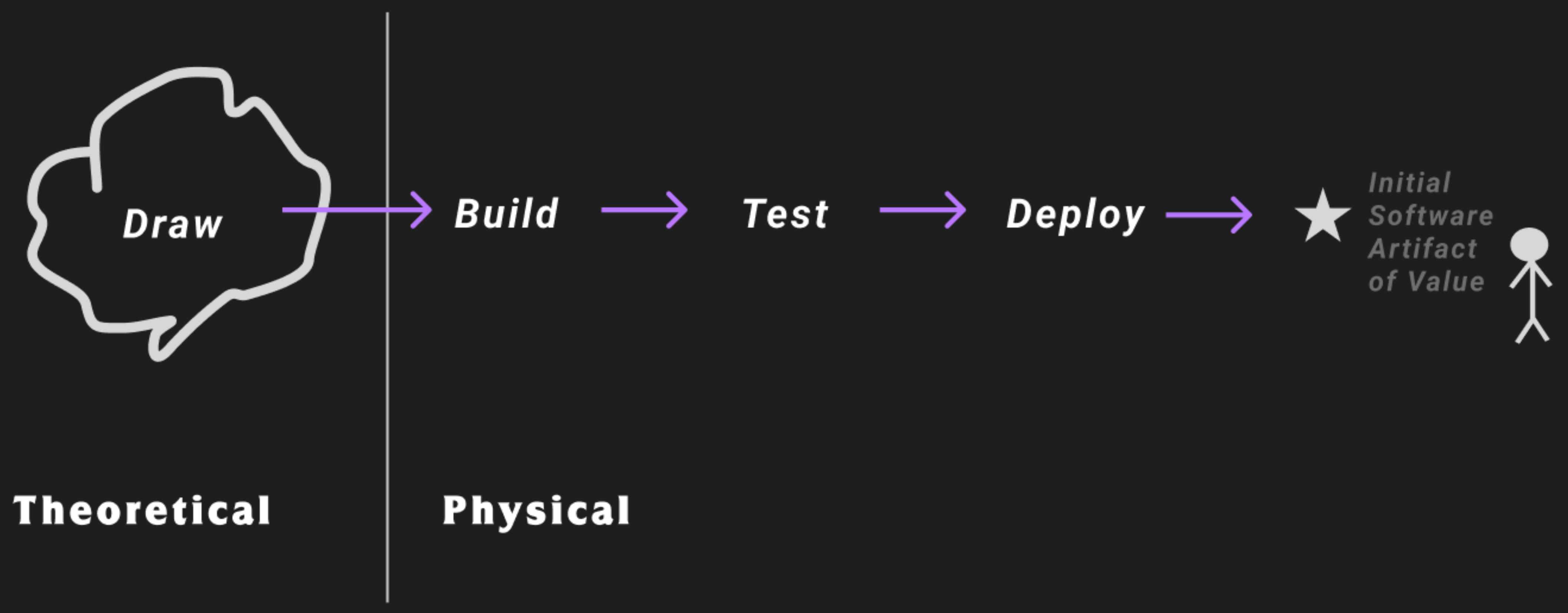
The SOFTWARE ESSENTIALIST

Physical Essential: The Walking Skeleton



Draw → Build → Test → Deploy → ★ Initial Software Artifact of Value

Theoretical

Physical

Physical Essential

The Walking Skeleton

There are a lot of snags and trouble spots that will occur when shipping anything of value.

