



# NEWS AND STATS

v1.3 DOCUMENTATION

# Table of Contents

<b>Requirements</b>	<b>3</b>
<b>Installation</b>	<b>4</b>
PHP and Database	4
<b>Add a Game and News</b>	<b>7</b>
<b>UNITY</b>	<b>10</b>
<b>Other Game Engines</b>	<b>12</b>
<b>Stats</b>	<b>13</b>
<b>List of Files</b>	<b>15</b>
FILES SERVER	15
FILES UNITY	16

**A short warning:** This package is not tracking *users*, it's tracking *unique devices*! A unique device may well be a secondary or third device of a paying customer or it may be a device using a pirated copy of your game. In other words: Don't expect a certain number of unique devices to translate to the same number of sales!

# Requirements



- A webhost/server supporting at least **PHP 4** and **MySQL 5.1**. (It will most likely run on MySQL 4 as well, but since I have no way of testing that, it's not supported.)
- An FTP client to upload the files. (The free FireFTP plugin for Firefox, for example)

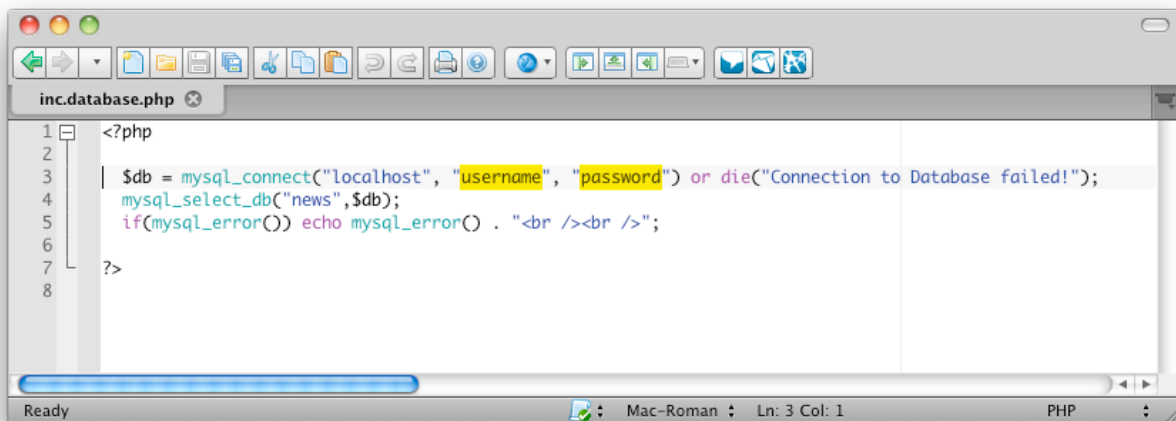
If you don't have a webhost and are looking for one in Europe, I can recommend this one (located in Germany): <http://all-inkl.com/?partner=145295> (cheapest package is EUR 5,- per month and supports both PHP and MySQL)

If you can highly recommend a webhost in a different part of the world, please [write a mail](#) and I'll add it here!

# Installation

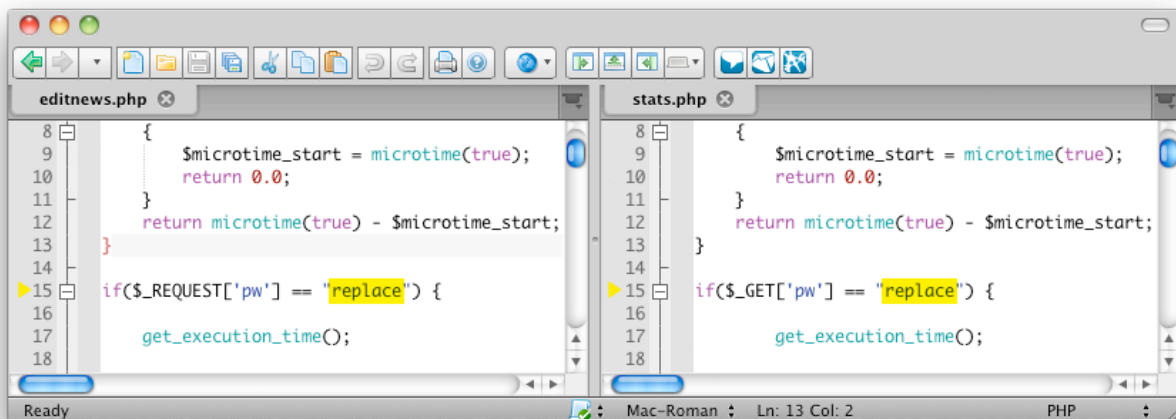
## PHP and Database

1. Extract **ServerSide.zip**. Open **inc.database.php** and fill in the **username** and **password** for your **MySQL** database. (If you don't know these, log into your webhost's account to find them or contact your webhost.)



```
1 <?php
2
3 $db = mysql_connect("localhost", "username", "password") or die("Connection to Database failed!");
4 mysql_select_db("news", $db);
5 if(mysql_error()) echo mysql_error() . "<br /><br />";
6
7 ?>
8
```

2. Edit **stats.php** and **editnews.php** and **assign a password in line #15** - you'll use these passwords to access the stats page and to create/edit news items.



```
editnews.php
8 {
9     $microtime_start = microtime(true);
10    return 0.0;
11 }
12 return microtime(true) - $microtime_start;
13 }
14
15 if($_REQUEST['pw'] == "replace") {
16     get_execution_time();
17 }
18

stats.php
8 {
9     $microtime_start = microtime(true);
10    return 0.0;
11 }
12 return microtime(true) - $microtime_start;
13 }
14
15 if($_GET['pw'] == "replace") {
16     get_execution_time();
17 }
18
```

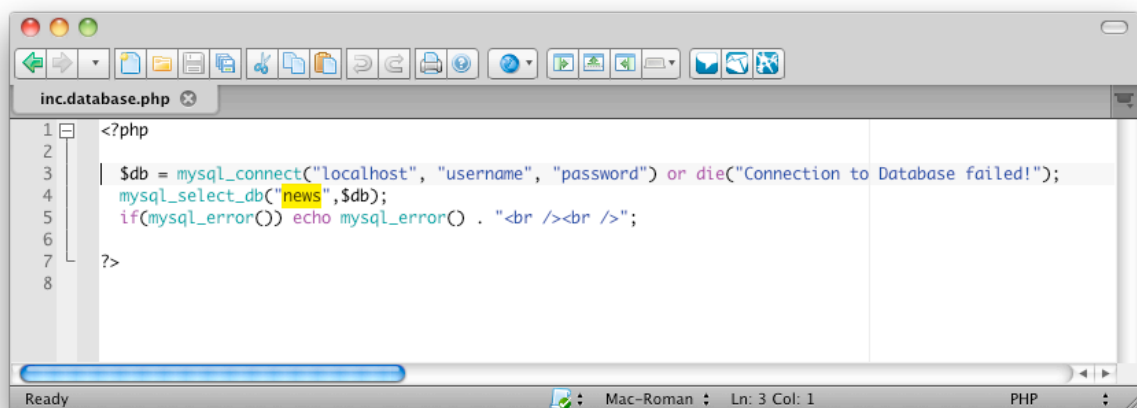
3. **Upload** all the files **except** **SQL\_Install.txt** to a directory on your server.
4. **Run** **install.php** on the server (For example: If you put all your scripts into a directory called "news" on your server, open a browser and go to <http://www.yourdomain.com/news/install.php>) Worked? **Next Chapter!**

If this resulted in an error message we'll have to jump through a few more hoops, sorry! But we'll make it work - *I promise!*

Why did it fail? Many webhosts don't allow you to create databases from script, so we'll have to **create a database through the interface that your webhost provides**. (If there is no database set up for you already, you will usually be able to create one by logging into your account at your webhost's website.)

