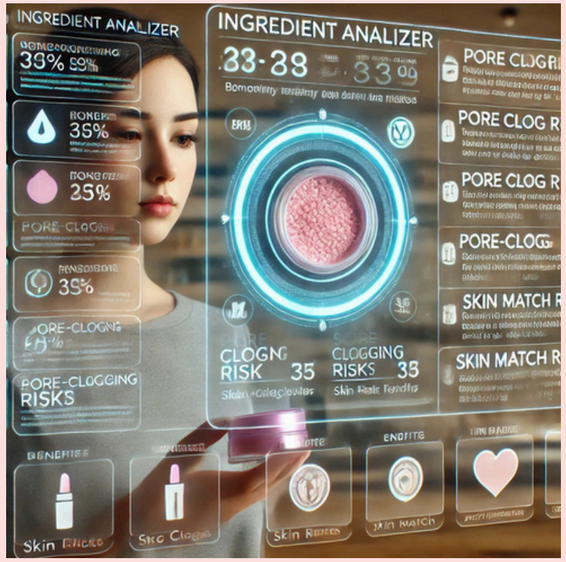
Makeup Ingredient Analyzer

By: Alondra Sanchez & Esmeralda Melendez



https://github.com/ksu-is/makeupanalyzer

INTRODUCTION

The Makeup Ingredient Analyzer is a tool-like bot designed to help users understand the composition of their cosmetic products and pore-clogging ingredients. Analyzing the ingredients listed on makeup labels provides detailed information about each ingredient, including its purpose (e.g., moisturizer, tint), potential benefits, possible risks (e.g., pore cloggers or irritants), and product-to-product & product-to-skin match (e.g., silicone vs. water-based and dry vs. oily skin type, respectively). The tool aims to promote safer and more informed cosmetic choices by educating users about what they're applying to their skin.

Packages Used

Telegram, Google Cloud Vision, Pandas, OS

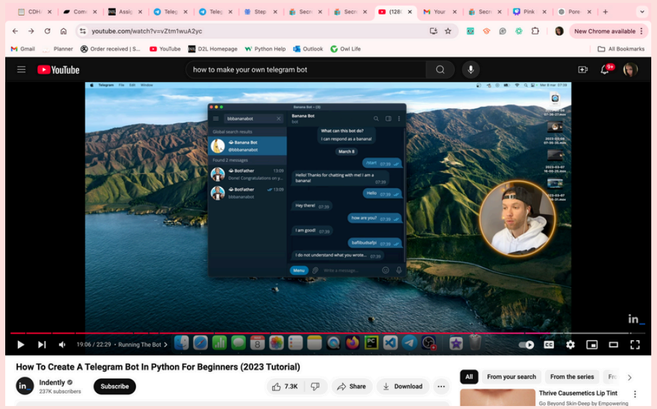
BACKGROUND

What was the idea?

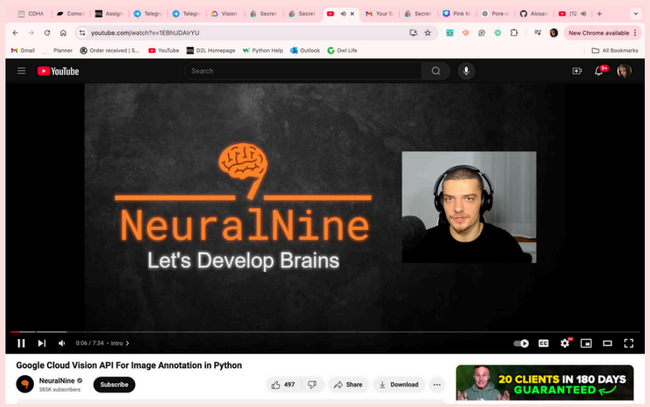
The idea stemmed from the need to decrease the amount of time, money and effort to compare a list of ingredients. We were inspired by immense list of cosmetic variations between brands and the struggles with acne limiting makeup choices. Not knowing what or how the products are causing acne is the basis for the program. Use this Bot not only for makeup products but skincare products and hair products as well. Have fun and may your best makeup routine shine!

