

# GameMaker Firebase Analytics extension

This is the extension that I created to integrate Firebase analytics into our game. It provides quite limited functionality - stuff I only needed for my game.

## Firestore version:

- Android: 16.0.8
- iOS: 5.18.1

**Note:** as for 15 July 2019 latest Firebase android version is 16.0.9 but it can not be used because it requires java 8 support and GameMaker does not support java 8.

**Platforms supported:** Android, iOS

## Installation instructions

In order to make this extension work, you need to download `google-services.json` for Android and `GoogleService-Info.plist` for iOS from your Firebase console. See instructions here:

<https://firebase.google.com/docs/android/setup>

<https://firebase.google.com/docs/ios/setup>

Then you need to put `google-services.json` into `runtime-x.y.z.bbb\android\runner\ProjectFiles` folder to use Firebase on Android.

And put `GoogleService-Info.plist` to `iOSSource` folder of this extension to use Firebase on iOS.

Where "runtime-x.y.z.bbb" it's a current runtime than GameMaker uses. It's default location is

C:\ProgramData\GameMakerStudio2\Cache\runtimes in Windows

Users\Shared\GameMakerStudio2\Cache\runtimes on MacOS

After these preparations Firebase extension should work with your project.

## Extension reference

- `Firestore_logEvent(eventName)`

log analytics event without parameters

- *Firebase\_logEventParamString(eventName,paramName,paramValue)*

log analytics event with one string parameter

- *Firebase\_logEventParamsStringInt(eventName,param1Name,param1Value,param2Name,param2Value)*

log analytics event with two parameter. First parameter is string, second - integer

- *Firebase\_logEventParamsIntBoolean(eventName,param1Name,param1Value,param2Name,param2Value)*

log analytics event with two parameter: integer and boolean

- *Firebase\_logEventParamsStringIntIntFloatBoolean(eventName,param1Name,param1Value,param2Name,param2Value,param3Name,param3Value,param4Name,param4Value,param5Name,param5Value)*

log analytics event with five parameter: string, integer, integer, float and boolean

- *Firebase\_logLevelStart(levelName)*

log Firebase build-in event "level\_start" with provided build in parameter "level\_name"

- *Firebase\_logLevelEndParamsIntIntFloatBoolean(levelName,param1Name,param1Value,param2Name,param2Value,param3Name,param3Value,param4Name,param4Value)*

log Firebase build-in event "level\_end" with four parameters: integer, integer, float, boolean

- *Firebase\_logEarnVirtualCurrency(currencyName,amount)*

log Firebase build-it event "earn\_virtual\_currency" with build-in parameters "virtual\_currency\_name" and "value"

- *Firebase\_logSpendVirtualCurrency(currencyName,itemName,amount)*

log Firebase build-in event "spend\_virtual\_currency" with build-in parameters "virtual\_currency\_name", item\_name and "value"

- *Firebase\_setUserProperty(propertyName,propertyValue)*

Set user property. See documentation here:

<https://firebase.google.com/docs/analytics/android/properties?authuser=1>

The functionality of this extension is quite limited since I've created it for internal use but you can freely modify and extend it.

To modify it for Android get Android Studio project from this repository:

<https://bitbucket.org/lazybonesgames/firebaseandroidextension/overview> and edit it as you

wish. After you edit source java files in Android Studio you need to put them to AndroidSource directory in extension folder. Before this you need to comment line started with /\*-\*/ and uncomment lines started with //+

To modify extension for iOS just compile your project into XCode and then edit objective-c source files within XCode project. After that just put those modified \*.h and \*.m files back into iOSSource directory in extension.

## Release notes

### 0.1.2

- Added function `Firestore_logEventParamLong`
- Updated Firebase SDK
  - iOS 5.18.1 -> 6.2.0

### 0.1

- Added `Firestore_setUserProperty()` function

### 0.0.1

- Initial release

With any questions write to [lazybones.making.games@gmail.com](mailto:lazybones.making.games@gmail.com)