

This is My Test!

ミスターシャチOO

# 自己紹介(1)

- 「草でも食わせておけ！」と言われる県の某動画では、本当の秘境と紹介されているところの出身
- 新卒でDBKのFJのグループ会社に入社  
(紆余曲折)
- OFWのソフトウェアテスト会社に入社  
(紆余曲折)

# 自己紹介(2)

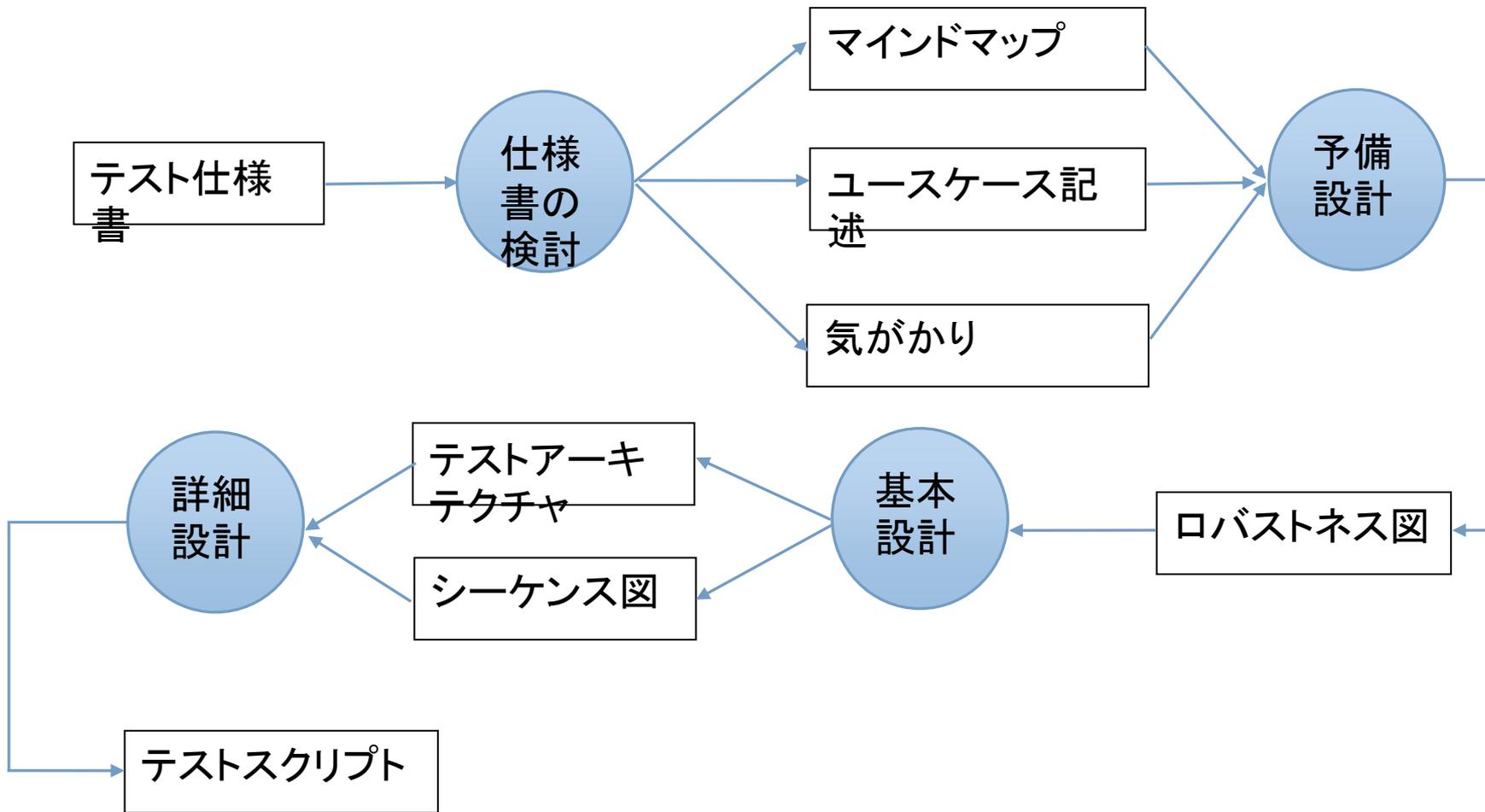
- 秘境での職探しのため・・・
- 職業訓練校で電気や機械についての訓練を受け、高度な分業制について学ぶ  
→しかし見つからず・・・
- DBKなどにてZMの指先として、年末調整やZ金、N金などの業務(窓口など)に従事

実は . . .

