공공데이터 API를 이용한

노인 일자리 분석

2팀 1조

김상희 김혜민 남윤아 이성희 이하윤

주제 선정 이유

노인일자리 공고와 사업 정보를 분석하여 노인일자리 구인 동향을 파악하고,
정부의 노인 복지 정책의 효과성을 평가해보고자 함

1

“분석목적”
최근 3개년 동안의 노인일자리 공고와 사업 정보를 수집하여
노인일자리 채용 현황과 고용 지속성을 분석함

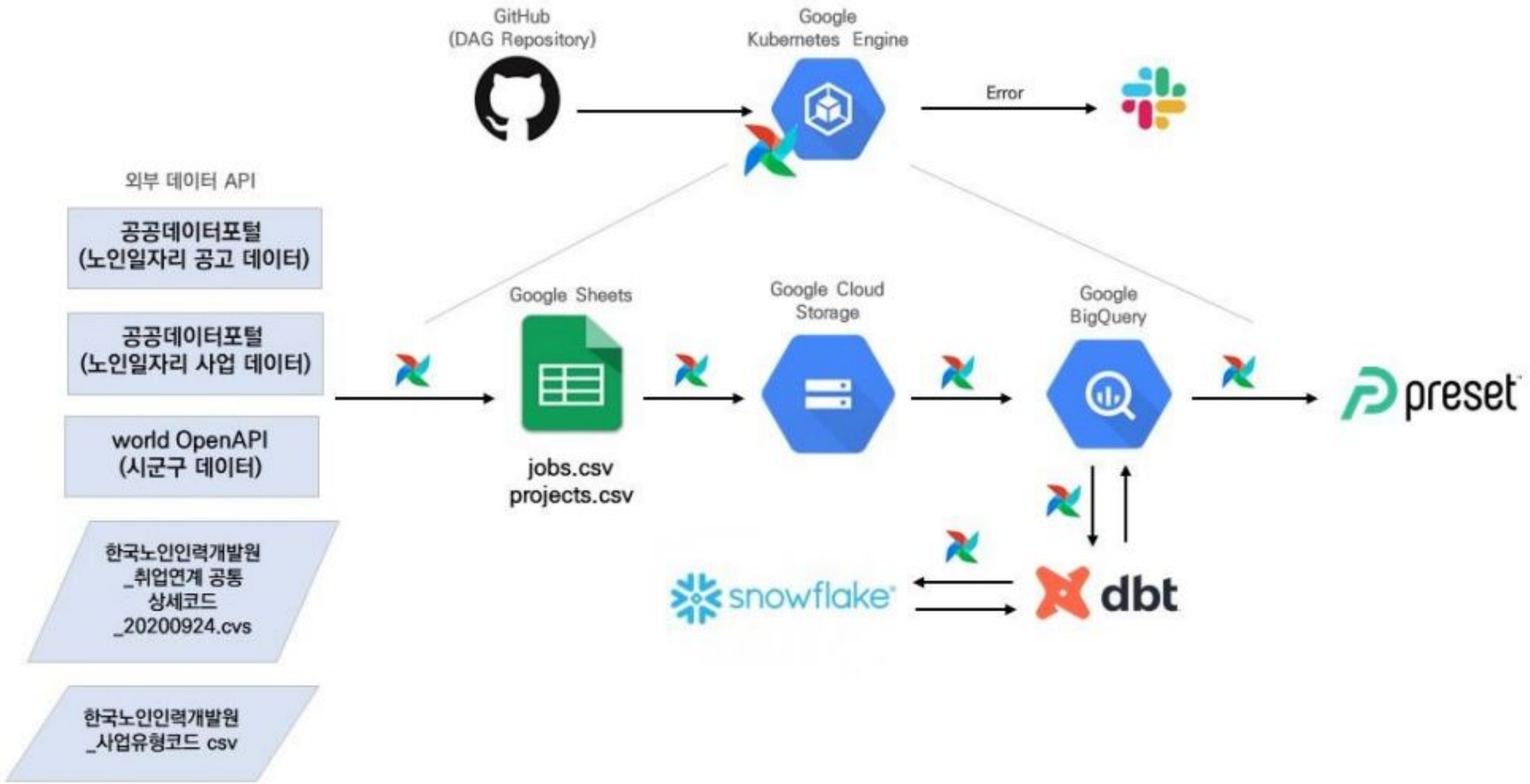
2

“다양한 기술 학습”
공공데이터 API를 통해 데이터를 읽어오는 작업을 Airflow로 구현하여 데이터 수집 작업 자동화, 읽어온 데이터를 시각화하기 쉬운 형태로 데이터 스키마를 결정하여 데이터 웨어하우스에 저장하는데 DBT 사용, Superset을 활용한 인터랙티브 대시보드 생성

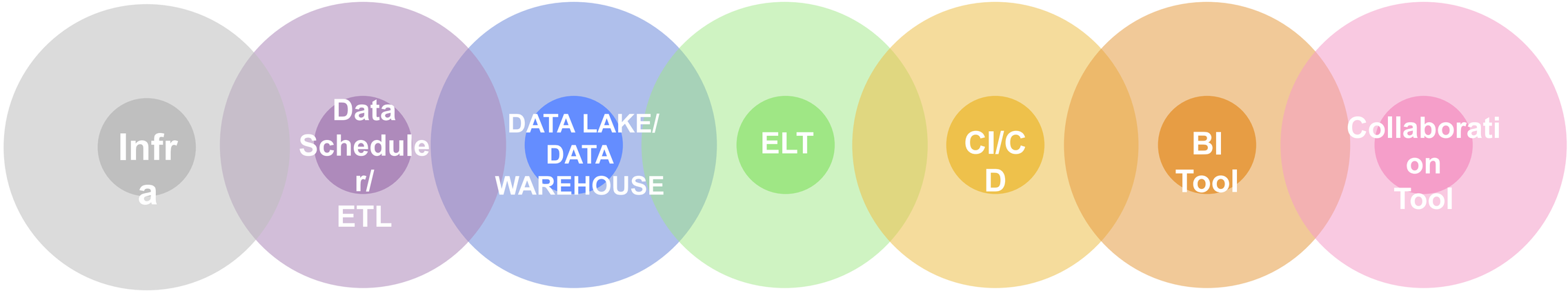
3

“기대 효과 및 활용 방안”
1. 노인일자리에 대한 상세한 이해를 도모할 수 있음
2. 노인일자리 정책의 성과와 한계를 파악하고 개선할 수 있는 방안을 모색할 수 있음

