Demographically -Enhanced Movie Recommendation System for Personalized Book Suggestion in Big Data

Team Medusa

Priya Varahan
Priyanka Akula
Pooja Manjunatha
Shreya Chikatmarla
Nandini Sreekumaran Nair

Introduction

- Massive amount of data and material have made personalized recommendations more important than ever in today's world.
- We're trying something new with our project: we're using big data to give personalized suggestions for both books and movies.
- The project is based on AWS's stable infrastructure, which provides a solid base for users to have engaging and flexible experiences.
- Using AWS and big data to make personalized suggestions is possible, which takes into account each person's tastes and makes users more interested.
- The project wants to use AWS's strong technology to do more than just make suggestions. It also wants to give users an immersive and responsive experience.

Objectives

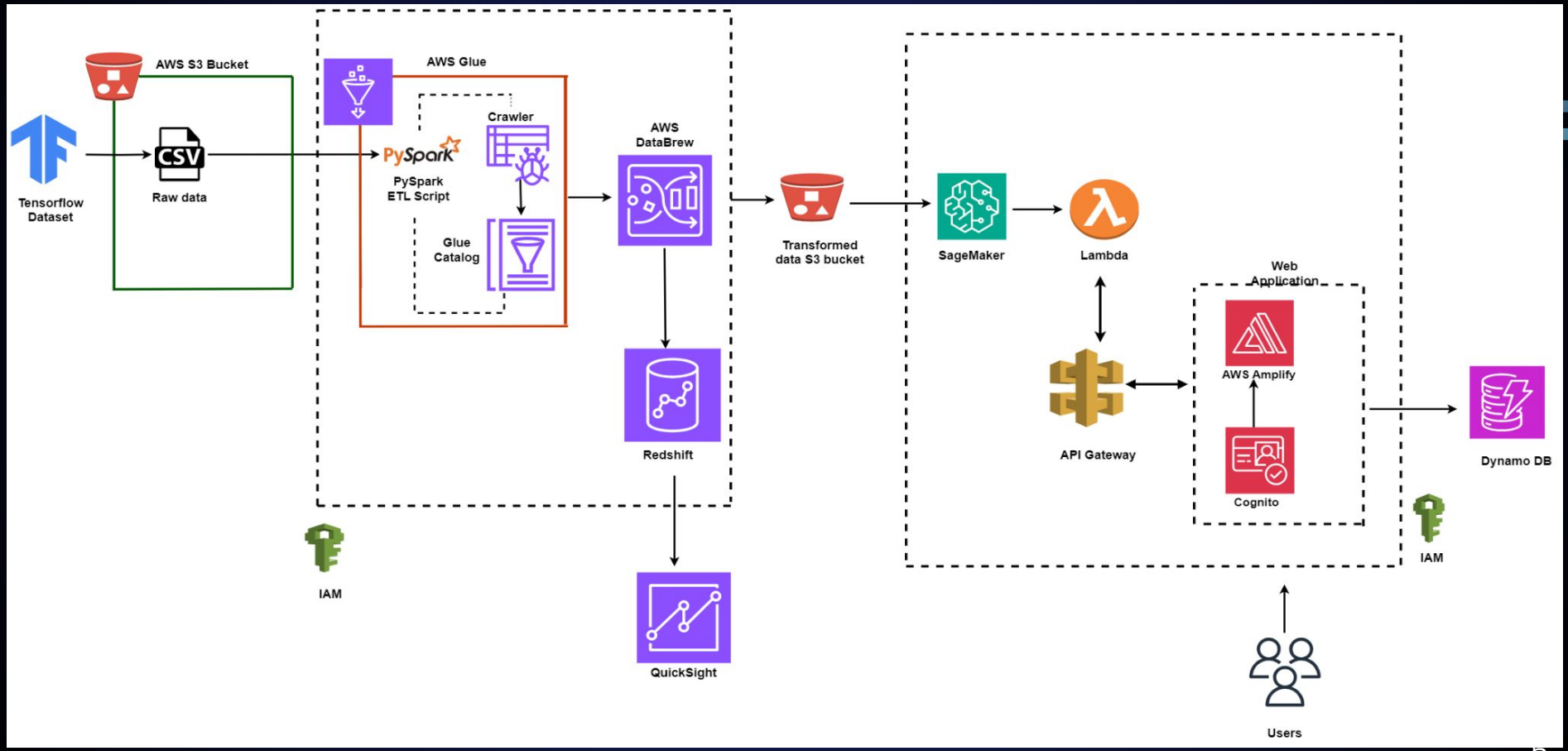
- Scalable Recommendation Engine
- Personalized User Experiences
- Cross-Domain Recommendation
- Optimized Resource Utilization

