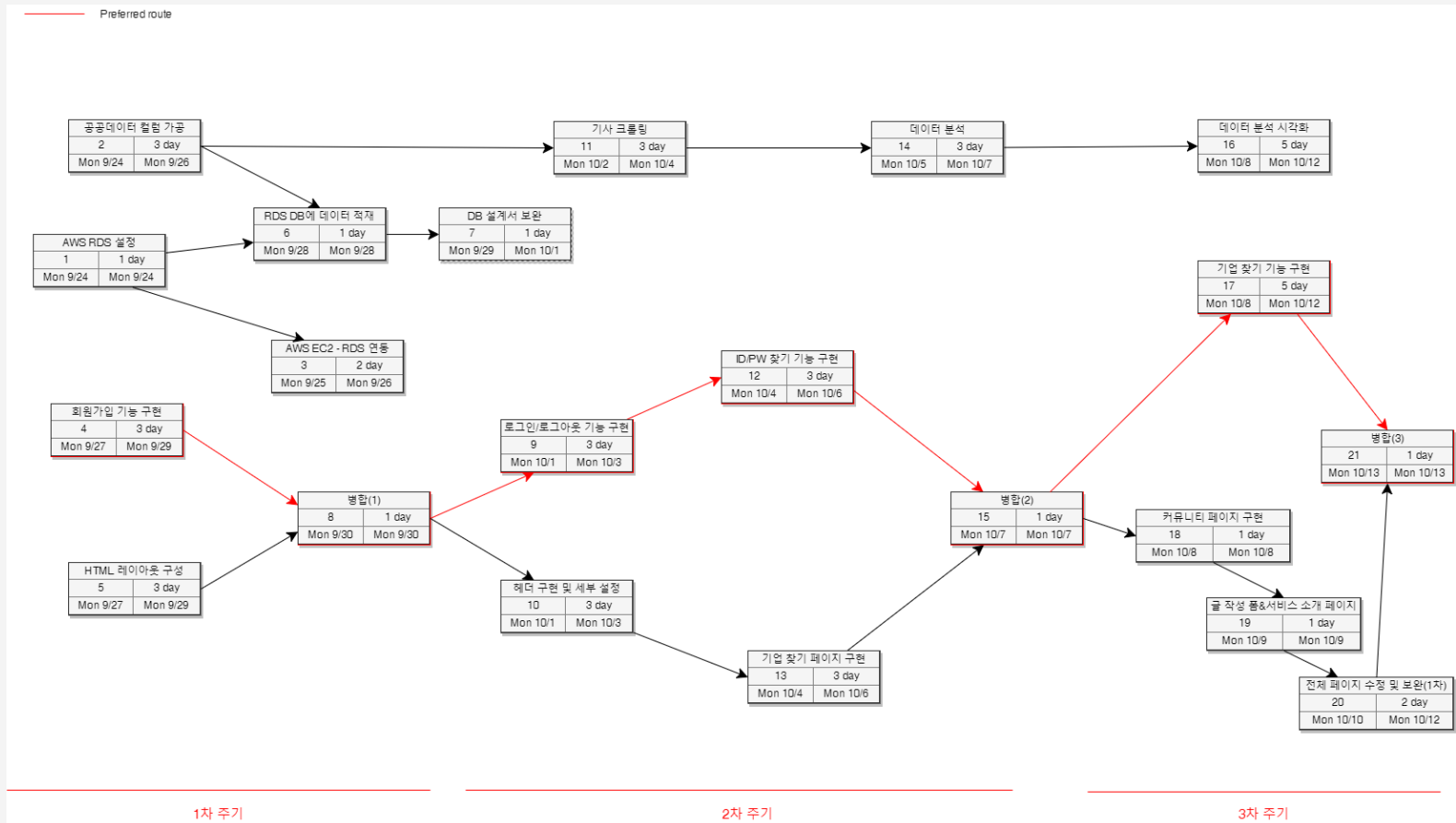
퍼트 차트/간트 차트
사이트맵/기능(요구사항) 정리
비용 산정
리스크 도출
화면 구성 설계서
프로세스별 플로우 차트
데이터베이스 설계서

