

The Ultimate Toast Extension For Android

A GameMaker Studio 2 Android Extension

About

Thank you for downloading the "Ultimate Toast Extension for Android"! If you downloaded the free version, please consider buying the full version to support this project and its future updates. If you have already bought the full version, thank you very much for your support, and I hope this extension to match your awesomeness!

The "Ultimate Toast Extension for Android" is compatible with GameMaker Studio 2.

This extension allows your GameMaker Studio 2 project, to show from simple to fully customizable toast messages natively on android devices. **It is very easy to use, with very simple API.** It is written with Java and it is lightweight and natively fast.

Prerequisites

This extension is for android devices that are able to deliver toasts, therefore the GameMaker Studio 2 Mobile Module is required.

Getting Started

Depending on the source you got this extension from, the installation procedure is different.

Downloading and installing from Yoyo Games Marketplace

You need to log in to your Yoyo Games account and on "Ultimate Toast Extension for Android" store listing page press the *Add to Account (Free)* or the *Buy* button, depending on which version of the extension you are browsing to. This procedure can be done either from your browser or from the build in Marketplace browser within the GameMaker Studio 2. Then the extension it should be visible and available to download from the Marketplace -> My Library on GameMaker Studio 2 menu. Alternatively you can download the extension from the browser. The *Add to Account (Free)* or the *Buy* button will change to *Download for Studio 2* after adding it into your library. Download the extension and then on GameMaker Studio 2 menu choose Tools -> Import Local Package.

Downloading and installing from itch.io

Unlike Yoyo Games Marketplace, the free and the full version of this extension can be found on the same page. You can download the free version with an optional small donation to support this project or go for the full version with minimum price or more if you feel generous. After downloading the extension open the GameMaker Studio 2 and choose Tools -> Import Local Package.

NOTE: *If you are going to use this extension for first time, it is highly advised to import to an empty project or create a new project when importing. If you are going to import the extension into an existing project, make sure to import the core element of the extension and omit the sprites, the objects and the room that are for example and demonstration purposes.*

The core elements of the extension are:

- The **script toast_constants** in the Toast Scrip folder
- The extension **TheUltimateToastExtensionForAndroid** in extensions folder
- The documentation in **Notes** and **Included Files** folder

Free vs Full Version

Full Version

The full version of the extension includes everything this extension has to offer plus all the possible future updates. You will be able to show from the simple default toasts to fully customizable toasts. The fully customizable toasts can include an image, background and font custom colors. You will be able to show the toast at any size and wherever you want on the screen. Additionally you will be able to force stop a toast for showing or your next toast to override the previous one if is still showing. The full version is supported by the developer and he will help you resolve any issue that you might have with the extension; exceptions apply*.

** Any additional toast customization or functionality that may or may not supported by official Android Developers documentation but not mentioned here, **is not supported by the extension developer**, but you can provide feedback that may result to customization additions in future updates.*

Free Version

The free version only offers the simple `android_show_toast` method, which has 2 fixed default display duration. Additionally, you can enable debugging logs on free version. The extension developer does not offer any support to free version users.

How to use

`android_enable_toast_logs`

To enable the debug logs use the following method:

```
android_enable_toast_logs();
```

Now you will be able to see the debug log in the output window of GameMaker Studio 2. To disable the debug logs simply remove or comment this line of code. Also, you will see a confirmation that the debug logs are enabled in the output window as soon as the compilation is over.

NOTE : *You might want to disable the debug logs before building the final .apk or .aab file.*

android_show_toast

The simplest way to show a toast is by the following method:

```
android_show_toast(msg, duration);
```

The msg argument accepts your toast message as string and the duration argument accepts an integer of value either 0 or 1. Passing the value of 0 will set the toast duration to 2 seconds, which is the default native short duration and passing the value of 1 will set the duration to 3.5 seconds, which is the default native long duration. You can also set those values via the following enumerated constants:

```
Toast.LENGTH_SHORT  
Toast.LENGTH_LONG
```

NOTE : *If you pass any value other than 0 or 1, it will be ignored and the Toast.LENGTH_SHORT will be set instead.*

Simple toast examples:

```
android_show_toast(" Hello from GML!\n(short toast 2 sec)",  
    Toast.LENGTH_SHORT);  
android_show_toast(" Hello from GML!\n(long toast 3.5 sec)",  
    Toast.LENGTH_LONG);
```

NOTE : *Beyond this point, the information is about for the full version only.*

android_show_toast_ext

If you need a more customizable toast, then you should use the following method:

```
android_show_toast_ext(msg, duration, gravity1, gravity2, h_offset, v_offset);
```

The first two arguments are the same as the simple toast. The third and fourth arguments gravity1 and gravity2, are customizing the position of the toast on device screen. This extension supports 27 enumerated gravity constants such as:

```
Gravity.NO_GRAVITY  
Gravity.TOP  
Gravity.BOTTOM  
Gravity.LEFT  
Gravity.RIGHT  
Gravity.CENTER
```

