**Hierarchical Clinical Data Visualization Framework**

**READ ME:**

**Overview**

**The project offers an interactive, hierarchical visualization tool for exploring clinical datasets. Developed using Flask (backend) and D3.js (frontend), the framework supports real-time data exploration and filtering. This guide provides step-by-step instructions to set up and run the project in Visual Studio or similar environments.**

**System Requirements**

**Hardware Requirements**

* **RAM: 8GB or higher**
* **Processor: Intel Core i5 or equivalent**
* **Disk Space: 2GB free**

**Software Requirements**

* **Visual Studio 2022 (with Python and Web Development workloads installed)**
* **Python 3.10+ (ensure it is added to the system PATH)**
* **Node.js (for managing frontend dependencies)**
* **Git (for version control)**

**Step-by-Step Setup:**

* 1. **Extract the Archive.zip and**
* **let the files be in same dir where envo is created.**
* **Open Git Bash or any terminal. { I am using Visual studios}**
* **Clone the repository to your local system:**
* **git clone <repository-url>**
* **cd <repository-folder>**

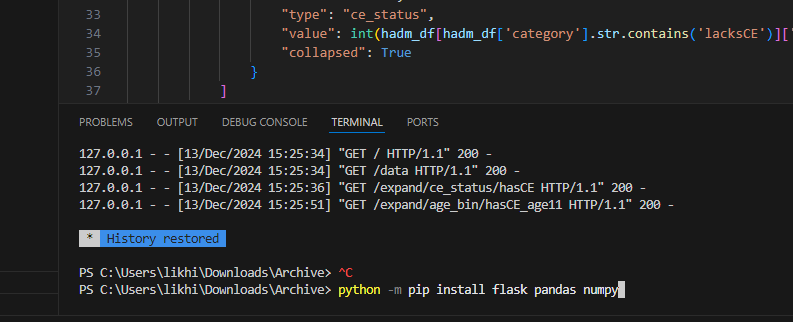
**2. Set Up the Python Environment**

1. **Open Visual Studio and select Clone a Repository to load the project.**
2. **Install the Python dependencies:**
   * **Open the terminal within Visual Studio or use a separate command line.**
   * **Run:**
   * **python -m venv venv**
   * **source venv/bin/activate # For Windows, use: venv\\Scripts\\activate**
   * **pip install -r requirements.txt**

**Example Output:**

**Collecting flask**

**Collecting pandas**

**Successfully installed flask pandas**

**3. Set Up the Frontend**

1. **Navigate to the frontend folder in the terminal:**
2. **cd frontend**
3. **npm install**

**This installs all Node.js dependencies required for the D3.js frontend.**

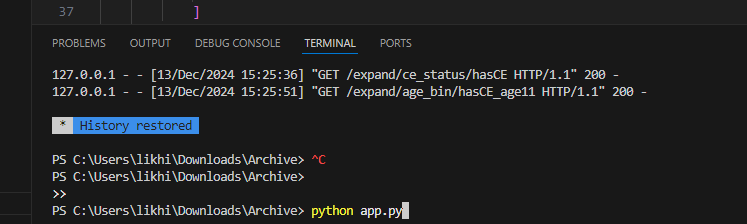
**A screen shot of a computer

Description automatically generated**

**4. Configure the Project**

1. **Locate the config.json file in the config folder.**
2. **Specify the paths to the clinical datasets (numOBS\_perAGE\_YR\_bin.csv, numHADM\_perAGE\_YR\_bin.csv).**

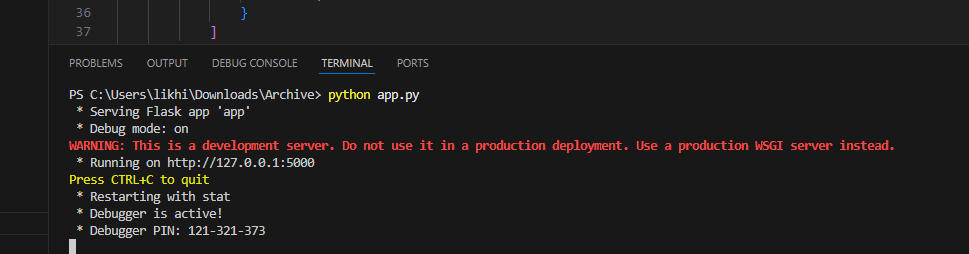
**5. Run the Backend**

1. **In the terminal, navigate to the project’s root folder:**
2. **python app.py**
3. **Flask will start the backend server, which can be accessed at http://127.0.0.1:5000.**

