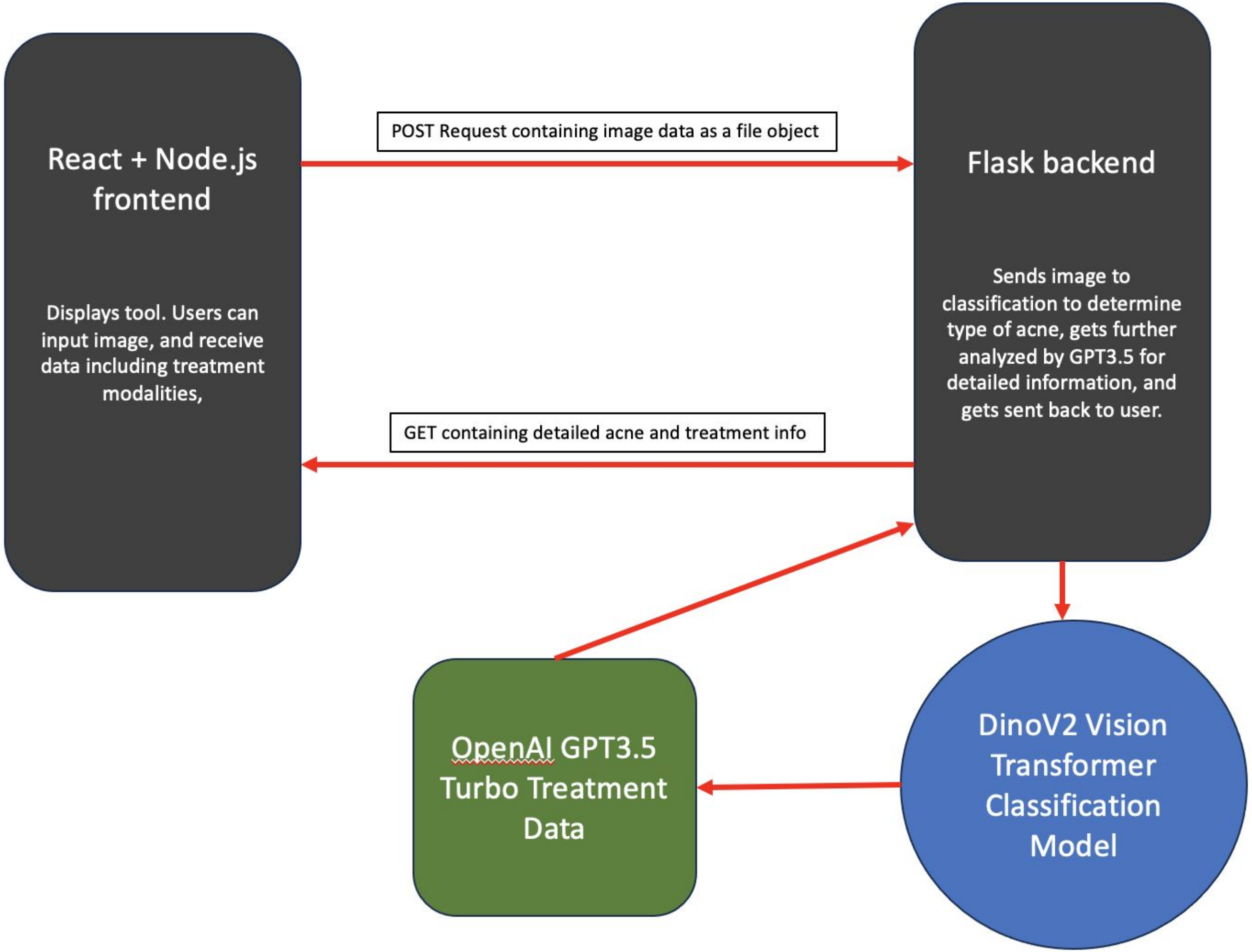
Acne AI

By Sooren Ghodsi, Kunsh Singh, Carlos Revilla, Dhruv Batra

OUR DESIGN



React + Node.js frontend

Displays tool. Users can input image, and receive data including treatment modalities,