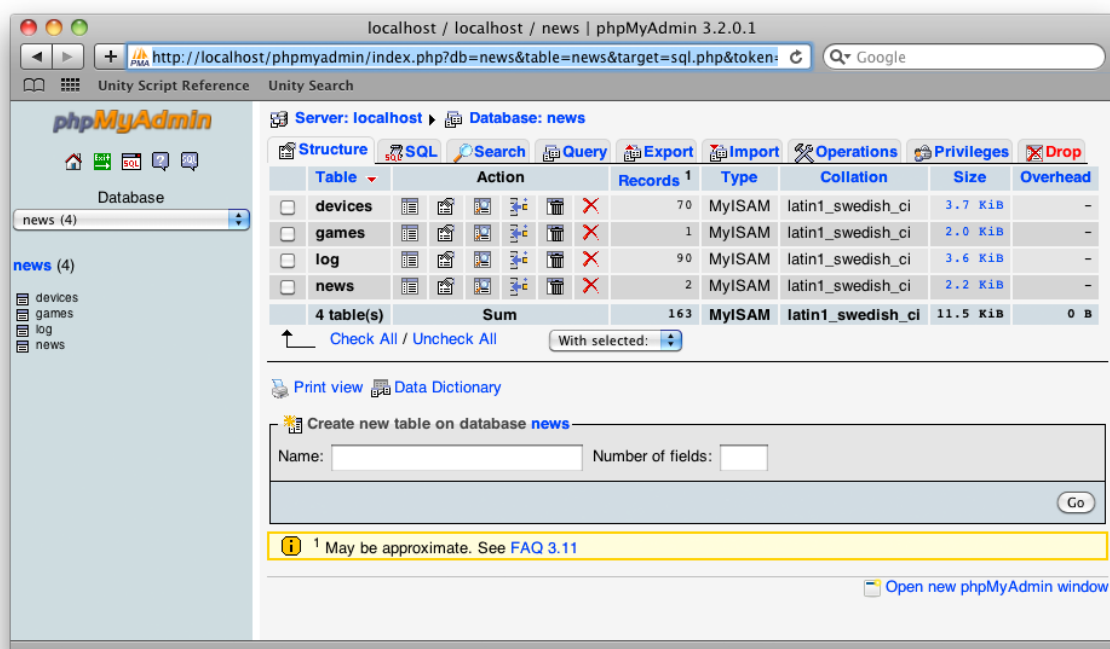
**IMPORTANT for non-english language users:** Make sure the DB Collation is set to **utf8\_general\_ci**

Once you have a database created (if you can, name it "news" - if the webhost doesn't let you choose your own name, update **inc.database.php** with the correct **name of your database!**)

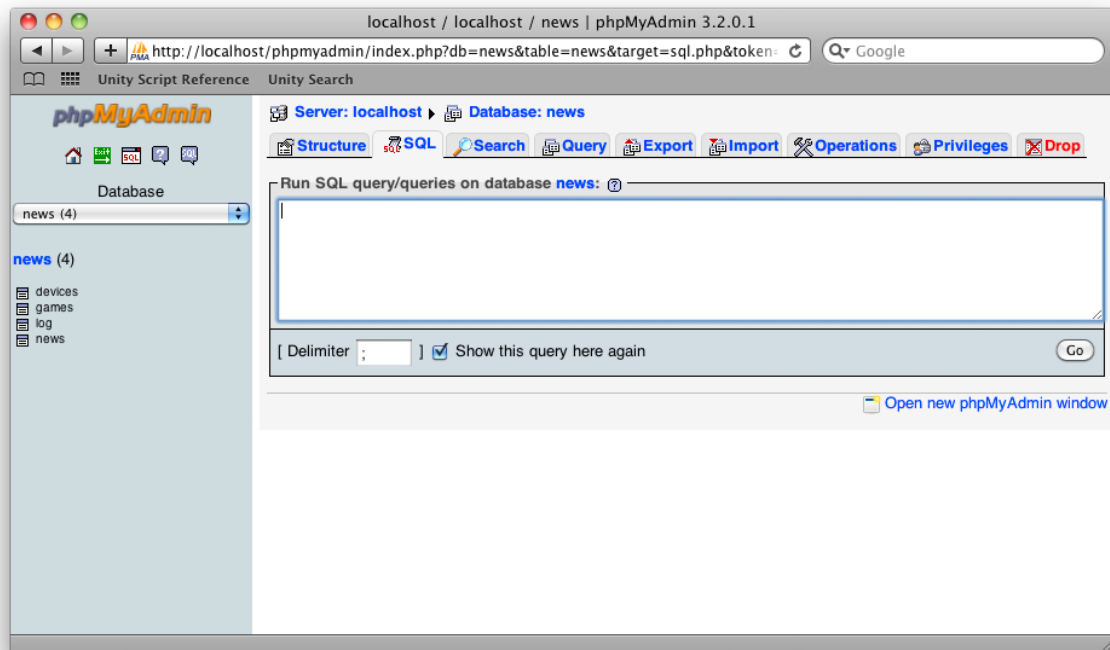


```
1 <?php
2
3 | $db = mysql_connect("localhost", "username", "password") or die("Connection to Database failed!");
4 | mysql_select_db("news", $db);
5 | if(mysql_error()) echo mysql_error() . "<br /><br />";
6
7 ?>
8
```

