Perfect 👍 — let’s build **Motivation Profile**, a simple publishable web app that:

1. Asks all **50 questions** (with 1–5 scale answers).
2. Automatically calculates the **10 motivation category scores**.
3. Generates both:
   * a **bar chart** (scores by category)
   * a **spider/radar chart** (visual profile)
4. Lets the user **export and print results**.

The best way to do this quickly and make it publishable is **Streamlit**, since it’s lightweight, interactive, and deployable (Streamlit Cloud, Hugging Face, or your own server).

Use this code and remove the lower code

import streamlit as st

import matplotlib.pyplot as plt

import numpy as np

import pandas as pd

import io

from fpdf import FPDF

# ----------------------------

# Survey Setup

# ----------------------------

categories = [

"Financial Rewards",

"Career Advancement",

"Leadership & Influence",

"Work-Life Balance",

"Learning & Growth",

"Recognition & Value",

"Autonomy & Independence",

"Purpose & Impact",

"Stability & Security",

"Team & Relationships"

]

questions = { ... } # (Keep the same 50 questions dictionary from before)

# ----------------------------

# Streamlit App

# ----------------------------

st.title("Motivation Profile")

st.write("Answer the following questions on a scale of 1 (Not Important) to 5 (Extremely Important).")

category\_scores = {}

for category, qs in questions.items():

st.subheader(category)

answers = [st.slider(q, 1, 5, 3) for q in qs]

category\_scores[category] = np.mean(answers)

# ----------------------------

# Charts + Report

# ----------------------------

if st.button("Generate Results"):

st.subheader("Your Motivation Profile Results")

df = pd.DataFrame(list(category\_scores.items()), columns=["Category", "Score"])

# ----- Bar Chart -----

st.bar\_chart(df.set\_index("Category"))

# ----- Radar Chart -----

N = len(categories)

values = df["Score"].tolist()

values += values[:1]

angles = np.linspace(0, 2 \* np.pi, N, endpoint=False).tolist()

angles += angles[:1]

fig, ax = plt.subplots(figsize=(6, 6), subplot\_kw=dict(polar=True))

ax.plot(angles, values, color="blue", linewidth=2)

ax.fill(angles, values, color="blue", alpha=0.25)

ax.set\_xticks(angles[:-1])

ax.set\_xticklabels(categories, fontsize=9)

ax.set\_yticks([1, 2, 3, 4, 5])

ax.set\_ylim(0, 5)

ax.set\_title("Motivation Radar Profile", size=14, weight="bold", pad=20)

st.pyplot(fig)

# ----- Interpretation -----

st.subheader("Your Top Motivators")

top3 = df.sort\_values(by="Score", ascending=False).head(3)

for i, row in top3.iterrows():

st.write(f"\*\*{row['Category']}\*\*: This is a key driver of your motivation with a score of {row['Score']:.1f}.")

st.subheader("Areas of Lower Motivation")

bottom3 = df.sort\_values(by="Score", ascending=True).head(3)

for i, row in bottom3.iterrows():

st.write(f"\*\*{row['Category']}\*\*: Less of a priority right now (score {row['Score']:.1f}). May not energize you compared to other factors.")

# ----- Export CSV -----

csv = df.to\_csv(index=False).encode('utf-8')

st.download\_button(

label="Download Results as CSV",

data=csv,

file\_name="motivation\_profile\_results.csv",

mime="text/csv",

)

# ----- Export Full PDF Report -----

def create\_pdf(dataframe, radar\_fig, top3, bottom3):

pdf = FPDF()

pdf.add\_page()

pdf.set\_font("Arial", 'B', 16)

pdf.cell(200, 10, "Motivation Profile Report", ln=True, align="C")

# Add bar chart

pdf.set\_font("Arial", size=12)

pdf.cell(200, 10, "Bar Chart of Motivation Scores", ln=True)

buf\_bar = io.BytesIO()

dataframe.plot(kind="bar", x="Category", y="Score", legend=False).figure.savefig(buf\_bar, format="png", bbox\_inches="tight")

buf\_bar.seek(0)

pdf.image(buf\_bar, x=30, y=None, w=150)

# Add radar chart

pdf.ln(10)

pdf.cell(200, 10, "Radar Chart of Motivation Profile", ln=True)

buf\_radar = io.BytesIO()

radar\_fig.savefig(buf\_radar, format="png", bbox\_inches="tight")

buf\_radar.seek(0)

pdf.image(buf\_radar, x=30, y=None, w=150)

# Add interpretation

pdf.ln(10)

pdf.set\_font("Arial", 'B', 14)

pdf.cell(200, 10, "Top Motivators", ln=True)

pdf.set\_font("Arial", size=12)

for i, row in top3.iterrows():

pdf.multi\_cell(0, 10, f"- {row['Category']}: Strong driver of your motivation (Score {row['Score']:.1f})")

pdf.ln(5)

pdf.set\_font("Arial", 'B', 14)

pdf.cell(200, 10, "Lower Motivators", ln=True)

pdf.set\_font("Arial", size=12)

for i, row in bottom3.iterrows():

pdf.multi\_cell(0, 10, f"- {row['Category']}: Lower priority for you right now (Score {row['Score']:.1f})")

# Output PDF

pdf\_buffer = io.BytesIO()

pdf.output(pdf\_buffer)

return pdf\_buffer

pdf\_buffer = create\_pdf(df, fig, top3, bottom3)

st.download\_button(

label="Download Full PDF Report",

data=pdf\_buffer,

file\_name="motivation\_profile\_report.pdf",

mime="application/pdf",

)