

だんだん動物園入場システム テスト設計のご提案



2026/08/31

1. チーム紹介
2. コンセプト
3. 作業方針
4. テスト設計 作業内容の紹介
 - [STEP1] 要求分析
 - [STEP2] アーキテクチャ設計
 - [STEP3] テスト詳細設計

1. チーム紹介

◎ [Teamわんこそば] の構成について

自社でシステム開発を行っている会社にて

[製品の検査 / 品質担保]を行う部門に所属するメンバーから選抜

→ **前回参加者2名 + 初参加メンバー4名の計6名**

◎ 参加の目的

前回大会の学びを活かし、新規性のあるテスト設計に挑戦する！！

➤ **生成AIを積極的に取り入れ**、メンバー全員がAI活用スキルを高められるよう取り組む

2. コンセプト

生成AIを積極的に活用したテスト設計

これまで社内で活用経験が少なかった生成AIを積極的に活用し、分析及びテスト設計を行う

生成AIによる分析を行うことによって、様々なケースを想定した抜け漏れの少ないテスト設計を行う

⇒「予期せぬパターン」による障害流出リスクを最小限に抑える

3. 作業方針の策定

◆ 目的達成のために必要な行動方針

前述の『目的』を達成するための活動に取り組むにあたり、
テスト設計の内容を後工程（実施）まで考えた際に、
より『**実現可能な内容**』に落とし込むため、下記の「**方針**」を定めて作業を進めることとした。

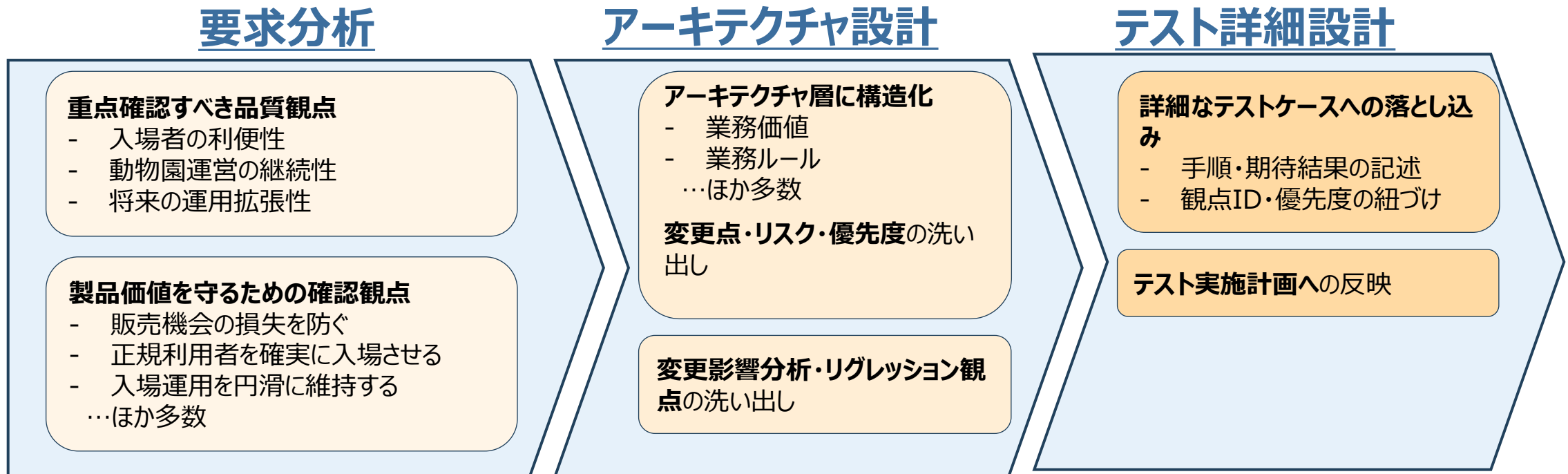
【作業方針】

- ① AIを積極的に活用する **★コンセプトを踏襲**
 - プロンプトの記述をチーム内で統一する
 - AIが作成した資料を必ず理解した上で必要に応じて修正する
- ② 進捗報告では成果物だけでなく、プロンプトも含めて報告
 - プロンプトの記述に問題が無いか随時チェック
 - 想定通りの成果物が作成されない場合チーム全体でプロンプト改善の検討を行う
 - これにより、AI活用のノウハウを蓄積し成果物の水準を一定にする

