

GameCenter Plugin for Unity 3.4

Note: This plugin works on iOS 4.2 and above..

Build

- a. After make Unity XCode project link project with library "Assets/Editor/GameCenter/libGameCenterPlugin.a"
- b. Add Frameworks: GameKit, CoreGraphics
- c. Add in the file Info.plist UIRequiredDeviceCapabilities array value with "gamekit" item

Features

- Support leaderboards and achievements
- Saving scores / achievements in case of unavailability of the Internet, followed by sending data to the extent possible
- Showing tables in the standard GameKit methods
- Displays of new achievement with name, icon and description
- Support for the iPad/iPhone/iPod all orientations (all view set orientation as status bar)
- Asynchronous receive scores of any player and synchronous receive scores for the current player
- Notification of changes in connection status with GameCenter
- Int-value scores to the leaderboards is sorted in descending order

Initialization

To initialize the plugin, you must call the static method in class GameCenter:

```
void Authenticate(bool _useLeaderboards, bool _useAchievements);
```

As parameters, you must specify - whether to use a leaderboards and achievements. If the GameCenter no one table that have to be specified as false. Otherwise - the system will constantly try to refresh the data, which could adversely affect performance. Similarly, and with awards.

In order to determine, is authorized at any given time a player or not:

```
bool IsAuthenticated;
```

After authorization, the system automatically (when you install flags `_useLeaderboards` and / or `_useAchievements`) caches the current data on scores and achievements player to always synchronously could find their current value. To determine whether the cache is over:

```
bool IsComplete;
```

If you do not need to obtain data about the player, then this flag can be ignored.

Information about this player

All information about the current players can be obtained via the following methods and property:

```
string PlayerID;  
string PlayerAlias;  
int GetScore(string _categoryID);
```

Leaderboards

For work with leaderboards use methods:

```
void ShowLeaderboard();  
void ShowLeaderboard(string _categoryID);  
void ReportScore(int _score, string _categoryID);
```

For asynchronous receive scores of any player on his PlayerID need to use the method:

```
void ReportScore(int _score, string _categoryID);
```

Data will be obtained through the event:

```
void ResponseScoresDelegate(string _playerID, int _scores, string  
_categoryID);
```

For example:

```
...  
  
GameCenter.ResponseScores += new  
ResponseScoresDelegate(Event_ResponseScoresEvent);  
  
...  
  
GameCenter.RequestScores(_playerID, _categoryID);  
  
...  
  
void Event_ResponseScoresEvent(string _playerID, int _scores, string  
_categoryID)  
{  
    // after a while, then get the necessary data  
}
```

Note: scores of the current players are available at any time immediately after the method

```
int GetScores(string _categoryID);
```

Achievements

To work with achievements, there are the following methods:

```
void ShowAchievements();  
void ReportAchievement(string _achievementID);  
void ReportAchievement(string _achievementID, float _percentage);
```

If a player really got a new award, he'll be alerted:

