**Project Title**:

PyConnect: An Integrated System for Authentication and Chatbot Interaction

**Description**:

PyConnect is a Python-based project designed to offer a seamless experience in user authentication and interaction through a chatbot interface. By integrating modules for database management, user signup/login, and chatbot functionality, PyConnect aims to provide a robust platform for real-time user engagement

**List of Modules:**

* **Database**
* **Login & Signup.py**
* **ApiCall.py**
* **Alert.py**
* **ChatBot.py**
* **MainFile.py**
* **Database**

**Description:**

The Database module serves as the backbone for storing user information securely.

In this project, We are using localhost database using **xampp** Software to create database in **MYSQL**

**Functions & Usage:**

1.get\_database\_connection():At the beginning of your program or whenever you need to interact with the database, call this function to establish a connection.

2. insert\_data\_into\_table(name, UserName, Password): After a user signs up and provides their information, call this function to insert their data into the database.

3.CheckUserNamme():Before allowing a user to sign up with a username, use this function to check if the username is already taken

4.login(UserName, Password): When a user attempts to log in, use this function to authenticate their credentials.

* **Login & Signup:**

**Description:**

The Signup and Login module is responsible for managing user authentication and registration within the application. It provides functionalities for users to sign up with unique credentials.

* **Submodules of Signup:**
* Fill All Blanks**: It verifies the completeness of user input during signup**
* User Already Exists: **It checks if the chosen username is available**
* Signup Successfully: **Upon successful signup, user data is stored securely**
* Signup Using API: **Use to store user data in database**
* **Submodules of login:**
* Incorrect Password/username: **if user not** fill the correct password or username during user gave for signup then user not login in chatbot
* Login User with API: that allow to user to connect with database
* **ApiCall:**

**Description:**

The ‘ApiCall’ module acts as an interface to connect the project with external APIs. It provides functions to interact with various endpoints, such as Login, signup using Request library.

* **Alerts:**

**Description:**

* In Login.py if user not gave correct password or username then show alert ‘**please enter correct username or password’** otherwise for correct password and username show username on alert
* In signup.py if user not fill inputs then show alert to fill all inputs.
* Elif username already exists then show alert user already exists
* Otherwise then show alert **successful signup and user data is stored securely in database**
* **ChatBot.py:**

**Description:**

The Chatbot module enables the application to interact with users through a conversational interface

* **Submodules of Chabot:**
* JSON File for Data:
* **Description:**

Stores predefined patterns and responses used by the chatbot to understand and respond to user queries.

* Chatbot Implementation:
* **Description:**

Implements the logic for processing user input, identifying intents, and selecting appropriate responses

* **Usage:**

The module reads data from a JSON file containing predefined patterns and responses.

It prompts the user for input and matches the input against the predefined patterns.

If a match is found, it selects a random response associated with the matched intent and displays it to the user.

If no match is found, it provides a default response indicating that it's unsure how to respond to the user's input.

* **MainFile.py:**
* **Description:** this is main file to run the whole project. To run the project use following commands

**Go to terminal and write main.py**

After doing this the project run successfully

**Thanks**