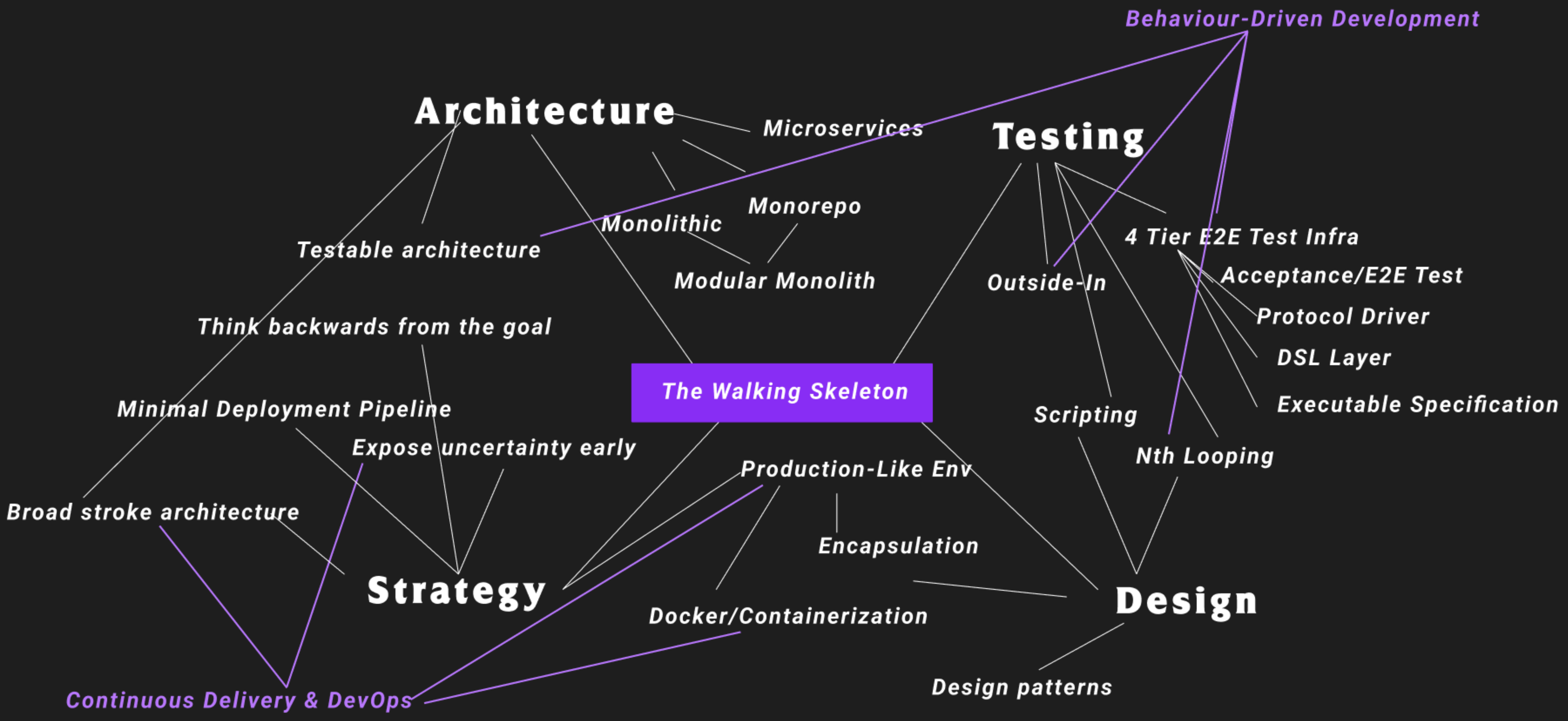
The best time to find those snags and deal with them is at the start of a project.

The Walking Skeleton is about building the smallest slice of useful functionality and deploying it as quickly as possible.

You want to Draw a broad stroke architecture and then Build, Test & Deploy the a tiny slice of E2E functionality.

Expect lots of scripting, problems, and weirdness to occur.

