CSCI 5408

DATA MANAGEMENT AND WAREHOUSING

Assignment - 1

Image Folder Link:
https://drive.google.com/drive/folders/1HfueL1SO3EMjy4uxPigycqawwNaIoDYQ?usp=sharing

GitLab Link: https://git.cs.dal.ca/jspatel/csci5408_s24_b00982253_jay_patel.git

Table of Contents

Problem 1

1.1 Identify 22 Entity Sets

- These entities are prepared from the websites given in the assignment [1] [2] [3].

Table 1: Entity sets with attribute explanation and justification.

Sr. No.	Entity	Attributes	Examples of some attributes	Reason
1.	County	● County_ID ● Name ● Email ● Helpline Number ● Office Location	Name - Halifax	Manage regional administrative information.
2.	Burn	● Burn_ID ● Ristriction Level ● Timing	Ristriction Level - Red, Yellow, and Green	Track burn restrictions and timings for each county.
3.	Resources	● Resource_ID ● Name ● Type		Catalog and manage natural resources available in the region.
4.	Park	● Park_ID ● Name ● Location ● Area ● Type ● Opening Hours	Location - 123 Park Ave, Halifax, Nova Scotia, B3H 1A1, Canada Area - 150 hectares Type - National Park, Provincial/State Park, or Historical Park	Manage information about parks, including their locations, areas, and types.
5.	Site	● Site_ID ● Site Number ● Location ● Capacity ● Rate ● Permissions ● Status ● Start Date ● End Date	Permissions - Allowed Equipment Capacity - Max number of people allowed at a time Site Number - Like 1, 2 in a park	Manage specific locations within parks, including their capacities, permissions, and booking statuses.

			Status - available or booked	
6.	Reservation	● Reservation_ID ● Date ● Time ● Status ● Payment Status	Status - confirmed, pending, canceled	To track bookings and their statuses for each site.
7.	User	● User_ID ● First Name ● Middle Name ● Last Name ● Height ● Eye Color ● DOB ● Age ● Mobile Number ● Email ● Password ● State ● Street Address ● Postal Code ● City ● Gender ● County		Personal information and credentials of individuals interacting with the system.
8.	Facility	● Facility_ID ● Name ● Type ● About	Name - Visitor Center, Picnic Area, Campground Type - Recreational, Accommodation, Dining	Manage various facilities available within parks.
9.	Job	● Job_ID ● Title ● Position ● Description ● Helpline Number ● Posting Date ● End Date ● Status		Manage job postings, descriptions, and application statuses within the park.
10.	Event	● Event_ID ● Name ● Description ● Date	Status - Ongoing, Completed, or canceled	Manage various events happening within the park.

		● Status		
11.	News	● News_ID ● Title ● Content ● Date Published ● Publisher Name ● Author Name ● Language ● Type	Type - Audio, Video, or Article	Information regarding news articles, their content, and publication details.
12.	Lake	● Lake_ID ● Name ● Location ● Surface Area ● Depth ● Water Quality	Water Quality - excellent, good, fair	Information about lakes, including their locations and water quality.
13.	Area	● Area_ID ● Area Number	Area Number - Recreation fishing area number.	More than one county groups together and form one recreational area.
14.	Monitoring Data	● Monitoring_ID ● Date ● Water Temperature ● PH Level ● Contaminants	Contaminants - detected in the lake water during monitoring.	Track environmental monitoring data, such as water quality in lakes.
15.	Natural Resources	● Resource_ID ● Type ● Location ● Area ● Status	Type - forest, mineral, wildlife habitat Status - protected, under management	Manage information on various natural resources, their statuses, and locations.
16.	Department	Department_ID Name Description Head of Department Email Department Code Phone State Street Address		Manage departmental details, including contact information and organizational structure.

		Postal Code City Established Date Number of Employees Annual Budget		
17.	Wildlife Species	● WildSpecies_ID ● Name ● Habitat ● Status	Habitat - where the species is typically found. Status - endangered, threatened, secure	Track information about different wildlife species and their conservation statuses.
18.	Forestry Management	● Forest_ID ● Management Plan ● Area Covered ● Activities ● Status	Status - ongoing, completed Activities - harvesting, reforestation	Manage forestry activities, areas covered, and project statuses.
19.	Mineral Resources	● Mineral_ID ● Type ● Location ● Quantity ● Extraction Methods	Type - oil, gas, gold, copper, iron Extraction Methods - open-pit mining, underground mining	Track information on mineral resources, including types, locations, and extraction methods.
20.	Publications	● Publication_ID ● Title ● Date Published ● Language ● Tags ● Size	Size - PDF Size	Manage publications, including their titles, dates, and file sizes.
21.	Courses	● Course_ID ● Course Name ● Discription ● Instructor Name ● Start Date ● End Date ● Fees ● Prerequisite Course ● Type	Type - Classroom or Home Study	Track educational courses, their details, and associated fees.
22.	Course_User	Course_ID User_ID		Manage the enrollment of users in various

				courses.
23.	Offices	● Office_ID ● Office Name ● Location ● Email ● Contact Number		Manage office locations and their contact details.
24.	Contact	● Contact_ID ● Email ● Phone ● State ● Street Address ● Postal Code ● City ● About ● Social Media Link ● Website ● Office Location	Type - Park, Lake, and Natural Resources	Store and manage contact details for all the resources present in Nova Scotia. (Park, Lake, and Natural Resources)
25.	Staff	● Staff_ID ● First Name ● Middle Name ● Last Name ● Salary ● Position ● Join Date ● Height ● Eye Color ● DOB ● Age ● Phone ● Email ● Password ● State ● Street Address ● Postal Code ● City ● Gender ● County		To manage personal and employment details of staff members within the department.

1.2 ERD_Initial

- This image shows the initial ERD model, which includes all the entities and their relationships with each other. This image may also contain some design issues that will be resolved in the next stage of the diagram.
- The flow starts from the Nova Scotia entity and then goes to the County, and so on.

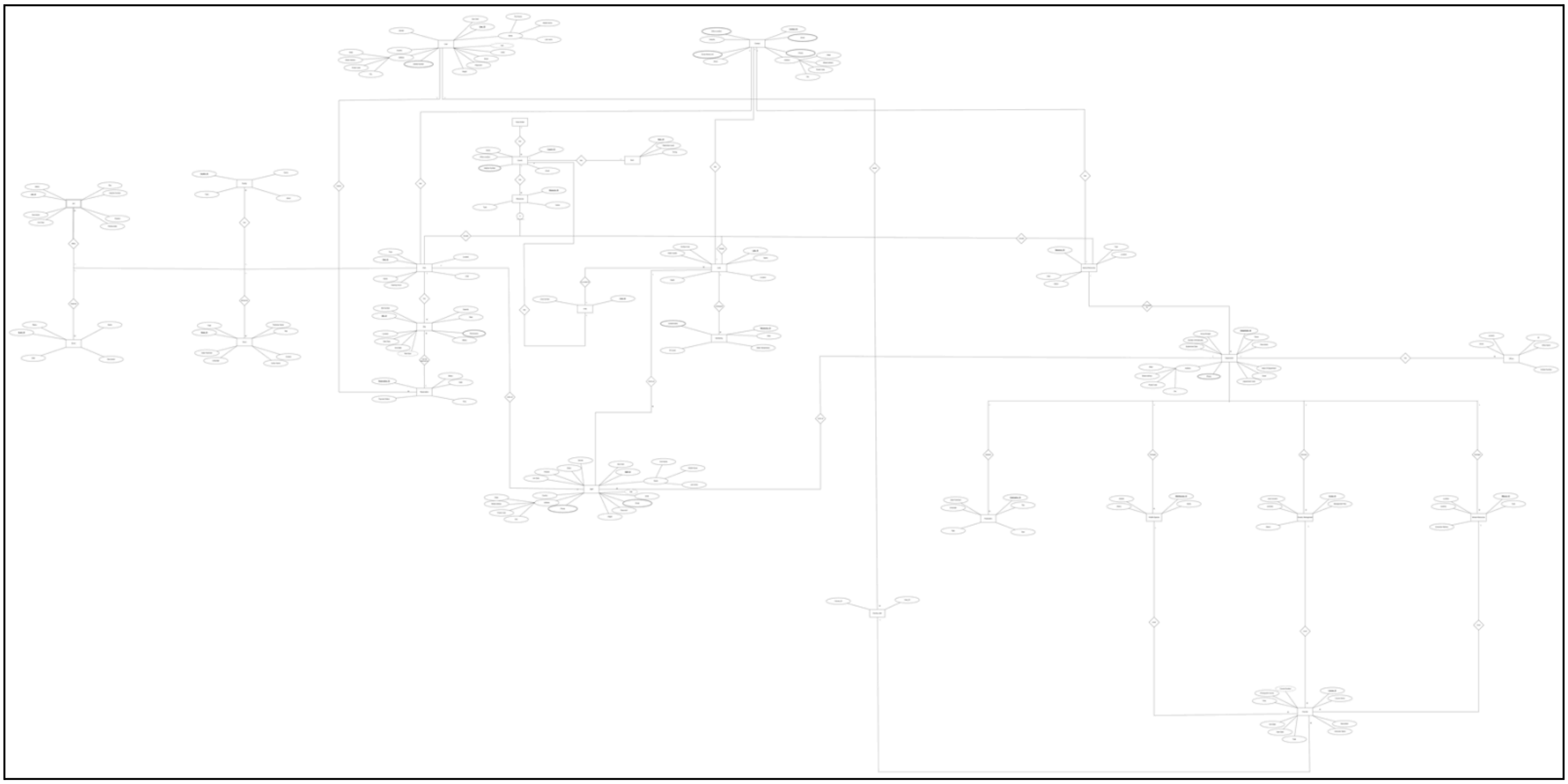


Figure 1.2: ERD_Initial [4]

1.3 Design Issues

Fan Trap

- In the initial ERD, there is a relationship where one area has many counties and one area has many lakes. However, this relationship does not allow us to conclude which lake is located in which county.
- This creates a fan trap, with M:1 and 1:M relationships. Therefore, we change these relationships to 1:M and then again 1:M.

