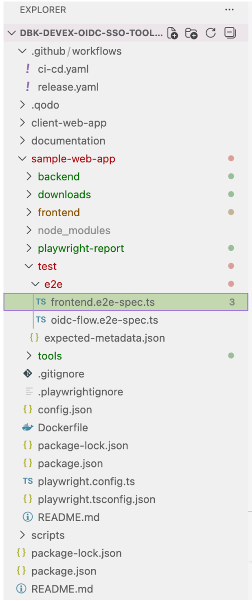
Frontend Automation E2E (OIDC & SAML)

1. Project Overview

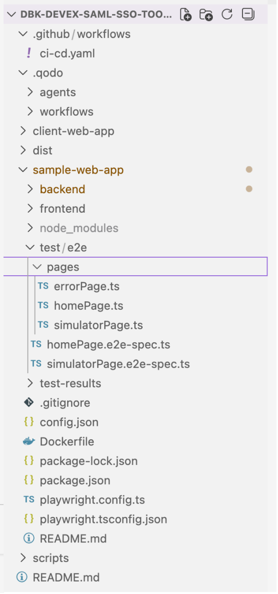
This project implements end-to-end test automation for the Sample Web Application using Playwright with TypeScript.
The automation suite is designed to run both locally and via GitHub Actions CI pipelines, producing HTML-based execution reports.

---

2. Frontend e2e test cases for OIDC and SAML can be find here under test -> e2e



```
EXPLORER
DBK-DEVEX-OIDC-SSO-TOOL...
  .github / workflows
    ci-cd.yaml
    release.yaml
  .qodo
  client-web-app
  documentation
  sample-web-app
    backend
    downloads
    frontend
    node_modules
    playwright-report
    test
      e2e
        frontend.e2e-spec.ts    3
        oidc-flow.e2e-spec.ts
      expected-metadata.json
    tools
    .gitignore
    .playwrightignore
    config.json
    Dockerfile
    package-lock.json
    package.json
    playwright.config.ts
    playwright.tsconfig.json
    README.md
  scripts
  package-lock.json
  package.json
  README.md
```



```
DBK-DEVEX-SAML-SSO-TOO...
  .github / workflows
    ci-cd.yaml
  .qodo
    agents
    workflows
  client-web-app
  dist
  sample-web-app
    backend
    frontend
    node_modules
    test / e2e
      pages
        errorPage.ts
        homePage.ts
        simulatorPage.ts
      homePage.e2e-spec.ts
      simulatorPage.e2e-spec.ts
    test-results
    .gitignore
    config.json
    Dockerfile
    package-lock.json
    package.json
    playwright.config.ts
    playwright.tsconfig.json
    README.md
  scripts
  README.md
```

Note: All Playwright-related code is isolated inside the test directory.

---

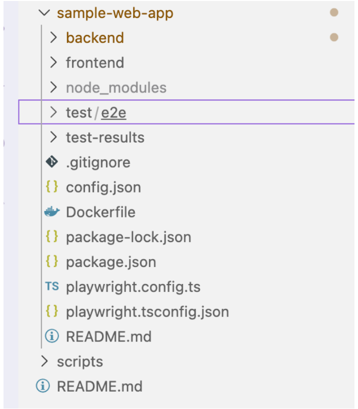
3. Technology Stack for OIDC & SAML

Component	Description
Language	TypeScript
Automation Tool	Playwright
Browsers	Chromium
CI/CD	GitHub
Reporting	Playwright HTML Report

---

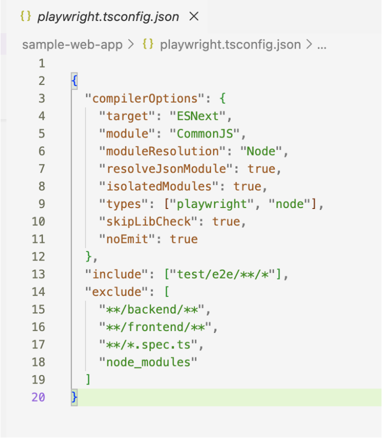
4. Playwright Configuration files:

Location of configuration files are as below:



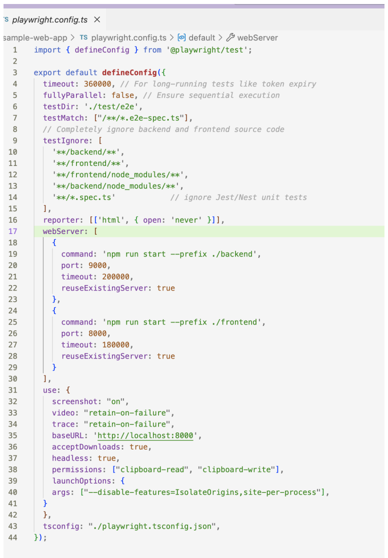
```
sample-web-app
  backend
  frontend
  node_modules
  test / e2e
  test-results
  .gitignore
  config.json
  Dockerfile
  package-lock.json
  package.json
  playwright.config.ts
  playwright.tsconfig.json
  README.md
scripts
README.md
```

Configuration for OIDC & SAML are similar.



```json

{
  "compilerOptions": {
    "target": "ESNext",
    "module": "CommonJS",
    "moduleResolution": "Node",
    "resolveJsonModule": true,
    "isolatedModules": true,
    "types": ["playwright", "node"],
    "skipLibCheck": true,
    "noEmit": true
  },
  "include": ["test/e2e/**/*"],
  "exclude": [
    "**/backend/**",
    "**/frontend/**",
    "**/*.spec.ts",
    "node_modules"
  ]
}
```



```typescript
import { defineConfig } from '@playwright/test';

export default defineConfig({
  timeout: 360000, // For long-running tests like token expiry
  fullyParallel: false, // Ensure sequential execution
  testDir: './test/e2e',
  testMatch: ["/**/*.e2e-spec.ts"],
  // Completely ignore backend and frontend source code
  testIgnore: [
    '**/backend/**',
    '**/frontend/**',
    '**/frontend/node_modules/**',
    '**/backend/node_modules/**',
    '**/*.spec.ts'            // ignore Jest/Nest unit tests
  ],
  reporter: [['html', { open: 'never' }]],
  webServer: [
    {
      command: 'npm run start --prefix ./backend',
      port: 9000,
      timeout: 200000,
      reuseExistingServer: true
    },
    {
      command: 'npm run start --prefix ./frontend',
      port: 8000,
      timeout: 180000,
      reuseExistingServer: true
    }
  ],
  use: {
    screenshot: "on",
    video: "retain-on-failure",
    trace: "retain-on-failure",
    baseURL: 'http://localhost:8000',
    acceptDownloads: true,
    headless: true,
    permissions: ["clipboard-read", "clipboard-write"],
    launchOptions: {
    args: ["--disable-features=IsolateOrigins,site-per-process"],
  }
  },
  tsconfig: "./playwright.tsconfig.json",
});
```

NOTE: For debugging purpose, headless property can be marked false to make browser visible.

---

5. Executing Playwright Tests

Run All Tests

cd dbk-devex-saml-sso-toolkit/sample-web-app/

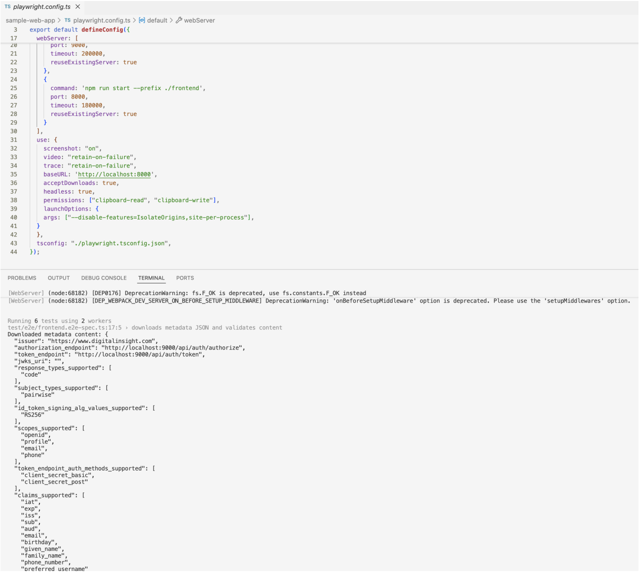
npx playwright test

Debug Tests

cd dbk-devex-saml-sso-toolkit/sample-web-app/

npx playwright test –debug

Run the project using the command in local.


