

OpenAI Application Customer Service: Send Email to Customer

By Sharon Cao
San Francisco Bay University



Table of Contents

01

Introduction

04

Test

02

Design

05

Conclusion

03

Implementation

06

Enhancement



01

Introduction_

Project Settings_

- You're a customer service assistant for a large electronics store
- The website of the store allows the customers to select language.
- The store's products
- Belong to different categories
- Each product has detailed description



Desired Interface_

Language	
Question	Answer
Submit	



02

Design_

Design_



Step 1

Generate a customer's comment

Input

The products' detailed descriptions



Output

A 100 words comment about the products.



Step 2

Generate email subject

Input

The customer's comment



Output

Generate the subject of an email from the customer's comment using Inferring technique



Step 4

Sentiment analysis of the customer's comment



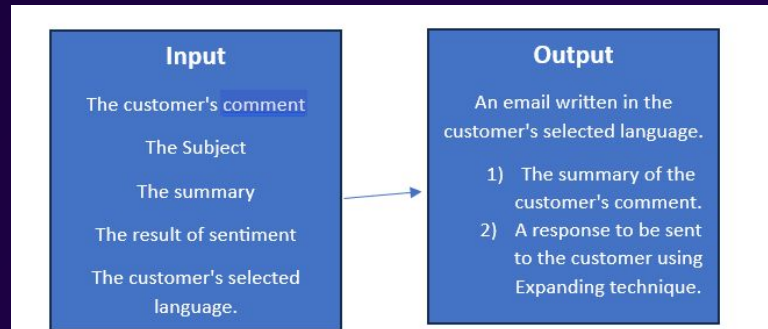
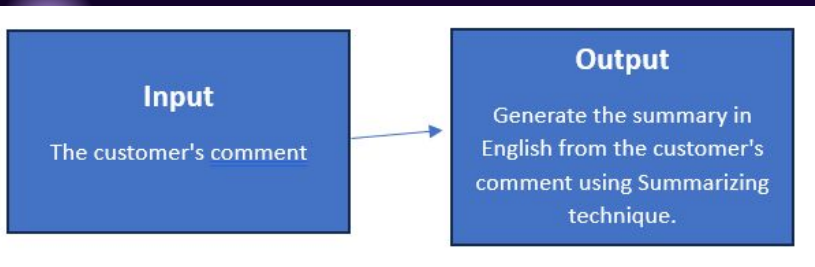
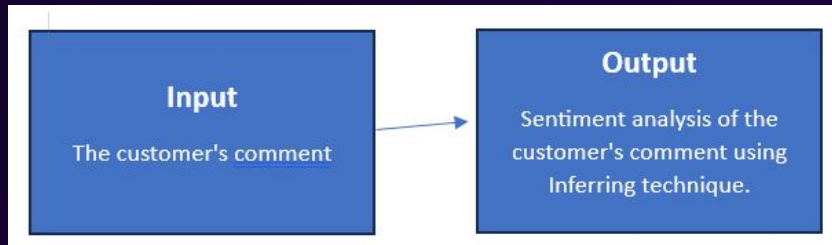
Step 3

Generate the summary of the customer's comment



Step 5

Generate an email to be sent to the customer.



Test Cases_

- Test cases

ID	Question	Answer
1	English	English
2	English	Non-English
3	Non-English	English
4	Non-English	Non-English
		<p>Note:</p> <ul style="list-style-type: none">ChatGPT can infer the language used in Question and then generate the Answer with the same language



03

Implementation_

Way to Implement_

Step by Step

Follow the steps, achieve the results one by one

Simple Code

Write simple code blocks for each steps

Playground

Use OpenAI playground to pre-test the result of the model get back.

<https://platform.openai.com/playground>

Combine the Blocks

Combine the chain of code together and test result.