POST Request containing image data as a file object

Flask backend

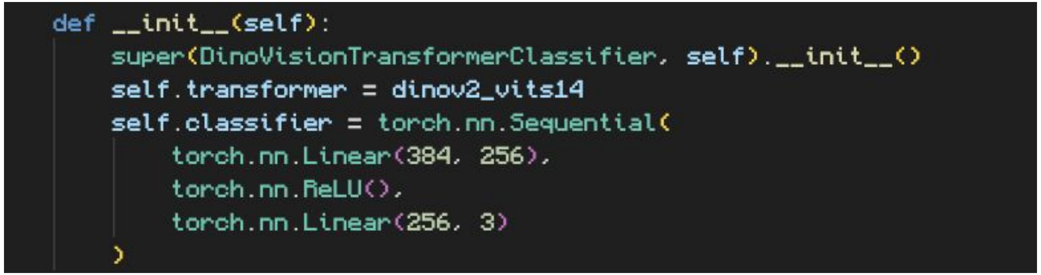
Sends image to classification to determine type of acne, gets further analyzed by GPT3.5 for detailed information, and gets sent back to user.

GET containing detailed acne and treatment info

OpenAI GPT3.5 Turbo Treatment Data

DinoV2 Vision Transformer Classification Model

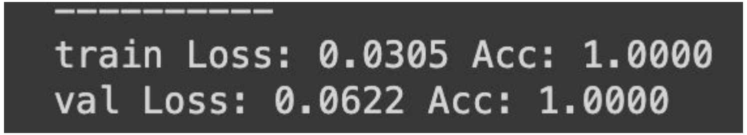
DinoV2 Model



```
def __init__(self):
    super(DinoVisionTransformerClassifier, self).__init__()
    self.transformer = dinov2_vits14
    self.classifier = torch.nn.Sequential(
        torch.nn.Linear(384, 256),
        torch.nn.ReLU(),
        torch.nn.Linear(256, 3)
    )
```

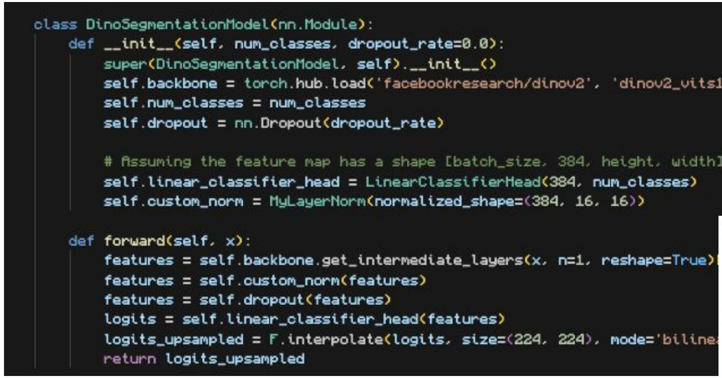
- Roboflow for datasets (754 open-source annotated +augmented images)
- https://universe.roboflow.com/taschenbier/acne-type-classification (sign-in before accessing, used v3)

- Classifies 3 classes: acne-comedonica, acne-conglobata, and acne-papulopustulosa
- Scheduler (step_size=9, gamma=0.1), Adam optimizer, CrossEntropyLoss function
- Trained on a T4 GPU (Google Colab) for 10 epochs

