



Lemuel Torrefiel Web Development and AI Specialist Portfolio



Confidentiality



- The purpose of this portfolio is to compile achievements from the past focusing only on UI snapshots.
- No data has been shared together with this portfolio to ensure clients' data privacy and security.
- Company logos are hidden in this portfolio.

My Tech Stacks



- In the world of AI, I have practiced several tech stacks including OpenAI API, Google Gemini API, Anthropic's Model Context Protocol and Claude.
- My main field before diving to AI Integrations are Data Analytics and Data Engineering. I have developed skills related to data analysis and database management. I want to dive into web development by utilizing what I have learned with my vast experiences in the field of data.
- My tech stacks in backend development are SQL, Mongodb, FastAPI, and Flask. Additionally, I have a lot of tools to offer from my past experiences with Data Analytics.
- I develop frontend apps with the help of AI tools. React Typescript is what I'm trying to learn so far.
- I am continuously learning to be a full stack developer. With the help of online tutorials and practical applications, I have developed some web apps that I want to share in this portfolio.

Home Tech Stacks Projects Video Introduction Certificates Contact

Hi, I'm Lemuel, Data Analyst and Data Engineer who aspires to dive into Web Development with AI integration.

Welcome to my Portfolio website. My vision is to utilize my past experiences with Data Analytics and Data Engineering to become an effective Web Developer with AI tools integrations.

[Download CV](#) [Download Portfolio](#)



lalala0095.github.io/my-portfolio-website/

A screenshot of a portfolio website titled "My Tech Stacks". The top navigation bar includes links for Home, Tech Stacks, Projects, Video Introduction, Certificates, and Contact. Below the title, a subtitle reads "Technologies I've worked with throughout my journey in data and web development". A "All Technologies" button is followed by a row of six circular icons representing different technology categories. The main content area is organized into three sections: "AI & API Integrations" featuring icons for OpenAI GPT, Google Gemini, Perplexity AI, and SERP APIs; "Frontend/Backend Web Development" featuring icons for Python FastAPI, React, Tailwind CSS, Python Flask, and Google AppSheet; and "Data Analytics" featuring icons for various data visualization and analysis tools.

Due to data confidentiality and data privacy, I have not been able to display the projects that benefited my former and existing clients. I may be able to talk about it in a separate meeting and give my prospects a brief description of those projects.

React JS + FastAPI - Travel Destination Finder



Full Stack Web Development with AI Backend

The objective of this web app is to assist users in their next potential destination based on their user-given preferences.

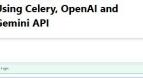
React JS - AdvPOS



Frontend Web Development

The objective of this app is to help users manage their expenses, bills and cash flows, to easily analyze their data based on the comparison of both. I envision this project to benefit community, by creating planner reports for their finance management.

FastAPI - Smart Task with Queue System Using Celery, OpenAI and Google Gemini API



Authentication

Frontend Web Development

Backend Web Development

The objective of this backend server is to assist the ongoing project of my client in a confidential project.

FastAPI - API Backend for AdvPOS



FastAPI - API Backend for AdvPOS

The objective of this API is to provide a robust backend for the AdvPOS system, handling various financial and operational requests efficiently.

Power BI - E-commerce Dashboard for Industrial Pipes Company



Power BI - E-commerce Dashboard for Industrial Pipes Company

This dashboard provides a comprehensive overview of the company's e-commerce operations, including sales trends, product performance, and customer behavior analysis.

Power BI - Real Estate Dashboard of London Sold Properties



Power BI - Real Estate Dashboard of London Sold Properties

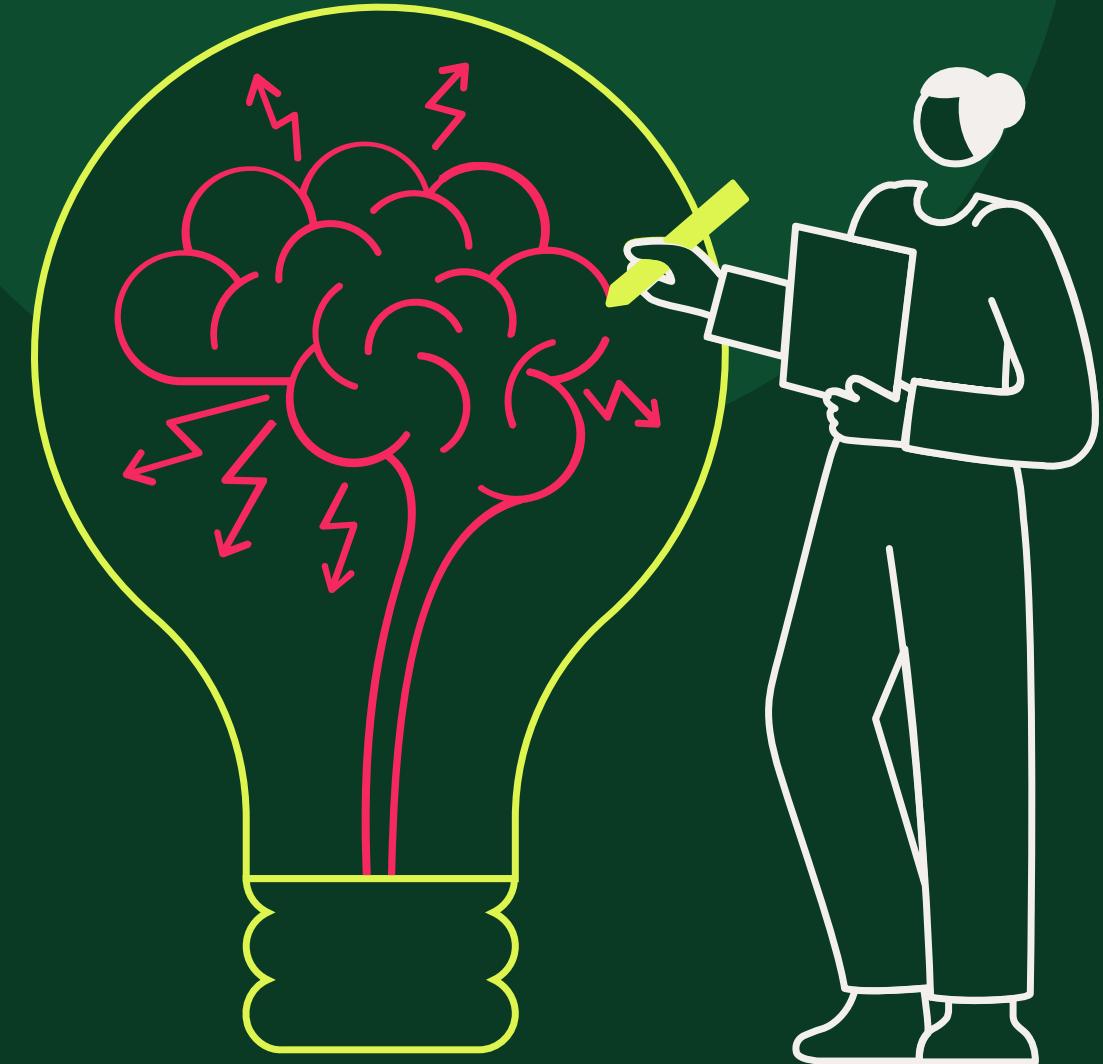
This dashboard tracks the real estate market in London, specifically focusing on sold properties, price trends, and geographical distribution.

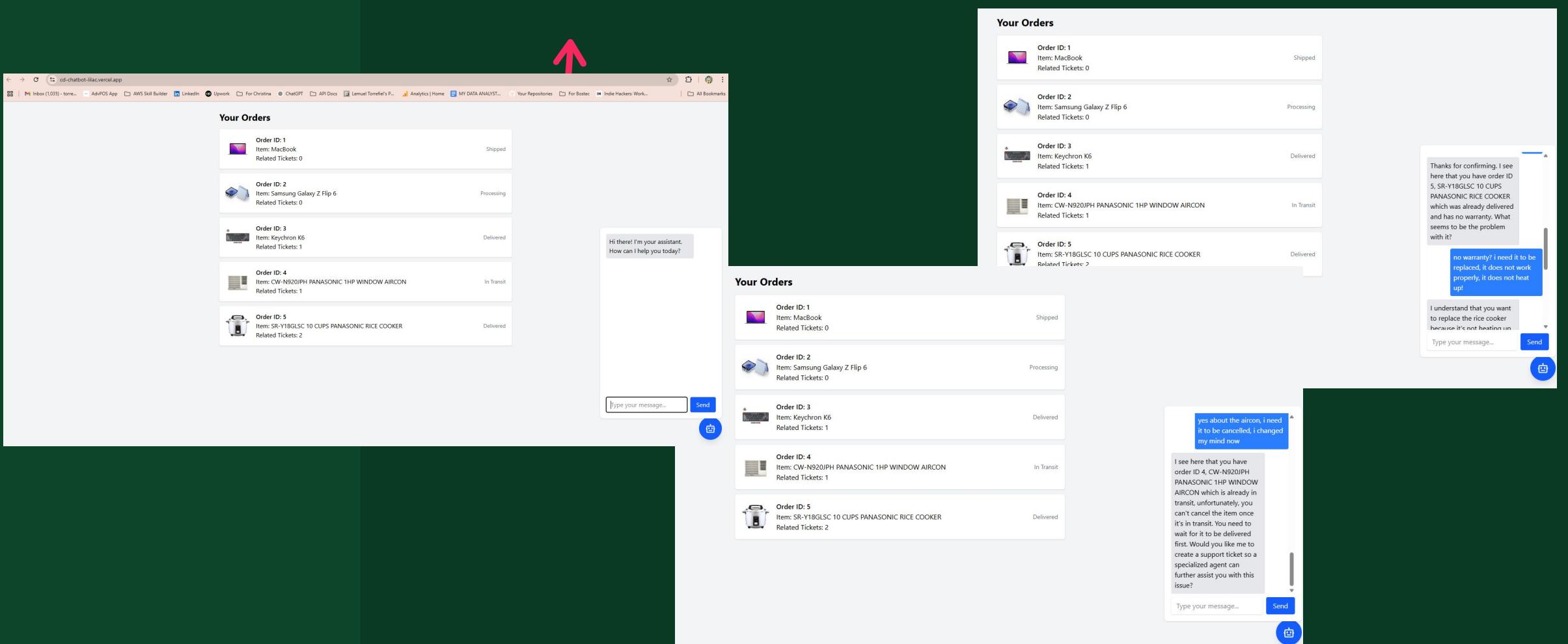
This is my portfolio website, deployed using Vercel.
You may view this link: [Lemuel Torrefiel's Portfolio](#)
This website contains more of my background information, projects and tech stacks.

