**Assignment 4 :** **Write an assignment to write a methodology using Test Driven ,Behaviour Driven and Feature driven model to create your own chat application ,somewhat similar to what’s up app.**

**Developing a Chat Application using TDD, BDD, and FDD**

This assignment outlines the development of a chat application similar to WhatsApp, incorporating elements of Test-Driven Development (TDD), Behaviour-Driven Development (BDD), and Feature-Driven Development (FDD).

**Project Goal:** Develop a basic chat application that allows users to register, create chat rooms, send messages, and view chat history.

**Development Methodology:**

This project will employ a hybrid approach combining elements of TDD, BDD, and FDD:

1. **Feature-Driven Development (FDD):**
   * **Identify Features:**
     + User registration and login
     + Creating chat rooms (public and private)
     + Sending and receiving text messages
     + Viewing chat history
   * **Plan by Feature:**
     + For each feature, decompose it into smaller, testable functionalities.
     + Break down functionalities into user stories representing user actions.
2. **Behaviour-Driven Development (BDD):**
   * **Develop User Stories:**
     + Define user stories for each functionality using the Gherkin syntax (Given, When, Then).
     + Example:
       - **Feature:** User Registration
       - **User Story:** As a user, I want to register with a username and password so I can access the chat application. (Given I am a new user, When I enter a username and password, Then I should be able to register and access the chat application.)
3. **Test-Driven Development (TDD):**
   * **Write Unit Tests:**
     + Before implementing functionalities, write unit tests that define the expected behaviour based on user stories.
     + Focus on testing individual components and functions.

**Development Process:**

1. **Iteration Planning:**
   * Select a small set of features for each development iteration.
   * Break down features into user stories and write acceptance criteria (BDD).
2. **Development:**

Write unit tests (TDD) for each user story before implementing the functionality.

* + Develop the code ensuring it passes the written unit tests.

1. **Integration Testing:**
   * Test the integration of different functionalities to ensure they work together as expected.
2. **User Acceptance Testing (UAT):**
   * Demonstrate the developed features to stakeholders and gather feedback.
   * Refine the application based on feedback.
3. **Iteration Review:**
   * Evaluate the progress and identify areas for improvement.
   * Plan for the next iteration based on remaining features and learnings.

**Deliverables:**

* A functional chat application with user registration, chat room creation, messaging, and chat history viewing functionalities.
* Unit tests covering core functionalities.
* User stories and acceptance criteria documents.

**Tools and Technologies:**

* Choose a preferred programming language and framework suitable for chat application development (e.g., Python with Django, JavaScript with Node.js).
* Utilize a testing framework (e.g., JUnit, Jest) to write unit tests.
* Consider using a version control system (e.g., Git) for code management.

**Evaluation Criteria:**

* The application should be functional, meeting the user stories and acceptance criteria.
* Unit tests should have high code coverage.
* The code should be well-structured, documented, and maintainable.
* Effective use of TDD, BDD, and FDD principles will be considered.

**Additional Notes:**

* This is a high-level overview.
* Research specific libraries or frameworks for functionalities like user authentication, chat message handling, and data storage.
* Consider implementing security measures like user authentication and data encryption.
* The focus is on core functionalities. Additional features like file sharing or group chats can be explored for further development.

By following these guidelines and combining TDD, BDD, and FDD principles, you can develop a robust and well-tested chat application with a clear development roadmap.