Dataset Description

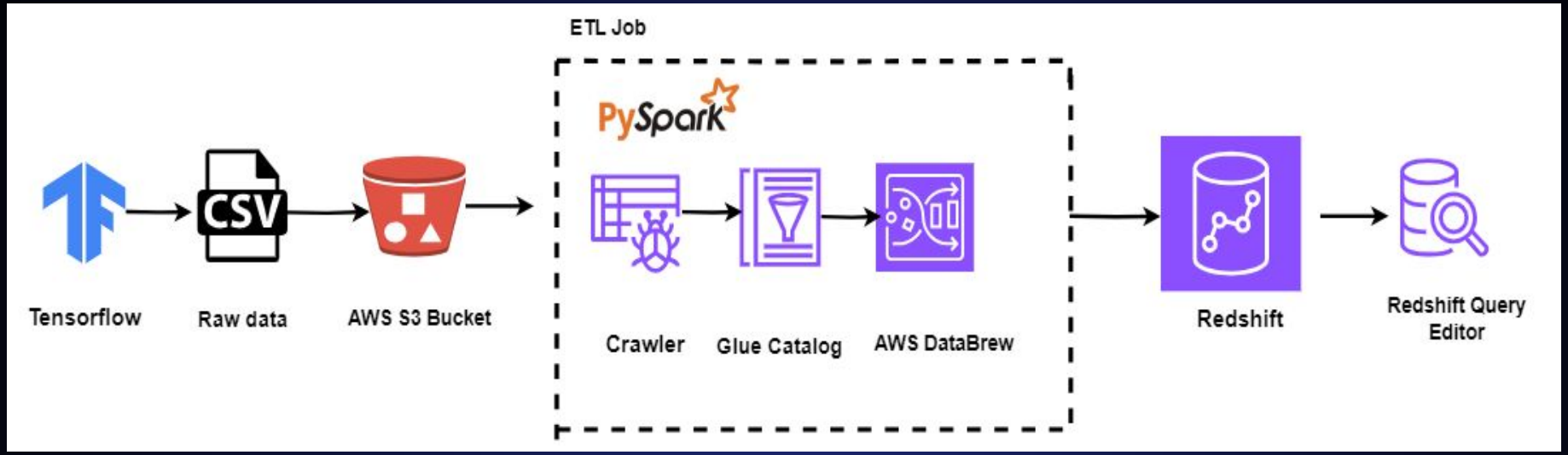
- The MovieLens dataset comprises user ratings for movies, collected by the GroupLens research group.
- The "1m-ratings" version, demographic features include user gender, bucketized user age, user occupation label, user occupation text, and user zip code.
- The dataset comprises information on books, including unique identifiers, titles, author details, publication information, and reader engagement metrics.
- The unnecessary columns from the movie and book dataset are removed.

  https://www.tensorflow.org/datasets/catalog/movie_lens#movie_lens1m-ratings

System Architecture



Data flow Diagram



1)The MovieLens Tensorflow dataset is initially stored in an **S3 bucket**.

2)ETL using **PySpark** with **AWS Glue Catalog** and **Crawlers**.

3)AWS **DataBrew** for Data Preparation.

4)Transformed data is stored in **Redshift**.

5)**Redshift Query editor** for queries and to visualize results.

Data Frame of Movie Dataset

```
# Display the entire DataFrame
print("Displaying the DataFrame:")
print(df)
```

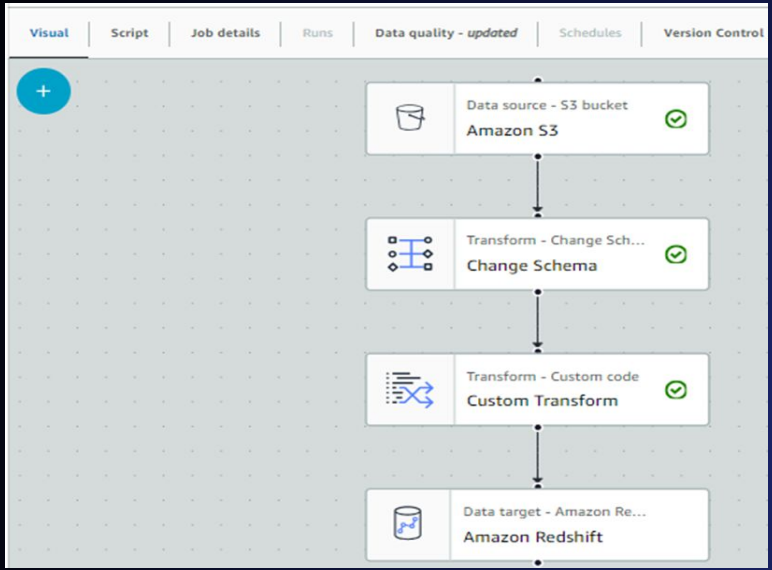
```
Displaying the DataFrame:
       age         genres  movieid                                   title
0       45            [7]   b'357'  b"One Flew Over the Cuckoo's Nest (1975)"
1       25        [ 4 14]   b'709'                b'Strictly Ballroom (1992)'
2       18            [4]   b'412'            b'Very Brady Sequel, A (1996)'
3       50          [5 7]    b'56'                   b'Pulp Fiction (1994)'
4       50        [10 16]   b'895'                       b'Scream 2 (1997)'
...    ...            ...     ...                                      ...
99995   25     [ 0  1 15]   b'228'     b'Star Trek: The Wrath of Khan (1982)'
99996   35        [13 16]   b'333'                     b'Game, The (1997)'
99997   18           [10]   b'567'       b"Wes Craven's New Nightmare (1994)"
99998   35  [ 0 10 15 16]   b'183'                          b'Alien (1979)'
99999   18            [4]  b'1140'           b'Road to Wellville, The (1994)'

       userage  timestamp  gender  userid        occupation
0           46  879024327    True  b'138'         b'doctor'
1           32  875654590    True   b'92'  b'entertainment'
2           24  882075110    True  b'301'        b'student'
3           50  883326919    True   b'60'     b'healthcare'
4           55  891409199    True  b'197'     b'technician'
...        ...        ...     ...     ...               ...
99995       30  888557237    True  b'774'        b'student'
99996       41  891012877    True  b'313'      b'marketing'
99997       19  879795430   False  b'262'        b'student'
99998       37  892839492   False  b'911'         b'writer'
99999       21  874791894    True  b'276'        b'student'

[100000 rows x 9 columns]
```