Backend:
FastAPI, MongoDB,
Redis, OpenAI and
Google Gemini API

Frontend: React
TypeScript

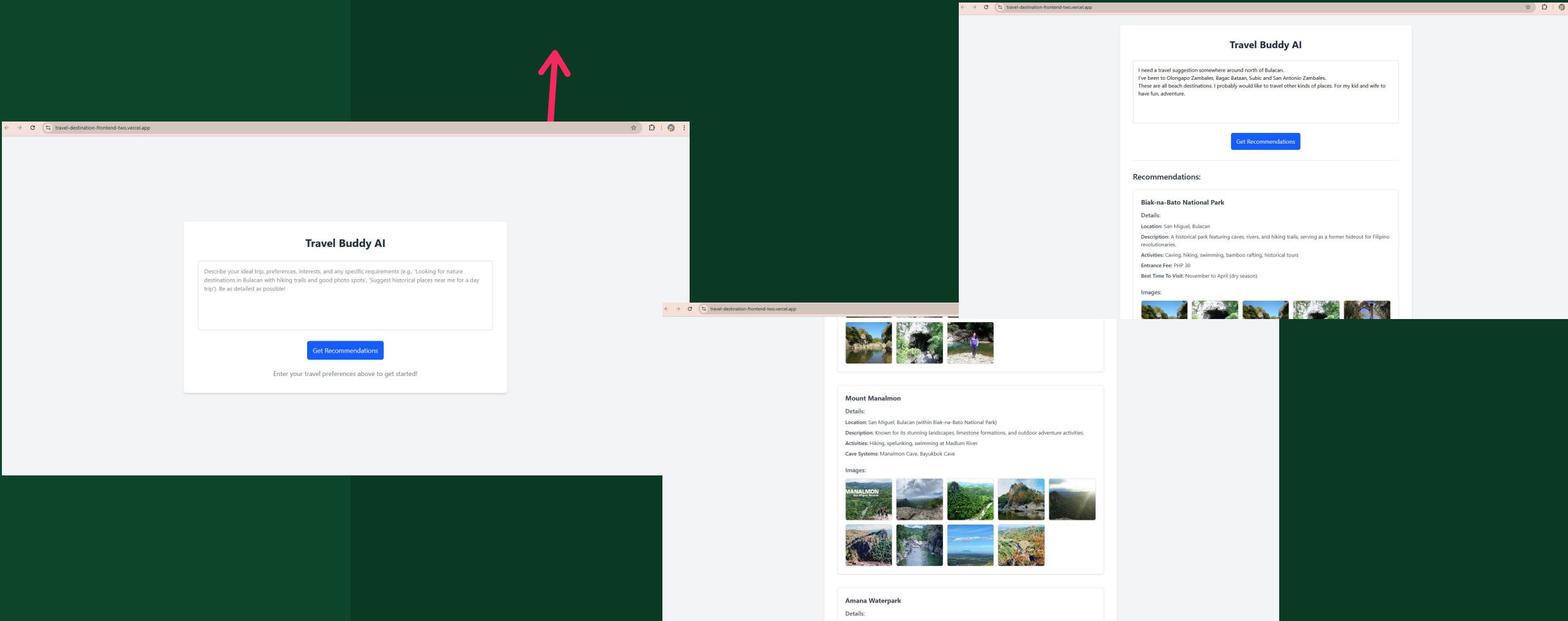
Deployment: AWS and
Vercel



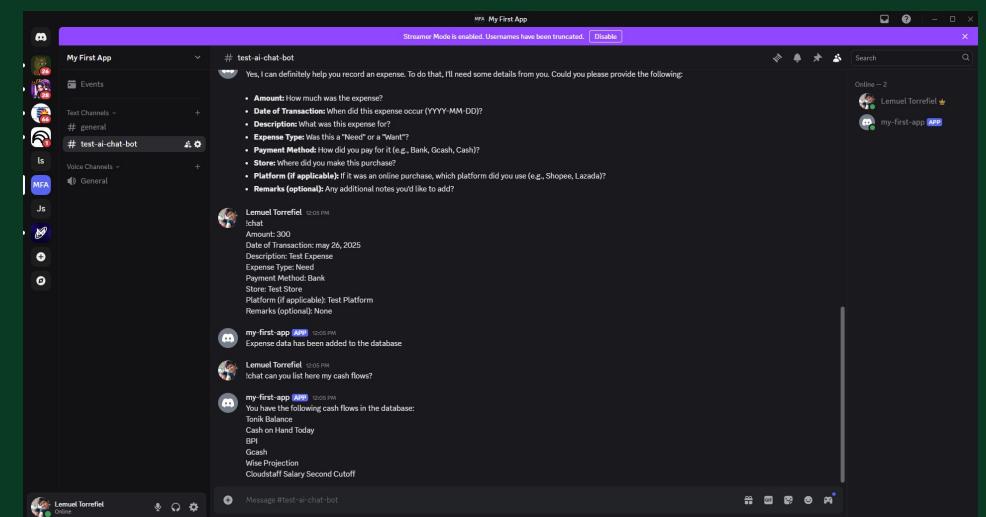
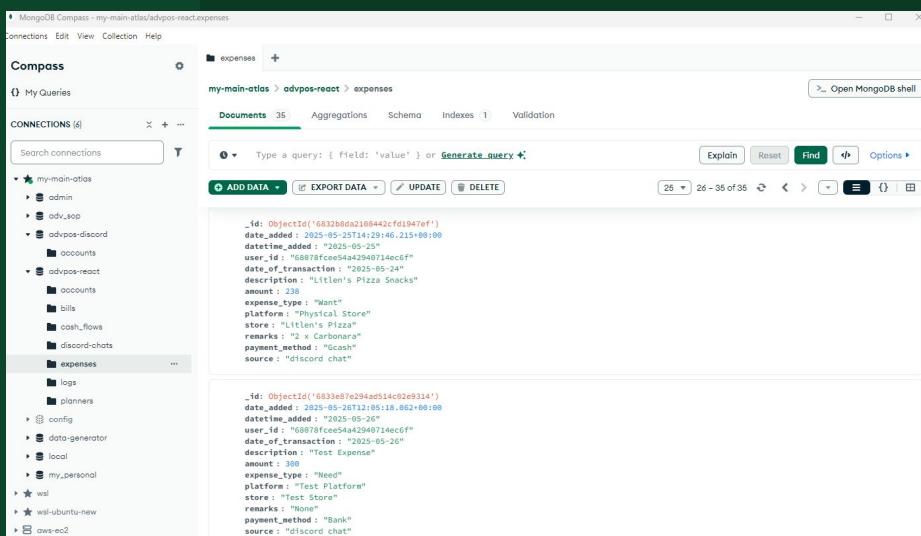
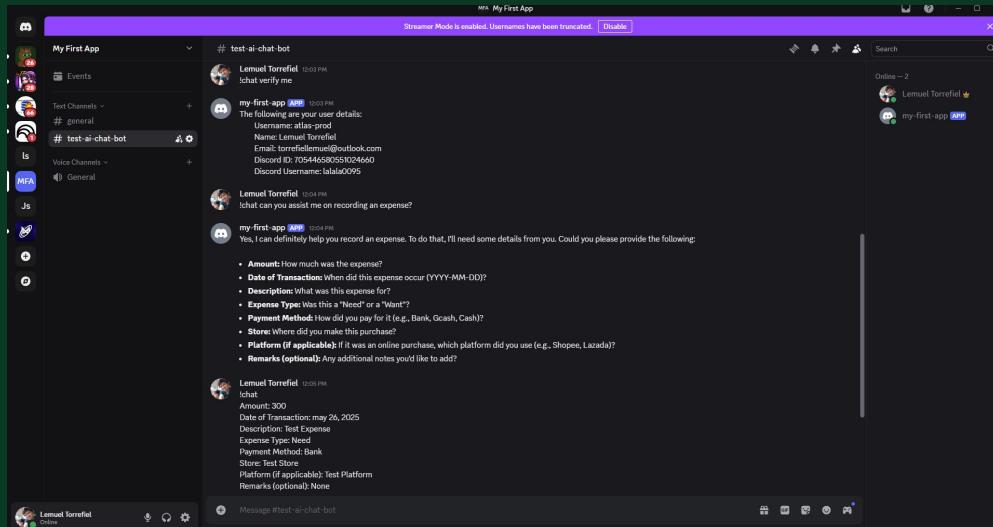
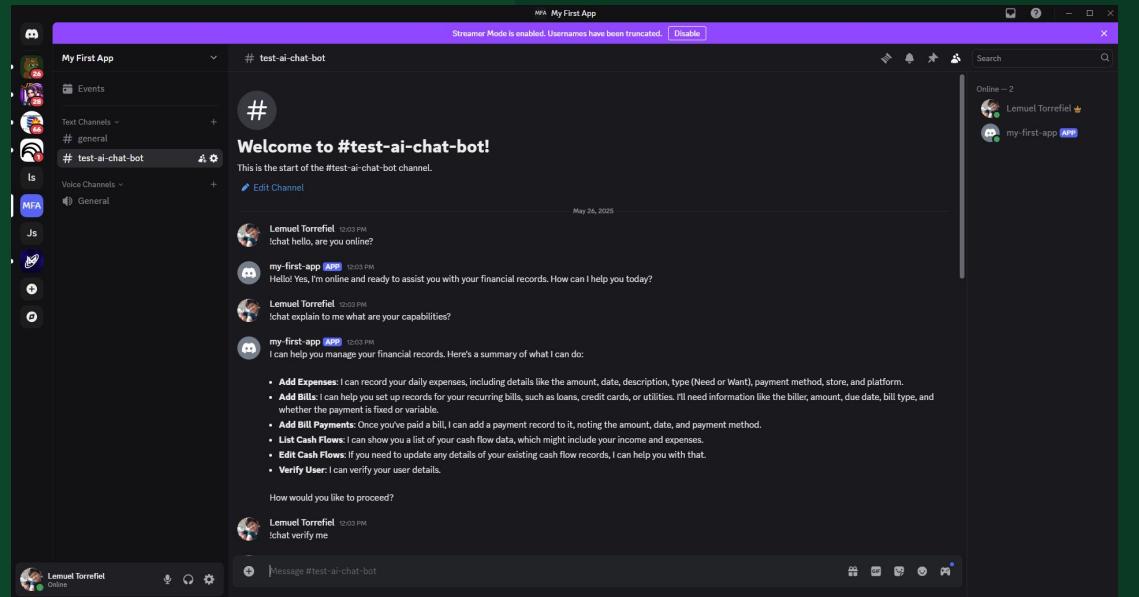


Customer Support Live Chat Bot: This project is a web app that mimics the functionality of an AI chat bot that is trained to be a Customer Support agent. The agent answers the inquiries of the user related to his/her orders like warranty, complaints and returns. The AI bot has also the capability to raise a ticket if the user wants to escalate the issue. The AI behind this is Google Gemini.

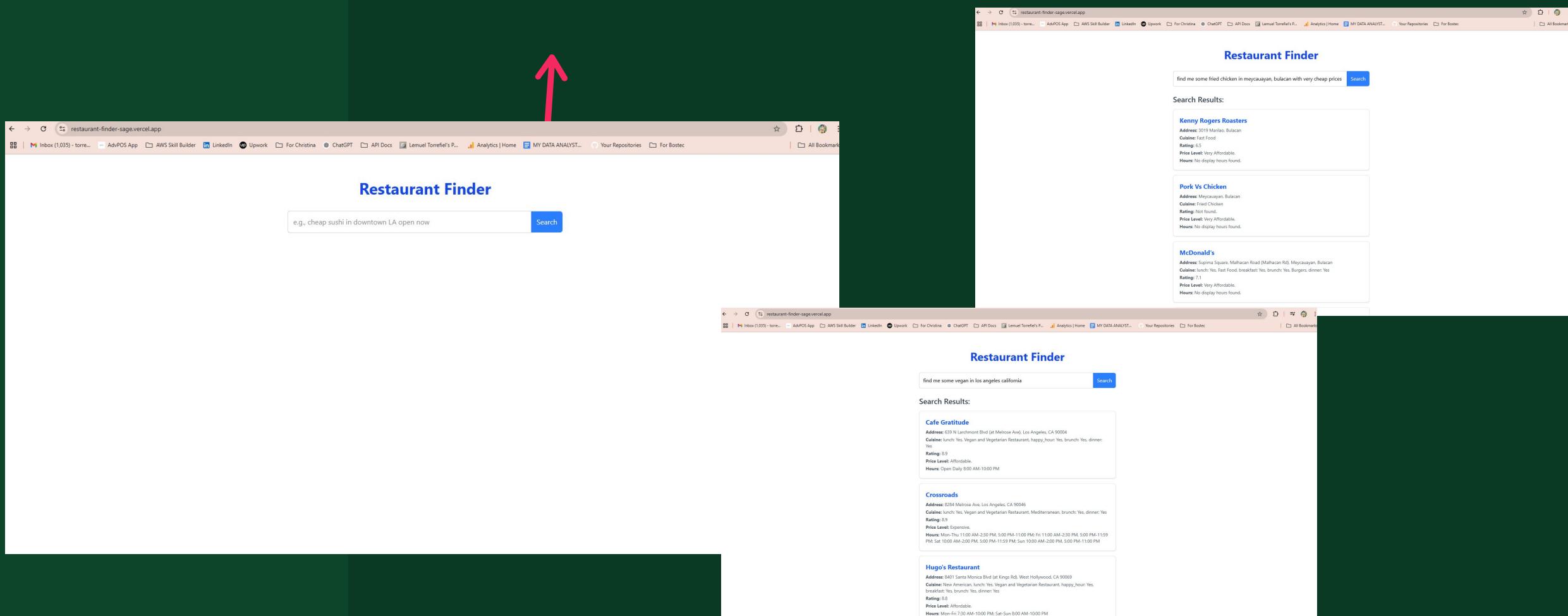
You may try refer to this link for a live demo: <https://cd-chatbot-lilac.vercel.app/>



Travel Destination Finder AI: This web app is a travel assistant buddy because it generates smart search results for potential travel destinations base on the user's prompt.
You may try this demo link: <https://travel-destination-frontend-two.vercel.app/>



Discord AI Bot: An AI Assistant that helps you manage financial metrics such as Expenses, Bills and Cash Flows.
For Live Demo, visit this link to join the server: <https://discord.gg/N9VvwBE3>



Restaurant Finder AI: This web app is dedicated for users to find relative restaurants or food stores base on their preferences. It uses Foursquare API to find relative restaurants base on the user's inquiry. The AI assistant behind this is Google Gemini, it processes the user into a prompt, generates results from Foursquare and displays them as categorized texts and images.

You may try this here in this live demo link: <https://restaurant-finder-sage.vercel.app/>



This is a web app to track Financial metrics such as Expenses and Cash Flows. The backend is deployed using AWS EC2 while the frontend is in Vercel. This project is still in progress, I am using this personally to help me track my financial metrics.