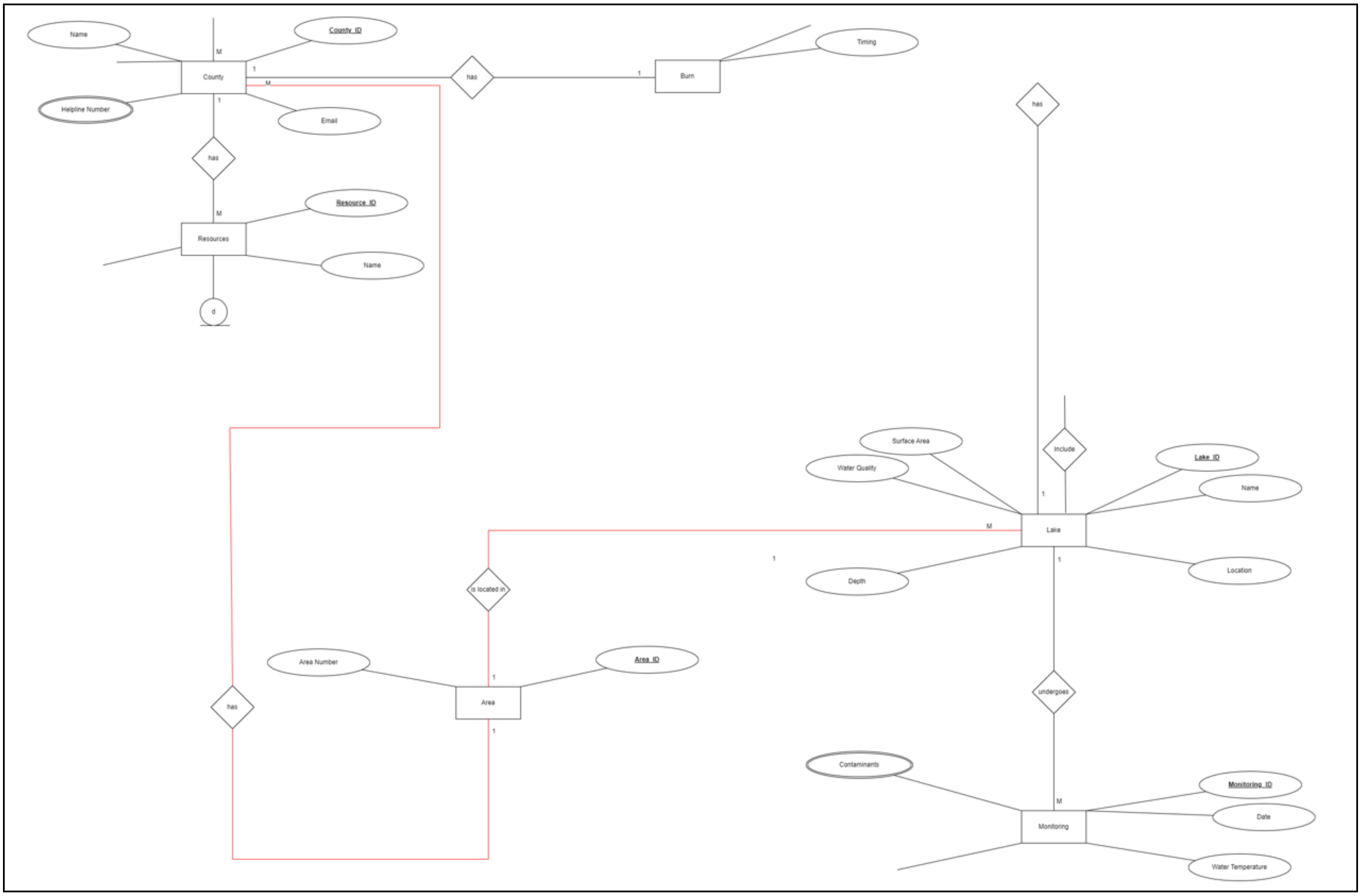


Figure 1.3.1: Fan Trap 1

- I solved this fan trap by rearranging the entities and their relationships.
- Now, the relationships are as follows: one area has many counties, and one county has many lakes.
- By doing this, we can determine that a particular lake is located in a specific county, which belongs to a specific area.

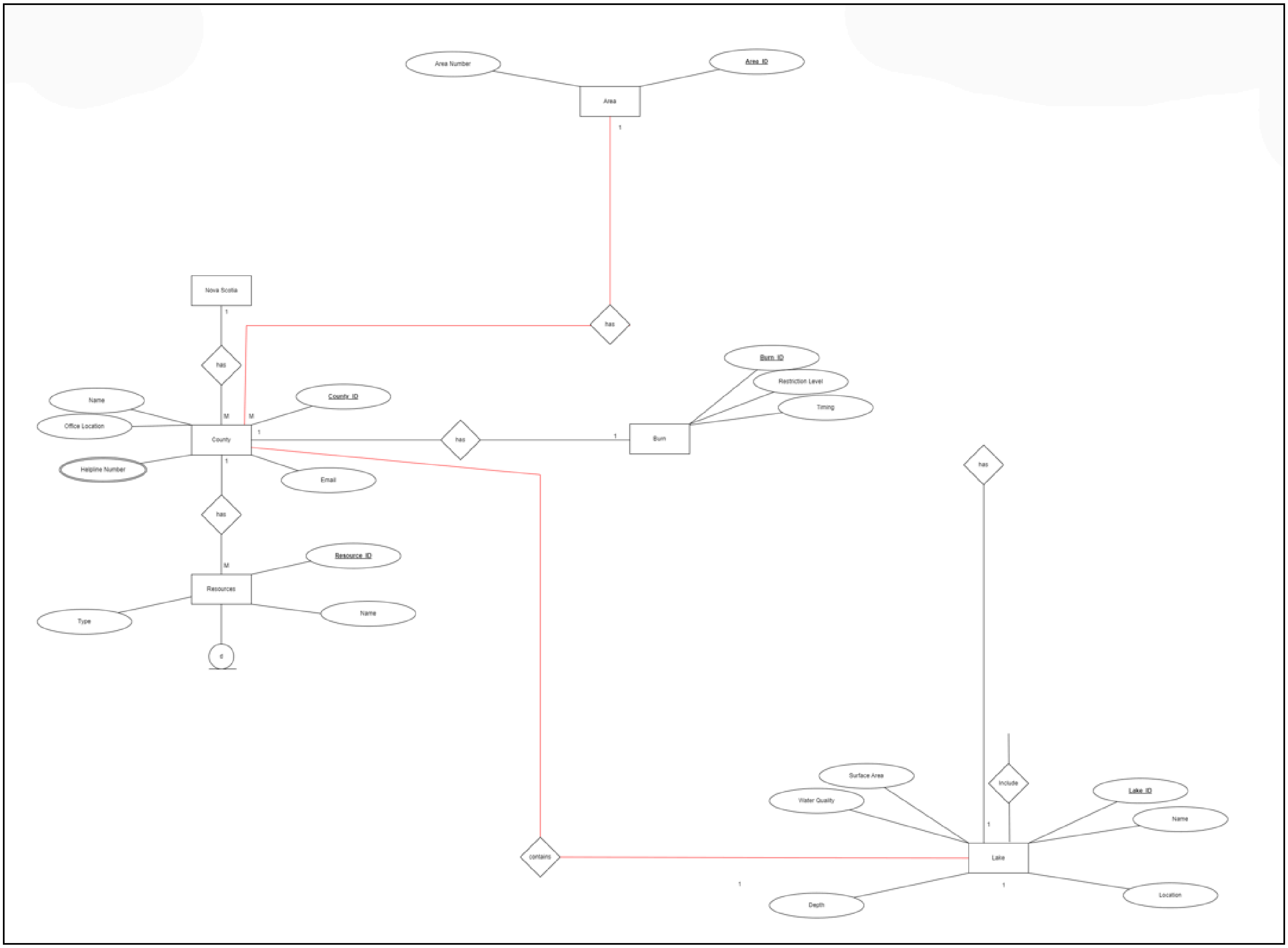


Figure 1.3.2: After solving fan trap 1

Fan Trap

- There is one more fan trap present, with relationships such as one department having many staff members and one department having many offices.
- With the relationships above, we cannot determine which staff member works in which office.

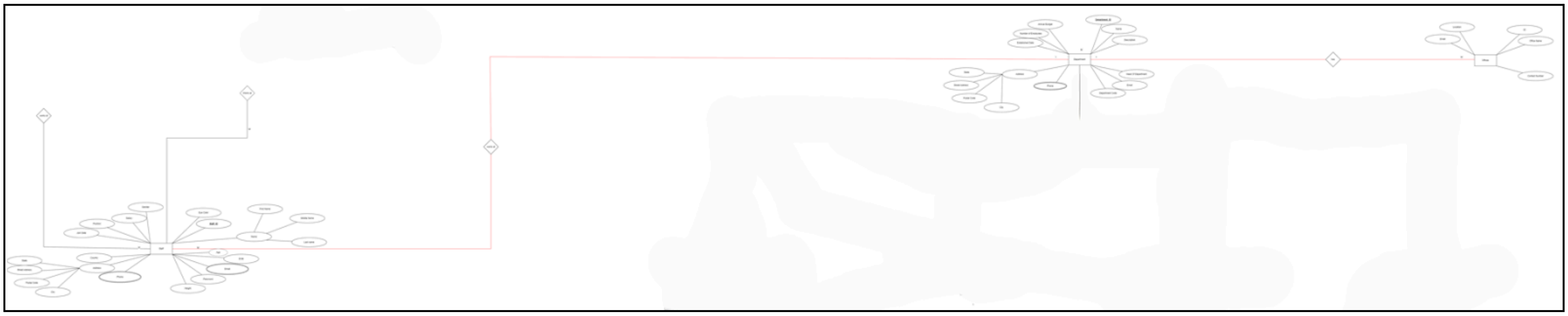


Figure 1.3.3: Fan Trap 2

- I solved this trap by changing the above relationships to one department has many staff members, and each staff member works in an office.

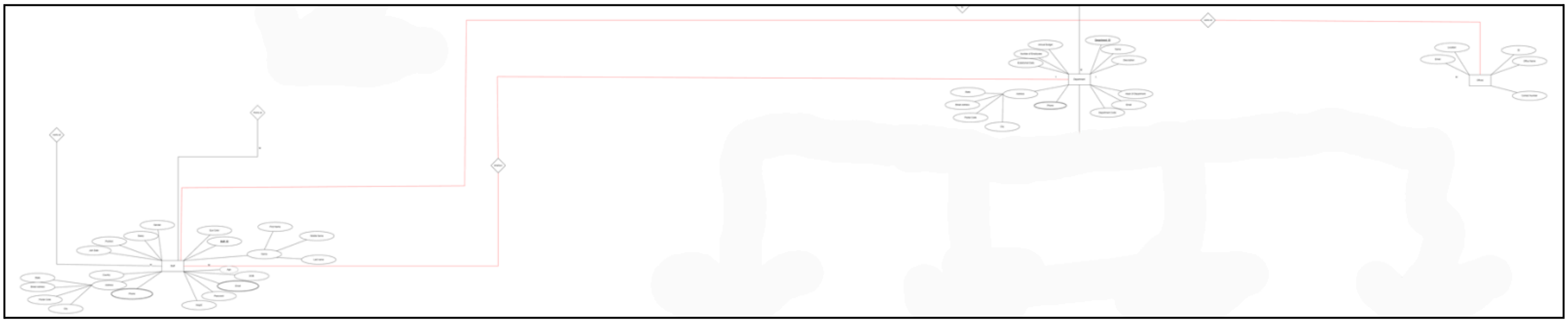


Figure 1.3.4: After solving fan trap 2

Chasm Trap

- I also found a chasm trap where we have a 1-to-M relationship followed by another 1-to-M relationship, but the crucial point is that there is a relationship between the first and third entities.
- We have a relationship where one department handles many resources, and resources offer many courses. If we don't have that resource still department must still offer a course, this situation is not handled in the current ERD model.

