

Test Catalog

This document catalogs all Playwright tests for the Dydra JSUI **Studio** application (`index.html`), organized by functional area and mapped to requirements from `doc/requirements-20260207.md`.

Scope: These tests cover the **User App (Studio)** component only. The **Admin App (admin.html)** has separate requirements (Section 11, Section 12) that are not covered by this test suite.

Test Organization

Each test category has its own spec file in the `tests/` directory. Tests reuse common helper functions for authentication, account management, repository management, and view management.

1. Authentication Tests (`authentication.spec.js`)

Coverage: REQ-1.1.x, REQ-1.2.x

Test Name	Description	Requirements
<code>login with valid credentials</code>	Login with correct username/password, verify successful authentication	REQ-1.1.1, REQ-1.1.2
<code>login with invalid credentials</code>	Attempt login with wrong password, verify error message	REQ-1.1.1, REQ-9.1.1
<code>logout functionality</code>	Login, then logout, verify redirect to login page	REQ-1.2.2
<code>navigation links based on auth status</code>	Verify different nav links shown when logged in vs logged out	REQ-1.2.3
<code>session persistence of account name</code>	Login, reload page, verify account name is remembered	REQ-1.2.1

Open Requirements

Requirement	Description	Notes
REQ-1.1.3	Token storage in memory	Internal implementation detail
REQ-1.1.4	Multiple authenticated accounts simultaneously	Requires multi-account test setup

2. Account Management Tests (`account.spec.js`)

Coverage: REQ-2.1.x, REQ-2.2.x

Test Name	Description	Requirements
display account information	View account details (email, fullname, homepage, etc.)	REQ-2.1.1, REQ-2.1.2
edit account profile fields	Modify account fields, save, verify persistence	REQ-2.2.1, REQ-2.2.3
view authentication token	Access and display account authentication token	REQ-2.2.4

Open Requirements

Requirement	Description	Notes
REQ-2.1.3	View account details for other accounts	Requires test account for “other” user
REQ-2.2.2	Track only edited fields (delta updates)	Internal implementation detail
REQ-2.3.1	Account creation	Admin component (not Studio)

3. Repository Management Tests (`repository.spec.js`)

Coverage: REQ-3.x

Test Name	Description	Requirements
<code>repository profile properties</code>	Modify homepage, summary, description, privacy; verify persistence	REQ-3.2.1, REQ-3.2.3
<code>repository collaboration</code>	Add collaborator with read access, verify persistence, remove collaborator	REQ-3.7.1, REQ-3.7.2, REQ-3.7.3
<code>repository creation</code>	Create new repository, verify it appears	REQ-3.3.1
<code>repository prefixes</code>	Edit repository prefixes, save, verify persistence	REQ-3.2.1
<code>repository clear</code>	Clear all triples with confirmation dialog	REQ-3.6.1

Open Requirements

Requirement	Description	Notes
REQ-3.1.2	Display quad count, disk size, license, import status	Verify all metadata fields displayed
REQ-3.2.2	Track only edited fields (delta updates)	Internal implementation detail

4. SPARQL Query Tests (`sparql.spec.js`)

Coverage: REQ-4.1.x, REQ-4.3.x

Test Name	Description	Requirements
<code>execute SPARQL query</code>	Run a simple SELECT query, verify results displayed	REQ-4.1.1, REQ-4.1.2
<code>query result formats - JSON</code>	Execute query with JSON response format	REQ-4.1.3

Test Name	Description	Requirements
<code>query result formats - CSV</code>	Execute query with CSV response format	REQ-4.1.3
<code>query result formats - XML</code>	Execute query with XML response format	REQ-4.1.3
<code>query execution time display</code>	Verify execution time is shown after query	REQ-4.3.4
<code>query keyboard shortcut</code>	Execute query using Ctrl/Cmd+Enter	REQ-4.3.2
<code>query reset functionality</code>	Modify query, reset to original	REQ-4.3.3
<code>multiple query tabs</code>	Create multiple query tabs, switch between them	REQ-4.3.6

Open Requirements

Requirement	Description	Notes
REQ-4.1.3	SVG, HTML, TSV, Turtle, N-Triples, RDF/XML, JSON-LD, SSE formats	Additional format tests needed
REQ-4.1.4	Display results in appropriate formats (tables, rendered HTML/SVG)	Format-specific rendering verification
REQ-4.1.5	Parameterized queries with dynamic input fields	Requires parameterized view setup
REQ-4.1.6	Query history with timestamps and execution times	Event log verification
REQ-4.3.1	Collapsible query editor interface	UI state test
REQ-4.3.5	Event log for query execution activities	Log display verification
REQ-4.3.7	Drag query tabs to standalone windows	Browser window handling
REQ-4.3.8	Preserve state when opening standalone editor window	Cross-window state test

