Playbook Methods Repository

# **Feasibility Prototype**

Build a technical prototype with just enough code to test out new technologies (e.g. algorithms, hardware, specific features, functional areas) in order to address technical feasibility risks, often related to performance, during product discovery.

### Remote Agility: **•** High

### Linked Tactic(s): Prototyping, Technical Validation

## Why we do it:

Reading or talking about a technology only gets you so far. You can learn what a technology *might* or *should* do, but you might have to get hands-on to see what’s actually possible. With a feasibility prototype, we can explore two different questions:

1. Is it possible?
   1. Is it possible to build the thing we want? Does the technology do what we think it does?
   2. An example of this might be, “Is vision AI smart enough to determine pizza toppings from an image?”
2. What risks and trade-offs should we be aware of?
   1. Just because it’s possible to build something, doesn’t mean it will work for our use case.
   2. For example, the AI may be capable, but the computation may be slow or expensive. These may be deal-breakers, or just things to keep in mind as we build.
   3. Some factors to consider are:
      1. Implementation time and complexity
      2. Performance time and resources
      3. Cost
      4. Reliability
      5. Maintenance

## 

## When to apply it:

* Concept Evaluation: During Discovery, use a Feasibility Prototype to assess the technical feasibility and complexity of a concept.
* Implementation: Feasibility Prototypes can also be used during Delivery to explore implementation options, such as for a new feature or a refactor.

## Best Practices & Considerations:

* Have clear goals. This helps with both:
  + Having a Definition of Done to avoid scope creep.
  + If something is taking longer than expected to implement, knowing if you should invest more time because it’s a priority, or just move on because it’s not impactful.
* Consult SMEs for how they would approach the problem and notable risks.

## Responsible roles:

* Software Engineer (Lead): to build the prototype. May also include hardware or ML engineers depending on the technology being explored.
* Product Manager to support by giving product requirements and considerations.
* Product Designer may support by giving input on the user flow and product requirements. However, keep in mind that the goal of a Feasibility Prototype is to assess Technical Feasibility and not Usability or Desirability.

## Tools:

### Online tools/platforms/services

* + xx

### Websites

* + xx

### Databases

* + xx
* Other
  + xx

## 

## Thoughtworks Examples - Linked

### Client working docs, airtable, miro/mural boards

### Client polished presentations/deliverables

* + Bose - Greenwich (exploring React Native vs Native mobile development)
    - [Benchmark report](https://docs.google.com/document/d/10T0_oDFOUU3BSkkk8sAcBJLvvAbdBEI148UoJ_kmvdA/edit)
  + Bose - Tortuga (exploring feasibility of an IOT messaging platform)
    - [Findings report](https://drive.google.com/file/d/1frBlAq97tPUzVv3mynUKexnBpf79WMiH/view)
  + Meta - Wombat WhatsApp (exploring feasibility of a Workplace <> WhatsApp integration)
    - [Readout to WhatsApp team](https://drive.google.com/file/d/1ixLqyAos9JXKJ10mPHSBQK0Vgnb2QoQy/view?usp=sharing)
    - [Demo video](https://drive.google.com/file/d/1rmUzDJHl9whCkzLlOzQ_-0t94PU9_PJg/view?usp=sharing)
    - [Knowledge Library investigation](https://docs.google.com/document/d/1koRu497Dt0T5Xq8gVEQW3jmLmuCe0ele/edit?usp=sharing&ouid=102340424014431027512&rtpof=true&sd=true)

### Internal assets - clinic materials / guild docs

* + Xx

## 

## Learn more: How we do this?

### Templates (docs, decks, sheets, miro, etc.)

* + xx

### How-To Resources (external or internal)

* + xx

### Outside References (articles, books, etc.)

* + xx

### Sub-set Activities

* + xx

## 