파트 차트



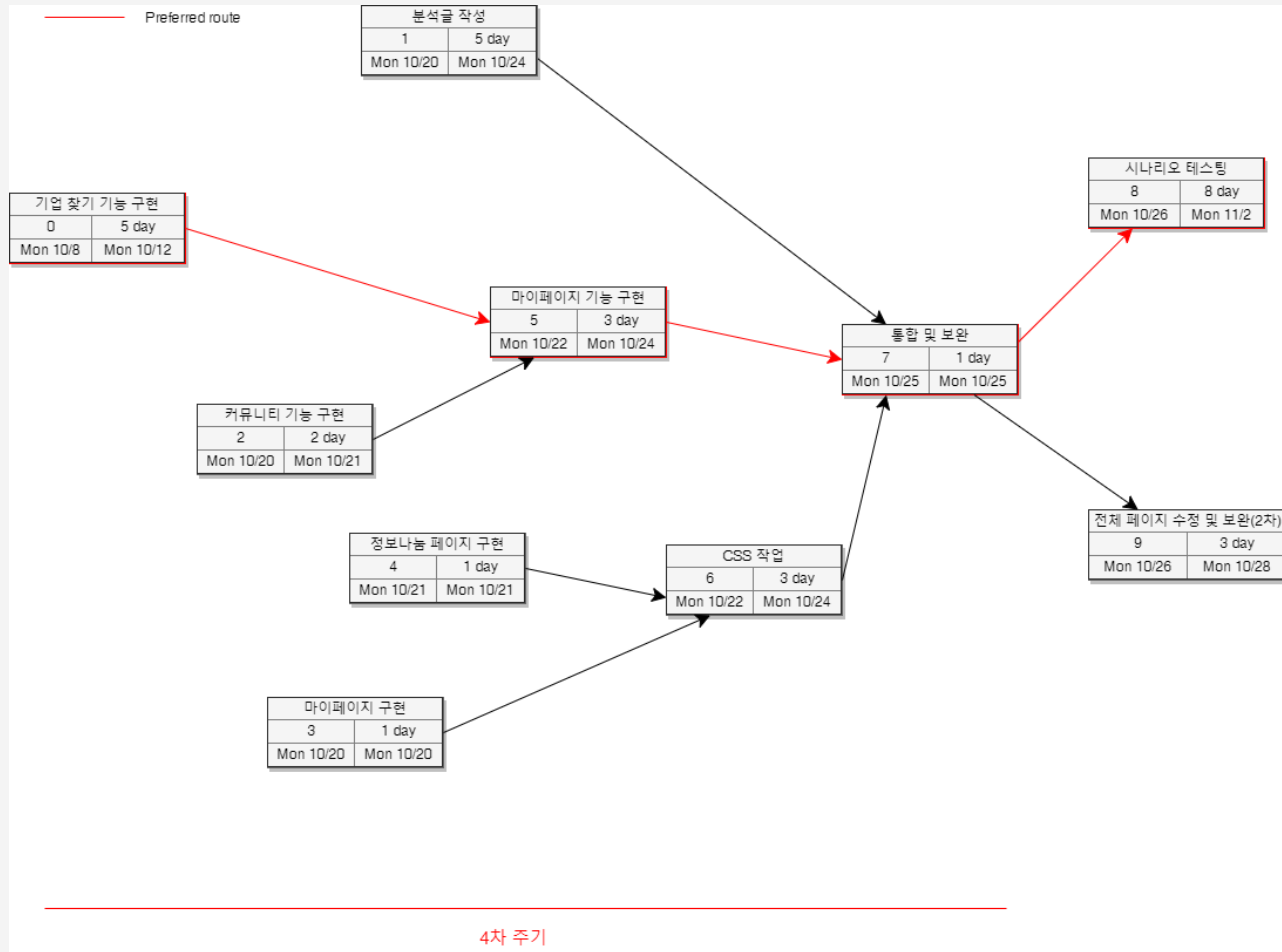
▲ 전체 파트 차트

퍼트 차트



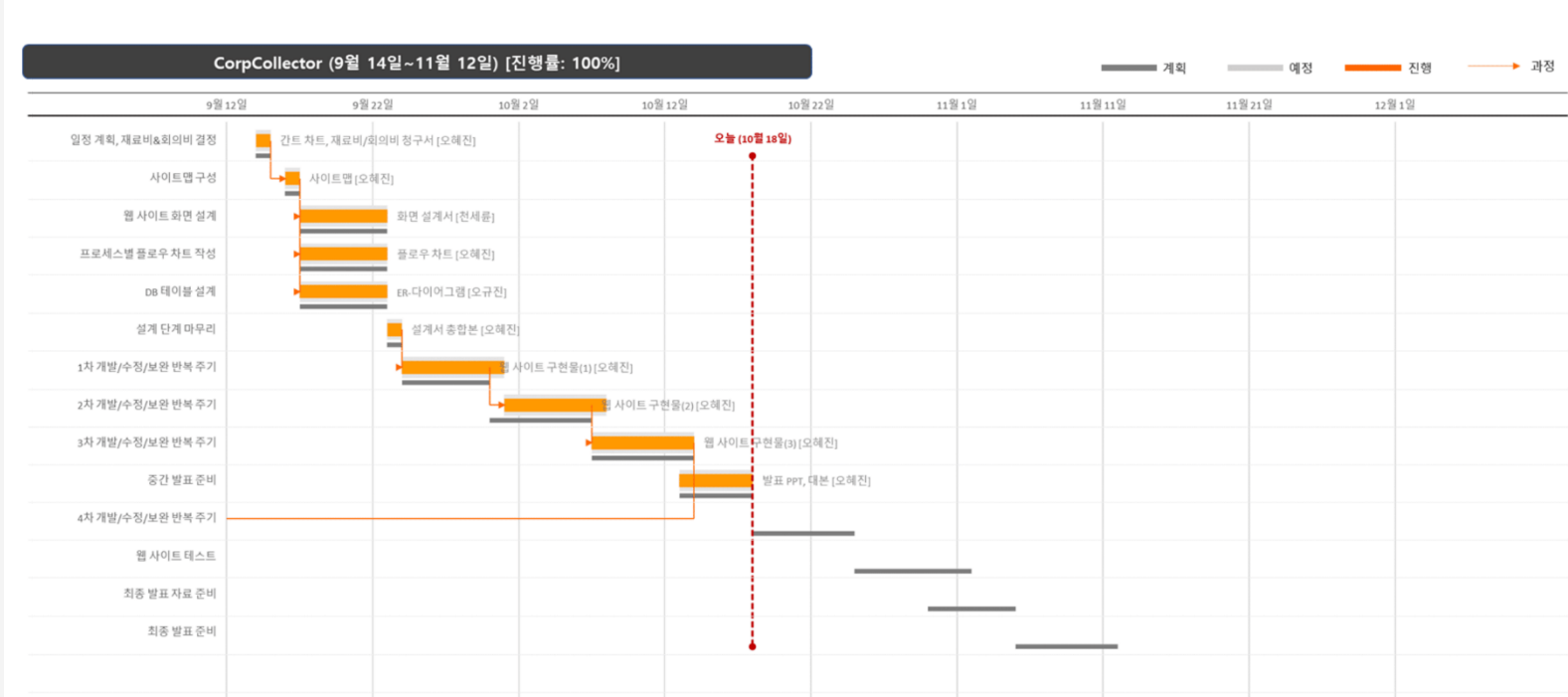
▲ 1~3차 주기 상세 퍼트 차트

퍼트 차트



▲ 4차 주기 상세 퍼트 차트

간트 차트



▲ 간트 차트(간략)



사이트맵



▲ 사이트맵 구조

기능(요구사항) 정리

1

기업 찾기

기업 리스트 조회 및 검색 기능

2

정보나눔

기업 관련 공공데이터를 바탕으로 데이터 분석 자료를 첨부,
기업 유형에 따른 기업 기사 링크 모음 제공

3

커뮤니티

공지사항 게시판, 고객후기 게시판 기능

4

마이페이지

개인정보 관리, 최근 검색 기업과 관심 기업 확인 가능,
고객후기 게시판에 남긴 내가 쓴 글 관리 가능

비용 산정

기능점수(FP) 방법

1. SW사업 대가산정 가이드(2021년) 참고했다.
2. 복잡도는 모두 낮음으로 통일한다.
3. 내부 논리 파일(ILF)은 화면 또는 보고서 출력을 위한 추출파일은 제외
>> 공공데이터 및 기업 관련 기사 데이터, 최근검색기업 데이터는 제외한다.
4. 외부 연계 파일(ELF), 외부 출력(EO)은 없다고 판단했다.