Data Frame of Book Dataset

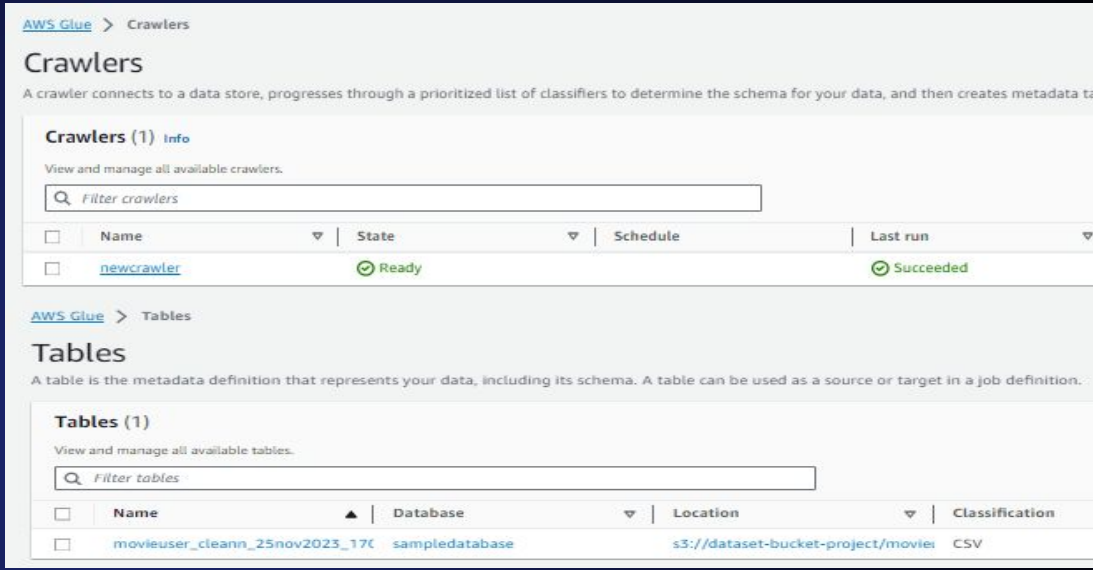
```
print("Displaying the DataFrame:")
print(df)
```

```
Displaying the DataFrame:
           id                                                title
0           0                                            The Martian
1           1  Under the Banner of Heaven: A Story of Violent...
2           2                                     Cutting for Stone
3           3                                We Need to Talk About Kevin
4           4                      The Immortal Life of Henrietta Lacks
...       ...                                                   ...
10013   10013                 Edge of Passion (Stealth Guardians, #1)
10014   10014  Tales from a Not-So-Popular Party Girl (Dork D...
10015   10015                                              Bottoms
10016   10016                                     The Wedding Dress
10017   10017  The Immortal Hunter (Argeneau #11; Rogue Hunte...

                                                 book_link  \
0       https://www.goodreads.com/book/show/18007564-t...
1       https://www.goodreads.com/book/show/10847.Unde...
2       https://www.goodreads.com/book/show/3591262-cu...
3       https://www.goodreads.com/book/show/80660.We_N...
4       https://www.goodreads.com/book/show/6493208-th...
...                                                     ...
10013   https://www.goodreads.com/book/show/13516444-e...
10014   https://www.goodreads.com/book/show/8274537-ta...
10015   https://www.goodreads.com/book/show/2024071.Bo...
10016   https://www.goodreads.com/book/show/783968.The...
10017   https://www.goodreads.com/book/show/3942622-th...

                                                    genre
0       Science Fiction,Fiction,Audiobook,Adventure,Sp...
1       Nonfiction,Religion,History,Crime,True Crime,M...
2       Fiction,Historical,Historical Fiction,Cultural...
3       Fiction,Contemporary,Thriller,Horror,Mystery,C...
4       Nonfiction,Science,History,Biography,Health,Me...
...                                                     ...
10013   Fantasy,Paranormal,Romance,Romance,Paranormal ...
10014   Childrens,Middle Grade,Realistic Fiction,Child...
10015                                                 NaN
10016                                                 NaN
10017   Romance,Paranormal Romance,Paranormal,Vampires...

[10018 rows x 4 columns]
```

ETL Process



AWS Crawlers and Glue Catalog



1. Initializes Spark and Glue contexts, and creates a Glue job.

2. Reads a CSV file from an S3 bucket into a dynamic frame .

3.Applies schema mapping to change data types of selected columns in the dynamic frame.

4.Runs a Glue crawler to discover and catalog data from the transformed dynamic frame.

5.Stores the transformed dynamic frame to Redshift for further analysis.