You can combine up to two gravity constants via the arguments gravity1 and gravity2:

```
android_show_toast_ext("An extended toast without gravity", Toast.LENGTH_SHORT, Gravity.NO_GRAVITY, Gravity.NO_GRAVITY, 0, 0);
android_show_toast_ext("Extended toast TOP", Toast.LENGTH_SHORT, Gravity.TOP, Gravity.NO_GRAVITY, 0, 0);
android_show_toast_ext("Extended toast BOTTOM", Toast.LENGTH_SHORT, Gravity.BOTTOM, Gravity.NO_GRAVITY, 0, 0);
android_show_toast_ext("Extended toast TOP / LEFT", Toast.LENGTH_SHORT, Gravity.TOP, Gravity.LEFT, 0, 0);
android_show_toast_ext("Extended toast WIDE", Toast.LENGTH_SHORT, Gravity.LEFT, Gravity.RIGHT, 0, 0)
```

For a complete list of the gravity constants and their description, see the official documentation of Android Developers [Gravity](#).

With the last two arguments `h_offset` and `v_offset`, you can fine tune the position of the toast in pixels. In the following example, the toast will be positioned to the top left of the screen and then it will be repositioned by 167 pixels horizontally to the right and 324 pixels vertically to the bottom:

```
android_show_toast_ext("Extended toast with offset", Toast.LENGTH_SHORT, Gravity.LEFT, Gravity.TOP, 167, 324);
```

android_show_toast_custom

You can show a fully customized toast with a custom image and custom colors with the following method:

```
android_show_toast_custom(msg, duration, gravity1, gravity2, h_offset, v_offset);
```

The arguments and the usage of `android_show_toast_custom` are identical to `android_show_toast_ext` method and additional customizations can be achieved via the xml file `custom_toast.xml`. The `custom_toast.xml` can be located to the following path:

```
[Project folder]\extensions\AndroidToast\AndroidSource\res\layout
```

To change the background color of the toast, change the hex value of the following line:

```
android:background="#F00"
```

To change the font color of the toast, change the hex value of the following line:

```
android:textColor="#FFF"
```

For more information about colors check the official Android Developers documentation: [Color](#)

Note : To quickly go to the extension's path, in GameMaker Studio 2, right click the name of the extension on the *recourses* tab and choose *Open in Explorer* from the pop up menu.

The custom image of the toast `toast_img.png`, can be located to the following path:

```
[Project folder]\extensions\AndroidToast\AndroidSource\res\drawable
```

You can replace the custom toast image toast_img.png by replacing with your own image. If you want to provide an image with different file name you should change the following line in custom_toast.xml accordingly:

```
android:src="@drawable/toast_img"
```

Eg if your custom image is named my_custom_toast_image.png then you should change the following line in custom_toast.xml to:

```
android:src="@drawable/my_custom_toast_image"
```

Note : *You should not include the file extension .png in the string value.*

For more information about accessing images via xml check the official Android Developers documentation: [Bitmap](#)

Note : *Image with bigger dimensions or wider aspect ratio may interfere to the toast message visibility.*

Note : *Any additional xml customization that may or may not supported by official Android Developers documentation but not mentioned here, **is not supported by the extension developer**, but you can provide feedback that may result to customization additions in future updates.*

Note : *Certain functionality of the `android_show_toast_custom` has deprecated in Android 11 (API 30). When the `android_show_toast_custom` is called, if version of the API is 30 or greater, an `android_show_toast_ext` with the same arguments will called instead.*

Additional resources related to `android_show_toast_custom`

Extra useful resources by the official Android Developers documentation:

[Layout resource](#)

[Linear Layout](#)

[ImageView](#)

[TextView](#)

android_set_toasts_as_cancelable

Sets all toasts as cancelable:

```
android_set_toasts_as_cancelable();
```

By calling this method, all the toasts at this point of your code flow and beyond, will be able to be canceled. If your toasts are cancelable then the next toast will cancel the previous toast, if the latter is still on the screen.

android_set_toasts_as_uncancelable

Sets all toasts as uncancelable

```
android_set_toasts_as_uncancelable();
```

By calling this method, , all the toasts at this point of your code flow and beyond, will remove the ability to cancel the toasts, which is the default and indented behavior. If you toasts are uncancelable, then next toast will wait to the previous toast to finish before showing, if the latter is still on the screen.

android_cancel_toast

Cancels the toast that currently is showing on screen, if exists.

```
android_cancel_toast();
```

By calling this method, you can cancel any toast that currently is showing on the screen, even is the `android_set_toasts_as_cancelable` method has not called. If no toast is showing when this method is called, it will be ignored.

Additional information and tips

The Toast Script

- This extension contain a single script named `toast_constants`; it contains the declaration of the enumeration constants, therefore it is not necessary to call it at any part of your code.
- Avoid to spam many toasts at the same time. It will possibly mess the things up at native level of the device.
- The default appearance of the toasts varies depending the android version and the device manufacturer.

Cross Platform Compilation

When debugging you project on a desktop computer or if you are targeting cross platform, it is a good idea to contain the extension related code under conditionals:

```
if os_type != os_android then android_show_toast(" Hello from GML!\n(short  
toast 2 sec)", Toast.LENGTH_SHORT);
```

Even though the GameMaker Studio 2 compiler will probably ignore this extension when building for a non android platform, it is highly recommended to follow this practice.