비용 산정 - 기능점수(FP) 방법

내부논리파일	가중치
회원 데이터	7
공지사항 데이터	
고객후기 데이터	
관심기업 데이터	

외부입력	가중치
회원가입	3
회원탈퇴	
개인정보 수정	
공지사항 등록	
공지사항 수정	
공지사항 삭제	
고객후기 등록	
고객후기 수정	
고객후기 삭제	
관심기업 등록	
관심기업 삭제	

외부조회	가중치
로그인	3
관리자 인증	
기업 리스트 조회	
기업 기사 리스트 조회	
관심기업 리스트 조회	
최근 검색 기업 리스트 조회	
공지사항 리스트 조회	
고객후기 리스트 조회	
내가 쓴 글 목록조회	

비용 산정 - 기능점수(FP) 방법

전통법 기능점수 산정

기능유형	가중치(낮음)	합계
내부논리파일	4×7	28
외부연계파일	0	0
외부입력	11×3	33
외부출력	0	0
외부조회	9×3	27
총 기능점수		88

비용 산정 - 기능점수(FP) 방법

보정계수 결정

1. 2021년 기준 기능점수당 단가: 553,114원
2. 보정전 개발원가 = $88 \times 553,114 = 48,674,032$ 원
3. 기능 점수가 500FP 미만이므로 **규모 보정계수 = 1.28**
4. 그 외 애플리케이션 복잡도 보정계수는 모두 제일 낮은 수준으로 측정

보정요인	난이도 수준	계수
연계복잡성 수준	타기관 연계 없음	0.88
성능 요구수준	특별한 요구사항 없음	0.91
운영환경 호환성	요구사항 없음	0.94
보안성 수준	보안 요구사항 1개 이상	0.97

비용 산정 - 기능점수(FP) 방법

보정후 개발원가

총 기능 점수	기능 점수당 단 가	보정계수					금액(원)
		규모	연계 복잡성	성능	운영 환경 호환성	보안성	
88 FP	553,114	1.2800	0.88	0.95	0.94	0.97	47,491,202 (원)

리스크 도출

리스크 내용 및 관리 방법

1. 개발 그룹 성숙도가 낮다고 판단 → 사후 처리 위주 방법 사용
2. 모든 리스크 관리 방법에는 지도교수님의 조언이 포함되어 있다.

번호	리스크 내용	리스크 관리 방법
1	기술적 취약	프로토타이핑 → 수정, 관리
2	실시간 성능 문제	시뮬레이션 → 조정
3	잘못된 SW 기능 개발	프로토타이핑 → 수정
4	지속적 요구사항 변경	최대 변경상한선 관리

리스크 도출

기술적 취약					
발생 가능성	PMhw	PMsw	PChw	PCsw	PD
	0.3	0.3	0.1	0.3	0.1
가중치	0.2	0.4	0.1	0.2	0.1
심각성	기술	비용	일정		
	0.5	0.1	0.3		
가중치	0.4	0.2	0.4		

$$P = (0.3 \times 0.2) + (0.3 \times 0.4) + (0.1 \times 0.1) + (0.3 \times 0.2) + (0.1 \times 0.1) = 0.26$$

$$C = (0.5 \times 0.4) + (0.1 \times 0.2) + (0.3 \times 0.4) = 0.34$$

$$\text{Risk} = P + C - (P \times C) = 0.5116$$

리스크 도출

실시간 성능 문제					
발생 가능성	PMhw	PMsw	PChw	PCsw	PD
	0.3	0.5	0.3	0.1	0.1
가중치	0.2	0.3	0.1	0.4	0.1
심각성	기술	비용	일정		
	0.3	0.1	0.1		
가중치	0.3	0.4	0.3		

$$P = (0.3 \times 0.2) + (0.5 \times 0.3) + (0.3 \times 0.1) + (0.1 \times 0.4) + (0.1 \times 0.1) = 0.29$$

$$C = (0.3 \times 0.3) + (0.1 \times 0.4) + (0.1 \times 0.3) = 0.16$$

$$\text{Risk} = P + C - (P \times C) = 0.4036$$

리스크 도출

잘못된 SW 기능 개발					
발생 가능성	PMhw	PMsw	PChw	PCsw	PD
	0.1	0.5	0.1	0.2	0.1
가중치	0.2	0.4	0.1	0.2	0.1
심각성	기술	비용	일정		
	0.3	0.1	0.5		
가중치	0.4	0.2	0.4		

$$P = (0.1 \times 0.2) + (0.5 \times 0.4) + (0.1 \times 0.1) + (0.2 \times 0.2) + (0.1 \times 0.1) = 0.28$$

$$C = (0.3 \times 0.3) + (0.1 \times 0.4) + (0.1 \times 0.3) = 0.34$$

$$\text{Risk} = P + C - (P \times C) = 0.5248$$

리스크 도출

지속적 요구사항 변경					
발생 가능성	PMhw	PMsw	PChw	PCsw	PD
	0.1	0.3	0.1	0.3	0.1
가중치	0.2	0.5	0.1	0.1	0.1
심각성	기술	비용	일정		
	0.3	0.1	0.3		
가중치	0.3	0.1	0.6		

$$P = (0.1 \times 0.2) + (0.3 \times 0.5) + (0.1 \times 0.1) + (0.3 \times 0.1) + (0.1 \times 0.1) = 0.22$$

