

# Founder-Investor Matcher Web Application

## 1. Introduction

The **Founder-Investor Matcher** is a Flask-based web application designed to help startup founders find suitable investors based on industry and startup stage. By uploading structured CSV files containing founder and investor data, users can identify potential investment opportunities efficiently.

## 2. Objectives

The main objectives of the application are:

- To provide a simple web-based platform for founders to find relevant investors.
- To filter investors based on **industry** and **startup stage**.
- To enable easy CSV file uploads for dynamic matching.

## 3. Methodology

The application follows a structured process:

1. **Data Input:** Users upload CSV files for founders and investors.
2. **Data Processing:** The system reads the CSV files and extracts relevant details.
3. **Matching Algorithm:**
  - The founder's industry is compared with investor preferences.
  - The startup stage is matched with investor interests.
4. **Output Display:** The filtered list of investors is shown on a simple HTML page.

## 4. Technologies Used

- **Backend:** Flask (Python-based web framework)
- **Frontend:** HTML (Jinja2 templating for dynamic content)
- **Data Handling:** Pandas (for CSV reading and processing)

## 5. Implementation Details

- **Uploading CSV Files:** The app allows users to upload **founders.csv** and **investors.csv**.
- **Dropdown Selection:** Users select a founder from the uploaded data.

- **Filtering Process:** Investors are matched based on relevant criteria.
- **User Interface:** A minimalistic webpage displays the matched investors.

## 6. Results & Discussion

The application successfully loads founders and investors, processes the data, and displays suitable investors for a selected founder. The matching system works efficiently, but the API-based match scoring was removed due to rate limits.

### Challenges Faced

- **API Limitations:** Originally, the app used the **Gemini API** for score-based matching, but quota exhaustion errors led to its removal.
- **Data Dependencies:** The effectiveness of the app depends on the quality and completeness of the uploaded CSV files.

## 7. Conclusion

This Flask app simplifies the founder-investor matching process by providing a straightforward, user-friendly solution. With further enhancements, it can incorporate advanced filtering mechanisms and improved UI/UX design for a more seamless experience.

## 8. Future Enhancements

- **Advanced Matching:** Adding ML-based investor-founder compatibility scoring.
- **Database Integration:** Storing and retrieving data using SQL databases instead of CSV files.
- **Enhanced UI:** Improving frontend design for a more interactive experience.