



Group 6

Yelp Business Intelligence Cloud Subscription Service

Group member::

Sheng Luo | Rocky D. Wang | Zhiyuan Liu | Yao Zhang | Harvey Wang

Based on **yelp**  Customer's Review

Background and Use case

Background

- Importance of online reviews:
 - Online reputation is crucial in today's digital world
 - Yelp is a popular platform for rating and reviewing businesses
- Impact on restaurants:
 - Positive reputation drives more customers and revenue
 - Negative reputation deters potential customers

Use case

- Automatic monthly report generation
- Analyzes Yelp review data to help restaurant owners
- Provides insights into customer preferences, opinions, and concerns
- Allows benchmarking against competitors and market trends

Team Member's Responsibilities

01

Sheng Luo - S3: Store Raw and application data in storage

02

Yao Zhang - Lambda & Cloudwatch: To implement event-based cloud computation and services.

03

Zhiyuan Liu - Sagemaker: To preprocess data and train models for application.

04

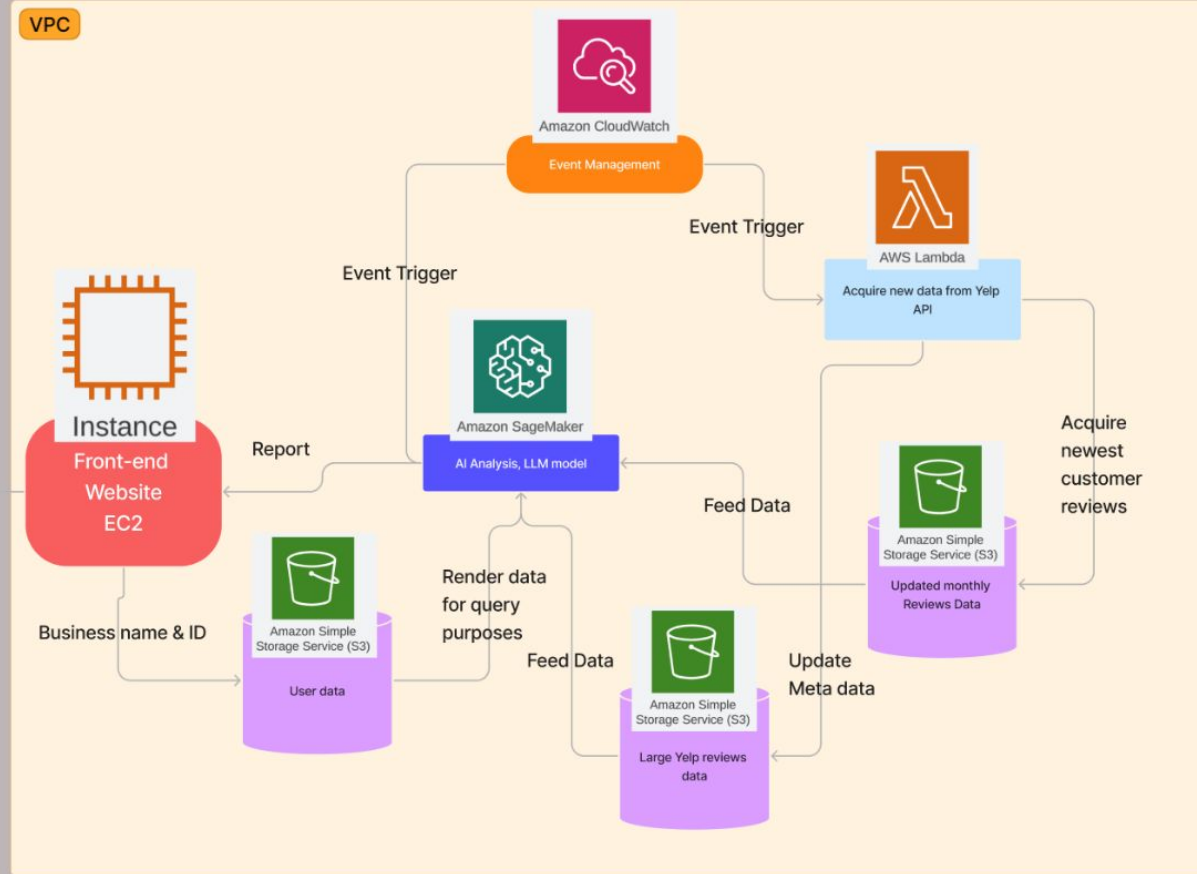
Harvey Wang - EC2: To collect user information