4. 作業内容の紹介

ここまでの『作業方針』を基に、具体的な作業内容の検討を行い、
下記のようなステップで作業を行うことで、テスト設計を進めた。

◆テスト設計の大まかな流れ



以降のスライドにて、各ステップ内で作成した成果物を参考にしながら説明を行う

要求分析

4. Step1 要求分析_品質保証方針サマリ

◆ 品質保証方針を策定

要求分析

重点確認すべき品質観点

- 入場者の利便性
- 動物園運営の継続性
- 将来の運用拡張性

製品価値を守るための確認観点

- 販売機会の損失を防ぐ
- 正規利用者を確実に入場させる
- 入場運用を円滑に維持する
- …ほか多数

入場者の利便性、動物園運営の継続性、将来の運用拡張性を確保するため、重点的に確認すべき品質観点を整理。

確認軸	保証したい価値	想定される品質リスク	確認方針
販売機会を損なわない	利用者が適切な条件で入場券を購入できる。	重複販売、残数不整合、購入済み未発券、決済失敗時の枠戻し漏れ。	販売上限、残数境界、Web×発券機競合、決済失敗時の補償処理を確認します。
正規利用者が確実に入場できる	購入済みの利用者が、正しい時間に、1回だけ入場できる。	正規QRで入れない、同一QRで複数入場できる、時間枠判定が誤る。	QR状態、時間枠境界、同時読取、使用済み反映を確認します。
円滑な入場運用を維持する	開園直後や人気枠でも入場処理が滞らない。	入口滞留、ゲート応答遅延、集中アクセス時の処理詰まり。	性能集中、ゲート並列、開園直後ピーク、長時間運用を確認します。
運用継続性を確保する	一部機器や通信に障害があっても影響範囲を制御できる。	ゲート停止、ハブ停止、通信断、復旧後不整合。	縮退運用、通信断復旧、データ復旧、同期再開を確認します。
利用者・係員が状況を判断できる	残数、エラー、障害状態が分かり、適切に案内・対応できる。	残数表示不整合、エラー理由不明、障害検知不可。	残数表示、エラー表示、通知、ログ、監視、コンソール表示を確認します。
将来変更時の品質低下を防止する	上限変更、ゲート追加、ハブ変更などの将来変更に対応できる。	変更影響が分からず、必要な回歸確認が漏れる。	変更種別ごとのリグレッション対象を定義します。

4. Step1 要求分析_確認方針の拡充

◆ 各領域ごとに確認方針を設定

要求分析

重点確認すべき品質観点

- 入場者の利便性
- 動物園運営の継続性
- 将来の運用拡張性

製品価値を守るための確認観点

- 販売機会の損失を防ぐ
- 正規利用者を確実に入場させる
- 入場運用を円滑に維持する
- …ほか多数

製品価値を守るため、販売、発券、入場、表示、運用、障害復旧、変更耐性の各領域で確認方針を設定。

確認軸	領域	確認方針	代表的に防ぐ失敗
販売機会を損なわない	販売	時間枠の販売上限を超過しない。	重複販売、残数超過、販売機会損失。
販売機会を損なわない	発券	購入済み入場券を確実に発券・表示できる。	購入済み未発券、発券途中停止、再発券制約違反。
正規利用者が確実に入場できる 円滑な入場運用を維持する	入場	QRコードを一回だけ利用可能にする。	同一QRの再利用、複数ゲート同時読取による重複入場。
正規利用者が確実に入場できる	時間枠	指定された時間条件に応じて入場可否を判定する。	開始前入場、終了後入場、日付境界誤り。
利用者・係員が状況を判断できる	表示/案内	利用者・係員が残数やエラー状態を判断できる。	表示不整合、誤誘導、問い合わせ増加。
円滑な入場運用を維持する 運用継続性を確保する	運用継続	一部障害時も影響範囲を制御して運用できる。	ゲート停止、ハブ停止、通信断による全面停止。
運用継続性を確保する	障害回復	復旧後に販売数、入場数、残数、QR状態が整合する。	復旧後の残数ズレ、使用済み反映漏れ。
将来変更時の品質低下を防止する	変更耐性	将来変更時に必要な再確認範囲を判断できる。	回帰漏れ、変更影響不明。

