



Docs, when?

I'm Adam

POLL: DOCS WHEN?

- a) **Never**
- b) **Last**
- c) **Parallel**
- d) **First**

...

Smell test

LOOK MOM! NO DOCS.





- Intuitive
- Discoverable
- Ubiquitous



- Intuitive
- Discoverable
- ~~Ubiquitous~~



NO



YES

“Look Mom! No docs.”

...

LOOK MOM! NO DOCS FOR MY API.



● ~~Intuitive~~

● ~~Discoverable~~

● ~~Ubiquitous~~

DOCS WHEN: PROCESS OF ELIMINATION

- a) ~~Never~~ ← do not want
- b) Last
- c) Parallel
- d) First

...

LOOK MOM! IT'S OK! WE'LL WRITE THEM LAST

Place API here

Documentation

Development



**Internal Silos
Increased Rework
Decreased Usage**

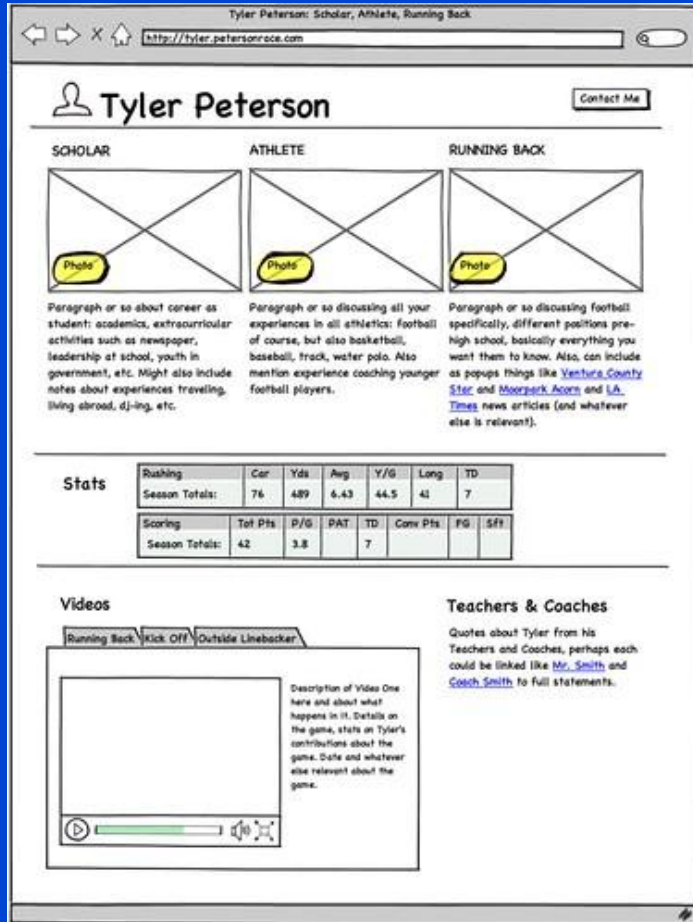
“Docs last can increase team tension.”

DOCS WHEN: PROCESS OF ELIMINATION

- a) ~~Never~~ ← do not want
- b) ~~Last~~ ← do not want
- c) Parallel
- d) First

...

DOCS PARALLEL



Wireframe
before,
parallel or
after?

DOCS WHEN: PROCESS OF ELIMINATION

- a) ~~Never~~ ← do not want
- b) ~~Last~~ ← do not want
- c) ~~Parallel~~ ← do not want
- d) First

...

DOCS WHEN: PROCESS OF ELIMINATION

- a) ~~Never~~ ← do not want
- b) ~~Last~~ ← do not want
- c) ~~Parallel~~ ← do not want
- d) **First** ← winner

DOCS FIRST: GOALS

Better Teamwork

DOCS FIRST: GOALS

Less Rework
Better Teamwork

DOCS FIRST: GOALS

Increase Usage
Less Rework
Better Teamwork

DOCS FIRST: GOALS

Reduce Support/Questions
Increase Usage
Less Rework
Better Teamwork

DOCS FIRST: GOALS

Reduce Time to Usage
Reduce Support/Questions
Increase Usage
Less Rework
Better Teamwork

**“Look Mom! No docs,
still.”**

How to get started

MY PHILOSOPHY

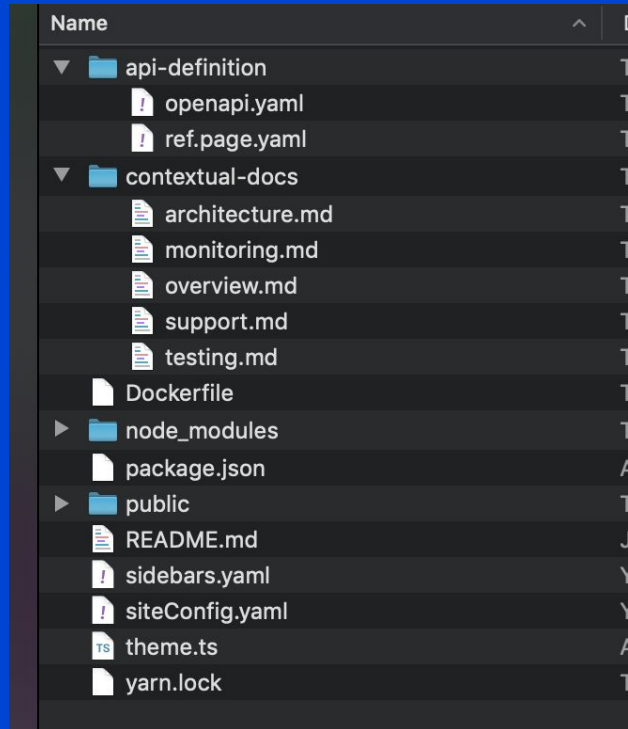
Docs like code

SSOT

Business Problem

What is the docs skeleton?

DOCS SKELETON



**Self-contained portal
enables internal-team
collaboration using
familiar “docs like
code.”**

Get instant feedback on API definition mistakes

Author

Each team authors their own docs and API definitions.

To do that, they will rely on tooling to preview and lint/validate.

Validate

Validate against global standards:

- OAS 3 specification itself
- Specific rules such as grammar/spelling/naming conventions

Commit / Publish

Publishing must be automated. Docs should be built to a preview and verified before publishing in production. From the author's perspective, that's purely:

> **git push**

RUN THE DOCS IN DEV ENV

```
→ Example-Structure cd docs
→ docs yarn install
yarn install v1.16.0

→ docs yarn start
yarn run v1.16.0
```

```
→ docs docker run \
> -it -rm --pid=host \
> redocly \
```

Get instant feedback on API definition mistakes

A walkthrough of the workflow

Questions?

adam@redoc.ly