

証券バックオフィス業務システム

内部連結テストケース一覧表

作成日: 2024年1月 | システム: Intra-mart

テストケースID	テスト項目	テスト内容	入力条件	期待結果	優先度	実行時間	担当者	実行日	結果
1. プレゼンテーション層⇄アプリケーション層連携テスト									
PT-APP-001	顧客新規登録 BPM呼び出し	メニュー「顧客新規登録」からBPMプロセス起動	メニューID: "M001" ユーザーID: "U001" 権限: "REGISTER"	BPMプロセスID: "BP_CUST_REG"開始 処理ID生成 ログ出力確認	高	2秒以内			
PT-APP-002	顧客名パラメータ送信	顧客登録画面からBPMへ顧客名送信	顧客名: "山田太郎" フィールド: "customer_name" 文字コード: UTF-8	BPM変数"customerName"に"山田太郎"設定確認 文字化けなし	高	1秒以内			
PT-APP-003	口座番号パラメータ送信	口座開設画面からBPMへ口座番号送信	口座番号: 1234567890 型: Long 桁数: 10桁	BMP変数"accountNumber"に1234567890設定確認 型変換正常	高	1秒以内			
PT-APP-004	生年月日パラメータ送信	顧客情報画面からBPMへ生年月日送信	生年月日: "1990-05-15" フォーマット: "yyyy-MM-dd" 検証: 有効日付	BPM変数"birthDate"にDate(1990-05-15)設定確認 日付妥当性チェック通過	高	1秒以内			
PT-APP-005	ログインユーザー情報継承	ユーザーセッション情報のBPM引き継ぎ	ログインユーザー: "営業担当A" ユーザーID: "EIGYO001" セッションID: "SES123456"	BPMでユーザーID: "EIGYO001"取得可能 権限情報継承確認	高	1秒以内			
PT-APP-006	処理成功結果受信	顧客登録BPM完了後の成功メッセージ表示	BPM戻り値: "SUCCESS" 顧客ID: "C00012345" 処理時刻: 現在時刻	画面に"顧客登録が完了しました" 顧客ID: C00012345"表示 成功アイコン表示	高	2秒以内			
PT-APP-007	バリデーションエラー表示	必須項目未入力時のエラー表示	顧客名: ""(空文字) BPMエラーコード: "E001" エラーメッセージ: "必須項目未入力"	画面に"顧客名は必須です"赤字表示 入力フィールドハイライト	高	1秒以内			
PT-APP-008	次画面遷移制御	顧客登録完了後の口座開設画面遷移	BPM処理結果: "NEXT_ACCOUNT" 顧客ID: "C00012345" 遷移先: "account_create"	口座開設画面 (account_create.html)表示 顧客ID引継ぎ確認	中	3秒以内			
PT-APP-009	非同期処理進捗表示	大量データ処理中のプログレスバー表示	処理時間: 10秒 進捗更新間隔: 1秒 処理件数: 1000件	プログレスバー0%→100%表示 画面操作可能状態維持 処理件数表示	中	10秒			
PT-APP-010	タイムアウトエラー処理	BPM処理30秒タイムアウト時の制御	タイムアウト設定: 30秒 実処理時間: 35秒 エラーコード: "T001"	"処理がタイムアウトしました 再度実行してください"表示 リトライボタン表示	中	30秒			
2. アプリケーション層⇄データ層連携テスト									
APP-DB-001	顧客情報登録サービス呼び出し	BPMからLogicDesigner顧客登録実行	サービス名: "CustomerRegistration" パラメータ数: 5個 呼び出し方式: 同期	LogicDesigner正常実行 戻り値: 顧客ID 実行ログ出力	高	3秒以内			
APP-DB-002	顧客マスタ新規登録	顧客情報の新規登録処理	顧客名: "佐藤花子" 口座番号: 9876543210 住所: "東京都港区1-1-1" 電話: "03-1234-5678"	T_CUSTOMER表に1件INSERT 顧客ID: "C00012346"採番 登録日時設定	高	2秒以内			
APP-DB-003	顧客情報更新処理	既存顧客の住所変更処理	顧客ID: "C00012345" 新住所: "大阪府大阪市2-2-2" 更新者: "EIGYO001" 更新理由: "転居"	T_CUSTOMER表UPDATE実行 更新日時、更新者設定 履歴テーブル登録	高	1秒以内			
APP-DB-004	顧客情報論理削除	顧客情報の論理削除処理	顧客ID: "C00012345" 削除フラグ: "1" 削除者: "ADMIN001" 削除理由: "契約終了"	T_CUSTOMER.DELETE_FLG=1 設定 削除日時、削除者設定 関連データ無効化	高	1秒以内			
APP-DB-005	顧客情報検索処理	顧客名による部分一致検索	検索条件: 顧客名 LIKE "%田%" 削除フラグ: "0" 取得上限: 100件	該当する顧客リスト取得 件数: 3件 ソート: 顧客ID昇順	高	2秒以内			
APP-DB-006	複数テーブル更新制御	顧客情報と口座情報の同時更新	顧客テーブル: 住所変更 口座テーブル: ステータス変更 トランザクション: 必須	両テーブル正常更新またはROLLBACK データ整合性確保 ログ出力	高	3秒以内			
APP-DB-007	楽観的排他制御	同一顧客情報の同時更新制御	更新対象: 顧客ID "C00012345" 同時更新ユーザー: 2名 バージョン管理: 有効	先行更新成功 後続更新で排他エラー エラーメッセージ表示	高	2秒以内			