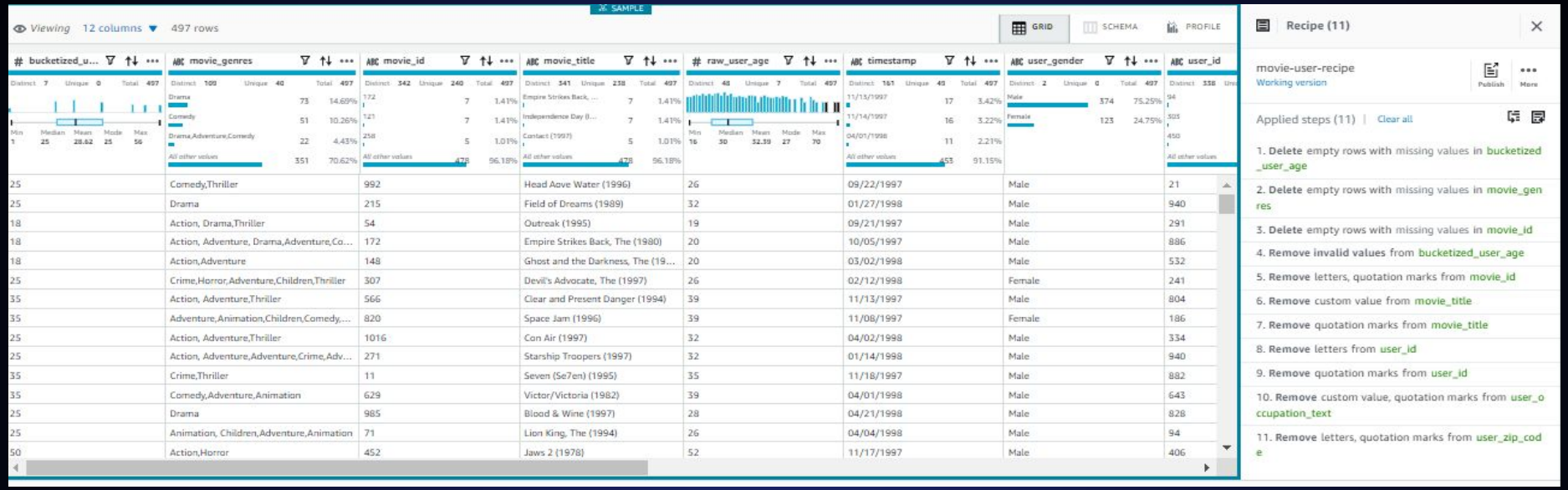
PySpark Script for Schema Transformation

Script (Locked) Info

```
# Script generated for node Amazon S3
AmazonS3_node1701323994859 = glueContext.create_dynamic_frame.from_options(
    format_options={
        "quoteChar": '"',
        "withHeader": True,
        "separator": ",",
        "optimizePerformance": False,
    },
    connection_type="s3",
    format="csv",
    connection_options={
        "paths": [
            "s3://dataset-bucket-project/movieuser-cleann_25Nov2023_170094094505 2/movie_clean.csv"
        ],
        "recurse": True,
    },
    transformation_ctx="AmazonS3_node1701323994859",
)

# Script generated for node Change Schema
ChangeSchema_node1701324004477 = ApplyMapping.apply(
    frame=AmazonS3_node1701323994859,
    mappings=[
        ("bucketized_user_age", "string", "bucketized_user_age", "int"),
        ("movie_genres", "string", "movie_genres", "string"),
        ("movie_id", "string", "movie_id", "int"),
        ("movie_title", "string", "movie_title", "string"),
        ("raw_user_age", "string", "raw_user_age", "int"),
        ("timestamp", "string", "timestamp", "timestamp"),
        ("user_gender", "string", "user_gender", "char"),
        ("user_id", "string", "user_id", "int"),
        ("user_occupation_label", "string", "user_occupation_label", "string"),
        ("user_occupation_text", "string", "user_occupation_text", "string"),
        ("user_rating", "string", "user_rating", "int"),
        ("user_zip_code", "string", "user_zip_code", "int"),
    ],
    transformation_ctx="ChangeSchema_node1701324004477",
)
```

Applied schema mapping to the dynamic frame, transforming the data types of selected columns.

AWS Glue Data Brew



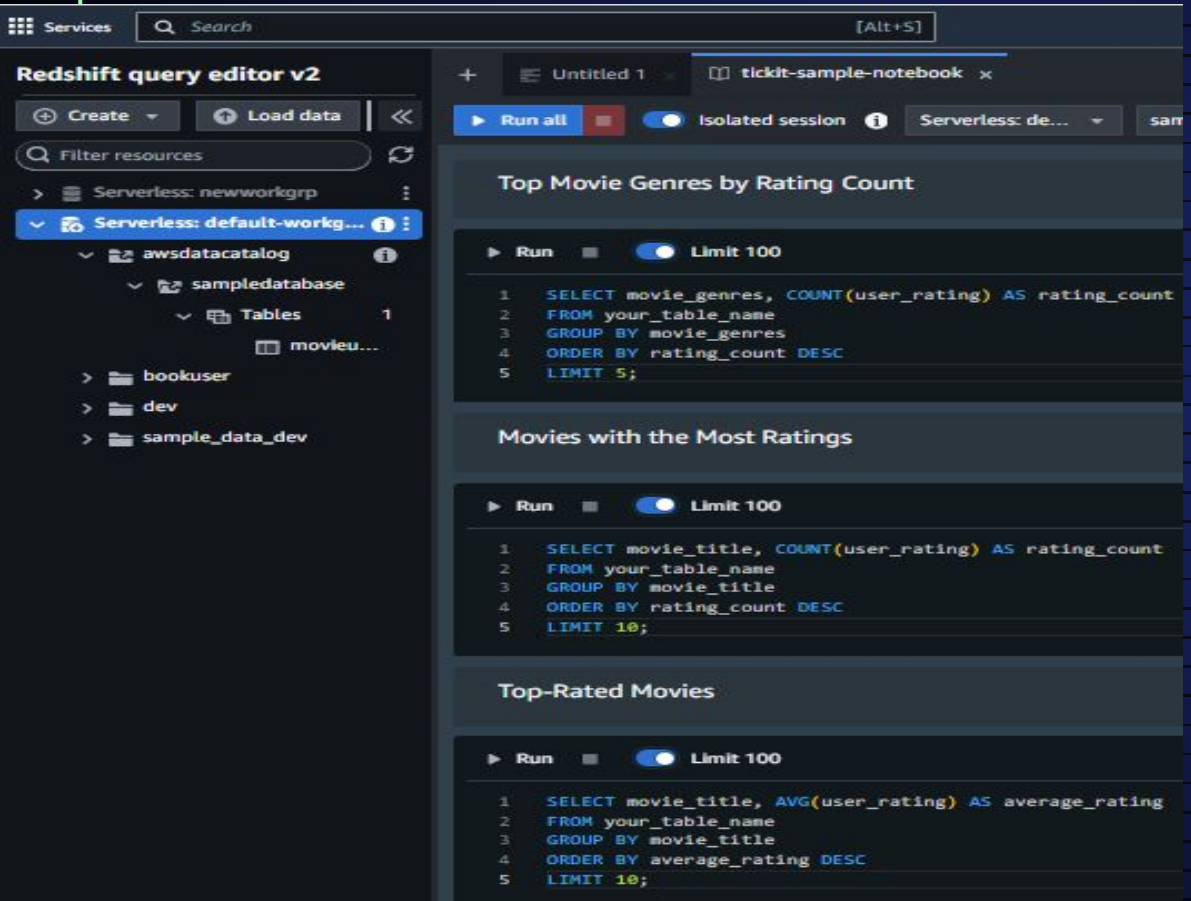
1.DataBrew will automatically profile and provide insights into the data structure and quality.

