

# Windows App Certification Kit - Test Results

App name:

**I, WOMAN**

App publisher:

**Occupational Hazard LLC**

App version:

**2.0.1.0**

App Architecture:

**x64**

Kit Version:

**10.0.26100.3916**

OS Version:

**Microsoft Windows 11 Home (10.0.26100.0)**

OS Architecture:

**x64**

Report time:

**5/9/2025 9:19:38 AM**

---

## Overall result: PASSED WITH WARNINGS

It is highly recommended that you investigate and root cause warnings to ensure that users of your app aren't negatively impacted.

---

### Package compliance test

PASSED	App manifest
PASSED	Enterprise Features
PASSED	Resource Packages

### Windows security features test

PASSED	Banned file analyzer
PASSED	Private code signing

## App manifest resources tests

**PASSED**

Branding

## App Capabilities test

**PASSED**

Special use capabilities

## Windows Runtime metadata validation

**PASSED**

ExclusiveTo attribute

**PASSED**

Type location

**PASSED**

Type name case-sensitivity

**PASSED**

Type name correctness

**PASSED**

Properties

## High-DPI support

**WARNING**

DPIAwarenessValidation

- **Warning:** The DPI-awareness validation test detected following Warnings:
  - File I, W.O.M.A.N\IWOMAN.exe neither has PerMonitorV2 manifested in the manifest nor calls into DPI Awareness APIs for ex:  
user32!SetProcessDpiAwarenessContext or  
user32!SetThreadDpiAwarenessContext.
  - The app OccupationalHazardLLC.IWOMAN\_2.0.1.0\_x64\_\_cge5rhfk7sw3p is not DPI Aware.
- **Impact if not fixed:** Apps that are not DPI-aware but are running on a high-DPI display setting can exhibit incorrect scaling of UI elements, clipped text, and blurry images.
- **How to fix:** It is recommended that you declare your app as DPI-aware in the app manifest. Otherwise, app should use DPI Awareness API calls for example: either SetProcessDpiAwarenessContext or SetThreadDpiAwarenessContext APIs.

[Writing High-DPI Apps](#)

---

# Results for optional tests

The tests in this section are informational only and will not be used to evaluate your app during Microsoft Store onboarding. Investigating failures is recommended to ensure that users will not be impacted.

---

## Digitally signed file test

**PASSED**

Install signed driver and executable files

## User account control (UAC) test

**PASSED**

User account control run level

## Package compliance test

**PASSED**

Application count

**PASSED**

File association verbs

**FAILED**

Registry checks

- **Error Found:** The registry checks test detected the following errors:
  - NT Services cannot be installed. Registry path REGISTRY\MACHINE\SYSTEM\CurrentControlSet\Services\bam] found.
  - NT Services cannot be installed. Registry path REGISTRY\MACHINE\SYSTEM\CurrentControlSet\Services\EventLog] found.
- **Impact if not fixed:** Apps should not install drivers or NT services.
- **How to fix:** Do not install drivers or NT services.

## App manifest resources tests

**FAILED**

App resources

- **Error Found:** The app resources validation test detected the following errors:
  - The "resources.pri" file must contain a resource map with a name that matches the package name "OccupationalHazardLLC.IWOMAN".
  - Manifest Resource Validation Test encountered error: hr = 0x80073B1F.
- **Impact if not fixed:** The app might not install if the strings or images declared in your app's manifest are incorrect. If the app does install with these errors, your app's logo or other images used by your app might not display correctly.
- **How to fix:** Fix the problem indicated by the error message. See the link below for more information:

## Debug configuration test

**PASSED**

Debug configuration

## Windows Runtime metadata validation

**PASSED**

General metadata correctness

## Package sanity test

**PASSED**

Archive files usage

**FAILED**

Blocked executables

- **Error Found:** The blocked executables test has detected the following errors:
  - File I, W.O.M.A.N\Engine\Binaries\Win64\EOSSDK-Win64-Shipping.dll contains a reference to a "Launch Process" related API kernel32.dll!CreateProcessW
  - File I, W.O.M.A.N\Engine\Binaries\Win64\EOSSDK-Win64-Shipping.dll contains a reference to a "Launch Process" related API shell32.dll!ShellExecuteW
  - File I, W.O.M.A.N\unins000.exe contains a reference to a "Launch Process" related API kernel32.dll!CreateProcessW
  - File I, W.O.M.A.N\unins000.exe contains a reference to a "Launch Process" related API shell32.dll!ShellExecuteW
- **Impact if not fixed:** Launching executable files is restricted on Windows 10 S systems. Apps that rely on this capability might not run correctly on Windows 10 S systems.
- **How to fix:** Identify which of the flagged entries represent a call to launch an executable file that is not part of your app and remove those calls. If the flagged files are part of your application, you may ignore the warning.

**PASSED**

Platform appropriate files