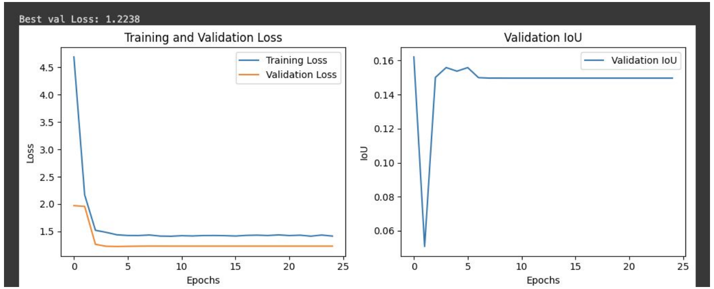


```
----------
train Loss: 0.0305 Acc: 1.0000
val Loss: 0.0622 Acc: 1.0000
```

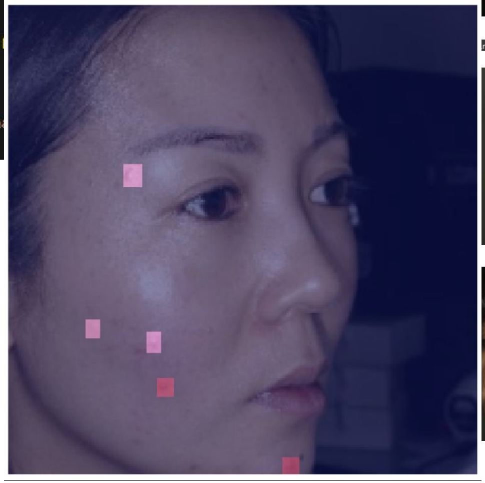
Future: Segmentation Model

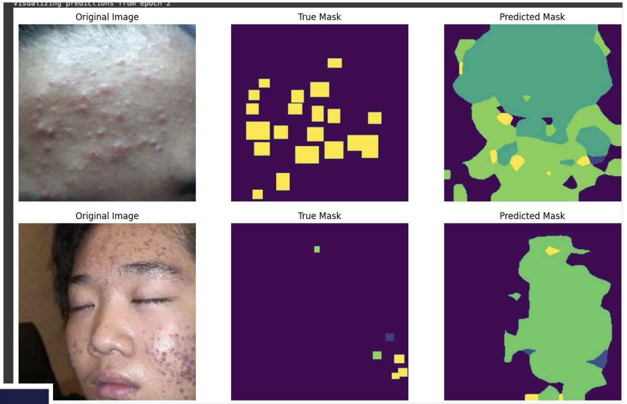


```
class DinoSegmentationModel(nn.Module):
    def __init__(self, num_classes, dropout_rate=0.0):
        super(DinoSegmentationModel, self).__init__()
        self.backbone = torch.hub.load('facebookresearch/dinov2', 'dinov2_vits14', pretrained=True)
        self.num_classes = num_classes
        self.dropout = nn.Dropout(dropout_rate)

        # Assuming the feature map has a shape [batch_size, 384, height, width]
        self.linear_classifier_head = LinearClassifierHead(384, num_classes)
        self.custom_norm = MyLayerNorm(normalized_shape=(384, 16, 16))

    def forward(self, x):
        features = self.backbone.get_intermediate_layers(x, n=1, reshape=True)[
        features = self.custom_norm(features)
        features = self.dropout(features)
        logits = self.linear_classifier_head(features)
        logits_upsampled = F.interpolate(logits, size=(224, 224), mode='biline
        return logits_upsampled
```