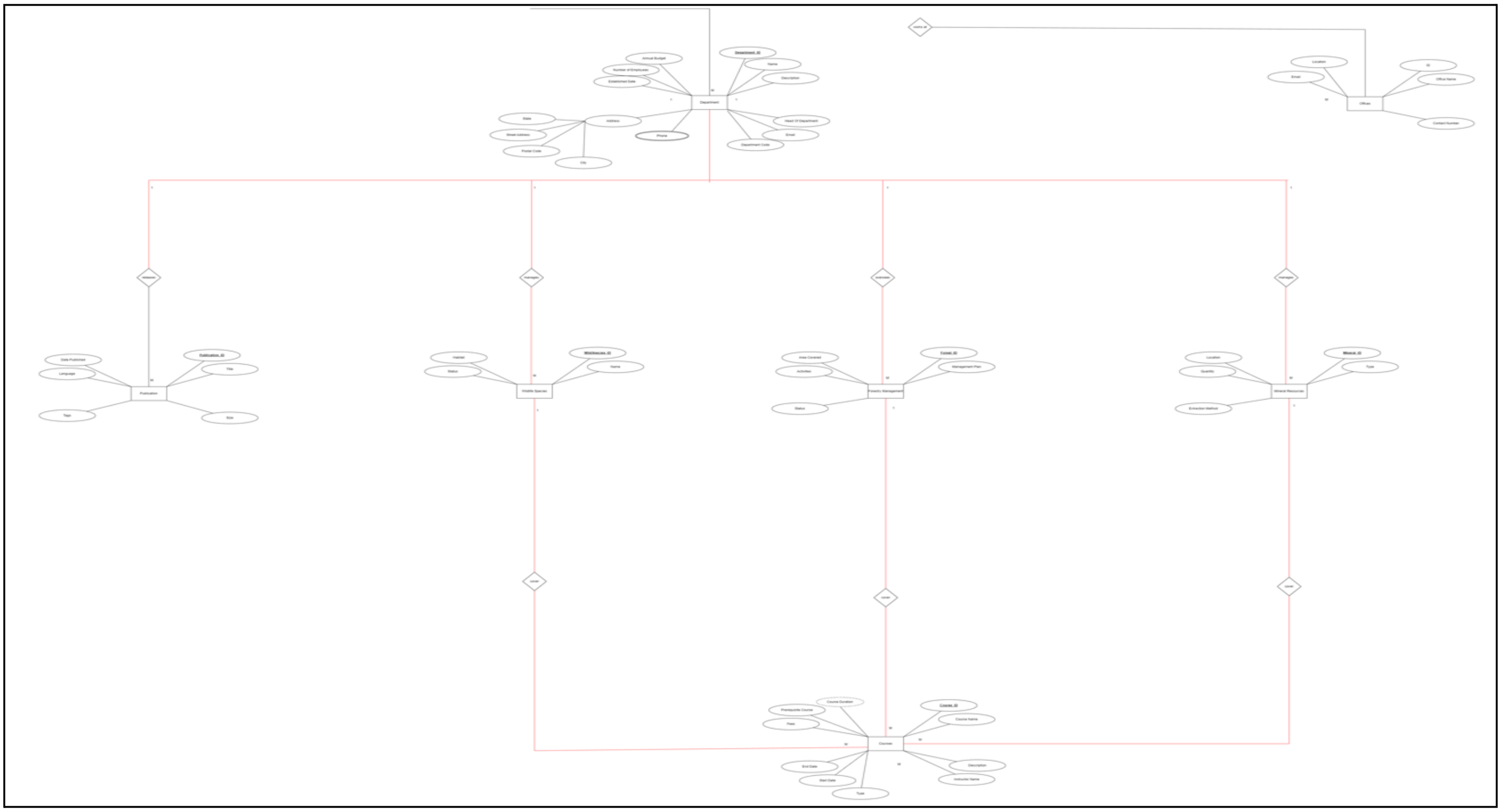


Figure 1.3.5: Chasm Trap

- I solved this chasm trap by adding a relationship between the department and the course. Now, if a resource is no longer available, we still have the course associated with that resource in our system.

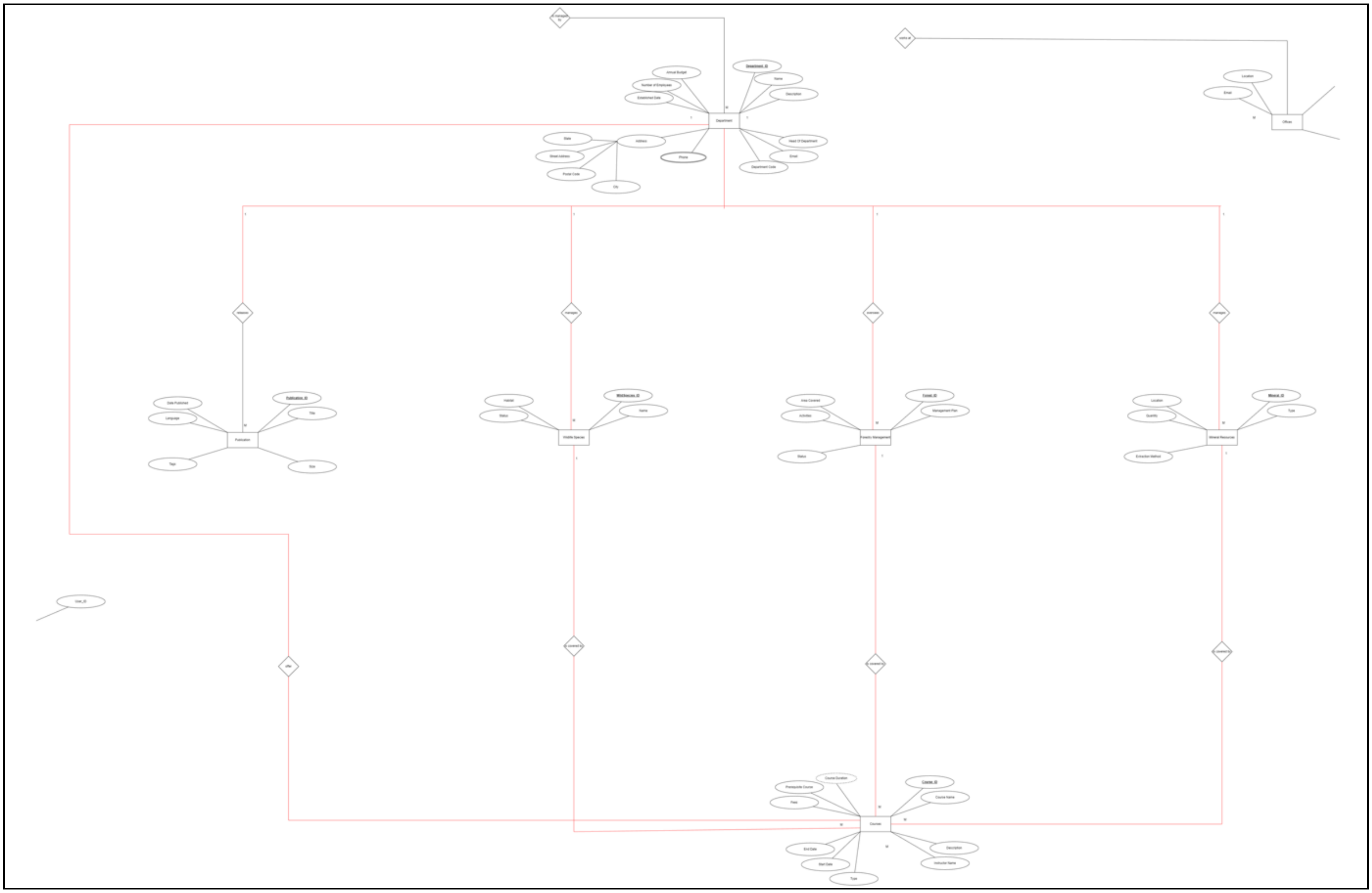


Figure 1.3.6: After solving the chasm trap

1.4 ERD_Final

- Here we have our final ERD by solving all the design issues that we have in our initial ERD model.

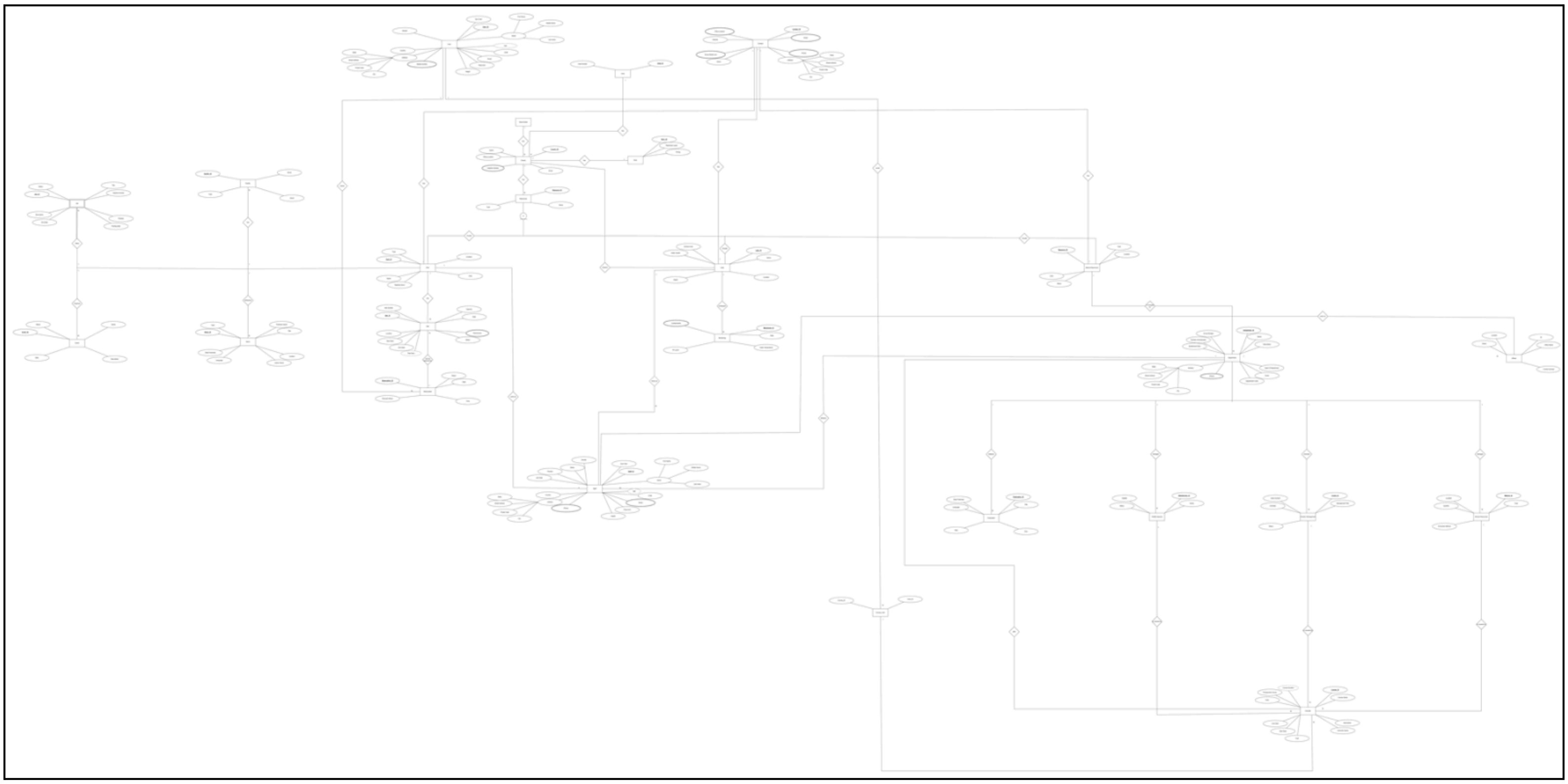


Figure 1.4: ERD_Final after solving all design issues

1.5 EERD_Final

Enhancing Course Entity

- We can extend our final ERD model by specifying the course entity. In the course entity, we have two types: classroom-based and home-study. By extending this, we can manage both types of courses more specifically, as each type has many different attributes that can now be handled by this extension.