アーキテクチャ設計

4. Step2 アーキテクチャ設計_構造化

アーキテクチャ設計

アーキテクチャ層に構造化

- 業務価値
- 業務ルール
- ...ほか多数

変更点・リスク・優先度の
洗い出し

変更影響分析・リグレーション
観点の洗い出し

8つのアーキテクチャ層に構造化

➤ 各アーキテクチャ層ごとにテスト方針・リスク・優先度を設定

Layer	アーキテクチャ層	何を守るか	なぜ必要か	主な確認内容	代表リスク	代表テスト	要求分析との繋がり	優先	優先度根拠
L1	業務価値保証	販売機会を損なわない / 正規利用者が入場できる / 運用継続できる	V03で利用者増加・入場上限拡大が発生するため、単体機能ではなく業務全体で成立していることを確認する。	購入→発券→入場のE2Eシナリオ、会員/ゲスト、Web/発券機、係員運用	業務不成立、購入済みなのに入れない、係員が判断できない	業務シナリオテスト	品質保証対象価値・販売/入場/運用価値	S	購入から入場までの業務成立は、利用者価値と施設運営価値そのものに直結するため。
L2	業務ルール保証	時間枠、残数、購入上限、購入可能期間、券利用可能時間	時間枠・残数・購入上限・使用制限を守れないと、販売/入場の秩序が崩れる。	30分枠、60人上限、0～9999設定、29日後購入、使用可能時間境界	重複販売、時間外入場、残数不整合、境界誤り	境界値テスト・同値分割・組合せテスト	時間枠・残数管理要求	S	時間枠・残数・上限の誤りは、重複販売や時間外入場へ直接つながるため。
L3	購入/発見保証	購入、決済、取消、タイムアウト、補償処理	販売や確保と決済結果の順序差が残数不整合に直結するため、処理単位で保証する。	Web購入、発券機購入、現金/QR/カード/電子マネー、取消、失敗、再試行	枠戻し漏れ、二重購入、購入済み未発券、決済後不整合	購入処理テスト・異常系テスト	購入/決済/発券要求	A	購入・決済は重要だが、残数や同時の真実はL1・L2・L5でも一部検出可能なため。
L4	状態遷移保証	券ライフサイクル、QR状態、使用済み更新、無効化	入場券は状態を持つため、処理結果だけでなく状態遷移そのものを確認する。	予約→購入→発券→未使用→使用済み→無効、再発券可否、期限切れ	同一QR再利用、使用済み反映漏れ、期限外入場、再発券制約違反	状態遷移テスト	QR/入場券仕様・券ライフサイクル	A	不正入場や再利用に關する重要領域だが、対象は券状態とQR状態に限定されるため。
L5	分散同期保証	Web/クラウド2/入場管理/ゲート/バゾ/ゲート/残数表示の同期	複数システムがデータを保持・コピーするため、同期遅延や反映漏れを独立保証する。	購入情報反映、残数同期、入場情報同期、使用済み反映、残数表示更新	購入情報反映漏れ、残数ズレ、使用済み同期漏れ、表示不整合	サブシステム連携テスト・データ整合テスト	データ連携仕様	S	同期不整合は販売・入場・残数表示・使用済み反映へ広く波及するため。
L6	障害耐性保証	通信断、機器停止、復旧、経過運転	営業中に一部障害が発生しても、影響範囲を把握し運用を継続/復旧できることを保証する。	通信断、入場管理再起動、ゲートバゾ停止、ゲート停止、発券機停止、残数表示器停止	全面停止、復旧後不整合、障害検知不可、係員判断不能	障害/リカバリテスト	運用/障害要求	A	常時発生するものではないが、発生時は営業継続や係員判断へ大きく影響するため。
L7	並列性保証	同時購入、同時入場、複数ゲート、競合制御	V03ではゲート追加・上限拡大により並列事象が増えるため、競合を独立保証する。	Web+発券機同時購入、同一QR同時利用、3ゲート同時入場、残数1競合	重複販売、重複入場、排他制御不備、同期競合	並列/競合テスト	V03差分・重複販売/重複入場/リスク	S	V03のゲート追加・上限拡大により、同時購入や同時入場が中心リスクになるため。
L8	性能・耐久保証	長時間稼働、大量購入、大量入場、ピーク運用	長時間稼働とV03の入場者増加に伴う処理能力を確保する。	24時間、72時間、1週間、開園直後ピーク、大量購入、大量入場、ログ/容量	性能劣化、容量枯渇、応答遅延、営業時間中停止	性能・耐久テスト	長時間稼働要求・V03利用者増	S	長時間稼働は重点要求であり、利用者増加・ゲート増設後の処理能力確認が必要となるため。

