Gamfari SDK Documentation

STEP ONE : IMPLEMENTATION ON

WWW.GAMFARI.COM

- Visit http://www.gamfari.com/developers
- Enter the NameOfGame as username and Secret Key as password. This was emailed to you after registration with gamfari.com
- Your dashboard should contain every information you need to set-up sdk on unity.
- Copy “Game ID” and Secret-Key. This will be used on unity
- Click Manage Ads, you can upload as much as 8 – Ad images and the links you want to visit when your ads are clicked.
- Make sure to enter links without “http://”.

STEP TWO: IMPLEMENTATION ON UNITY

Initial Set-up

- Locate “Gamfari” at Assets/Gamfari/Resources and double click
- Fill the public variables as follows
    - Game ID
    - Secret Key
- To check out our demo scenes locate scenes at Assets/Gamfari/demo/scenes or scripts at Assets/Gamfari/demo/scripts

For Personal Implementation

- Create a new C# Script
- Add namespace to the top of script (Using GamfariHandler;)

Get Score Method

- Before calling any methods, Please call GF.Init(). To initialize plugin in your awake method.
- Then log in user to retrieve scores
- To do this call the function `GF.GetScore();`
- GF.GetScore() collects two parameters:
    - A string (Username of user you which to login)
    - A call back method (Handles Connection Results)
- Example call GF.GetScore(Username,RequestScoreStatus);
- This method callback should collect to parameters
    - A bool – to get the status of the connection, Returns true if it was successful and false if it wasn’t
    - A String – Contains the score if status was true or contains error messages

Implementation Example:

```
Example: GetScore CallBack Method

public void RequestScoreStatus(bool status,string ScoreCallBack)
    {
        if (status)
        {
            int.TryParse(ScoreCallBack, out LocalUserScore);
            Score.text = LocalUserScore.ToString();
            LoggedIn.SetActive(true);
            NotLoggedIn.SetActive(false);
            Status.text = "Successfully logged in user and got scores";
            print(status + " " + LocalUserScore);
        }
        else
        {
            Status.text = ScoreCallBack;
        }
```

Save Score Method

- To do this call the function `GF.SaveScore()`
- GF.SaveScore() collects two parameters:
    - An Int (Score of user you which to submit)
    - A bool to tell server whether to charge users or not.
    - A call back method (Handles Connection Results)
- Example call GF.SaveScore(1000, true,RequestSaveStatus);
- This method callback should collect to parameters
    - A bool – to get the status of the connection, Returns true if it was successful and false if it wasn't
    - A String – Contains more information about the status of connection, Usually contains error messages but can also show success messages

Implementation Example:

```
Example: SaveScore CallBack Method

public void RequestSaveStatus(bool status, string SaveCallBack)
    {
        if (status)
        {
            Status.text = "Successfully Saved Score";
        }
        else
        {
            Status.text = SaveCallBack;
        }
```

Get LeaderBoard Method

- To do this call the function `GF.GetLeaderboard()`
- GF.GetLeaderboard() collects one parameters:
    - A call back method (Handles Connection Results)
- Example call GF.GetLeaderboard(RequestLeaderboardStatus);
- This method callback should collect to parameters
    - A bool – to get the status of the connection, Returns true if it was successful and false if it wasn't
    - A List of type String – Returns a list of each row of users separated by space. Each row contains (Username Score Rank). The Username, Score and Rank can be can be put into individual columns by looping through each row in the array and splitting each row of the results array by space.

      Implementation Example:

```
Example: LeaderBoard CallBack Method

public void RequestLeaderboardStatus(bool status,List<string> LeaderboardCallBack)
    {
        if (status)
        {
            SplitUsers(LeaderboardCallBack.ToArray());

                Status.text = "Successfully Downloaded Leaderboard";
        }
        else
        {
            for(int i = 0; i< LeaderboardCallBack.Count; i++)
            {
                print(LeaderboardCallBack[i]);
            }
        }
```

```
Example: Split Each Users Details Method

    List<string> UserName = new List<string>();
    List<string> UserScore = new List<string>();
    List<string> UserRank = new List<string>();
public void SplitUsers(string[] Users)
    {
        UserName.Clear();
        UserScore.Clear();
        UserRank.Clear();
        Leadercount = Users.Length;
        string[] split;

        for (int i = 0; i < Users.Length; i++)
        {
            split = Users[i].Split(' ');
            UserName.Add(split[0]);
            UserScore.Add(split[1]);
            UserRank.Add(split[2]);
        }
    }
```

GET-Ads Control User Method

- To do this call the function `GF.FetchStatus();`
- GF.FetchStatus() collects one parameters:
    - A call back method (Handles Connection Results)
- Example call GF .FetchStatus(AdsStatusCallback);
- This method callback should collect to parameters
    - A bool – to get the status of the connection, Returns true if it was successful and false if it wasn't
    - A String – Contains more information about the status of connection, Usually contains error messages but can also show success messages

Implementation Example:

```
Example: Registration CallBack Method

   public void AdsStatusCallback (bool status, string AdsCallBack)
    {
        if (status)
        {
            CanShowAds = true;

            print(AdsCallBack);

        }
        else
        {
           print(AdsCallBack);
        }
    }
```

Show-Ads Control User Method

- To do this call the function `GF. ShowAds();`
- GF. ShowAds collects no parameters:

Implementation Example

```
Example: Registration CallBack Method

   public void ShowMyAds (bool status, string showAdsCallBack)
    {
        if (CanShowAds)
        {
            GF.ShowAd();        }

        else
        {
           print("Ads not ready yet");
        }
    }
```

Register User Method

- To do this call the function `GF.RegisterUser();`
- GF.RegisterUser() collects four parameters:
    - A String (Username of user trying to register)
    - A String (Password of user trying to register)
    - A String (Email of user trying to register)
    - A call back method (Handles Connection Results)

- Example call GF.RegisterUser(RUsername.text, RPassword.text, REmail.text, RequestRegisterStatus);
- This method callback should collect to parameters
    - A bool – to get the status of the connection, Returns true if it was successful and false if it wasn't
    - A String – Contains more information about the status of connection, Usually contains error messages but can also show success messages

  Implementation Example:

```
Example: Registration CallBack Method

   public void RequestRegisterStatus(bool status, string RegisterCallBack)
    {
        if (status)
        {
            Status.text = RegisterCallBack;
        }
        else
        {
            Status.text = RegisterCallBack;
        }
    }

}
```

For support, please contact Emmanuel (emmanuel@gamfari.com)