API documentation testing with AI user simulation

Lisa Driukova
Senior Technical Writer
JetBrains

AI in API docs: current use cases

- Improved search

- Intelligent code snippets

- Chatbots within documentation

- Documentation summarization
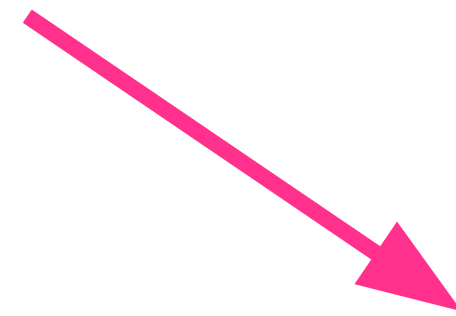
- Automated changelogs

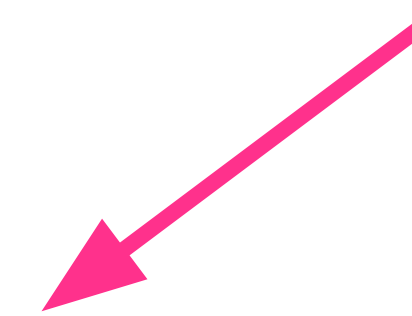**Humans** need to give information to the AI.

Barriers to feedback

Time constraints

- Stakeholders: *LGTM*
- Users: *Need a solution. Now.*

Expertise

- Experienced users: *Details are unnecessary.*
- Beginners: *My bad.*

Multiplied in an open source ecosystem

AI-driven user simulation as a solution
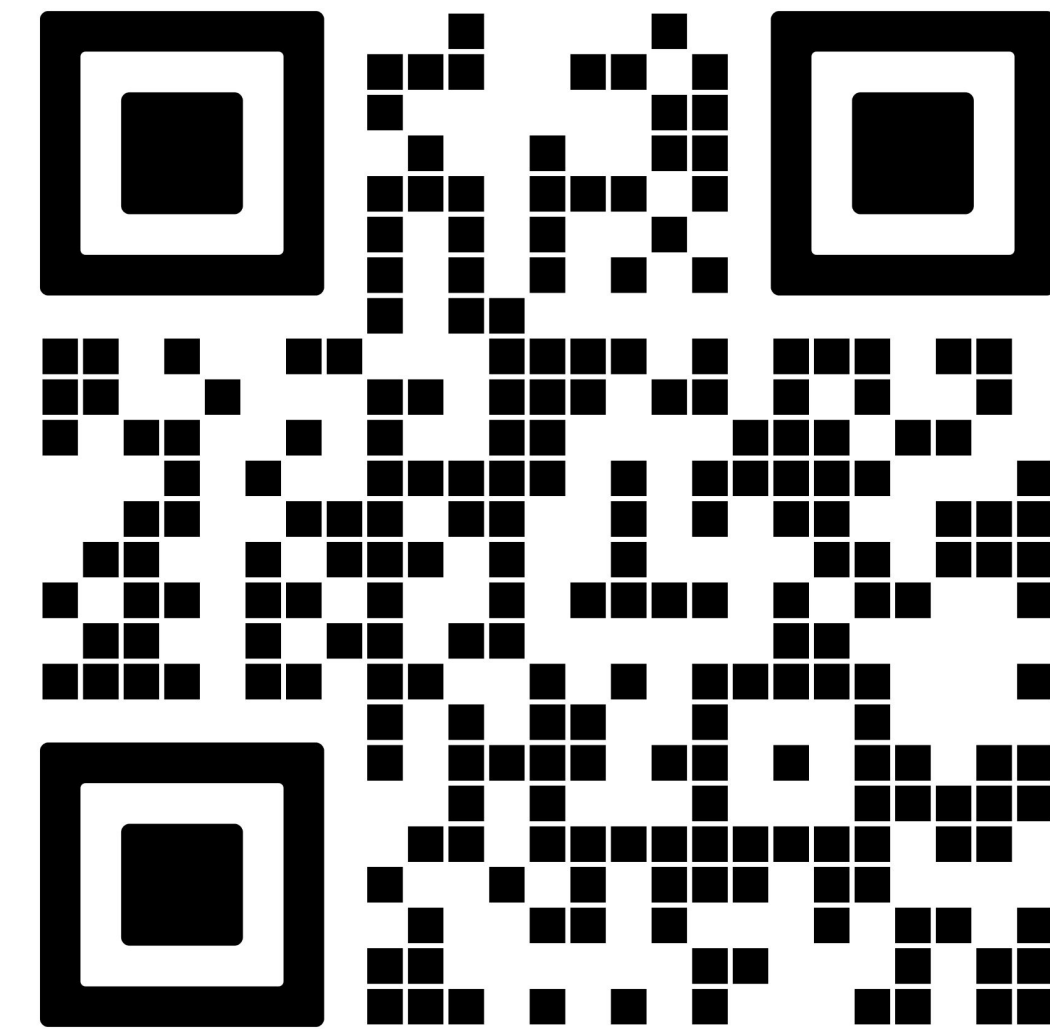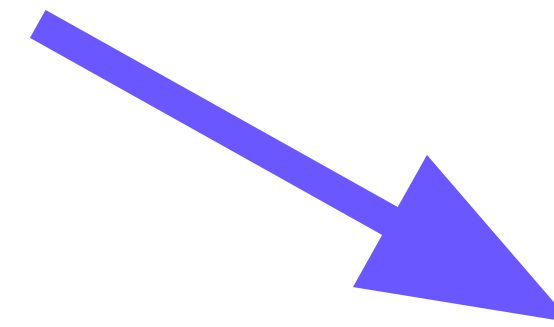
What?

- Test documentation for gaps from multiple perspectives.
- Identify edge cases and unanswered questions without relying solely on human reviewers.
- Reduce the cognitive load on writers and stakeholders.

Why?

- Role-specific context.
- Scenario-based questions.
- Reduced bias.

Workflow

1. Input: OpenAPI specification
2. Extraction of information
3. Question generation
4. Gap identification
5. Output: List of questions

Try it out!

Prompt anatomy

- **Role:** Act as a developer who uses the API documentation to understand how to implement, interact with, and integrate the API into their own applications or systems.

- **Objective:** Analyze the following OpenAPI 3.0 specification and make a list of potential questions a developer might have. Check these questions against the provided specification again to ensure that they are not already answered.

- **Restrictions or limitations:** Concentrate on practical questions, not impossible or philosophical ones.

- **Output requirements:** Provide as output only a list of questions that are not answered within the specification, considering aspects such as usage, authentication, error handling, and edge cases.

Swagger Petstore gaps

- *Is it possible to delete an entity that doesn't exist?*

- How do you handle duplicate pets (e.g. same name and attributes)?

- Can the ID's of deleted pets be reused?

- Is there a way to bulk add multiple pets in a single request?

- What happens if one of the provided tags does not exist in the system?

+ Missing error codes and error messages

Customization

- Different user personas

- Scenario-based testing

- LLM choice

- ...

Key takeaways

- Documentation quality remains a cornerstone of successful adoption and usability.

- AI can help, but humans must lead.

- The user simulation technique is the most useful for testing documentation.

- This PoC is only a starting point. It's your turn to improve and customize it.

Thank you!

Lisa Driukova

driukova.com
linkedin.com/in/driukova