05

Rocky Wang - VPC: To Manage the accessibility and security.



Cloud Architecture Design



Data source and details

Yelp datasets - 5 GB

[This dataset is a subset of Yelp's businesses, reviews, and user data provided by Kaggle.](#)

Dataset Composition

- Information about businesses across 8 metropolitan areas in the USA and Canada
- Focus on local businesses: restaurants, bars, retail stores, etc.
- User-generated reviews of local businesses
- Users rate businesses on a scale of 1 to 5 stars and write detailed reviews

- 
- `yelp_business_attributes.csv`
 - `yelp_business_hours.csv`
 - `yelp_business.csv`
 - `yelp_review.csv`
 - `yelp_tip.csv`
 - `yelp_user.csv`

Virtual Server by EC2

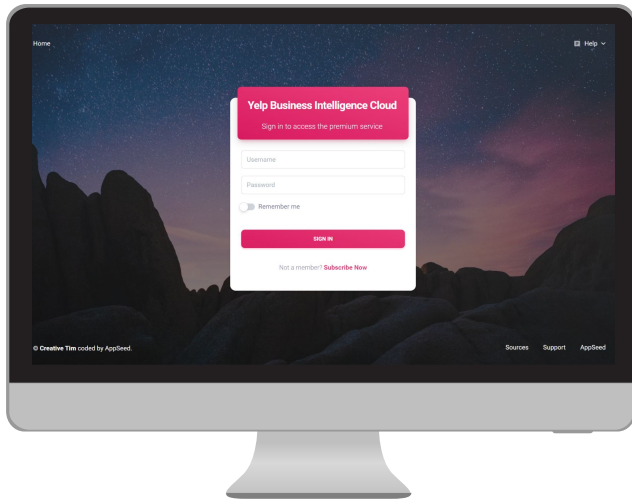
When new subscriber
is added

- **Website Hosting:**

EC2 hosts the front-end website, allowing users to register and log in to their accounts.

- **Integration with AWS Services:**

EC2 integrates seamlessly with other AWS services.



- **Management and Monitoring:**

EC2 provides various tools and APIs for managing and monitoring your instances.

- **Pay-as-you-go Pricing:**

EC2 follows a pay-as-you-go pricing model, where you pay only for the compute resources you consume.

EC2 Live Demo: <http://44.204.87.217:5000/login>

Home

Help ▾

Yelp Business Intelligence Cloud

Enter your email and password to register

☒ I agree to be **charged \$30 monthly**

PAY NOW

Already have an account? [Sign IN](#)

Yelp Business Intelligence Cloud

Sign in to access the premium service

☐ Remember me

SIGN IN

Not a member? [Subscribe Now](#)

userdb/

Objects

Properties

Objects (1)

Objects are the fundamental entities stored in Amazon S3. [Y](#)
[more](#) [↗](#)