# というわけで

- 元(?)テストエンジニアで元(!)給付金の審査をした者が忖度なしでテスト設計してみた件

# PFD



# テストスコープ(WhyとWhat)(1)

- テストベースより  
各システムのテストは実施済み  
テスト重点項目
  - システムが長時間稼動することを保証
  - 時間枠の重複販売が発生しないこと
  - 「密回避」が達成できたことの効果

# テストスコープ(WhyとWhat)(3)

- テストベースより
- 園内およびWebでの予約購入、発券そして入場の流れについて確認

# マインドマップ\_園内チケットシステム

2024.12.8

園内チケットシステム

入場管理機能

残数表示

入場ゲート機能

発券機能

H. Ancho

# How

- Webチケットシステムや園内チケットシステムの要求仕様書の機能の記述
- ユースケース記述にできる
- ユースケースと設計にはギャップがある  
→ICONIXアプローチ

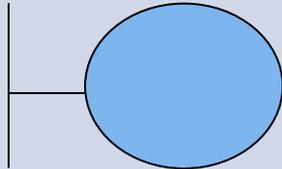
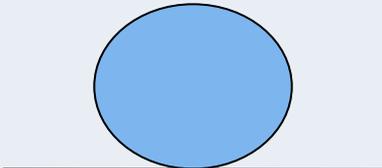
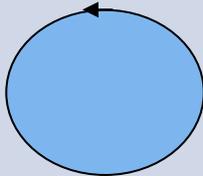
# テストスコープ(WhyとWhat)(2)

- テストベースより
- 追加システム
  - 時間枠の追加
  - Webチケットシステムの追加
    - 「会員」を追加
    - 30日後まで購入可能
  - 時間枠あたりの購入枚数制限を設ける
  - 動物園入口で残数がわかるように残数インジケータを2台追加

# ICONIXアプローチ

- イヴァー・ヤコブソン(UML(ユースケース図)の発案者、スリーアミーゴスの一人)によって考案され、タグ・ローゼンバーグらによって広められた手法
- ロバストネス図などを用いて予備設計を行う