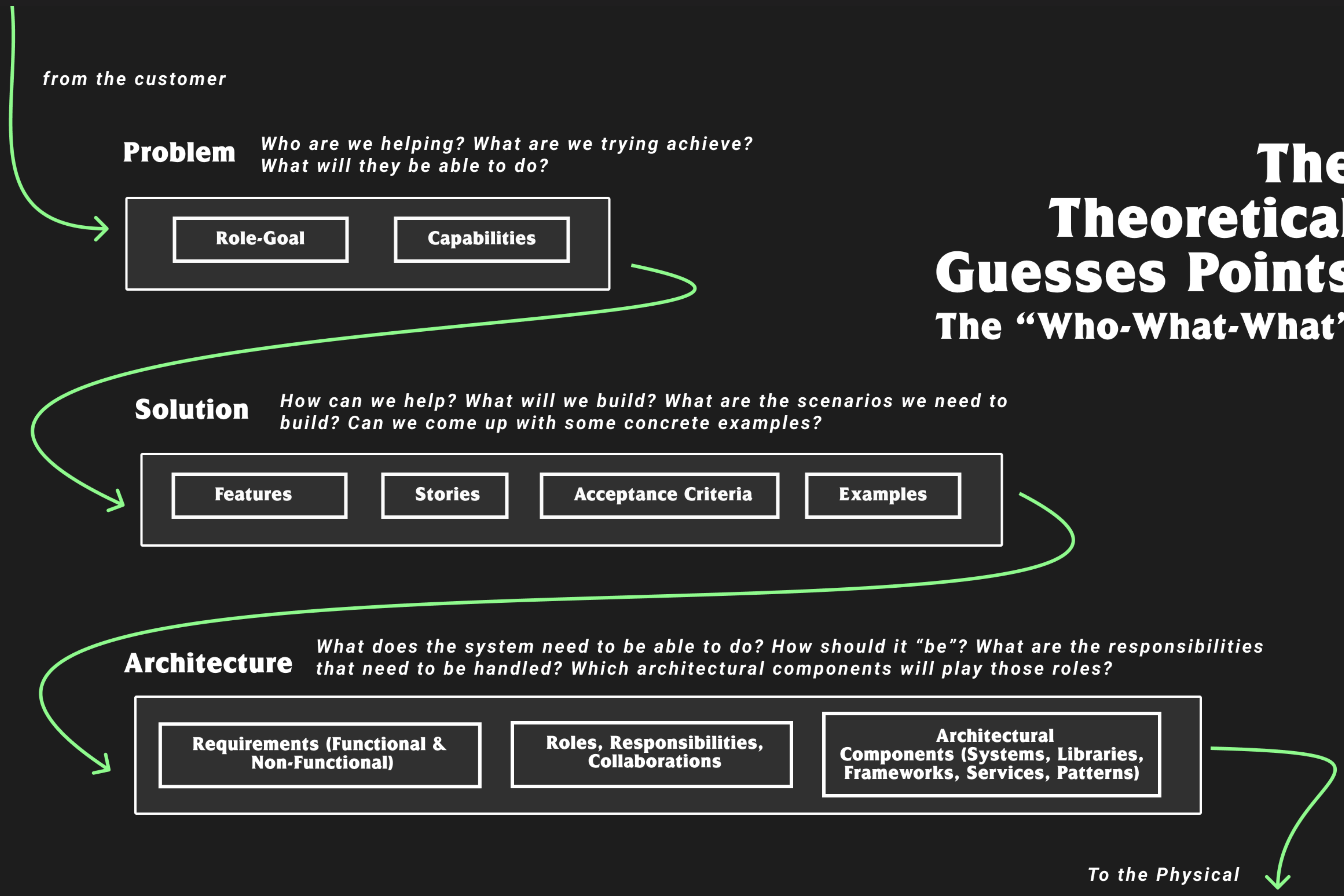
Once this is done, you have a testable architecture and the foundation upon which to write quality code.



essentialist.dev

The Theoretical Guesses Points

The “Who-What-What”



from the customer

Problem *Who are we helping? What are we trying achieve? What will they be able to do?*

- Role-Goal
- Capabilities

Solution *How can we help? What will we build? What are the scenarios we need to build? Can we come up with some concrete examples?*

- Features
- Stories
- Acceptance Criteria
- Examples

Architecture *What does the system need to be able to do? How should it “be”? What are the responsibilities that need to be handled? Which architectural components will play those roles?*