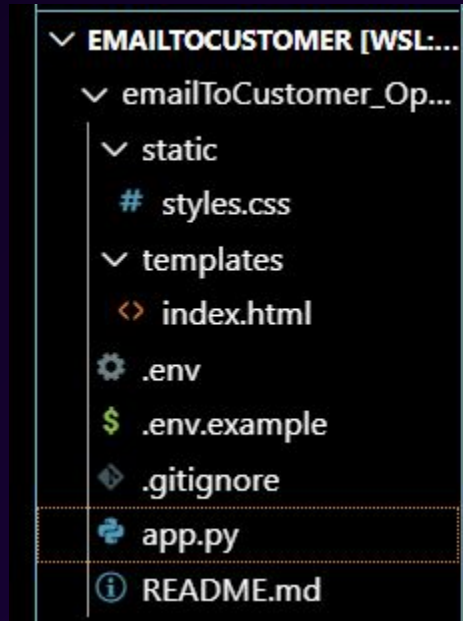
Convert Code to Project

Use Flask framework to convert the code into web based application.

Use ChatGPT

Use ChatGPT to fine tune the project to look better.

Struct_



Using Flask framework

Keep your API_KEY in .env file

General Function_

```
# Define your delimiter
delimiter = "####"

# General function to get response with gpt-3.5
def get_completion_from_messages(messages,
                                model="gpt-3.5-turbo",
                                temperature=0,
                                max_tokens=1000):
    response = openai.ChatCompletion.create(
        model=model,
        messages=messages,
        temperature=temperature,
        max_tokens=max_tokens,
    )
    return response.choices[0].message["content"]
```

Step 1_

```
# Step 1: Generate a customer's comment
# System message include product details
def generate_comment():
    system_message_comment=f"""
    Product details can be found as below

    "TechPro Ultrabook":
        "name": "TechPro Ultrabook",
        "category": "Computers and Laptops",
        "brand": "TechPro",
        "model_number": "TP-UB100",
        "warranty": "1 year",
        "rating": 4.5,
        "features": ["13.3-inch display", "8GB RAM",
                    | "256GB SSD", "Intel Core i5 processor"],
        "description": "A sleek and lightweight ultrabook for everyday use.",
        "price": 799.99

    "BlueWave Gaming Laptop":
        "name": "BlueWave Gaming Laptop",
        "category": "Computers and Laptops",
        "brand": "BlueWave",
        "model_number": "BW-GL200",
        "warranty": "2 years",
        "rating": 4.7,
        "features": ["15.6-inch display", "16GB RAM",
                    | "512GB SSD", "NVIDIA GeForce RTX 3060"],
        "description": "A high-performance gaming laptop with a high refresh rate display.",
        "price": 1299.99
```

```
user_message_comment=f"""
A less than 100 words comment about the products"""

messages_comment = [
    {'role':'system',
     'content': system_message_comment},
    {'role':'user',
     'content': f"{delimiter}{user_message_comment}{delimiter}"},
    {'role':'assistant',
     'content':'talk as a customer'}
]
comment = get_completion_from_messages(messages_comment)
print("Comment from customers: ")
print(comment+"\n")
return comment
```

Product detail information
exceed the token limit, so I
did shorten the list.

Step 2_

```
# Step 2: Generate email subject
def get_subject(comment):
    system_message_subject=comment
    user_message_subject=f""
    Subject of an email from the comment using Inferring technique within 10 words""
    messages_subject = [
        {'role': 'system',
         'content': system_message_subject},
        {'role': 'user',
         'content': f"{delimiter}{user_message_subject}{delimiter}"},
    ]
    subject = get_completion_from_messages(messages_subject)
    print("Subject of customer comment: ")
    print(subject+"\n")
    return subject
```


Step 3: Generate the summary of the customer's comment

```
def get_summary(comment):
    system_message_summary=comment
    user_message_summary=f"""
    Give the summary in English of the comment using Summarizing technique within 35 words."""
    messages_summary = [
        {'role':'system',
        'content': system_message_summary},
        {'role':'user',
        'content': f"{delimiter}{user_message_summary}{delimiter}"},
    ]
    summary=get_completion_from_messages(messages_summary)
    print("Summary of customer comment:")
    print(summary+"\n")
    return summary
```

Step 3: Translate to user language

```
def get_translation(summary, language):
    system_message_translate=summary
    user_message_translate=f"""
    Translate the summary into {language} using Transforming technique"""
    messages_translate = [
        {'role':'system',
        'content': system_message_translate},
        {'role':'user',
        'content': f"{delimiter}{user_message_translate}{delimiter}"},
    ]
    translate=get_completion_from_messages(messages_translate)
    print("Translation of customer comment summary in "+language+":")
    print(translate+"\n")
    return translate
```

