SOFTWARES AND CONFIGURATION STEPS

This project integrates various development and deployment tools that work together to support machine learning model training, web development, and database management. Below is a detailed explanation of the software used and their configuration steps.

1. Python (Version 3.10 or Higher)

Purpose:

Core programming language for developing machine learning models and backend logic using Flask.

Configuration Steps:

1. Download Python from https://www.python.org/downloads.
2. During installation, check the box **"Add Python to PATH."**
3. Verify installation using:

   python –version

4. Install necessary Python libraries:

   pip install pandas numpy scikit-learn xgboost flask seaborn matplotlib

2. Anaconda

Purpose:

Used for managing Python environments and packages efficiently. Simplifies the installation of scientific libraries and tools.

Configuration Steps:

1. Download Anaconda from https://www.anaconda.com/products/distribution.
2. Install and launch **Anaconda Navigator** or **Anaconda Prompt**.
3. Create a new environment:

   conda create -n upi_fraud_env python=3.10

   conda activate upi_fraud_env

4. Install required libraries:
   conda install pandas numpy scikit-learn flask
   pip install xgboost seaborn matplotlib

3. Flask (Python Web Framework)

Purpose:

Flask was used to build the backend web application, enabling user registration, dataset uploads, model training, and prediction.

Configuration Steps:

1. Flask is installed via pip:

   pip install flask

2. A basic Flask app includes:

```
from flask import Flask
app = Flask(__name__)
@app.route('/')
def home():
  return "Welcome to UPI Fraud Detection"
if __name__ == '__main__':
 app.run(debug=True)
```

4. XAMPP Server

Purpose:

Used to host the MySQL database locally, manage tables for user authentication, transaction data, and prediction results.

Configuration Steps:

1. Download and install XAMPP from https://www.apachefriends.org.

2. Start the **Apache** and **MySQL** modules in the XAMPP Control Panel.
3. Access **phpMyAdmin** via http://localhost/phpmyadmin.
4. Create a new database (e.g., upi_fraud_detection) and required tables using SQL or the GUI.

5. Node.js

Purpose:

Used for managing frontend dependencies or supporting additional features like dynamic behavior, APIs, or npm packages.

Configuration Steps:

1. Download Node.js from https://nodejs.org.
2. Install and verify using:

   node -v

   npm -v

3. Initialize a project (if using npm packages for frontend):

   npm init -y

   npm install

📊 Summary Table

Software	Purpose	Key Role
Python	ML Model Development	Core programming language
Anaconda	Environment & Dependency Management	ML Environment Setup
Flask	Web Application Framework	Backend API/Web Pages
XAMPP	Local Server & MySQL Database Management	Data Storage & Access
Node.js	JavaScript Runtime & Dependency Management	Frontend Enhancements