5. View Management Tests (`view.spec.js`)

Coverage: REQ-4.2.x

Test Name	Description	Requirements
<code>create view</code>	Create new SPARQL view with query text	REQ-4.2.1
<code>edit view</code>	Modify view query text, save changes	REQ-4.2.1, REQ-4.2.3
<code>delete view</code>	Delete view with confirmation	REQ-4.2.1
<code>open view in pane editor</code>	Open existing view in the repository pane editor	REQ-4.2.5
<code>execute view</code>	Run a saved view and verify results	REQ-4.2.2

Open Requirements

Requirement	Description	Notes
REQ-4.2.4	List views from repository configuration	Implicit in view tests
REQ-4.2.6	Open view in a new pane	Multi-pane test
REQ-4.2.7	Open view results in new window with table display	Browser window handling

6. Navigation Tests (`navigation.spec.js`)

Coverage: REQ-5.x

Test Name	Description	Requirements
<code>client-side routing</code>	Navigate to different routes, verify URL changes	REQ-5.1.1
<code>browser back/forward</code>	Use browser navigation, verify correct pages	REQ-5.1.2
<code>tab management</code>	Open multiple tabs, switch between them, close tabs	REQ-6.3.1, REQ-6.3.2

Test Name	Description	Requirements
<code>location bar</code>	Verify location bar shows current path	REQ-5.2.4
<code>display direct URL</code>	Navigate directly to repository URL	REQ-5.1.3, REQ-5.1.4
<code>navigation navigation between account and repositories</code>	Switch between account and repository views	REQ-5.2.5

Open Requirements

Requirement	Description	Notes
REQ-5.1.5	Configurable base path for subdirectory deployment	Deployment configuration test
REQ-5.2.1	Navigation links for Home, About, Docs, Blog	External link verification
REQ-5.2.2	Navigation links for My Account, Logout (authenticated)	Partial coverage via auth tests
REQ-5.2.3	Navigation links for Signup, Login (not authenticated)	Partial coverage via auth tests
REQ-5.2.6	Cross-app navigation between user and admin SPAs	Requires admin app

7. Import/Export Tests (`import-export.spec.js`)

Coverage: `REQ-3.4.x, REQ-3.5.x`

Test Name	Description	Requirements
<code>import Turtle file</code>	Upload .ttl file, verify import success	REQ-3.4.4, REQ-3.4.7
<code>import N-Triples file</code>	Upload .nt file, verify import success	REQ-3.4.4, REQ-3.4.7
<code>import N-Quads file</code>	Upload .nq file, verify import success	REQ-3.4.4, REQ-3.4.7
<code>import from URL</code>	Import from remote URL	REQ-3.4.3

Test Name	Description	Requirements
<code>import progress</code>	Verify progress indicator	REQ-3.4.6
<code>display</code>	during import	
<code>export Turtle</code>	Export repository as	REQ-3.5.1
<code>format</code>	Turtle	
<code>export N-Triples</code>	Export repository as	REQ-3.5.1, REQ-3.5.2
<code>format</code>	N-Triples	

Open Requirements

Requirement	Description	Notes
REQ-3.4.1	Basic import support	Covered by file tests
REQ-3.4.2	Display import job status and notifications	Notification verification
REQ-3.4.5	Asynchronous imports with AcceptAsynchronous header	Async header test
REQ-3.4.6	Import progress with abort capability	Abort functionality test
REQ-3.4.7	Auto-detect .rdf, .xml, .trig, .jsonld, .json, .csv	Additional format tests
REQ-3.5.1	Export N-Quads, RDF/XML, JSON-LD, TriG, CSV formats	Additional format tests

8. UI Component Tests (`ui-components.spec.js`)

Coverage: REQ-6.x, REQ-9.x

Test Name	Description	Requirements
<code>flash message</code>	Trigger action that shows flash message	REQ-6.2.1
<code>display</code>		
<code>loading indicator</code>	Verify loading indicator during async operations	REQ-6.2.2
<code>error message</code>	Trigger error, verify user-friendly message	REQ-9.1.2
<code>display</code>		
<code>404 page</code>	Navigate to unknown route, verify 404 page	REQ-9.1.3
<code>modal dialog</code>	Open and close modal dialogs	REQ-6.2.3
<code>behavior</code>		

Test Name	Description	Requirements
confirmation	Verify confirmation	REQ-6.2.3
dialog	dialog behavior	
responsive layout	Test layout at different viewport sizes	REQ-6.1.3
elements		
keyboard	Tab through interactive	Accessibility
navigation	elements	
tooltip display	Verify tooltip display on hover	UI feedback