Copy S3 URI



Copy URL



Find objects by prefix



Name



Type



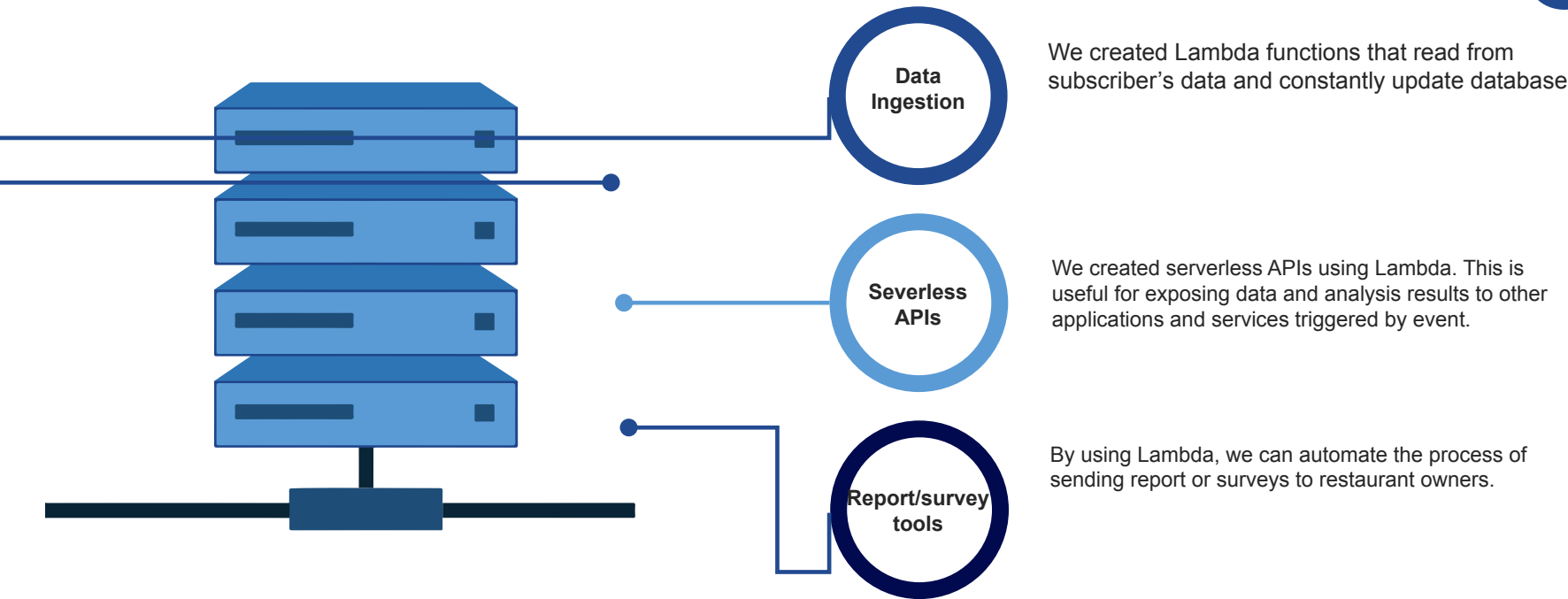
db.sqlite3

sqlite3

Event Based Pipeline

by Lamda and Event Bridge(Cloudwatch Event)

Monthly API request



monthlyUpdate

▼ Function overview [Info](#)



monthlyUpdate



Layers

(1)



EventBridge (CloudWatch Events)

+ Add trigger

voc-codebuild-cw-rule

Rule details [Info](#)

Rule name
voc-codebuild-cw-rule

Description
monthly update

Status
🟢 Enabled

Rule ARN
arn:aws:events:us-east-1:925398600691:rule/voc-codebuild-cw-rule

[Event schedule](#)

[Targets](#)

[Monitoring](#)

[Tags](#)

Event schedule [Info](#)

