ZOMATO DATA ANALYSIS

INTRODUCTION

The objective of this project is to build a data solution that analyzes the Zomato dataset and visualizes the analytics on Power BI dashboards. The assumption is made that updated datasets are received at specific intervals and uploaded into a data lake, which is then processed through a data pipeline to produce modeled data in a data warehouse. The pre-defined dashboards in Power BI can be used to answer various questions, such as determining if certain cities or countries have a preference for a specific cuisine, the correlation between locality and average cost for two, the highest rated restaurants based on cuisine and location, and the factors that affect the rating. The flow of the project includes data movement from ADLS Gen 2 to Azure Data Factory, Snowflake and Snowpark, and then to Power BI. The expected outcomes include source code committed into version control, end-to-end data pipeline implementation, an Azure DevOps pipeline to deploy ADF pipeline on Azure cloud, a solution deck, a demo recording, and all the artifacts published in the given SharePoint folder.

Figure: ZOMATO DATA ANALYSIS

PROBLEM STATEMENT

We need to build a solution that processes Zomato Dataset and visualize analytics on Power BI Dashboard(s). We need to assume that we are receiving updated data set at specific interval. That dataset raw files are uploaded in data lake. Once we have the new file in data lake, we process the data through a data pipeline and produce modeled data in data warehouse. We will be able to visualize the data in Power BI through pre-defined dashboards.

DESCRIPTION OF DATASETS

We have been provided 2 dataset of zomato that is the csv files. The first file is the detailed descriptiowhichincludesRestaurantID,RestaurantName,CountryCode,City,Address,LocalityLocality Verbose, Longitude, Latitude, Cuisines,Average Cost for two Currency,Has Table booking,Has Online delivery,Price rang,Aggregate rating, Rating color ,,Rating text Votes

SYSTEM ARCHITECTURE DIAGRAM

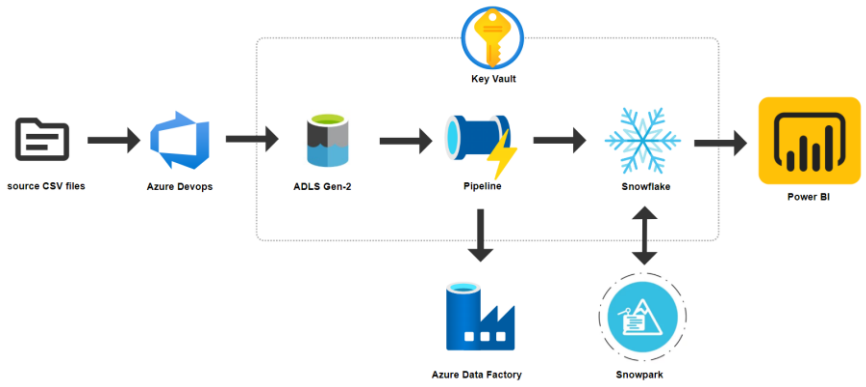


Figure: System architecture.

SYSTEM WORKFLOW DIAGRAM

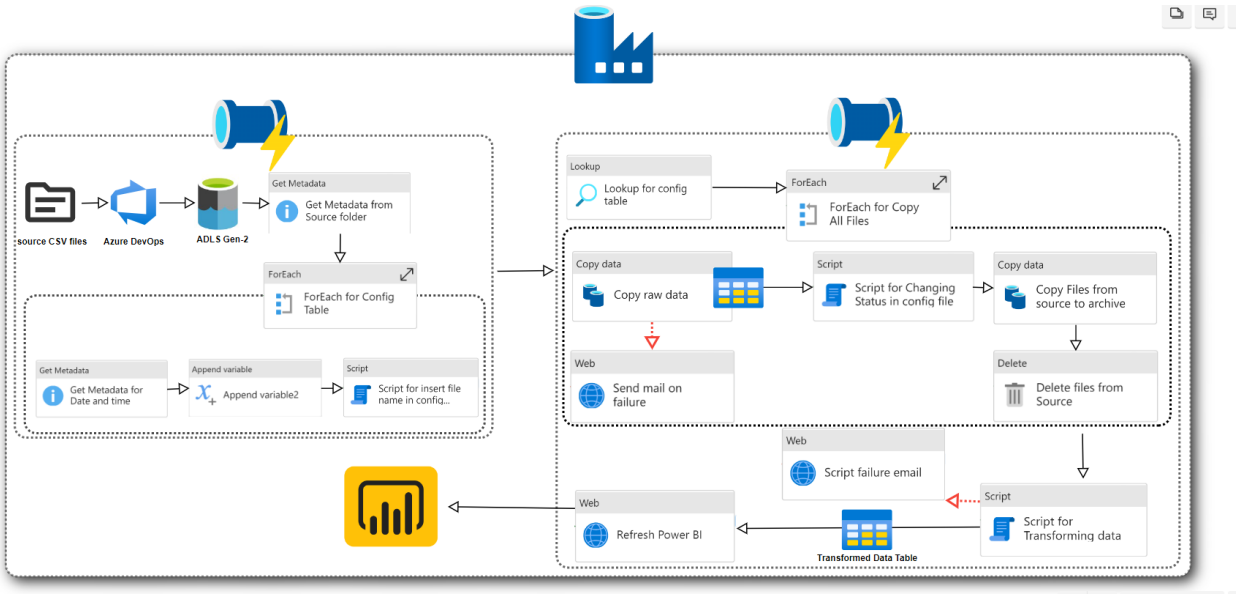


Figure:Work Flow of Process

Whenever new files been uploaded in the azure repo, the data pipeline will be triggered automatically and the file is moved to the adf where the files is been processed using specific data pipeline and create modeled data using the snowflake script.

