**Getting Started with Mikros Unity SDK**

**Introduction**

This document is provided to demonstrate MIKROS SDK capabilities and explain the MIKROS service.

**What is MIKROS?**

MIKROS is a service that aims to provide a wide functional set. Users can view and track real-time statistics. MIKROS also offers the ability to monitor traffic sources, track where users come from, and even predict the likelihood any particular user would have an interest in your product based on gaming history. There are a lot of other features offered by MIKROS as a service, including direct advertising campaigns and quicker mobile game development due to the out-of-box chat, achievements, leaderboard, registration/authentication, in-app-purchases and many more features.

**Prerequisites**

1. In order to use Mikros, clients (game developers/publishers) must go to <https://developer.tatumgames.com/>
2. Select a subscription plan. The options are FREE, STARTUP, ENTERPRISE
3. Register an account, and verify this account via email
4. Once inside the Mikros dashboard, the client can "create app". The process of creating an app is filling out a form that includes details such as company name, game title, description, package name (bundle id), product assets, videos, category {RPG, Puzzle, Platformer, CCG, Strategy, Other}

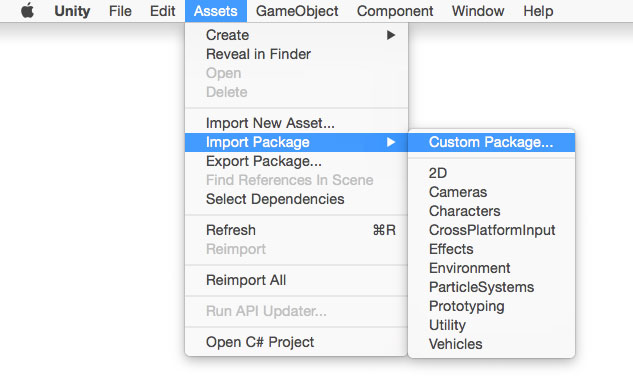
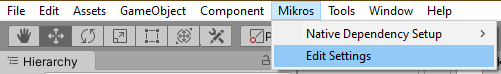
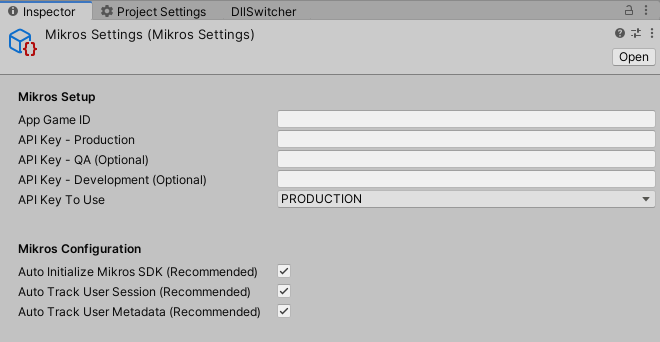
Once steps 1-4 are completed, the following will be generated:

1. App Game ID (auto-generated)
2. Production API Key (auto-generated)
3. QA API Key (Manual)
4. Development API Key (Manual)

**Add Mikros SDK to Project**

**N.B:***You need to have* ***minimum Android API Level of 21 (Android 5.0 ‘Lollipop’)*** *in your Unity project to work with Mikros. After successful importing of Mikros SDK, if its detected that the project has minimum Android API level below 21, then it is automatically upgraded.*

Now, the client has to include the Mikros SDK into their gaming product and setup the SDK by the following method:

1. Download the latest Mikros Unity SDK from [here](https://developer.tatumgames.com/public/assets/sdks/MikrosUnitySDK_v1.0.0.zip).
2. In the Unity editor, select **Assets > Import Package > Custom Package**. Navigate to the directory where you downloaded the Mikros Unity SDK and select *MikrosUnitySDK\_v1.0.0.unitypackage*.  
   N.B: You will need to remove previously integrated Mikros Unity SDK packages before importing a newer version.  
   
3. Import the assets in the package.  
     
   These are the contents of the Mikros Unity SDK.
4. Save your project. A post-build script will add a **Mikros** menu item to the Unity editor. If you don't see this, check your build for compilation errors and try building again.
5. In the Unity Editor, select **Mikros > Edit Settings  
   **
6. In the Inspector tab of **Mikros Settings**, paste in your App Game ID and API keys that have been generated. You also have option to edit other relevant settings from here.  
     
   1. *(Optional)* If you want manual initialization of the SDK, disable the ***Auto Initialize Mikros SDK*** option from the Mikros Settings. By default, it is kept enabled.
   2. *(Optional)* You also have the option to select which API key to use at any development stage. By default, Production API Key is used.
   3. *(Optional)* Mikros tracks user session for better user insights and is also crucial for Analytics purpose. This can be enabled or disabled by the ***Auto Track User Session*** option from the Mikros Settings. By default, it is kept enabled and is recommended to keep it that way.
   4. *(Optional)* Mikros collects and sends metadata for better user insights and is also crucial for Analytics purpose. This can be enabled or disabled by the ***Auto Track User Metadata*** option from the Mikros Settings. By default, it is kept enabled and is recommended to keep it that way.
7. Mikros Unity SDK depends on some gradle files for native Android support. These are automatically generated or modified at **Assets\Plugins\Android** after successful importing of Mikros SDK in your project.  
   In case, you need to generate any/all of those manually, that can be done from here:  
   