This document is deals with the best standards of Swagger documentation

1. **Authentication**

This is the information about using authentication schemes to start consuming your API. Most APIs have authentication schemes, and consumers have to authenticate before gaining access to the API. Make sure this section is properly documented, and hand-holds users to successfully authenticate against the API.

1. **Error Messages**

Error messages are important because they tell end consumers when they're integrating with your API services in an incorrect way. Explain your error standards. detailing all of their possible error codes that consumers may receive.

1. **Avoid Jargon**

Keep in mind that many people working with the API may not have intimate knowledge of the domain or jargon you may be using. Documentation should cater to the “very technical” developer audience, and the less technical decision makers (like Product Managers). A big mistake technical writing teams make is assuming their audience is fully technical and have complete understanding of how to work with APIs. Start your documentation by writing in plain English, and have easy-to-understand domain explanations for every resource. Avoid using too much technical jargon, and write in a way that can be easily understood by people who are new to the API or industry**.**

1. **Treat Requests and Responses**

Describe request-response cycles in detail. Document API with context around the parameters and responses. Responses are the guides for your consumers, indicating whether they’re on the right track, or providing guidance with error messages to help them succeed. Your API users should know exactly what to expect from a successful API call.  Describe the full sample response body in every supported format. Think of not only the format, like XML or JSON, but also of HTTP headers, error codes, and messages.

1. **Give Examples of Responses**

A well- designed API also has an example of the kind of response one can expect on successful call against a URL. This example response should be simple, plain, and quick to comprehend

1. **Keep It UpToDate**

It’s very important and often causes mix-ups! If you make any changes to your API, don’t forget to implement them into your documentation as well;

1. **Be Consistent**

Maintain consistency throughout API documentation to avoid unnecessary confusions;

**Appendix**

1. [Best Practices in API Design | Swagger](https://swagger.io/blog/api-design/api-design-best-practices/)