DATASET(data.csv)

Restaurant ID	Restaurant Name	Country Code	City	Address	Locality
7402935	Skye	94	Jakarta	Menara BCA, Lantai 56, Jl. MH. Thamrin, Thamrin, Jakarta	Grand Indonesia Mall, Thamrin
7410290	Satoo - Hotel Shangri-La	94	Jakarta	Hotel Shangri-La, Jl. Jend. Sudirman	Hotel Shangri-La, Sudirman
7420899	Sushi Masa	94	Jakarta	Jl. Tuna Raya No. 5, Penjaringan	Penjaringan
7421967	3 Wise Monkeys	94	Jakarta	Jl. Suryo No. 26, Senopati, Jakarta	Senopati
7422489	Avec Moi Restaurant and Bar	94	Jakarta	Gedung PIC, Jl. Teluk Betung 43, Thamrin, Jakarta	Thamrin
18352452	Lucky Cat Coffee & Kitchen	94	Jakarta	Plaza Festival, South Parking, Jl. HR Rasuna Said, Kuningan, Jakarta	Plaza Festival, Kuningan
18386856	Onokabe	94	Tangerang	Alam Sutera Town Center, Jl. Alam Utama, Serpong, Tangerang	Alam Sutera Town Center, Serpong Utara
7423482	Lemongrass	94	Bogor	Jl. Raya Pajajaran No. 21, Bogor Utara, Bogor	Bogor Utara
18391256	MONKS	94	Jakarta	Komplek Graha Boulevard Timur, Summarecon Kelapa Gading Blok ND1/51, Kelapa Gading, Jakarta	Kelapa Gading
7422633	Talaga Sampireun	94	Jakarta	Jl. Lingkar Luar Barat	Cengkareng
18425821	OJJU	94	Jakarta	Gandaria City, Lantai Upper Ground, Jl. Sultan Iskandar Muda	Gandaria City Mall, Gandaria
7422751	Union Deli	94	Jakarta	Grand Indonesia Mall, Lantai Ground, East Mall, Jl. MH Thamrin, Thamrin, Jakarta	Grand Indonesia Mall, Thamrin
7400818	Zenbu	94	Jakarta	Kota Kasablanka, Lantai Upper Ground, Food Society, Jl. Casablanca Raya, Tebet, Jakarta	Kota Kasablanka, Tebet
7417455	Talaga Sampireun	94	Jakarta	Taman Impian Jaya Ancol, Jl. Lapangan Golf 7, Ancol, Jakarta	Taman Impian Jaya Ancol, Ancol
7417450	Talaga Sampireun	94	Tangerang	Jl. Boulevard Bintaro Jaya Blok B7/N1, Bintaro Sektor 7, Pondok Aren, Tangerang	Pondok Aren
7405789	Toodz House	94	Jakarta	Jl. Cipete Raya No. 79, Fatmawati, Jakarta	Fatmawati
18400530	Noah's Barn Coffeenery	94	Bandung	Jl. Dayang Sumbi No. 2, Dago, Bandung	Dago
18370659	Flip Burger	94	Jakarta	Jl. Senopati No. 27, Senopati, Jakarta	Senopati
18409146	Fish Streat	94	Jakarta	Jl. Tanjung Duren Utara III, Blok M Kav. 32, Tanjung Duren, Jakarta	Tanjung Duren
18408381	Fish Streat	94	Jakarta	Jl. Tebet Timur Dalam Raya 44B, Tebet, Jakarta	Tebet
7423620	Momo Milk	94	Bogor	Jl. Kantor Pos No. 6, Bogor Timur, Bogor	Bogor Timur
2701	Orient Express - Taj Palace Hotel	1	New Delhi	Taj Palace Hotel, Diplomatic Enclave, Chanakyapuri, New Delhi	The Taj Palace Hotel, Chanakyapuri
309548	Tian - Asian Cuisine Studio - ITC Maurya	1	New Delhi	ITC Maurya, Diplomatic Enclave, Chanakyapuri, New Delhi	ITC Maurya, Chanakyapuri
2742	Bukhara - ITC Maurya	1	New Delhi	ITC Maurya, Chanakyapuri, New Delhi	ITC Maurya, Chanakyapuri
6300010	Spiral - Sofitel Philippine Plaza Manila	162	Pasay City	Plaza Level, Sofitel Philippine Plaza Manila, CCP Complex, Pasay City	Sofitel Philippine Plaza Manila, Pasay City
301523	Nostalgia at 1911 Brasserie - The Imperial	1	New Delhi	The Imperial, Janpath, New Delhi	The Imperial, Janpath
2724	1911 - The Imperial	1	New Delhi	The Imperial, Janpath, New Delhi	The Imperial, Janpath
2725	The Spice Route - The Imperial	1	New Delhi	The Imperial, Janpath, New Delhi	The Imperial, Janpath
2694	Wasabi by Morimoto - The Taj Mahal Hotel	1	New Delhi	The Taj Mahal Hotel, 1, Mansingh Road, New Delhi	The Taj Mahal Hotel, Mansingh Road
6812	MEGU - The Leela Palace	1	New Delhi	The Leela Palace, Diplomatic Enclave, Chanakyapuri, New Delhi	The Leela Palace, Chanakyapuri
2689	House of Ming - The Taj Mahal Hotel	1	New Delhi	The Taj Mahal Hotel, 1, Mansingh Road, New Delhi	The Taj Mahal Hotel, Mansingh Road
3910	24/7 Restaurant - The Lalit New Delhi	1	New Delhi	The Lalit, Barakhamba Avenue, Barakhamba Road, New Delhi	The Lalit New Delhi, Barakhamba Road
2443	Wildfire - Crowne Plaza	1	Gurgaon	Crowne Plaza, National Highway 8, Sector 29, Gurgaon	Crowne Plaza, Sector 29

