PROJECT 1 REPORT

1. Business Brief

Your project must start with a short business brief (max 1 page) including:

Objective: one-sentence project goal
→ In which US county is the safest for bee population in the context of beekeeping.

Stakeholders: who will use the results (e.g., finance team, policymakers)
→ US beekeepers, policy makers, agriculture

Key Metrics (KPIs): 2–3 relevant measures
→ What is the overall bee mortality rate per county (ratio per 100 bees)?
→ How has the spread of pests changed over time per county?
→ How has the spread of viruses changed over time per county?

Business Questions: at least 5 questions answerable with your model

1. Which viruses are yearly the most common per county?
2. In which county are bees least infected by Varroa mites?
3. Which county is most popular for beekeeping?
4. Which bee breed is the most common in a given county, and therefore indicates a county with a good environment for beekeeping?
5. In which county is Nosema (fungus) the least prevalent?

2. Datasets

Sources:
Two datasets from open data portals

- What viruses have spread to what type of bee - dataset
- Varroa and Nosema spread - US Bee Data - APHIS Honey Bee Survey: Data Download

→ These are biology datasets of very high quality; the data is consistent and well-structured.

3. Tooling

The following list is based on the main steps of the data engineering lifecycle and doesn't account for all "undercuts" discussed in the lectures.

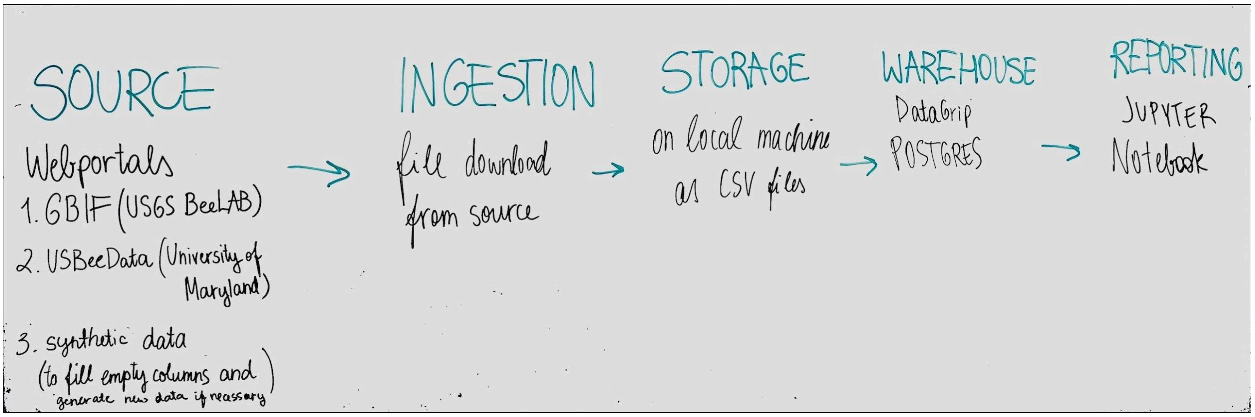
1. Since the generation of the data is out of our control, no tools can be specified for that.

2. Most likely Postgres would be used for storing the database since that is something we all have some sort of familiarity with. Although ClickHouse could also be an option once some more information about it has been covered during the lectures/practicals.
3. Ingestion could hopefully be automated with Airflow, but since the data updates very slowly, if at all, it could possibly be done manually with the help of scripts.
4. Transformation of the data could be done with dbt, since it is meant for that part of the data engineering process.
5. Serving of the data would be done by the same tool that handles storage.
6. For visualization or representing analysis results, Apache Supersert could be used alongside Python. Where the use of Python might be more useful in deeper analysis and visualising those results, compared to Apache Superset, which would provide a more constant overview of some metrics

4. Data Architecture

Provide a diagram showing:

Data flow (source → ingestion → storage → warehouse → reporting)


SOURCE
Webportals
1. GBIF (USGS BeeLAB)
2. USBeeData (University of Maryland)
3. synthetic data (to fill empty columns and generate new data if neassary)
INGESTION
file download from source
STORAGE
on local machine as CSV files
WAREHOUSE
DataGrip
POSTGRES
REPORTING
JUPYTER
Notebook

Ingestion method (API pull, file load, streaming, etc.)

→ CSV file download from source (GBIF, US Bee Data)

GBIF dataset citation:

When using this dataset please use the following citation and pay attention to the rights documented in rights.txt:

Droege S, Maffei C (2025). Insect Species Occurrence Data from Multiple Projects Worldwide with Focus on Bees and Wasps in North America. Version 1.27. United States Geological Survey. Sampling event dataset https://doi.org/10.15468/6autvb accessed via GBIF.org on 2025-09-26.