Fixed rate of
30 day

File Edit Find View Go Tools Window

Test

Deploy

Go to Anything (Ctrl-P)

lambda_function ×

Execution results ×



monthlyUpdate /
lambda_function.py

```
1 import json
2 import sqlite3
3 import requests
4 import json
5 import boto3
6 import os
7 import tempfile
8
9 s3 = boto3.client('s3')
10 bucket_name = '5450-bestguy-s3'
11 object_key = 'userdb/db.sqlite3'
12
13 # Download the file to a temporary location
14 with tempfile.NamedTemporaryFile() as tmp_file:
15     s3.download_file(bucket_name, object_key, tmp_file.name)
16
17     # Connect to the SQLite database in the temporary file
18     conn = sqlite3.connect(tmp_file.name)
19
20 # Connect to database and get businessID
21 # conn = sqlite3.connect('db.sqlite3')
22 cursor = conn.cursor()
23 # Execute SQL query to get all the business IDs
24 cursor.execute("SELECT businessID FROM Users")
25 results = cursor.fetchall()
26
27
28 reviews = []
29 for row in results:
30
31     BusinessId_OR_Alias = row[0]
32
33     # Pull the most recent reviews for the business
34     ClientID = 'cxyOXT_x8D6anIGTaodKjw'
```

Test/One time Running the Function

File Edit Find View Go Tools Window Test Deploy

Go to Anything (Ctrl-P)

Environment

monthlyUpdate /

lambda_function.py

Execution results

Test Event Name

test

Response

```
{
  "statusCode": 200,
  "body": "\"Hello from Lambda!\""
}
```

Function Logs

```
NQF7wgb3mHfMPzy5owzxxQ fetched!
NQF7wgb3mHfMPzy5owzxxQ fetched!
NQF7wgb3mHfMPzy5owzxxQ fetched!
X18400qgC5MKvePbsSe-nA fetched!
X18400qgC5MKvePbsSe-nA fetched!
X18400qgC5MKvePbsSe-nA fetched!
1FDInNa5N1fLnWF8d6JN2Q fetched!
1FDInNa5N1fLnWF8d6JN2Q fetched!
1FDInNa5N1fLnWF8d6JN2Q fetched!
_eWAW26123n17F5c1o030w fetched!
_eWAW26123n17F5c1o030w fetched!
_eWAW26123n17F5c1o030w fetched!
KTB_JzB6uQDs5s5cS65XrMA fetched!
KTB_JzB6uQDs5s5cS65XrMA fetched!
KTB_JzB6uQDs5s5cS65XrMA fetched!
```

START RequestId: 633014cf-6349-4f06-94f5-336590317d40 Version: \$LATEST

END RequestId: 633014cf-6349-4f06-94f5-336590317d40

REPORT RequestId: 633014cf-6349-4f06-94f5-336590317d40 Duration: 1.13 ms Billed Duration: 2 ms Memory Size: 128 MB Max Memory Used: 81 MB Init Duration: 1615.53 ms

Request ID

633014cf-6349-4f06-94f5-336590317d40

Status: Succeeded Max memory used: 81 MB Time: 1.13 ms

userdb/

Objects Properties

Objects (1)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3](#) to store and retrieve objects.

Copy S3 URI Copy URL

Find objects by prefix

<input type="checkbox"/>	Name	Type
<input type="checkbox"/>	db.sqlite3	sqlite3

monthlyReviewUpdate/

Objects Properties

Objects (1)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3](#) to store and retrieve objects.