Architecture



활용 기술



Google Cloud Platform

Google Kubernetes Engine

Apache Airflow

Cloud Storage

Google BigQuery

snowflake

dbt

GitHub Actions

docker

preset

GitHub

slack

Notion

DB Schema 설계

Jobs

공공데이터포털에서 2018년, 2020년, 2022년에 대한 한국노인인력개발원의 노인 구인정보 데이터를 수집한 테이블 정보

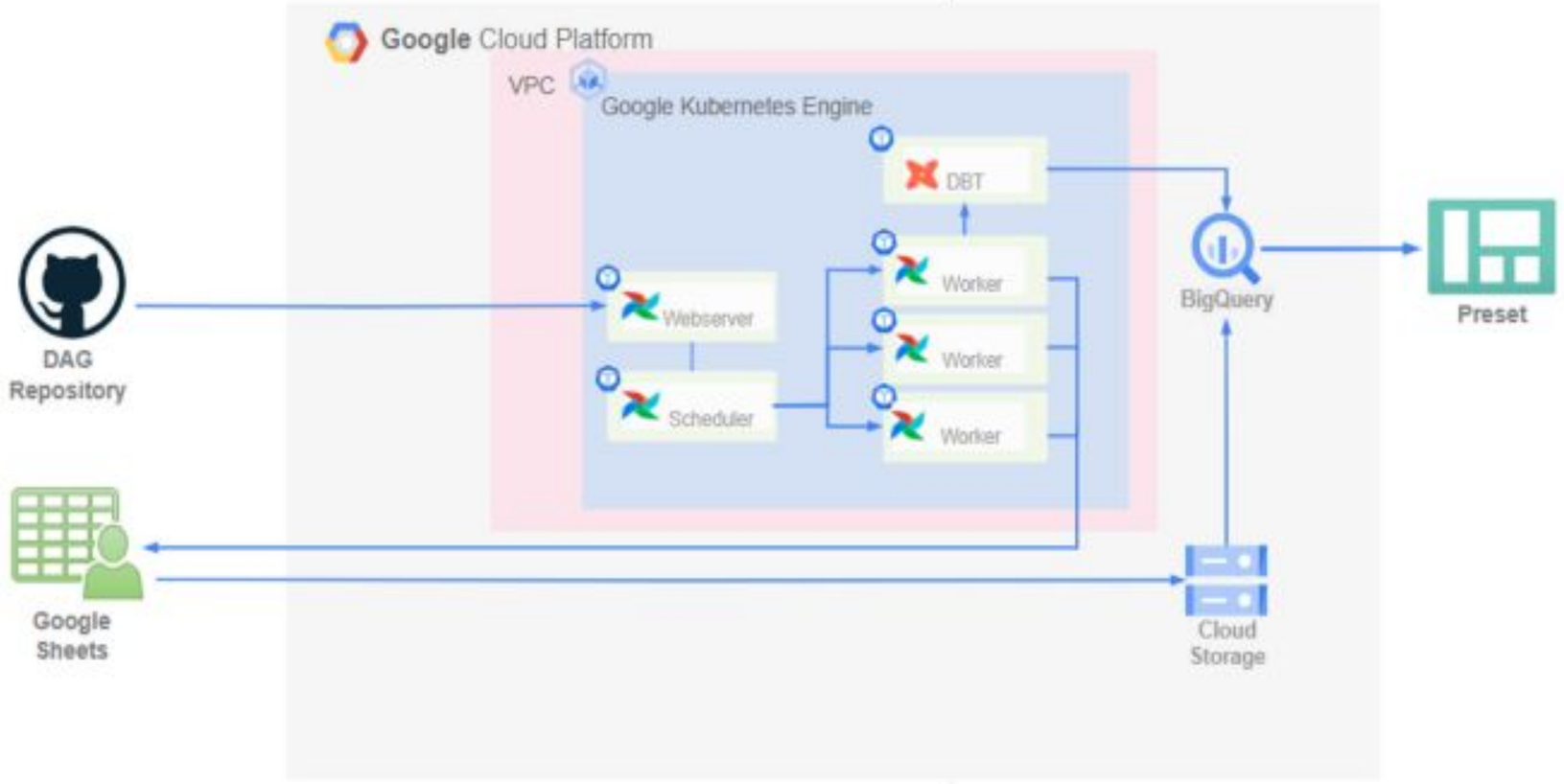
jobs	
jobId	**varchar(20)**
jobcls	varchar(50)
jobclsNm	varchar(50)
acptMthd	varchar(2000)
deadline	varchar(50)
emplymShp	char(6)
emplymShpNm	varchar(6)
startDd	char(10)
oranNm	varchar(50)
organYn	char(1)
recrtTitle	varchar(1000)
stmId	char(1)
stmNm	varchar(50)
endDd	varchar(10)
workPlc	varchar(50)
acptMthdCd	varchar(6)
age	varchar(2)
ageYn	char(1)
clerk	varchar(50)
clerkContt	varchar(15)
cllfPrnnum	varchar(4)
createDt	varchar(100)
detCnts	varchar(4000)
etcItm	varchar(4000)
homepage	varchar(100)
plDetAddr	varchar(200)
plbizNm	varchar(1000)
updDt	varchar(100)
sysCreatedAt	timestamp_ntz

projects

공공데이터포털에서 2018년, 2020년, 2022년에 대한 한국노인인력개발원의 노인 일자리 사업 통합 정보 데이터를 수집한 테이블 정보

projects	
projNo	**char(20)**
projType	varchar(30)
projPlanChangeNo	int
projYear	char(4)
contProjYn	char(1)
contProjStartYear	char(4)
projTypeCd	char(10)
projTypeNm	varchar(100)
nonBudgYn	char(1)
specProjCd	varchar(10)
projNm	varchar(100)
admProvNm	varchar(10)
admDistCd	varchar(10)
admDistNm	varchar(50)
institutionId	varchar(10)
projStartDd	char(10)
projEndDd	char(10)
planStatusCd	varchar(20)
targetEmployment	int
firstlAttachment	varchar(100)
recentApprovalAttachment	varchar(100)
delYn	char(1)
sysCreatedAt	timestamp_ntz

Infra Architecture



GCP 인프라 구축

1

Helm 차트를 활용하여 GKE에 Airflow 구축

Apache Airflow Helm Chart를 사용하여 GKE에 Airflow를 배포함
- Airflow는 GKE 클러스터 내에서 실행되며 웹 서버, 스케줄러, 워커, PostgreSQL 등의 컴포넌트가 Pod로 실행됨
- Airflow를 사용하여 ETL 및 ELT 등 작업 스케줄링 및 워크플로우 관리를 가능하게 함

2

Helm 차트를 활용하여 GKE에 values.yaml 파일 사용자 정의

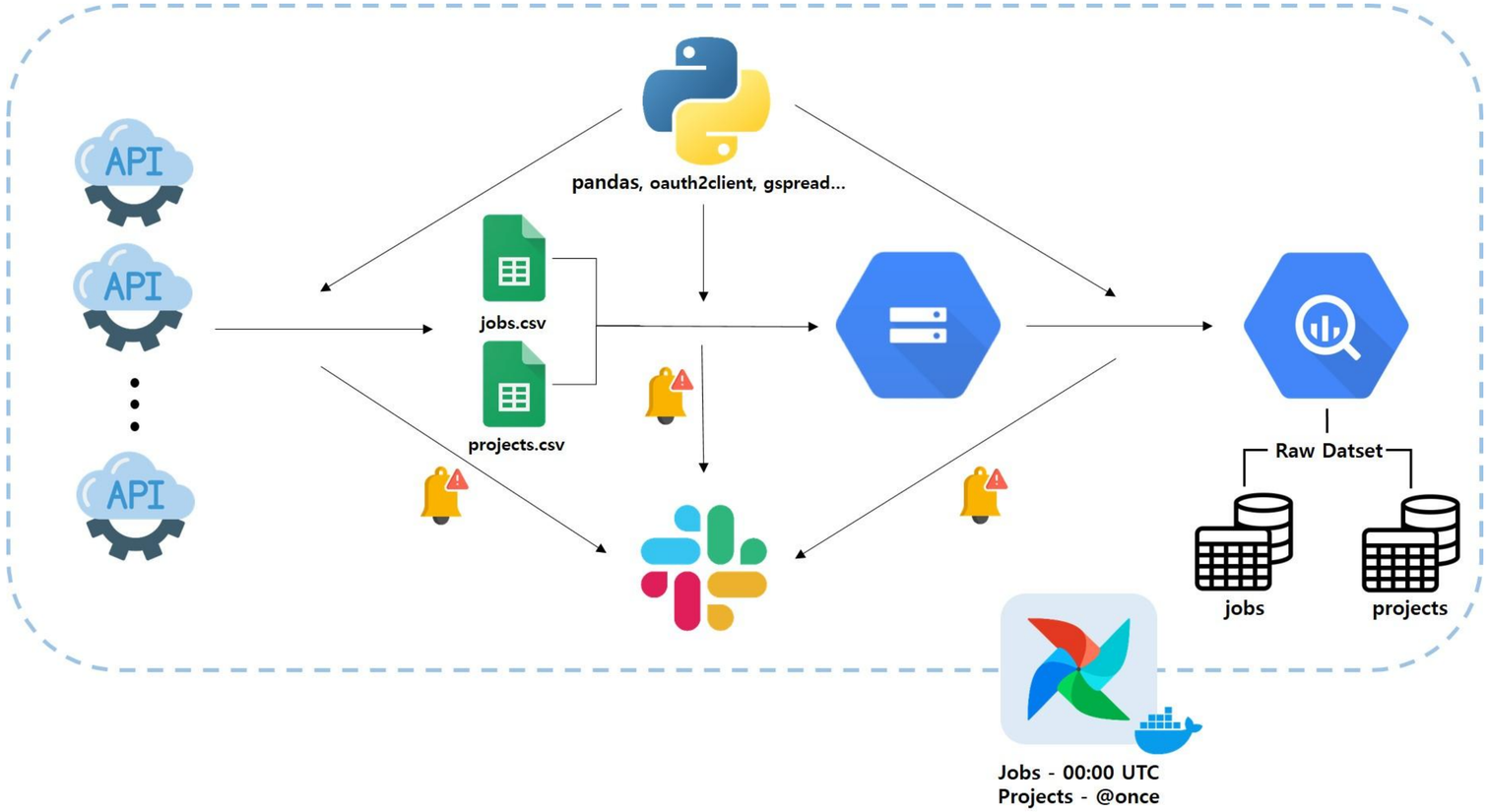
- KubernetesExecutor 사용
- Git Sync DAGs 배포하기
- Airflow API 활성화