2.“Recipe” is used to build a series of transformations for data preparation.

3.Applied transformations like duplicate records, handling missing values, and formatting columns.

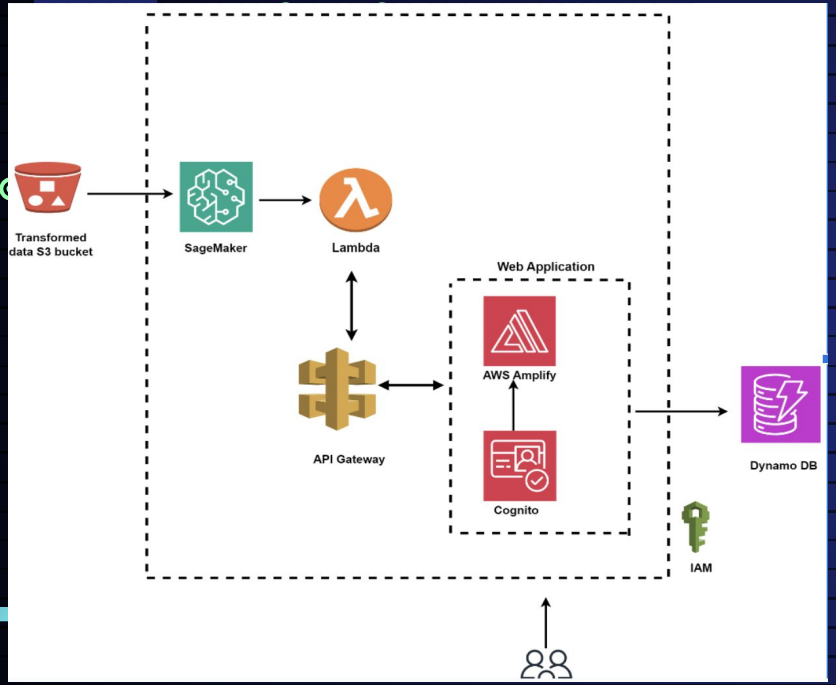
4.Exported the cleaned data to Redshift.

Querying Data in Redshift Query Editor



- Redshift query editor is a Web -Based SQL interface.
- The data after transformation is analysed and queried for meaningful insights.
- Queries were visualized using Built-in chart in redshift editor.

Workflow-Web Application



1. Setting up Authentication with **AWS Cognito**

2.Frontend Development with **AWS Amplify**

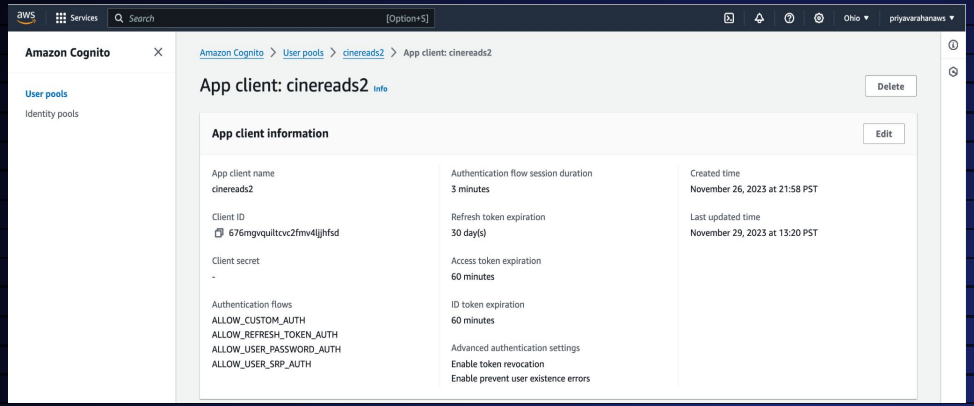
3.Backend API with **API Gateway** and **AWS Lambda**

