





TestPilot

Problem Statement

Manual functional testing is:

-  Time-consuming and repetitive
-  Dependent on human interpretation
-  Prone to missed edge cases and errors
-  A bottleneck in fast-paced delivery pipelines

Functional testers often spend hours converting specifications into test cases and executing them manually.

Solution: TestPilot

TestPilot is an AI-powered system that:

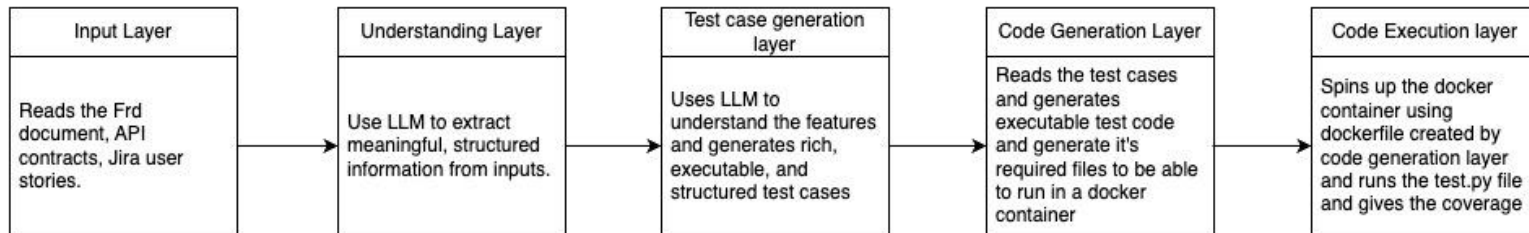
- Reads product requirements (FRDs, user stories, API contracts)
- Understands features and expected behaviors using LLMs
- Generates structured test cases (positive, negative, edge)
- Writes executable test scripts (Selenium, API tests)
- Runs tests, validates results, and reports coverage

→ **A complete replacement for the functional testing role.**

System Architecture

TestPilot

The **TestPilot** is an AI-powered system designed to automate the role of a functional tester by transforming application requirements into executable test assets and generates bug free code which is executable in a docker container. It eliminates the need for manual test case writing and aims to augment or replace traditional QA testers by leveraging large language models (LLMs) and intelligent automation.



Key Modules

- **Input Parser:** Accepts FRDs, user stories, and API specs
- **LLM Understanding:** Extracts feature-level understanding
- **Test Generator:** Builds structured, traceable test cases
- **Code Generator:** Converts cases into runnable test scripts
- **Executor:** Runs tests and reports coverage via Docker

Whats working in the system

- Till Code generation the system would give the correctly
- The code execution may not happen because of the test data

Future Scope

- Addition of user stories which are exposed via jira api's
-
- This can be added as an **extension** in **developer's IDE**
- Addition of other contexts which might be stored in some DB's can be expose via **MCP servers** and can add different contexts
-