REFERENCES AND SOURCE CODE

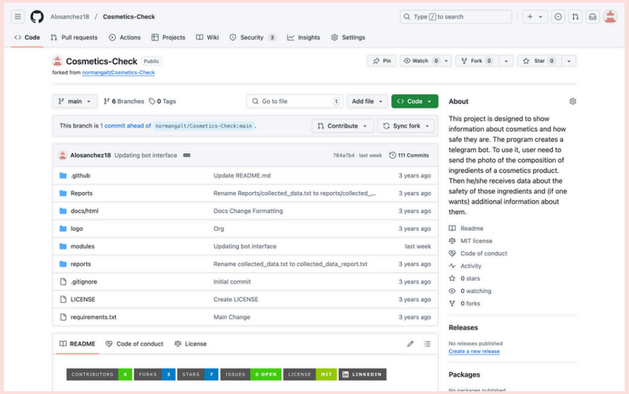
What source code & tutorials did we find?



Telegram Tutorial



Google Cloud Vision Tutorial



Cosmetics Checker

DEVELOPMENT

What surprised you? How did the scope change? Why?

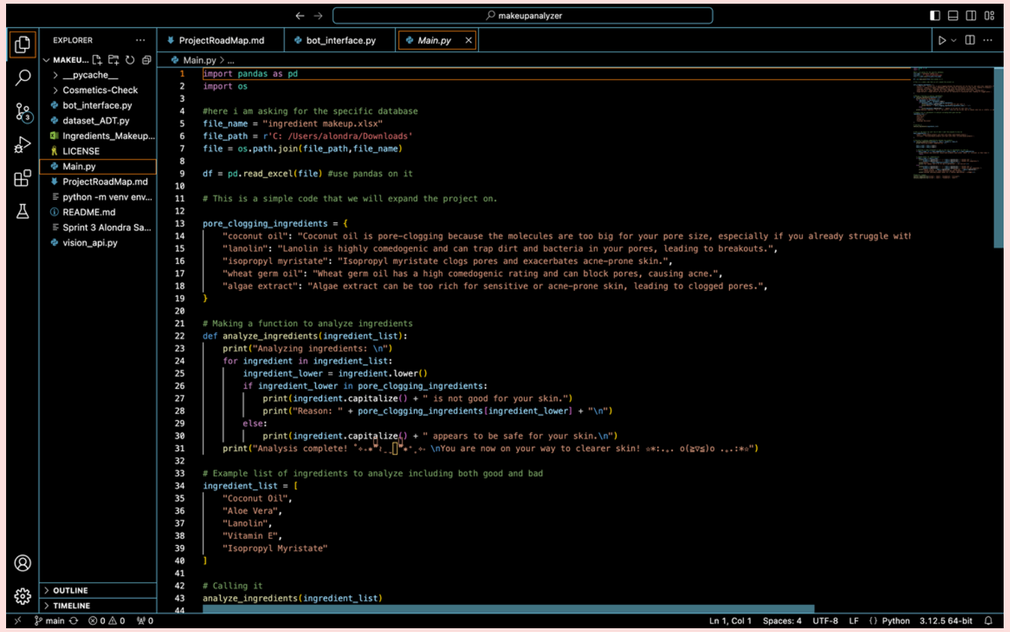
- Number of packages to use.
- How Telegram allows you to easily create a bot
- Even though it was easy to invision how it COULD come together, we had to replan to change the scope of the project
  - Instead of the bot being able to do two different tasks, our focus became one.

What roadblocks did you face? How did you overcome them?