Figure: Dataset

RESULTS

Result of the modeled data

RESTAURANT_ID	RESTATURANT_NAME	CITY	ADDRESS
96814	Saffron Mantra	Secunderabad	The Purple Leaf Hotel, Karkhana, Se
96814	Saffron Mantra	Secunderabad	The Purple Leaf Hotel, Karkhana, Se
310691	Singz	New Delhi	A-5C/32A, Janakpuri, New Delhi
310691	Singz	New Delhi	A-5C/32A, Janakpuri, New Delhi
18124345	Green Chick Chop	Gurgaon	Shop 114-E, HUDA Market, Sector
18124345	Green Chick Chop	Gurgaon	Shop 114-E, HUDA Market, Sector
304913	Moti Mahal Delux	New Delhi	Lower Ground Floor, Epicuria Food
304913	Moti Mahal Delux	New Delhi	Lower Ground Floor, Epicuria Food
304913	Moti Mahal Delux	New Delhi	Lower Ground Floor, Epicuria Food
18362677	Michael's Kitchen	Bhubaneshwar	7798, Sainik School Road, Gajapati
18362677	Michael's Kitchen	Bhubaneshwar	7798, Sainik School Road, Gajapati
18362677	Michael's Kitchen	Bhubaneshwar	7798, Sainik School Road, Gajapati

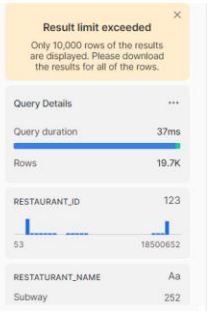


Figure: Modeled dataset

TABLES CREATED

```
USE ZOMATO_ANALYSIS;


CREATE OR REPLACE TABLE zomato_raw_data(
   Restaurant_ID NUMBER,
   Restaturant_Name VARCHAR(100),
   Country_Code INT,
   City VARCHAR(150),
   Address VARCHAR(200),
   Locality VARCHAR(100),
   Locality_Verbose VARCHAR(100),
   Longitude DOUBLE,
   Latitude DOUBLE,
   Cuisines VARCHAR(100),
   Average_Cost_for_two  INT,
   Currency VARCHAR(100),
   Has_Table_booking VARCHAR(4),
   Has_Online_delivery VARCHAR(4),
   Price_range INT,
   Aggregate_rating FLOAT,
   Rating_color VARCHAR(50),
   Rating_text VARCHAR(50),
   votes INT,
   unique_key VARCHAR(200));

CREATE OR REPLACE TABLE zomato_data(
   RESTAURANT_ID NUMBER,
   RESTATURANT_NAME VARCHAR(100),
   CITY VARCHAR(150),
   ADDRESS VARCHAR(200),
   LOCALITY VARCHAR(100),
   LOCALITY_VERBOSE VARCHAR(100),
   LONGITUDE DOUBLE,
   LATITUDE DOUBLE,
   CUISINES VARCHAR(100),
   AVERAGE_COST_FOR_TWO  DOUBLE,
   CURRENCY VARCHAR(100),
   HAS_TABLE_BOOKING VARCHAR(4),
   HAS_ONLINE_DELIVERY VARCHAR(4),
   PRICE_RANGE INT,
   AGGREGATE_RATING FLOAT,
   RATING_COLOR VARCHAR(50),
   RATING_TEXT VARCHAR(50),
   VOTES INT,
   COUNTRY_CODE INT,
   COUNTRY_NAME VARCHAR(100),
   unique_key VARCHAR(200)
```

```
);

create or replace table config (filename varchar(150),date_time varchar(500),status
varchar(10),destination_table varchar(200),file_count int);
```