APP-DB-008	金額データ型変換	BPMからの金額データ変換処理	BPM金額: String "1000000" 通貨: "JPY" 小数点: 0桁	LogicDesigner: BigDecimal 1000000.00 精度保持 桁数チェック	中	1秒以内			
APP-DB-009	NULL値データ処理	任意項目NULL値での登録処理	顧客名: "田中次郎" 電話番号: NULL メールアドレス: NULL 備考: NULL	T_CUSTOMER表正常INSERT NULL項目はNULL格納 必須項目チェック通過	中	1秒以内			
APP-DB-010	大量データ一括処理	15,000件の顧客データ一括登録	CSVファイル: 15,000行 バッチサイズ: 1,000件 コミット間隔: 1,000件	15回に分割してINSERT 処理時間: 5分以内 エラー件数: 0件	中	300秒以内			
3. バッチ機能とオンライン機能連携テスト									
BATCH-ON-001	日次バッチ結果画面反映	夜間バッチ処理結果の翌日画面表示	処理日: "2024-01-15" 処理件数: 5,280件 エラー件数: 3件 実行時間: 2時間15分	ダッシュボード画面に"処理件数: 5,280件 エラー件数: 3件"表示 詳細リンク表示	高	1秒以内			
BATCH-ON-002	バッチ実行状況監視	オンライン画面でのバッチ進捗確認	バッチID: "BATCH_001" 進捗: 45% 処理中ステータス 開始時刻: 23:00	管理画面に進捗バー45%表示 "実行中"ステータス表示 残り時間表示	中	1秒以内			
BATCH-ON-003	オンライン起動バッチ実行	画面操作による手動バッチ起動	バッチ名: "顧客データ集計" 実行ユーザー: "ADMIN001" パラメータ: "2024-01" 実行権限: 有	バッチキュー登録 実行ID: "EXEC_20240115_001" 開始通知表示	中	2秒以内			
BATCH-ON-004	バッチ処理排他制御	バッチ実行中のオンライン更新制限	バッチ実行中テーブル: T_CUSTOMER オンライン更新要求 排他レベル: テーブルロック	"バッチ処理中のため更新できません"表示 リトライ案内表示	高	1秒以内			
4. 外部システムモック連携テスト									
MOCK-001	STAR API正常応答模擬	モックSTARからの顧客情報取得	顧客ID: "ST001234" APIエンドポイント: "/api/customer" 認証トークン: 有効	JSON形式顧客情報取得 ステータスコード: 200 レスポンス時間: 500ms以内	高	500ms以内			
MOCK-002	STAR API異常応答模擬	モックSTARからのエラー応答	顧客ID: "ST999999" 存在しない顧客 エラーシナリオ: NOT_FOUND	エラーレスポンス取得 ステータスコード: 404 エラーメッセージ: "顧客が見つかりません"	高	500ms以内			
MOCK-003	Adobe Sign署名完了模擬	モック署名サービスからの完了通知	文書ID: "DOC_20240115_001" 署名者: "customer@example.com" 署名方法: "電子署名"	署名完了イベント受信 署名日時: 現在時刻 文書ステータス: "完了"	中	1秒以内			
MOCK-004	UiPath実行完了模擬	モックRPAからの処理完了通知	プロセス名: "DataEntry_Process" 処理件数: 100件 実行環境: "DEV"	実行完了ステータス受信 処理結果サマリー取得 実行ログ確認	中	10秒以内			
MOCK-005	Exchange送信完了模擬	モックメールサーバーからの送信結果	宛先: "client@example.com" 件名: "口座開設完了通知" 添付ファイル: PDF	送信成功レスポンス受信 メッセージID取得 送信ログ確認	中	2秒以内			
5. パフォーマンス・負荷連携テスト									
PERF-001	画面応答性能	顧客検索画面の応答時間測定	同時ユーザー数: 50名 検索条件: 顧客名部分一致 データ件数: 100万件	平均応答時間: 2秒以内 95%ile応答時間: 3秒以内 エラー率: 0%	高	300秒			
PERF-002	BPM処理性能	複雑なワークフロー処理時間	承認フロー: 5段階 並列処理: 3タスク 処理データ: 1,000件	処理完了時間: 10秒以内 CPU使用率: 80%以下 メモリ使用量: 2GB以下	中	60秒			
PERF-003	データベース負荷	大量データ検索時の性能	検索対象: 100万件テーブル 複合インデックス使用 同時実行: 20クエリ	検索時間: 5秒以内 メモリ使用量: 4GB以下 ロック競合: なし	中	300秒			
PERF-004	メモリリーク検証	8時間連続処理でのメモリ監視	連続実行時間: 8時間 処理間隔: 10秒 監視間隔: 1分	メモリ使用量増加: 10%以内 OutOfMemoryError発生なし GC実行回数: 正常範囲	低	28800秒			

テスト実施要領

優先度別実施順序:

- 高: システムの基本機能、必須実施
- 中: 業務運用上重要、実施推奨
- 低: 安定性確認、時間に余裕があれば実施

實施環境：

- 開発環境: プライベートクラウド
- テスト環境: AWS
- ツール: Intra-mart, LogicDesigner, BPM

— — — — —

- 顧客データ: 10万件
- 取引データ: 100万件

△地盤沈下

- 機能テスト: 100%成功
- 性能テスト: 基準値以内

このテストケース一覧表は、証券バックオフィス業務システムのIntra-mart内部連結テストを包括的にカバーしています。
各テストケースは段階的に実施し、前工程の成功を確認してから次工程に進んでください。