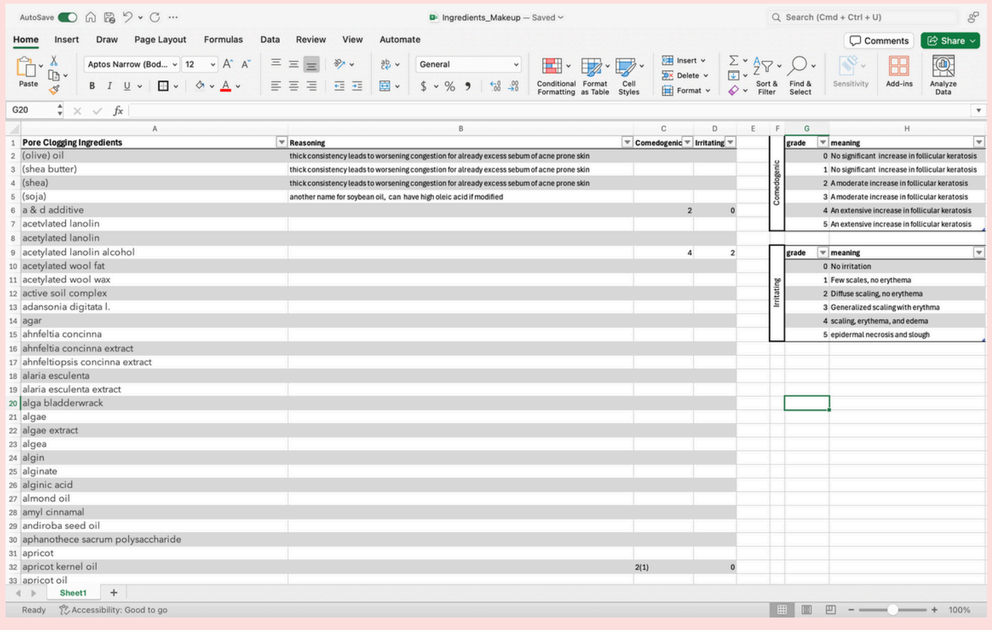
- We began with a small list to get the interface started as turned in on Sprint two but how we soon realized having a spreadsheet would have been better
- We also found lists of acne-aggrivating ingredients but not a correspinding list as to why
- We had to compile descriptions as to why to overcome the issue.
- Because the list of ingredients is almost 500, we only got to about 50% before moving on to another step.
- Not understanding quite what each point in the reference code meant. Using CHAT to interpret to better gage where the changes were going to be.

DEVELOPMENT

This is was our main code before before.

