**Exercise 1 - Ticket Categorization-**

Query 1: Category: Bug-

Response: "Thank you for reaching out. It sounds like you are experiencing a technical issue with adding a new product to your pricing plan. We will investigate why the button is grayed out. In the meantime, please ensure that you have the necessary permissions to make changes. We will get back to you shortly with a solution."

Query 2: Category: feature Request-

Response: "Thank you for your suggestion! We appreciate your feedback regarding the automated monthly summaries of invoices. Currently, this feature is not available, but we will pass your request to our product team for consideration in future updates. If you have any other ideas or needs, please let us know!"

Query 3: Category: Customer Training-

Response: "Thank you for your inquiry! We understand that configuring a usage-based billing schema can be complex. We would be happy to provide you with guidance and documentation. Please check the attached document or let us know if you would like to schedule a training session for more personalized assistance."

**Exercise 2 – Onboarding Plan-**

**Onboarding Plan for New Client**

Step 1: Account Configuration

Objective: Set up the client's account to ensure they have access to all necessary features and settings.

Actions:

* Schedule a meeting to understand the client's specific needs and preferences.
* Guide the client through the account setup process, including:

1. Creating user profiles and permissions.
2. Configuring account settings .

Step 2: Uploading Usage Data

Objective: Assist the client in uploading their initial set of usage data .

Actions:

* Provide a template or format for the usage data to ensure consistency and accuracy.
* Offer a step-by-step walkthrough of the data upload process, including:

1. How to access the data upload feature.
2. Best practices for data entry and validation.
3. Troubleshooting common issues during the upload.

Step 3: Generating the First Invoice

Objective: Enable the client to generate their first invoice based on the uploaded usage data.

Actions:

* Demonstrate how to generate an invoice using the uploaded data.
* Review the invoice settings, including payment terms, invoice templates, and customization options.

Supporting Documents and Resources

1. **Onboarding Checklist**: A comprehensive checklist outlining all steps and tasks to be completed during the onboarding process.
2. **Account Configuration Guide**: A detailed document that includes screenshots and instructions for setting up the account and configuring settings.
3. **Data Upload Template:** A pre-formatted spreadsheet or document that the client can use to prepare their usage data for upload.
4. **Data Upload Instructions:** A step-by-step guide on how to upload data .
5. **Invoice Generation Manual:** A guide that explains how to generate and customize invoices, including examples of different invoice formats.
6. **FAQs Document:** A list of frequently asked questions and answers to address common concerns or issues that may arise during onboarding.
7. **Contact Information for Support:** A document with contact details for support resources, including email, phone numbers, and hours of availability.

By following this onboarding plan and utilizing the provided resources, the client will be well-equipped to successfully configure their account, upload their data, and generate invoices efficiently.

**Exercise 3 - Customer Feedback Analysis**

**Action Items to Address Client Feedback**

**UI/UX Redesign Initiative**

* Conduct User Research: Gather insights from users to understand specific pain points in the current UI. This can include surveys, or usability testing sessions.
* Hire a UI/UX Designer: Engage a professional designer to create a modern, intuitive interface that enhances navigation and overall user experience. Consider implementing a design system for consistency.

**Improve Support Ticket Response Times**

* Analyze Support Processes: Analyze the current support ticket workflow to identify bottlenecks causing delays. This may involve reviewing ticket categorization, prioritization, and response protocols.
* Increase Support Staff or Resources: If necessary, hire additional support staff or implement a ticketing system that automates responses for common inquiries to ensure timely follow-ups.

**Draft Email to Client**

Subject: Acknowledgment of Your Feedback and Next Steps

Dear [Client's Name],

Thank you for sharing your valuable feedback during our recent meeting. We truly appreciate your insights regarding Zenskar's functionality and the challenges you've faced with the user interface and support response times.

We are committed to enhancing your experience with our product. To address your concerns, we are taking the following steps:

UI/UX Redesign Initiative: We will conduct user research to better understand the specific areas of the interface that need improvement. Our goal is to create a more modern and intuitive design that enhances navigation.

Support Ticket Process Improvement: We are currently evaluating our support processes to identify any bottlenecks and will work on increasing our response times to ensure you receive timely assistance.

We will keep you updated on our progress and look forward to implementing these improvements. If you have any further suggestions or questions, please feel free to reach out.

Thank you for your continued partnership.

Best regards,

[Your Name]

**Zenskar API Integration Documentation**

1. Introduction

This document provides a step-by-step guide to setting up the environment, using the Zenskar API to create customers, products, and contracts programmatically in Python.

2. Environment Setup

2.1 Prerequisites

Python: Ensure Python is installed on your machine.

Pip: Python package manager should be installed (comes with Python).

2.2 Install Required Libraries

Open Command Prompt

Install the requests library by running:

2.3 Create a Python Script

Open a text editor (e.g., Notepad, VSCode).

Copy the provided Python script into the editor.

Replace YOUR\_API\_KEY and YOUR\_ORG\_ID with your actual Zenskar API key and Organization ID.

Save the file with a .py extension (e.g., zenskar\_script.py).

3. API Endpoints Used

Create Customer: POST /customers

Create Products: POST /products

Create Contract: POST /contracts

Get Customers: GET /customers (for testing purposes)

4. Python Script Explanation

The following sections explain the key components of the Python script.

4.1 Authentication

The authenticate function sets up the headers required for API requests, including the API key.

4.2 Create Customer

The create\_customer function sends a POST request to create a new customer with specified details.

4.3 Create Products

The create\_product function creates products based on the provided parameters (name, type, billing, frequency, price, and quantity).

4.4 Create Contract

The create\_contract function combines the created products into a contract for the specified customer.

4.5 Main Execution

The script executes the functions in sequence to create a customer, products, and a contract.

6. Challenges Faced

6.1 Authorization Error

Issue: Encountered the error {'Message': 'User is not authorized to access this resource with an explicit deny'} when trying to create a customer.

Resolution: Verified API key and permissions. Contacted Zenskar support for assistance.

6.2 Testing API Endpoints

Issue: Difficulty in testing the API due to lack of valid credentials.

Resolution: Used mock APIs and public APIs for practice until valid credentials were obtained.

7. Conclusion

This document outlines the steps to set up the environment and interact with the Zenskar API using Python. By following the instructions, users can create customers, products, and contracts programmatically.