# 🧑‍💻 Project Proposal: Sales Process Automation using Python and AI/ML Concepts

## 🎯 Objective:

To automate the sales process through:  
1. Web scraping based on Ideal Customer Profile (ICP)  
2. Exporting data to Excel  
3. Email template creation  
4. Automated outreach using email  
5. Analytics collection to evaluate outreach effectiveness

## ✅ Task 1: Scraping Data Based on ICP

Objective:  
Scrape B2B company data from relevant websites based on a defined ICP.  
  
ICP Definition:  
- Industry: Information Technology  
- Location: Ahmedabad  
  
Source:  
- Website Scraped: Clutch.co  
  
Data Fields Extracted:  
- Company Name  
- Location  
- Size (Number of Employees)  
- Industry (fixed: Information Technology)  
  
Technologies Used:  
- curl\_cffi: For making requests while bypassing bot protections  
- scrapy.Selector: For HTML parsing  
- pandas: For data handling  
- SQLAlchemy + pymysql: For storing data in MySQL  
  
Output:  
- Data saved in MySQL  
 - Database: linkedin  
 - Table: data2  
  
Refer to ab.py for the scraping code.

## ✅ Task 2: Export Data into Excel Format

Objective:  
Export the scraped MySQL data to an Excel file for further processing.  
  
Steps:  
- Connect to the linkedin MySQL database  
- Fetch data from data2 table  
- Export to linkedin\_data.xlsx  
  
Refer to excel.py for the export script.

## ✅ Task 3: Create a Dynamic Email Template

Objective:  
Create a personalized HTML email template for outreach.  
  
Features:  
- HTML structure with placeholders: {{ name }}, {{ company }}  
- Includes:  
 - Tracking Pixel → /track/<tracking\_id>.png  
 - Call-to-Action Link → /click/<tracking\_id>

## ✅ Task 4: Automate the Email Campaign

Objective:  
Send personalized emails to leads and manage batch email campaigns.  
  
Email Sending Page:  
- Flask-based UI  
- Inputs: Sender email, password, number of recipients  
- Source: recipients.xlsx (contains name, company, email)  
- Sends emails in batches using user-defined count

## ✅ Task 5: Gathering Analytics from the Campaign

Objective:  
Track user engagement and categorize leads based on interaction.  
  
MongoDB Integration:  
- Stores email status: Sent, Opened, Clicked  
- Encrypts emails using Caesar Cipher  
  
Status Page:  
- Visual indicators:  
 - ✅ = Link clicked (Hot Lead)  
 - ❌ = No click (Cold Lead)  
  
Analytics Page:  
- Total Emails Sent  
- Total Clicked  
- Hot Leads (Clicked)  
- Cold Leads (Not Clicked)  
- Click Rate %

## ✅ Task 6: Recording a Demo Video

Objective:  
Demonstrate the full working of the project.  
  
Method:  
- Recorded using OBS Studio  
- Covered:  
 - Web scraping  
 - Excel export  
 - Email tracking app  
 - Live email sending and result tracking  
  
Uploaded to GitHub and linked in README

## ✅ Task 7: Documentation and GitHub Repository

Objective:  
Provide a comprehensive overview of the project and code.  
  
Deliverables:  
- GitHub Repository containing:  
 - Web scraping script  
 - Excel export script  
 - Email tracking app  
 - Demo video walkthrough  
- README.md includes:  
 - Step-by-step explanation  
 - Libraries and tools used  
 - Challenges faced and skills applied

## 📌 Project Deliverables

1. GitHub Repository  
 - Python scripts  
 - Email app with UI  
 - Analytics Dashboard  
 - Demo Video  
 - README.md  
  
2. Demo Video  
 - Full process walkthrough  
 - Face visible, narrated explanation

## 📈 Skills Evaluation Criteria

- ✅ Python Programming: requests, BeautifulSoup, pandas, smtplib  
- ✅ Web Scraping & Extraction: From Clutch.co using modern tools  
- ✅ Email Campaign Automation: Flask, tracking links, and batch sending  
- ✅ AI/ML Concepts: Email personalization, user activity analysis  
- ✅ Reporting & Automation: Analytics dashboard with lead scoring