

# A Friendly Introduction to Software Documentation

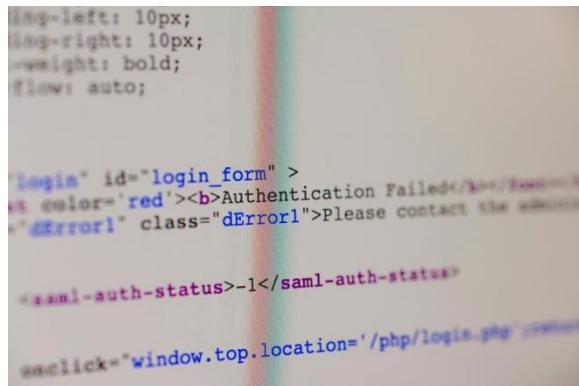
---

**Recap and Concluding Thoughts**

By Arist Bravo

# What is Documentation?

- Any item which conveys information about software
- 2 types: technical docs (for SWEs) and user manuals (for users)
- Our focus: technical docs
- EX of technical docs:



```
margin-left: 10px;
margin-right: 10px;
font-weight: bold;
font-flow: auto;



Authentication Failed!



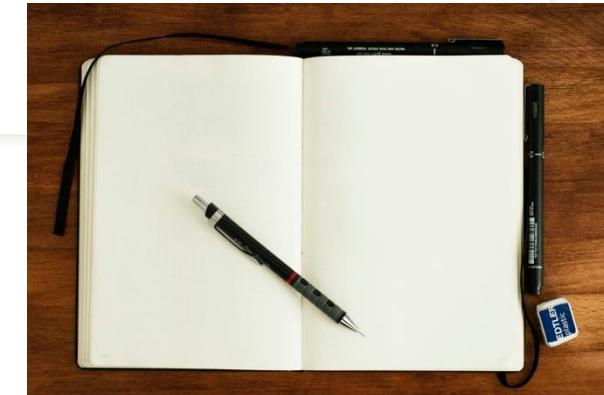
Please contact the administrator.


</saml-auth-status>-1</saml-auth-status>
window.top.location='/php/login.php'
```

Source Code

```
// Output the tickets
System.out.println(ticket_string);
```

Comments and  
READMEs



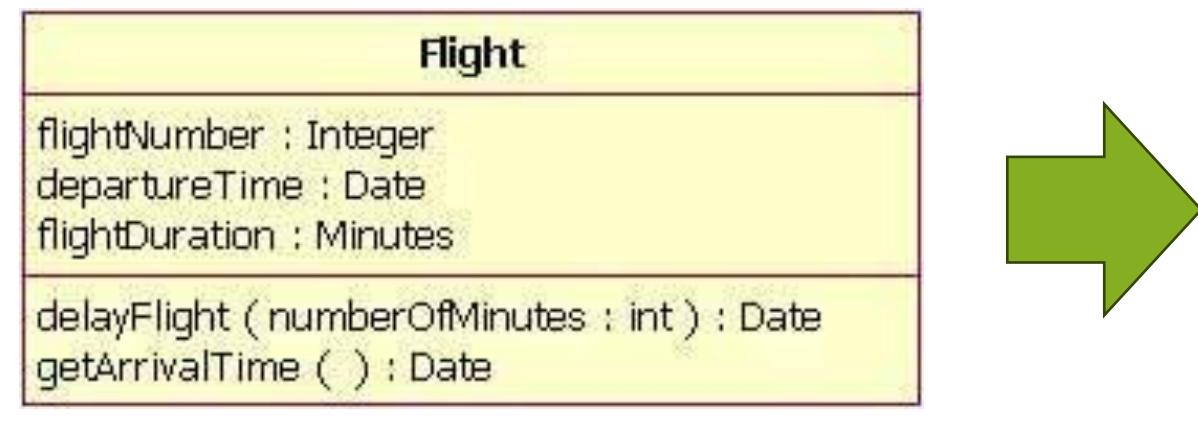
List of Specifications

Flight
flightNumber : Integer
departureTime : Date
flightDuration : Minutes
delayFlight ( numberOfMinutes : int ) : Date
getArrivalTime ( ) : Date

Diagrams and Code for Classes

# Why is Documentation Important?

- Researchers found:
- Documentation gives key info about the projects you code
- OOP developers look at source code, comments, classes, etc. for help as they program
- 61.5% of surveyed developers saw documentation as moderately important (Plösch et al.)