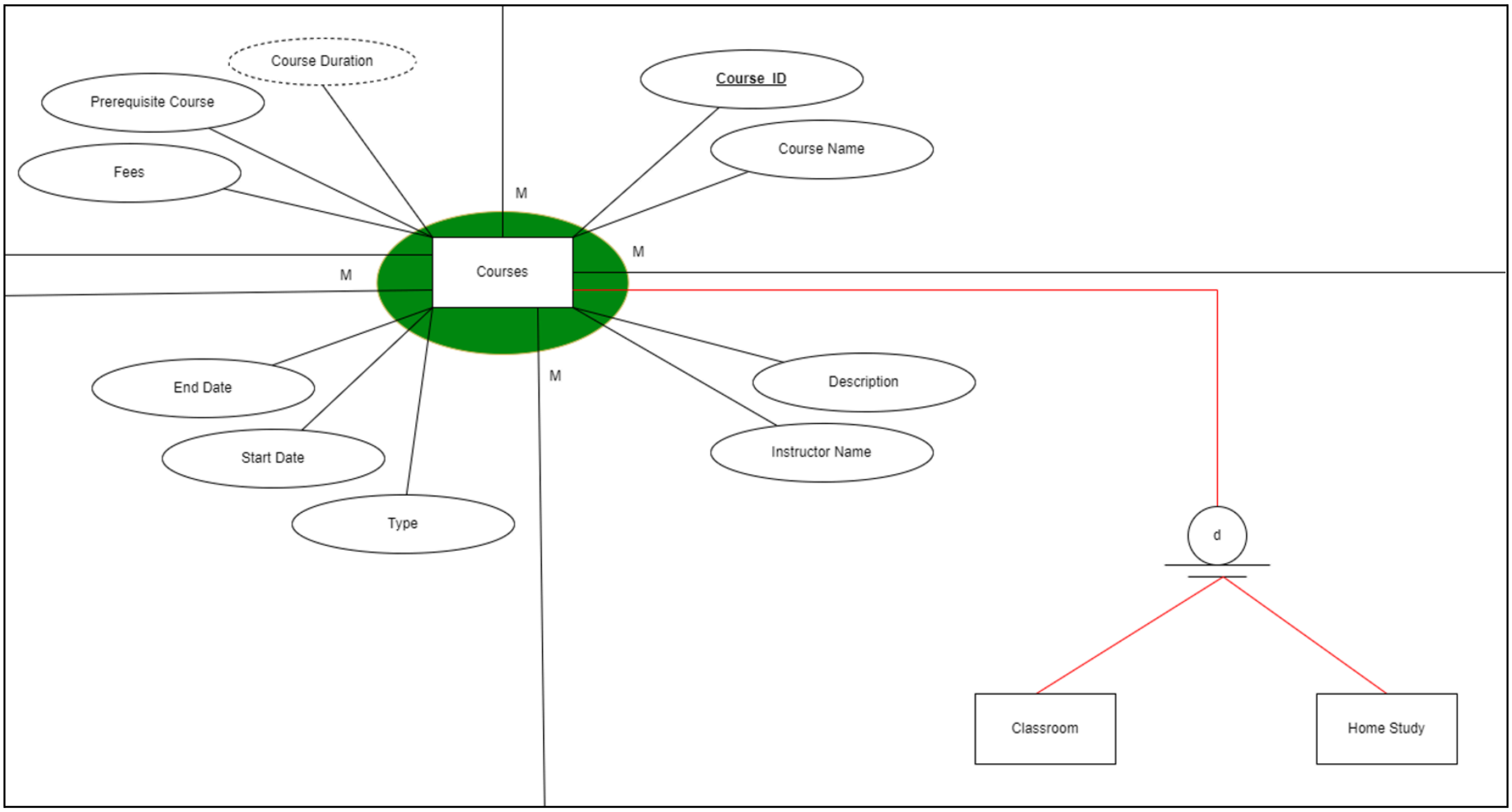


Figure 1.5.1: Enhancing course entity

Fianl_EERD

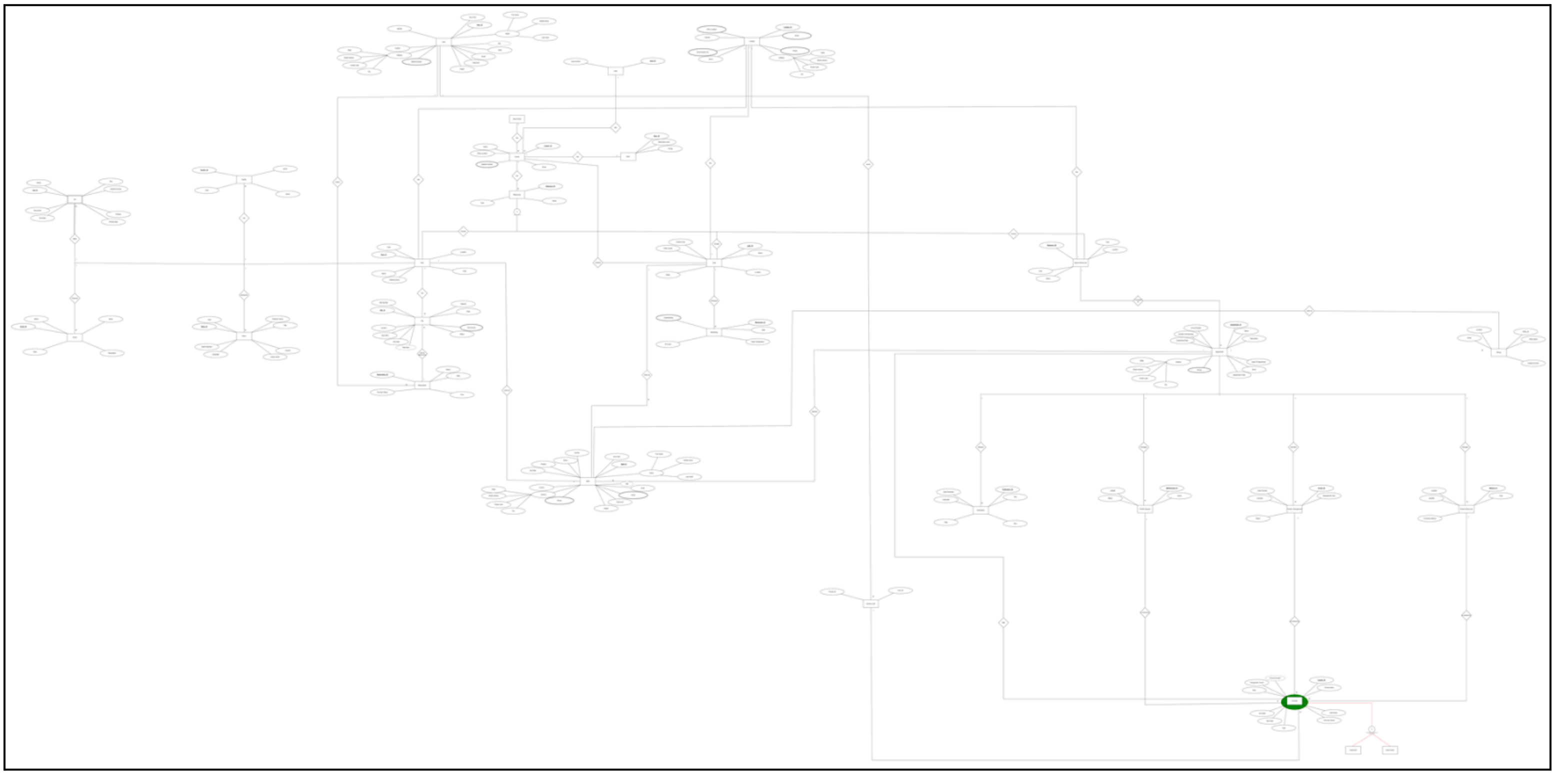


Figure 1.5.2: EERD_Final

Problem 2

2.1 Explanation of Fragmentation

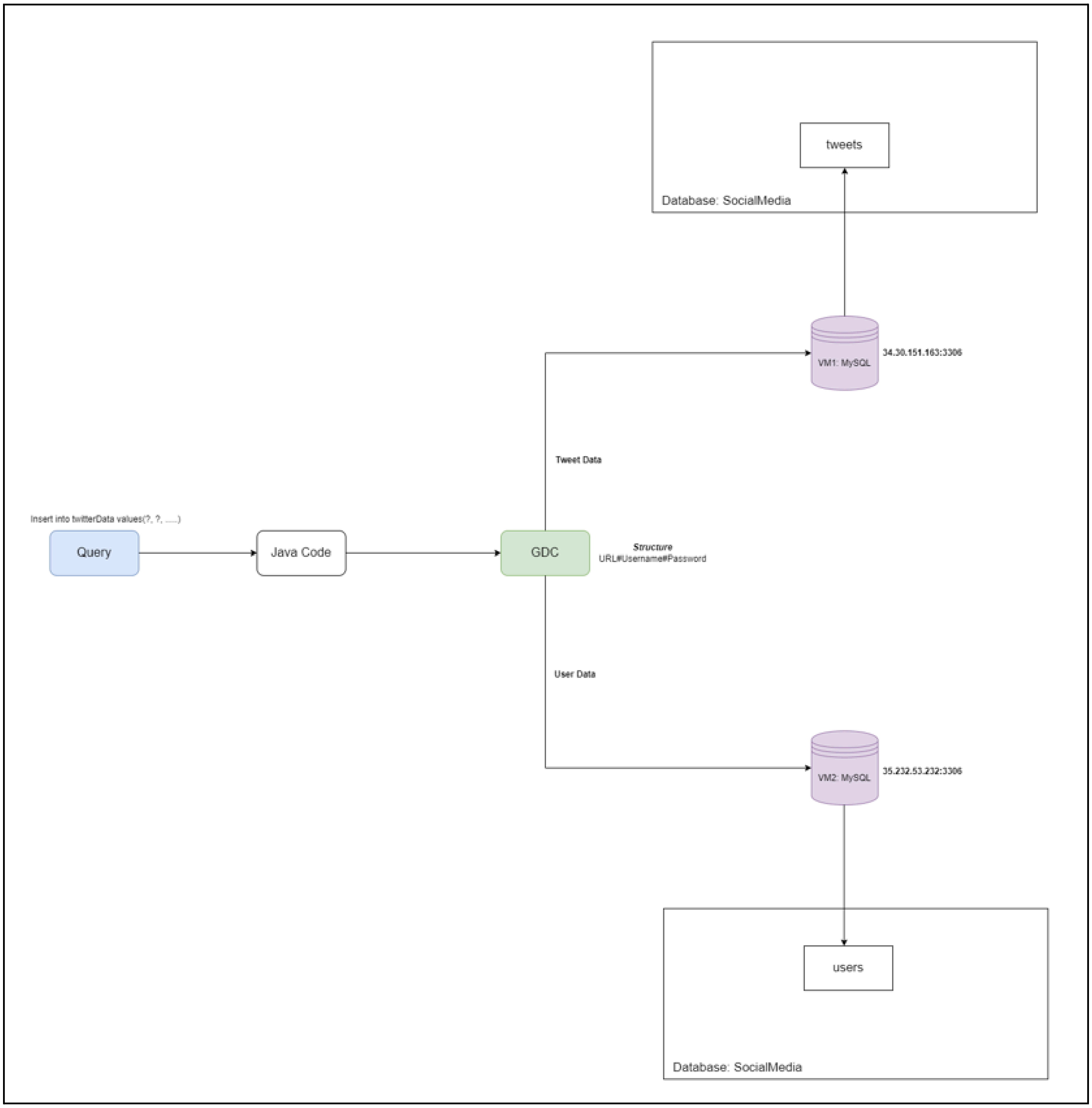


Figure 2.1: Fragmentation Structure [5]

- Here's an explanation of why I choose vertical fragmentation over horizontal or mixed fragmentation:

    - **Query Optimization:** Storing user information in each row leads to unnecessary storage, and fetching such data requires processing repeated information, which increases processing time and reduces system response time.

    - **Storage Utilization:** Horizontal fragmentation would lead to redundant storage of all user information for each tweet, whereas vertical fragmentation allows storing unique user data in one VM, optimizing storage usage.

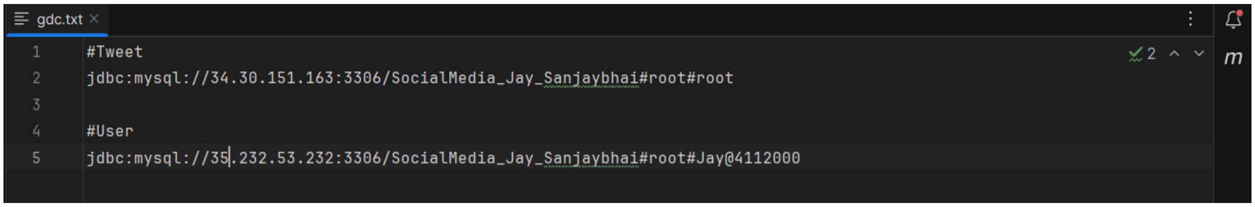
    - **Security Requirements:** Vertical fragmentation is chosen when certain columns contain sensitive information, as it allows for more granular access control and enhances data security.

    - **Data Distribution:** Vertical fragmentation reduces the amount of data transferred over the network because only specific columns needed for queries are transmitted. This is particularly advantageous in distributed systems.

2.2 GDC

GDC Structure

- Here is the structure of the GDC configuration for my Java program. I store the database URL, username, and password for both VMs. I use the '#' operator to distinguish which configuration is for which VM, ensuring information segregation for each specific VM.
- Each line contains the VM URL in the first position, the VM username in the second position, and the VM password in the third position.



```
#Tweet
jdbc:mysql://34.30.151.163:3306/SocialMedia_Jay_Sanjaybhai#root#root

#User
jdbc:mysql://35.232.53.232:3306/SocialMedia_Jay_Sanjaybhai#root#Jay@4112000
```

Figure 2.2: GDC File Structure

2.3 Java Code

- In the Java program, I have a total of 6 classes: DriverClass, Manager, InsertExcelData, DistributedTransaction, and two model classes, Tweet and User.