Copy S3 URI Copy URL Download

Find objects by prefix

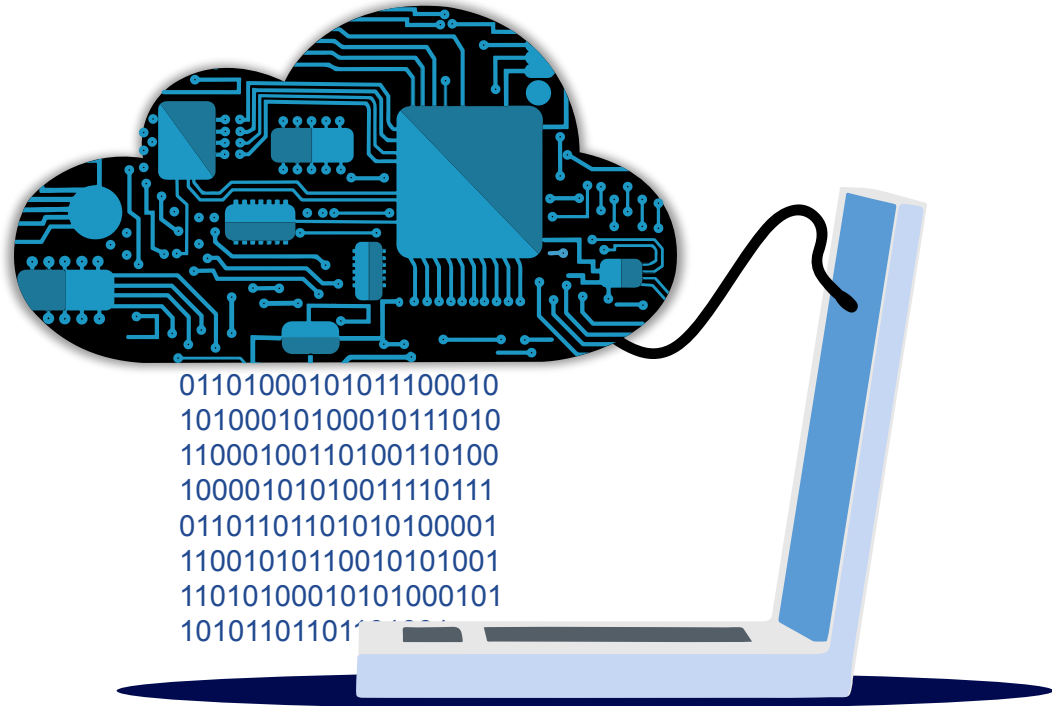
<input type="checkbox"/>	Name	Type
<input type="checkbox"/>	reviews.json	json

Data Storage

by S3

Store Raw data in Amazon S3

- Upload the raw data
 - Create a new bucket.
 - Configure the bucket permissions
 - Allow Lambda or Sagemaker to access it
-
- Sentiment analysis
 - Store the processed data in S3 Amazon
 - SageMaker for further analysis or visualization



datasets/

Copy S3 URI

Objects

Properties

Objects (6)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

Copy S3 URI Copy URL Download Open Delete Actions Create folder Upload

Find objects by prefix

<input type="checkbox"/>	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	yelp_business_attributes.csv	csv	April 25, 2023, 16:33:14 (UTC-04:00)	39.5 MB	Standard
<input type="checkbox"/>	yelp_business_hours.csv	csv	April 25, 2023, 16:33:18 (UTC-04:00)	13.2 MB	Standard
<input type="checkbox"/>	yelp_business.csv	csv	April 25, 2023, 16:33:14 (UTC-04:00)	30.3 MB	Standard
<input type="checkbox"/>	yelp_review.csv	csv	April 25, 2023, 09:43:57 (UTC-04:00)	3.5 GB	Standard
<input type="checkbox"/>	yelp_tip.csv	csv	April 25, 2023, 16:33:15 (UTC-04:00)	141.2 MB	Standard
<input type="checkbox"/>	yelp_user.csv	csv	April 25, 2023, 16:33:14 (UTC-04:00)	1.3 GB	Standard

5450-bestguy-s3

Info

Objects Properties Permissions Metrics Management Access Points

Bucket metrics

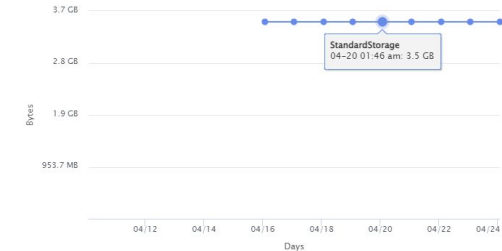
Explore metrics for usage, request, and data transfer activity within your bucket. Metrics are also available in Amazon CloudWatch. [Learn more](#)

1d 1w 2w

Total bucket size

Amount of data in bytes stored in this bucket.