Now log into **phpMyAdmin** (a popular visual interface to MySQL that most webhosts provide - you should be able to access it through your webhost's account page). This is what it looks like:



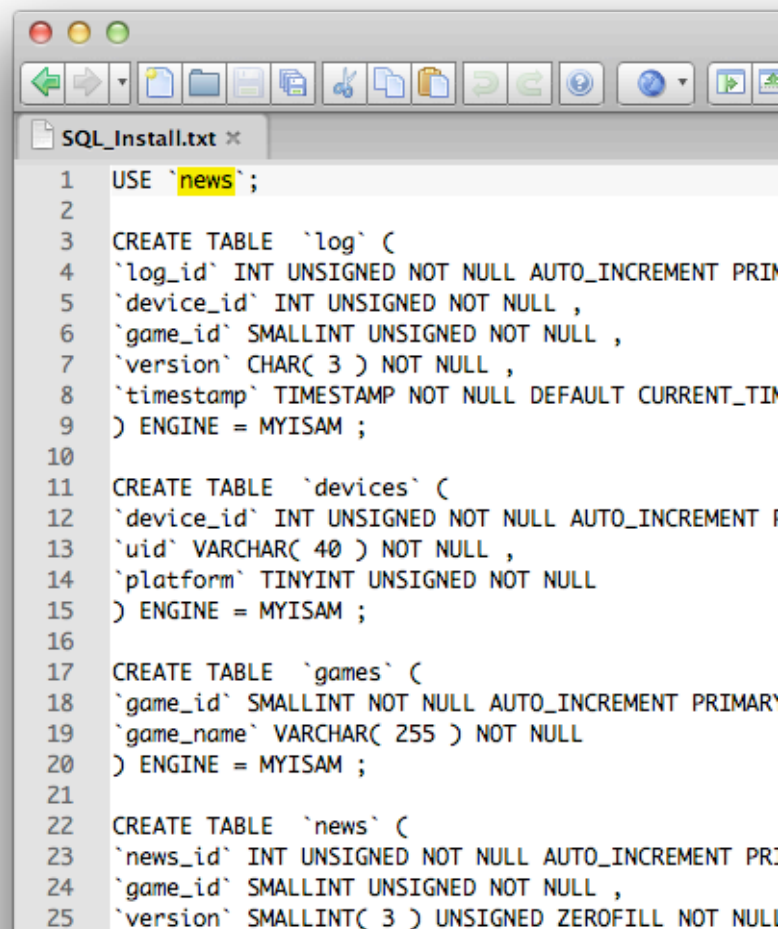
Select your database and navigate to the **SQL tab** - now we're going to copy/paste all the code that will create the tables and fields of the database into that textfield.



Open [SQL\\_Install.txt](#). But just one more thing before we copy/paste! If your database is NOT called "news", enter the correct name of your database at the very top of the file! (As marked in the image below.)

