Engineering Intern Technical Challenge: AI Support Assistant

🎯 Overview

Your challenge is to **design and implement a simple AI-powered system** that automates a basic customer support use case using any tools or frameworks you prefer (Python, JavaScript, Node.js, etc.).

This exercise evaluates your **technical ability**, **clarity of thinking**, and **communication skills**.

📦 Use Case: Support Ticket Assistant

> A logistics company receives multiple emails every day from drivers asking about shipment status, delivery issues, and payment updates.
>
> Your goal is to **build a simple tool** that classifies incoming support messages and stores them in a database.

🤖 Objective 1: Implement a Simple Classifier

- Build a small **script or web app** that:
  - Reads support messages (you can use a simple `.csv` or `.json` file with 10–20 example messages).
  - Automatically **classifies** each message into one of 3 categories:
    - `Shipment Status`
    - `Delivery Issue`
    - `Payment / Invoice`
  - Optionally: detects the **sentiment** (positive / negative / neutral).

> 💡 Hint: You can use any open-source NLP model (e.g., OpenAI API, Hugging Face, or your own keyword-based logic).

📊 Objective 2: Display the Results

- Create a **small dashboard or console output** showing:
  - The list of messages and their predicted category.
  - (Optional) Sentiment analysis results.
  - A simple **count of messages per category**.

> 💡 If you prefer, you can do this in a notebook (e.g., Jupyter / Colab) or as a small web app (Flask, Streamlit, React, etc.).

⚙️ Objective 3: Deployment and Documentation

- Containerize your solution with **Docker** (optional but recommended).
- Create a **README.md** in your repo explaining:
  - How to run your code locally.
  - Which libraries or tools you used.
  - How your logic works.

🧪 Deliverables

1. **GitHub Repository** with your code and instructions.
2. **Short video (max. 5 minutes)** in English explaining:
   - What you built.
   - How it works (short demo).
   - What you would improve with more time.

⏱️ Recommended Time

This challenge is designed to be completed in **1 day** (4–6 hours of work).

We're not evaluating the size of the project — but **clarity, execution, and communication**.

📬 Submission

Once finished, please send:

- Your **GitHub repo link**
- Your **demo video link**
- Send it to david@happyrobot.ai - cc ewen@happyrobot.ai and varez@happyrobot.ai