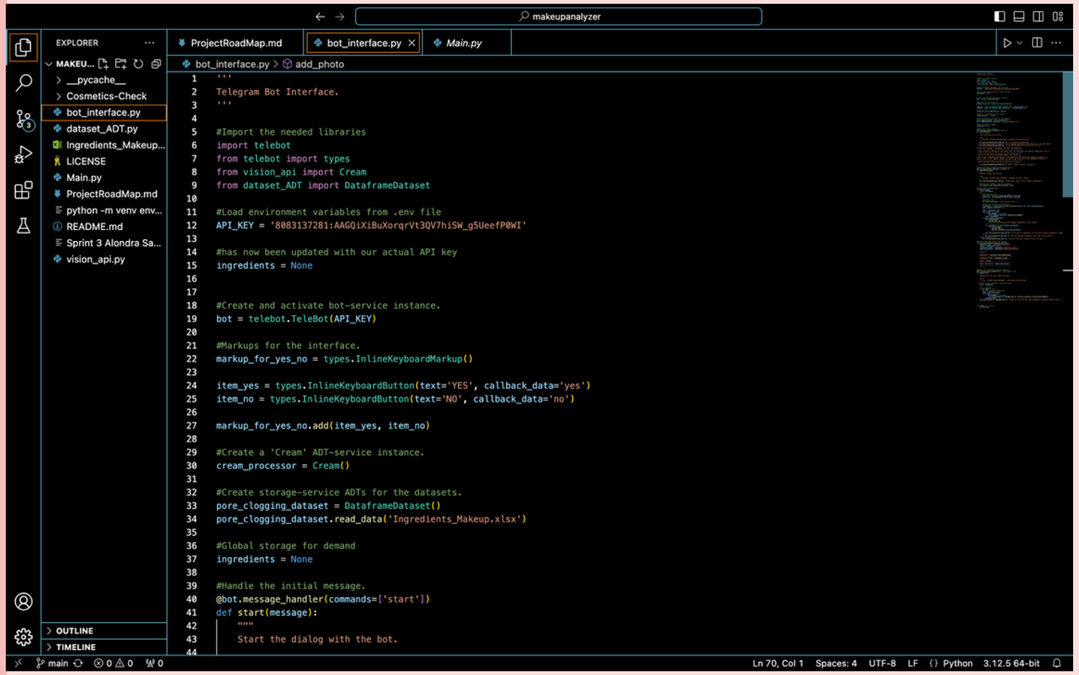


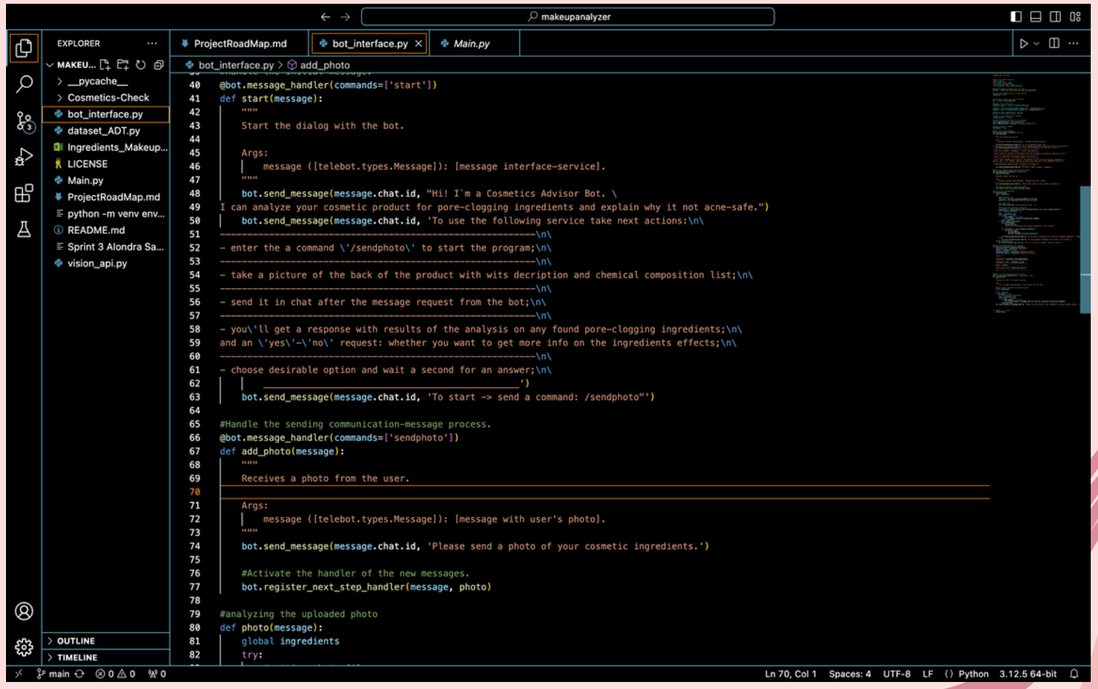
This is how our list looks like now.