3

CI/CD

Git Action을 이용하여 dags 폴더안의 위치한 DAG의 코드를 검사하는 과정을 구현함

Airflow를 이용한 ETL



✔ **데이터 전 처리**

- API를 통해 데이터를 추출하여 구글 스프레드시트에 csv로 저장
- 데이터 전 처리 수행 후 Google Cloud Storage에 적재

✔ **데이터 적재**

- BigQuery에서 raw_data 데이터 셋(Data set) 아래에 jobs(공고), projects(사업) 테이블 생성
- Google Cloud Storage에 있는 csv 파일을 각 테이블에 벌크 업데이트

Airflow Dag

1) jobs_crawling.py

노인 일자리 공고 데이터 수집을 실행하는 DAG파일로 매일 자정(UTC 기준)에 한번씩 ETL 시행

✓ **job_list**
- 구인 정보 API의 데이터를 구글 스프레드 시트 내 **‘job_list_crawl’** 시트로 저장

✓ **job_detail**
- 채용 공고 ID를 사용하여 상세 공고 정보를 구글 스프레드 시트내 **‘job_detail_crawl’** 시트에 저장

✓ **Joining**
‘job_list_crawl’와 **‘job_detail_crawl’** 시트 내 데이터를 전 처리 후 **‘jobs’**시트로 통합

✓ **GtoB**
Google Cloud Storage에 csv 파일로 저장 및 BigQuery에 벌크 업데이트

Airflow Dag

2) projects_crawling.py

노인 일자리 사업 데이터 수집을 실행하는 DAG파일로 DAG 실행 시 한번 만 ETL 시행

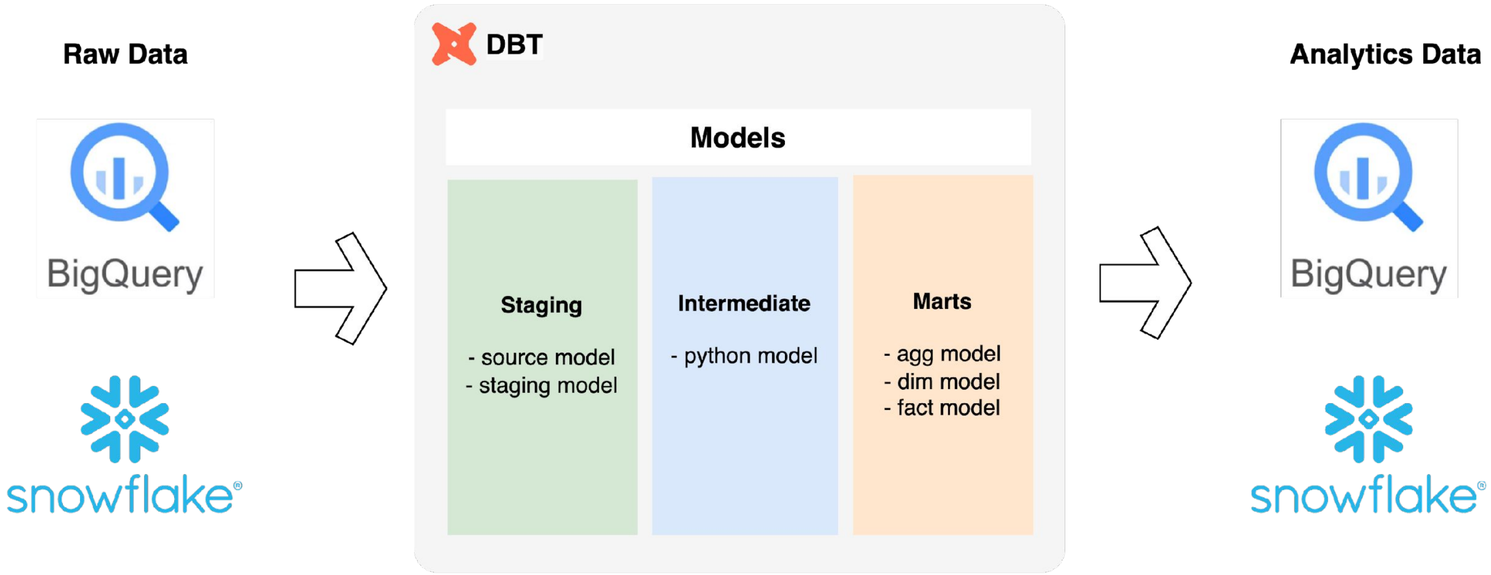
✓ **ProjectList**
- 사업 목록을 API로 가져와 구글 스프레드시트 내 '**projects**' 시트로 저장

✓ **GsheetToBigquery**
- '**projects**'시트 내 데이터를 Google Cloud Storage에 '**projects.csv**'로 업로드
- 이후 BigQuery에 벌크 업데이트

DBT를 이용한 ETL

1) DBT ELT 프로세스 구축

모델링 레이어를 3단계로 나누어, 단계별로 가공과 품질을 검증



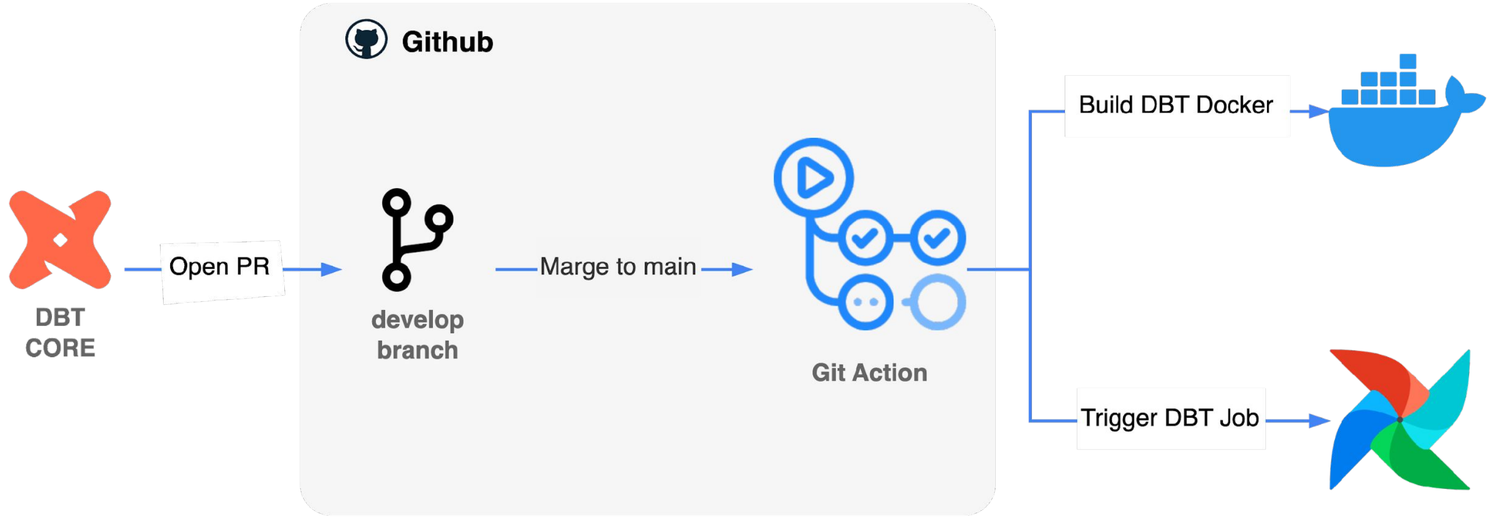
✓ **스테이징 모델**
- 원시 데이터를 가공하여 스테이징 모델을 정의
- JOIN과 집계함수를 제외한 데이터 가공만 수행