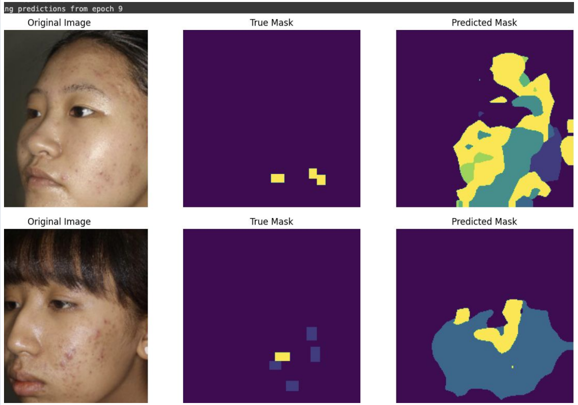


```
Best val Loss: 1.2238
```



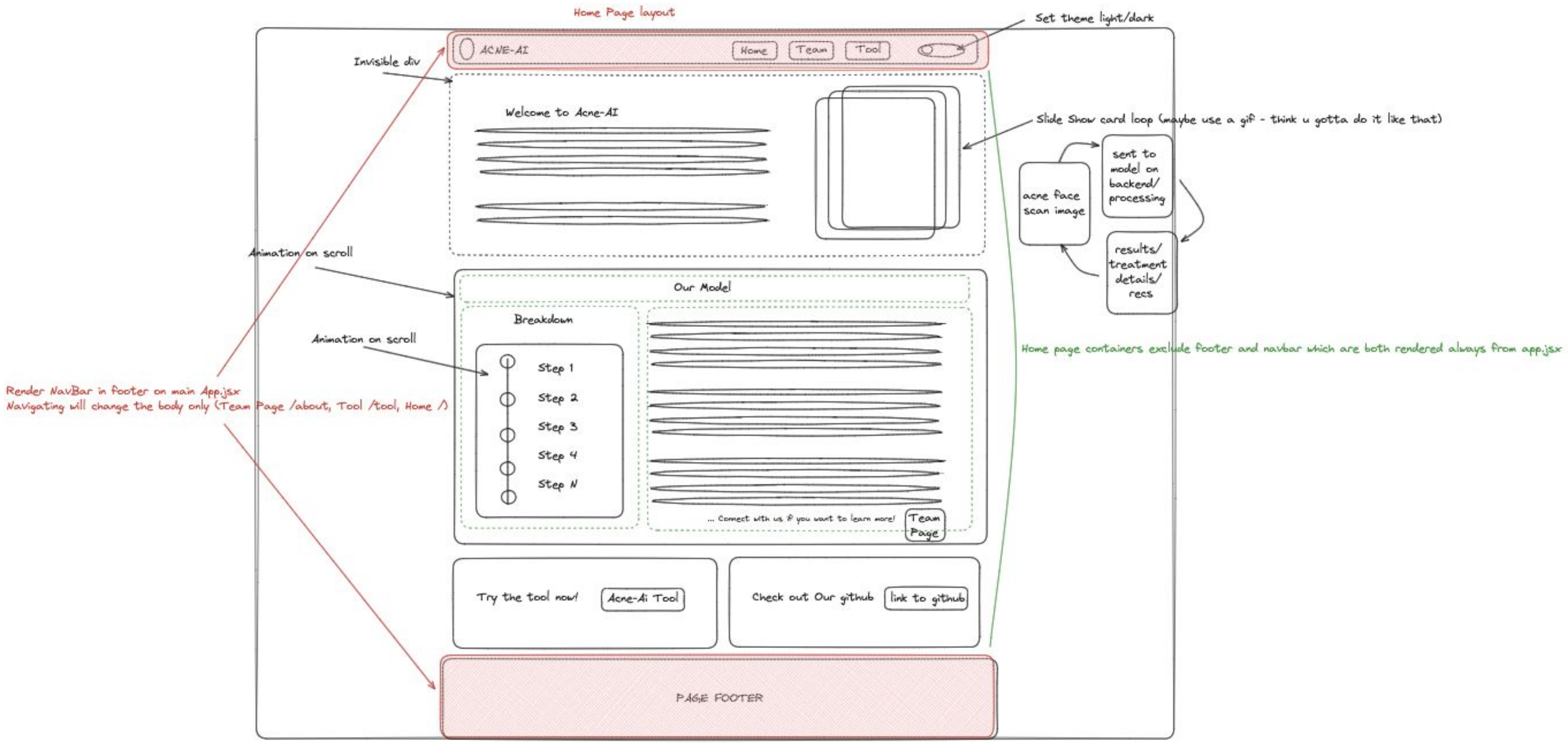


```
Visualizing predictions from epoch 2
```



```
ng predictions from epoch 9
```