- DriverClass calls different functions as needed and handles the overall flow of the program.
- Manager contains code to read and configure the gdc.txt file, and also calls another function to read an Excel sheet and run distributed transactions.
- InsertExcelData imports data into our system; it has functionality to read CSV files and copy that data to the respective VM according to the gdc.txt file.
- DistributedTransaction is where the actual transaction code happens.
- Model Classes (Tweet and User): These classes help in storing and fetching data in the database.

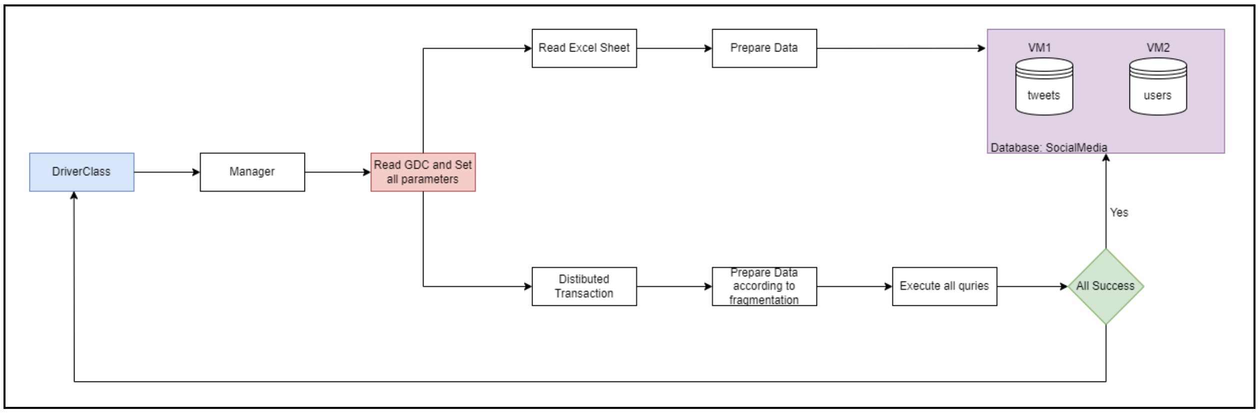
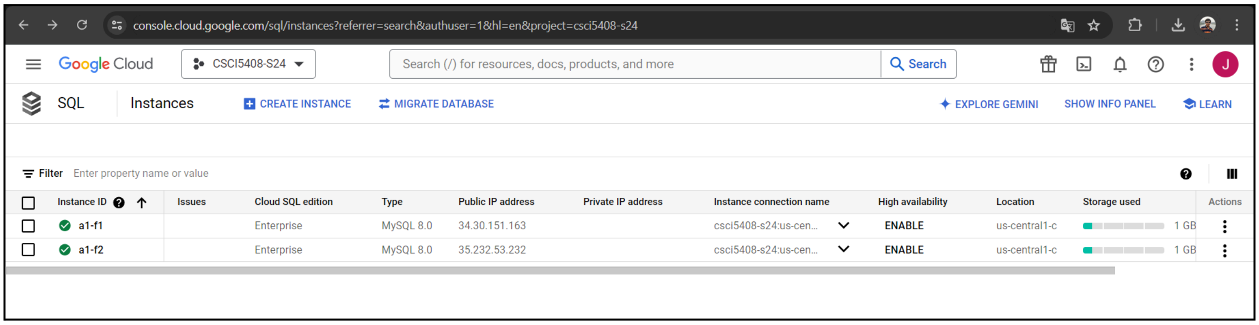


Figure 2.3: Flowchart of Java Program

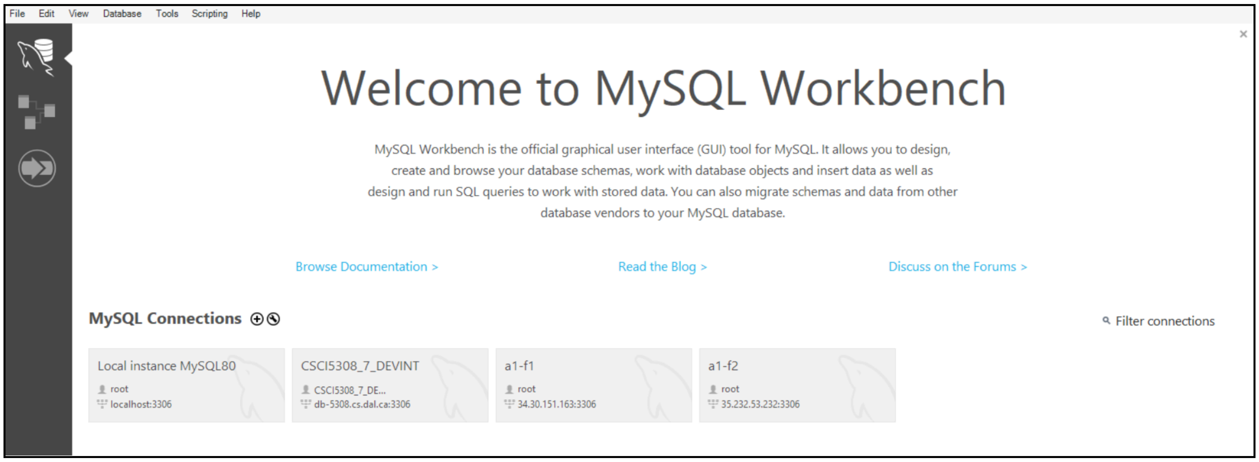
2.4 Creation of two Mysql instances on GCP

- Here I create two Mysql Instances on GCP and also connect them to MySQL workbench for easy access.
- MySQL Version - 8.0
- 1 VM
    - ➔ Instance ID: a1-f1 (assignment 1-fragment 1)
    - ➔ IP Address: 34.30.151.163
- 2 VM
    - ➔ Instance ID: a1-f2 (assignment 1-fragment 2)
    - ➔ IP Address: 35.232.53.232


console.cloud.google.com/sql/instances?referrer=search&authuser=1&hl=en&project=csci5408-s24
Google Cloud
CSCI5408-S24
Search (/) for resources, docs, products, and more
Search
SQL
Instances
CREATE INSTANCE
MIGRATE DATABASE
EXPLORE GEMINI
SHOW INFO PANEL
LEARN
Filter
Enter property name or value

□	Instance ID	Issues	Cloud SQL edition	Type	Public IP address	Private IP address	Instance connection name	High availability	Location	Storage used	Actions
□	✓ a1-f1		Enterprise	MySQL 8.0	34.30.151.163		csci5408-s24:us-cen...	ENABLE	us-central1-c	1 GB	⋮
□	✓ a1-f2		Enterprise	MySQL 8.0	35.232.53.232		csci5408-s24:us-cen...	ENABLE	us-central1-c	1 GB	⋮

Figure 2.4.1: Two Mysql instances on GCP [6]



Welcome to MySQL Workbench

MySQL Workbench is the official graphical user interface (GUI) tool for MySQL. It allows you to design, create and browse your database schemas, work with database objects and insert data as well as design and run SQL queries to work with stored data. You can also migrate schemas and data from other database vendors to your MySQL database.

