**Hackathon Submission Document**

**Project Title**

Customer Frustration Alerting System Snap-In

**Team/Participant Details**

* **Name**: SINCHANA S D
* **Contact**: shpsinchanasd@gmailcom
* **GitHub Repository Link**: https://github.com/SinchanaSD-26/DevRev-Hackathon
* **Video Demo Link**: https://youtu.be/VclYouNZ0f8?si=aKamoSJxXo6XRpHY

**Problem Statement**

**Challenge**: Detect customer frustration in real-time and automatically notify support agents to intervene promptly.  
**Objective**: Build a Snap-In that analyzes customer conversations, identifies frustration, and triggers alerts through Slack and email notifications.

**Solution Overview**

The **Customer Frustration Alerting System Snap-In** solves the problem by:

* Using **sentiment analysis** to evaluate customer conversation text.
* Detecting frustration based on a configurable threshold.
* Sending **real-time alerts** via Slack and email to support agents.

## ****Key Features****

1. **Real-Time Sentiment Analysis**:
   * Uses the Sentiment library to calculate sentiment scores.
   * Identifies negative conversations.
2. **Alert Mechanism**:
   * **Slack Notifications**: Sends alerts to a designated Slack channel.
   * **Email Alerts**: Sends email notifications to support agents.
3. **Configurable Threshold**:
   * Adjustable thresholds for sentiment detection via a config.json file.

## ****Architecture Overview****

## Data Flow:

1. Input: Customer conversation text (mock data or live API).
2. Process:
   * Analyze sentiment using Sentiment.
   * Compare scores against the threshold in config.json.
   * Trigger Slack and email notifications for negative sentiment.
3. Output: Notifications sent to Slack and email.

### Tools and APIs Used:

* **Sentiment Analysis**: Sentiment library.
* **Slack API**: For sending notifications to channels.
* **Nodemailer**: For sending email alerts.
* **DevRev Snap-In APIs** (optional): For integration with customer workflows.

## ****Setup Instructions****

1. Clone the repository:

git clone [https://github.com/SinchanaSD-26/DevRev-Hackathon]

cd Customer-Frustration-Alerting-System

1. Install dependencies:

npm install

1. Configure API keys:
   * Create a .env file and add the following:
   * SLACK\_API\_TOKEN=slack-token
   * EMAIL\_USER=sinchanasd@gmail.com

EMAIL\_PASS=sinchana@123

1. Compile the TypeScript files:

npm run build

1. Run the application:

npm start

## ****How It Works****

### Workflow:

1. The system processes conversations from mockConversations.
2. Each conversation is analyzed using Sentiment to calculate a score.
3. If the score is below the threshold:
   * A Slack alert is sent to a designated channel.
   * An email notification is sent to the support agent.

### Commands Used:

* **Install Dependencies**: npm install
* **Build Project**: npm run build
* **Run Application**: npm start

### Example Input and Output:

#### Input:

[

{ "id": 1, "text": "This service is terrible!" },

{ "id": 2, "text": "I am very unhappy with this product." },

{ "id": 3, "text": "I love this!" }

]

#### Output:

Slack and email alerts for:

* "This service is terrible!"
* "I am very unhappy with this product."

## ****Video Demo****

Link: https://youtu.be/VclYouNZ0f8?si=aKamoSJxXo6XRpHY

## ****Future Improvements****

1. To integrate advanced NLP models for more accurate sentiment detection.
2. To provide detailed reporting and trend analysis for customer feedback.
3. To extend alert mechanisms to include SMS or in-app notifications.

## ****Conclusion****

The Customer Frustration Alerting System Snap-In efficiently detects customer frustration in real-time and ensures timely intervention, improving overall customer satisfaction and support efficiency. Thank you for reviewing our submission!