Open Requirements

Requirement	Description	Notes
REQ-6.1.1	Consistent page layout (header, nav, content, footer)	Layout structure test
REQ-6.1.2	Home layout variant for landing pages	Home page layout test
REQ-6.3.3	Track open accounts and repositories in tabs	Tab state persistence
REQ-6.3.4	Tabbed sub-panes within dashboard panes	Sub-pane test
REQ-9.1.1	Error messages for authentication failures	Covered in auth tests
REQ-9.2.1	Handle network errors gracefully	Network failure simulation
REQ-9.2.2	Retry mechanisms for failed operations	Retry behavior test

Common Test Helpers (`test-helpers.js`)

Shared functions used across all test specs:

```
// Authentication
login(page, username, password)
logout(page)

// Repository Management
openRepository(page, accountName, repositoryName)
deleteRepository(page, accountName, repositoryName)

// View Management
```

```

createView(page, viewName, queryText)
deleteView(page, viewName)
openViewInEditor(page, viewName)

// Navigation
openCollaborationPanel(page)
openProfilePanel(page)
openSettingsPanel(page)
openPrefixesPanel(page)

// Utilities
waitForSaveComplete(page, urlPattern)
waitForNetworkIdle(page)
getPaneId(accountName, repositoryName)

```

Test Execution

Run all tests:

```
npx playwright test
```

Run specific category:

```

npx playwright test tests/authentication.spec.js
npx playwright test tests/repository.spec.js
npx playwright test tests/sparql.spec.js

```

Run tests matching pattern:

```

npx playwright test -g "repository"
npx playwright test -g "login"

```

Requirements Coverage Summary

Studio Component Coverage

Section	Spec File	Status	Open Requirements
1. Authentication	authentication.spec.js	Partial	REQ-1.1.3, REQ-1.1.4
2. Account Management	account.spec.js	Partial	REQ-2.1.3, REQ-2.2.2
3. Repository Management	repository.spec.js, import-export.spec.js	Partial	REQ-3.1.2, REQ-3.2.2, REQ-3.4.2, REQ-3.4.5, REQ-3.4.6 (abort)

Section	Spec File	Status	Open Requirements
4. SPARQL Queries	sparql.spec.js, view.spec.js	Partial	REQ-4.1.3 (formats), REQ-4.1.4, REQ-4.1.5, REQ-4.1.6, REQ-4.2.6, REQ-4.2.7, REQ-4.3.1, REQ-4.3.5, REQ-4.3.7, REQ-4.3.8
5. Navigation	navigation.spec.js	Partial	REQ-5.1.5, REQ-5.2.1, REQ-5.2.6
6. UI Components	ui-components.spec.js	Partial	REQ-6.1.1, REQ-6.1.2, REQ-6.3.3, REQ-6.3.4
7. Data Persistence	—	Not Tested	REQ-7.x (internal implementation)
8. Invitation System	—	Not Tested	REQ-8.1.1, REQ-8.1.2, REQ-8.1.3
9. Error Handling	ui-components.spec.js	Partial	REQ-9.2.1, REQ-9.2.2
10. Configuration	—	Not Tested	REQ-10.x (deployment configuration)

Admin Component (Not Covered by Studio Tests)

Section	Description	Status
2.3 Account Creation	REQ-2.3.1	Admin component
11. Administration Interface	REQ-11.1.x through REQ-11.6.x	Admin component
12. Data Visualization	REQ-12.1.x	Admin component

Non-Functional Requirements (Not Directly Tested)

Section	Description	Notes
13. Performance	REQ-13.x	Requires performance benchmarking
14. Browser Compatibility	REQ-14.x	Requires multi-browser test matrix
15. Security	REQ-15.x	Requires security audit
16. Maintainability	REQ-16.x	Code quality metrics

Section	Description	Notes
17. External Services	REQ-17.x	API integration tests
18. External Libraries	REQ-18.x	Dependency verification

Future Test Additions

Priority 1: Functional Gaps

- Multiple authenticated accounts (REQ-1.1.4)
- Parameterized queries (REQ-4.1.5)
- Query history display (REQ-4.1.6)
- Async import with AcceptAsynchronous header (REQ-3.4.5)
- Import abort capability (REQ-3.4.6)

Priority 2: UI/UX Completeness

- All SPARQL result formats (REQ-4.1.3)
- Standalone editor windows (REQ-4.3.7, REQ-4.3.8)
- View in new pane/window (REQ-4.2.6, REQ-4.2.7)
- Collapsible editor interface (REQ-4.3.1)
- Event log display (REQ-4.3.5)

Priority 3: Admin Component

- Separate test suite for `admin.html`
- Account administration (REQ-11.2.x)
- Repository administration (REQ-11.3.x)
- Invitation management (REQ-11.4.x)
- History and monitoring (REQ-11.5.x)
- Data visualization graphs (REQ-12.x)