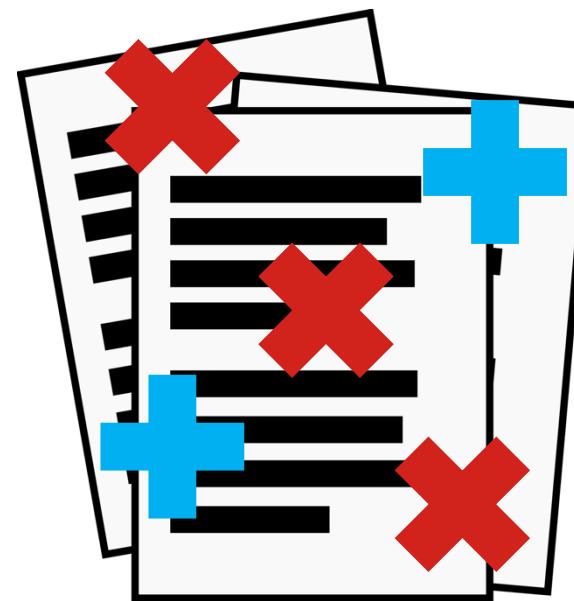


# Defining "Good Documentation"

- There are different approaches or theories which define "good documentation"
- Broadly 2 types of theories:
  - How to *design* docs
  - How to assess docs



VS



# Our Dimension-Based Framework, ACCEU

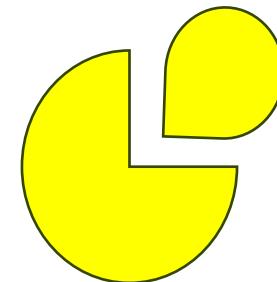
- We choose the dimension-based framework since it is commonly used (see references) and provides rules-of-thumb for "good documentation"
- After all: most documentation is written without standards!
- We study five (unordered) qualities/dimensions based on common use:



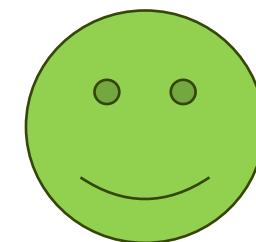
**Accuracy**



**Clarity**



**Completeness**

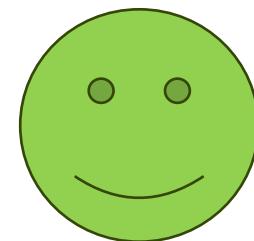
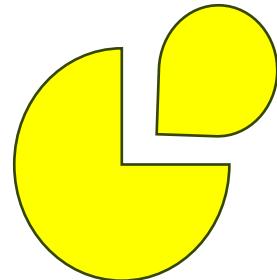