# ロバストネス図(1)

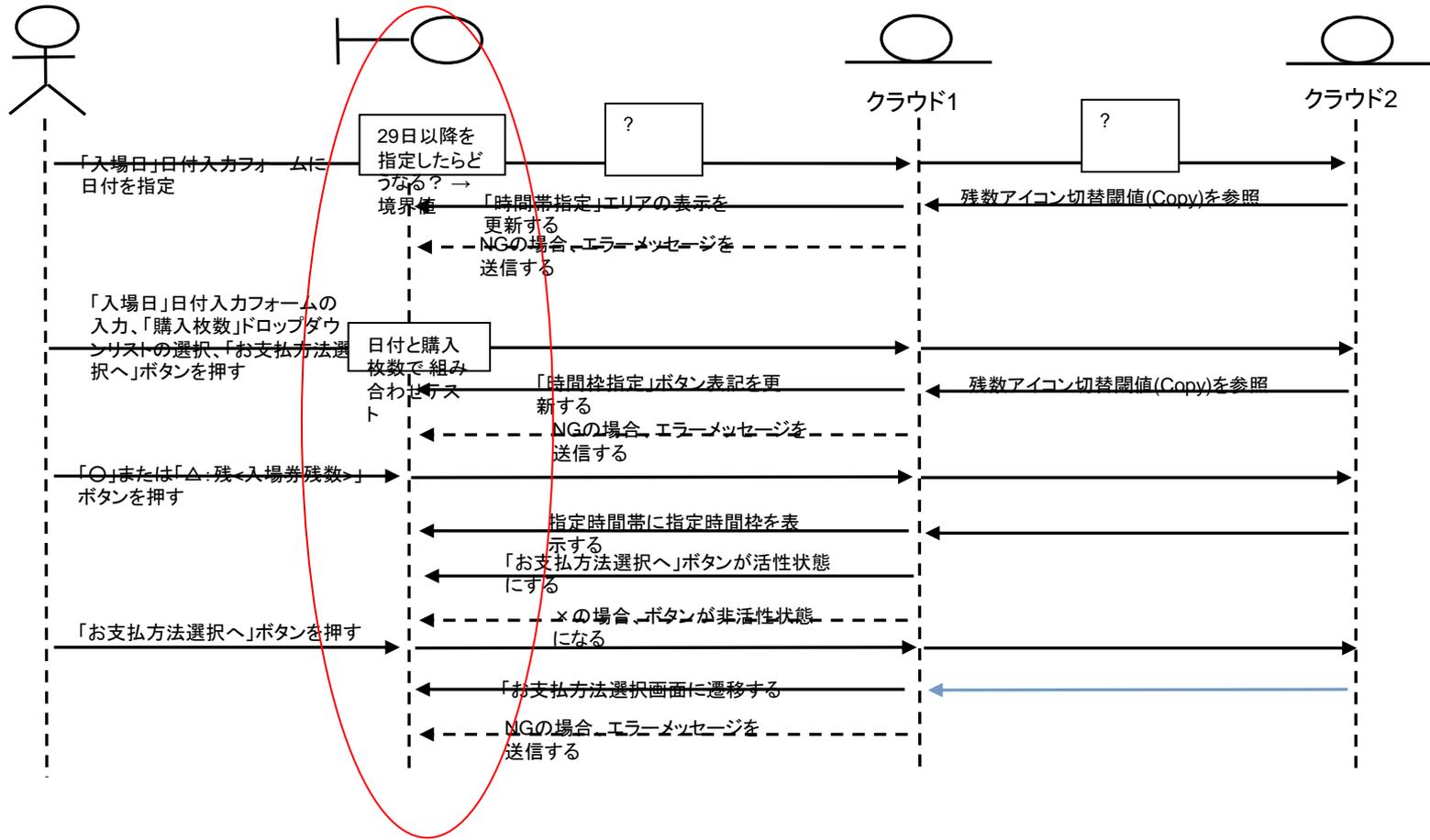
要素	アイコン	概要	コンピュータ5大装置
バウンダリ		アクターとシステム内部の境界にあるオブジェクトタイプ	入出力
エンティティ		ソフトウェアシステム内部で管理するデータを示すオブジェクトタイプ	記憶
コントロール		バウンダリとエンティティを繋ぎ、システムが行う処理を定義するオブジェクトタイプ	制御/演算

# ユースケース記述

項目		説明	
ユースケース名		W-0008 予約購入	
アクター		入場者	
		<p>予約購入画面がデフォルト表示されていること。</p> <p>デフォルト表示は以下の通り。</p> <ul style="list-style-type: none"><li>・「入場日」日付入力フォームは現在の日付になる。</li><li>・「購入枚数」ドロップダウンリストが両方とも0枚選択になる。</li></ul>	