Step 3_

Step 4_

```
#step 4: Sentiment analysis of the customer's comment
def get_sentiment(comment):
    system_message_sentiment=comment
    user_message_sentiment=f""
    Sentiment analysis of the customer's comment using Inferring technique. Positive or Negative?""
    messages_sentiment = [
        {'role':'system',
        'content': system_message_sentiment},
        {'role':'user',
        'content': f"{delimiter}{user_message_sentiment}{delimiter}"},
    ]
    sentiment=get_completion_from_messages(messages_sentiment)
    print(sentiment+"\n")
    return sentiment
```

Step 5_

```
# Step 5: Generate an email to be sent to the customer
def get_email(comment, subject, summary, sentiment):
    system_message_email = comment + subject + summary + sentiment
    user_message_email = f""
    Please create an email to be sent to the customer based on {comment}, including the {subject}, {summary} and {sentiment} with proper email format
    messages_email = [
        {'role': 'system',
         | 'content': system_message_email},
        {'role': 'user',
         | 'content': f"{delimiter}{user_message_email}{delimiter}"},
    ]
    email = get_completion_from_messages(messages_email)
    return email # Return the email without printing it
```

Main_

```
@app.route('/', methods=['GET', 'POST'])
def index():
    comment = None
    email = None
    language = "English" # Initialize the language variable here

    if request.method == 'POST':
        language = request.form['language']
        comment = generate_comment()
        subject = get_subject(comment)
        summary = get_summary(comment)
        sentiment = get_sentiment(comment)
        email = get_email(comment, subject, summary, sentiment)

        # Check if the user wants to translate the comment and email
        translate_comment = 'translate-comment' in request.form
        translate_email = 'translate-email' in request.form

        if translate_comment:
            comment = get_translation(comment, language)

        if translate_email:
            email = get_translation(email, language)

    return render_template('index.html', comment=comment, email=email, language=language)

if __name__ == '__main__':
    app.run(debug=True)
```

To run the project:

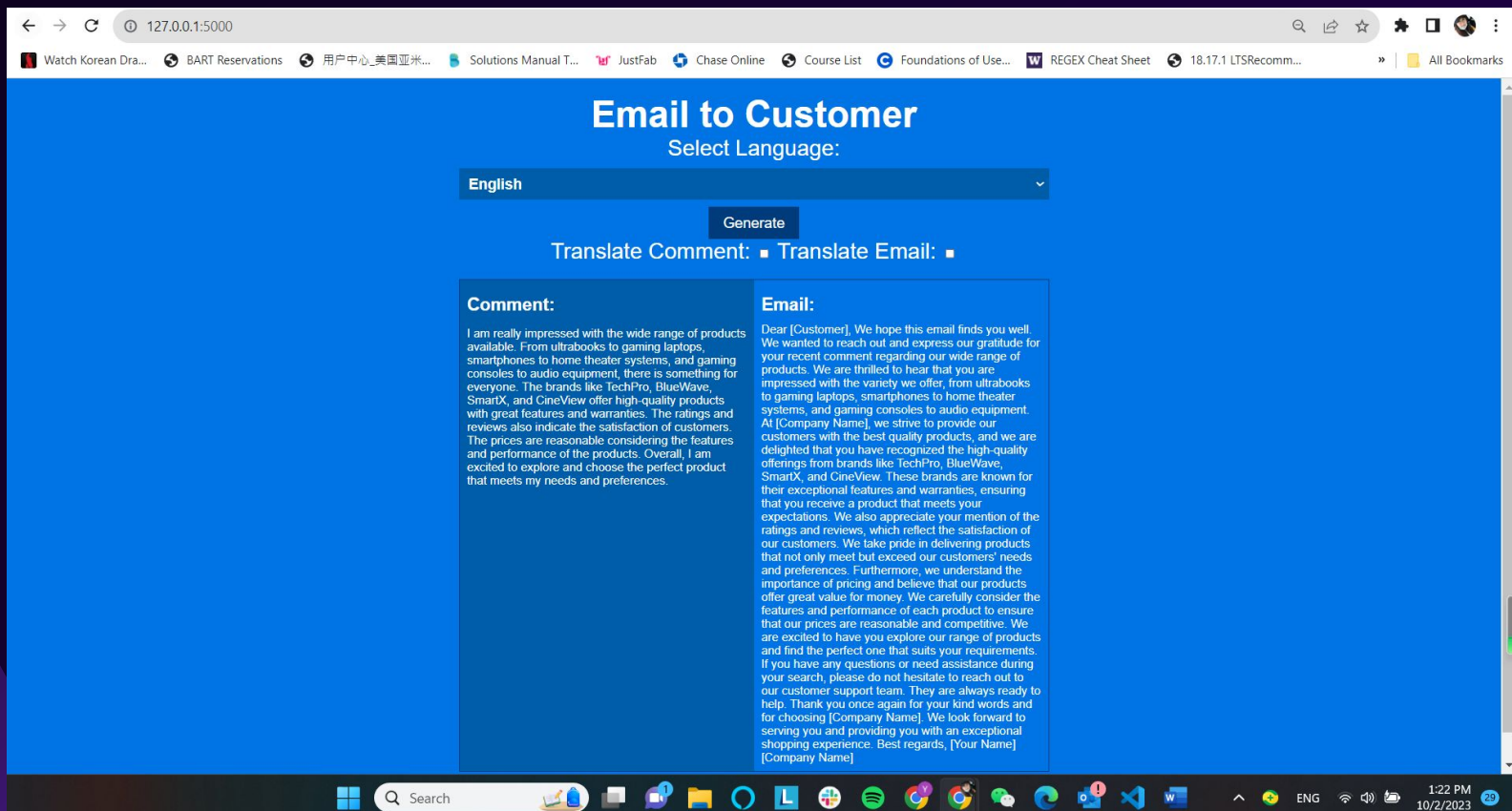
\$ python3 app.py



04

Test_

English->English_



English -> Non-English_

← → ↺ 127.0.0.1:5000

Watch Korean Dra... BART Reservations 用户中心_美国亚米... Solutions Manual T... JustFab Chase Online Course List Foundations of Use... REGEX Cheat Sheet 18.17.1 LTSRecomm... All Bookmarks

Email to Customer

Select Language:

English

Generate

Translate Comment: ■ Translate Email: ■

Comment:

I am really impressed with the wide range of products available. From ultrabooks to gaming laptops, smartphones to home theater systems, and gaming consoles to audio equipment, there is something for everyone. The brands like TechPro, BlueWave, SmartX, and CineView offer high-quality products with great features and warranties. The ratings and reviews also indicate the satisfaction of customers. The prices are reasonable considering the features and performance of the products. Overall, I am excited to explore and choose the perfect product that meets my needs and preferences.

