WEB-X CA PREREQUISITES
DONOR CONNECT

Name of Student	**Shravani Anil Patil**
Class Roll no	**D15A-37**
D.O.P	**20.03.25**
D.O.S	**27.03.25**
Sign and Grade	

Introduction:

Donor Connect is a smart blood donation platform designed to bridge the gap between blood donors and those in urgent need. It provides a secure and seamless environment where requestors can post blood requirements and donors can respond based on their availability and blood group compatibility. With real-time status tracking, location-based filtering, and user authentication, Donor Connect ensures efficient and life-saving connections in times of need.

System Requirements:

1. Hardware Requirements:

- **Processor:** Intel Core i5 or higher
- **RAM:** Minimum 8GB (Recommended: 16GB)
- **Storage:** At least 10GB of free disk space
- **Operating System:** Windows 10/11, macOS, or Linux

2. Software Requirements:

- **Node.js (v18.16.1 or later)** – For running the React frontend
- **npm (v9.5.1 or later)** – To manage frontend dependencies
- **Python (3.11.9 or later)** – For the Flask backend
- **MongoDB Atlas** – Cloud-based NoSQL database
- **VS Code** – Recommended IDE for development
- **Postman** – For testing REST APIs

Technology Stack:

1. Frontend:

- **React.js** – JavaScript library for building user interfaces
- **Tailwind CSS** – Utility-first CSS framework for rapid styling

2. Backend:

- **Flask** – Lightweight Python-based web framework
- **Flask-CORS** – Handle Cross-Origin Resource Sharing
- **PyMongo** – Connects Flask with MongoDB
- **PyJWT** – Implements secure JWT-based authentication
- **python-dotenv** – Manages secure environment variables
- **flask-login** – Manages user sessions and secure login/logout

3. Database:

- **MongoDB Atlas** – Cloud-hosted NoSQL database to store donor and requestor data, blood requests, and responses

Setup Instructions:

1. Install Node.js and NPM

- **Download from: https://nodejs.org**
- **Verify installation:**

  node -v
  npm -v

2. Install Python and Flask

- Download Python from: https://www.python.org
- Verify installation:

  python --version
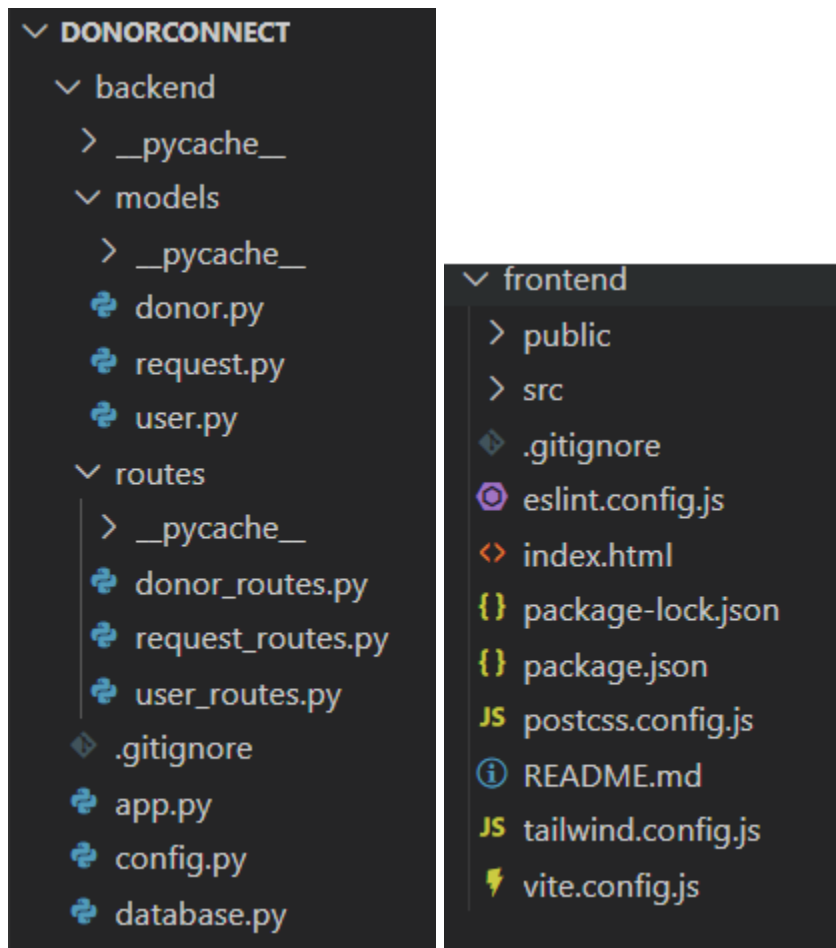 pip --version

- Install Flask and dependencies:

  pip install flask flask-cors pymongo pyjwt python-dotenv flask-limiter flask-login

3. Setup MongoDB Atlas

- Create a free MongoDB Atlas account at https://www.mongodb.com/cloud/atlas

- Replace connection string in the backend `.env` file

- For local setup (optional):

  mongod --dbpath /path/to/data

PROJECT STRUCTURE:

```
∨ DONORCONNECT
  ∨ backend
    > __pycache__
    ∨ models
      > __pycache__
      donor.py
      request.py
      user.py
    ∨ routes
      > __pycache__
      donor_routes.py
      request_routes.py
      user_routes.py
    .gitignore
    app.py
    config.py
    database.py
```

```
∨ frontend
  > public
  > src
  .gitignore
  eslint.config.js
  <> index.html
  {} package-lock.json
  {} package.json
  JS postcss.config.js
  i README.md
  JS tailwind.config.js
  vite.config.js
```

Conclusion:

This document outlines the key prerequisites for setting up the **Donor Connect** platform, ensuring a smooth and secure development environment. By following these guidelines and installing the required tools, developers can effectively collaborate and deploy a life-saving solution that connects donors with those in urgent need.