You may view this live demo link: <https://advpos-lemuel-torrefield-projects.vercel.app>

	Dec 23, 2024	Metrobank Borland Housing Loan 3	Rent	36807.69	Fix	12	hehhehehehehehh	
	Dec 23, 2024	Metrobank Borland Housing Loan	Loan	12036	Fix		hahaha	
	Dec 23, 2024	Biller 2	Credit Card	36807.69	Changing	10	hehehee	
	Dec 23, 2024	Billers Again	Loan	36807.69	Changing	10	remarks try	

Page 1 of 3 [Next](#) [Go to page: 1](#) [Set page limit: 10](#) [Go](#)

Cash Flows

Expenses

Settings

Add New Biller*

Biller Name*

Biller Type*

Select Biller Type ▾

Select Biller Type

Utilities

Rent

Loan

Credit Card

Investment

Insurance

Others

Amount Type*

Select Amount Type ▾

Remarks

Save

Logout

Sample data to demonstrate the table view with pop out add new data form.

[Home](#)[Billers](#)[Cash Flows](#)[Expenses](#)[Settings](#)

AdvPOS App

Manage Expenses

[Add New Expense](#)

Total Expenses: 8

Total Pages: 1

Expenses List

Date of Transaction	Description	Amount	Expense Type	Platform	Store	Remarks	Payment Method	Actions
2025-01-01	Baking Ingredients	36807.69	Need	Physical Store	Shopee - K BOX	test remarks	Cash	
2025-01-03	Aluminum Foil Tray edited	12036	Want	Tiktok	Shopee - K BOX edit	test edit	Cash edit	
2025-01-02	Baking Ingredients	12036	Need	Lazada	Shopee - K BOX	test remarks 2	Cash	

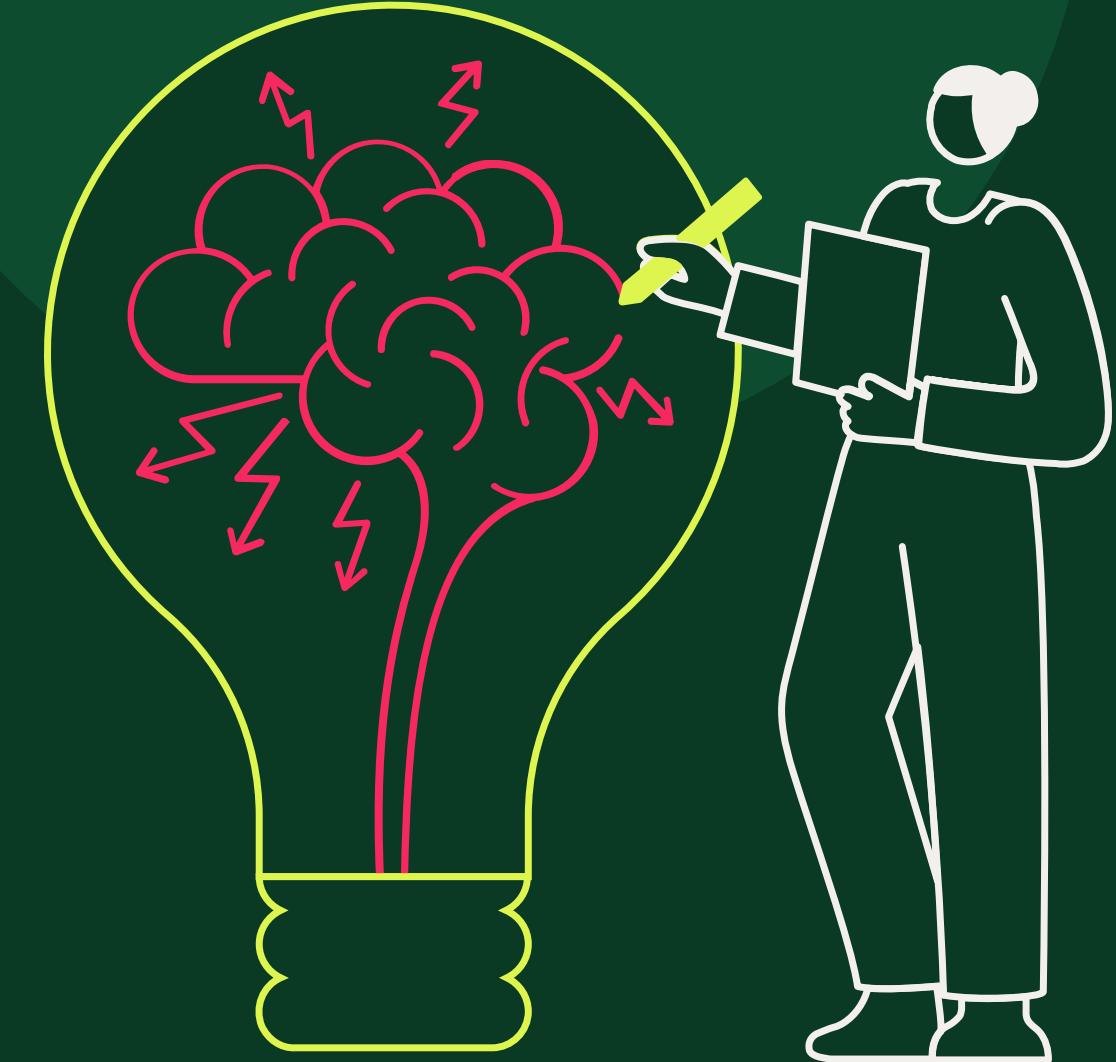
Page 1 of 1 Go to page: Set page limit: Go[Logout](#)

Another sample data view, this React frontend communicates with FastAPI server to perform other CRUD operations.

Backend:
Flask, Google Sheets
(GSpread), Docker

**Frontend: HTML and
CSS**

**Deployment: Azure
Web Apps**



← → ⌂ advpos-hxa6adhwfdbd2d9.southeastasia-01.azurewebsites.net/bills/bills_planners_view/675fdfb45898aa7e07e8cc64

Inbox (1,035) - to... Cognitive Class Great Learning AWS Skill Builder LinkedIn Islamps.unaux.c... Customize: Ls La... Upwork For Christina ChatGPT API Docs All Bookmarks

Adv POS

Dashboard

Cost of Goods ▾

Products ▾

Orders ▾

Finance Planning ▾

Feedbacks ▾

Admin Sections ▾

Users ▾

Summary

Date Created Dec 16, 2024

Total Bills: ₱86,005.31

Total Cash Flows: ₱78,695.12

Remaining Funds After Bills Payment: ₱7,310.19

Bills Overview

Bill Name	Amount	Allocation
[REDACTED]	₱10,000.00	₱10,000.00
UnionBank Personal Loan	₱7,443.05	₱7,443.05
Allowance	₱10,000.00	₱10,000.00
Metrobank Borland Housing Loan	₱12,036.00	₱12,036.00
MetroBank CC	₱9,499.59	₱9,499.59
RCBC CC	₱13,227.67	₱13,227.67
PSBank Car Loan	₱13,799.00	₱13,799.00
Allowance	₱10,000.00	₱10,000.00

Cash Flows

Cash Flow Name	Amount
Cloudstaff Salary Second Cutoff	₱36,723.69
Cash on Hand	₱2,200.00
Tonik Balance	₱35,069.09

This is the first version of the Adv POS app with the same objective of tracking and analyzing Expenses and Cash Flows. I also added Orders and Cost of Goods data for revenue calculation. This was developed with Flask.

Cost of Goods Records									
Date of Transaction	Description	Price	Type	Platform	Store	Payment Method	Remarks	Date Updated	Options
2024-12-05	Egg	₱500.00	Raw Materials	Physical Store	Minerva	Cash		2024-12-07	<button>Edit</button> <button>Delete</button>
2024-12-05	Foil	₱395.00	Packaging	Shopee	K BOX	Cash		2024-12-07	<button>Edit</button> <button>Delete</button>
2024-12-05	Box of Brownies	₱404.00	Packaging	Shopee	T-box	Cash		2024-12-07	<button>Edit</button> <button>Delete</button>
2024-12-07	Sugar 2 kilos	₱150.00	Raw Materials	Physical Store	Robinsons Guiguinto	Cash		2024-12-07	<button>Edit</button> <button>Delete</button>
2024-12-07	Foil	₱549.00	Packaging	Shopee	K BOX	Cash		2024-12-07	<button>Edit</button> <button>Delete</button>
2024-12-06	Bubbles Chocolate Chips	₱546.00	Raw Materials	Shopee	Milin_Store	Cash		2024-12-07	<button>Edit</button> <button>Delete</button>
2024-12-06	Aluminum Foil Tray	₱602.00	Packaging	Shopee	K BOX	Cash		2024-12-07	<button>Edit</button> <button>Delete</button>
2024-12-06	NECO Vanilla	₱134.00	Raw Materials	Shopee	pick2go	Cash		2024-12-07	<button>Edit</button> <button>Delete</button>
2024-12-06	Baking Ingredients	₱732.00	Raw Materials	Shopee	Your Ate PH	Cash		2024-12-07	<button>Edit</button> <button>Delete</button>
2024-12-06	Hand Mixer	₱189.00	Others	Shopee	HH1.SHOP	Cash		2024-12-07	<button>Edit</button> <button>Delete</button>

Sample data to demonstrate the data view.

Bills Deadlines Chat

Welcome to the Bills Tracker! Do you want to send bills data or view the report? (Type 'send' or 'report')

SEND

Record your bills data in this format for faster sending:

Biller:

Minimum Payment:

Statement Balance:

Due Date:

Paid or Not:

Biller: Test Biller
Minimum Payment: 1000
Statement Balance: 1000
Due Date: 1/6/2025
Paid or Not: Not

Bill recorded successfully! Send more if needed, or type 'report' to view the report.

report

Bill ID	Date	Biller ID	Biller	Minimum Payment	Statement Balance	Allocation	Due Date	Paid or Not	Add to Email	Month Cutoff	Cutoff	Payment	Payment Date
9ad49120	11/5/2024	PHP	RCBC	\$13,227.67	\$13,227.67	\$13,227.67	12/4/2024	Not	4	Next month	Second Cutoff		
746b56e3	11/5/2024	PHP	Metrobank Borland	\$12,036.00	\$12,036.00	\$12,036.00	12/4/2024	Not	4	Next month	Second Cutoff		
456e710a	10/30/2024	PHP	PSBank Car Loan	\$13,799.00	\$13,799.00	\$13,799.00	11/27/2024	Paid	27	Next month	Second Cutoff	\$13,799.00	11/23/2024
9c194926	10/30/2024	PHP	Unionbank Personal Loan	\$7,443.05	\$7,443.05	\$7,443.05	11/27/2024	Paid	27	Next month	Second Cutoff	\$7,443.05	11/23/2024
3f59096e	11/5/2024	PHP	Metrobank CC	\$1,000.00	\$10,995.32	\$10,995.32	11/25/2024	Paid	25	Next month	Second Cutoff	\$7,000.00	11/23/2024

Type your message here...

Send

This is the first version of the Bills tracker. I envisioned it to be a chat bot and during this timeline, I tried to start my web development learning with Flask.

FastAPI backend servers



Not secure 0.0.0:8000/docs

Inbox (1,035) - to... Cognitive Class Great Learning AWS Skill Builder LinkedIn Islamps.unaux.c... Customize: Ls La... Upwork For Christina ChatGPT API Docs All Bookmarks

My FastAPI App 1.0.0 OAS 3.1

/openapi.json

This is a custom OpenAPI schema for Adv POS App developed by Lemuel Torrefiel.

For Authorization, signup with the /api/v1/accounts/signup endpoint, then /api/v1/accounts/login to get your Bearer access token.

Include the access token for all of the REST requests the you'll be making.

Have fun using the service.

Not secure 0.0.0:8000/docs

Inbox (1,035) - to... Cognitive Class Great Learning AWS Skill Builder LinkedIn Islamps.unaux.c... Customize: Ls La... Upwork For Christina ChatGPT API Docs All Bookmarks

Admin

POST /api/v1/accounts/signup Admin Signup

POST /api/v1/accounts/login Login

GET /api/v1/accounts/protected Protected Route

POST /api/v1/accounts/logout Logout

GET /api/v1/accounts/get_options Get Options

Biller

POST /api/v1/billers/ Create Biller

GET /api/v1/billers/ Get Billers

PUT /api/v1/billers/{biller_id} Update Biller

DELETE /api/v1/billers/{biller_id} Delete Biller

GET /api/v1/billers/get_biller/{biller_id} Get Biller

GET /api/v1/billers/get_options Get Options

Reports

GET /api/v1/reports/dashboard Get Options

Bills

POST /api/v1/bills/ Create Bill

GET /api/v1/bills/ Get Bills

PUT /api/v1/bills/{bill_id} Update Bill

DELETE /api/v1/bills/{bill_id} Delete Bill

Support Tickets

POST /api/v1/support_tickets/ Create Support Ticket

POST /api/v1/support_tickets/test_post Test Post

This is the backend of the React web app. The routes are functional with user authentication. The server lives in AWS EC2 alongside the Mongodb database.

These are just snapshots of how I structure my codes. I develop my skills with the assistance of AI tools for productivity and rapid development. Python programming is my strength.

