Playbook Methods Repository

# **Systems Architecture**

Define the interaction of software components and services using common design patterns to trade off isolation, maintainability, scalability, and extensibility.

### Remote Agility: **•** High

### Linked Tactic(s): Solution Architecture

## Why we do it:

Xx

## 

## When to apply it:

* Xx

## Best Practices & Considerations:

* Xx

## Responsible roles:

* Xx

## Tools:

### Online tools/platforms/services

* + xx

### Websites

* + xx

### Databases

* + xx
* Other
  + xx

## 

## Thoughtworks Examples - Linked

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

* + xx

### Client polished presentations/deliverables

* + xx

### Internal assets - clinic materials / guild docs

* + xx

## 

## Learn more: How we do this?

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

* + [Project Barry Presentation](https://docs.google.com/presentation/d/1YCwvqALCaMV5jaw8IOTz6Ef6DIHMyHpksDKgyPgcZIk/edit#slide=id.p1) (see architecture slides)
  + [Project Truffle - Solution Design](https://docs.google.com/document/d/1p7KA8_g41m5RtsYxR0jYAhs76qmvLcpsl5EogbsocW4/edit#heading=h.kxoj05k820fy)

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

* + xx

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

* + xx

### Sub-set Activities

* + xx

## 