TS playwright.config.ts ×
sample-web-app > TS playwright.config.ts > default > webServer
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS

```
export default defineConfig({
  webServer: [
      port: 9000,
      timeout: 200000,
      reuseExistingServer: true
    },
    {
      command: 'npm run start --prefix ./frontend',
      port: 8000,
      timeout: 180000,
      reuseExistingServer: true
    }
  ],
  use: {
    screenshot: "on",
    video: "retain-on-failure",
    trace: "retain-on-failure",
    baseURL: 'http://localhost:8000',
    acceptDownloads: true,
    headless: true,
    permissions: ["clipboard-read", "clipboard-write"],
    launchOptions: {
    args: ["--disable-features=IsolateOrigins,site-per-process"],
  }
  },
  tsconfig: "./playwright.tsconfig.json",
});
```

```
[WebServer] (node:68182) [DEP0176] DeprecationWarning: fs.F_OK is deprecated, use fs.constants.F_OK instead
[WebServer] (node:68182) [DEP_WEBPACK_DEV_SERVER_ON_BEFORE_SETUP_MIDDLEWARE] DeprecationWarning: 'onBeforeSetupMiddleware' option is deprecated. Please use the 'setupMiddlewares' option.


Running 6 tests using 2 workers
test/e2e/frontend.e2e-spec.ts:17:5 › downloads metadata JSON and validates content
Downloaded metadata content: {
  "issuer": "https://www.digitalinsight.com",
  "authorization_endpoint": "http://localhost:9000/api/auth/authorize",
  "token_endpoint": "http://localhost:9000/api/auth/token",
  "jwks_uri": "",
  "response_types_supported": [
    "code"
  ],
  "subject_types_supported": [
    "pairwise"
  ],
  "id_token_signing_alg_values_supported": [
    "RS256"
  ],
  "scopes_supported": [
    "openid",
    "profile",
    "email",
    "phone"
  ],
  "token_endpoint_auth_methods_supported": [
    "client_secret_basic",
    "client_secret_post"
  ],
  "claims_supported": [
    "iat",
    "exp",
    "iss",
    "sub",
    "aud",
    "email",
    "birthday",
    "given_name",
    "family_name",
    "phone_number",
    "preferred username"
```

Note: The headless property is set to true.

---

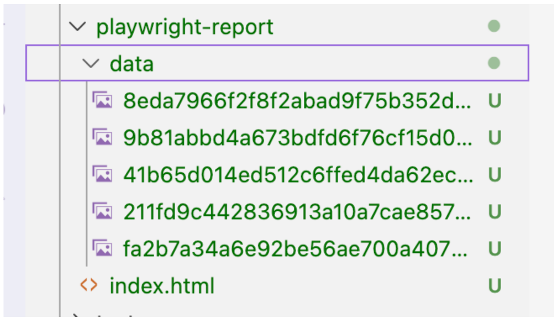
6. Reporting

HTML Report (Local)