You may also view my Github profile: <https://github.com/lalala0095>

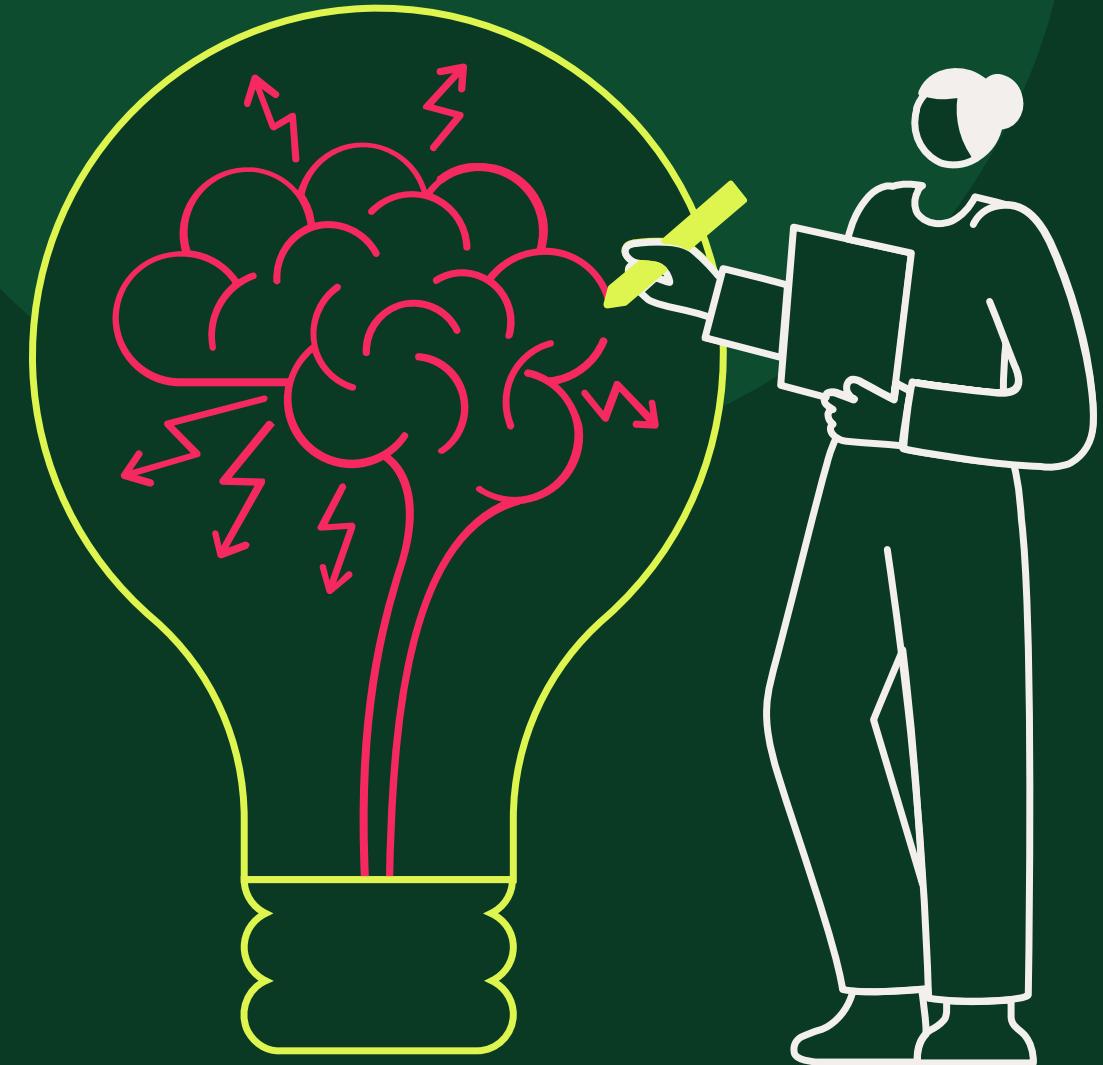


Lemuel Torrefiel

Data Analytics Portfolio



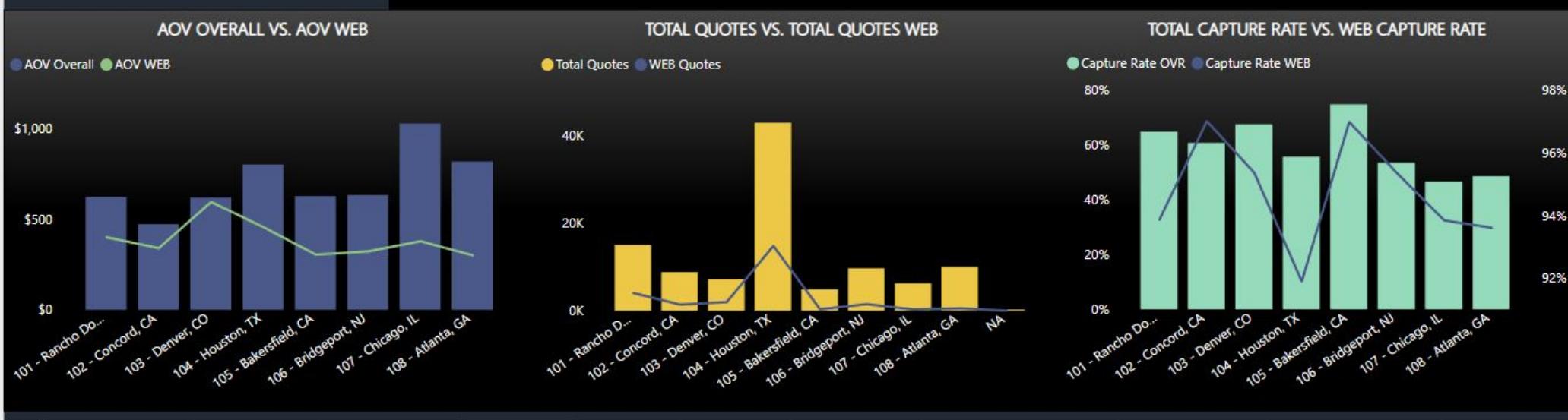
Data Analytics - Advanced Dashboard Developments in Power BI, Excel, and Google Sheets



E-COMMERCE - KPIs PER CUSTOMERS' BRANCH



\$43.21M	\$10.05M	23.27%
Sales Overall	Web Sales	WEB Sales %
60K	23K	38.17%
Total Orders	WEB Orders	WEB Orders %
\$714.25	\$886.40	435
AOV Overall	AOV Direct	AOV WEB
105K	25K	23.66%
Total Quotes	WEB Quotes	WEB Quotes %
57.72%	46.75%	93.10%
Capture Rate OVR	C Rate Direct	Capture Rate WEB
2023	Multiple	Multiple
Selected Year	Selected Quarter	Selected Month



[Power BI] E-commerce dashboard for sales monitoring.

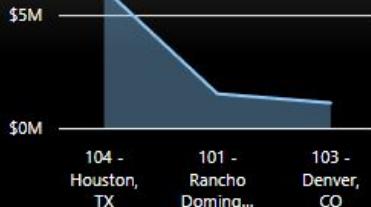
E-COMMERCE SAMPLE



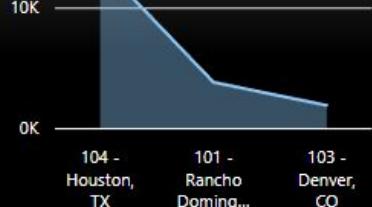
KPIS

\$10.05M
Web Sales 23K
WEB Orders 435
AOV WEB 25K
WEB Quotes 93.10%
Capture Rate WEB 381
Unq Buyers Web 60.60
WEB Frequency 0.28
WEB Recency 89.44%
UB Error Rate WEB

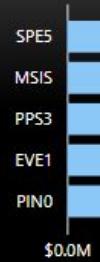
TOP 3 WEB SALES BRANCHES



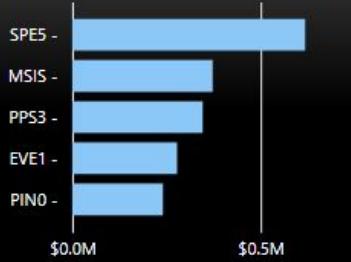
TOP 3 WEB ORDERS



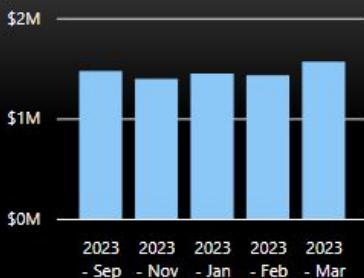
TOP 5 CUSTOMERS



TOP 5 BUYERS



TOP 5 FISCAL YEAR - MONTHS



SURVEYS

41
Made a Purchase 96
Total Responses 42.71%
Purchaser Rate

TOP BOX CSAT %

78.05%
CSAT Top Box (10-9)

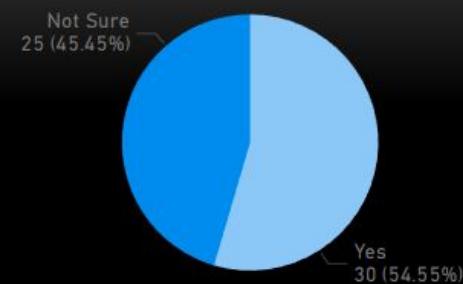
PURCHASER RATE

42.71%
Purchaser Rate
96
Total Responses
41
Made a Purchase
55
Haven't Purchase

TOP 5 BENEFITS OF THE WEBSITE

Benefit	Count
Time Saving	38
Quicker Quoting	33
Item Availability	36
Fast Product Search	31
Easy to Find Products	27
24/7 Online Availability	28
More Resources Online	13
5% Discount Offered	32
Sum of quotinghistoryavailable	19
Sum of orderhistoryavailable	26

NON-WEB PURCHASER; IF THEY PLAN TO PURCHASE IN THE NEAR FUTURE FROM WEB.



[Power BI] Ecommerce dashboard homepage.

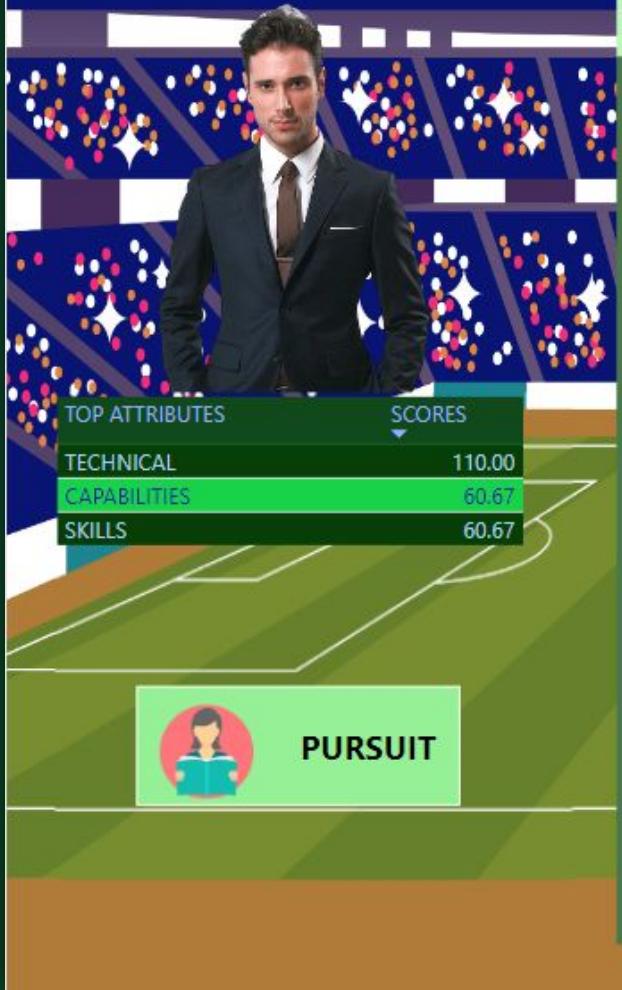
LEMUEL TORREFIEL

★ LVL 3

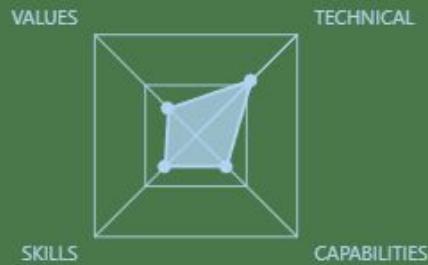
28.44

12.00

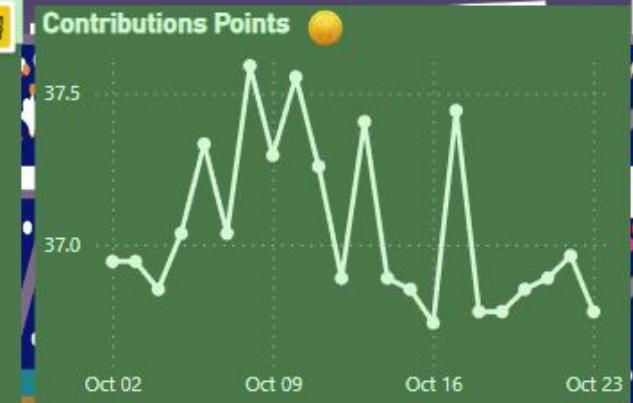
3



SUMMARY



VALUES

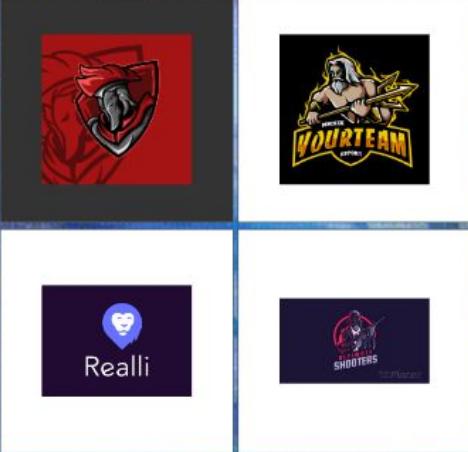


NAME	DURATION
EXCEL MACROS	1.50
EXCEL INTRODUCTION	1.50
EXCEL VBA	3.00
POWER BI - DAX	3.00
POWER BI - POWER QUERY	1.50
POWER BI INTRODUCTION	1.50
DATE	BADGES
Saturday, October 01, 2022	BADGE 1
Sunday, October 02, 2022	BADGE 2
Monday, October 03, 2022	BADGE 3

[Power BI] Game-like interface dashboard for employee growth.