**Example Output:**

**\* Running on http://127.0.0.1:5000**

**\* Debug mode: on**

****

**6. Run the Frontend**

1. **Open a new terminal and navigate to the frontend folder:**
2. **cd frontend**
3. **npm start**
4. **The frontend will launch at** [**http://127.0.0.1:3000**](http://127.0.0.1:3000)**.**

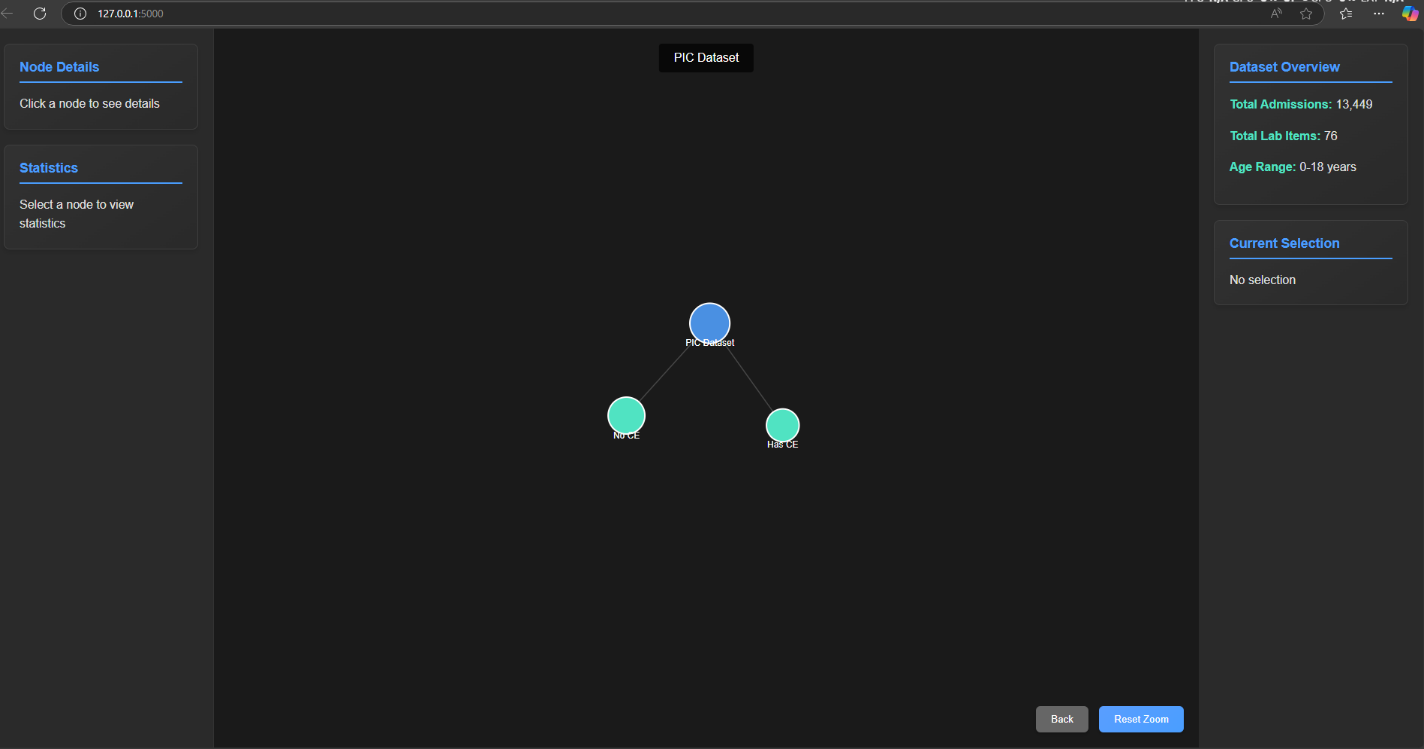
**Features in Action:**

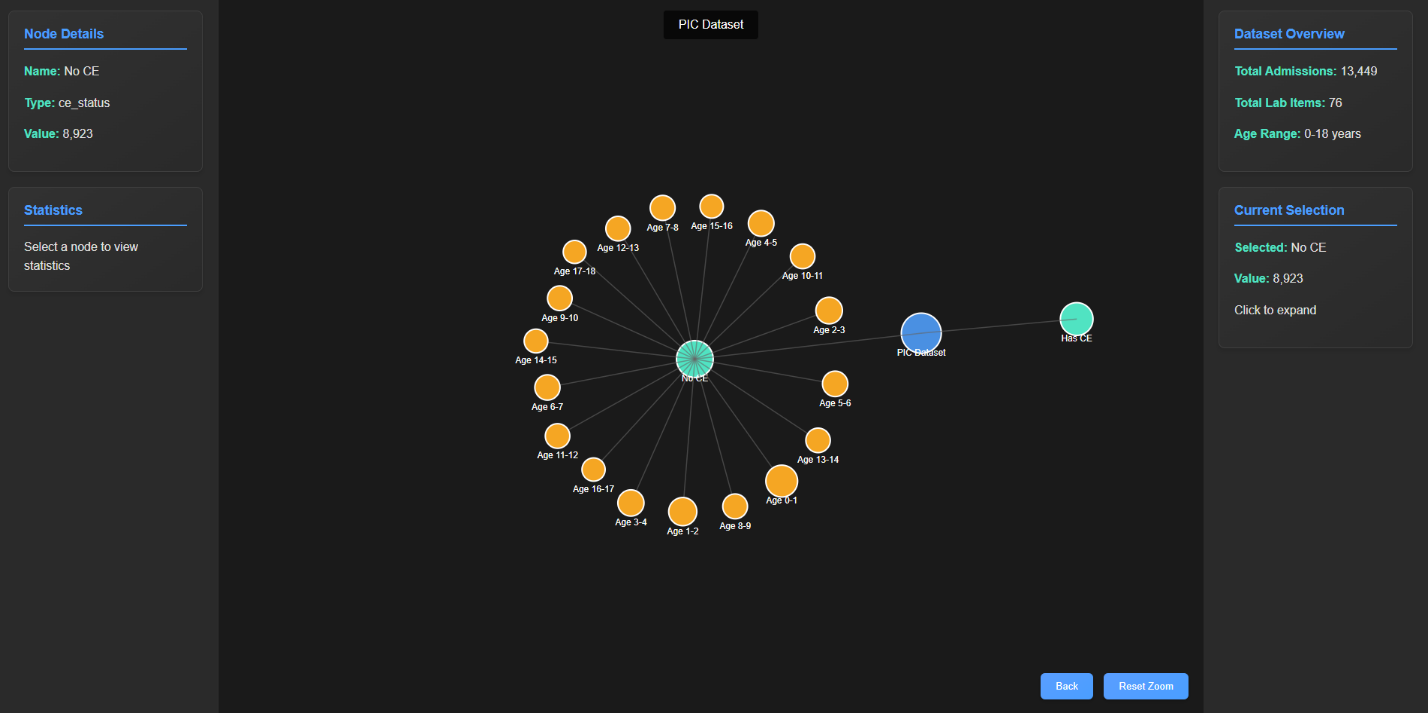
**1. Hierarchical Visualization**

* **Access the visualization dashboard at http://127.0.0.1:3000.**
* **Nodes represent hierarchical categories (e.g., demographic groups, clinical events).**
* **Click on nodes to expand further details or filter data dynamically.**

**Example: Node Expansion**

* **Root node: PIC Dataset**
* **Child nodes: No CE and Has CE**

****

****

**2. Backend API Logs**

**During execution, the backend logs API requests in real-time. For example:**

**127.0.0.1 - - [13/Dec/2024 15:25:36] "GET /expand/ce\_status/hasCE HTTP/1.1" 200 -**

**127.0.0.1 - - [13/Dec/2024 15:25:51] "GET /expand/age\_bin/hasCE\_age11 HTTP/1.1" 200-**

**Troubleshooting**

**Common Issues:**

1. **Missing Dependencies:** 
   * **Re-run the installation commands:**
   * **pip install -r requirements.txt**
   * **npm install**
2. **Port Conflicts:** 
   * **Change the Flask backend or Node.js frontend ports in app.py and package.json, respectively.**

**A screenshot of a computer program

Description automatically generated**

**Contact**

For questions or support, contact:

* **Yesala Likhith Raj**: likhithraj0213@gmail.com