APHIS dataset by University of Maryland citation:

©2025 US Bee Data https://www.usbeedata.org/

Update frequency (hourly, daily, weekly, monthly?)
→ We are going to generate new data monthly

At least one example data quality check (e.g. null check, uniqueness check)

NULL CHECK

1. Organism, location and date have to be set
    a. Location has to have at least County
    b. Date has to have at least Month and Year
    c. Organism has to have at least Type, Name

CONSISTENCY CHECK/ VALID VALUE

1. Pest cannot have Bees scientific name

RANGE CHECK

1. Occurrence cannot be negative

UNIQUENESS

1. Fact table row uniqueness is defined by date, location, and organism keys

5. Data Model

Design a star schema with: o ≥ 1 fact table (state the grain clearly) o ≥ 3 dimension tables

→ Our database granularity is explained based on the defined null checks above.
<u>dim.Date</u>

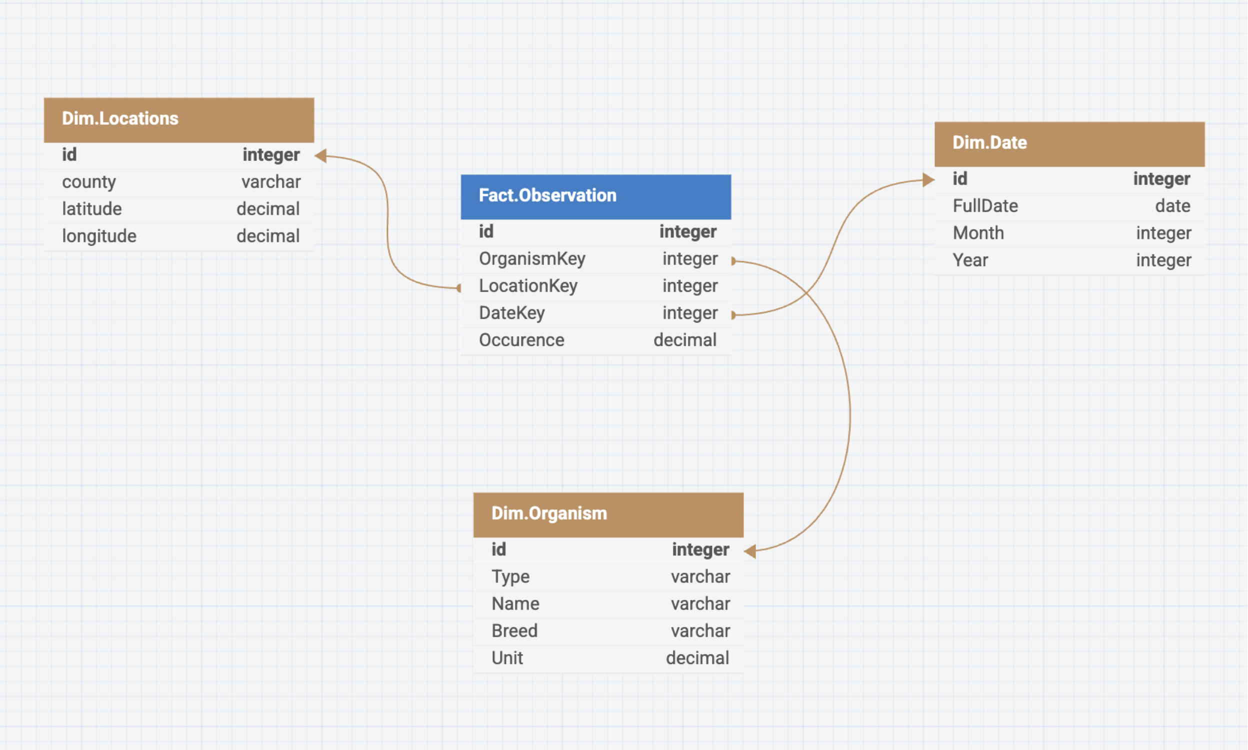
- Biologically, the distribution of specimens (bees, pests, viruses) becomes significant starting from the monthly level, since the changes result from seasonal weather variations. In addition, bees are observed on a hive basis, and the impact of pests and diseases only becomes apparent at the monthly level.

<u>dim.Location</u>

- A county is a sufficiently large area with similar natural conditions.

<u>dim.Organism</u>

- According to our data, the species level is sufficient for distinguishing bees, the genus level is sufficient for *Nosema* and *Varroa*, and the name level is sufficient for viruses. More detailed classification would not affect the results of our study.



Dim.Locations	
id	integer
county	varchar
latitude	decimal
longitude	decimal

Fact.Observation	
id	integer
OrganismKey	integer
LocationKey	integer
DateKey	integer
Occurence	decimal

Dim.Date	
id	integer
FullDate	date
Month	integer
Year	integer

Dim.Organism	
id	integer
Type	varchar
Name	varchar
Breed	varchar
Unit	decimal

For each dimension, justify your choice of Slowly Changing Dimension (SCD) type: o Type 1, Type 2, or Static (with reasoning)

→ The dataset follows SCD Type 0, because we analyze observations without personalization/individualization. Therefore, every update in the dataset is represented as a new row based on the timeline. However, when being very specific, human error may occur, and in such cases, some rows might need to be updated, which would correspond to SCD Type 1 in every dimension table.

