

IkeaDocuScan V3 - Business Test Plan Overview

Version: 1.0 **Date:** 2025-11-14 **Framework:** .NET 10.0 **Application Type:** Blazor Hybrid (Server + WebAssembly)

Table of Contents

1. [Introduction](#)
2. [Test Plan Structure](#)
3. [User Roles & Capabilities](#)
4. [Test Environment Setup](#)
5. [Test Data Requirements](#)
6. [Test Execution Guidelines](#)
7. [Defect Reporting](#)
8. [Sign-off Criteria](#)

Introduction

This test plan provides comprehensive business acceptance testing for the IkeaDocuScan V3 document management system. The testing covers functional requirements, edge cases, security validation, and role-based access control.

Objectives

- Verify all business use cases function correctly
- Validate role-based permissions and data filtering
- Identify edge cases that could corrupt data or cause system failures
- Ensure security controls are properly implemented
- Validate configuration management through GUI
- Test real-time updates and concurrent user scenarios

Scope

In Scope: - Document registration, editing, and check-in workflows - User permission management and filtering - Reference data management (countries, currencies, counter parties, etc.) - Action reminder functionality - Search and reporting - Email notifications - Configuration management - Security and authorization - Real-time updates (SignalR) - Audit trail verification

Out of Scope: - Performance/load testing (separate test plan) - Database migration testing (covered in deployment) - Infrastructure testing (IIS, SQL Server configuration)

Test Plan Structure

The test plan is divided into **role-specific documents** plus a **security test plan**:

Role-Specific Test Plans

Document	Role	Focus Areas
BUSINESS_TEST_PLAN_READER.md	Reader	View-only access, search, filtering, export, permission boundaries
BUSINESS_TEST_PLAN_PUBLISHER.md	Publisher	Document registration, check-in, editing, email sending, Reader capabilities
BUSINESS_TEST_PLAN_SUPERUSER.md	SuperUser	Full administrative functions, reference data management, user permissions, configuration, Publisher capabilities

Cross-Cutting Test Plans

Document	Focus Areas
SECURITY_TEST_PLAN.md	API security, authentication bypass attempts, input validation, SQL injection, XSS, CSRF, authorization escalation

Test Scenario Format

Each test scenario follows this structure:

```
##### TC-XXX: Test Case Title

**Objective:** What are we testing?

**Pre-conditions:**
- System state before test
- Required test data
- User logged in as [Role]

**Test Steps:**
1. Step 1
2. Step 2
3. Step 3

**Expected Result:**
- What should happen

**Actual Result:**
[To be filled during testing]

**Status:** Not Run | Pass | Fail | Blocked

**Notes:** 
[Additional observations]
```

User Roles & Capabilities

Role Summary

Capability	Reader	Publisher	SuperUser
View Documents	<input type="checkbox"/> Filtered	<input type="checkbox"/> Filtered	<input type="checkbox"/> All
Search Documents	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Export to Excel	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Register Document	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Edit Document	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Delete Document	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Check-in PDF	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Send Email	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
View Action Reminders	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Manage Users	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Manage Reference Data	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
System Configuration	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
View Audit Trail	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Permission Filtering

Reader & Publisher: - See only documents matching their **UserPermission** records - Filtering by: Document Type, Country, Counter Party - Cannot see documents outside their permissions

SuperUser: - Sees all documents regardless of permissions - No filtering applied

Test Environment Setup

Required Components

1. **Application Server**
 - Windows Server with IIS
 - IkeaDocuScan V3 deployed
 - .NET 10.0 Runtime
 - URL: <https://docuscan-test.company.com> (example)
2. **Database Server**
 - SQL Server with **IkeaDocuScan** database
 - All migration scripts executed
 - Test data loaded
3. **Test User Accounts**
 - **Reader User:** DOMAIN\test_reader
 - **Publisher User:** DOMAIN\test_publisher
 - **SuperUser:** DOMAIN\test_superuser
 - All users created in DocuScanUser table
 - Appropriate permissions assigned
4. **SMTP Server**
 - Configured for email testing
 - Test mailbox for receiving notifications
5. **Action Reminder Service**
 - Windows Service installed and configured
 - Pointing to test database

Test Data Setup

Run the following SQL script to create test data:

-- See [TEST_DATA_SETUP.sql](#) in this folder

Browser Compatibility

Test on the following browsers: - Microsoft Edge (latest) - Google Chrome (latest) - Mozilla Firefox (latest)

Test Data Requirements

Minimum Test Data

Data Type	Minimum Required	Purpose
Users	3	One per role (Reader, Publisher, SuperUser)
Document Types	5	Test filtering and permissions
Countries	5	Test filtering and permissions
Counter Parties	10	Test filtering and permissions
Currencies	5	Test financial data entry
Document Names	10	Test autocomplete and selection
Documents	50+	Mix of statuses, types, and dates
PDF Files	20+	Various sizes and formats
User Permissions	15+	Mix of restricted and broad permissions

Test Document Scenarios

Create documents with: - Valid data (happy path) - Missing optional fields - All fields populated - Very long text in fields (test limits) - Special characters in text fields - Past, current, and future dates - ActionDate set (for reminder testing) - Documents with and without attached PDFs - Duplicate document numbers (different types) - Same counter party across multiple documents