PROCEDURE FOR CONFIG TABL

```
CREATE OR REPLACE PROCEDURE CONFIG_PROC(config_table_name VARCHAR(50),
file_name VARCHAR(50), date_time_txt VARCHAR(50), status
VARCHAR(50),destination_table_name VARCHAR(50))
  RETURNS VARCHAR(50)
  LANGUAGE PYTHON
  RUNTIME_VERSION = '3.8'
  PACKAGES = ('snowflake-snowpark-python')
  HANDLER = 'run'
AS
$$
import snowflake.connector
from snowflake.snowpark.session import Session
from snowflake.snowpark.functions import col,lit,concat
from snowflake.snowpark.functions import udf, col, lit, translate, is_null, iff
from snowflake.snowpark.types import IntegerType, StringType, StructType, StructField
def run(session,config_table_name,file_name,date_time_txt,status,destination_table_name):


        #count same name file
  count_file=session.table(config_table_name).where(col('filename').in_(file_name)).count()

  #check if file is already exits or not
  if count_file == 0:

        #if file is not exits so insert current file details in config table

    session.sql("insert into "+config_table_name+"
values('"+file_name+"','"+date_time_txt+"','"+status+"','"+destination_table_name+"',1)").collect()

  else:

        #if file is exits so upadate a load status and date and time


#session.table(config_table_name).update({"status":status,"date_time":date_time_txt},col("filena
me")==file_name)
    session.table(config_table_name).update({"status":status,"date_time":concat(
col("date_time").try_cast(StringType()),lit(','),lit(date_time_txt)),"file_count":col("file_count")+1},c
ol("filename")==file_name)
```

```
  return "SUCCESS"
$$;

CALL CONFIG_PROC('config','data.csv','9/10/2023','TRUE','zomato_raw_data');
```