Now copy/paste the contents of [SQL\\_Install.txt](#) into the "Run SQL query/queries" textfield and press "Go"

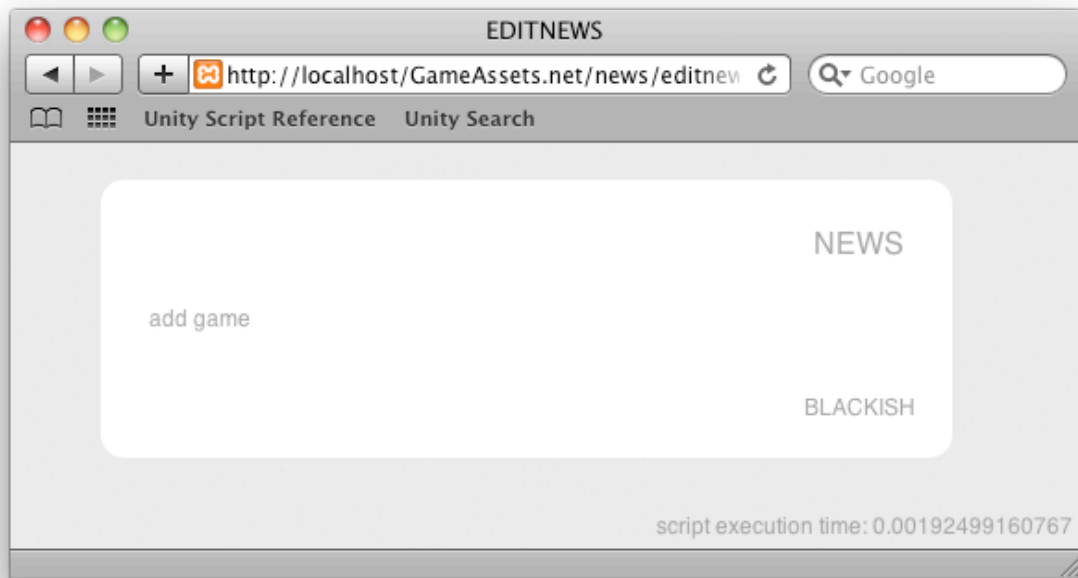
You should see a **success-message** and the individual tables should show up in the sidebar on the left. Now we're ready to go!



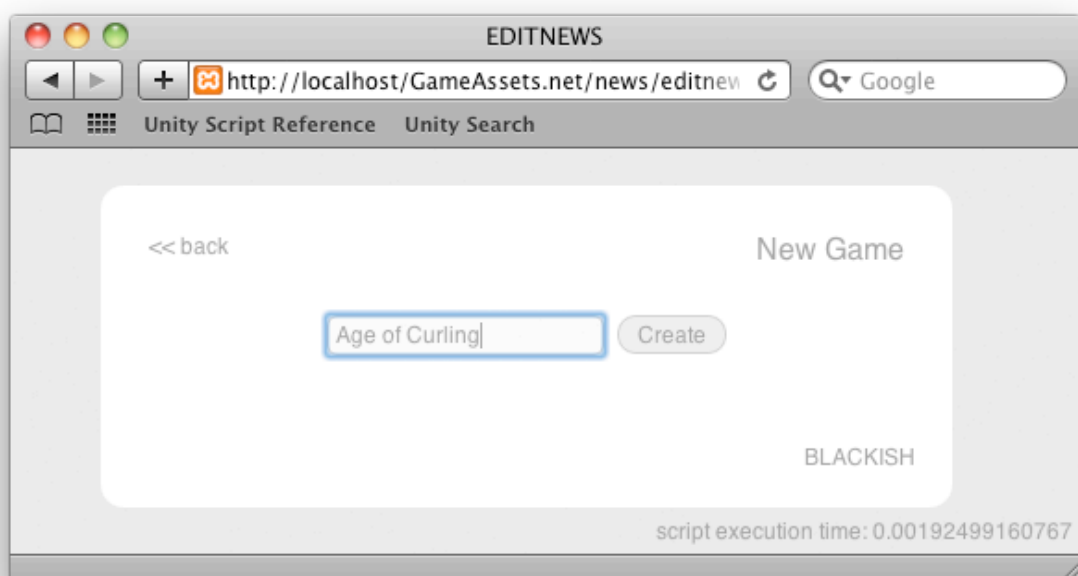
# Add a Game and News

1. Open a browser and visit <http://www.yourdomain.com/yourdirectory/editnews.php?pw=yourpassword> (With your correct domain, directory and password, of course...)