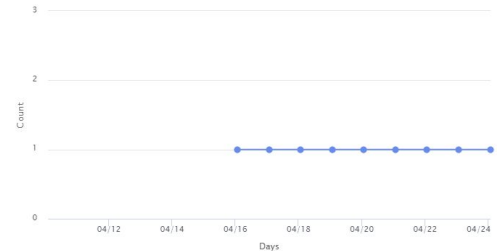
[View in CloudWatch](#)



Total number of objects

Total number of objects stored in this bucket for all storage classes.

[View in CloudWatch](#)



S3 Datasets And Metrics

With AWS S3 as the data storage solution, data is automatically replicated across multiple availability zones, which provides resiliency and ensures data availability. Additionally, EC2 and SageMaker both offer automatic backup and recovery options, which further enhances resiliency.

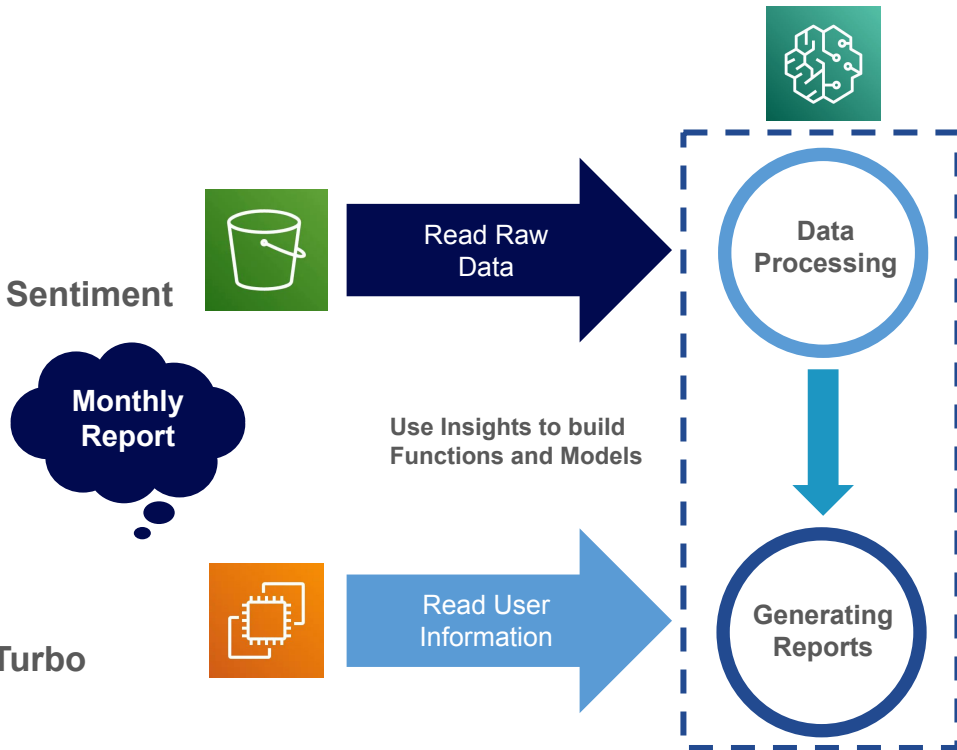


Accelerating Data Analysis

by SageMaker Notebook

- Data Processing
 - Raw Data: **Over 5GB** Yelp data
 - Analysis: **Competitor, Geographic and Sentiment**

- Generating Report:
 - Method: Python ``reportlab`` library
 - Visualization: **10+ graphs or charts**
 - Suggestions: Generating from **GPT 3.5 Turbo**



Sagemaker Screenshots

yelp-analysis

Delete

Stop

Start

Open Jupyter

Open JupyterLab

Notebook instance settings

Edit

Name yelp-analysis	Status ⌚ Pending	Notebook instance type ml.r5.xlarge	Platform identifier Amazon Linux 2, Jupyter Lab 3 (notebook-ai2-v2)
ARN arn:aws:sagemaker-us-east-1:925398600691:notebook-instance/yelp-analysis	Creation time Apr 25, 2023 15:32 UTC	Elastic Inference -	Minimum IMDS Version 2
Lifecycle configuration -	Last updated Apr 26, 2023 03:42 UTC	Volume Size 16GB EBS	

Git repositories

Name	Repository URL	Type
------	----------------	------

There are currently no resources.