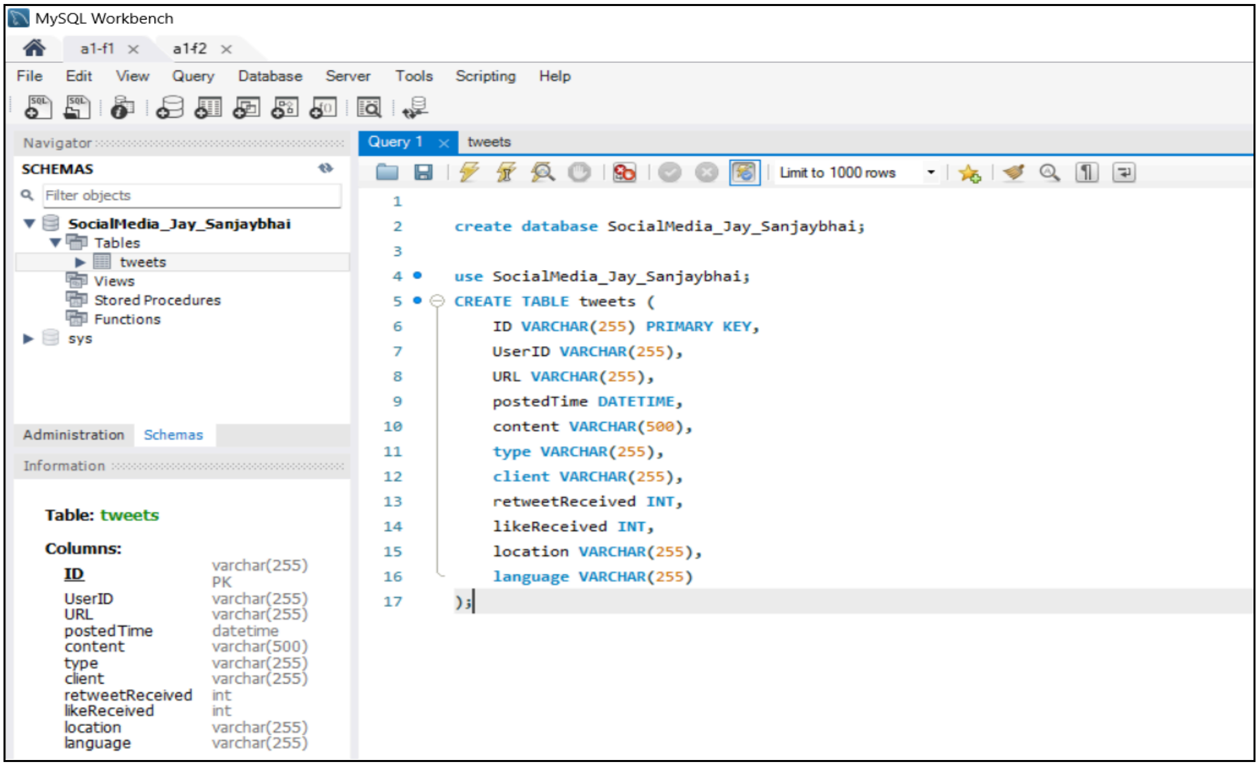
Local instance MySQL80
root
localhost:3306

CSCI5308_7_DEVINT
CSCI5308_7_DE...
db-5308.cs.dal.ca:3306

a1-f1
root
34.30.151.163:3306

a1-f2
root
35.232.53.232:3306

Figure 2.4.2: Connection of GCP Mysql Instance to workbench



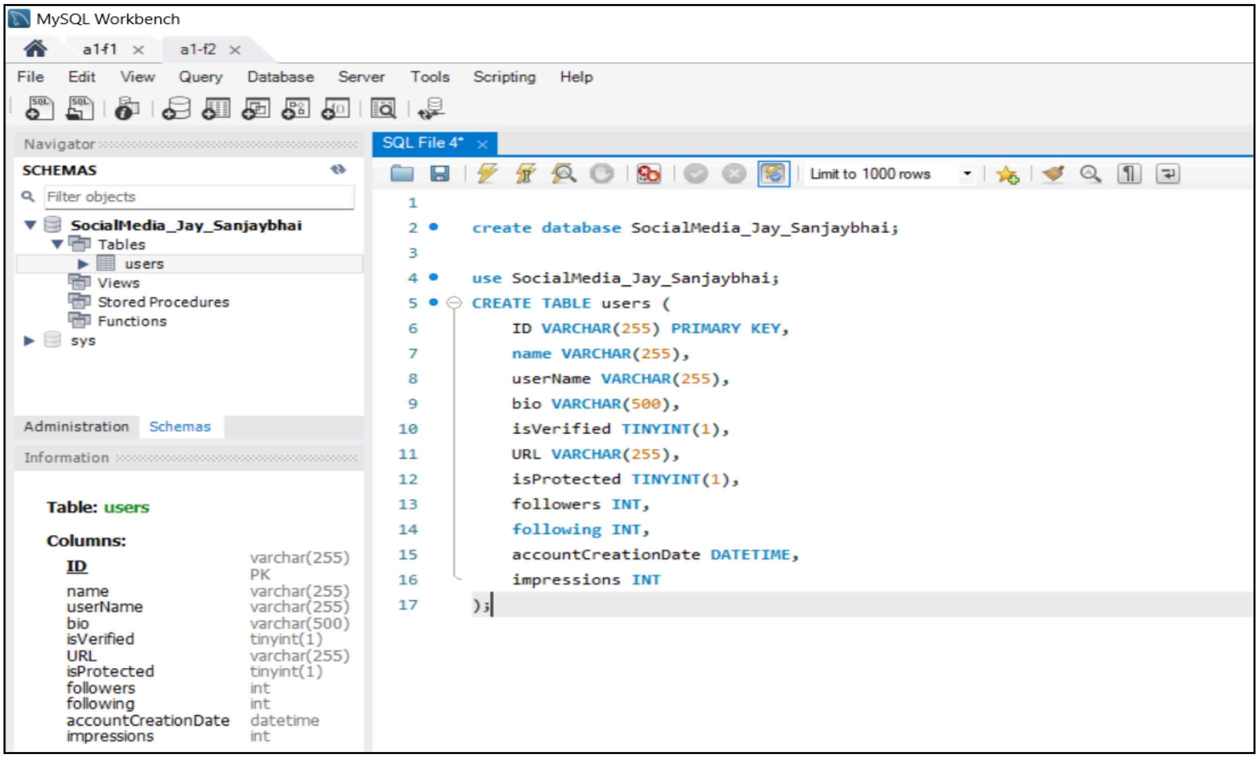
```sql

create database SocialMedia_Jay_Sanjaybhai;

use SocialMedia_Jay_Sanjaybhai;
CREATE TABLE tweets (
    ID VARCHAR(255) PRIMARY KEY,
    UserID VARCHAR(255),
    URL VARCHAR(255),
    postedTime DATETIME,
    content VARCHAR(500),
    type VARCHAR(255),
    client VARCHAR(255),
    retweetReceived INT,
    likeReceived INT,
    location VARCHAR(255),
    language VARCHAR(255)
);
```

Table: tweets

Columns:

ID	varchar(255) PK
UserID	varchar(255)
URL	varchar(255)
postedTime	datetime
content	varchar(500)
type	varchar(255)
client	varchar(255)
retweetReceived	int
likeReceived	int
location	varchar(255)
language	varchar(255)

Figure 2.4.3: Create a database in one VM and also create table tweets into that VM



```sql

create database SocialMedia_Jay_Sanjaybhai;

use SocialMedia_Jay_Sanjaybhai;
CREATE TABLE users (
    ID VARCHAR(255) PRIMARY KEY,
    name VARCHAR(255),
    userName VARCHAR(255),
    bio VARCHAR(500),
    isVerified TINYINT(1),
    URL VARCHAR(255),
    isProtected TINYINT(1),
    followers INT,
    following INT,
    accountCreationDate DATETIME,
    impressions INT
);
```

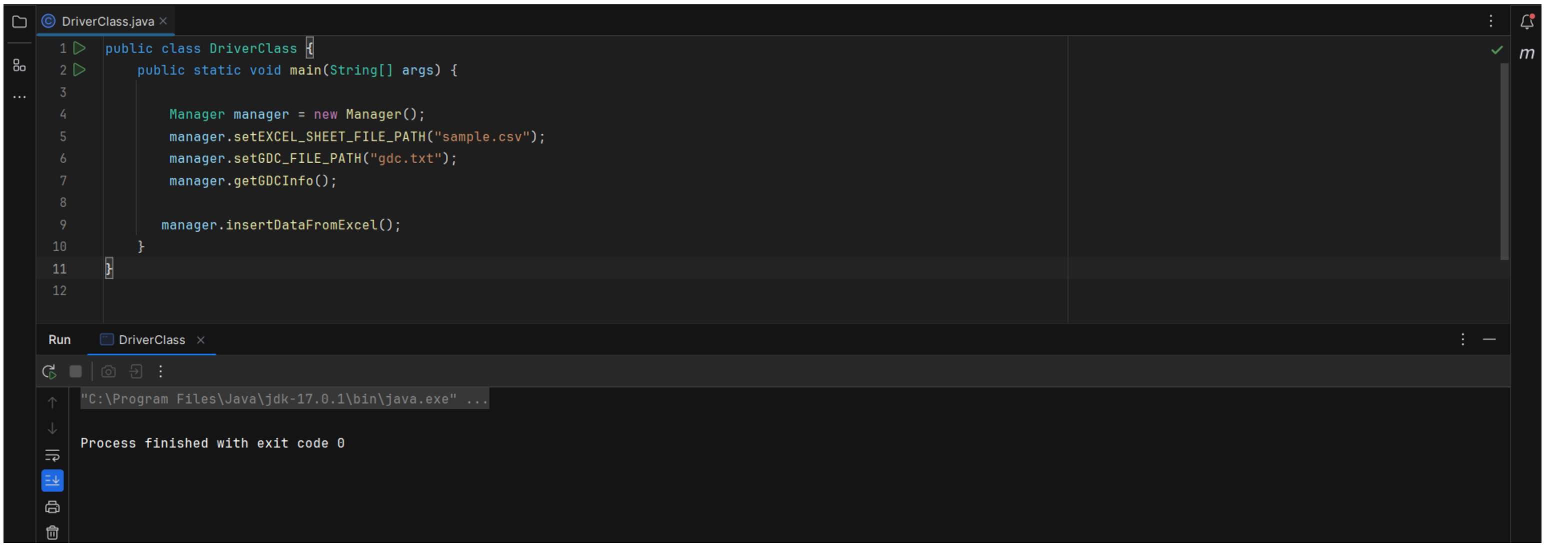
Table: users

Columns:

ID	varchar(255) PK
name	varchar(255)
userName	varchar(255)
bio	varchar(500)
isVerified	tinyint(1)
URL	varchar(255)
isProtected	tinyint(1)
followers	int
following	int
accountCreationDate	datetime
impressions	int