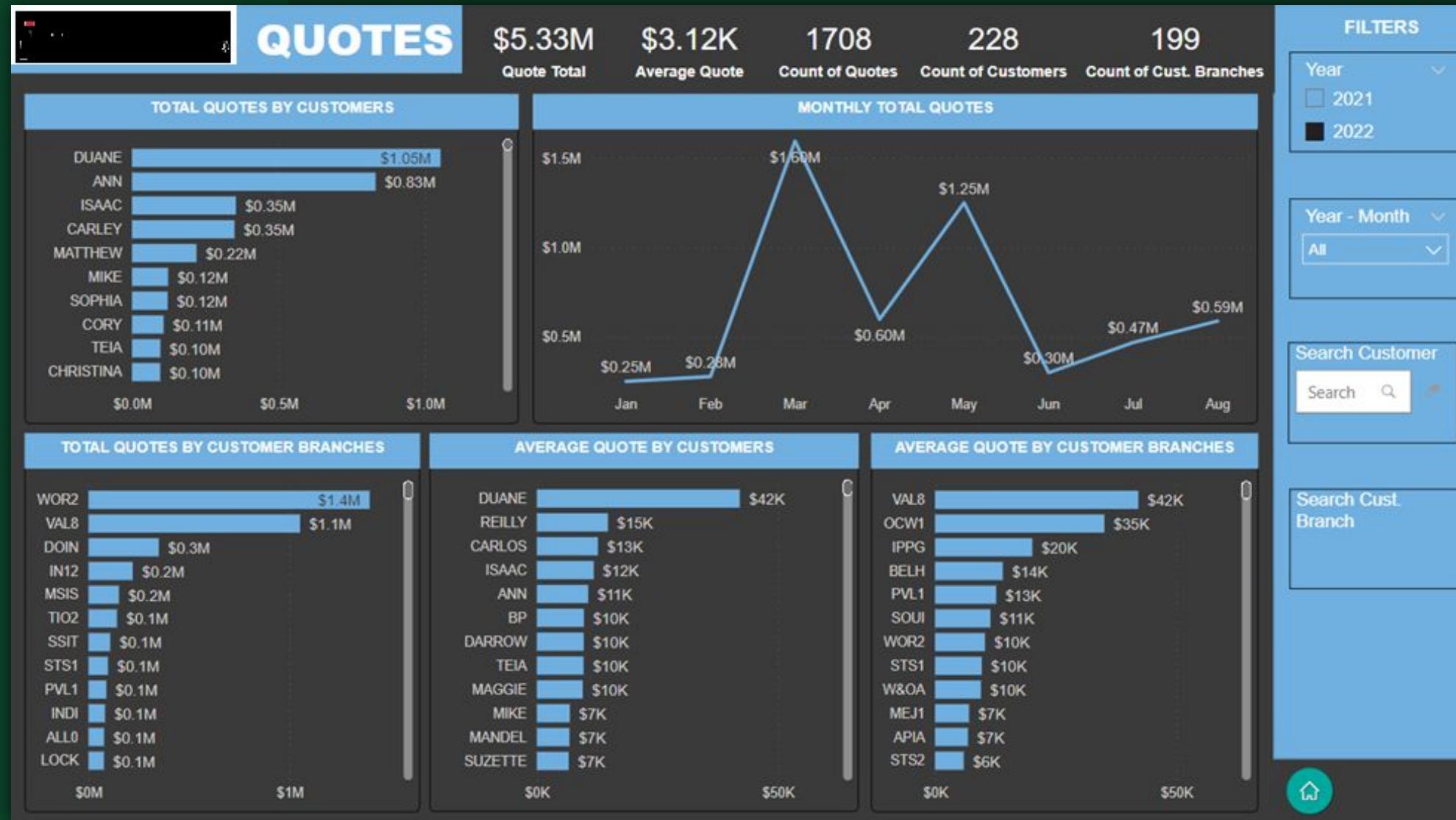
LOGIN

SELECT A TEAM AND IMAGE TO FILTER - ALISSA CLARK



LOGIN

[Power BI] Game-like interface dashboard for employee growth.



[Power BI] Quotes report for Ecommerce client.

