



What I Learned Writing API Documentation with AI

Robert Delwood



About Me

Robert Delwood

API documentation writer

Developer

Programmer-writer

12 years API/developer documentation writer

12 years as a titled developer

NASA, Microsoft, Walmart, American Express



Disclaimers and caveats

My experiences.

Not the best or only ways.

Each company, product, process, and team is different, so the solutions will be different.

API Documentation Writing vs. Technical Writing

Reliance on development experience.

Not necessarily coding or code snippets.

Thinking like a developer.

Field based not article based.

Less descriptive writing. Not article based.

Covers fields across more pages.

Shorter sentences.

Writing for skimming

OpenAPI

```
deliveryDetails:
  type: string
  format: uuid
  maxLength: 36
  minLength: 36
  nullable: false
  title: Customer ID
  description: Unique identifier for the customer.
  default: 00000000-0000-0000-0000-000000000000
  examples:
    - 550e8400-e29b-41d4-a716-446655440000
  deprecated: false
  readOnly: true
  writeOnly: false
```

AI Uses

Tools

Tooling

Tasks

Tools

Tools

Tooling

Tooling

Tasks

These are actions, individual or collective.

Spellcheck, grammar check.

Linting.

Create OpenAPI from code.

Add endpoint groups.

Check for drift.

Write articles.

Article Writing...

Great for first drafts.

Start with a source.

- Common knowledge.

- Specialized source.

 - GitHub ReadMe.

 - Jira ticket.

 - Confluence spec.

 - Source code.

Ask AI to ask you questions.

Write procedures.

Explain OAuth to me.

Now, explain OAuth to my clients.

Include create and refresh.

Inline create and refresh examples.

Write a three-paragraph overview for the /shipping endpoint group.

Now include examples and explanation of the create, update, and list shipment orders.

Add a See Also list at the end of the overview. Use the format like “To create a new shipment, see “ and add the endpoint.

Add links to each of those See Also endpoints using the summary tag of the endpoint as the link anchor.
Add an HTML anchor tag.

...But

Edit carefully.

Review carefully.

Check procedures.

Add your idiomatic voice.

Add formatting and style.

Saves typing but doesn't save time.

I don't want a reputation of being an AI writer.

API Documentation...

API documentation authoring.

This is the largest category.

Add new endpoint groups directly from the code base.

Extremely quick and accurate.

Writing section overviews.

Enum headings with DTO names

Adding examples to fields (example tag).

Combining shared schemas.

Finding and correcting wording changes.

Explain concepts and terminology.

Figure out what's not clear.

API Documentation...

Spec analysis and quality checks.

This is audit work on the OpenAPI source.

Check for drift vs code base.

Check conformity to the specification .

Finding fields/Booleans without a default.

Listing and reordering required fields.

Locating unused schema files.

Running link and lint checker before pushing.

API Documentation...

Git, GitHub and PR workflow.

Creating detailed PRs.

Creating check in process.

Git commands.

Recovering from Git mistakes.

AI writing PR descriptions.

API Documentation...

Postman and API testing.

Write Postman calls and field values.

Analyze results with documentation.

Comment on results and suggest changes to documentation.

Something went wrong...

Error resolving \$ref pointer "http://localhost:9000/build/openapi.json#paths/~1user/post/requestBody/content/application~1vnd.example%2Bjson/schema". Token "application/vnd.example%2Bjson" does not exist.

▼ Stack trace

SyntaxError: Error resolving \$ref pointer "http://localhost:9000/build/openapi.json#paths/~1user/post/requestBody/content/application~1vnd.example%2Bjson/schema". Token "application/vnd.example%2Bjson" does not exist.

```
at Function.syntax (https://cdn.jsdelivr.net/npm/redoc@next/bundles/redoc.standalone.js:55:17310)
at u.resolve (https://cdn.jsdelivr.net/npm/redoc@next/bundles/redoc.standalone.js:61:2114)
at o.resolve (https://cdn.jsdelivr.net/npm/redoc@next/bundles/redoc.standalone.js:55:46044)
at s.resolve (https://cdn.jsdelivr.net/npm/redoc@next/bundles/redoc.standalone.js:87:89888)
at l [https://cdn.jsdelivr.net/npm/redoc@next/bundles/redoc.standalone.js:87:94887]
at https://cdn.jsdelivr.net/npm/redoc@next/bundles/redoc.standalone.js:87:94770
at Array.forEach (<anonymous>)
at s [https://cdn.jsdelivr.net/npm/redoc@next/bundles/redoc.standalone.js:87:84692]
at https://cdn.jsdelivr.net/npm/redoc@next/bundles/redoc.standalone.js:87:94789
at Array.forEach (<anonymous>)
```

Redoc Version: 2.0.0-alpha24

Commit: bc924a

Fix it

API Documentation...

Postman and API testing.

Write Postman calls and field values.

Analyze results with documentation.

Comment on results and suggest changes to documentation.

...But

Verify all changes personally.

Review extensively.

AI can be, and is often, wrong.

Edit carefully.

Check procedures.

Add formatting and style.

Still run each endpoint in Postman

Look for details AI couldn't have caught.

Still run code examples.

AI can be, and is often, wrong.

Saves typing but doesn't save time.

Takes minutes for even simplest of questions.

I don't want a reputation of being an AI writer.

AI (Claude) prone to many mistakes

Use other AI (ChatGPT, and Claude Code) to check.

Vibe Programming...

Vibe programming (often called **vibe coding**) is a style of software development where you use AI to generate much of the code from natural-language instructions, rather than writing the code yourself line by line.

You describe what you want in plain terms. Let the AI write the code, run it, and keep telling the AI what to change.

Very powerful.

Write terminal scripts.

Write JavaScript, C#, and Java projects.

Write Git commands.

Examples

Git. Write the command to remove commit #6a5b1d from the current PR and put it in a new PR.

Write a web base application that opens an OpenAPI file and list all the endpoints. Include the method, like “POST /customer”. After selecting an endpoint, in another textbox, list the project file that code is in.

Write a C# application that takes an OpenAPI file, deconstruct it into a REST project handling each endpoint. Ask me any questions. Add copious inlined comments so I can understand and learn from the code.

...But

Don't think yourself as a developer or programmer.

Will It Replace Us?

Will AI replace writers and API documentation writers?

Will our evil overlords think it's good enough?

AI shouldn't replace the API documentation writer.
It changes what the writer spends time doing.

Prompts

Start small.

Don't have to do everything in one command.

Add details.

Never too many details.

Let AI create prompts.

Automated but overall doesn't save time.

Suggestions

Always with clean commit – can restore if needed. Even push.

In Github, safe to experiment.

Confirm and verify. A lot.

Read other people's prompts. A lot.

What AI is and is Not

Is: Excellent for tools and tooling.

Is: Excellent for first drafts.

Is: Excellent for updates.

Is: Excellent for document drift.

Does: Save me typing and attention.

Is Not: A one click operation.

Does Not: Save me time.

Conclusions

Operations take a long time.

- I browse a lot more than I used to.

- Watch for prompts AI posts to you.

AI saves typing but doesn't save time overall.

Check AI often.

- It makes mistakes.

- It doesn't always understand concepts.

- May not know your perspective.

I feel I'm becoming more of an AI operator than a skilled technician.

I feel I'm losing touch with my programming skills.

I feel I'm losing touch with the product.

Contact Me

Robert Delwood

Website: WriteOnce.org

Email: Robert@WriteOnce.org

LinkedIn: linkedin.com/in/robert-delwood

Medium: robertdelwood.medium.com