✓ **중간 모델**
- 스테이징 모델을 기반으로 집계함수, JOIN 및 복잡한 가공 처리
- Google Cloud DataProc , Pandas를 이용한 Python 모델 처리

✓ **마트 모델**
- 중간 모델과 스테이징 모델을 활용하여 분석팀 요청에 맞는 모델 정의
- fact 모델은 Incremental update를 수행

DBT를 이용한 ETL

2) DBT 배포 프로세스 구축



- ✓ **Airflow** 와 코드베이스를 분리하기 위해 **DBT** 프로젝트를 별로도 생성
- ✓ **develop, main** 브랜치 **PR** 이벤트가 발생 시, **dev, stage DW**에 모든 모델을 생성하고 검증
- ✓ **main** 브랜치 **push** 이벤트가 발생 시, 도커 이미지 생성 및 **Airflow DAG** 트리거

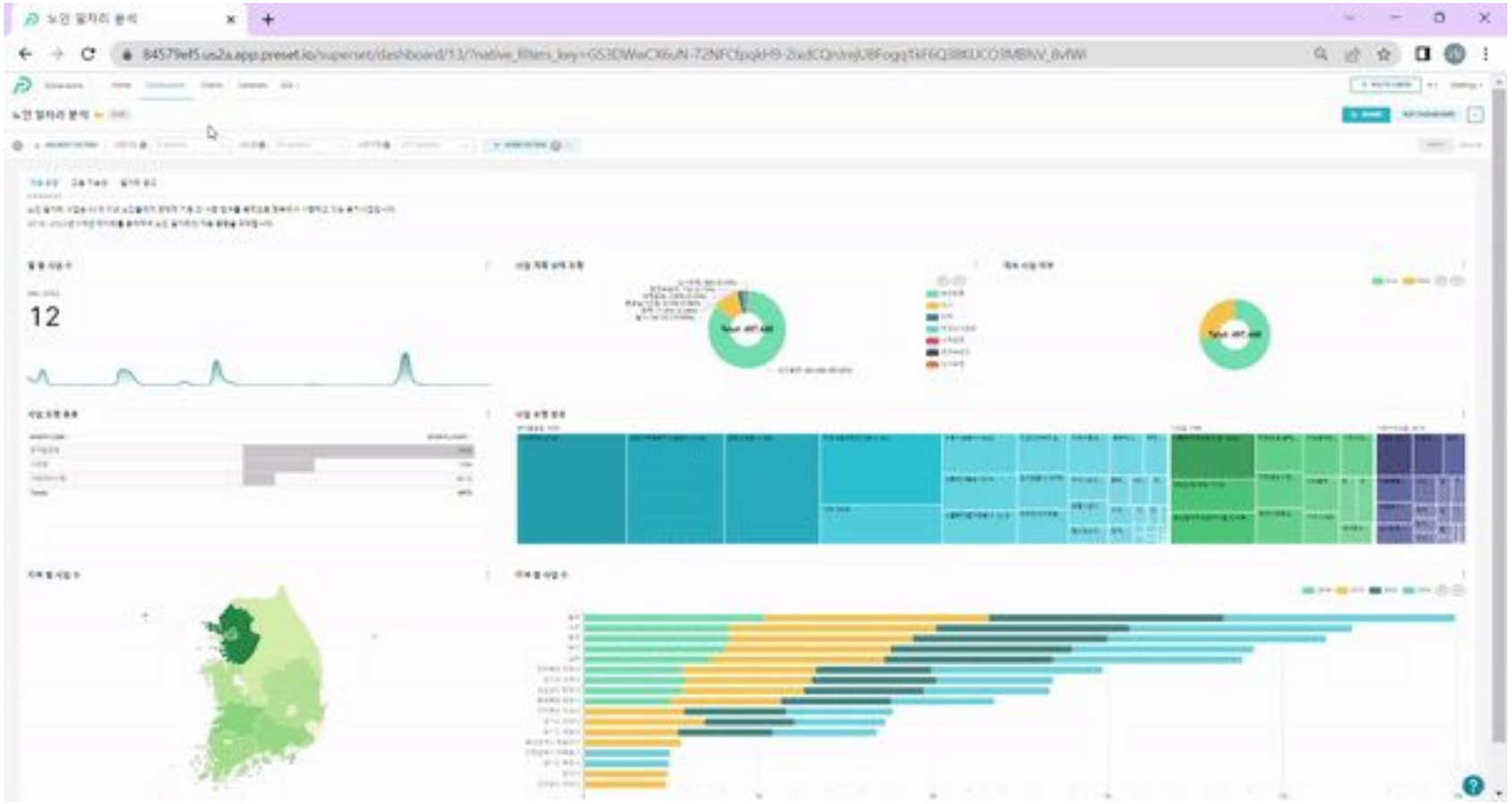
DBT를 이용한 ETL

2) DBT DAG 생성

- ✓ Airflow KubernetesPodOperator를 이용해 DBT 이미지를 가져와 DBT 명령 수행
- ✓ 매일 1회 DAG를 수행
- ✓ 에러가 발생할 경우 Slack 채널에 알림

