



Effort

How extensive is the set of automated unit, API, functional or UI tests that this change requires creating or maintaining?



How much documentation this change requires?



Shifting tests left decreases the effort. Can we comprehensively test the change in any environment (NXI, NXQ, Pre-Prod)?



How laborious is the creation of the test data?



What is this change's impact in sensitive areas that need extensive testing? (eg. pricing)



Effort - Question Bank

Does implementing this change require a large volume of repetitive tasks? (i.e: refactoring)

How many technology layers require changing to implement this change?

Into how many brands do we need to implement this change?

What are the major system capabilities (and how much effort will each require)? /// What are the functional requirements and non-functional requirements and how big are they?

What are the system constraints and assumptions? E.g system must use a certain framework or must integrate into another system in a certain way.

What is required to release this change? Is it a simple release or will it require extensive stakeholder and market engagement?

How much monitoring or alerting needs to be implemented



Complexity

How much change is required in the current architecture to accommodate this change?



How much effort is required to master the technologies, frameworks or architecture used to implement this change?



How much collaboration with people outside the E2E Core team is required to finish this change, including vendors?



What is the level of dependency required to implement this change? (i.e.: part of the system needs to be done for the next to start)



How likely this change will require an assessment from the security team, pen or performance tests?



Complexity - Question Bank

How many integration points this change will required or impacts?

How many technology layers require changing to implement this change?

Another Question

What is the likelihood that this change could impact other application areas?

How likely is this change to depend on the knowledge of a specialist?

What are the odds that this change could turn into a compliance breach?

Another Question

Another Question

Another Question

Another Question



Uncertainty

How unsure are you regarding the desired outcomes of this change and how to measure it?



What is the volume of outstanding decisions and assumptions regarding this implementation?



Do we understand the dependencies that might impede us from starting or finishing the work, including non-technical ones? (i.e: training)



How many spikes do we need to run to understand the codebase and provide an accurate estimation?



How many team movements will we have during the implementation of this change? (i.e.: booked leave; someone new to the team)



Uncertainty - Question Bank

How likely do we require design documents to complete this change? (i.e: UX, architecture, process)

How likely do we need to write a test approach for this implementation?

How uncertain are we regarding the UAT sign-off process?

How unavailable are the stakeholders responsible to sign-off this change?

Another Question

Another Question

Another Question

Another Question

Another Question

Another Question