PROCEDURE FOR DATA TRANSFORMATION

```
CREATE OR REPLACE PROCEDURE ZOMATO_PROC(zomato_raw_data_table
VARCHAR(100),zomato_country_table VARCHAR(100),zomato_clean_data_table
VARCHAR(100))
  RETURNS STRING
  LANGUAGE PYTHON
  RUNTIME_VERSION = '3.8'
  PACKAGES = ('snowflake-snowpark-python')
  HANDLER = 'run'
AS
$$
import snowflake.connector
from snowflake.snowpark.session import Session
from snowflake.snowpark.functions import col,split,regexp_replace,when,lit
from snowflake.snowpark.functions import udf, col, lit, translate, is_null, iff,trim
from snowflake.snowpark.types import IntegerType, StringType, StructType, StructField
def run(session,zomato_raw_data_table,zomato_country_table,zomato_clean_data_table):


  #update a unique key in zomato raw data using hash key
  session.sql("update "+zomato_raw_data_table+" set
unique_key=HASH(Restaurant_ID,Country_Code,City,Address,Cuisines,Average_Cost_for_two
,Has_Table_booking,Has_Online_delivery,Price_range,Aggregate_rating,Rating_color,Rating_t
ext,votes)").collect()

  #get a zomato raw data from snowflake
  zomato_raw_data=session.table(zomato_raw_data_table)

  #get a country data from snowflake
  country_data=session.table(zomato_country_table)

  #get only a unique key from zomato clean data table
  unique_key=session.table(zomato_clean_data_table).select("unique_key")

  #get only unique record from zomato raw data table which is not transformed

zomato_data=zomato_raw_data.filter(col("unique_key").isin(unique_key)).select("RESTAURAN
T_ID",
"RESTATURANT_NAME","COUNTRY_CODE","CITY","ADDRESS","LOCALITY","LOCALITY_V
ERBOSE","LONGITUDE","LATITUDE","CUISINES","AVERAGE_COST_FOR_TWO","CURREN
CY","HAS_TABLE_BOOKING","HAS_ONLINE_DELIVERY","PRICE_RANGE","AGGREGATE_
RATING","RATING_COLOR","RATING_TEXT","VOTES","UNIQUE_KEY")
```

```
  zomato_data=zomato_raw_data.subtract(zomato_data)


        #join a zomato raw data table and country data table
  new_zomato_data=zomato_data.join(country_data, zomato_data.col("COUNTRY_CODE") ==
country_data.col("COUNTRY_CODE")).select("RESTAURANT_ID","RESTATURANT_NAME","
CITY","ADDRESS","LOCALITY","LOCALITY_VERBOSE","LONGITUDE","LATITUDE","CUISIN
ES","AVERAGE_COST_FOR_TWO","CURRENCY","HAS_TABLE_BOOKING","HAS_ONLINE
_DELIVERY","PRICE_RANGE","AGGREGATE_RATING","RATING_COLOR","RATING_TEXT"
,"VOTES",zomato_data['COUNTRY_CODE'].as_("COUNTRY_CODE"),"COUNTRY_NAME","U
NIQUE_KEY")

  #DROP DUPLICATE IF SAME RECORD IS EXITS IN SAME DATAFRAME

new_zomato_data=new_zomato_data.drop_duplicates("RESTAURANT_ID","RESTATURANT_
NAME","CITY","ADDRESS","LOCALITY",
"LOCALITY_VERBOSE","LONGITUDE","LATITUDE","CUISINES","AVERAGE_COST_FOR_T
WO","CURRENCY","HAS_TABLE_BOOKING","HAS_ONLINE_DELIVERY","PRICE_RANGE","
AGGREGATE_RATING","RATING_COLOR","RATING_TEXT","VOTES","COUNTRY_CODE","
COUNTRY_NAME","UNIQUE_KEY")



#drop row when cuisines is null
  new_zomato_data=new_zomato_data.filter(col("CUISINES").isNotNull())


        #convert the avg cost for two into dollar amount and replace all currency with the dollar

    #Convert indonesain Rupiah to Dollar($) and repalce currency Indonesian Rupiah(IDR) to
Dollar($)

new_zomato_data=new_zomato_data.withColumn("AVERAGE_COST_FOR_TWO",when(col("
COUNTRY_CODE")==94,
col("AVERAGE_COST_FOR_TWO")*0.000066).otherwise(col("AVERAGE_COST_FOR_TWO")
))\
.withColumn("CURRENCY",when(col("COUNTRY_CODE")==94,"Dollar($)").otherwise(col("CU
RRENCY")))\

.withColumn("AVERAGE_COST_FOR_TWO",when(col("COUNTRY_CODE")==1,col("AVERAG
E_COST_FOR_TWO")*0.012).otherwise(col("AVERAGE_COST_FOR_TWO")))\
.withColumn("CURRENCY",when(col("COUNTRY_CODE")==1,"Dollar($)").otherwise(col("CUR
RENCY")))\
.withColumn("AVERAGE_COST_FOR_TWO",when(col("COUNTRY_CODE")==162,col("AVER
AGE_COST_FOR_TWO")*0.076).otherwise(col("AVERAGE_COST_FOR_TWO")))\
.withColumn("CURRENCY",when(col("COUNTRY_CODE")==162,"Dollar($)").otherwise(col("C
URRENCY")))\
.withColumn("AVERAGE_COST_FOR_TWO",when(col("COUNTRY_CODE")==191,col("AVER
AGE_COST_FOR_TWO")*0.0028).otherwise(col("AVERAGE_COST_FOR_TWO")))\
.withColumn("CURRENCY",when(col("COUNTRY_CODE")==191,"Dollar($)").otherwise(col("C
URRENCY")))\
```

```
.withColumn("AVERAGE_COST_FOR_TWO",when(col("COUNTRY_CODE")==215,col("AVERAGE_COST_FOR_TWO")*1.22).otherwise(col("AVERAGE_COST_FOR_TWO")))\
.withColumn("CURRENCY",when(col("COUNTRY_CODE")==215,"Dollar($)").otherwise(col("CURRENCY")))\
.withColumn("AVERAGE_COST_FOR_TWO",when(col("COUNTRY_CODE")==148,col("AVERAGE_COST_FOR_TWO")*0.63).otherwise(col("AVERAGE_COST_FOR_TWO")))\
.withColumn("CURRENCY",when(col("COUNTRY_CODE")==148,"Dollar($)").otherwise(col("CURRENCY")))\
.withColumn("AVERAGE_COST_FOR_TWO",when(col("COUNTRY_CODE")==208,col("AVERAGE_COST_FOR_TWO")*0.053).otherwise(col("AVERAGE_COST_FOR_TWO")))\
.withColumn("CURRENCY",when(col("COUNTRY_CODE")==208,"Dollar($)").otherwise(col("CURRENCY")))\
.withColumn("AVERAGE_COST_FOR_TWO",when(col("COUNTRY_CODE")==30,col("AVERAGE_COST_FOR_TWO")*0.19).otherwise(col("AVERAGE_COST_FOR_TWO")))\
.withColumn("CURRENCY",when(col("COUNTRY_CODE")==30,"Dollar($)").otherwise(col("CURRENCY")))\
.withColumn("AVERAGE_COST_FOR_TWO",when(col("COUNTRY_CODE")==214,col("AVERAGE_COST_FOR_TWO")*0.27).otherwise(col("AVERAGE_COST_FOR_TWO")))\
.withColumn("CURRENCY",when(col("COUNTRY_CODE")==214,"Dollar($)").otherwise(col("CURRENCY")))\
.withColumn("AVERAGE_COST_FOR_TWO",when(col("COUNTRY_CODE")==166,col("AVERAGE_COST_FOR_TWO")*0.27).otherwise(col("AVERAGE_COST_FOR_TWO")))\
.withColumn("CURRENCY",when(col("COUNTRY_CODE")==166,"Dollar($)").otherwise(col("CURRENCY")))\
.withColumn("AVERAGE_COST_FOR_TWO",when(col("COUNTRY_CODE")==189,col("AVERAGE_COST_FOR_TWO")*0.056).otherwise(col("AVERAGE_COST_FOR_TWO")))\
.withColumn("CURRENCY",when(col("COUNTRY_CODE")==189,"Dollar($)").otherwise(col("CURRENCY")))\
.withColumn("AVERAGE_COST_FOR_TWO",when(col("COUNTRY_CODE")==14,col("AVERAGE_COST_FOR_TWO")*0.69).otherwise(col("AVERAGE_COST_FOR_TWO")))\
.withColumn("CURRENCY",when(col("COUNTRY_CODE")==14,"Dollar($)").otherwise(col("CURRENCY")))\
.withColumn("AVERAGE_COST_FOR_TWO",when(col("COUNTRY_CODE")==37,col("AVERAGE_COST_FOR_TWO")*0.75).otherwise(col("AVERAGE_COST_FOR_TWO")))\
.withColumn("CURRENCY",when(col("COUNTRY_CODE")==37,"Dollar($)").otherwise(col("CURRENCY")))\
.withColumn("AVERAGE_COST_FOR_TWO",when(col("COUNTRY_CODE")==184,col("AVERAGE_COST_FOR_TWO")*0.75).otherwise(col("AVERAGE_COST_FOR_TWO")))\
.withColumn("CURRENCY",when(col("COUNTRY_CODE")==184,"Dollar($)").otherwise(col("CURRENCY")))\
```