$$C = (0.3 \times 0.3) + (0.1 \times 0.1) + (0.3 \times 0.6) = 0.28$$

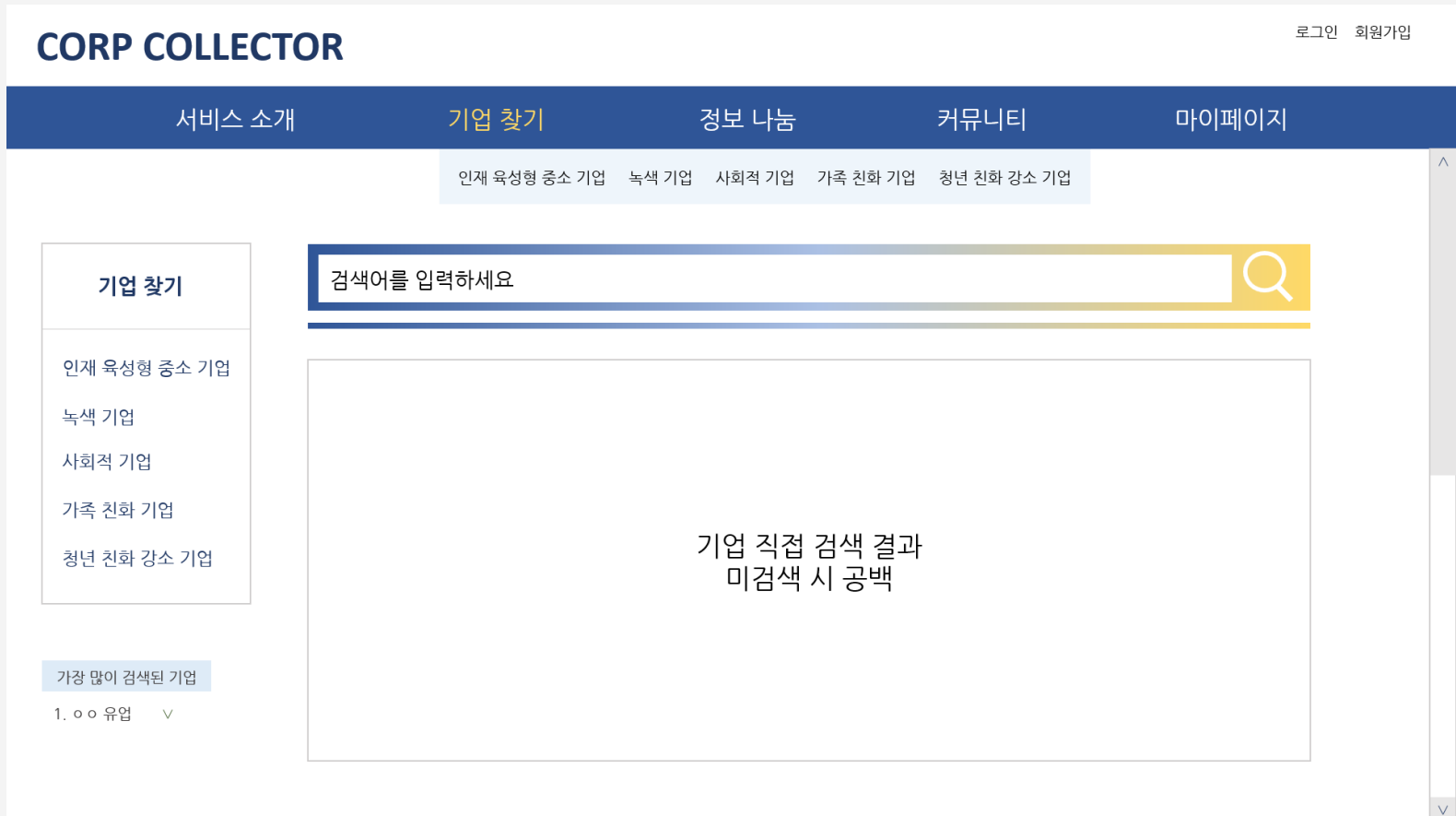
$$\text{Risk} = P + C - (P \times C) = 0.4384$$