06_優先度評価基準			
優先度	意味	判定の目安	レビュー時の見方
S	確認しないと品質保証目的のものが成立しない。	業務価値へ直結し、障害時の影響が大きく、V03変更点にも直結する。	原則として最優先で確認対象に含める。
A	業務影響が大きく、主要リスクを検出するために必要。	影響は大きい、他レイヤでも一部検出可能、または発生箇所が限定される。	主要テストに含め、条件に応じて範囲を具体化する。
B	重要だが、変更内容や条件に応じて実施有無を判断する。	局所的な影響、特定条件での発生、補充的な確認。	変更内容に応じて実施を判断する。
C	大型変更や特定案件時に重点確認する。	通常変更では優先度が下がるが、大規模変更時は必要。	大型変更時の追加確認候補とする。

07_L1～L8優先度の根拠			
Layer	優先度	根拠キーワード	根拠説明
L1 業務価値保証	S	業務価値直結	購入から入場までの業務が成立しない場合、システムの提供価値そのものを失うため。
L2 業務ルール保証	S	重複販売防止の根幹	時間枠・残数・上限の誤りは、重複販売や時間外入場へ直接つながるため。
L3 トランザクション保証	A	補充検出可能	購入・決済の影響は大きい、L1・L2・L5でも一部異常を検出できるため。
L4 状態遷移保証	A	券状態に限定	同一QR再利用など重要リスクにつながるが、対象は券ライフサイクルに限定されるため。
L5 分散同期保証	S	全システムへ波及	連携不整合は販売、入場、残数表示、使用済み反映へ広く波及するため。
L6 障害耐性保証	A	運用継続に影響	発生頻度は常時ではないが、発生時は営業継続や係員判断に影響するため。
L7 並列性保証	S	V03重点変更	ゲート追加・上限拡大により、同時購入・同時入場・競合制御がV03の中心リスクになるため。
L8 性能・耐久保証	S	重点要求事項	長時間稼働保証が重点要求であり、利用者増加・ゲート増設後の処理能力を確認する必要があるため。

4. Step2 アーキテクチャ設計_リグレッション

アーキテクチャ設計

アーキテクチャ層に構造化

- 業務価値
- 業務ルール
- …ほか多数

変更点・リスク・優先度の
洗い出し

変更影響分析・リグレッシ
ョン観点の洗い出し

今回の変更点がどのアーキテクチャ層へ影響するかを確認し、テスト範囲を設定

変更内容	影響レイヤ	影響理由	代表的に再確認する観点	回復判断
時間枠仕様変更	L2,L4,L5	時間枠判定、券の有効/無効、入場券 情報同期へ波及するため。	時間枠境界、期限切れ無効化、購入情報同期	必須回復
残数/上限変更	L2,L3,L5,L7,L8	販売可否、販売枠確保、残数同期、同 時購入、負荷へ影響するため。	残数境界、販売枠確保、Web+券券機同時購入、ピー ク負荷	必須回復
ゲート追加/ゲート制御 変更	L4,L5,L6,L7,L8	QR状態、使用済み同期、障害時運用、 同時入場、性能へ影響するため。	同一QR同時利用、使用済み同期、ゲート停止、複数 ゲート同時入場	必須回復
決済方式追加/決済IF 変更	L3,L5	購入結果と残数更新、外部連携、購入 情報同期に影響するため。	支払手段差、取消/失敗補償、購入情報同期	条件付き回復
表示/インジケータ変更	L1,L5,L6	利用者案内、残数表示同期、障害通知 へ影響するため。	利用者導線、残数表示同期、監視/通知	条件付き回復
同期方式/通信方式変 更	L5,L6,L7	購入情報、使用済み情報、復旧後突 合、競合処理へ影響するため。	購入情報同期、使用済み同期、通信断、同一QR同時 利用	必須回復
ログ/監視/運用手順変 更	L1,L6,L8	係員判断、障害検知、長時間稼働監視 に影響するため。	係員運用、監視/通知、長時間稼働	条件付き回復
券種/QR仕様変更	L2,L4,L5,L7	購入期間、券状態、同期、同時入場判 定へ影響するため。	購入期間、券状態遷移、購入情報同期、同一QR同時 利用	必須回復