#Convert CUISINES column datatype comma separated to array

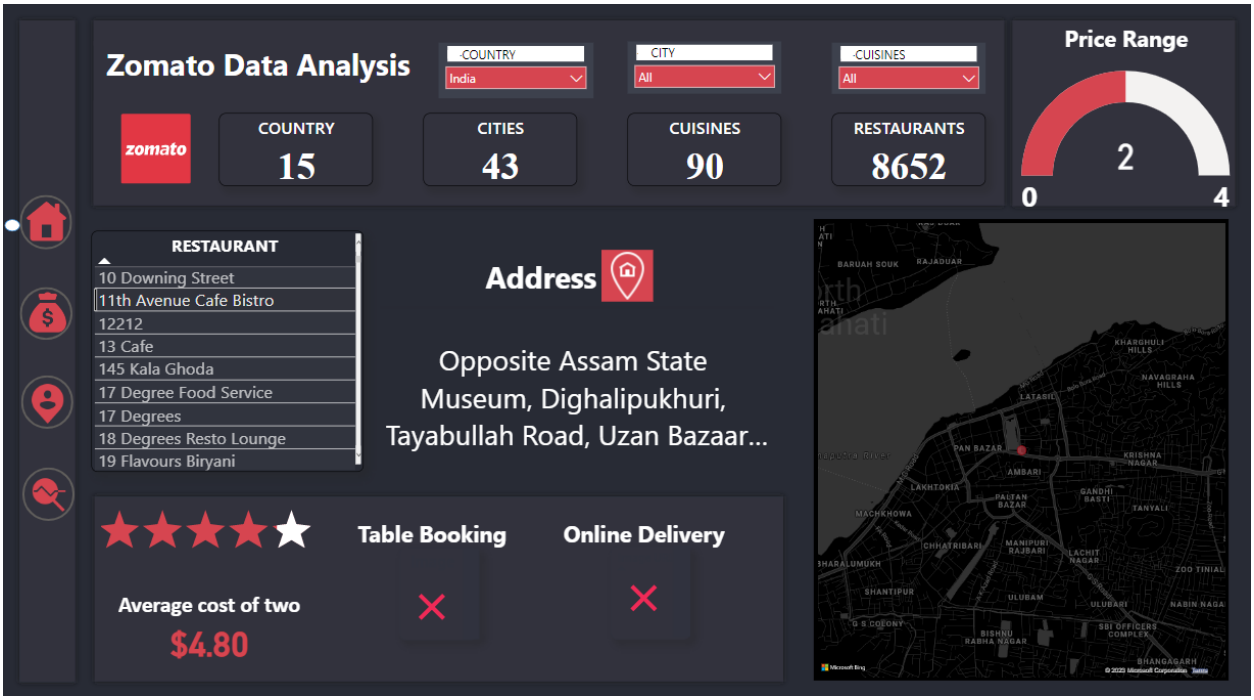
```
new_zomato_data=new_zomato_data.select(col('RESTAURANT_ID'),col('RESTATURANT_NAME'),
col('CITY'),col("ADDRESS"),
col("LOCALITY"),col("LOCALITY_VERBOSE"),col("LONGITUDE"),col("LATITUDE"),split(col('CUISINES'),lit(',')).alias("CUISINES"),col("AVERAGE_COST_FOR_TWO"),col("CURRENCY"),col('HAS_TABLE_BOOKING'),col('HAS_ONLINE_DELIVERY'),col("PRICE_RANGE"),col("AGGRE
```

```
GATE_RATING"),col("RATING_COLOR"),col("RATING_TEXT"),col("VOTES"),col("COUNTRY_
CODE"),col("COUNTRY_NAME"),col("UNIQUE_KEY"))

    #Split the CUISINES Column into separated rows
  new_zomato_data=new_zomato_data.flatten(new_zomato_data["CUISINES"])


    #Create a final Dataframe

new_zomato_data=new_zomato_data.select(col('RESTAURANT_ID'),col('RESTATURANT_NA
ME'),col('CITY'),
col("ADDRESS"),col("LOCALITY"),col("LOCALITY_VERBOSE"),col("LONGITUDE"),col("LATIT
UDE"),trim(regexp_replace(col('VALUE'),"\"","")).alias("CUISINES"),col("AVERAGE_COST_FO
R_TWO"),col("CURRENCY"),col('HAS_TABLE_BOOKING'),col('HAS_ONLINE_DELIVERY'),col
("PRICE_RANGE"),col("AGGREGATE_RATING"),col("RATING_COLOR"),col("RATING_TEXT"
),col("VOTES"),col("COUNTRY_CODE"),col("COUNTRY_NAME"),col("UNIQUE_KEY"))

  #Save a transform data in the table
  new_zomato_data.write.mode("append").save_as_table(zomato_clean_data_table)


#Create a view for the requirement field

session.table(zomato_clean_data_table).select(col("RESTAURANT_ID"),col("RESTATURANT_
NAME"),
col("CITY"),col("ADDRESS"),col("LONGITUDE"),col("LATITUDE"),col("LOCALITY"),col("CUISI
NES"),col("COUNTRY_CODE"),col("COUNTRY_NAME"),col("AVERAGE_COST_FOR_TWO"),
col("CURRENCY"),col("HAS_TABLE_BOOKING"),col("HAS_ONLINE_DELIVERY"),col("AGGR
EGATE_RATING"),col("PRICE_RANGE"),col("VOTES")).create_or_replace_view("zomato_anal
ysis_view")


  session.table(zomato_raw_data_table).delete()


  return "SUCCESS"

$$;

CALL ZOMATO_PROC('zomato_raw_data','country','zomato_data')
```

