



FREE PLUGIN **CRASH LYTICS** FOR UNREAL ENGINE 4



1. License

The MIT License (MIT)

Copyright (c) 2017 gameDNA

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

2. Introduction

Crashlytics integrates out-of-the-box automatic crash reporting solution for **iOS & Android** into an Unreal Engine 4 project. Thanks to it you can spend less time on finding and more time on fixing crashes. Crashlytics provides deep and actionable insights, a real-time crash reporting solution, information about the number of crashes and affected users, the exact line of code game crashed on, and notifications about crashes via e-mail. Plugin includes all needed tools like symbols uploader and application configurator.

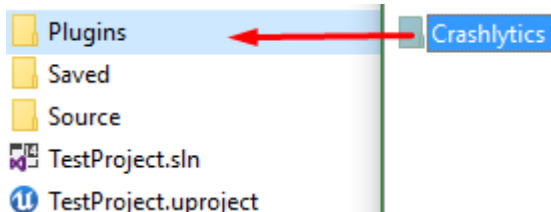
FEATURES:

- Real-time crash reporting tool
- Information about the number of crashes and affected users
- Provides the exact line of code your app crashed on
- Notifications about crashes via e-mail
- Works with dev & shipping builds
- Out-of-the-box for iOS & Android
- Additional SDK included
- Works with Blueprint-only & source code projects
- Works with Launcher & GitHub UE4 versions

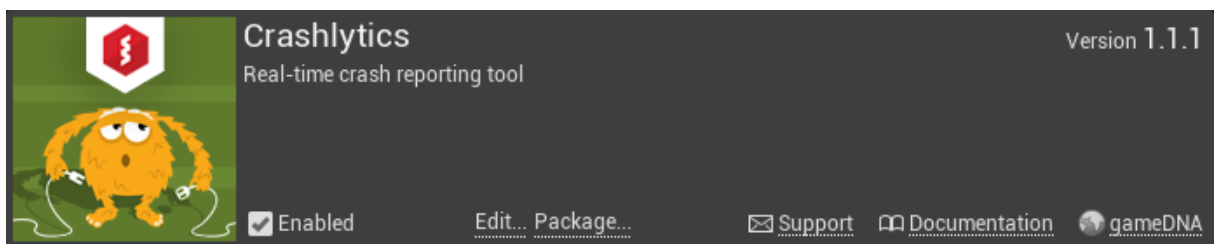
Current plugin version: 1.4.0

3. Getting started

1. Copy this folder to the *Plugins* folder located in the main path of your project.



2. Enable *Crashlytics* in *Edit -> Plugins -> Misc -> Crashlytics* .



3. Go to *Edit -> Project Settings -> Crashlytics* and set your *Crashlytics* API Key. You can obtain it on fabric.io
4. Turn on *Generate dSYM bundle for third party crash tools* in *Project Settings*.
5. Go to `[PROJECT_FOLDER]/Build/Android/`
 - a) If you already have created *custom_rules.xml* file, simply add:

```
<import file="crashlytics_build.xml" />
```

- b) If you don't have created *custom_rules.xml* file yet, copy it from `[PLUGIN_FOLDER]/Extras/Android/custom_rules.xml`.

4. Uploading symbols

You need to upload symbols in order to properly handling crashes. You should do this after every submission to App Store or Google Play.

iOS

1. Find your Xcode project:
 - a. If you are directly packaging on Mac, go to `/[PATH_TO_PROJECT_ON_MAC]/Intermediate/ProjectFilesIOS/`
 - b. If you are packaging with remote connection from Windows to Mac, go to `/Users/[USER_NAME]/UE4/Builds/[COMPUTER_NAME]/[PATH_TO_PROJECT_ON_WINDOWS]/Binaries/IOS/`
2. Copy content from `[PLUGIN_FOLDER]/Extras/iOS` to path from previous point. Now you should have *Crashlytics* and *Fabric* folders in folder from previous point.
3. Run `[PROJECT_NAME].xcodeproj` or `UE4_FromPC.xcodeproj`
4. In the *Project Navigator*, click on your project and add a new run script build phase
5. Add following code to new run script build phase with replacing `[API_KEY]` and `[BUILD_SECRET]`

```
./Fabric/run [API_KEY] [BUILD_SECRET]
```
6. Build your project by clicking on *Product -> Build*
7. Go to *Fabric dashboard -> Settings -> Apps -> Your game -> Missing DSYMs*. Here upload new DSYMs from `[PROJECT_FOLDER]/Binaries/IOS`.

Android

1. Add new *ant* system variable.
2. Go to `[PATH_TO_PROJECT]/Intermediate/Android/APK/`
3. Open *Command Prompt (Terminal)* and run following command:

```
ant crashlytics-symbols
```