화면 구성 설계 - 인덱스 페이지



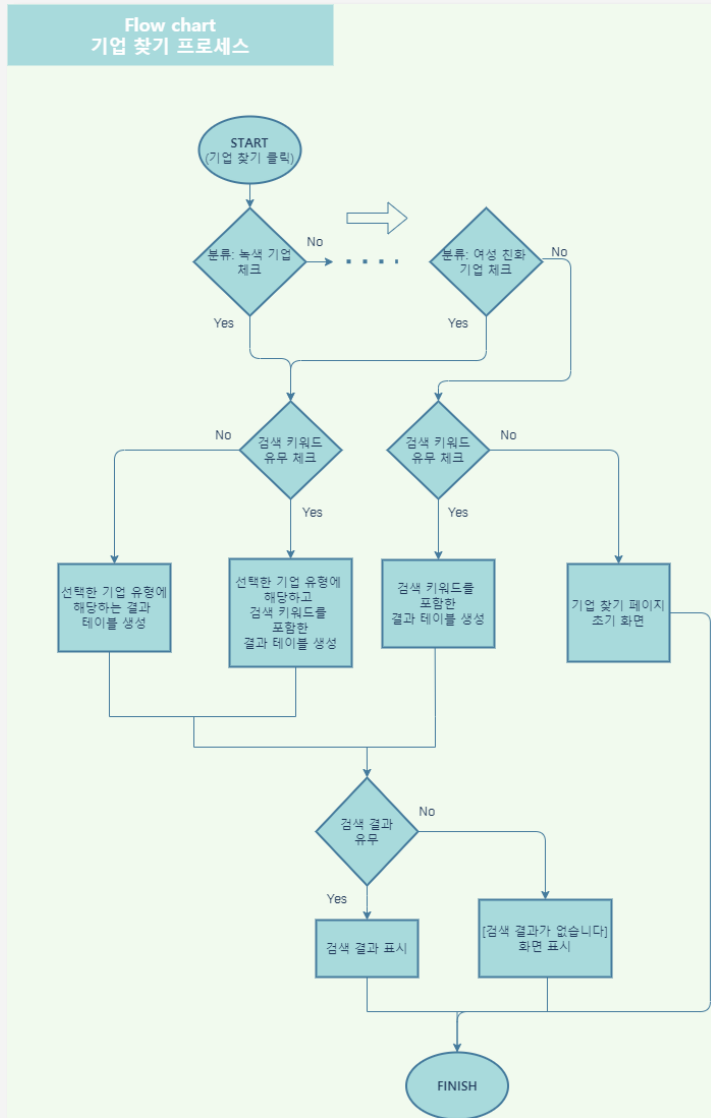
▲ 인덱스 페이지 구성

화면 구성 설계 - 기업 찾기 페이지



▲ 기업 찾기 페이지 구성

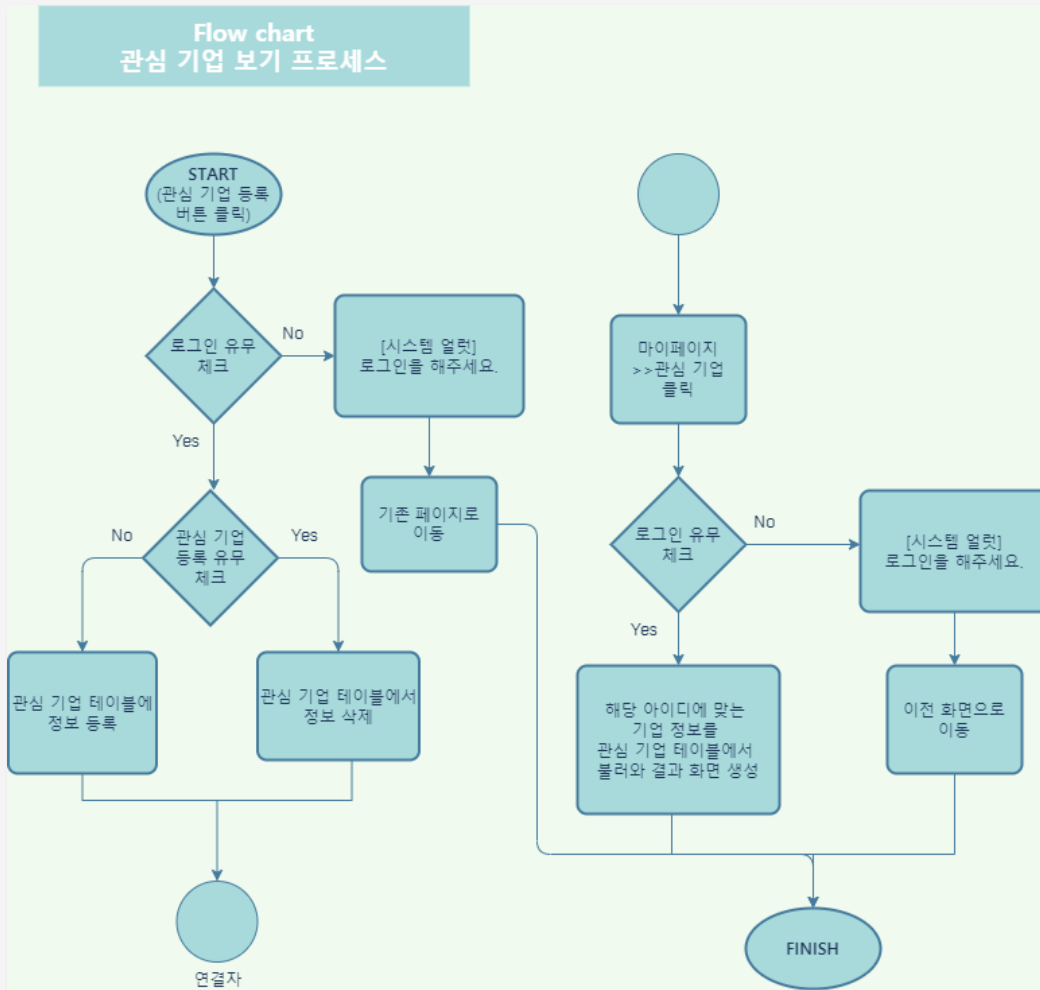
플로우 차트 - 기업 찾기 프로세스



기업 찾기 흐름

1. 기업 유형 선택 여부를 확인한다.
2. 검색 키워드 유무를 확인한다.
3. 선택한 기업 유형에 따라 해당하는 결과 테이블을 생성하고, 검색 키워드와 기업 유형이 정해지지 않았으면 기업 찾기 메인 페이지로 이동한다.
4. 검색 결과 유무를 판단하여 결과가 없다면 [검색 결과가 없습니다.] 메시지를 표시한다.

플로우 차트 - 관심 기업 보기 프로세스



관심 기업 보기 흐름

1. 로그인 여부를 확인하고, 로그인을 하지 않았다면 알럿 메시지를 표시한다.
2. 기업 찾기 페이지에서 관심 기업 등록 버튼을 클릭해서 관심기업 테이블에 추가 또는 삭제할 수 있다.
3. 마이페이지 관심기업 카테고리를 클릭하면 해당 아이디에 맞는 기업 정보를 불러와 리스트를 보여준다.

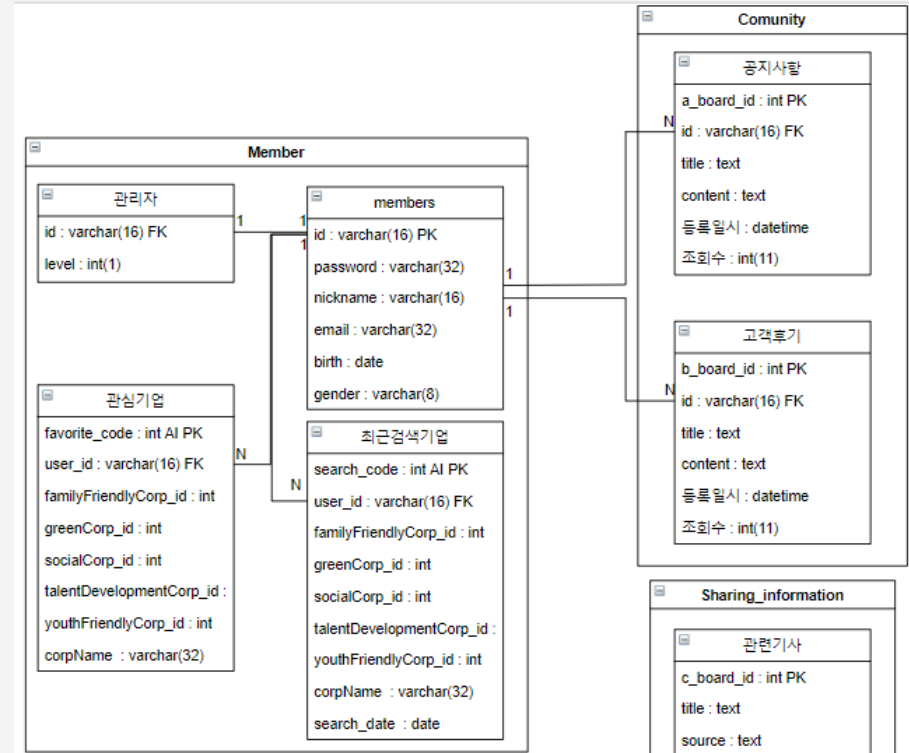
데이터베이스 설계

1. Member schema :

1. 웹서비스에서 회원 정보를 관리하거나 저장하는 DB
2. 회원이 기업을 스크랩하거나 검색한 기업을 데이터로 저장한다.
3. 각 테이블들은 members의 id를 참조(reference)한다.