Figure 2.4.4: Create a database in one VM and also create table users in that VM

2.5 Insert Excel Sheet Data into VM



```java
public class DriverClass {
    public static void main(String[] args) {

        Manager manager = new Manager();
        manager.setEXCEL_SHEET_FILE_PATH("sample.csv");
        manager.setGDC_FILE_PATH("gdc.txt");
        manager.getGDCInfo();

       manager.insertDataFromExcel();
    }
}
```

```
"C:\Program Files\Java\jdk-17.0.1\bin\java.exe" ...

Process finished with exit code 0
```

Figure 2.5.1: Insert CSV file (sample.csv) into two MySQL VMs running on GCP

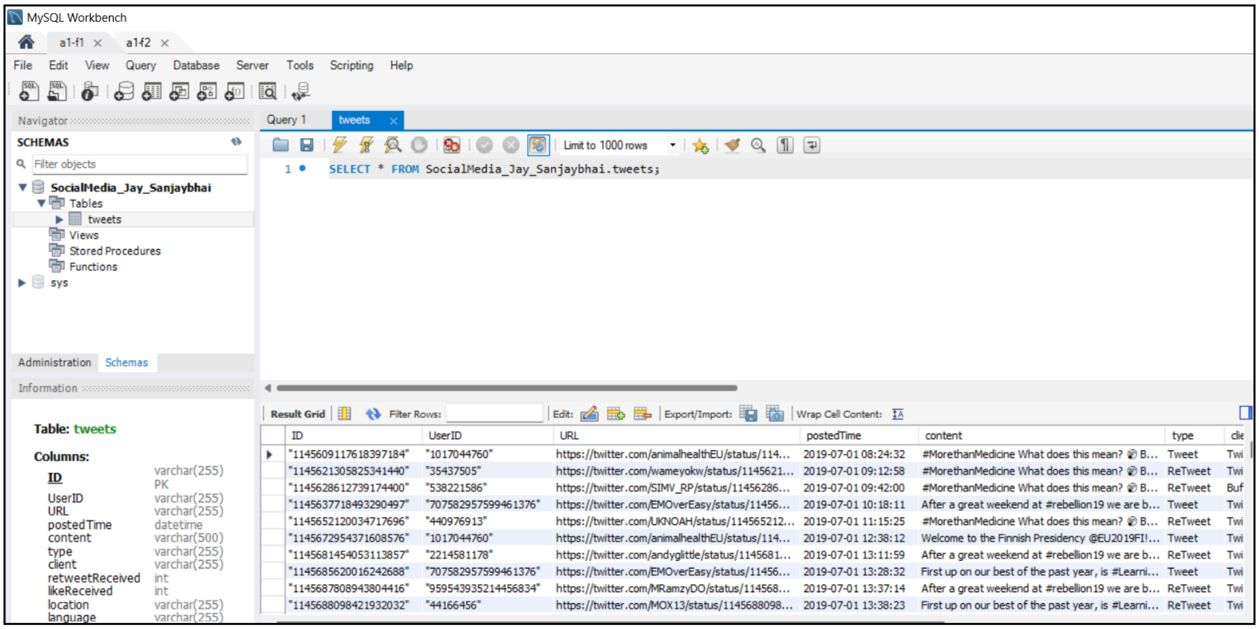


Table: tweets

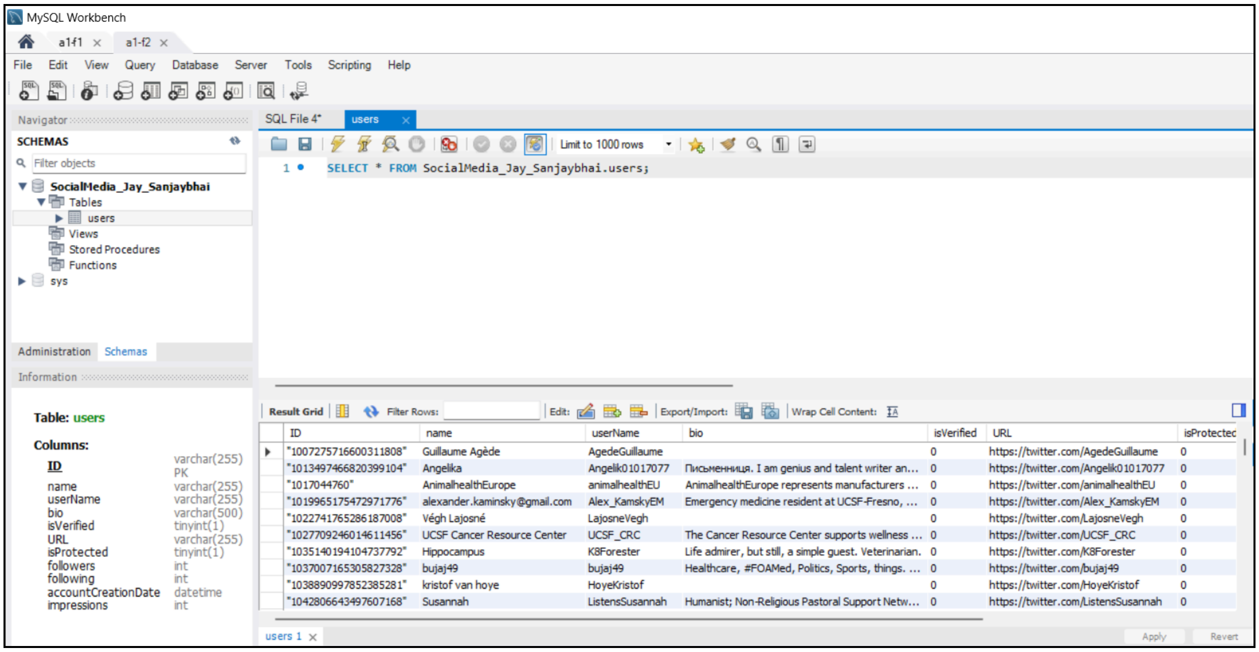
Columns:

ID	varchar(255) PK
UserID	varchar(255)
URL	varchar(255)
postedTime	datetime
content	varchar(500)
type	varchar(255)
client	varchar(255)
retweetReceived	int
likeReceived	int
location	varchar(255)
language	varchar(255)

```sql
SELECT * FROM SocialMedia_Jay_Sanjaybhai.tweets;
```

ID	UserID	URL	postedTime	content	type	clie
"1145609117618397184"	"1017044760"	https://twitter.com/animalhealthEU/status/114...	2019-07-01 08:24:32	#MorethanMedicine What does this mean? 🌍 B...	Tweet	Twi
"1145621305825341440"	"35437505"	https://twitter.com/wameyokw/status/1145621...	2019-07-01 09:12:58	#MorethanMedicine What does this mean? 🌍 B...	ReTweet	Twi
"1145628612739174400"	"538221586"	https://twitter.com/SIMV_RP/status/11456286...	2019-07-01 09:42:00	#MorethanMedicine What does this mean? 🌍 B...	ReTweet	Buf
"1145637718493290497"	"707582957599461376"	https://twitter.com/EMOverEasy/status/11456...	2019-07-01 10:18:11	After a great weekend at #rebellion19 we are b...	Tweet	Twi
"1145652120034717696"	"440976913"	https://twitter.com/UKNOAH/status/114565212...	2019-07-01 11:15:25	#MorethanMedicine What does this mean? 🌍 B...	ReTweet	Twi
"1145672954371608576"	"1017044760"	https://twitter.com/animalhealthEU/status/114...	2019-07-01 12:38:12	Welcome to the Finnish Presidency @EU2019FI!...	Tweet	Twi
"1145681454053113857"	"2214581178"	https://twitter.com/andyglittle/status/1145681...	2019-07-01 13:11:59	After a great weekend at #rebellion19 we are b...	ReTweet	Twi
"1145685620016242688"	"707582957599461376"	https://twitter.com/EMOverEasy/status/11456...	2019-07-01 13:28:32	First up on our best of the past year, is #Learni...	Tweet	Twi
"1145687808943804416"	"959543935214456834"	https://twitter.com/MRamzyDO/status/114568...	2019-07-01 13:37:14	After a great weekend at #rebellion19 we are b...	ReTweet	Twi
"1145688098421932032"	"44166456"	https://twitter.com/MOX13/status/1145688098...	2019-07-01 13:38:23	First up on our best of the past year, is #Learni...	ReTweet	Twi

Figure 2.5.2: Show tweets table data after data insertion.



```sql
SELECT * FROM SocialMedia_Jay_Sanjaybhai.users;
```

Table: users

Columns:

ID	varchar(255) PK
name	varchar(255)
userName	varchar(255)
bio	varchar(500)
isVerified	tinyint(1)
URL	varchar(255)
isProtected	tinyint(1)
followers	int
following	int
accountCreationDate	datetime
impressions	int

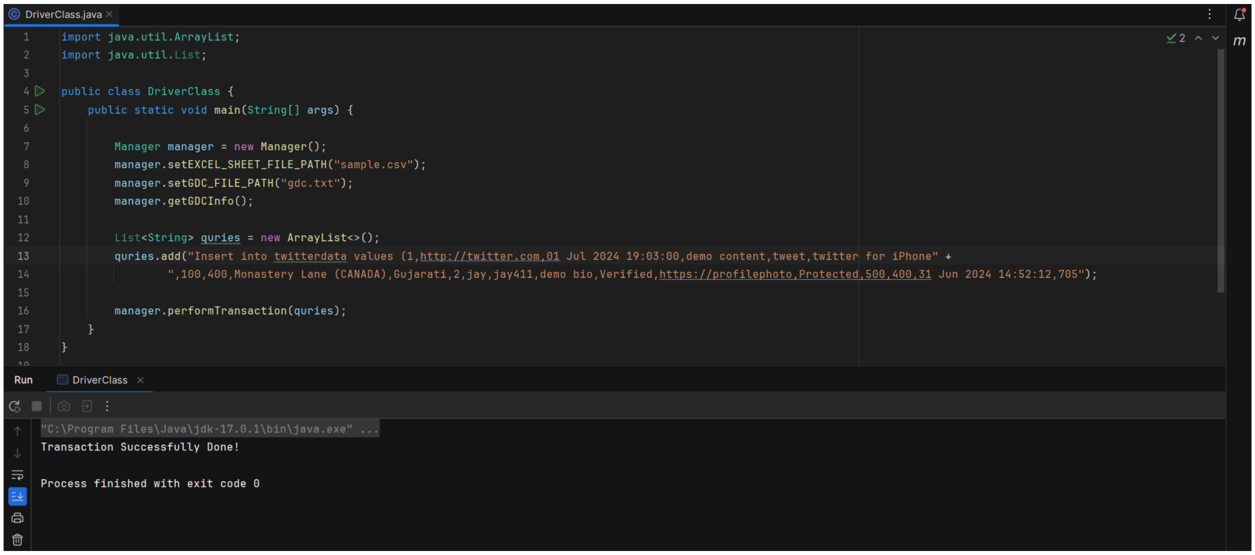
ID	name	userName	bio	isVerified	URL	isProtected
"1007275716600311808"	Guillaume Agède	AgedeGuillaume		0	https://twitter.com/AgedeGuillaume	0
"1013497466820399104"	Angelika	Angelik01017077	Письменниця. I am genius and talent writer an...	0	https://twitter.com/Angelik01017077	0
"1017044760"	AnimalhealthEurope	animalhealthEU	AnimalhealthEurope represents manufacturers ...	0	https://twitter.com/animalhealthEU	0
"1019965175472971776"	alexander.kaminsky@gmail.com	Alex_KamskyEM	Emergency medicine resident at UCSF-Fresno, ...	0	https://twitter.com/Alex_KamskyEM	0
"1022741765286187008"	Végh Lajosné	LajosneVegh		0	https://twitter.com/LajosneVegh	0
"1027709246014611456"	UCSF Cancer Resource Center	UCSF_CRC	The Cancer Resource Center supports wellness ...	0	https://twitter.com/UCSF_CRC	0
"1035140194104737792"	Hippocampus	K8Forester	Life admirer, but still, a simple guest. Veterinarian.	0	https://twitter.com/K8Forester	0
"1037007165305827328"	bujaj49	bujaj49	Healthcare, #FOAMed, Politics, Sports, things. ...	0	https://twitter.com/bujaj49	0
"1038890997852385281"	kristof van hoye	HoyeKristof		0	https://twitter.com/HoyeKristof	0
"1042806643497607168"	Susannah	ListensSusannah	Humanist; Non-Religious Pastoral Support Netw...	0	https://twitter.com/ListensSusannah	0

Figure 2.5.3: Show users table data after data insertion.