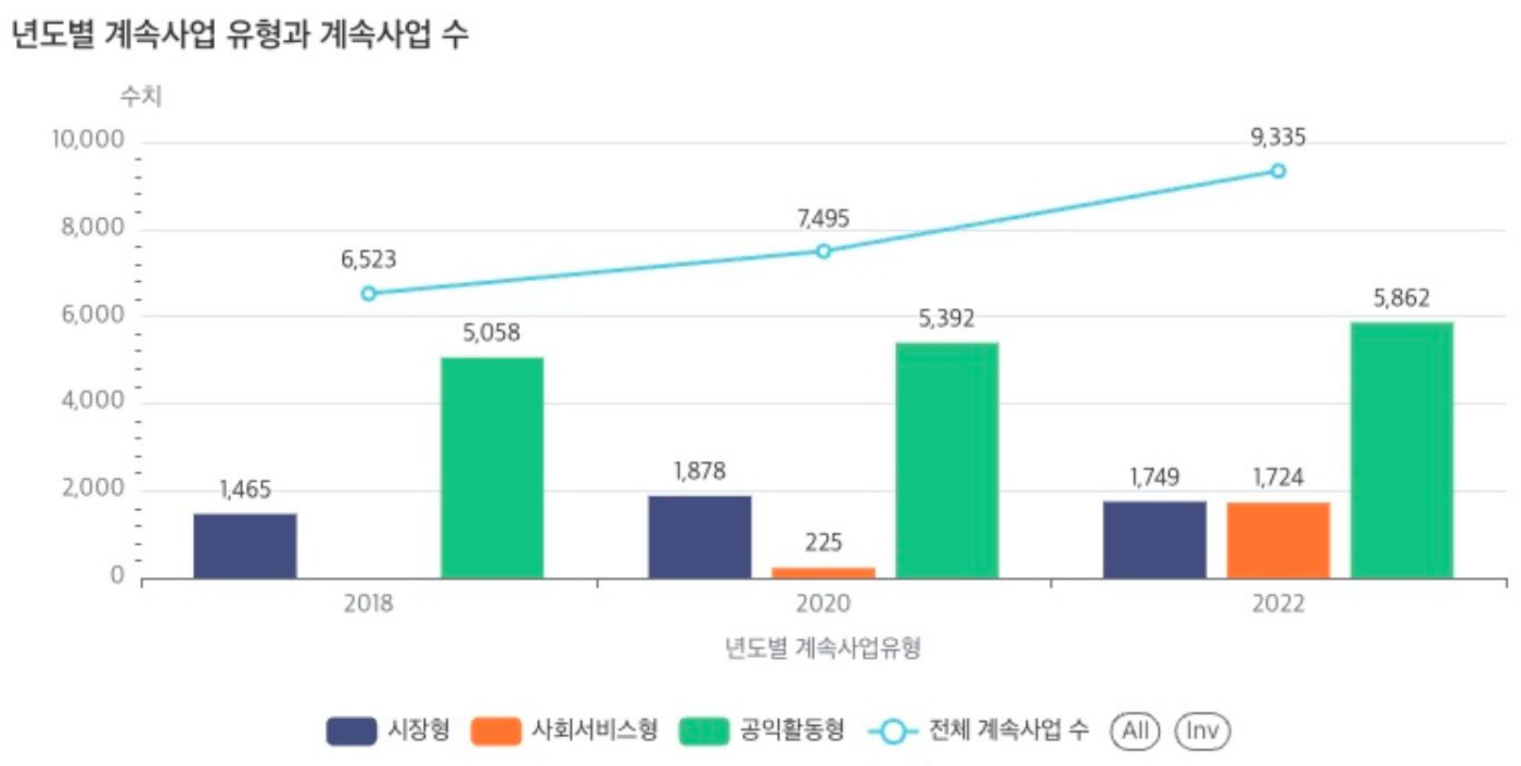
대시보드 가시성 향상

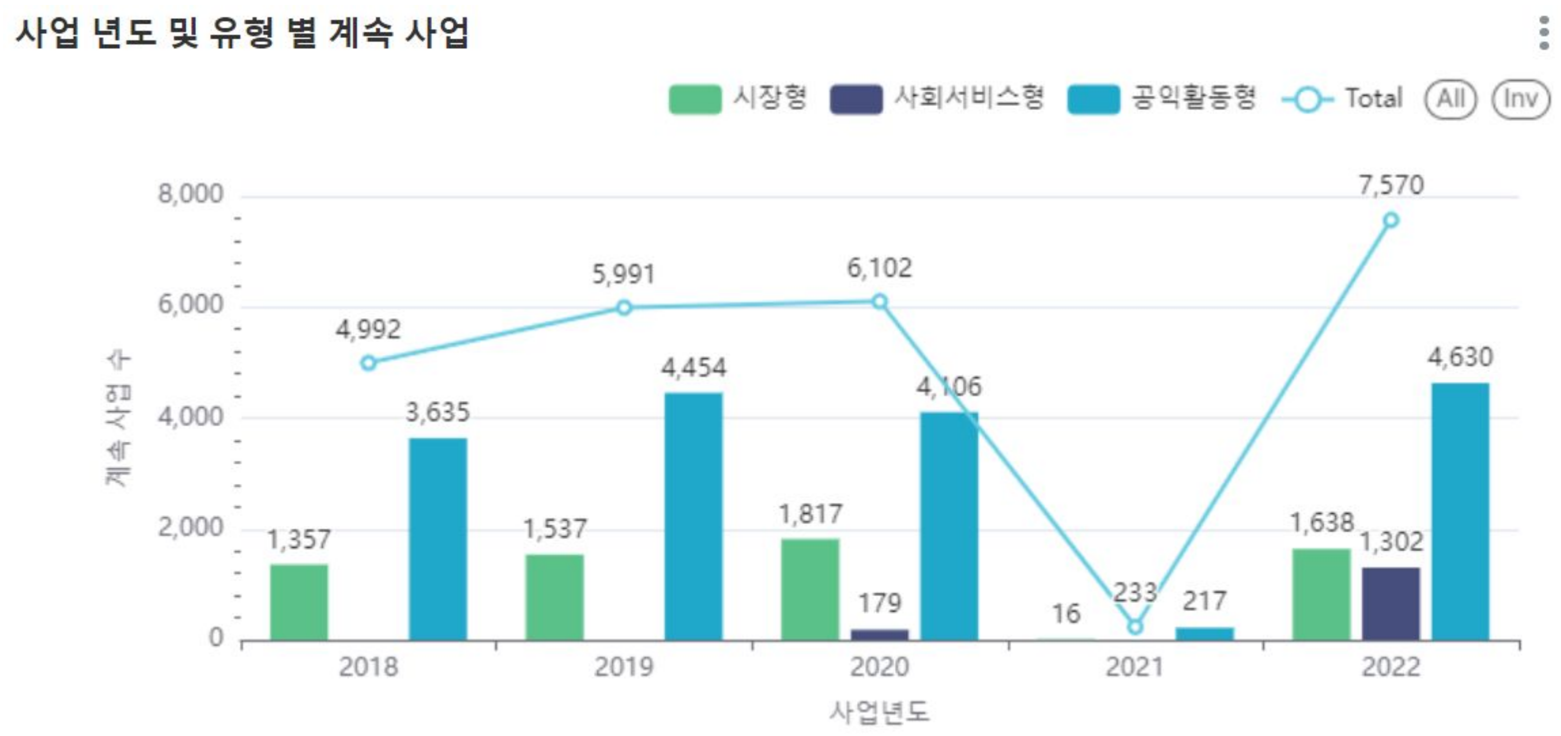
1) 탭 기능 추가: 연관된 차트를 하나의 탭으로 구성



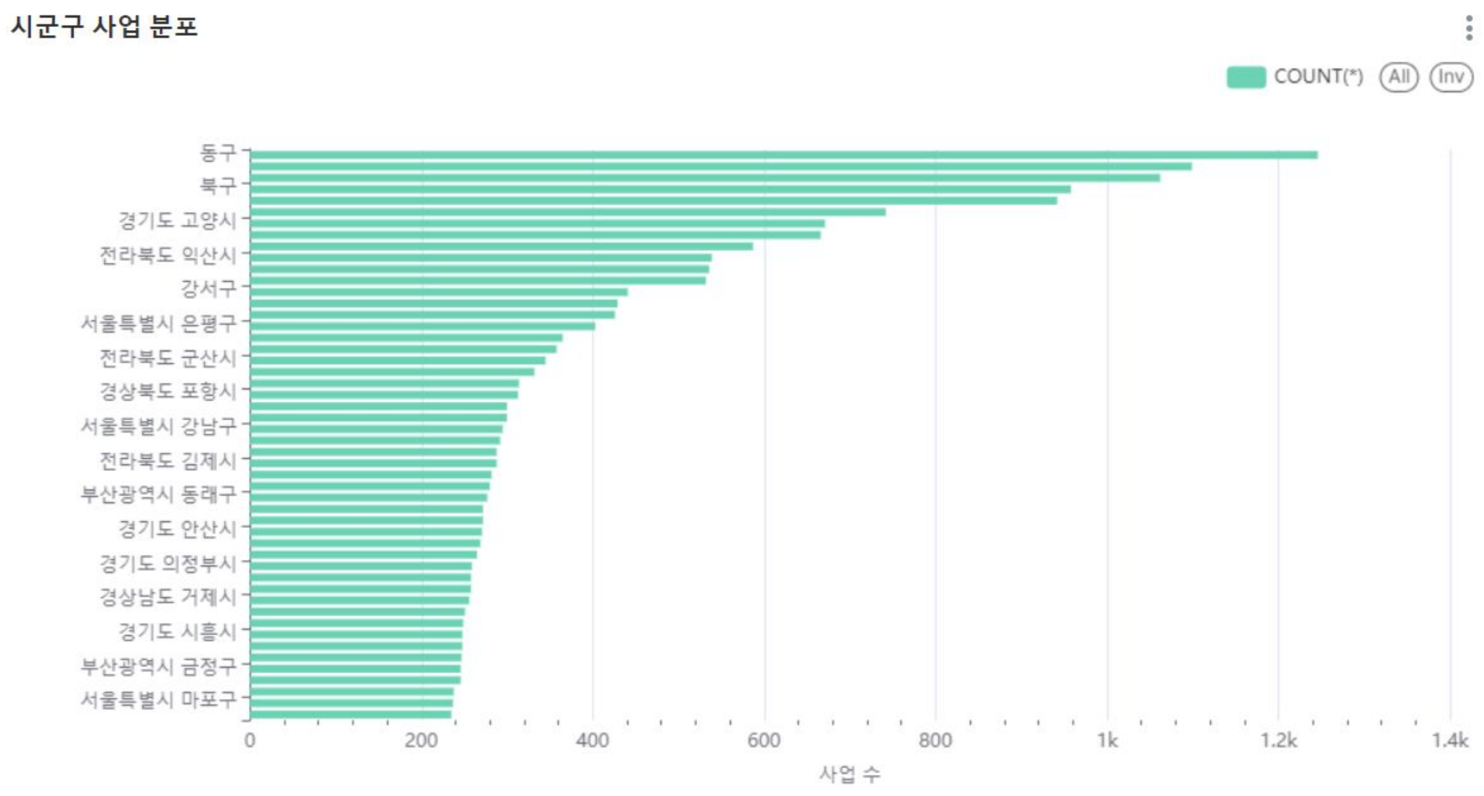
대시보드 가시성 향상

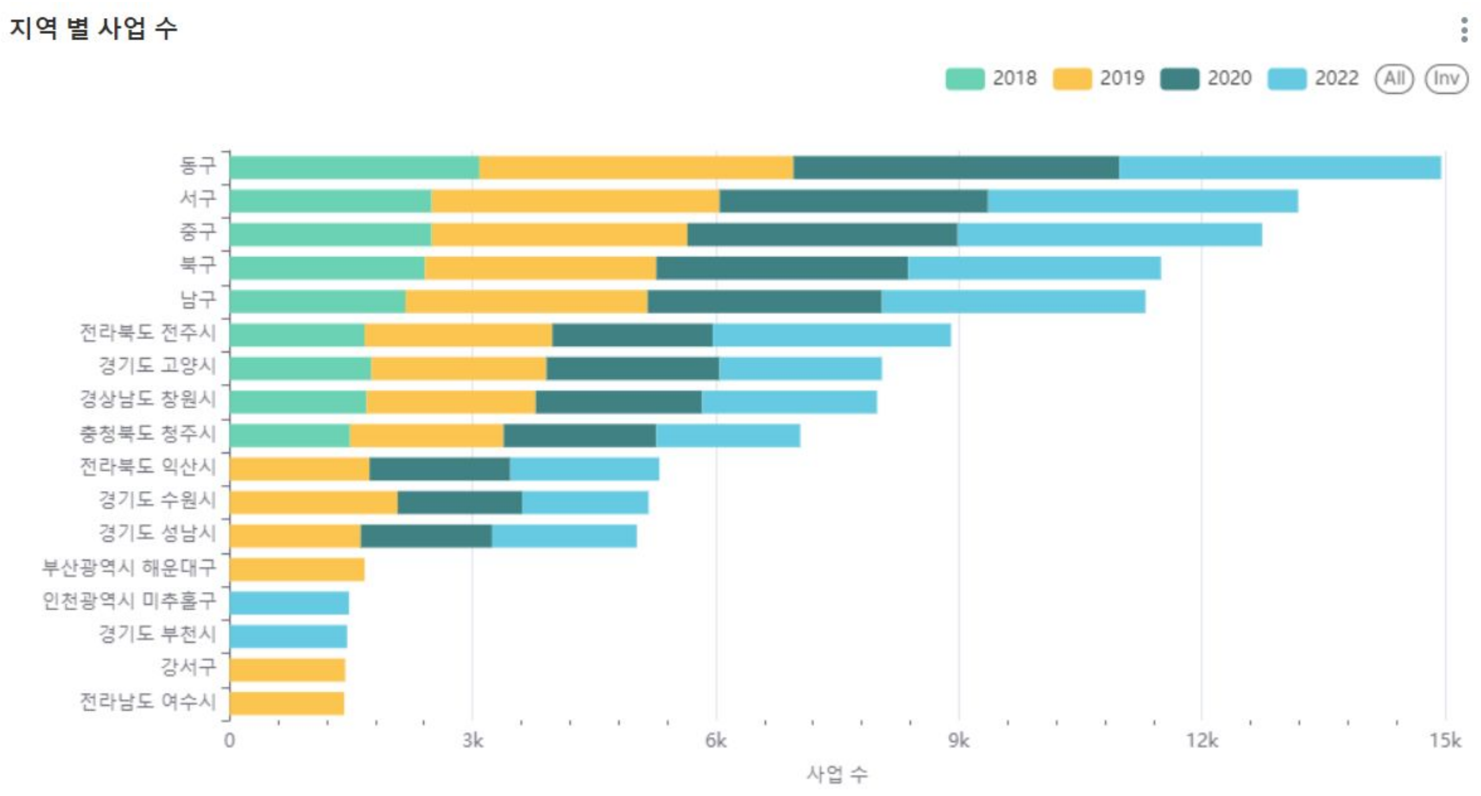
2) 다양한 옵션 추가 – title, dimension