Generated automatically after execution

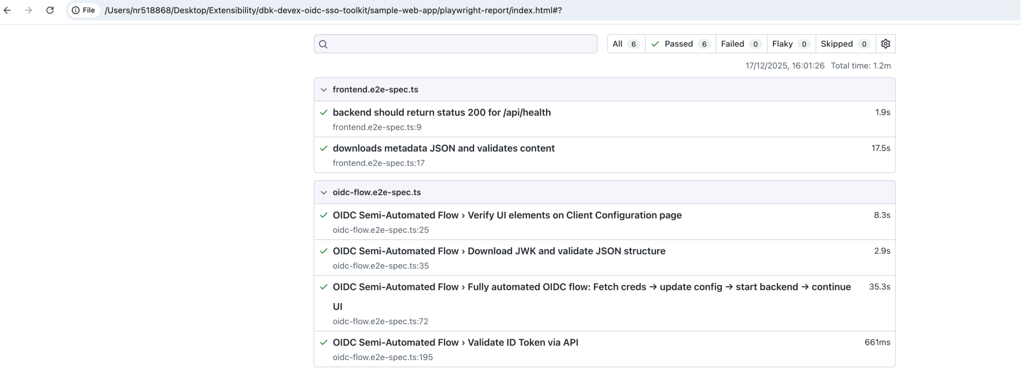
Stored in *playwright-report/*

View HTML Report - *npx playwright show-report*



playwright-report
data
8eda7966f2f8f2abad9f75b352d... U
9b81abbd4a673bdfd6f76cf15d0... U
41b65d014ed512c6ffed4da62ec... U
211fd9c442836913a10a7cae857... U
fa2b7a34a6e92be56ae700a407... U
index.html U

Report will look like this:



File /Users/nr518868/Desktop/Extensibility/dbk-devex-oidc-sso-toolkit/sample-web-app/playwright-report/index.html#?

All 6 | ✓ Passed 6 | Failed 0 | Flaky 0 | Skipped 0

17/12/2025, 16:01:26 Total time: 1.2m

frontend.e2e-spec.ts

✓ backend should return status 200 for /api/health 1.9s
frontend.e2e-spec.ts:9

✓ downloads metadata JSON and validates content 17.5s
frontend.e2e-spec.ts:17

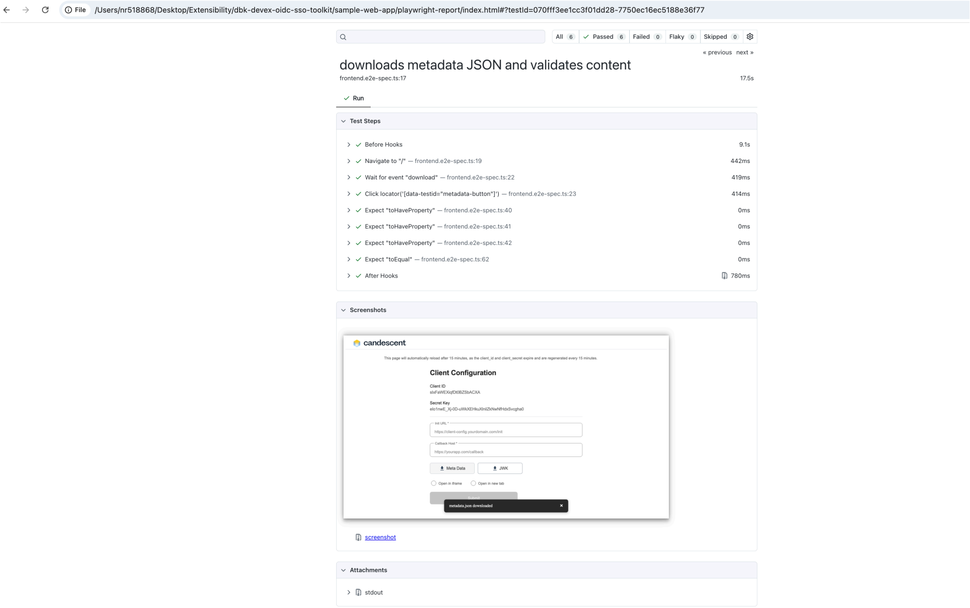
oidc-flow.e2e-spec.ts

✓ OIDC Semi-Automated Flow › Verify UI elements on Client Configuration page 8.3s
oidc-flow.e2e-spec.ts:25

✓ OIDC Semi-Automated Flow › Download JWK and validate JSON structure 2.9s
oidc-flow.e2e-spec.ts:35

✓ OIDC Semi-Automated Flow › Fully automated OIDC flow: Fetch creds → update config → start backend → continue UI 35.3s
oidc-flow.e2e-spec.ts:72

✓ OIDC Semi-Automated Flow › Validate ID Token via API 661ms
oidc-flow.e2e-spec.ts:195



