BIAT TEST MANAGER

Project Description :

BIAT Test Manager is an intelligent test management and automation platform that leverages artificial intelligence to revolutionize the software quality assurance lifecycle in banking environments. Inspired by industry leaders like Focus & Thunders , this platform combines comprehensive test management with AI-powered test generation and live execution capabilities.

Project Problematic :

Banking software testing is complex, time-consuming, and highly constrained by regulatory and quality requirements. Existing test management and automation tools rely heavily on manual effort and rigid scripts, making them slow to adapt to changing specifications and user interfaces. There is a need for an intelligent testing platform that assists testers through AI-driven test generation and flexible execution while preserving human control and compliance.

list of functionalities :

- **Core Platform:**
  - Design and implement the foundational platform that supports the entire BIAT Test Manager system. This includes project and test management structures, user authentication and roles, core data models (requirements, test suites, scenarios, test cases, and execution sessions)
- **Requirement Ingestion & Test Generation:**
  - Design and implement a unified module that ingests and normalizes software requirements from multiple sources, including uploaded files (CSV...) and Jira issues. Based on the ingested requirements, the platform generates test suites, test scenarios with AI-driven generation for advanced coverage. All generated tests must be fully traceable to their source requirements and remain reviewable, editable, and approvable by human testers before execution, ensuring control, reliability, and compliance in a banking environment**.**
- **Execution Engine & Streaming:**
  - Design and implement an automated execution engine that transforms approved test cases into executable scripts using browser automation tools such as Playwright or Selenium. The engine must support parallel execution, real-time streaming of execution states, logs, and screenshots, and reliable storage of execution results to ensure transparency and efficient debugging.

- **Live Control & Stabilization:**
    - Implement live execution control capabilities that allow testers to pause, resume, or stop automated test runs and take manual control of the browser when necessary. This feature focuses on stabilizing executions, handling unexpected UI behavior, and ensuring a smooth end-to-end demonstration of the platform's human–AI collaboration capabilities in real testing scenarios.
- **Test Results Dashboard & CI/CD Export:**
    - Create a dashboard to display automated test results (pass/fail, execution time, logs, screenshots) and support exporting runnable Playwright/Selenium test artifacts so they can be executed in a Jenkins CI/CD pipeline and included as build artifacts (if possible).