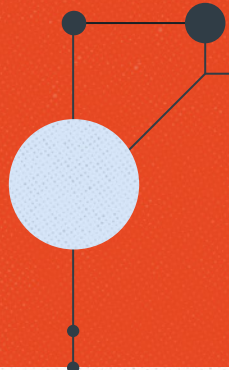


10 steps for getting started building agentic testing agents

Power up your testing with Agent Builder in Studio, the flexible, customizable toolkit for building AI agents tailored specifically to your testing needs. Dive into the world of agentic testing, where these agents work alongside you to deliver unprecedented efficiency, coverage, and quality:



1. Start with a well-defined testing scenario

- Identify existing or new testing scenarios with repetitive tasks suitable for automation
- Create flowcharts to visualize the testing process
- Focus on testing scenarios that can benefit most from AI agents

2. Design focused testing agents

- Create agents with specific, narrow testing tasks (e.g., data retrieval, compliance checker)
- Ensure each agent has a single, well-defined purpose in the testing lifecycle
- Design agents to complement your existing test automation framework

3. Incorporate human-in-the-loop for testing

- Implement human review and validation stages in your agent-driven testing workflow
- Allow for manual intervention in complex test scenarios or when dealing with edge cases
- Use human expertise to validate and refine agent-generated test cases

4. Establish clear testing goals and metrics

- Define specific objectives for each testing agent (e.g., test coverage, defect detection rate)
- Align agent performance metrics with overall testing goals
- Set up measurable success criteria for your AI agents

5. Comprehensive agent evaluation for testing

- Test agents using diverse datasets covering various test scenarios and edge cases
- Include both positive and negative test cases in your evaluation sets
- Continuously expand your test datasets to improve agent performance

6. Determine realistic testing conditions for agents

- Debug agents within actual testing workflows to ensure real-world applicability
- Validate agent performance under various system loads and data conditions
- Simulate different testing environments to ensure agent adaptability

7. Leverage context effectively in testing agents

- Provide agents with necessary context about the application under test
- Use clear, concise descriptions of testing objectives for each agent
- Reduce prompt complexity by leveraging pre-existing test documentation

8. Utilize well-defined tools for testing agents

- Keep testing tools focused and clearly described (e.g., specific test case generators, data validators)
- Integrate agents with your existing testing tools and frameworks
- Ensure agents can interact effectively with your test management systems

9. Continuously monitor and analyze testing agents

- Regularly review agent performance logs and test execution traces
- Analyze patterns in agent-generated test cases and results
- Add problematic test scenarios to your evaluation sets for ongoing improvement

10. Iteratively refine testing agents

- Continuously refine agents, prompts, and test workflows based on results
- Maintain version control for your agents to track improvements over time
- Regularly update agents to adapt to changes in the software under test

Ready to unlock flexibility for your testing with **customizable agents**?

Get started building your testing agents today.

UiPath Test Cloud