File /Users/nr518868/Desktop/Extensibility/dbk-devex-oidc-sso-toolkit/sample-web-app/playwright-report/index.html#?testId=070fff3ee1cc3f01dd28-7750ec16ec5188e36f77

All 6 | ✓ Passed 6 | Failed 0 | Flaky 0 | Skipped 0

« previous next »

downloads metadata JSON and validates content
frontend.e2e-spec.ts:17 17.5s

✓ Run

Test Steps

```
✓ Before Hooks                                                              9.1s
✓ Navigate to "/" — frontend.e2e-spec.ts:19                                442ms
✓ Wait for event "download" — frontend.e2e-spec.ts:22                      419ms
✓ Click locator('[data-testid="metadata-button"]') — frontend.e2e-spec.ts:23 414ms
✓ Expect "toHaveProperty" — frontend.e2e-spec.ts:40                          0ms
✓ Expect "toHaveProperty" — frontend.e2e-spec.ts:41                          0ms
✓ Expect "toHaveProperty" — frontend.e2e-spec.ts:42                          0ms
✓ Expect "toEqual" — frontend.e2e-spec.ts:62                                 0ms
✓ After Hooks                                                              780ms
```

Screenshots

candescent

This page will automatically reload after 15 minutes, as the client_id and client_secret expire and are regenerated every 15 minutes.

Client Configuration

Client ID
slxFaWEXiqfDt0BZSbACXA

Secret Key
elo1nwE_Xj-0D-uWkXEHkuXlnilZkNwNfHdxSvcgha0

Init URL *: https://client-config.yourdomain.com/init

Callback Host *: https://yourapp.com/callback

metadata.json downloaded ×

screenshot

Attachments

stdout

7. GitHub Actions CI Pipeline

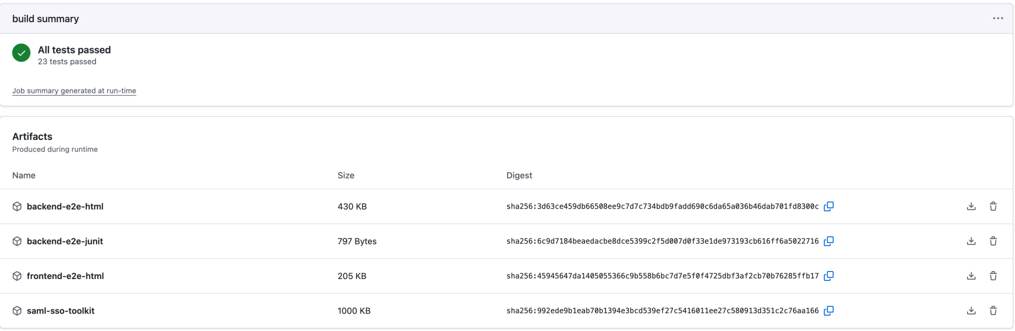
Pipeline Workflow File

```
- name: Install Playwright Browsers
  working-directory: sample-web-app
  run: npx playwright install chromium

- name: Run frontend e2e tests
  working-directory: sample-web-app
  run: npx playwright test

- name: Upload e2e HTML frontend Report
  uses: actions/upload-artifact@v4
  with:
    name: frontend-e2e-html
    path: sample-web-app/playwright-report/
```

In pipeline you can see the files in the bottom section:



build summary

✓ All tests passed
23 tests passed

Job summary generated at run-time

Artifacts
Produced during runtime

Name	Size	Digest
backend-e2e-html	430 KB	sha256:3d63ce459db66508ee9c7d7c734bdb9fadd690c6da65a036b46dab701fd8300c
backend-e2e-junit	797 Bytes	sha256:6c9d7184beaedacbe8dce5399c2f5d007d0f33e1de973193cb616ff6a5022716
frontend-e2e-html	205 KB	sha256:45945647da1405055366c9b558b6bc7d7e5f0f4725dbf3af2cb70b76285ffb17
saml-sso-toolkit	1000 KB	sha256:992ede9b1eab70b1394e3bcd539ef27c5416011ee27c580913d351c2c76aa166