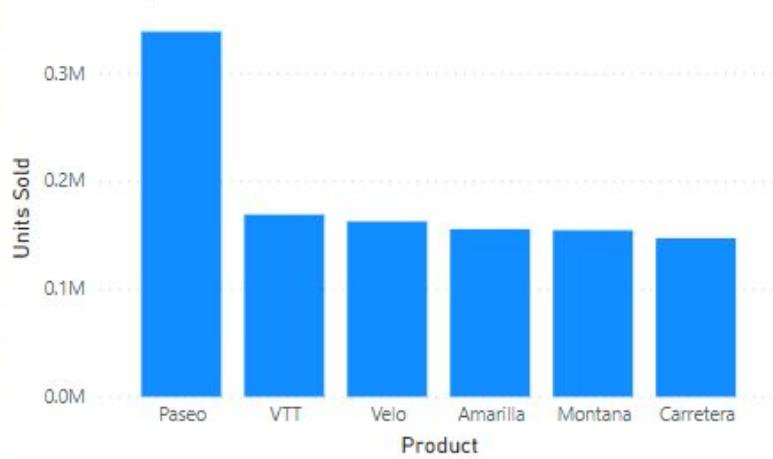
Financial Dashboard



Profit by Country



Units Sold by Product



16.89M

Profit

1.13M

Units Sold

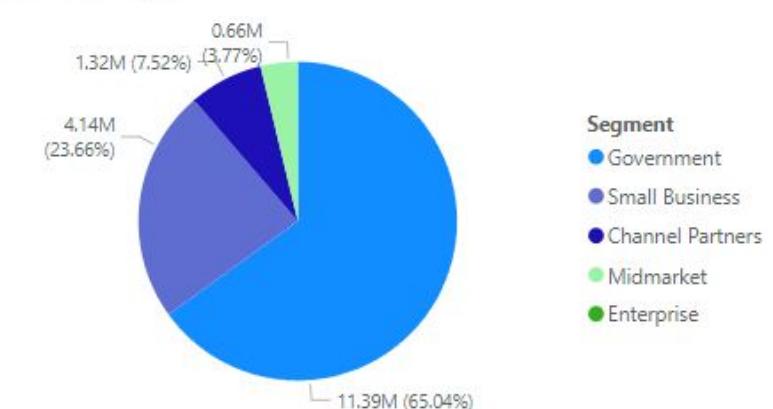
127.93M

Gross Sales

Profit by Date



Profit by Segment



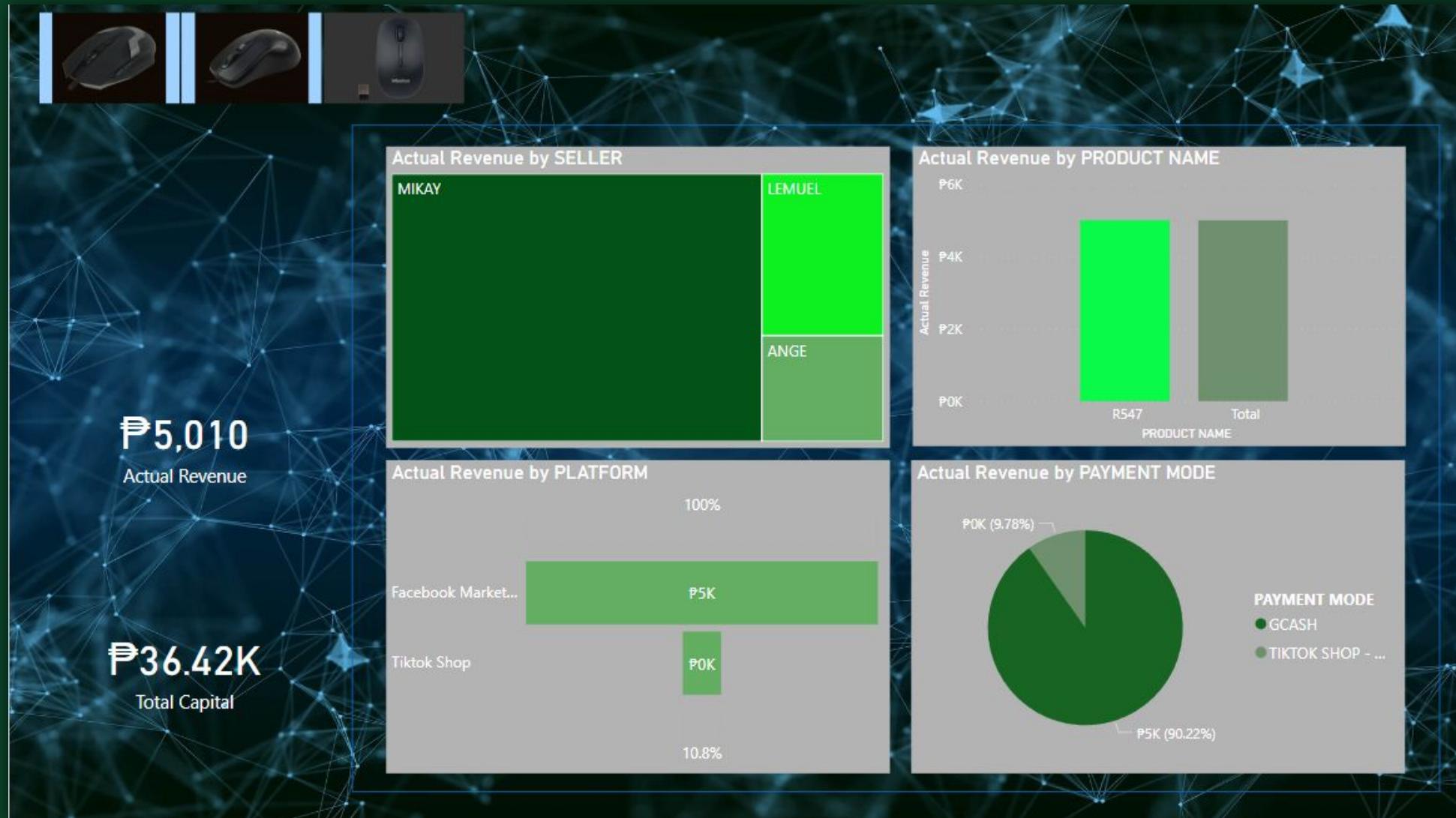
9.21M

Discounts

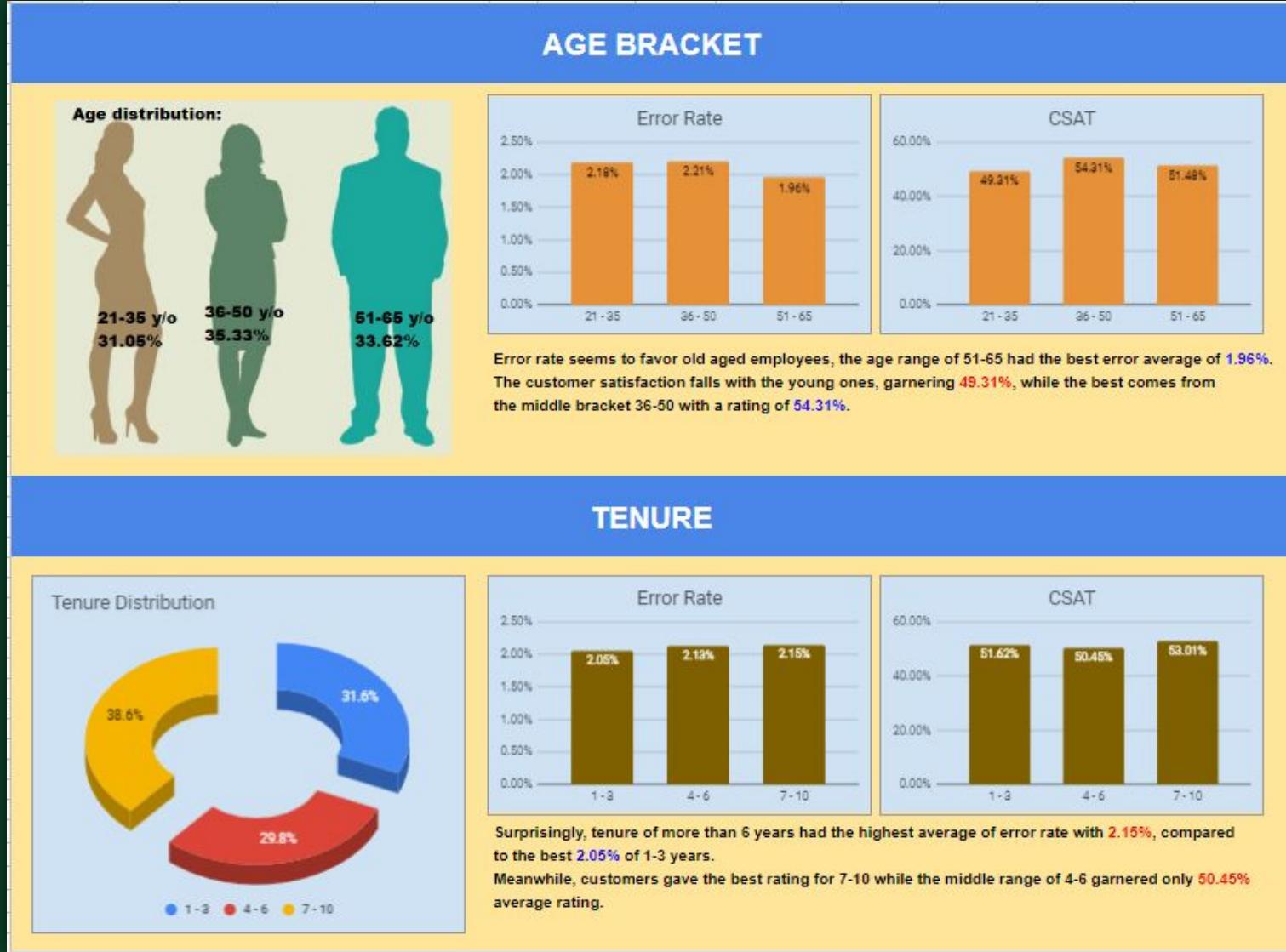
[Power BI] This is a practice dashboard with sample sales data from Microsoft.



[Power BI] This was my former personal business dashboard; the data source is from an Excel file with automated inputs.



[Power BI] This was my former personal business dashboard; the data source is from an Excel file with automated inputs.



[Google Sheets] Infographics with statistical insights.

ERROR RATE AND CSAT DATA VISUALIZATION

DEMOGRAPHICS

3 Team Leads

446,132 Transactions

53 Unique Usernames

21-65 Age Range

1-10 Years Tenure

MONTHLY ERROR RATE TREND



The average error rate trend is slightly consistent at around 1.5% to maximum of 2.69%.

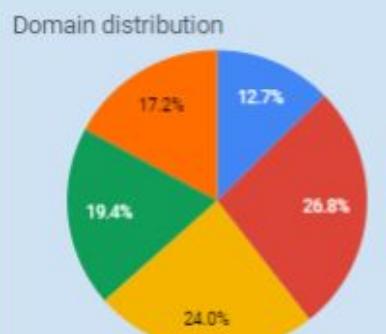
The monthly error rates has good numbers, not exceeding 5% as goal.

The months of September and December have the best averages of error rate with 1.66% and 1.73% respectively.

Meanwhile, the month of October needs attention with the highest error rate of 2.69%.

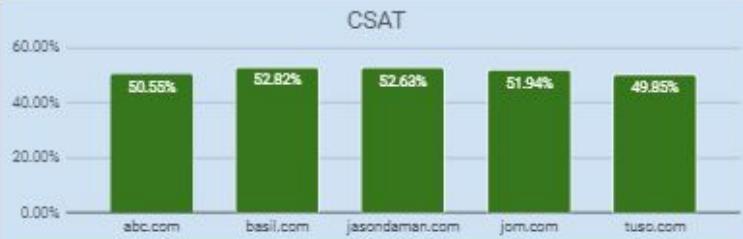
[Google Sheets] Infographics with statistical insights.

DOMAIN



The domain [tuso.com](#) had the best error rate of **1.89%**, but receiving the least customer satisfaction of **49.85%**

Meanwhile, the domain [basil.com](#) had the highest CSAT of **52.82%** together with a manageable error rate of **2.02%**.



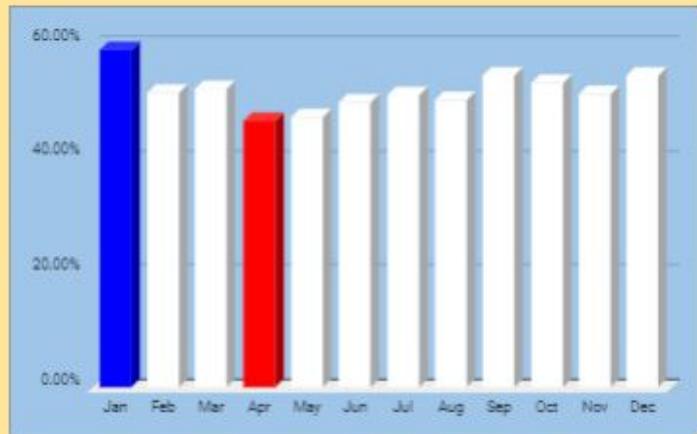
SUMMARY AND SUGGESTIONS

- 1) The average customer satisfaction trend is very low and did not meet the goal of 90%. Focus more on improving the satisfactory rating for the upcoming month of April, May, and June where the 2017 data is below 50%.
- 2) It seemed that customers often dissatisfied when it is summer season. Possible factor is the time being spent when a customer calls. Focus on reducing the time being spent by the customers when having transactions during summer.
- 3) Customers prefer fast transactions, team leader Buckey Barnes' data showed it. He had the best customer rating while having the highest total transactions. Although, he have to give focus on his teams' error rate for improvement.
- 4) Team leaders should focus more on their approach, working with young ones. The young ones have the least satisfaction rating from customers. Team leaders can ask the adult ones on how they manage to give customers a good service.
- 5) The domain [tuso.com](#) can be approached by the team leaders, to improve their user interface, timeline and etc. [tuso.com](#) falls below 50% with satisfactory rating from customers, which impacts the employees' experience during a transaction.
- 6) There are 19 users that have below 50% average CSAT. Team leaders may approach their respective people for improvement strategies.

[Google Sheets] Infographics with statistical insights.

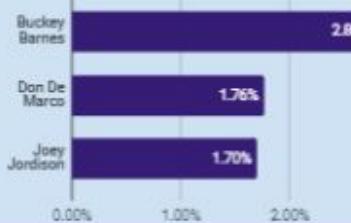
MONTHLY CUSTOMER SATISFACTION TREND

The month of January had the best average of **59.11%**, while the month of April had the worst average of **46.58%**.

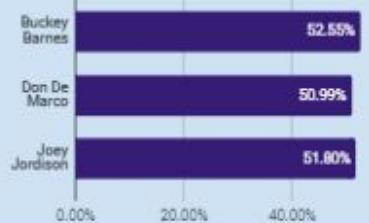


TEAM LEADERS

Average Error Rate



Average CSAT



Sum of Total Transactions



Eventhough having the highest average of **2.80%** error rate, team leader Buckey Barnes managed to gain the highest average customer satisfaction rating of **52.55%** while doing the highest total of transactions of **165,039 (37.0%)**.

[Google Sheets] Infographics with statistical insights.

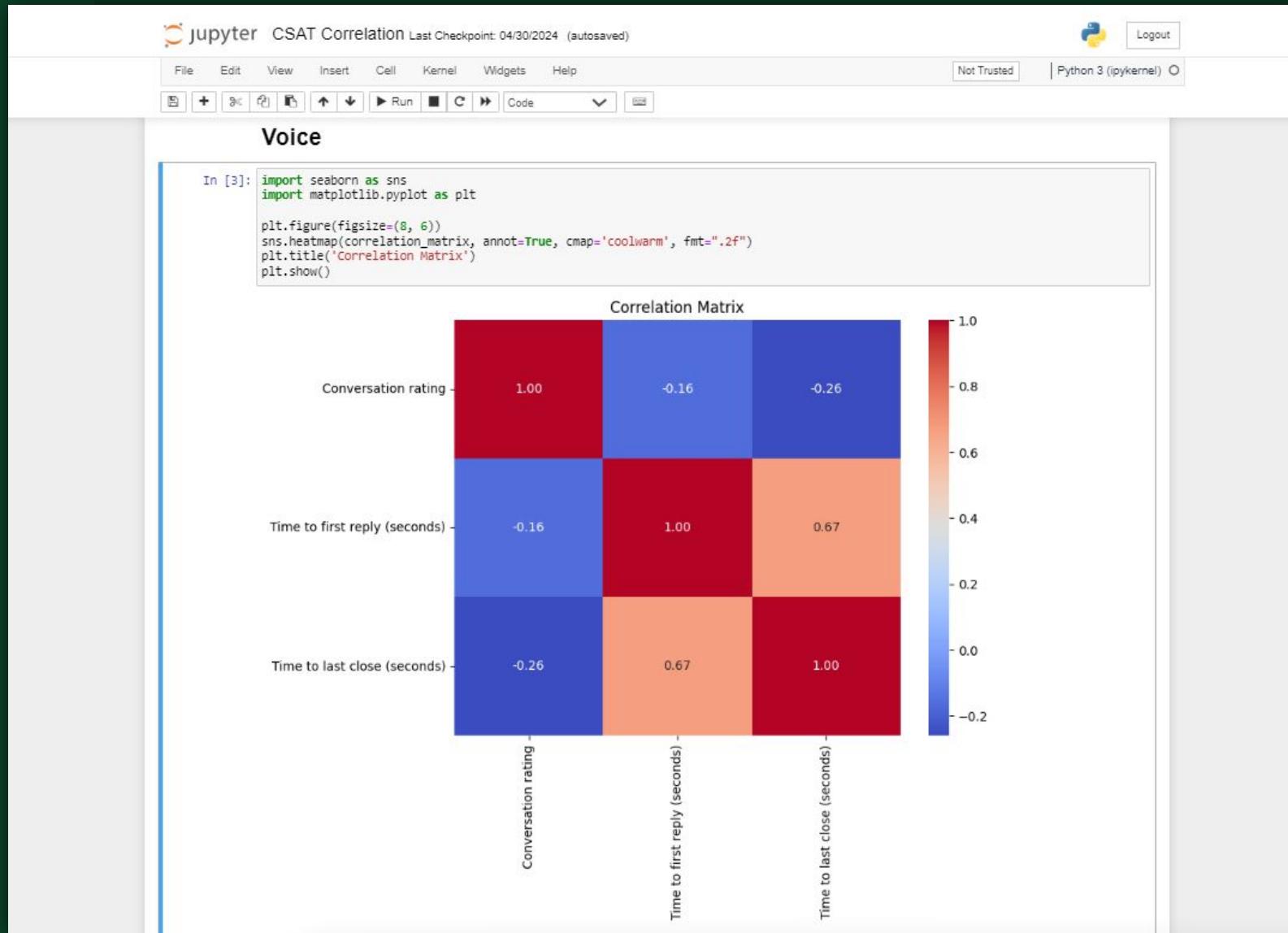
Advanced Scripting and Programming for Data Analytics (Python, R, SQL, Access, Excel)