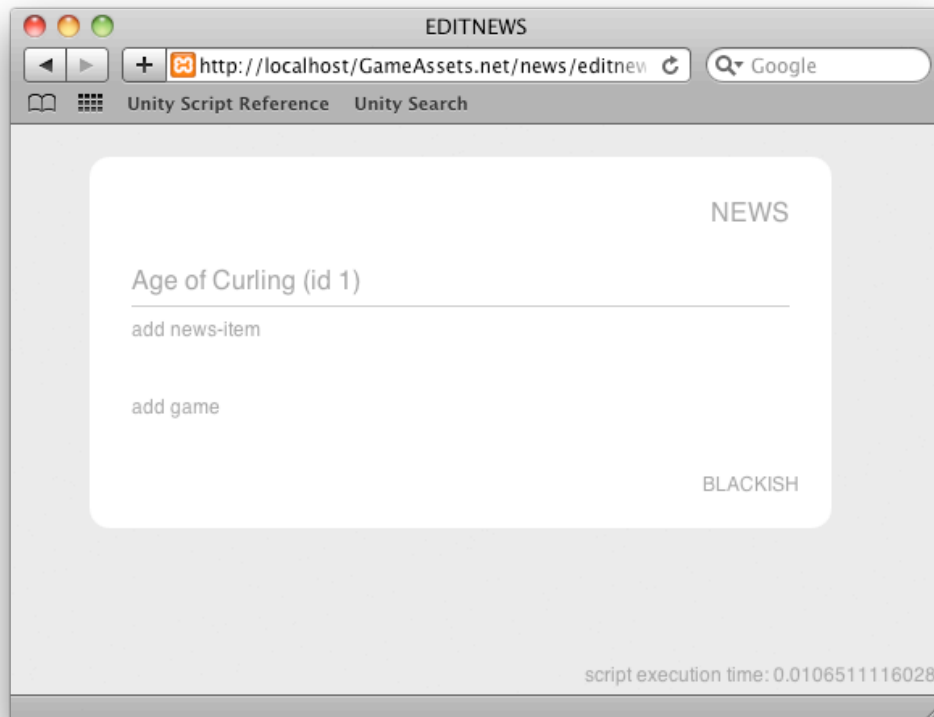
If this is the first time you visit this page it should look like this:



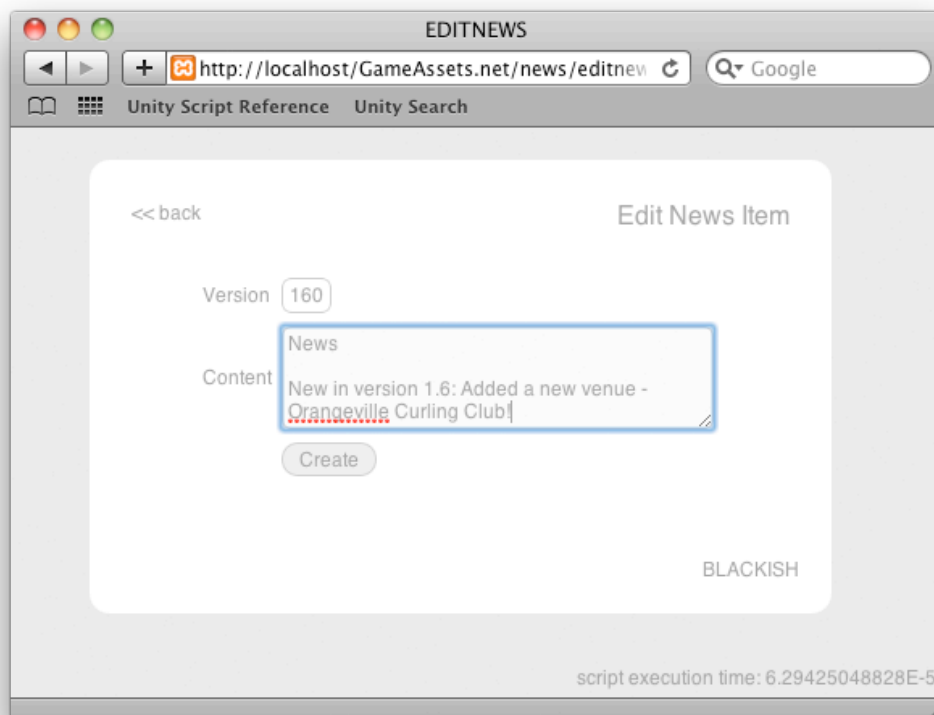
2. Not much there yet, so let's click on "add game" and enter the name of your game!



