# Home Assignment – Team Lead & Senior Engineer

## Context

We're building a growing platform with increasing demand from clients to view and understand their data. Today, all visualizations are built manually, which doesn't scale. We need a solution that allows product, ops, and even clients to quickly gain insights from activity data.

This assignment is designed to test both backend and integration capabilities, including how you structure work, make architectural decisions, and deliver a clear, scalable outcome.

## Goal

Deploy the admin panel and PostgreSQL, insert meaningful dummy data across selected tables, integrate Apache Superset, and embed dashboards in the admin panel to visualize that data.

## Setup (To Be Done With the Team)

Ibrahim or Tim will walk you through starting the application (admin and PostgreSQL). Once running, you will:

1. Create 3 meaningful tables

Examples: client_sessions, conversation_steps, message_logs

2. Insert dummy data

- Data should reflect realistic conversations

- Vary conversation length, time, and flow structure

- Suggest using ORM or DB seed scripts – avoid manual SQL

## Superset Integration

3. Set up Apache Superset

- Connect it to the same PostgreSQL instance

- Load the 3 tables

- Build 2 or more dashboards showing:

- Flow-level statistics (e.g. steps per conversation, drop-off rates)

- Time-based insights (e.g. average response time, flow duration)

- You can include other useful views as you see fit

4. Embed Superset in the Admin

- Suggest a way to render iframes or embed Superset dashboards

- Add a clear section in the admin for analytics

- Make sure it loads correctly and is easy to navigate

## Deliverables

- Code and config for admin, DB, and Superset

- Dummy data logic (script/seeding)

- Dashboards (with saved state or screenshots)

- Short README explaining:

- Table and data choices

- What insights the dashboards provide

- Assumptions or limitations

## Review Criteria

- Code quality and commits

- Structure and clarity of setup

- Relevance and realism of data

- Usefulness and clarity of dashboards

- Quality of admin integration

- Explanation and reasoning in README