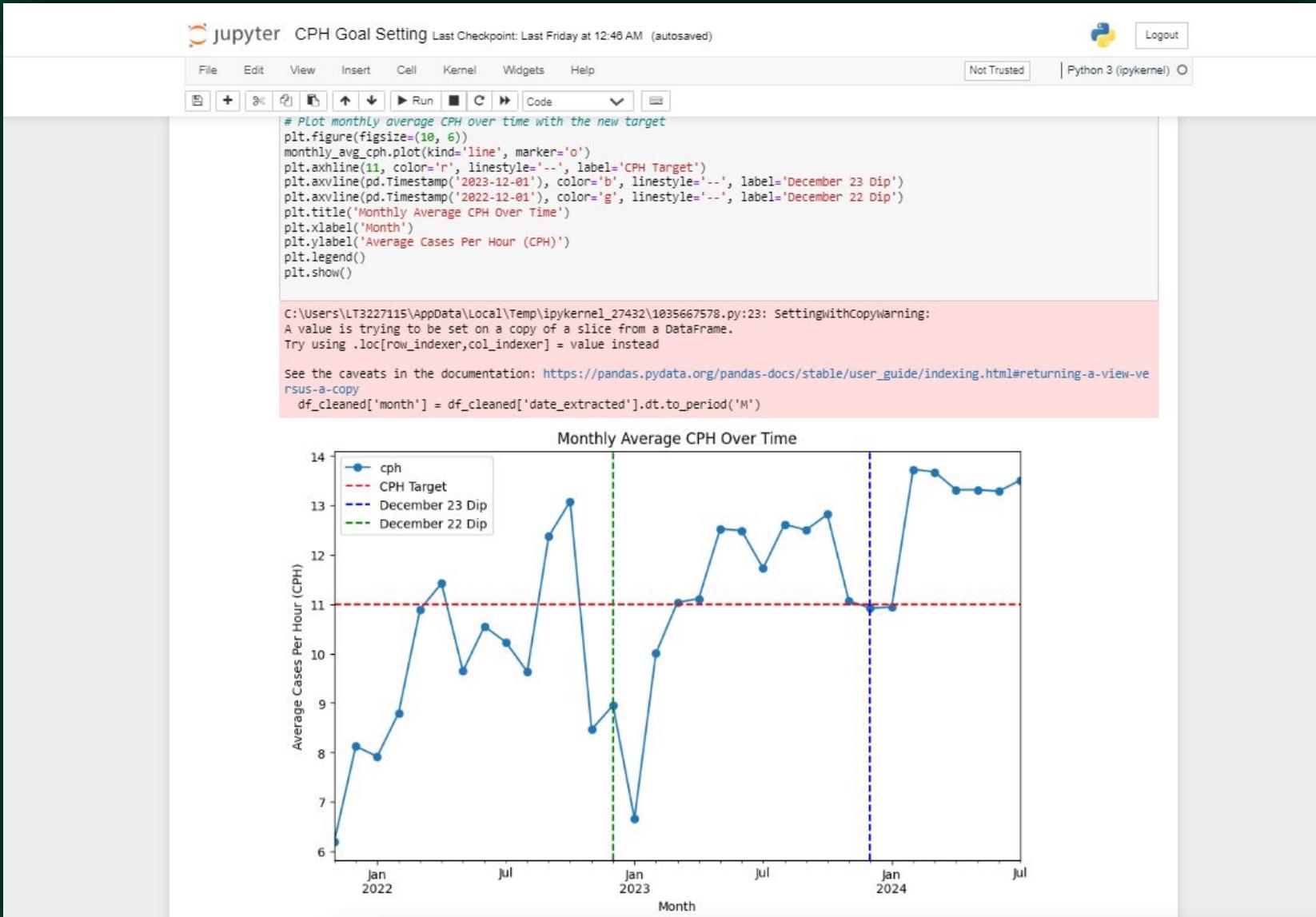


```
File Edit Selection View Go Run Terminal Help
ETLs daily_etlmy 05_leads_to_gsheet.py 04_people_to_gsheet.py test.txt
EXPLORER
ETLs .github\workflows daily_etlmy followupboss > 04_people_to_gsheet.py ...
14 api_key = os.getenv("FOLLOWUPBOSS_APIKEY")
15 X_System_Key = os.getenv("FOLLOWUPBOSS_XSYSTEMKEY")
16 X_System = os.getenv("FOLLOWUPBOSS_XSYSTEM")
17 mongopass = os.getenv("MONGODB_PASSWORD")
18 gsheetId = os.getenv("GSHEET_ID")
19
20 working_directory = os.getcwd()
21 # r'c:\\Users\\ENDUSER\\OneDrive\\FOR CHRISTINA\\Python\\ETLs\\followupboss\\logs.txt'
22 logfile = os.path.join(working_directory, "followupboss", "logs.txt")
23
24 hoover_tz = pytz.timezone('America/Chicago')
25
26 current_time_initial = dt.now(hoover_tz)
27 current_time_ph_initial = dt.now()
28
29 with open(logfile, 'a') as file:
30     file.write(f'\nPeople Extract Start time in USA: {current_time_initial}')
31     file.write(f'\nPeople Extract Start time in PH: {current_time_ph_initial}\n')
32
33 # Encode API key in Base64
34 encoded_api_key = base64.b64encode(api_key.encode('utf-8')).decode('utf-8')
35
36 # API endpoint and query parameters
37 url = "https://api.followupboss.com/v1/people?sort=lastActivity&limit=100&offset=0&includeTrash=false&includeUnclaimed=false"
38
39 r = requests.get(url, headers={'accept': 'application/json',
40                                 'Authorization': "Basic " + encoded_api_key,
41                                 'X-System-Key': X_System_Key,
42                                 'X-System': X_System
43                               })
44
45 data = r.json()
46 total = data["metadata"]["total"]
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS AZURE
powershell + ... ^ x
(venv) PS C:\Users\ENDUSER\OneDrive\FOR CHRISTINA\Python\ETLs> git push origin main
git: 'credential-manager-core' is not a git command. See 'git --help'.
Enumerating objects: 12, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 16 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 932 bytes | 932.00 KiB/s, done.
Total 6 (delta 4), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (4/4), completed with 3 local objects.
To https://github.com/lalala0095/christina
 44b2e4a..55c7cbb main -> main
(venv) PS C:\Users\ENDUSER\OneDrive\FOR CHRISTINA\Python\ETLs>
* History restored!
Ln 8, Col 32 (6 selected) Spaces: 4 UTF-8 CRLF {} Python 3.12.2 (venv: venv) ⚡ Go Live
```

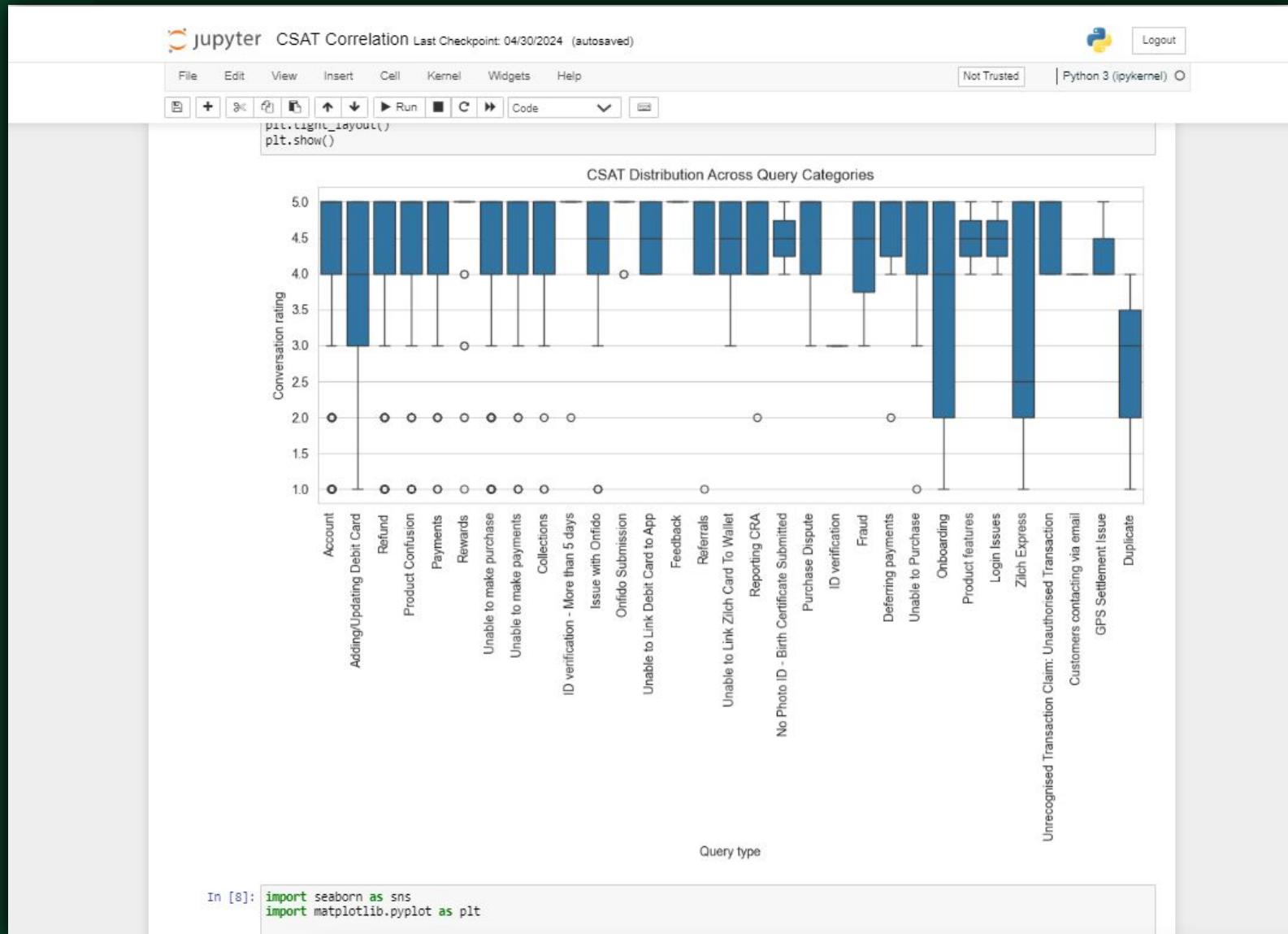
[Python] ETL workflow for daily scheduled refresh of data extraction from CRM to Google Sheets as database using Github actions as cloud server.



[Python] Heat map visualization for Customer Satisfaction to other variables correlation analysis.



[Python] Trend visualization for Goal setting of Cases Per Hour KPI.



[Python] Customer Satisfaction distribution visualization per Query type.

The screenshot shows a Jupyter Notebook interface running on localhost:8888. The notebook title is "Followup boss API - Jupyter Notebooks". The code in the cell (In [29]) is written in Python and performs the following tasks:

- Imports pandas and json.
- Loads JSON data from a file named "followupboss_people_data.json".
- Initializes lists to hold email values and types, including columns for Client Email, Type, Phone, and Address details (Street, City, State, Zip, Country, Type).
- Extracts emails and types for each person in the data.
- Creates columns for Client Email, Type, Phone, and Address details (Street, City, State, Zip, Country, Type) by appending values or None based on the length of the extracted list.
- Creates a DataFrame from the JSON data.

```
import pandas as pd
import json

# Load JSON data from file
with open('followupboss_people_data.json', 'r') as f:
    data = json.load(f)

# Initialize Lists to hold email values and types
email_columns = {f'Client Email {i+1}': [] for i in range(6)}
email_type_columns = {f'Client Email {i+1} - Type': [] for i in range(6)}
phone_columns = {f'Client Phone {i+1}': [] for i in range(6)}
phone_type_columns = {f'Client Phone {i+1} - Type': [] for i in range(6)}
address_street_columns = {f'Client Address {i+1} - Street': [] for i in range(6)}
address_city_columns = {f'Client Address {i+1} - City': [] for i in range(6)}
address_state_columns = {f'Client Address {i+1} - State': [] for i in range(6)}
address_zip_columns = {f'Client Address {i+1} - Zip': [] for i in range(6)}
address_country_columns = {f'Client Address {i+1} - Country': [] for i in range(6)}
address_type_columns = {f'Client Address {i+1} - Type': [] for i in range(6)}

# Extract emails and types
for person in data:
    emails = person.get('emails', [])
    for i in range(6):
        if i < len(emails):
            email_columns[f'Client Email {i+1}'].append(emails[i]['value'])
            email_type_columns[f'Client Email {i+1} - Type'].append(emails[i]['type'])
        else:
            email_columns[f'Client Email {i+1}'].append(None)
            email_type_columns[f'Client Email {i+1} - Type'].append(None)
    phones = person.get('phones', [])
    for i in range(6):
        if i < len(phones):
            phone_columns[f'Client Phone {i+1}'].append(phones[i]['value'])
            phone_type_columns[f'Client Phone {i+1} - Type'].append(phones[i]['type'])
        else:
            phone_columns[f'Client Phone {i+1}'].append(None)
            phone_type_columns[f'Client Phone {i+1} - Type'].append(None)
    addresses = person.get('addresses', [])
    for i in range(6):
        if i < len(addresses):
            address_street_columns[f'Client Address {i+1} - Street'].append(addresses[i]['street'])
            address_city_columns[f'Client Address {i+1} - City'].append(addresses[i]['city'])
            address_state_columns[f'Client Address {i+1} - State'].append(addresses[i]['state'])
            address_zip_columns[f'Client Address {i+1} - Zip'].append(addresses[i]['zip'])
            address_country_columns[f'Client Address {i+1} - Country'].append(addresses[i]['country'])
            address_type_columns[f'Client Address {i+1} - Type'].append(addresses[i]['type'])
        else:
            address_street_columns[f'Client Address {i+1} - Street'].append(None)
            address_city_columns[f'Client Address {i+1} - City'].append(None)
            address_state_columns[f'Client Address {i+1} - State'].append(None)
            address_zip_columns[f'Client Address {i+1} - Zip'].append(None)
            address_country_columns[f'Client Address {i+1} - Country'].append(None)
            address_type_columns[f'Client Address {i+1} - Type'].append(None)

# Create a DataFrame from the JSON data
df_people = pd.DataFrame(data)
```

[R] Develop R and SQL scripts to automate ETL jobs before jumping into dashboard development.