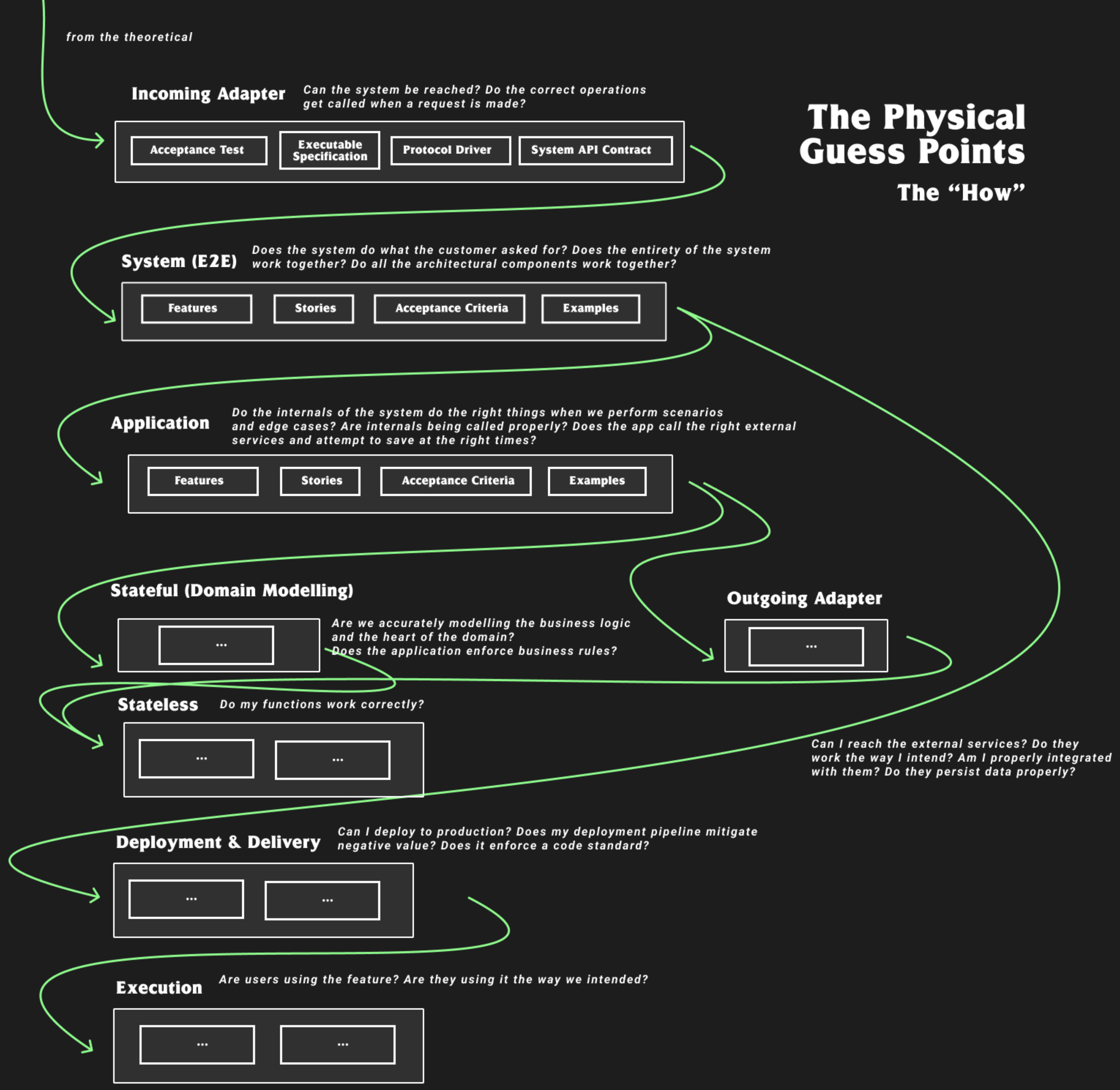
- Requirements (Functional & Non-Functional)
- Roles, Responsibilities, Collaborations
- Architectural Components (Systems, Libraries, Frameworks, Services, Patterns)

To the Physical

essentialist.dev

The Physical Guess Points

The "How"



from the theoretical

Incoming Adapter

Can the system be reached? Do the correct operations get called when a request is made?

- Acceptance Test
- Executable Specification
- Protocol Driver
- System API Contract

System (E2E)

Does the system do what the customer asked for? Does the entirety of the system work together? Do all the architectural components work together?

- Features
- Stories
- Acceptance Criteria
- Examples

Application

Do the internals of the system do the right things when we perform scenarios and edge cases? Are internals being called properly? Does the app call the right external services and attempt to save at the right times?

- Features
- Stories
- Acceptance Criteria
- Examples

Stateful (Domain Modelling)

Are we accurately modelling the business logic and the heart of the domain? Does the application enforce business rules?

- ...

Outgoing Adapter

- ...

Can I reach the external services? Do they work the way I intend? Am I properly integrated with them? Do they persist data properly?

Stateless

Do my functions work correctly?

- ...
- ...

Deployment & Delivery

Can I deploy to production? Does my deployment pipeline mitigate negative value? Does it enforce a code standard?

- ...
- ...

Execution

Are users using the feature? Are they using it the way we intended?

- ...
- ...

essentialist.dev

Where we’ll learn more