React and Node.js Frontend

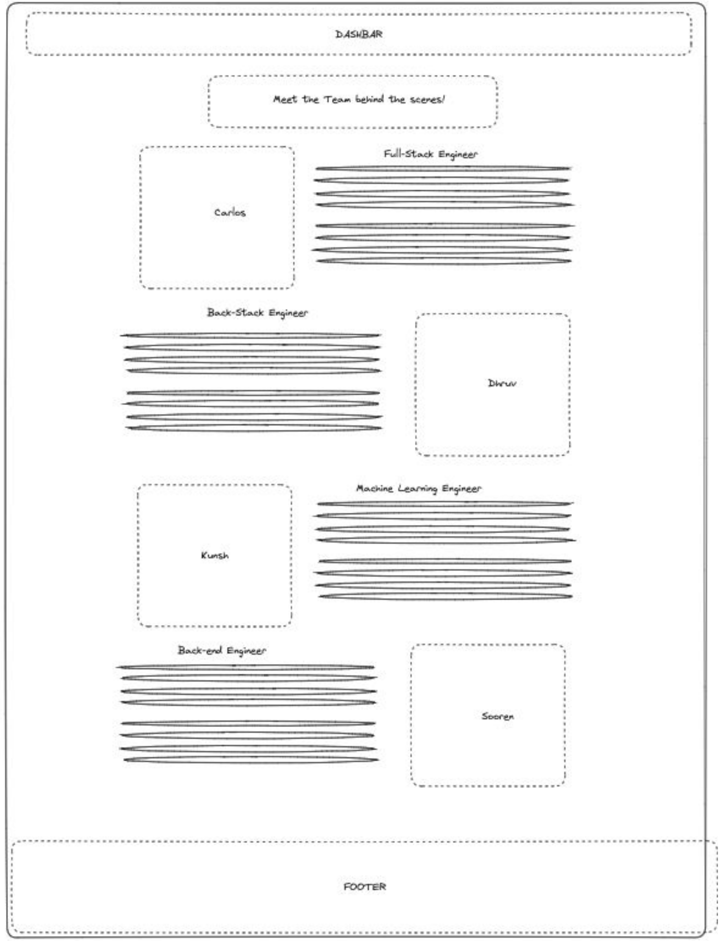


Set theme light/dark

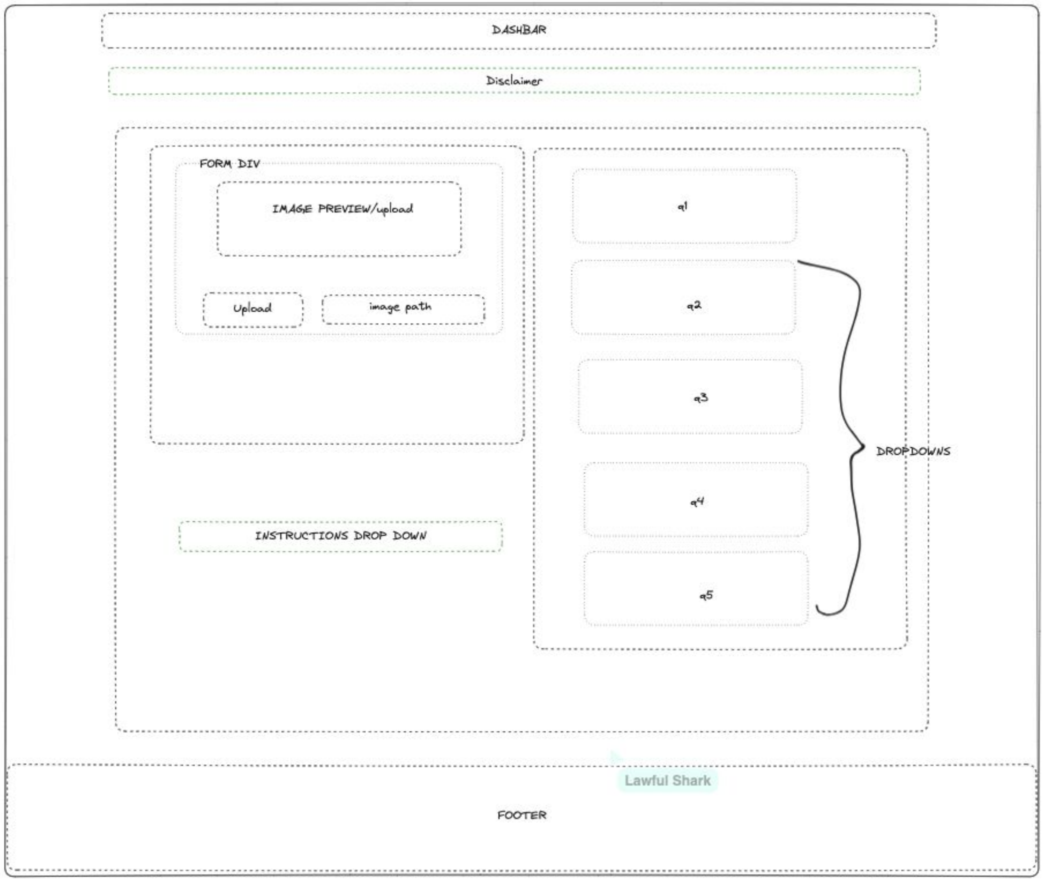
Slide Show card loop (maybe use a gif - think u gotta do it like that)

Home page containers exclude footer and navbar which are both rendered always from app.jsx