BOT_INTERFACE.PY

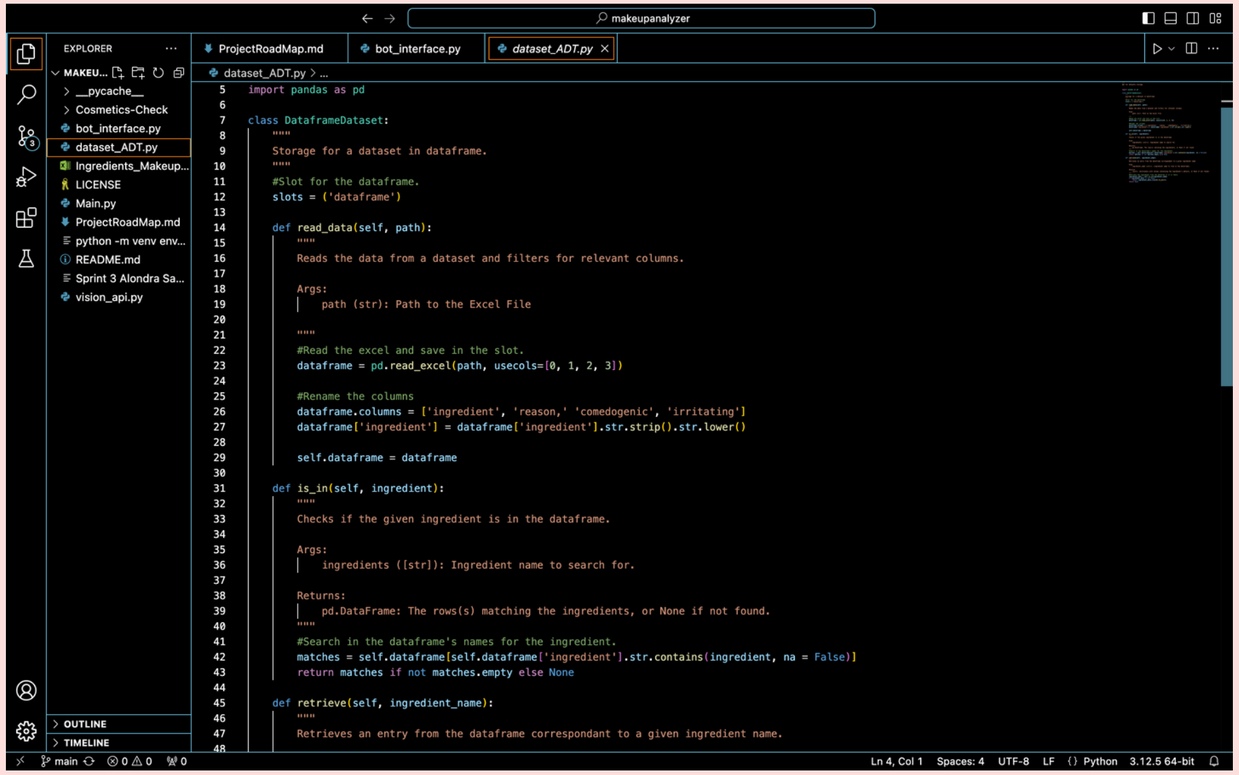
All the packages are gathered in the interface. It is the “main” code where it all happens. This handles the behavior of the bot.