4. Machine Learning with **SageMaker**

5. Connecting the Components using **IAM**

6. Storing the data in **DynamoDB**

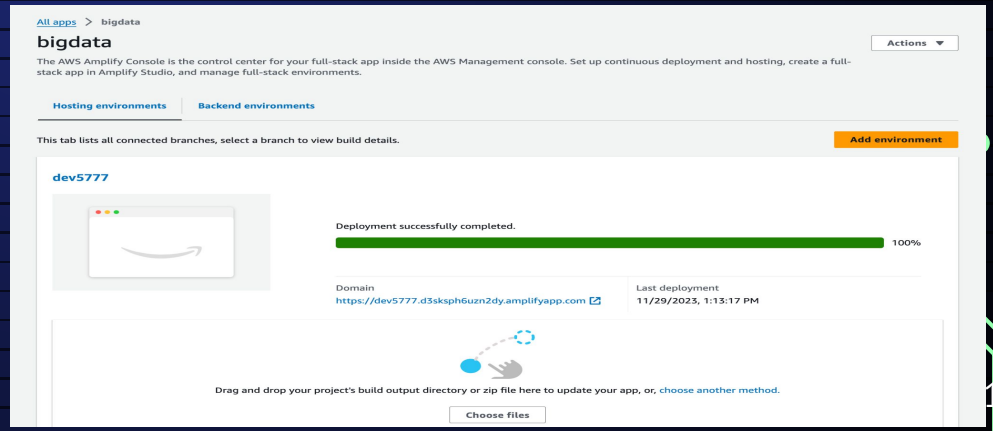
1. Setting up Authentication with AWS Cognito



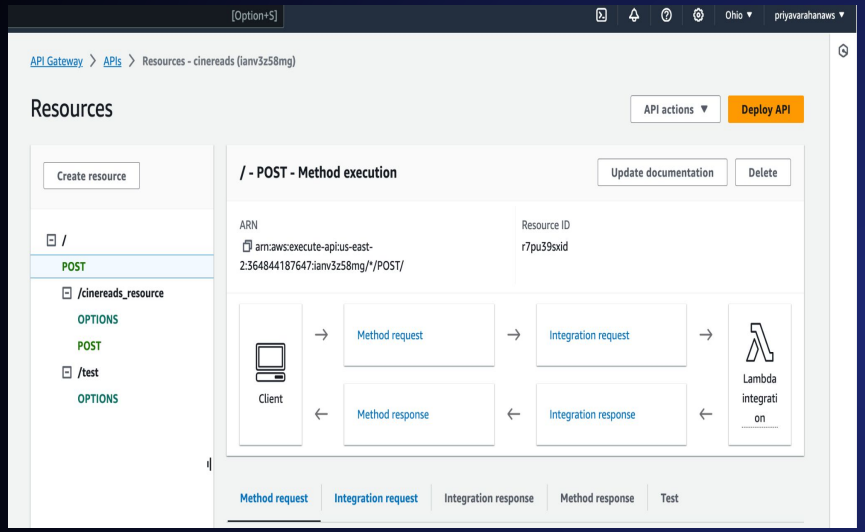
- User pool creation
- Identity pool creation
- Integration with front end

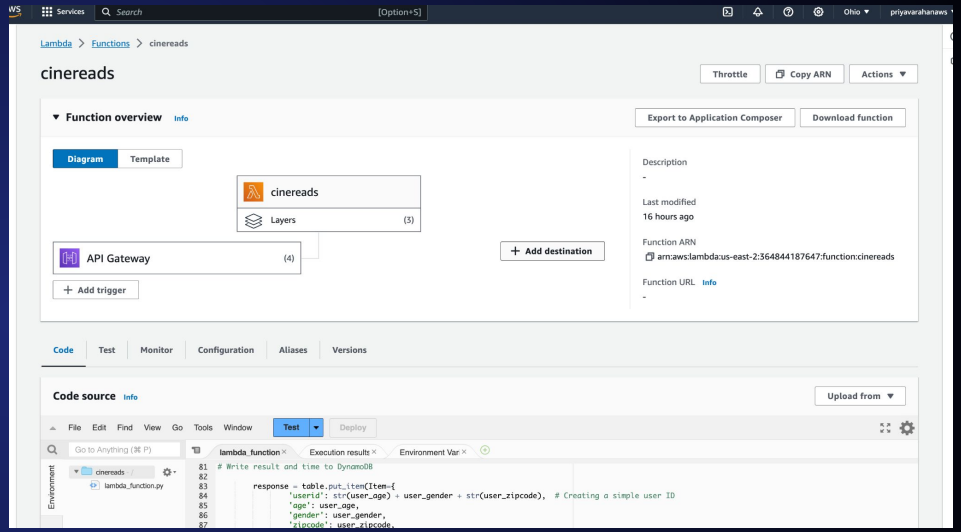
2.Frontend Development with AWS Amplify

- Initialize Amplify
- Hosting
- Frontend Frameworks
- Amplify Libraries



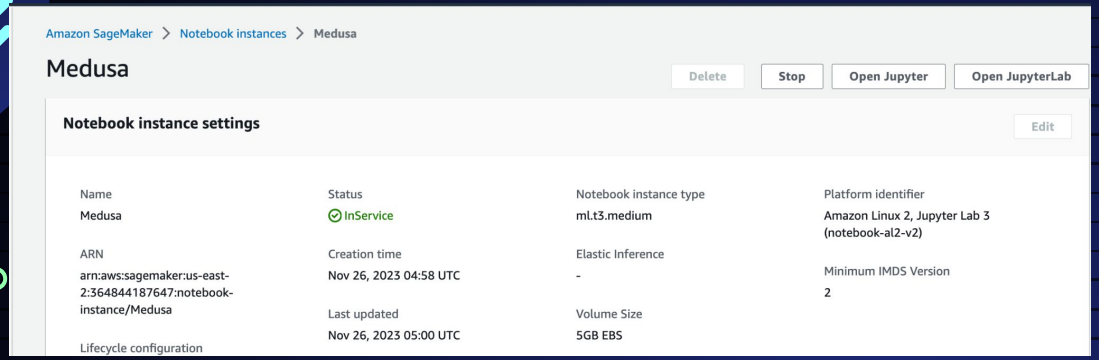
3.Backend API with API Gateway and AWS Lambda