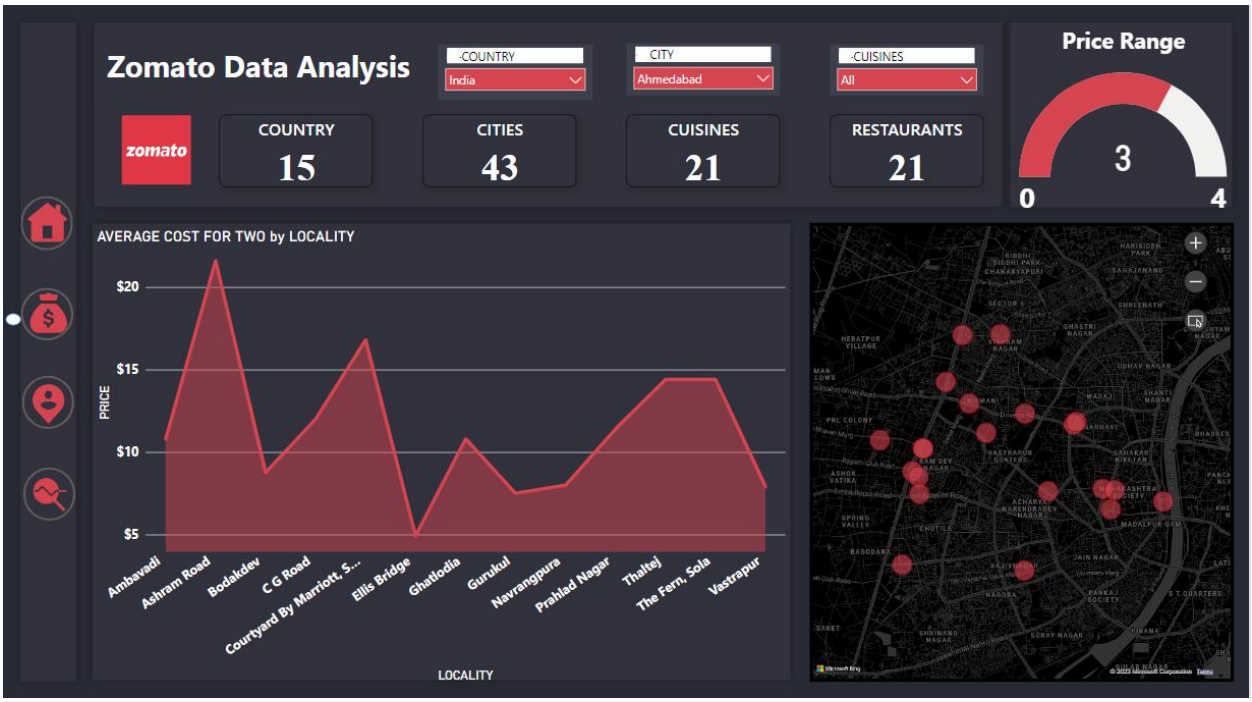
Output

We can see the Data Insight on Power BI.