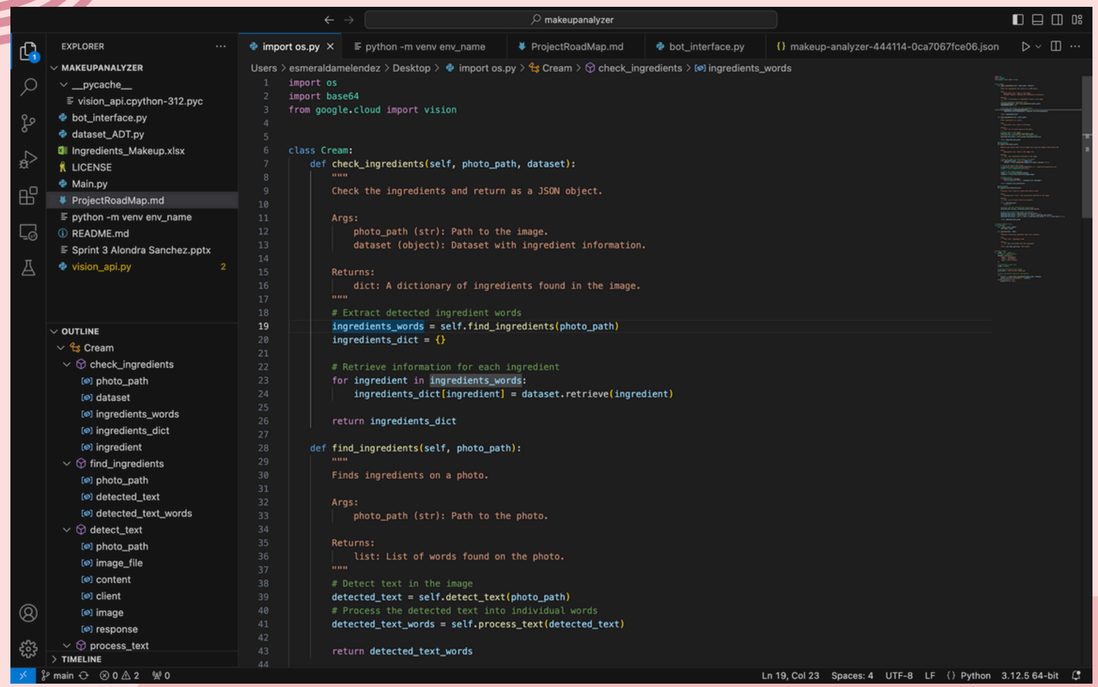


DATASET_ADT.PY

The reason for setting it up this way without the direct path here is that the code can be more versatile for future reference if we need to update the list.