DBeaver 23.2.4 - <dev> Matching KPIe and MV.sql

File Edit Navigate Search SQL Editor Database Window Help

SQL Commit Rollback N/A dev < N/A >

<dev> Script-2 <dev> Check Col or Table names.sql <dev> Script-7 <dev> Tls not having business metrics scores in IPS.sql <dev> ips <dev> Matching KPIe and MV.sql

```
SELECT
    a.kpi_name AS kpi_name_kpie,
    a.hierarchy_code,
    a.line_of_business,
    a.sub_line_of_business,
    a.process,
    a.nom AS nom_kpie,
    a.denom AS denom_kpie,
    b.nom AS nom_gpr,
    b.denom AS denom_gpr,
    a.kpi_value AS kpi_value_kpie,
    b.kpi_value AS kpi_value_gpr,
CASE
    WHEN round(a.kpi_value, 2) = round(b.kpi_value, 2) THEN 'Yes' ELSE 'No'
END AS matching,
a.kpi_value - b.kpi_value AS difference
FROM
(
    SELECT
        'kpie view' AS source,
        ds.kpi_name AS kpi_name,
        er.hierarchy_code,
        er.line_of_business,
        er.sub_line_of_business,
        er.process,
        SUM(ds.measure_nom) AS nom,
        SUM(ds.measure_denom) AS denom,
        CASE
            WHEN SUM(ds.measure_denom) = 0 THEN 0
            ELSE (SUM(ds.measure_nom) / SUM(ds.measure_denom)) * 100
        END AS kpi_value
    FROM
        kpi_engine.v_utl_mindbody_quality ds /* replace this with your KPIe view/KPIe master table */
    JOIN
        public.epms_roster_v2 er
)
```

[SQL] Realtime based SQL querying for up to date data checking, data validation and data management.

The screenshot shows the RStudio interface with an R script open in the main editor window. The script contains code for data manipulation, including merging datasets and connecting to a Microsoft SQL Server database. The R console window shows the standard R startup message and workspace loading information. The top navigation bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, and Help. The bottom taskbar shows various application icons.

```
85 buyers = dbGetQuery(con, "SELECT * FROM AR_CustomerContact")
86 branches = dbGetQuery(con, "SELECT * FROM ivalco.WarehouseInfo")
87 branches$WarehouseName = paste0(branches$WarehouseId, " - ", branches$WarehouseName)
88 uniqueBuyersPerCompany = count(buyers, CustomerNo)
89 customerNoBranches = dbGetQuery(con, "select CustomerNo, UDF_0504620_BRANCH from dbo.FROMMAS_AR_Customer")
90
91 salesHistHeader = filter(salesHistHeaderOrig, !is.na(salesHistHeaderOrig$SalesOrderNo))
92
93 salesHistHeader = merge(salesHistHeader, customerNoBranches, by = "CustomerNo", all.x = T)
94 salesHistHeader = merge(salesHistHeader, branches, by.x = "UDF_0504620_BRANCH", by.y = "WarehouseId", all.x = T)
95 salesHistHeader = merge(salesHistHeader, uniqueBuyersPerCompany, by = "CustomerNo", all.x = T)
96 salesHistHeader$CustomerBranch = salesHistHeader$WarehouseName
97 salesHistHeader$PlacedBy = salesHistHeader$UDF_0504106_PLACEDBY
98 salesHistHeader = salesHistHeader[,c("SalesOrderNo", "CustomerNo", "OrderDate", "SalesPersonNo",
99 "NonTaxableAmt", "DiscountAmt", "OrderStatus",
100 "PlacedBy", "WarehouseCode", "EmailAddress",
101 "CustomerBranch", "n")]
102 )
103
```

R 4.2.1 (2022-06-23 ucrt) -- "Funny-Looking Kid"
Copyright (C) 2022 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[Workspace loaded from ~/RData]

[R] R scripts for data manipulation and integration to Power BI and other reporting tools.

Database1 : Database- D:\OneDrive\FOR CHRISTINA\Database1.accdb (Access 2007 - 2016 file format) - Access

Lemuel Torrefiel IT

File Home Create External Data Database Tools Help Tell me what you want to do

All Access ... Search... Tables Queries Forms

Views Clipboard Sort

CLJ Real Estate System - Home

Homepage

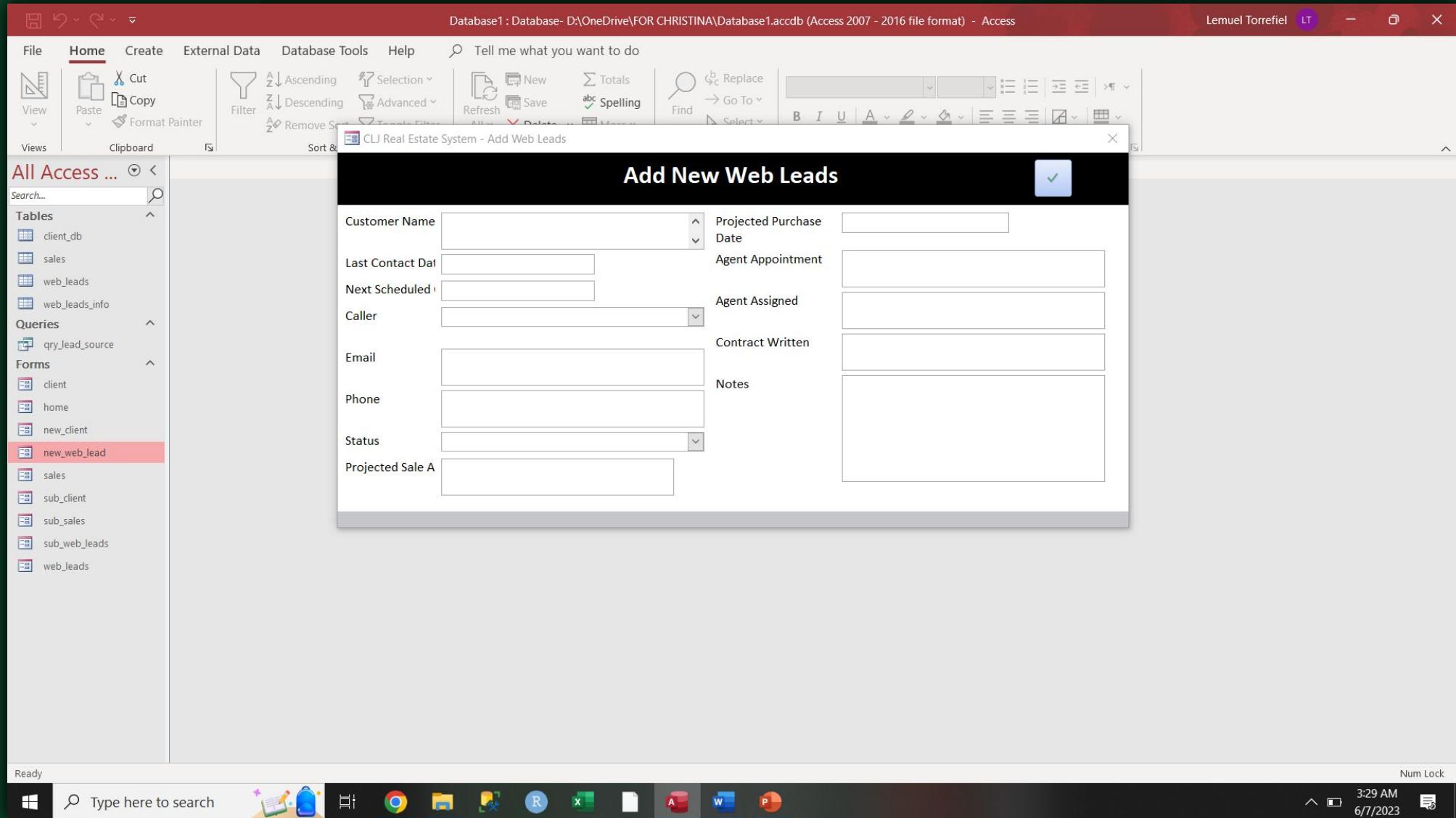
CLIENT LEAD SALES

Form View Num Lock

Type here to search

3:27 AM 6/7/2023

[MS Access] Real Estate Team CRM.



[MS Access] Real Estate Team CRM.

LANDM EXCELxism - Excel

File Home Insert Page Layout Formulas Data Review View Developer Help

Record Macro Use Relative References Add-ins Insert Design View Code Properties Import Expansion Packs Export Refresh Data Source COM Add-ins Add-ins Run Dialog XML

Visual Basic Macros Macro Security

Code Add-ins Controls

Products - Show all products panel

Search Product Name Here:

PRODUCT ID	SUPPLIER	PRODUCT TYPE	PRODUCT NAME	SKU PREFIX	PRICE IN USD	PRICE IN PHP	SELLING PRICE
PR001	MEITION	MOUSE	M940	GWRNRMO-5	2.98	160.62	350
PR002	MEITION	MOUSE	R547	OWLNRMO-3	1.89	101.87	250
PR003	MEITION	MOUSE	M361	OWRNRMO-4	0.98	52.82	100
PR004	MEITION	KEYBOARD	MK005	GWRNRKB-3	13.91	749.75	1200
PR005	MEITION	KEYBOARD	MK600RD	GWRNRKB-3	14.98	807.42	1300
PR006	MEITION	HEADSET	HP021	GWRNRHS-3	6.25	336.88	550
PR007	MEITION	MOUSE	M380	OWRNRMO-4	5.2	280.28	500
PR008	HENGTONGDA	MOUSE	D5200	OWRNRCO-1	2.33	125.59	250
PR009	HENGTONGDA	KM COMBO	G21B	GWRNRCO-2	3.33	179.49	350
PR010	HENGTONGDA	KM COMBO	9122	GWRNRCO-2	3.58	192.96	380
PR011	HENGTONGDA	HEADSET	A65 WITH LED	GWRNRHS-2	2.92	157.39	320
PR012	HENGTONGDA	HEADSET	809	OWRNRHS-2	1.33	71.69	150
PR013	HENGTONGDA	HEADSET	V58	OWRNRHS-2	2.17	116.96	220
PR014	HENGTONGDA	HEADSET	A3	GWRNRHS-3	2.34	126.13	260
PR015	HENGTONGDA	MOUSE	Q13	GWLRCMO-0	3.5	188.65	370
PR016	HENGTONGDA	KEYBOARD	K830	OWRNRKB-1	1.75	94.33	200
PR017	HENGTONGDA	SPEAKER	JM018	OWRNRSP-2	0.63	33.96	100

Add Product Delete Product Edit Product Full Specs Sheet Specs Sheet for Buyers Exit

DASHBOARD

Type here to search

12:34 PM 10/2/2022

[Excel] Inventory software in an Excel file using VBA and Macros.

LANDM EXCELxism - Excel

File Home Insert Page Layout Formulas Data Review View Developer Help

Record Macro
Visual Macros Basic
Add-ins Excel COM Add-ins Add-ins Basic
Properties Import
View Code Refresh Data
Source Export
Expansion Packs Run Dialog

Products - Show all products panel.

Purchase Section

Search

PURCHASE_ID	SUPPLIER	DATE	USD_FACTOR	PHP_FACTOR
PU001	MEITION	2/11/2022	1	53.32
PU002	HENGTONGDA	3/22/2022	1	53.95
PU003	MEITION	5/10/2022	1	53.9

PURCHASE_ID	PRODUCT_ID	PRODUCT_NAME	COST IN USD
PU001	PR001	M940	2.98
PU001	PR025	M371	1.6
PU001	PR026	M915	2.73
PU001	PR027	K9300	6.3
PU001	PR028	R560	2.1
PU001	PR029	R545	1.83
PU001	PR030	M360	0.77

Add Purchase
Edit Purchase
Delete Purchase
Exit to Dashboard

L & M'
PRODUCTS
PURCHASE
SHOW ALL

DASHBOARD

Type here to search

12:35 PM 10/2/2022

[Excel] Inventory software in an Excel file using VBA and Macros.

**My skills
doesn't end
here, as I
continue to
grow in the
field of AI
Integrations.**



- AI Model Training (Beginner)
- Machine Learning Concepts (Beginner)
- App Sheets (Intermediate)
- Azure Document Intelligence (Beginner)

Future initiatives

1. Actively learning other principles of AI in Python such as Machine Learning, to be able to train locally available AI models and fine tune them based on clients' specifications.
2. Exploring other emerging technologies such as Model Context Protocol and Google Agent Development Kit which I see as an emerging tech stacks in AI.
3. Collaborative partnerships. Foster collaborations with tech innovators and industry leaders to drive innovation. Observe and learn from experienced AI Experts, influencers and content creators.
4. Practice more by doing projects, which is my natural way of learning tech stacks.

This is the end of my presentation.

Thank you very
much for your
time!

[My Portfolio Website Link](#)