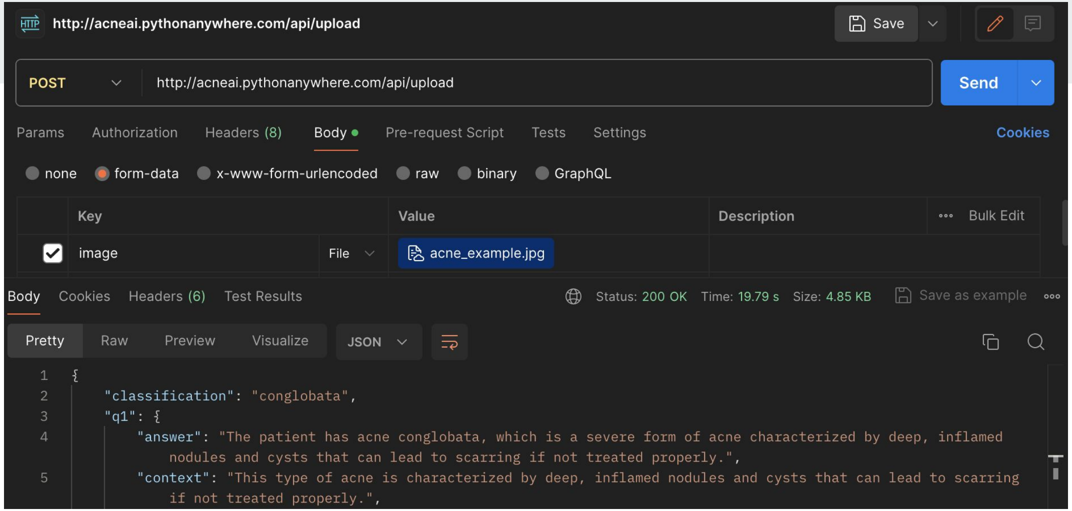
Render NavBar in footer on main App.jsx
Navigating will change the body only (Team Page /about, Tool /tool, Home /)


DASHBAR
Meet the Team behind the scenes!
Full-Stack Engineer
Carlos
Back-Stack Engineer
Dhruv
Machine Learning Engineer
Kunsh
Back-end Engineer
Sooren
FOOTER