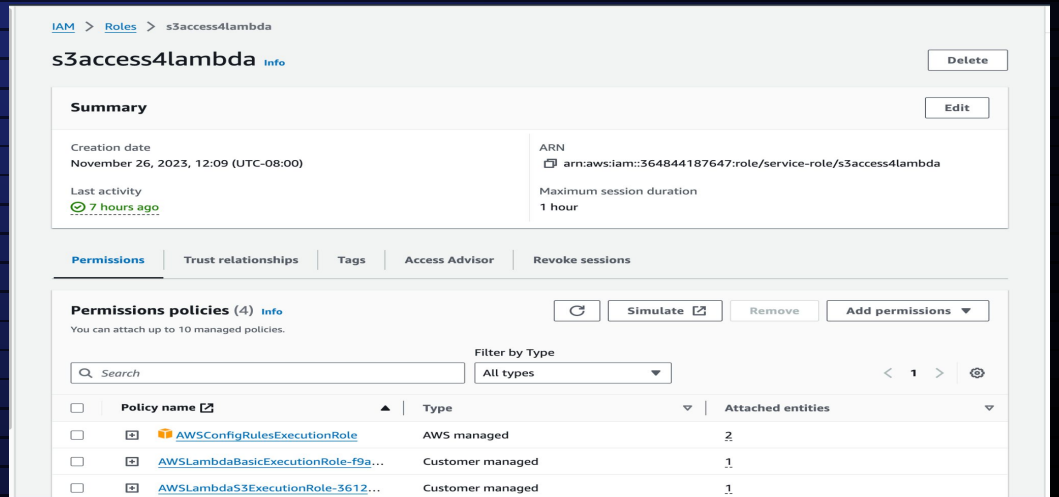
- API Gateway
- Lambda functions to handle API requests
- integration between API Gateway and Lambda functions.

4. Machine Learning with SageMaker



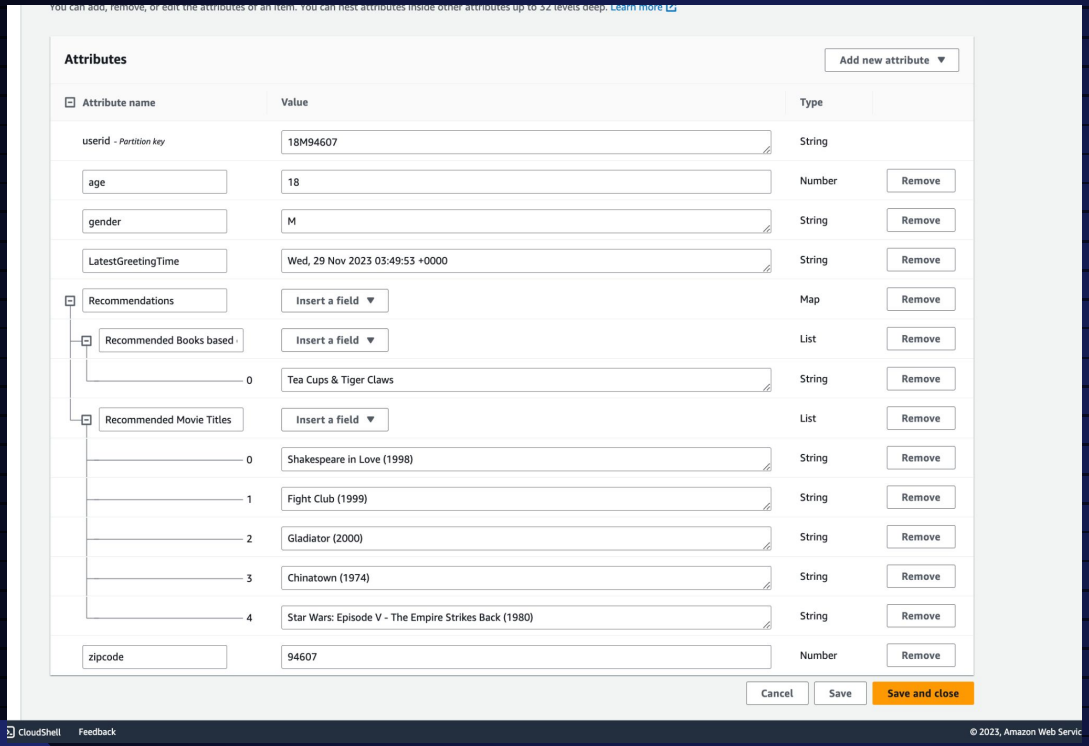
- Model Training for book and movie recommendation