Permissions and encryption

IAM role ARN arn:aws:iam::925398600691:role/LabRole 🔗	Root access Disabled	Encryption key arn:aws:kms:us-east-1:925398600691:key/bc7a15c9-0700-47c4-ab4e-46bf269fb02c 🔗
--	-------------------------	---

Network

Subnet(s)
subnet-046d3f867924c593a

Security Group(s)
sg-0693a103c2c26eaaa

Direct internet access
Disabled: [Learn more](#) [🔗](#)

Sagemaker Screenshots



```
# Section 2: Review Analysis
elements.append(Paragraph("2. Review Analysis", styles["Headings3"]))

elements.append(Paragraph("The following charts provide insights into the business's review trends, including star rating distribution, and the 'useful', 'funny', and 'cool' properties of the reviews. By analyzing these charts, users can better understand the business's performance and areas for improvement.", custom_style))

image_files = [
    (review_path[0], "Figure 1: Number of Reviews per Month",
     "This chart shows the total number of reviews received by the business each month."),
    (review_path[1], "Figure 2: Average Stars per Month",
     "This chart shows the average star rating given by customers each month."),
    (review_path[2], "Figure 3: Star Distribution",
     "This chart displays the distribution of star ratings received by the business."),
    (review_path[3], "Figure 4: Useful, Funny, and Cool Distribution",
     "This chart shows the distribution of the 'useful', 'funny', and 'cool' properties in the reviews."),
    (os.path.join(save_folder, "Afinn_Analysis.png"), "Figure 5: Sentiment Analysis using Afinn",
     "This bar chart shows the sentiment distribution using Afinn sentiment scores, categorized into Extreme Negative, Negative, Neutral, Positive, and Extreme Positive."),
    (os.path.join(save_folder, "NRC_Analysis.png"), "Figure 6: Sentiment Analysis using NRC",
     "This bar chart displays the frequency of NRC emotions detected in the reviews, providing insights into the emotional content of the reviews."),
    (os.path.join(save_folder, "Combined_Word_Clouds_with_Titles.png"),
     "Figure 7: Combined Word Clouds for Different Sentiment Categories",
     "This combined word cloud image shows the most frequently used words in different sentiment categories. It can help users identify the most common themes and key words associated with each sentiment category."),
    (os.path.join(save_folder, "Combined_Bar_Charts.png"),
     "Figure 8: Top 20 Words Frequency for Different Sentiment Categories",
     "This combined horizontal bar plot image shows the frequency of the top 20 words for different sentiment categories. It can help users compare the most common words across different sentiment categories.")
]

for image_file, title, description in image_files[0:2]:
    put_image(image_file, title, description, pagebreak=False)

for image_file, title, description in image_files[2:4]:
    put_image(image_file, title, description)

for image_file, title, description in image_files[4:6]:
    put_image(image_file, title, description)

for image_file, title, description in image_files[6:8]:
    put_image(image_file, title, description, height=600)

elements.append(Paragraph("GPT Analysis of Negative Reviews", styles["Heading4"]))

gpt_answer = response['choices'][0]['message']['content'].replace('\n', '<br>')

elements.append(Paragraph(gpt_answer, styles["BodyText"]))

# Add the copyright statement
copyright_statement = "© 2023 Bestguys Tech Ltd."
elements.append(Paragraph("para align=center spaceAfter=10><font size=12><br>{}/</font>{}/para".format(copyright_statement), styles["Normal"])))
```

Yelp In-Depth Report

"Gangnam Asian BBQ Dining" (hihud--QRriCYZw1zZvW4g)

1. Market Analysis

Competitive Density Analysis

Number of businesses within 5 km of business hihud--QRriCYZw1zZvW4g: 7613.