2.6 Insert Query

Insert into twitterdata values (1,http://twitter.com,01 Jul 2024 19:03:00,demo content,tweet,twitter for iPhone,100,400,Monastery Lane (CANADA),Gujarati,2,jay,jay411,demo bio,Verified,https://profilephoto,Protected,500,400,31 Jun 2024 14:52:12,705);

Note: Here we set tweetID as 1 and userID as 2

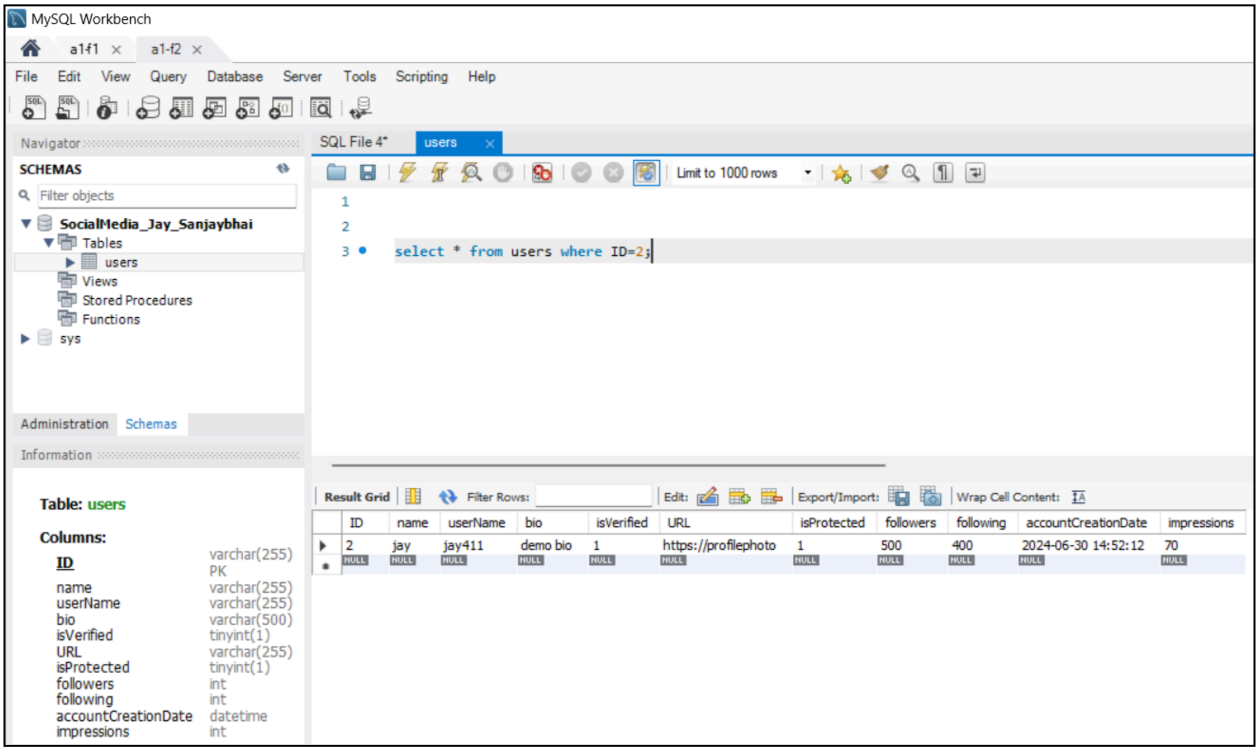


```java
import java.util.ArrayList;
import java.util.List;

public class DriverClass {
    public static void main(String[] args) {

        Manager manager = new Manager();
        manager.setEXCEL_SHEET_FILE_PATH("sample.csv");
        manager.setGDC_FILE_PATH("gdc.txt");
        manager.getGDCInfo();

        List<String> quries = new ArrayList<>();
        quries.add("Insert into twitterdata values (1,http://twitter.com,01 Jul 2024 19:03:00,demo content,tweet,twitter for iPhone" +
                ",100,400,Monastery Lane (CANADA),Gujarati,2,jay,jay411,demo bio,Verified,https://profilephoto,Protected,500,400,31 Jun 2024 14:52:12,705");

        manager.performTransaction(quries);
    }
}
```

```
"C:\Program Files\Java\jdk-17.0.1\bin\java.exe" ...
Transaction Successfully Done!

Process finished with exit code 0
```

Figure 2.6: Perform Insert Transaction

2.7 Testing



```
select * from users where ID=2;
```

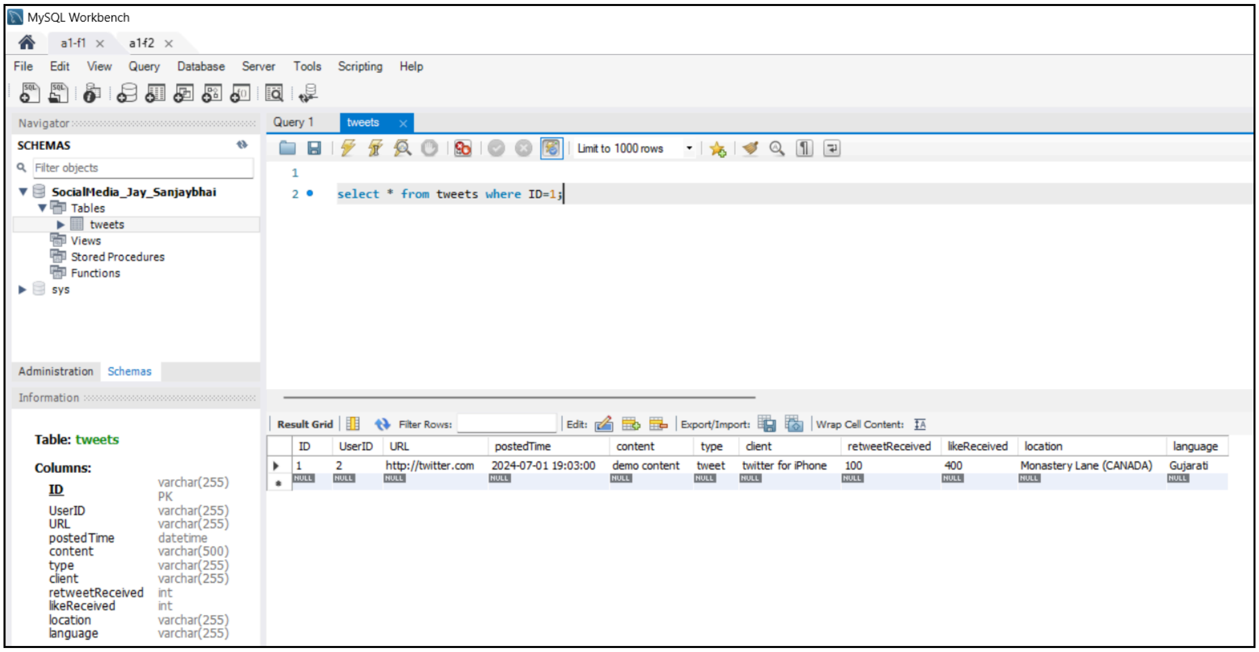
ID	name	userName	bio	isVerified	URL	isProtected	followers	following	accountCreationDate	impressions
2	jay	jay411	demo bio	1	https://profilephoto	1	500	400	2024-06-30 14:52:12	70
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Table: users

Columns:

ID	varchar(255) PK
name	varchar(255)
userName	varchar(255)
bio	varchar(500)
isVerified	tinyint(1)
URL	varchar(255)
isProtected	tinyint(1)
followers	int
following	int
accountCreationDate	datetime
impressions	int

Figure 2.7.1: Test user table after a distributed transaction is performed.



```
select * from tweets where ID=1;
```

ID	UserID	URL	postedTime	content	type	client	retweetReceived	likeReceived	location	language
1	2	http://twitter.com	2024-07-01 19:03:00	demo content	tweet	twitter for iPhone	100	400	Monastery Lane (CANADA)	Gujarati
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Table: tweets

Columns:

ID	varchar(255) PK
UserID	varchar(255)
URL	varchar(255)
postedTime	datetime
content	varchar(500)
type	varchar(255)
client	varchar(255)
retweetReceived	int
likeReceived	int
location	varchar(255)
language	varchar(255)

Figure 2.7.2: Test tweet table after a distributed transaction is performed.

References

[1] "Nova Scotia Parks", *novascotia.ca* [Online]. Available: https://parks.novascotia.ca/. [Accessed: Jul 1st, 2024].

[2] "Our Lakes - Government of Nova Scotia, Canada", *novascotia.ca* [Online]. Available: https://novascotia.ca/fish/sportfishing/our-lakes/. [Accessed: Jul 1st, 2024].

[3] "Department of Natural Resources and Renewables", *novascotia.ca* [Online]. Available: https://novascotia.ca/natr/. [Accessed: Jul 1st, 2024].

[4] "Flowchart Maker; Online Diagram Software", draw.io [online]. Available: https://app.diagrams.net/. [Accessed: Jul 1st, 2024].

[5] "Recovery of distributed servers and clusters in multiple locations", IBM [Online]. Available: https://www.ibm.com/docs/en/samfess/8.2.0?topic=recovery-distributed-servers-clusters-in-multiple-locations. [Accessed: Jul 1st, 2024].

[6] "Google Cloud console", Google Cloud [online]. Available: https://console.cloud.google.com/welcome/new?hl=en&authuser=1&project=csci5408-s24. [Accessed: Jul 1st, 2024].