✔ 어떤 데이터를 표현하고 있는지 **title**을 통해 확인할 수 있음

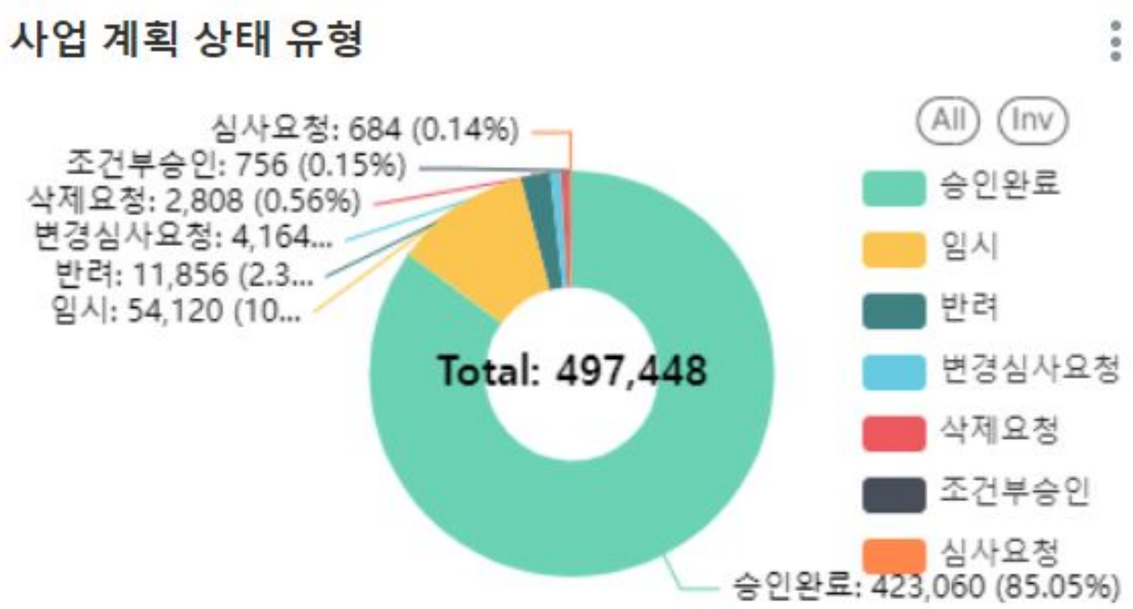




✔ 지역 및 사업 년도 별 사업 수를 파악할 수 있음

대시보드 가시성 향상

3) Pie 차트 개선

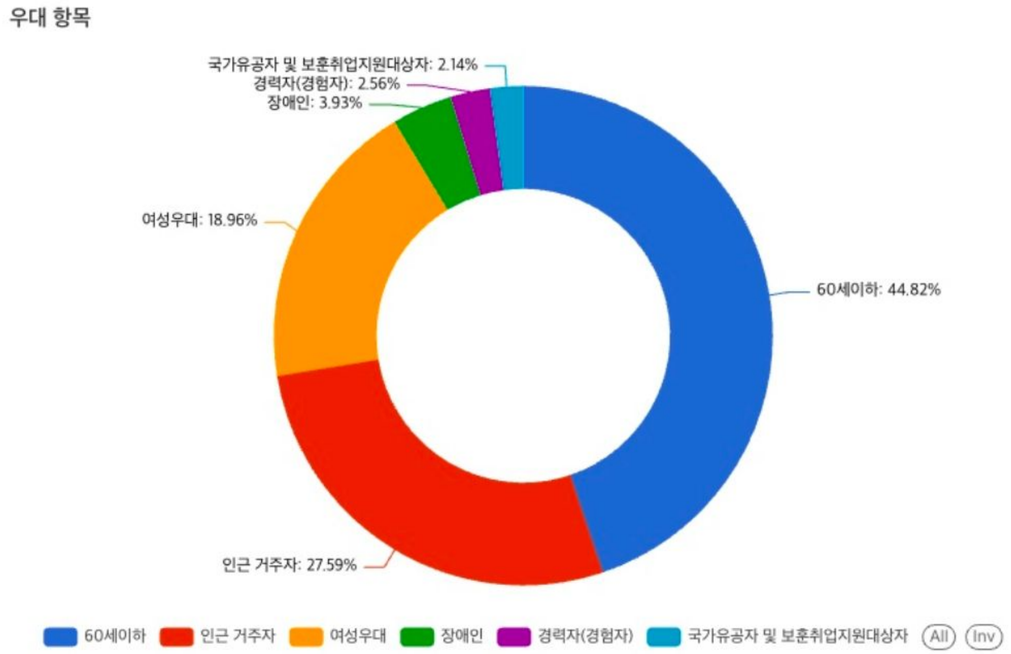


1 **Total Value 추가**
전체 value의 합을 추가

2 **lagend 추가**
legend 목록이 잘 보이도록 우측 혹은 상단에 배치