**Ease-of-use**



**Up-to-dateness**

# ACCEU's Components



## Accuracy

True and relevant

## Clarity

Only interpreted one way

## Completeness

Provides all info on how the code works

## Ease-of-use

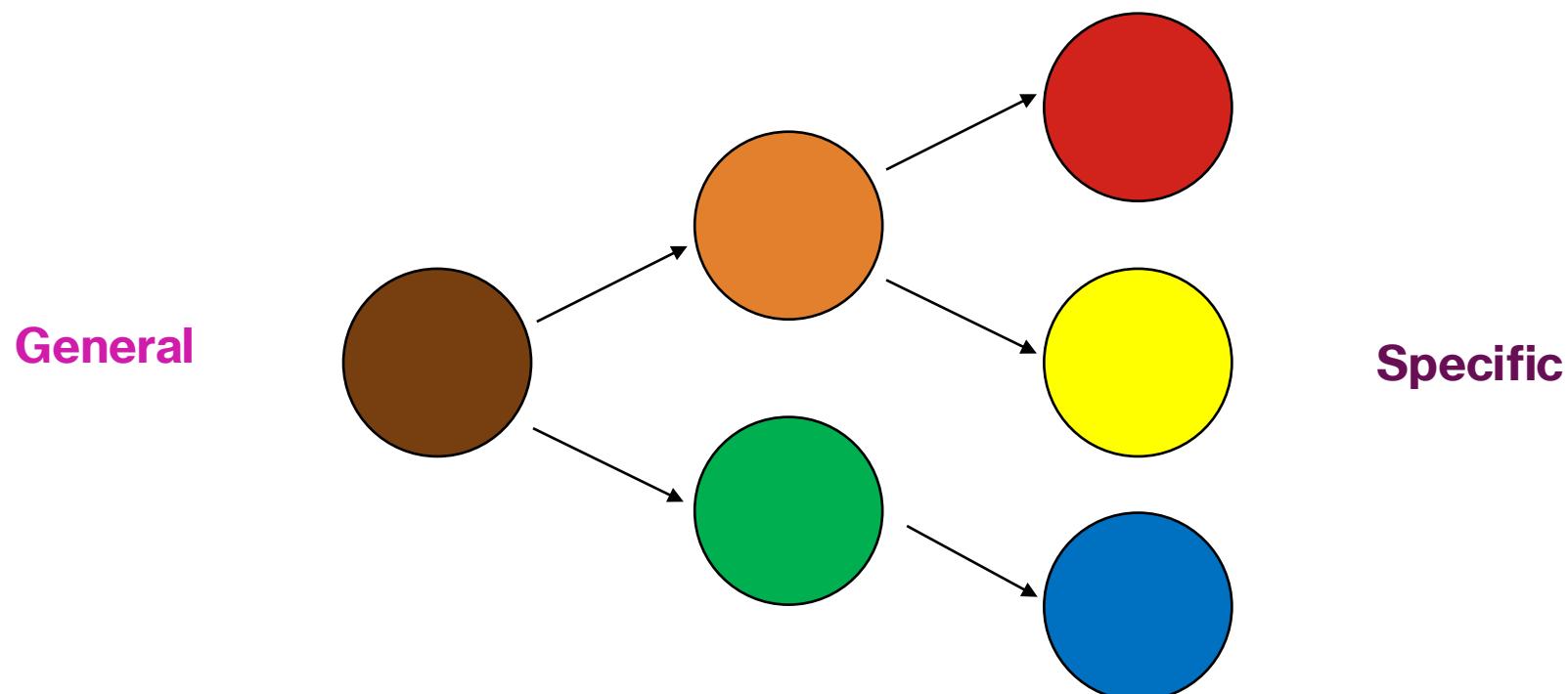
Easily find the info you need

## Up-to-dateness

Docs stay accurate as code is updated

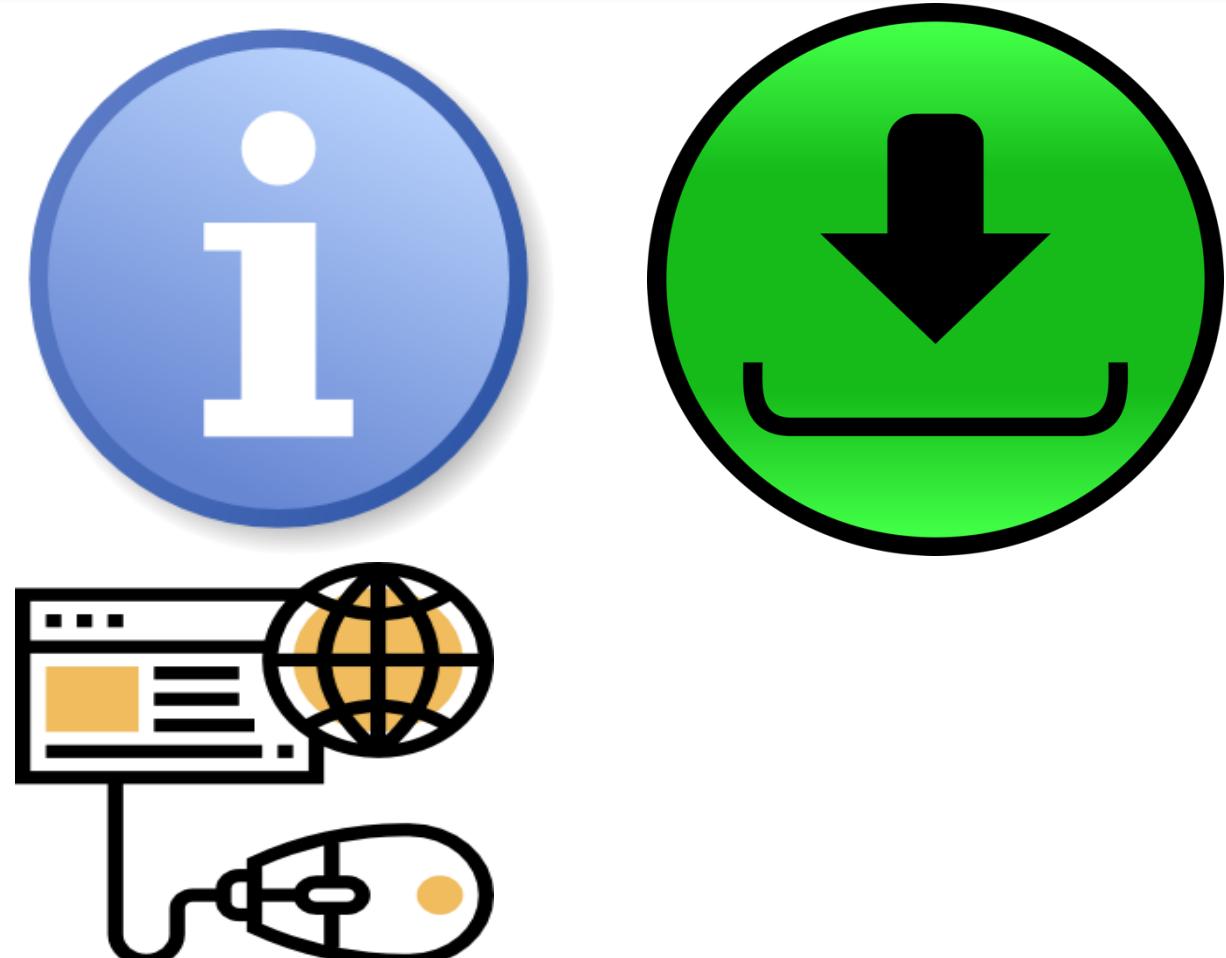
# ACCEU and Higher-Level Views

- Key: high-level documentation links to/references other files (i.e. "graph of files")
- High-level docs must be complete (covering all related items), easy to use (in an obvious place), and consistent (with other files' documentation)



# Possible Sections of a GitHub README

- Title, authors, and short description of the repo or project's purpose
- Table of contents
- Features (what the code can do)
- Installation instructions
- Usage (how to use the project, with examples)
- License (how others can use your code)
- Feedback and Contributions (how others can give feedback or help)



# Final Thoughts

"The faintest ink is more powerful than the strongest memory."

## - Chinese proverb

**Best of luck with your software projects!**