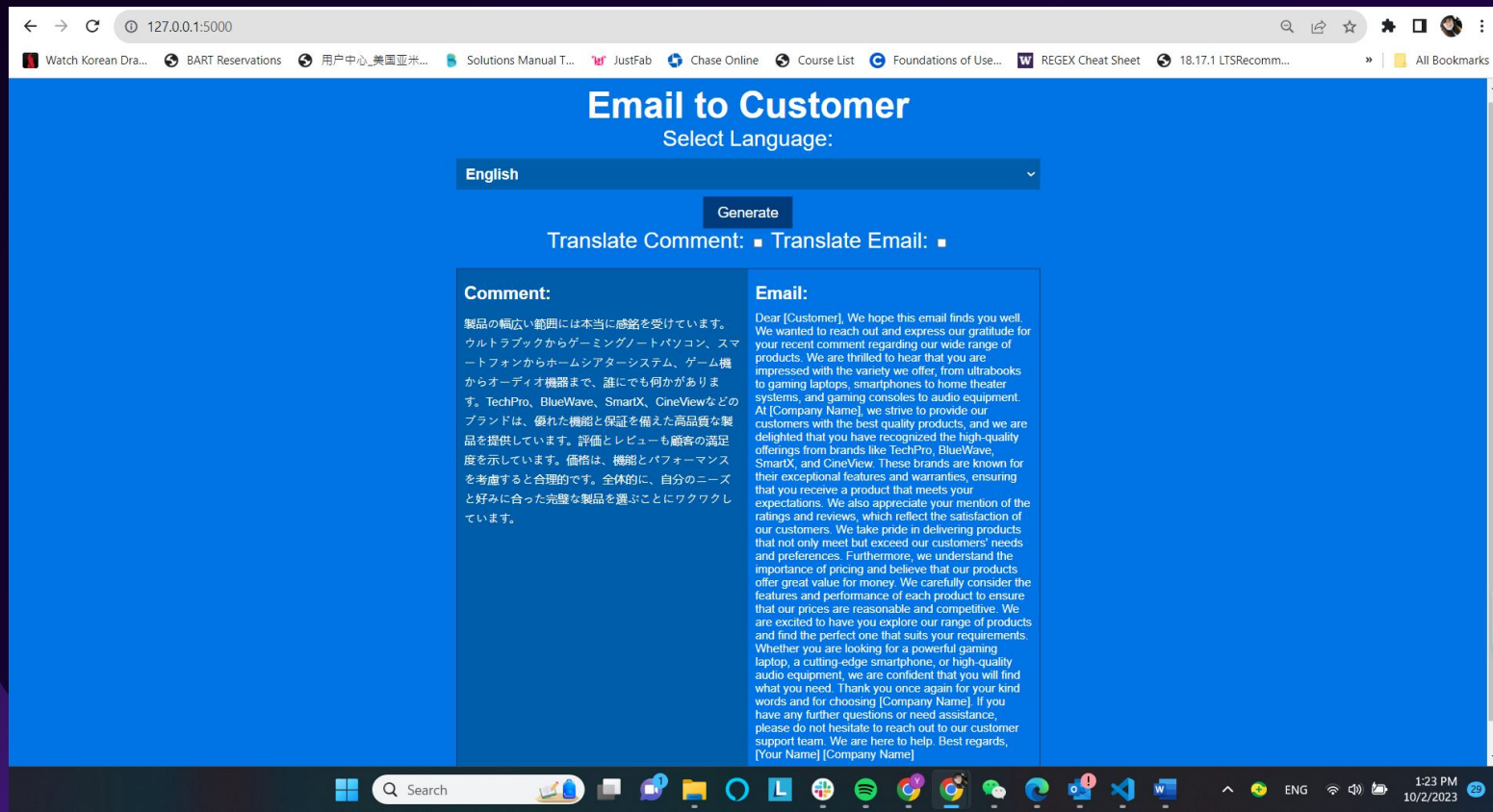
Email:

[고객님께], 안녕하세요. 이메일이 잘 도착했기를 바랍니다. 최근에 저희 제품 다양성에 대한 소중한 의견을 주셔서 감사드립니다. 저희가 제공하는 다양한 제품에 대해 귀하께서 인상 깊게 생각하신다는 것을 듣게 되어 기쁩니다. 저희는 울트라북부터 게이밍 노트북, 스마트폰부터 홈 시어터 시스템, 게임 콘솔부터 오디오 장비까지 다양한 제품을 제공하고 있습니다. [회사명]은 고객님께 최상의 품질 제품을 제공하기 위해 노력하고 있으며, 테크프로, 블루웨이브, 스마트엑스, 시네뷰와 같은 브랜드의 고품질 제품에 대한 연식을 고마워합니다. 이러한 브랜드들은 탁월한 기능과 보증을 제공하여 고객님의 기대에 부응하는 제품을 제공합니다. 또한, 고객님의 제품 평가와 리뷰에 대한 언급에도 감사드립니다. 우리는 고객님의 만족도를 반영하는 제품을 제공하는 데 자부심을 가지고 있습니다. 뿐만 아니라, 가격에 대한 중요성을 이해하고 있으며, 저희 제품들은 가성비가 뛰어난 것으로 알려져 있습니다. 우리는 각 제품의 기능과 성능을 신중하게 고려하여 합리적이고 경쟁력 있는 가격을 제시하고 있습니다. 고객님의께서 저희 제품 다양성을 탐색하시고 요구에 맞는 완벽한 제품을 찾으실 수 있기를 기대합니다. 강력한 게이밍 노트북, 최신 스마트폰, 최고급 홈 시어터 시스템 또는 기타 전자 기기를 찾고 계시다면, 고객님의 요구에 부응하는 제품을 찾을 수 있다고 자신합니다. 다시 한번 소중한 의견과 [회사명]을 선택해 주셔서 감사드립니다. 추가 문의 사항이 있으시거나 도움이 필요하시면 언제든지 고객 지원팀에 문의해 주시기 바랍니다. 도움을 드리기 위해 항상 준비되어 있습니다. 감사합니다. [귀하의 이름] [회사명]

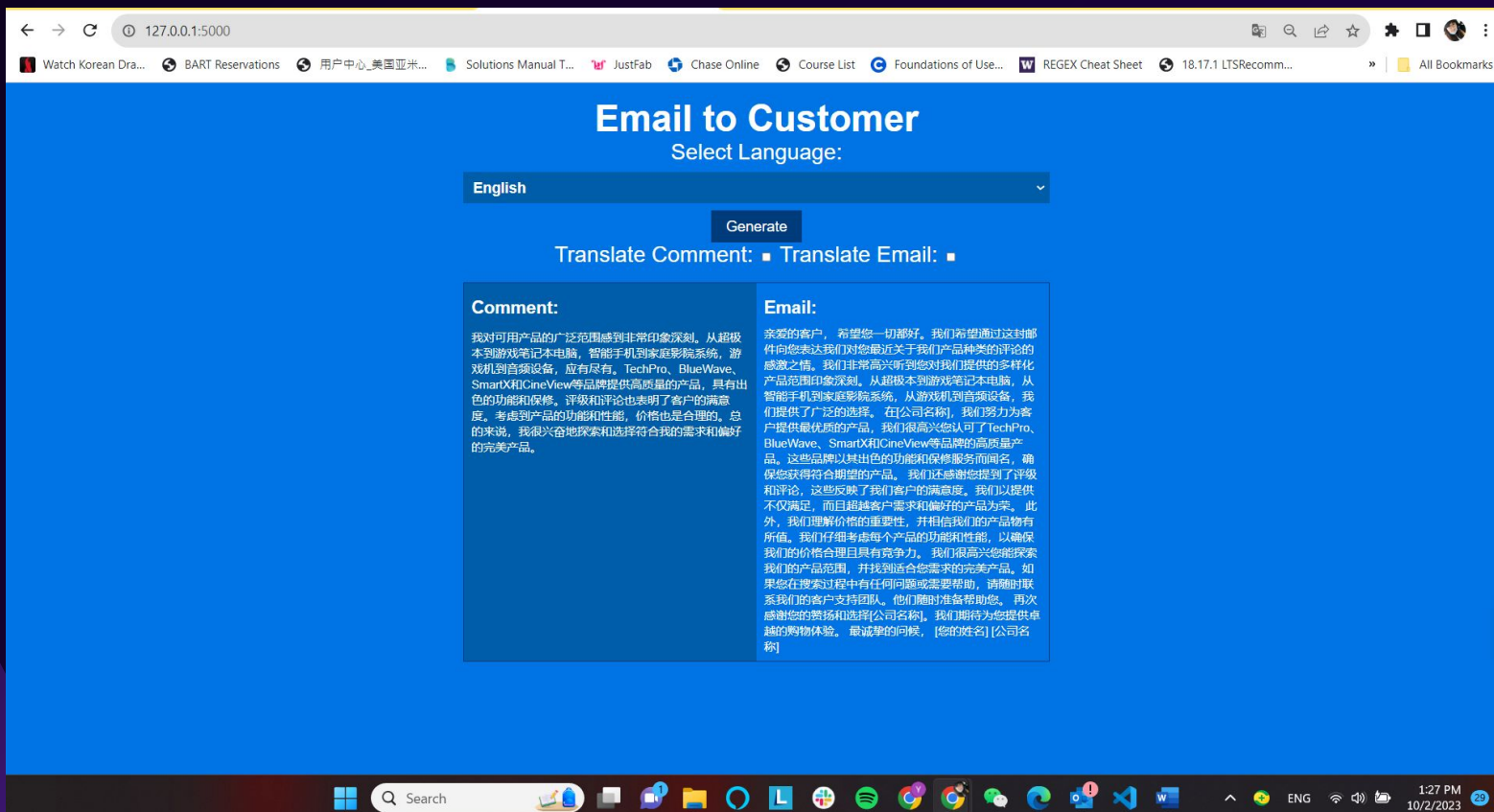
Search

1:25 PM 10/2/2023

Non-English->English_



Non-English -> Non-English_



Terminal_

127.0.0.1 - - [02/Oct/2023 02:19:19] "GET / HTTP/1.1" 200 -

Comment from customers:

I am really impressed with the wide range of products available. From ultrabooks to gaming laptops, smartphones to home theater systems, and gaming consoles to audio equipment, there is something for everyone. The brands like TechPro, BlueWave, SmartX, and CineView offer high-quality products with great features and warranties. The ratings and reviews also indicate the satisfaction of customers. The prices are reasonable considering the features and performance of the products. Overall, I am excited to explore and choose the perfect product that meets my needs and preferences.

Subject of customer comment:

Impressed with the wide range of products available.

Summary of customer comment:

The commenter is impressed with the wide range of products available, including laptops, smartphones, gaming consoles, and audio equipment. They mention brands like TechPro, BlueWave, SmartX, and CineView, and are excited to find a product that meets their needs.

Translation of customer comment summary in Japanese:

コメント者は、ラップトップ、スマートフォン、ゲーム機、オーディオ機器など、幅広い製品の取り扱いに感心しています。TechPro、BlueWave、SmartX、CineViewなどのブランドに触れ、自分のニーズに合った製品を見つけることに興奮しています。

Positive

Terminal_

The email to be sent to the customer is as below:

Dear [Customer],

We hope this email finds you well. We wanted to reach out and express our gratitude for your recent comment regarding our wide range of products. We are thrilled to hear that you are impressed with the variety of options we offer, from ultrabooks to gaming laptops, smartphones to home theater systems, and gaming consoles to audio equipment.

At [Company Name], we strive to provide our customers with high-quality products that meet their needs and preferences. We are proud to offer brands like TechPro, BlueWave, SmartX, and CineView, which are known for their exceptional features and warranties. We are glad to hear that you appreciate the quality and performance of our products.

Furthermore, we understand the importance of customer satisfaction, and we are delighted to hear that the ratings and reviews reflect the positive experiences of our customers. We value your feedback and are committed to continuously improving our products and services.

In terms of pricing, we aim to offer competitive prices that are reasonable considering the features and performance of our products. We believe in providing value for money and ensuring that our customers receive the best possible products at affordable prices.

We are excited to have you as a customer and look forward to assisting you in finding the perfect product that meets your needs and preferences. If you have any questions or require further assistance, please do not hesitate to reach out to our customer support team. They are available to provide guidance and help you make an informed decision.

Thank you once again for your positive feedback. We appreciate your support and look forward to serving you in the future.

Best regards,

[Your Name]

[Company Name]



05

Conclusion_

Conclusion_

In conclusion, this project successfully implemented an automated customer support process utilizing ChatGPT, revolutionizing our approach to handling customer inquiries and feedback for our electronic product company. Through a well-structured series of steps, we were able to streamline the customer support workflow, providing several benefits:

- **Efficiency:** Automation significantly reduced response time, enabling us to address customer queries promptly and effectively.
- **Consistency:** The use of ChatGPT ensured uniformity in responses, maintaining a high standard of customer service.
- **Multilingual Support:** We could cater to customers in their preferred languages, enhancing the overall user experience.

Conclusion_

- **Sentiment Analysis:** The sentiment analysis helped us gauge customer satisfaction, enabling us to take proactive measures when negative sentiments were detected.
- **Resource Allocation:** Human agents could focus on more complex issues, while routine inquiries were handled automatically.

This project represents a significant leap forward in customer support, offering a seamless and efficient experience for our customers. It's worth noting that continuous monitoring and improvement will be essential to refine the automated process further. We look forward to leveraging the power of ChatGPT in our ongoing commitment to providing exceptional customer service.



06

Enhancement_

Ideas for Future_



User Feedback Integration

Implement a feedback system to continually refine responses based on customer input.



Customization Options

Allow customers to personalize their interactions for a tailored experience.



Advanced Sentiment Analysis

Enhance sentiment analysis to provide more empathetic and relevant responses to customer emotions.

Thanks!_

Do you have any questions?



CREDITS: This presentation template was created by [Slidesgo](#), and includes icons by [Flaticon](#) and infographics & images by [Freepik](#)

Please keep this slide for attribution