TOOL PAGE


DASHBAR
Disclaimer
FORM DIV
IMAGE PREVIEW/upload
Upload
image path
INSTRUCTIONS DROP DOWN
q1
q2
q3
q4
q5
DROPDOWNS
Lawful Shark
FOOTER

Flask Backend



http://acneai.pythonanywhere.com/api/upload

POST http://acneai.pythonanywhere.com/api/upload

	Key		Value	Description
☑	image	File	acne_example.jpg	

Status: 200 OK Time: 19.79 s Size: 4.85 KB

```
{
    "classification": "conglobata",
    "q1": {
        "answer": "The patient has acne conglobata, which is a severe form of acne characterized by deep, inflamed
            nodules and cysts that can lead to scarring if not treated properly.",
        "context": "This type of acne is characterized by deep, inflamed nodules and cysts that can lead to scarring
            if not treated properly.",
```

Image Upload

- User clicks upload button.
- Initiates a POST request to the Flask server.

Machine Learning Model

- Processes and classifies the type of acne.

OpenAI Integration

- Acne type used to generate custom content.
- Provides treatment options, skincare tips, and causes.

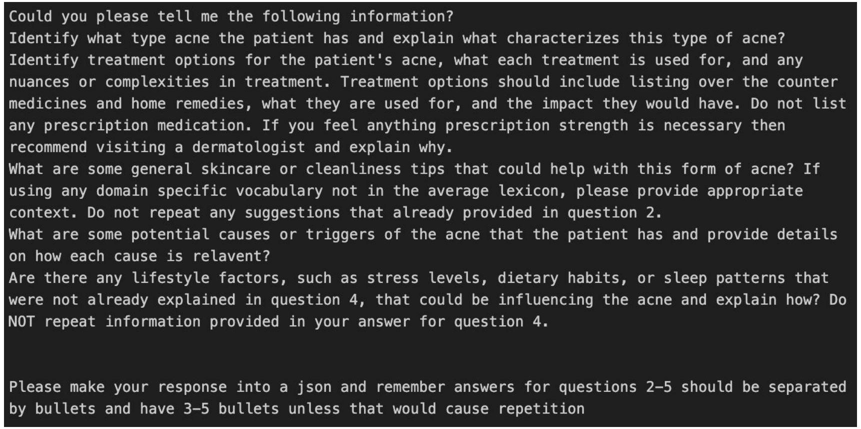
Response Handling

- POST request returns a JSON.
- Contains comprehensive acne information and a unique key.

OpenAI Data

GPT 3.5 TURBO

- Combine predefined prompts with classified acne type.
- Send combined prompt to OpenAI GPT-3.5-turbo for tailored skincare recommendations.
- Process OpenAI's JSON response; extract and return treatment plans.



```
Could you please tell me the following information?
Identify what type acne the patient has and explain what characterizes this type of acne?
Identify treatment options for the patient's acne, what each treatment is used for, and any
nuances or complexities in treatment. Treatment options should include listing over the counter
medicines and home remedies, what they are used for, and the impact they would have. Do not list
any prescription medication. If you feel anything prescription strength is necessary then
recommend visiting a dermatologist and explain why.
What are some general skincare or cleanliness tips that could help with this form of acne? If
using any domain specific vocabulary not in the average lexicon, please provide appropriate
context. Do not repeat any suggestions that already provided in question 2.
What are some potential causes or triggers of the acne that the patient has and provide details
on how each cause is relavent?
Are there any lifestyle factors, such as stress levels, dietary habits, or sleep patterns that
were not already explained in question 4, that could be influencing the acne and explain how? Do
NOT repeat information provided in your answer for question 4.


Please make your response into a json and remember answers for questions 2-5 should be separated
by bullets and have 3-5 bullets unless that would cause repetition
```

THANK YOU FOR WATCHING