Number of similar businesses with similarity score > 0.3: 6

Key Competitors

- "Azusa Hookah Lounge & Cafe" (Similarity: 0.53)
- "Dog Haus" (Similarity: 0.53)
- "Joe's New York Pizza" (Similarity: 0.44)
- "Capriotti's Sandwich Shop" (Similarity: 0.44)
- "Stussy" (Similarity: 0.37)
- "High Roller Cigar & Smoke Shop" (Similarity: 0.37)
- "Lok Acupuncture Clinic" (Similarity: 0.30)
- "Djanel Spa" (Similarity: 0.30)
- "Pharaoh Beads" (Similarity: 0.30)
- "Oyster Bar" (Similarity: 0.30)

Table 1: Key Competitors Information

name	neighborhood	address	stars	review_count	is_open
"Azusa Hookah Lounge & Cafe"	Eastside	"4480 Paradise Rd"	4.5	239	1
"Dog Haus"	Eastside	"4480 Paradise Rd"	4.5	282	1
"Joe's New York Pizza"	Eastside	"4480 Paradise Rd"	3.5	384	1
"Capriotti's Sandwich Shop"	Eastside	"4480 Paradise Rd"	4.0	368	1
"Stussy"	Eastside	"4480 Paradise Rd"	3.5	25	0
"High Roller Cigar & Smoke Shop"	Eastside	"4480 Paradise Rd, Ste 250"	2.0	9	1
"Lok Acupuncture Clinic"	Eastside	"1818 E Desert Inn Rd"	4.5	34	1
"Djanel Spa"	Eastside	"372 E Tropicana Ave"	4.5	21	1
"Pharaoh Beads"	Eastside	"3528 S Maryland Pkwy"	4.5	10	1
"Oyster Bar"	Eastside	"4455 Paradise Rd"	4.5	61	1



VPC Resource Map and Security Group

Inbound rules : IPv4 HTTP TCP 80

Outbound rules : IPv4 All-traffic All All

sg-0693a103c2c26eaaa - Bestguy security group

Details | Inbound rules | Outbound rules | Tags

Details

Security group name
Bestguy security group

Security group ID
sg-0693a103c2c26eaaa

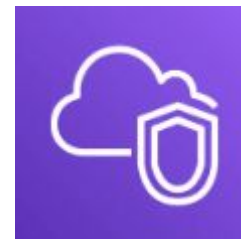
Description
Enable HTTP access

VPC ID
vpc-0f1b48a104d6bef89

Owner
925398600691

Inbound rules count
1 Permission entry

Outbound rules count
1 Permission entry



Cost Analysis

AWS COST - Here the upfront cost is **\$42.05** and the monthly cost is **\$40.12**. The long term cost for one year is **\$523.49**. Our selection of services includes EC2, S3, SageMaker, VPC.



Google Cloud Cost- The monthly cost is **\$52.17**. The long term cost for one year is **\$626.04**. Our selection of services contains Compute Engine, Cloud Storage, BigQuery ML, Cloud VPN

Success Criteria

01

Security

- ✓ Virtual Private Cloud (VPC) - a private network helps protect in transit data
- ✓ Security Groups - control inbound and outbound rules for our instance, separate security groups for different tiers in the future

02

Reliability

- ✓ Multiple availability zones - application is deployed across multiple data centers, ensuring high availability with a higher failure tolerance

03

Performance Efficiency

- ✓ Auto scaling - automatic scale computing resources to handle spikes or low demand

04

Cost Optimizer

- ✓ AWS Cost Explorer - identify opportunities for spending optimization
- ✓ AWS Budgets - control costs by setting a spending limits as well as receiving alerts when close to the limit