2. Community schema :

1. 게시판 서비스의 정보를 저장하는 DB
2. Member.members의 id를 참조한다.



▲ DB 설계서(Member, Community)

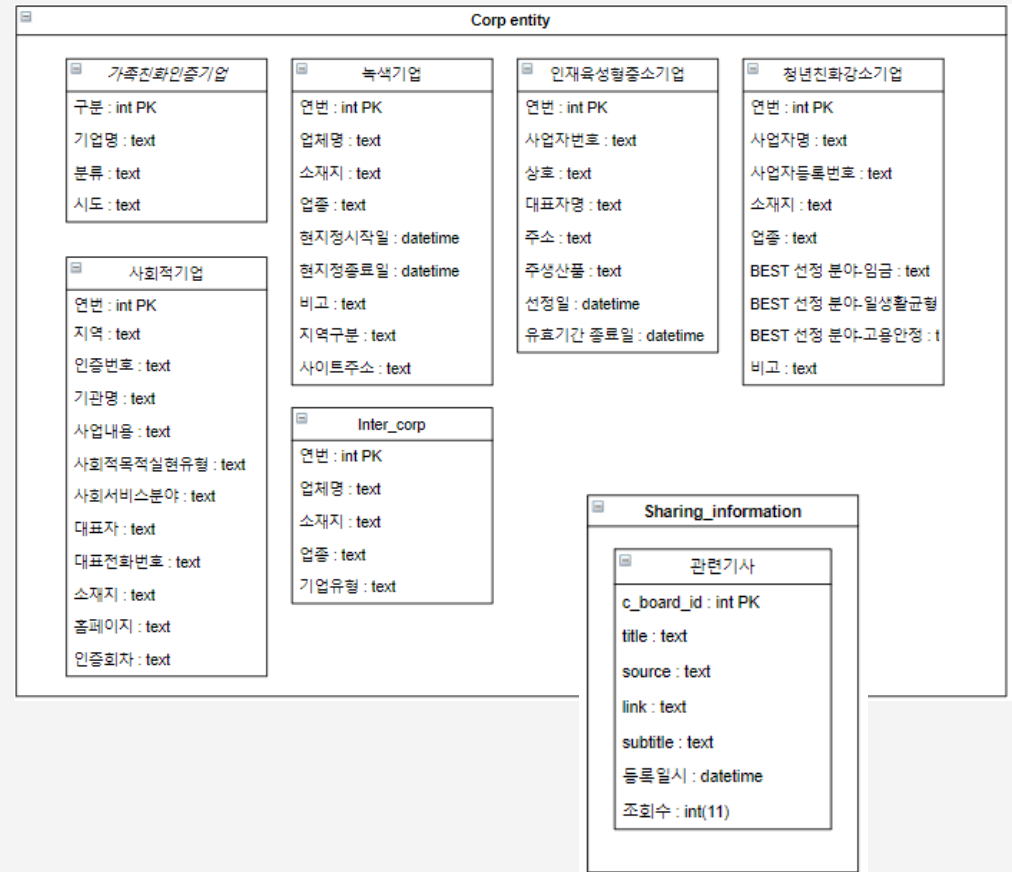
데이터베이스 설계

3. Corp schema :

1. 각 카테고리 별로 선정된 기업들의 정보를 저장한 DB
2. 공공데이터포털에서 기업정보를 가져와 적재한 후 전처리

4. Sharing_information schema :

1. 해당 관련 기업들의 기사를 크롤링 후 저장한 DB
2. 크롤링 언어 : 파이썬
(Beautifulsoup, requests)



▲ DB 설계서(Corp, Sharing_information)

