# Tizen Debugging and Testing Tools

## Important concepts

All Tizen application developed with Titanium really are Web Applications(widgets)

Developer may run widget in one of the following environments:

* **Simulator** - The Tizen Web simulator allows you to run widgets that use the Tizen Web APIs.
* **Emulator** - The device Emulator, provided with the Tizen SDK, imitates the target environment running Tizen Web applications. Using this replicated environment, you can test your widget before deploying it to the real target device.
* **Target Device** - Running your widget on a target device allows you to debug and test your widget in a real-time environment.

## Required Software and Documentation

It is required to have Tizen SDK installed to use Tizen tools. Emulator, Simulator and Emulator Manager applications comes with Tizen SDK.

The Tizen Web Simulator runs on the Google Chrome browser. To use the Tizen Web Simulator you must first download and install Google Chrome on your development platform: http://www.google.com/chrome/

All Tizen specific documentation for these tools available on Tizen web site in Dev Guide document in [Running and Debugging Applications](http://developer.tizen.org/help/topic/org.tizen.web.appprogramming/running_debugging.htm) Section.

## Installing Tizen development Environment

#### Download Tizen SDK 2.0

Download Install Manager and Image at: <https://developer.tizen.org/sdk>

For Windows 32bit

<http://download.tizen.org/sdk/InstallManager/tizen-sdk-2.0-windows32.exe>

<http://download.tizen.org/sdk/sdk-images/tizen-sdk-image-2.0.0a-windows32.zip>

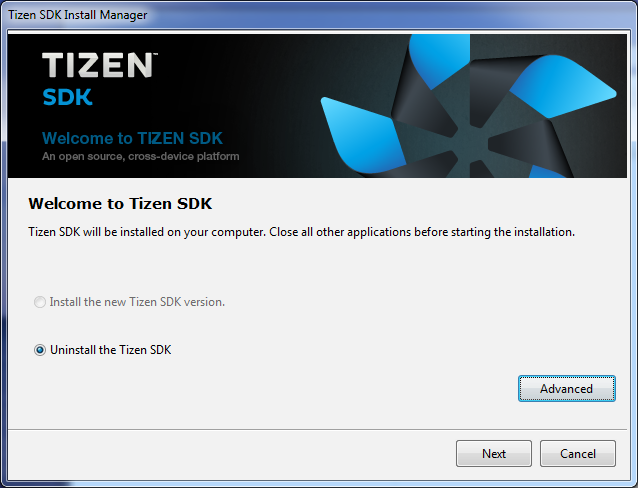
For Windows 64bit

<http://download.tizen.org/sdk/InstallManager/tizen-sdk-2.0-windows64.exe>

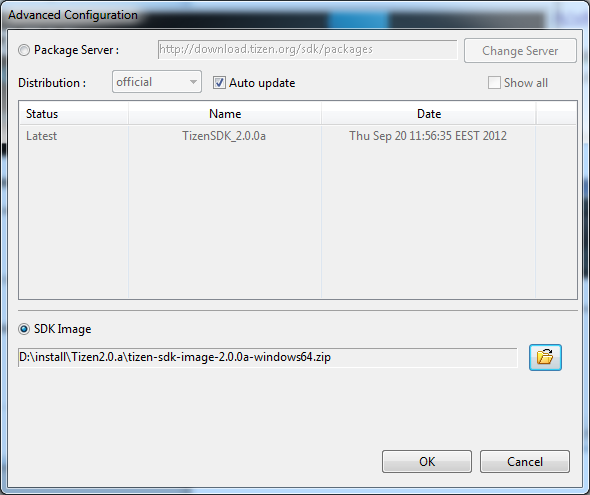
<http://download.tizen.org/sdk/sdk-images/tizen-sdk-image-2.0.0a-windows64.zip>

#### Install Tizen SDK

Run tizen-sdk-2.0-windows64.exe and click button ***Advanced***



In the next window select radio button “SDK Image”



And select downloaded tizen-sdk-image-2.0.0a-windows64.zip . Click Ok.

SDK installation starts. Use default settings for all next steps and go throw installation process. As result Tizen SDK installed into ***C:\tizen-sdk*** .

## Tizen CLI Tools from SDK

### Check is emulator available

It is important to check is there are connection between developer tools and emulator.

In console run C:\tizen-sdk\tools\ide\bin\web-list.bat

It should shout that one emulator instance available if there is running Emulator.

Note that it freezes. In this case you have to restart thing called “Samsung Debug Bridge”. It is just a bit modified Android tool originally called adb. To restart execute sdb located in ***C:\tizen-sdk\tools\:***

***sdb kill-server***

***sdb start-server***

***sdb devices***

First two commands restarts service. Last one shows list of available devices/emulators after successful restart.

## Build and run Titanium Application for Tizen

### Titanium CLI for Tizen

Tizen is platform as well as iphone or android. The easy way to build Titanium application is use CLI build command and specify platform:

**titanium build --platform=tizen**

Also Tizen have additional command line options required to sign the application with developer keys:

|  |  |
| --- | --- |
| --cert | path to certificate file |
| --alias | the alias for the keystore |
| --password | password for keystore |
| --keypass | password for the key |

Without these parameters it uses default debugging certificate that comes with TizenSDK.

## Emulator

### Install Intel HAXM for Tizen

Without HW Acceleration Tizen emulator performance on Windows are far away from excellence even with latest Core i7 processor. It is mandatory configure hardware acceleration. Go to page <https://developer.tizen.org/downloads/sdk/installing-sdk/hardware-accelerated-execution-manager>

And download and install [IntelHaxmTizen.exe](http://download.tizen.org/sdk/haxm/beta/IntelHaxmTizen.exe)

HAXM for Tizen conflicts with HAXM for Android. Uninstall Android version first and rrebnoot.