ˇˇ

6. Data Dictionary

• Provide table & column descriptions, including data types.

- **Dim. Locations** - observation event location data based on US counties, i.e information pertaining to the location where the organism was observed.

Column name	Data type	Description
ID	integer	
county	varchar	County in which observation occurred
latitude	numeric	Observation location latitudinal coordinates
longitude	numeric	Observation location longitudinal coordinates
locationID	integer	Surrogate key???

- **Dim. Organism** - table for data relating to the organisms observed

Column name	Data type	Description
ID	integer	
type	varchar	Indicates what group the observed organism belongs to (bee, pest, virus)
name	varchar	Organism name
scientificName	varchar	Only relevant for bees, indicates breed name
unit	varchar	Value indicates what units the occurrence is measured in
organismID	integer	Surrogate key

- **Dim. Date -** observation event time (date) data

Column name	Type	Description
ID	integer	

fullDate	date	Observation event full date
year	integer	Observation event year number
month	integer	Observation event month number
dateID	integer	Surrogate key

- **Fact. Observations -** fact table combining together the aforementioned dimensions and adding the number of specific organism occurrences observed at a specific location on a specific date.

Column name	Type	Description
ID	integer	
Organism	foreign key	From dim. Organism
Location	foreign key	From dim. Location
Date	foreign key	From dim. date
Occurrence	numeric	How many organism occurrences were measured during the observation

7. Demo Queries

• Write SQL queries answering the business questions from your brief.

Which viruses are yearly the most common per county?

```
WITH virus_counts AS (
        SELECT
        loc.county,
        date.year,
        org.name AS virus,
        COUNT(obs.occurrence) AS total_occurrence

FROM Fact_Observation obs
JOIN Dim_Organism org ON org.id = obs.organismkey
JOIN Dim_Locations loc ON loc.id = obs.locationkey
JOIN Dim_Date date on date.id = obs.datekey
WHERE org.type = “virus”
GROUP BY loc.county, date.year, org.name
)
ranked AS (
SELECT *,
        DENSE_RANK() OVER(
        PARTITION BY county, year
        ORDER BY total_ocurrence desc
        ) AS rank
FROM virus_counts
)
SELECT county, year, virus, total_ocurrence
FROM ranked
WHERE rank = 1
ORDER BY county, year, virus
```

In which county are bees least infected by Varroa mites?

```
SELECT
        loc.county,
        AVG(obs.occurrence) AS average_varroa
FROM Fact_Observation obs
JOIN Dim_Organism org ON org.id = obs.organismkey
JOIN Dim_Locations loc ON loc.id = obs.locationkey
WHERE org.type “pest” AND org.name = “Varroa”
GROUP BY loc.county
ORDER BY average_varroa
```

Which county is most popular for beekeeping?

```
SELECT
        loc.county,
        COUNT(obs.occurrence) AS bee_count
FROM Fact_Observation obs
JOIN Dim_Organism org ON org.id = obs.organismkey
JOIN Dim_Locations loc ON loc.id = obs.locationkey
WHERE org.type “bee”
GROUP BY loc.county
ORDER BY bee_count DESC
```

Which bee breed is the most common in a given county, and therefore indicates the county has a good environment for that breed?

```
SELECT
        org.scientificName,
        sum(obs.occurrence) AS breed_total
FROM Fact_Observation obs
JOIN Dim_Organism org ON org.id = obs.organismkey
JOIN Dim_Locations loc ON loc.id = obs.locationkey
WHERE org.type “bee” AND loc.county = :county_name
GROUP BY org.scientificName
ORDER BY breed_total DESC
```

In which county is Nosema (fungus) the least prevalent?

```
SELECT
        loc.county,
        AVG(obs.occurrence) AS average_nosema
FROM Fact_Observation obs
JOIN Dim_Organism org ON org.id = obs.organismkey
JOIN Dim_Locations loc ON loc.id = obs.locationkey
WHERE org.type “pest” AND org.name = “Varroa”
GROUP BY loc.county
ORDER BY average_varroa
```

8. Include each group members’ roles and contribution (% per group member)

25% Annaliisa offline teamwork, additionally SQL queries
25% Victoria offline teamwork, additionally data dictionary
25% Selene offline teamwork, additionally transferred star schema with proper online tool
25% Remi offline teamwork, additionally specifying tools

9. GitHub link with README and any relevant DDL, DML or sample data

GitHub repo link: https://github.com/Lollover45/data-engineering-project

10. LLM disclosure (links to any AI/LLM chats used for this project)