5. Connecting the Components using IAM



- Secure Access
- Data Flow

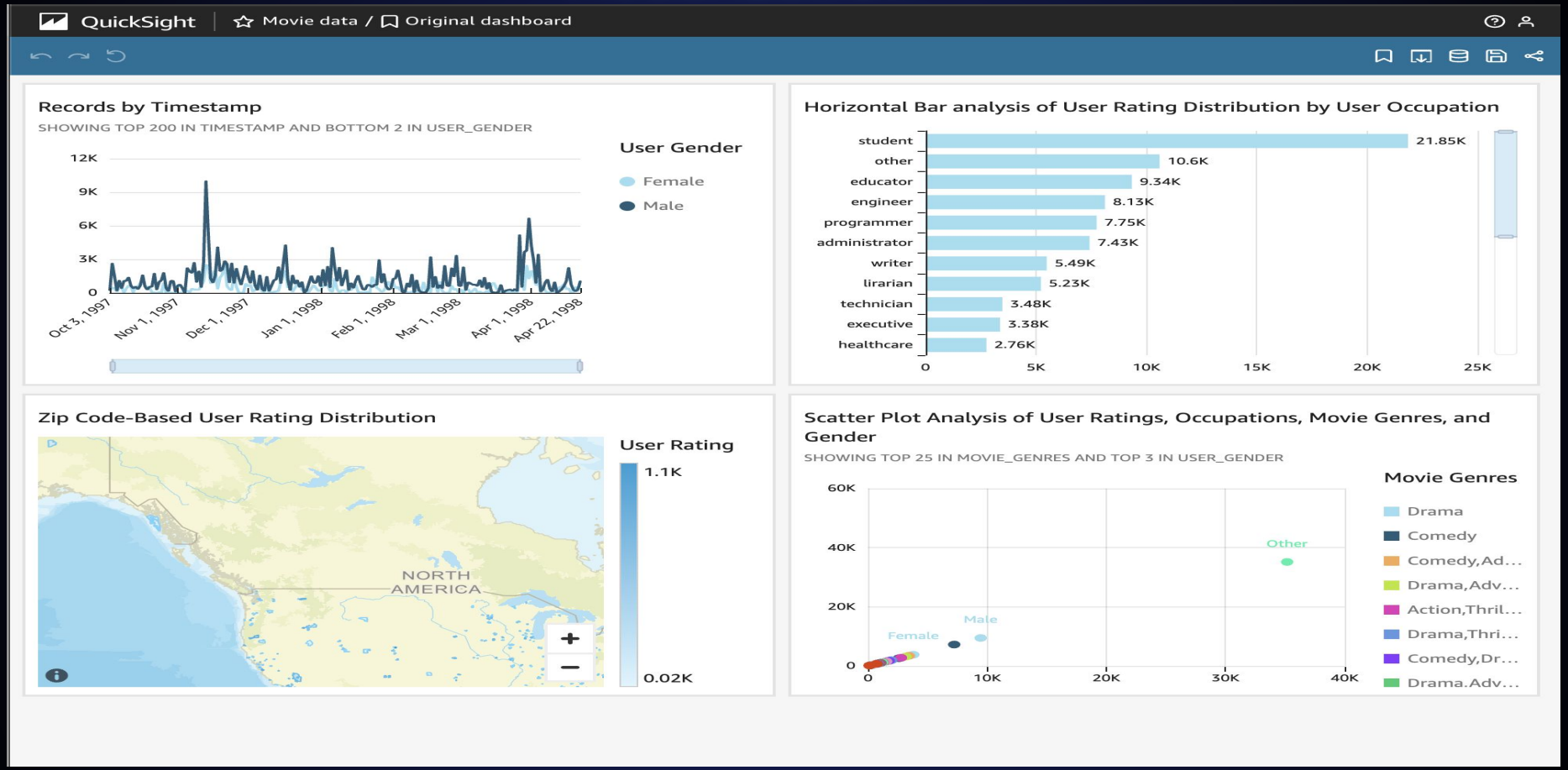
6. Storing the Data in DynamoDB



- store the recommendations made by the system

Demo

Visualization



https://us-east-2.quicksight.aws.amazon.com/sn/dashboards/0b1b3d74-f84b-4e13-af08-4cff4d4fcbfe

Project Impact

- The project has the capacity to significantly influence multiple sectors and demography. These are the main areas that have been impacted.

Consumer Personalization and Experience

- Enhanced User Experience
- Cross-Media Recommendations

Data Analytics and Big Data

- Leveraging Big Data

Demographic Insights

- Understanding Consumer Trends

Future Work

- Integration with Other Technologies like virtual and augmented reality, voice assistants, and other emerging technologies could further enhance user interaction and engagement.
- Avoiding Filter Bubbles
  - Serendipity in Recommendations strictly based on the user's past behavior.
  - Multi-Faceted Profiling based on a variety of factors (e.g., mood, recent news, time of day) in addition to historical preferences.
- Ethical Considerations

Conclusion

- Analyzed the data set and identified the descriptive features required for our book and movie recommendation system
- Identified and implemented algorithms to predict movie and books based on the user inputs.
- Visualisation of the available data
- Implemented a system to accept the users details to recommend movies and books

Reference

- https://github.com/Priyankaakula/Medusa/blob/main/README.md
- https://us-east-2.quicksight.aws.amazon.com/sn/dashboards/0b1b3d74-f84b-4e13-af08-4cff4d4fcbfe
- https://dev5777.d3sksph6uzn2dy.amplifyapp.com
- https://app.diagrams.net/?src=about
- https://www.tensorflow.org/datasets/catalog/movie_lens#movie_lens1m-ratings
- https://ianv3z58mg.execute-api.us-east-2.amazonaws.com/cinereads

Thank You