04.현재 진행 상황

산출물 검토 기법 및 변경 사항

각 파트별 진행 상황

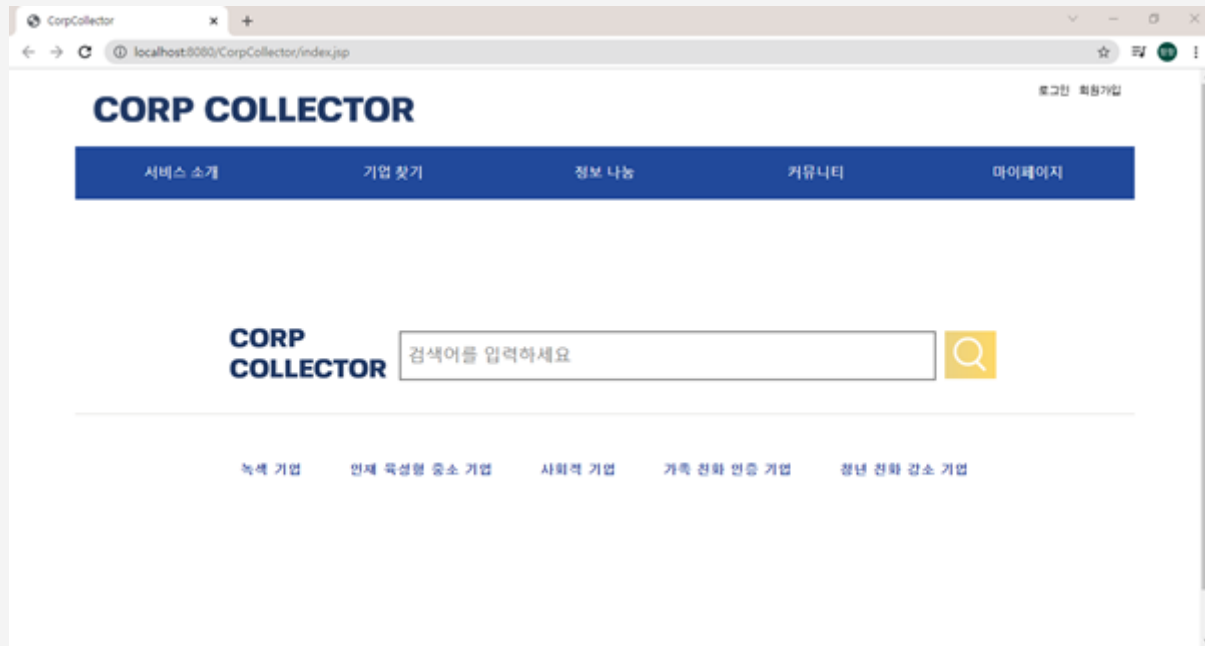
형상관리 - 깃 플로우 차트

산출물 검토 기법 및 변경 사항

1. 동료 검토 기법 사용 >> 약 1주일 단위로 회의 시간을 가졌다.
2. 변경 사항: 회원가입, ID/PW 찾기 시 구글/네이버 SMTP를 통해 사용자의 이메일로 인증 번호를 발송한다.
 - 과거에는 앱 비밀번호 발급으로 관리자의 구글/네이버 이메일 계정으로도 타 사용자 컴퓨터 환경에서 인증이 가능했으나 현재 불가능하다.
 - 추후 AWS SES(Simple Email Service)를 통해 이메일 인증 코드를 첨부할 예정이다.



현재 진행 상황

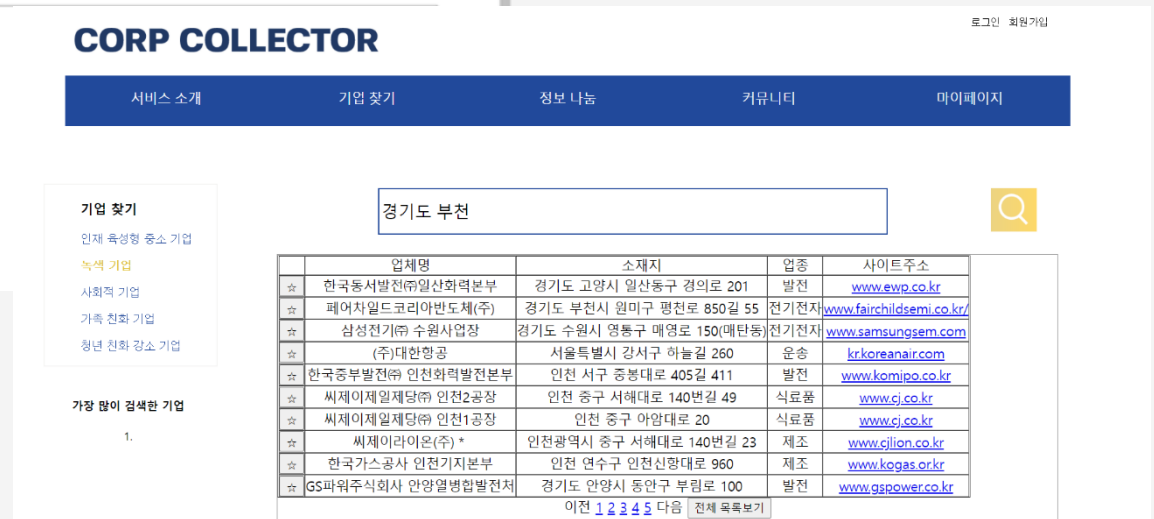


▲ **인덱스 페이지 작업물**

현재 진행 상황

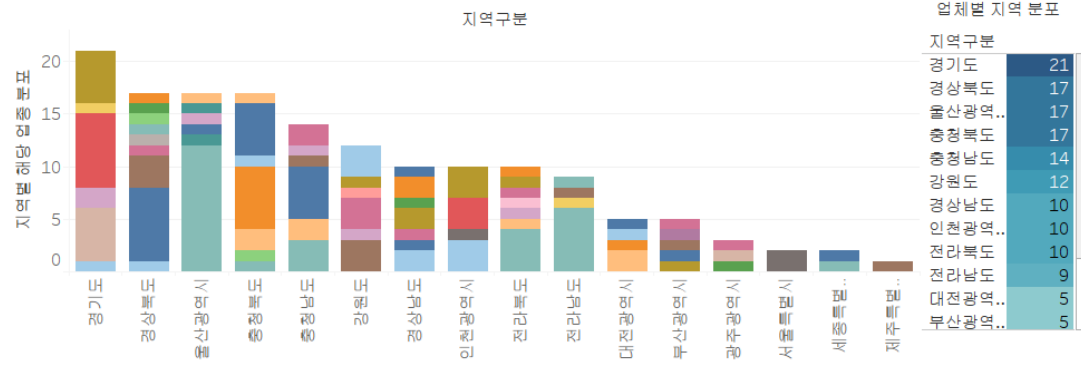
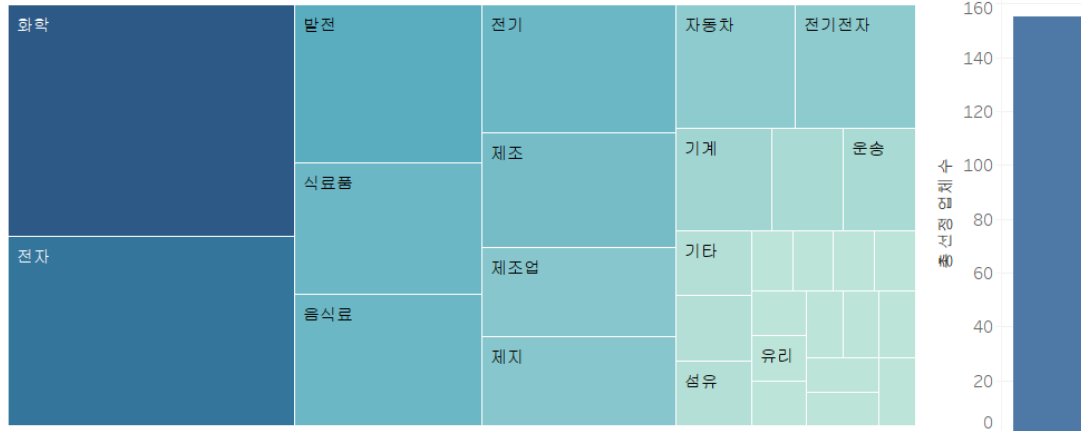


