Intern Screening Task – Hybrid Web + Desktop Application

You are required to build a hybrid application that runs both as a Web Application and a Desktop Application. The project focuses on data visualization and analytics for chemical equipment.

Project Title:

Chemical Equipment Parameter Visualizer (Hybrid Web + Desktop App)

Project Overview

Create a Web + Desktop application that allows users to upload a CSV file containing a list of chemical equipment with columns such as Equipment Name, Type, Flowrate, Pressure, and Temperature. The Django backend will parse the data, perform analysis, and provide summary statistics via API. Both React (Web) and PyQt5 (Desktop) frontends will consume this API to display data tables, charts, and summaries.

Tech Stack (Fixed)

Layer	Technology	Purpose
Frontend (Web)	React.js + Chart.js	Show table + charts
Frontend (Desktop)	PyQt5 + Matplotlib	Same visualization in desktop
Backend	Python Django + Django REST Framework	Common backend API
Data Handling	Pandas	Reading CSV & analytics
Database	SQLite	Store last 5 uploaded datasets
Version Control	Git & GitHub	Collaboration & submission
Sample Data	sample_equipment_data.csv	Sample CSV file provided for testing & demo

Key Features Required

1. CSV Upload – Web and Desktop must allow users to upload a CSV file to the backend.
2. Data Summary API – Django API should return total count, averages, and equipment type distribution.
3. Visualization – Display charts using Chart.js (Web) and Matplotlib (Desktop).

4. History Management – Store last 5 uploaded datasets with summary.
5. Generate PDF report and add basic authentication.
6. **(Extra)** Use the provided sample CSV (sample_equipment_data.csv) for demo and testing.

Task Instructions

Develop both Web and Desktop frontends connected to the same Django backend. The application should demonstrate proper data handling, API integration, and UI/UX consistency.

Submission Guidelines

• Source code on GitHub (backend + both frontends)

• README file with setup instructions (should be made available with the Github Source Code)

• Short demo video (2–3 minutes)

• Optional: Deployment link for web version

Submission Link

Kindly submit all the details in the google form: https://forms.gle/rEgLy6fQU1UgdB5LA