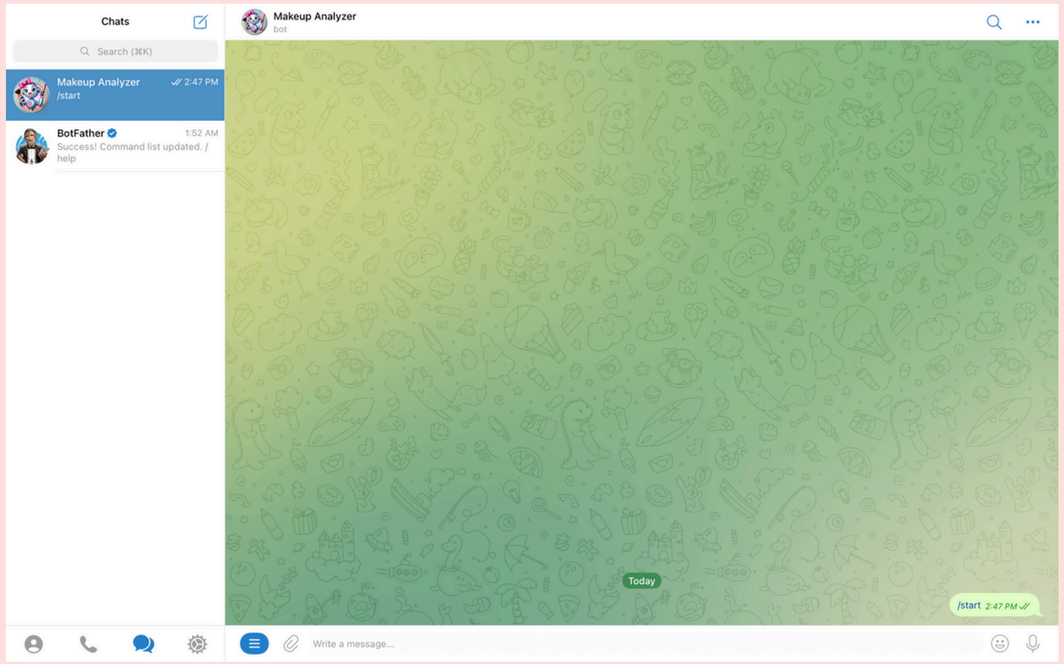


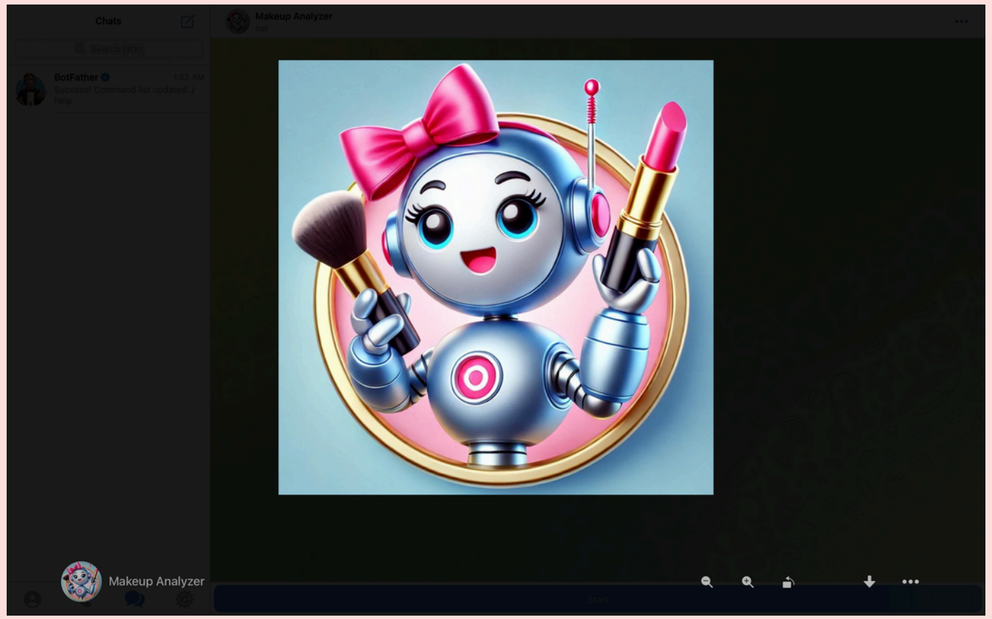
VISION_API.PY



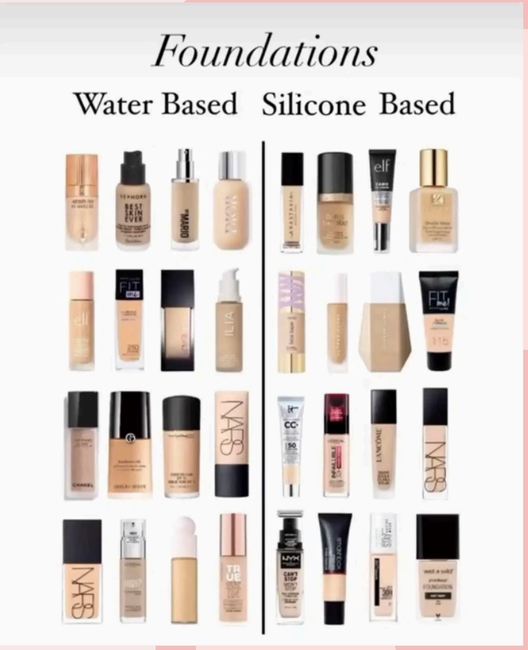
The Google Cloud Vision API simplifies image analysis by providing powerful pre-trained machine learning models. It enables developers to extract meaningful information from images, solving problems across a variety of industries efficiently. By providing a code the machine will generate an image analysis of what is being portrayed in the photo.

NOW LETS RUN IT....





@makeup_analyzer_bot chatbox



CONCLUSION

It did not work....

Before we would be able to make anything happen, we would have to go in and make improvements to the development, the data set, and the various different interfaces we are trying to run. The next steps would be to get the product to product compatibility code implemented and working. We also would want the Bot to recommend products for both types of analyzes.

THANK YOU