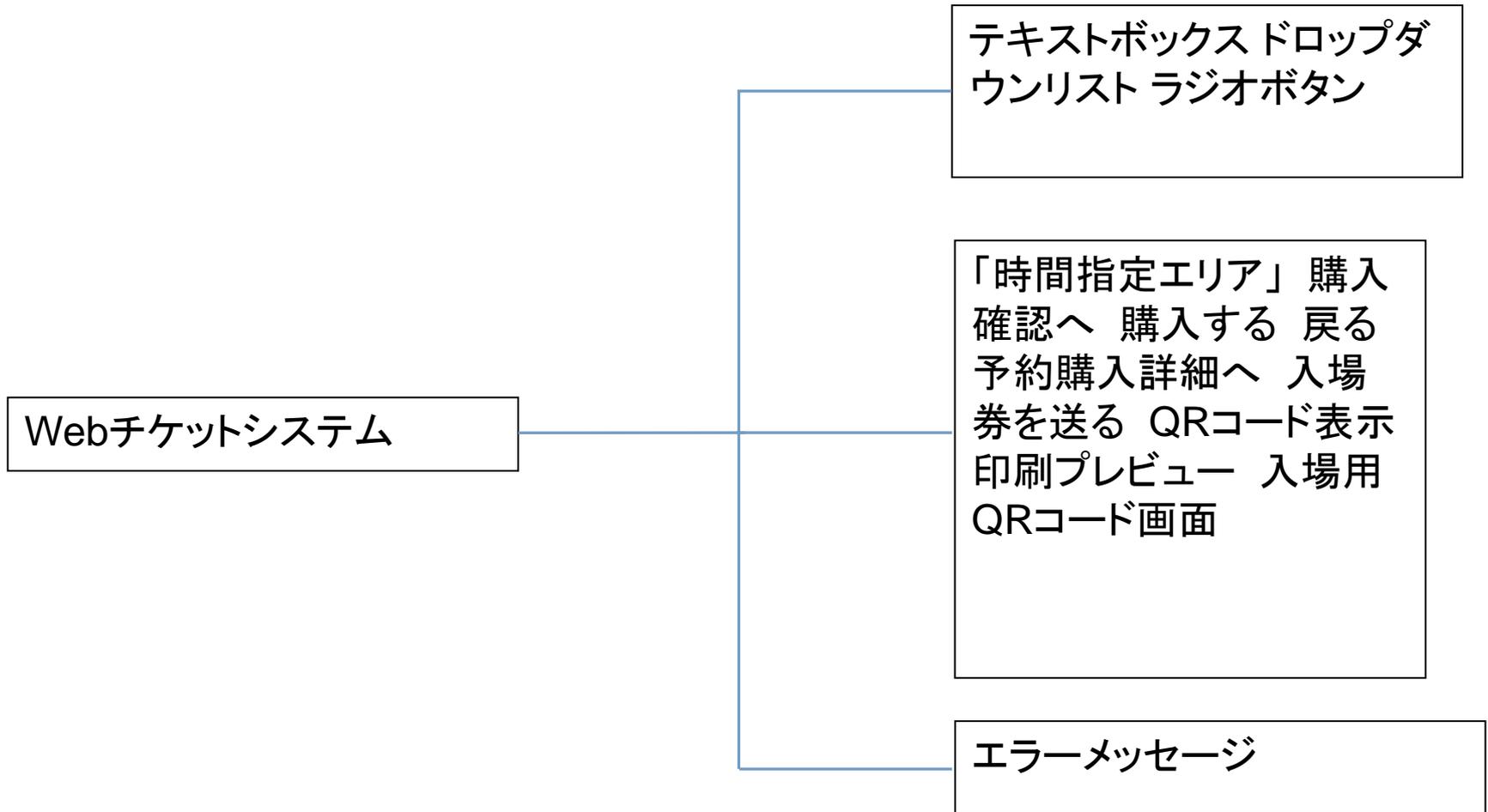
# シーケンス図



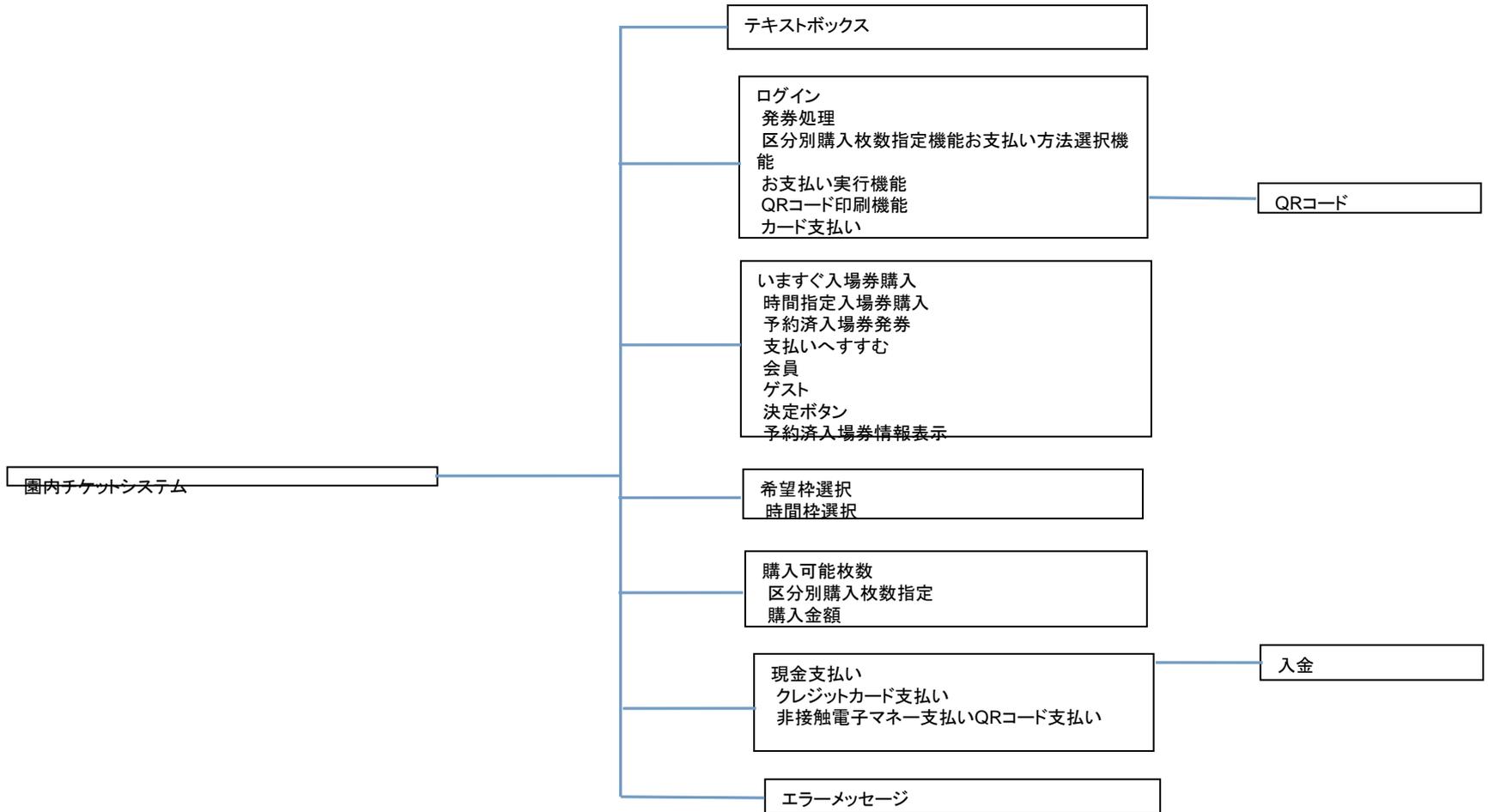
# テストアーキテクチャ(1)

バウンダリ	コントロール	エンティティ
入力	演算	主記憶
出力	制御	キャッシュ
入出力	割り込み	補助記憶

# テストアーキテクチャ\_Webチケットシステム



# テストアーキテクチャ\_園内チケットシステム



# 体験ベースのテスト

- 新紙幣や硬貨は使えないことがある
- メールだ・・・
- システムを止めるな(でも、止まる)

# テスト重点項目

- システムが長時間稼動することを保証  
→1週間ほど終夜通電しておく
- 「密回避」が回避できたことを示すこと  
→G-220などにある入場券の利用実績の集計機能を使えば、回避できたことを示す
- 時間枠の重複販売が発生しないこと  
→ログインする/しないを併用している以上、重複販売は発生する可能性が高い  
→運用でカバー

# 時間枠の重複販売

- シニア層のお客様が孫をつれて来場
- スマホや発券機の使い方に不慣れ
- スマホを忘れてくる場合もある
- →こういう場合も想定して予約購入後にいますぐ購入も可能にしておく必要がある

# まとめ

- 分析と設計の間にギャップがある場合は、ICONIXアプローチを使うとロバストネス図を描くことにより、各機能のシーケンスが見えるようになるので、分析と設計の間のギャップがうまる。

# 参考文献

- オブジェクト指向開発論 海谷治彦
- UMLを描こうVol 1 , Vol3, Vol5, Vol6
- ロバストネス図を活用したシステム設計
- IT導入補助金2024 公募要領
- 令和5年度補正サービス等生産性向上IT導入支援事業費補助金 交付規定

# Future

テストシナリオ層

接合層

機能層