## Target Device

Work with target device is fully similar to work with emulator. If there are no working emulator and deveic

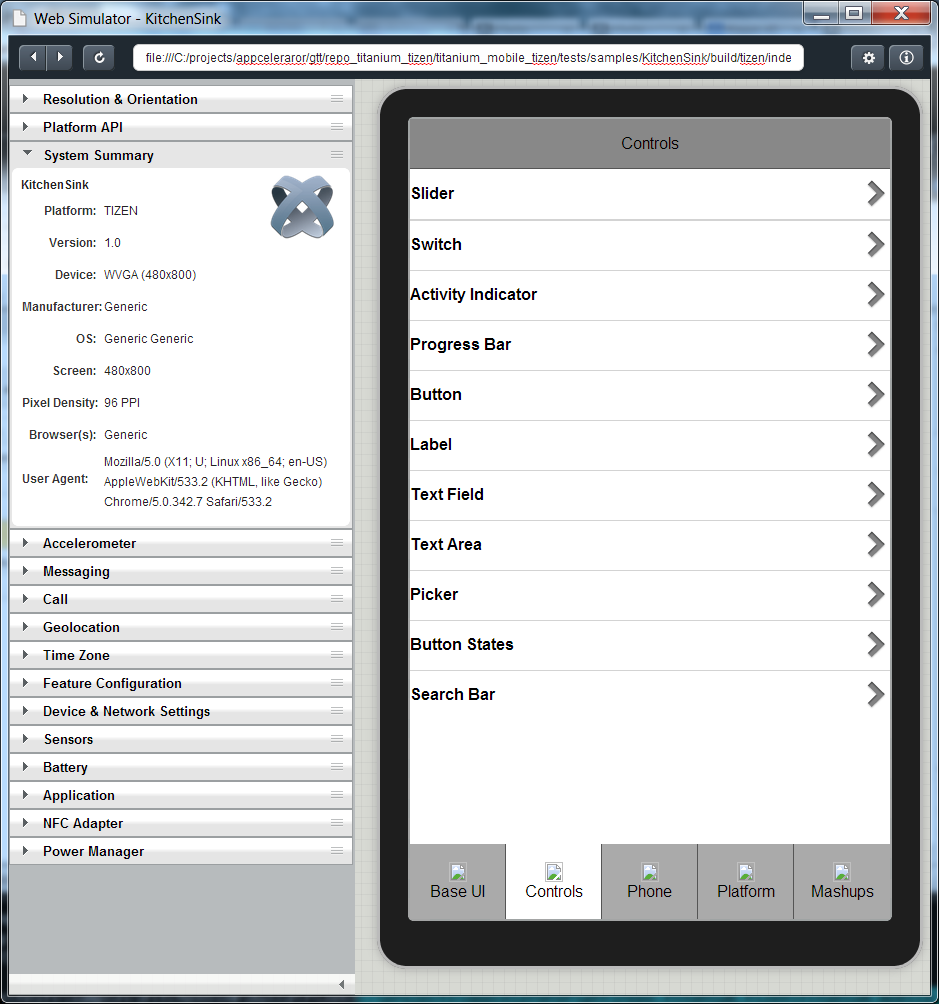
Device drivers required for work with concrete device are required. E.g. for Samsung devices developer have to install Samsung USB driver available here <http://down2.local.sec.samsung.com/uploadimg2/comLocal/service/support/consumer/download/SAMSUNG_USB_Driver_for_Mobile_Phones.zip>

**Launch your application in the Simulator**

Open Tizen Web Simulator via application menu (in Windows).

<file:///C:/projects/appceleraror/gtt/repo_titanium_tizen/titanium_mobile_tizen/tests/samples/KitchenSink/build/tizen/index.html>

KitchenSink application in the simulator



When the simulator is started from the Tizen IDE, the path to your application's main content file is loaded into the application address bar. You can change which file or project is being run by editing this field. But we using CLI tools to build project and have to enter this path manually. It is easy to find this path:

<YOUR\_TITANIUM\_PROJECT>/build/tizen/index.html

Then run Web Simulator manually and copy the path into address bar.

More information about Web Simulator features available on [Tizen documentation site](https://developer.tizen.org/help/index.jsp?topic=%2Forg.tizen.web.appprogramming%2Fweb_simulator_overview.htm).

## Using Tizen IDE to Debug Titanium Applications

It is possible to install, run and debug Titanium application with Tizen IDE (available as part of Tizen SDK). It may make sense if developer needs some features not available from Titanium CLI. E.g. It allow us use event inspector

## Debugging a Widget

Developers can debug a widget by debugging its JavaScript code while running it in the target. JavaScript debugging uses the Remote Inspector connected with the WebKit on the target. There are no any Tizen specific tips for Remote Inspector

web-debug command from SDK available at <TIZEN\_SDK>\tools\ide\bin initiates debug session. Minimal required parameters are path to wgt file and widget ID

**web-debug.bat -w "path to wgt" -id <ID>**

On successful application start it shows debug URL in console, e.g.:

DEBUG URL : http://localhost:57283/inspector.html?page=1

Open this URL to get access to Remote Inspector used to debug the application.

CAUTION: Remote Inspector doesn't support multiple debugging.

CAUTION: avoid use spaces in application path and wgt file name. It does not works on Beta version available now.