テスト詳細設計

4. Step3 テスト詳細設計_テストケース作成

テスト詳細設計

詳細なテストケースへの落とし込み

- 手順・期待結果の記述
- 観点ID・優先度の紐づけ

テスト実施計画への反映

各アーキテクチャ層ごとに詳細なテストケースを作成

01_テストケース候補一覧												
TCは実行可能な単位です。詳細手順は実装時に操作・確認手順へ展開します。												
TC	TP	Layer	条件ID	カテゴリ	条件	目的	操作/状態	期待確認	優先	証跡		
TC-001	TP-01	L1	C-01-01	業務シナリオ	代表正常条件	購入→発券→入場E2Eを確認する	標準操作	期待どおり	S	画面/D8/ログ/同期結果		
TC-002	TP-01	L1	C-01-02	業務シナリオ	境界/異常条件	購入→発券→入場E2Eを確認する	境界値または失敗操作	制御・エラーが正しい	S	画面/D8/ログ/同期結果		
TC-003	TP-02	L1	C-02-01	係員運用	代表正常条件	障害・残数・エラー時の判断を確認する	標準操作	期待どおり	A	画面/D8/ログ/同期結果		
TC-004	TP-02	L1	C-02-02	係員運用	境界/異常条件	障害・残数・エラー時の判断を確認する	境界値または失敗操作	制御・エラーが正しい	A	画面/D8/ログ/同期結果		
TC-005	TP-03	L1	C-03-01	利用者案内	代表正常条件	表示・エラー・案内導線を確認する	標準操作	期待どおり	A	画面/D8/ログ/同期結果		
TC-006	TP-03	L1	C-03-02	利用者案内	境界/異常条件	表示・エラー・案内導線を確認する	境界値または失敗操作	制御・エラーが正しい	A	画面/D8/ログ/同期結果		
TC-007	TP-04	L2	C-04-01	時間帯境界	代表正常条件	開始前/開始時/終了時/終了後を確認する	標準操作	期待どおり	S	画面/D8/ログ/同期結果		

…ほか多数

4. Step3 テスト詳細設計_テスト実施計画

テスト詳細設計

詳細なテストケースへの落とし込み

- 手順・期待結果の記述
- 観点ID・優先度の紐づけ

テスト実施計画への反映

詳細設計からテスト実装へ移行。

01_実装タスク一覧						
実装で漏れやすいデータ、環境、証拠、耐久監視をタスク化しています。						
タスクID	タスク	内容	参照シート	完了条件	分類	
IMP-01	TP/条件確定	13/14をレビューし、TPと条件の過不足を解消する。	13_テストパターン体系、14_テスト条件設計	TP/条件がレビュー済み	詳細設計	
IMP-02	組合せ確定	必須組合せと条件付き組合せを確定する。	15_組合せ分析	実行予定組合せが確定	詳細設計	
IMP-03	TC具体化	TCごとに前提、操作、期待結果、証拠を具体化する。	16_テストケース設計	実行者が実施可能	詳細設計	
IMP-04	データ作成	残数、QR状態、通信断中データ、大量データを作成する。	18_テストデータ設計	データ作成/初期化可能	実装	
IMP-05	環境準備	ST、複数ゲート、通信断、性能、耐久環境を準備する。	19_環境設計	環境条件が満たされる	実装	
IMP-06	確認手段実装	画面/API/DB/ログ/突合/負荷の実装手段を準備する。	17_テスト実装方針	証拠取得可能	実装	
IMP-07	耐久シナリオ実装	24h/72h/1週間相当の実行手順、監視、収集方法を準備する。	21_性能・耐久テスト詳細	監視・収集・判定が可能	実装	

…ほか多数

5. まとめ

本活動では『要求分析→アーキテクチャ設計→詳細設計』という流れでテスト設計を進め、最終的な各成果物の作成を行った。

最終的な成果物を振り返ると...

- **反省点**としては、「要求分析」「アーキテクチャ設計」とは何かの学習から始めたため、要求分析・アーキテクチャ設計に時間がかかってしまい、詳細設計に割く時間が足りなかった。
 - 決勝では詳細設計を突き詰め、最終成果物の完成を目指す。
- ただし、今回作成した要求分析・アーキテクチャ設計を基にテスト詳細設計を突き詰めれば**[だんだん動物園-入場システム]の改修に対する品質担保は達成可能**と考える
- また、前回参加時にご指摘いただいた資料間のトレーサビリティを意識して作業を進めたため、内容を追いやすい資料構成になっていると考える

ご清聴ありがとうございました

