Tizen Web Application Checker

Xu Zhang
(xu.u.zhang@intel.com)
Agenda

• Tizen Compliance and Application Compatibility
• Tool Features
• Design Overview
• Checker Modules
• Getting Started
• Contributing to Tizen Web Application Checker
Tizen Compliance and Application Compatibility

• Tizen Compliance Goal: ensure Tizen device implementations and applications work together.

• Tizen Compliance Specification (TCS) is a set of requirements for devices and applications to ensure that they work together smoothly.

  TCS can be found here: https://source.tizen.org/compliance

• Tizen Application Requirements:
  • API USE
  • Application Packaging
  • Security
Why We Need Tizen Web Application Checker

Develop | Debug | Package | Distribute

Tizen SDK
- Emulator
- Simulator

.Tpk

Submit

Store Validation
- reject
- accept

Tizen App Store

• Developers need to know how to create compatible applications
• Developers need a tool to provide guidance during development of Tizen application.
• App Store needs a tool to provide to assist in screening submitted Tizen applications before acceptance
Tool Features

- Checks Tizen web application using rules defined in TCS
- Identifies incompatible behavior and provides a clear report
- Runs on Linux/Windows/Mac environment
- Implemented with Node.js

Difference from other JS analyzing tool
- Checks JavaScript syntax as well as Tizen package and security risks
- No JS code style checking as JSLint
- Does not stop at the first issue like the closure compiler. Reports all possible issues
- Checks Tizen Web APIs usage
Design Overview

UX

CLI

Report (HTML/XML/JSON)

Factory Engine

Package Checker

API Checker

Security Checker

Third Part Libraries

Esprima
CSSOM
Zipfile
Commander

elementtree
HtmlParser
XmlCrypto
...

Services

Utils

Error handler
Factory engine
Report template

DB

Web API DB
Schema of config.xml
Checker Modules – Packaging

- Ensure package format is compliant with Tizen standards (W3C Widget)
  - Validate the given package is a valid WGT package

<table>
<thead>
<tr>
<th>TestName</th>
<th>Status</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>zip validity</td>
<td>FAILED</td>
<td>Error The given package is not complete.</td>
</tr>
<tr>
<td>Checking whether</td>
<td></td>
<td></td>
</tr>
<tr>
<td>the file is a</td>
<td></td>
<td></td>
</tr>
<tr>
<td>complete zip</td>
<td></td>
<td></td>
</tr>
<tr>
<td>archive.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Validate the content of a given WGT package

<table>
<thead>
<tr>
<th>TestName</th>
<th>Status</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>package elements attributes</td>
<td>FAILED</td>
<td>Error About the attribute &quot;name&quot; of the element &quot;tizen:privilege&quot;,</td>
</tr>
<tr>
<td>Checking whether attributes</td>
<td></td>
<td>its value &quot;<a href="http://tizen.org/privilege/calandar.read">http://tizen.org/privilege/calandar.read</a>&quot; is invalid.</td>
</tr>
<tr>
<td>of elements in config.xml are</td>
<td></td>
<td>in file config.xml, line 12</td>
</tr>
<tr>
<td>valid.</td>
<td></td>
<td>Evidence:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&lt;tizen:privilege name=&quot;<a href="http://tizen.org/privilege/calandar.read%22/%3E">http://tizen.org/privilege/calandar.read&quot;/&gt;</a></td>
</tr>
</tbody>
</table>
Checker Modules – Security

- Ensure no violation in widget access request policy, API privilege use, and security signature

<table>
<thead>
<tr>
<th>TestName</th>
<th>Status</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTML resource validation</td>
<td>FAILED</td>
<td>Error 'index.html' requesting access to '<a href="http://google.com">http://google.com</a>' is not allowed: no origin area matches. in file index.html, line 22, column 32</td>
</tr>
<tr>
<td>CSS resource validation</td>
<td>PENDING</td>
<td>Warning CSS file 'css/style.css' requires access to '../images/noexist.png', but it cannot be found in the package.</td>
</tr>
<tr>
<td>authorized use of API</td>
<td>FAILED</td>
<td>Error Current privilege level is 'public', but '<a href="http://tizen.org/privilege/appmanager.kill">http://tizen.org/privilege/appmanager.kill</a>' need 'partner' privilege level. Warning Privilege '<a href="http://tizen.org/privilege/calendar.read">http://tizen.org/privilege/calendar.read</a>' is declared in config.xml, but app neen't '<a href="http://tizen.org/privilege/calendar.read">http://tizen.org/privilege/calendar.read</a>'</td>
</tr>
<tr>
<td>integrity of package files</td>
<td>FAILED</td>
<td>Error The author has signed file 'invalid.html', but it can't be found in the package. Error File 'app/app.js' is included in the package, but the author hasn't signed it.</td>
</tr>
<tr>
<td>author signature</td>
<td>FAILED</td>
<td>Error Hash error happens on 'config.xml'.</td>
</tr>
</tbody>
</table>
Checker Modules – API Validation