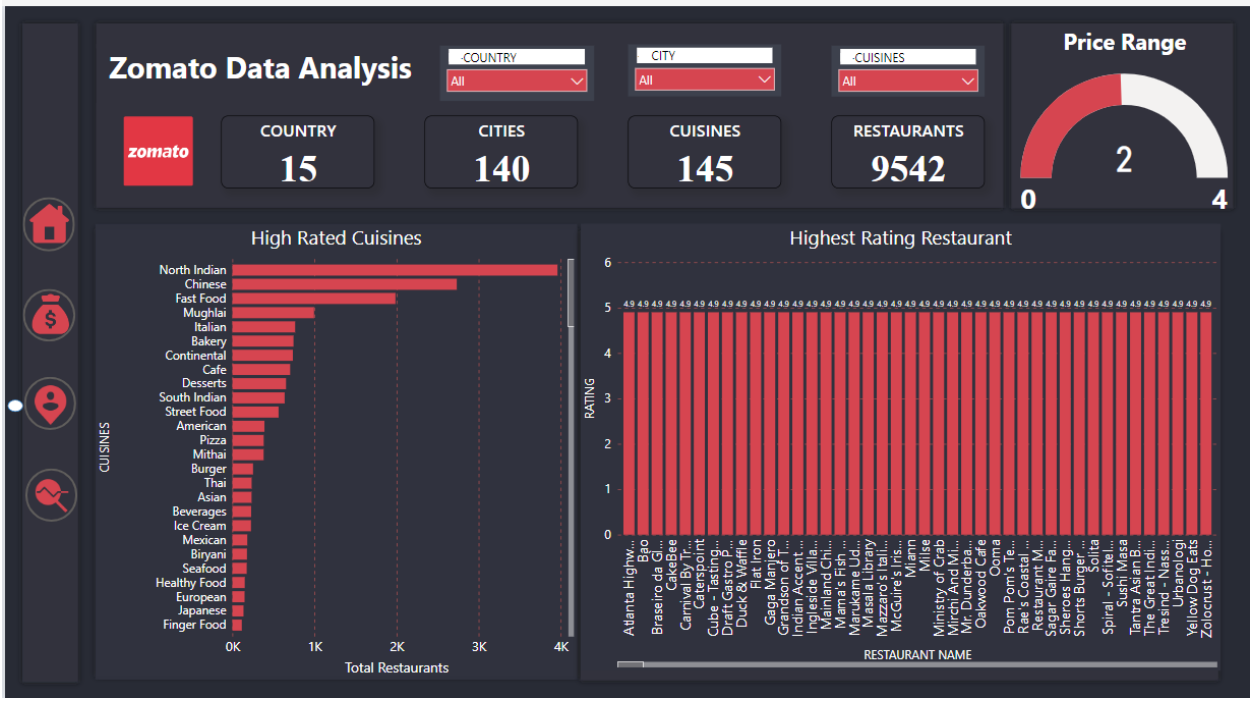
1.1. Home Page



1.2. Average cost for 2 by locality



1.3. Best Restaurants by multiple cities



2. Observation

2.1. Most famous cuisines in Ahmedabad

ZOMATO_ANALYSIS.PUBLIC

```
1  select count(RESTAURANT_ID),CUISINES from zomato_data where CITY='Ahmedabad' group by CUISINES order by count(restaurant_id) DESC;
2  select count(RESTAURANT_ID),CUISINES from zomato_data where COUNTRY_NAME='India' group by CUISINES order by count(restaurant_id) DESC;
3
4  select DISTINCT(RESTATURANT_NAME),AVERAGE_COST_FOR_TWO,CURRENCY from zomato_data where LOCALITY='Bodakdev' order by AVERAGE_COST_FOR_TWO DESC;
5
6  select RESTAURANT_NAME,AGGREGATE_RATING from zomato_data where cuisines='Western' order by AGGREGATE_RATING DESC;
7
```

Objects | Editor | Results | Chart

...	COUNT(RESTAURANT_ID)	CUISINES
	12	Continental
	10	Italian
	9	North Indian
	6	Desserts
	5	Cafe
	5	Chinese
	4	Fast Food
	3	South Indian
	3	Mexican
	3	Pizza
	3	Beverages
	2	Asian

2.2. Trending cuisines


```
select count(RESTAURANT_ID),CUISINES from zomato_data where CITY='Ahmedabad' group by CUISINES order by count(restaurant_id) DESC;
select count(RESTAURANT_ID),CUISINES from zomato_data where COUNTRY_NAME='India' group by CUISINES order by count(restaurant_id) DESC;

select DISTINCT(RESTATURANT_NAME),AVERAGE_COST_FOR_TWO,CURRENCY from zomato_data where LOCALITY='Bodakdev' order by AVERAGE_COST_FOR_TWO DESC;

select RESTATURANT_NAME,AGGREGATE_RATING from zomato_data where cuisines='Western' order by AGGREGATE_RATING DESC;
```

Objects | Editor | Results | Chart

COUNT(RESTAURANT_ID)	CUISINES
3,946	North Indian
2,690	Chinese
1,963	Fast Food
992	Mughlai
726	Bakery
724	Continental
682	Italian
631	South Indian
627	Cafe
597	Desserts
554	Street Food
380	Mithai

2.3. Average cost of 2 by locality

	RESTATURANT_NAME	AVERAGE_COST_FOR_TWO	CURRENCY
1	Mocha	12	Dollar($)
2	Fozzie's Pizzaiolo	10.8	Dollar($)
3	#REF!	9.6	Dollar($)
4	La Pino'z Pizza	6	Dollar($)
5	Huber & Holly	3.6	Dollar($)

2.4. High Rated Restaurant based on cuisines


```
5
6  select RESTATURANT_NAME,AGGREGATE_RATING from zomato_data where cuisines='Western' order by AGGREGATE_RATING DESC;
7
```

Objects | Editor | Results | Chart

	RESTATURANT_NAME	...	AGGREGATE_RATING
1	Union Deli		4.6
2	Satoo - Hotel Shangri-La		4.6
3	Toodz House		4.6
4	Lucky Cat Coffee & Kitchen		4.3
5	Avec Moi Restaurant and Bar		4.3
6	MONKS		4.2
7	Noah's Barn Coffeenery		4.2
8	Fish Streat		4
9	Fish Streat		3.4