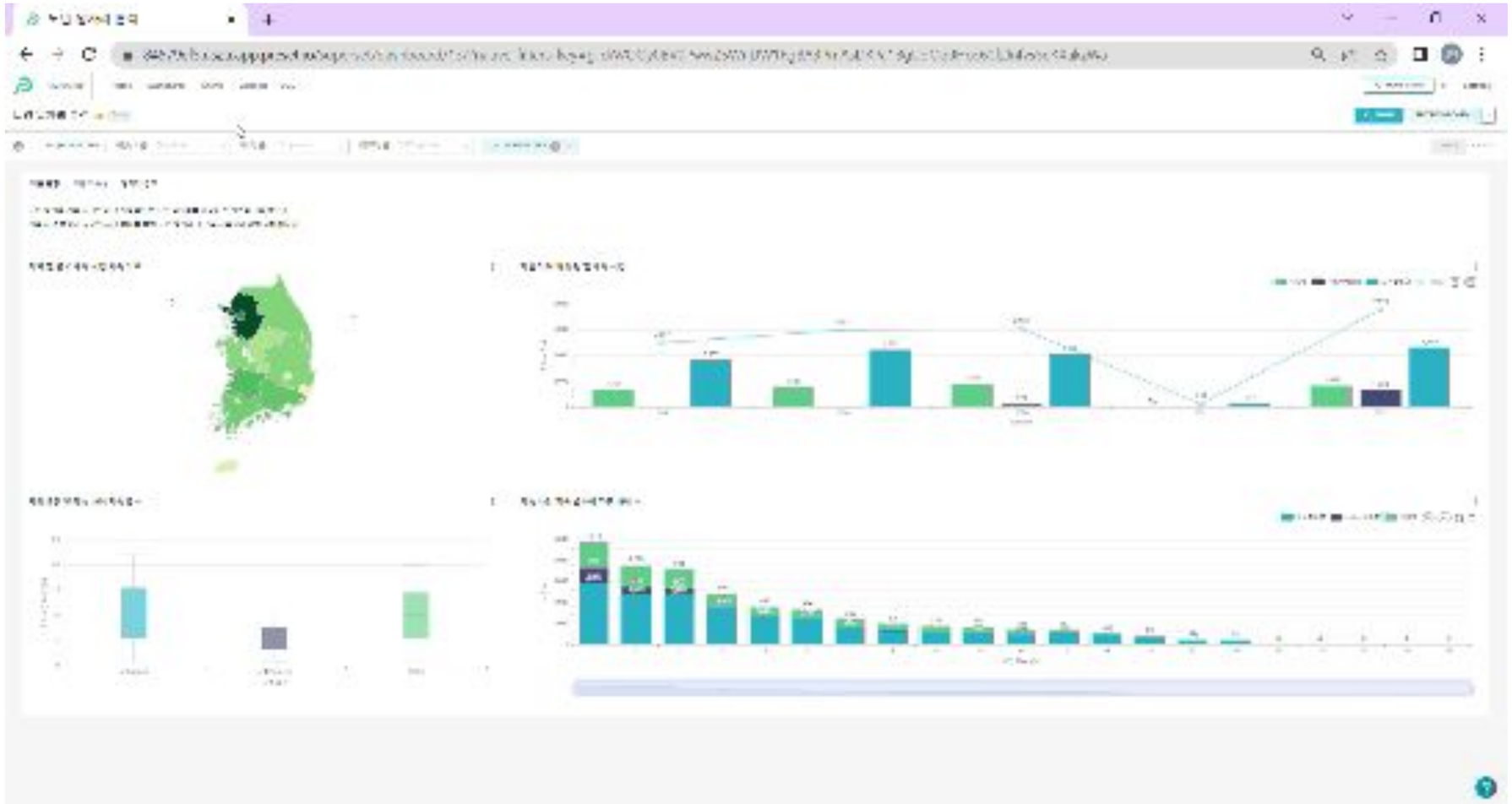
대시보드 가시성 향상

4) 차트 유형 변경: 데이터를 더 잘 표현할 수 있는 차트로 변경



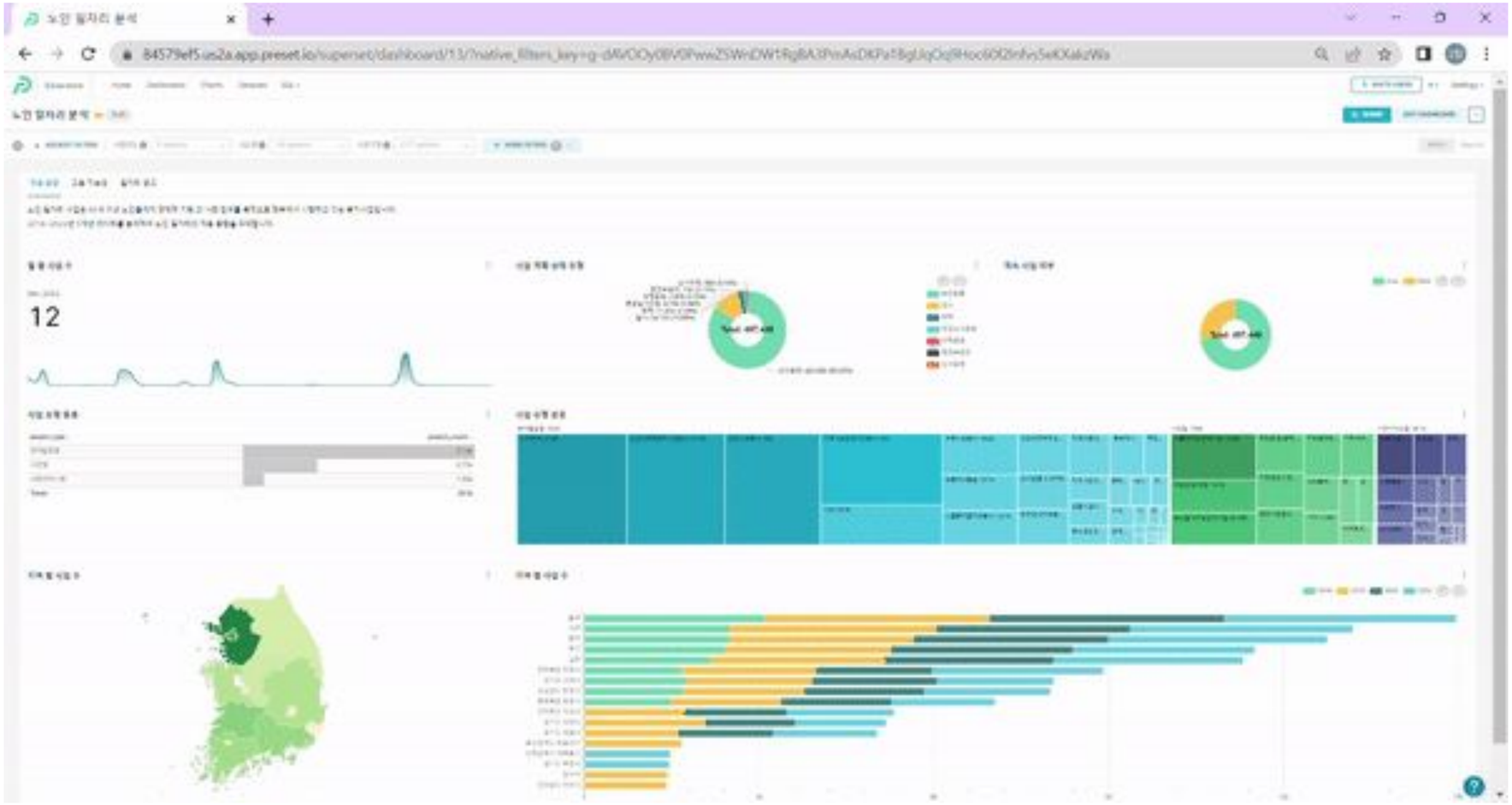
Interactive Chart 구현

1) 전체 필터 기능: 필터를 적용하여 원하는 데이터만 확인할 수 있음



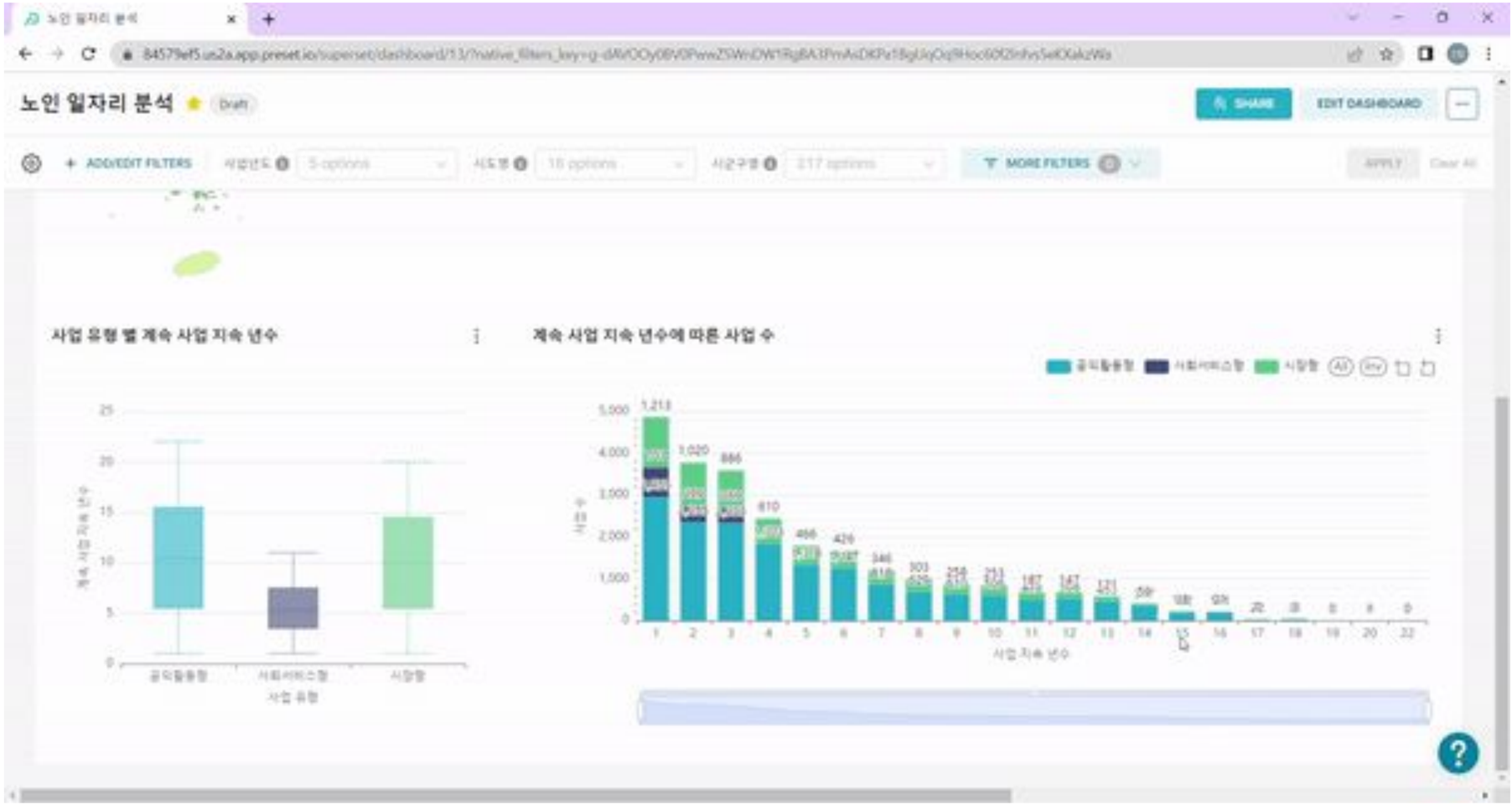
Interactive Chart 구현

2) Cross-Filtering 기능: 차트의 특정 영역을 선택하면 다른 차트의 데이터가 변경됨



Interactive Chart 구현

3) Data Zoom 기능: 차트의 특정 영역을 확대할 수 있음



대시보드 개선

기대 효과

A

차트간 연관성을
파악하기 쉬움

B

유의미한 데이터를
확인할 수 있는 차트
형식으로 변경

C

원하는 데이터만
필터링 할 수 있음

D

차트의 다양한 옵션을
추가하여 보다 많은
정보를 제공

Thank you!

“It is never too late to plan your life ahead.”