▲ 기업 찾기 페이지 작업물 ▶



현재 진행 상황

업체별 업종 분포



▲ 데이터 시각화

1. 사용 시각화 도구 : Tableau
2. Tableau를 Corp DB에 연결해 사용
3. 사용한 기업 데이터 목록 :
가족친화인증기업, 녹색기업,
사회적기업, 인재육성형
중소기업, 청년친화강소기업

현재 진행 상황

```

crawling_sitemap
def crawler(sitemap,page,query,sort,s_date,e_date):
    s_from = s_date.replace("-",":")
    e_to = e_date.replace("-",":")
    page = 1
    maxpage = (int(sitemap)-1)+10+1
    while page <= maxpage:
        url = "https://search.naver.com/search.naver?where=news&query=" + query + "&sort=" + sort + "&sd=" + s_date + "&ed=" + e_date + "&no=s03Ar%2Cp3Afrom" + s_from + "to" + e_to + "%2C3A"
        html = response.text

        # BeautifulSoup 인자값 지정
        soup = BeautifulSoup(html, 'html.parser')

        # <p>태그에서 제목과 링크주소 (a 태그 중 class 명이 news_title인 것)
        atags = soup.find_all('a', 'news_title')
        for atag in atags:
            title = atag.get('title')
            title_text.append(title)
            link_text.append(atag['href'])

        # 신문사 추출 (a 태그 중 class 명이 info_press인 것)
        source_lists = soup.find_all('a', 'info_press')
        for source_list in source_lists:
            source_text.append(source_list.text)

        # 본문요약본 (a 태그 중 class 명이 api_txt_lines_desc_txt_wrap인 것)
        contents_lists = soup.find_all('a', 'api_txt_lines_desc_txt_wrap')
        for contents_list in contents_lists:
            contents_cleaning(contents_list)

        # 모든 리스트를 딕셔너리형태로 저장
        result = [{"title":title_text, "source":source_text, "link":link_text, "sub_title":query}]
        df = pd.DataFrame(result)
        df = df
        page += 10

[8]
import pandas as pd
green = pd.read_csv('/content/녹색기업 merging.csv')
family = pd.read_csv('/content/가족친화인증기업 merging.csv')
social = pd.read_csv('/content/사회적기업 merging.csv')
iniae = pd.read_csv('/content/인재개발원소기업 merging.csv')
gangs = pd.read_csv('/content/청년친화원소기업 merging.csv')

관련기사 = pd.concat([green, family, social, iniae, gangs], axis=0)

관련기사 = 관련기사.reset_index(drop=True).reset_index().rename(columns={'index':'c_board_id'})
관련기사.head()

c_board_id title source link sub_title
0 0 홍남기, 열린 직접 민선 "반도체 정보제공, 확기업 우려" 전달 이시애경제전문사 선정 https://www.asiae.co.kr/article/2021101510222... 녹색기업
1 1 RE100 참여 기업들, 정부에 "재정에서 보금 확대해 달라" 뉴스스 http://www.newsis.com/view/?id=NSK20211013_00... 녹색기업
2 2 신원 "국내 최초 녹색구조화재로 발행" 연합뉴스 http://yna.kr/AKR2021102063200002?tid=1195m... 녹색기업
3 3 기업 편향 녹색성장전략 OUT 뉴스1 https://www.news1.kr/photos/view/75016244... 녹색기업
4 4 중국 석탄 문제, 효묘번호 아닌 '녹색고양이'가 참을까 프레스인문전문사 선정 https://www.pressian.com/pages/articles/202110... 녹색기업

from sqlalchemy import create_engine
engine = create_engine("mysql+pymysql://0hkyu1n1:*@corpcollector.cjctekukwo.ap-northeast-2-rds.amazonaws.com:3306/Sharing_inforamation?charset=utf8", encoding = "utf-8")
conn = engine.connect()

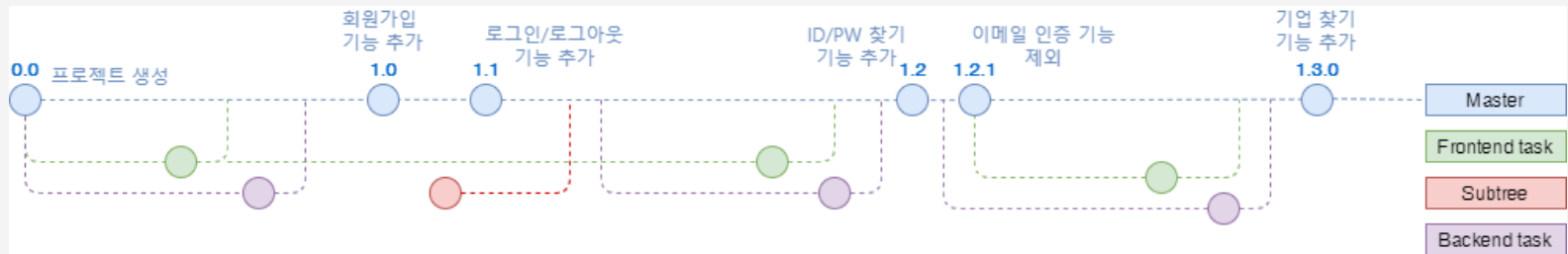
[ ]
관련기사.to_sql(name='관련기사', connengine, if_exists='append', index=False)

```

▲ 기업 관련기사 크롤링

1. 크롤링 사용 언어 : 파이썬
(Beautifulsoup, requests)
- Colab
2. 네이버 뉴스의 url을 requests로
받아 해당 뉴스 페이지의
뉴스제목, 신문사, link를
beautifulsoup으로 파싱해
데이터(csv)로 저장하는
방식의 크롤링
3. sqlalchemy 라이브러리를 활용
하여 파이썬으로 SQL에 연결해
데이터를 적재하는 방식

형상 관리 - 깃 플로우 차트



▲ **구현 리포지토리 깃 플로우**

TEAM 그릴그린

THANKYOU