- Ensure all APIs used are supported on Tizen compliant systems

<table>
<thead>
<tr>
<th>Test Name</th>
<th>Status</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>access of used api</td>
<td>FAILED</td>
<td><strong>Error</strong> API 'tizen.bluetooth.getDefaultAdapter' is not allowed to be used, according to the 'tizen:privilege' list in the config.xml file. In file js/main.js, line 476, column 18</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Evidence: <code>this.adapter = tizen.bluetooth.getDefaultAdapter();</code></td>
</tr>
<tr>
<td>usage of api</td>
<td>PENDING</td>
<td><strong>Warning</strong> Type error on symbol 'tizen.alarm.PERIOD_HOUR', maybe used attribute as a method or called abstract class, etc.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>in file js/main.js, line 518</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Evidence: <code>new tizen.alarm.PERIOD_HOUR();</code></td>
</tr>
<tr>
<td>build-in namespace</td>
<td>PENDING</td>
<td><strong>Warning</strong> The existing API 'tizen.alarm' has been overwritten.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>in file js/getMessage.js, line 202, column 8</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Evidence: <code>tizen.alarm = {};</code></td>
</tr>
</tbody>
</table>
Checker Modules – API Validation

Known Issues

```javascript
// no define of "test" before
if (condition) {
    var test = "hello.World";
    var person = function () {...};
} else {
    test.split(".");
    person = "a string";
}
test.indexOf("o");
p = new person;
```

```html
"<div id="footer"/>"> // in HTML
footer.setAttribute(); // in JavaScript code
```

```javascript
document.getElementById('touchable').addEventListener('touchend', function(ev){
    document.write('Remaining on element: ' + ev.targetTouches.length);
    // other usage of argument ev
    ....
}, false);
```
Checker Modules – Reporting

- Three types of reports: HTML, XML, and JSON. HTML format is recommended.
- XML and JSON data can be used by other tools to generate your preferred report. Useful for automation.
## Checker Modules – Reporting

### Report Description

<table>
<thead>
<tr>
<th>Category</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test Names</td>
<td>Lists all checked items with their names and descriptions</td>
</tr>
</tbody>
</table>
| Status    | • PASS: No problems are found for the corresponding items.  
• FAIL: Problems were found by the checker.  
• PENDING: There are warnings. The issue is suspicious but may not be an error. The Tizen Web App Checker cannot make a definitive decision and suggests the user manually double check it. |
| Reason    | List all issues checked. Each item is a link leading to the details of the test run and results.                                             |
## Checker Modules – Reporting

### Report Filtering

<table>
<thead>
<tr>
<th>Option</th>
<th>Report Filter</th>
</tr>
</thead>
<tbody>
<tr>
<td>Error</td>
<td>Only show problems marked “Error”</td>
</tr>
<tr>
<td>Warning</td>
<td>Only show problems marked “Warning”. The developer should check each of these to verify if it is an error.</td>
</tr>
<tr>
<td>All</td>
<td>Show all items in all modules (Package, API, and Security)</td>
</tr>
<tr>
<td>Package</td>
<td>Only show items in the Package module</td>
</tr>
<tr>
<td>API</td>
<td>Only show items in the API module</td>
</tr>
<tr>
<td>Security</td>
<td>Only show items in the Security module</td>
</tr>
</tbody>
</table>
Getting Started

Step 1: Set Up Environment

1. Download Python and install it. Set python in your system variables (Windows Only)
3. Test to make sure you can use the Node.js and Python.

Step 2: Download Tizen Web Application Checker

• Download from https://source.tizen.org/compliance
Getting Started (Cont.)

Step 3: Install on your development PC

- Extract the package to your working directory
- Make sure Node.js and Python have been set in your system variables.

Step 4: Run Your Application

- Open a terminal and run command below
  
  $ node ./bin/tizen-web-app-checker.js -a Tizen22Demo.wgt

- More helpful usage can be found by running
  
  $ node ./bin/tizen-web-app-checker.js --help

- More options are provided as table
## Command-line Options

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>h, --help</td>
<td>Outputs usage information</td>
</tr>
<tr>
<td>-V, --version</td>
<td>Outputs the version information</td>
</tr>
<tr>
<td>-l, --log &lt;log&gt;</td>
<td>Specifies the log file path</td>
</tr>
<tr>
<td>-t, --type &lt;HTML</td>
<td>XML</td>
</tr>
<tr>
<td>-r, --report &lt;report&gt;</td>
<td>Specifies report file path</td>
</tr>
<tr>
<td>-v, --verbose</td>
<td>shows verbose information in the console</td>
</tr>
<tr>
<td>-s, --spec &lt;2.1, 2.2&gt;</td>
<td>Specifies the Tizen API version used to check compliance. Tizen 2.2 is used by default.</td>
</tr>
<tr>
<td>-a --apiCheck</td>
<td>Enables the API checker. This is an experimental feature</td>
</tr>
<tr>
<td>-p, --privLevel &lt;public</td>
<td>partner</td>
</tr>
</tbody>
</table>
Getting Started (Cont.)

Step 5: Modify your application according to generated report if needed

More details online in User Guide:

https://source.tizen.org/compliance/application-compliance/web-application-checker-user-guide
Contribute to Tizen Web Checker Tool

• Open source project initiated by Intel Open Source Technology Center
• Apache 2.0 license
• Source code
  git clone ssh://username@review.tizen.org:29418/sdk/tools/web-app-checker.git
  Note: before getting source code, please read Work Flow and Developer Guide to configure your environment

• Next Steps
  • Integrate into Tizen SDK IDE
  • Support to check the Tizen UI Framework API
Questions?