Test PDF Files

Prepare PDF files with: - Valid barcode in filename: 12345678.pdf - Invalid barcode format: ABC123.pdf - No barcode: document.pdf - Very long filename - Special characters in filename: test@#.pdf - Large file (near 50MB limit) - Empty PDF (0 bytes) - Corrupted PDF - Non-PDF file with .pdf extension - Multiple pages vs single page PDFs

Test Execution Guidelines

Execution Order

1. **Setup Phase**
 - Verify test environment is ready
 - Load test data
 - Verify all services running
 - Clear browser cache
2. **Reader Tests**
 - Execute all Reader test cases
 - Document results
3. **Publisher Tests**
 - Execute all Publisher test cases
 - Includes all Reader capabilities
 - Document results
4. **SuperUser Tests**
 - Execute all SuperUser test cases
 - Includes Publisher and Reader capabilities
 - Document results
5. **Security Tests**

- Execute security test cases
 - Attempt to bypass controls
 - Document vulnerabilities
6. **Edge Case & Stress Tests**
- Concurrent user scenarios
 - Large data volumes
 - Boundary conditions
 - Document results

Test Execution Best Practices

- **Fresh Session** - Start each test suite with a fresh browser session
- **Clean State** - Reset test data between major test suites if needed
- **Screenshots** - Capture screenshots of failures
- **Detailed Notes** - Document unexpected behavior even if test passes
- **Real-Time Updates** - Open multiple browser windows to verify SignalR updates
- **Audit Trail** - Check audit trail after each destructive operation
- **Event Logs** - Monitor Windows Event Log during testing

Test Result Tracking

Use the following status indicators:

Status	Symbol	Meaning
Not Run	□	Test not yet executed
Pass	□	Test passed as expected
Fail	□	Test failed, defect found
Blocked	□	Cannot test due to blocker
Skip	□	Test skipped (out of scope for this cycle)

Defect Reporting

Defect Severity Levels

Level	Description	Example
Critical	System crash, data loss, security breach	SQL injection vulnerability, document data corrupted
High	Major feature broken, no workaround	Cannot save documents, check-in fails completely
Medium	Feature partially works, workaround exists	Search returns incorrect results sometimes
Low	Cosmetic issue, minor inconvenience	Button alignment off, typo in label

Defect Template

****Defect ID:**** DEF-XXX
****Severity:**** Critical | High | Medium | Low
****Test Case:**** TC-XXX
****User Role:**** Reader | Publisher | SuperUser

****Summary:****
Brief description of the issue

****Steps to Reproduce:****

1. Step 1
2. Step 2
3. Step 3

****Expected Result:****
What should happen

****Actual Result:****
What actually happened

****Screenshots:****
[\[Attach screenshots\]](#)

****Environment:****

- Browser: Chrome 120
- User: test_publisher
- Date/Time: 2025-11-14 10:30

****Notes:****
Additional information

Sign-off Criteria

Test Completion Criteria

- All Critical & High severity defects** - Resolved or accepted
- 95% test coverage** - At least 95% of test cases executed
- Role-based access** - All three roles verified
- Edge cases** - All identified edge cases tested
- Security tests** - All security tests passed
- Audit trail** - Verified for all destructive operations
- Real-time updates** - SignalR updates working across users
- Email notifications** - All email scenarios verified
- Action Reminder Service** - Tested and verified

Acceptance Criteria

The application is ready for production when:

1. All sign-off criteria met
 2. Stakeholder approval obtained
 3. User training completed
 4. Documentation reviewed and approved
 5. Deployment plan approved
 6. Rollback plan in place
 7. Production support team briefed
-

Test Schedule (Example)

Phase	Duration	Activities
Test Preparation	2 days	Environment setup, test data creation, test account creation
Reader Testing	1 day	Execute all Reader test cases
Publisher Testing	2 days	Execute all Publisher test cases
SuperUser Testing	2 days	Execute all SuperUser test cases
Security Testing	1 day	Execute all security test cases
Edge Case Testing	1 day	Execute edge case and concurrent user tests
Regression Testing	1 day	Retest after defect fixes
Sign-off	1 day	Compile results, obtain approvals
TOTAL	11 days	

Test Deliverables

1. **Test Plan Documents** (this folder)
 - TEST_PLAN_OVERVIEW.md
 - BUSINESS_TEST_PLAN_READER.md
 - BUSINESS_TEST_PLAN_PUBLISHER.md
 - BUSINESS_TEST_PLAN_SUPERUSER.md
 - SECURITY_TEST_PLAN.md
2. **Test Data Scripts**
 - TEST_DATA_SETUP.sql
3. **Test Results**
 - Completed test case documents with status
 - Screenshots of failures
 - Test execution summary
4. **Defect Log**
 - All defects found during testing
 - Resolution status
5. **Sign-off Document**
 - Test summary
 - Stakeholder approvals

Quick Reference

Test Execution Order: 1. Setup test environment 2. Reader tests → Publisher tests → SuperUser tests 3. Security tests 4. Edge case tests 5. Compile results and sign-off

Key Test Focus Areas: - Document lifecycle (Register → Check-in → Edit → Delete) - Permission filtering (ensure users see only their documents) - Edge cases (duplicates, invalid data, concurrent edits) - Security (API access, injection attacks, authorization bypass) - Real-time updates (SignalR) - Audit trail (all changes logged)

Critical Edge Cases to Test: - Duplicate barcodes - Invalid file upload attempts - Concurrent document editing - Permission changes during active session - Deleting in-use reference data - Invalid email templates - SMTP connection failures - SQL injection and XSS attempts

For detailed test cases, refer to the role-specific test plan documents.