3. Now we have a game in our database. Notice the **id** written next to the name? This is the **Game ID** we have to use for the [News script](#) in Unity.



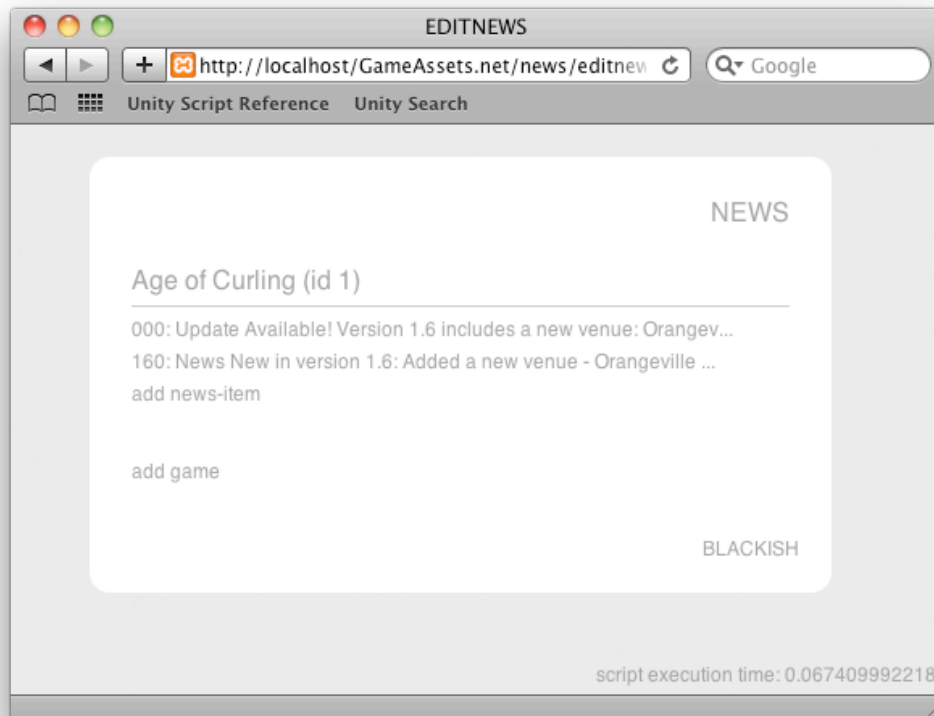
4. But first let's **create a news item** for this game! Click on add news-item, enter a **version number** (no dots! 160 is 1.6.0) and some **content**.





**Special Case:** The news item for version number 0 will be displayed for all versions that don't have their own news item!

In the picture below I've set up two news items, 000 will be displayed for all older versions and 160 will be displayed for the newest version of the game (which is 1.6).



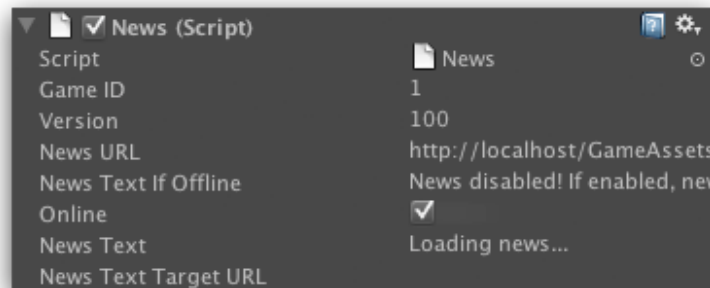
You can of course add as many games and as many news items per game as you like.

Click on any news item to be able to **change** or **delete** it!

# UNITY

Our server-side work is done! Let's switch to Unity and add the news to our game! (If you're not using Unity skip to the [next chapter...](#))

1. **Import** the **news.cs** script and put it on an empty GameObject.



2. In the inspector enter the correct **Game ID** for your game - as seen in the [editnews.php](#) script, [remember?](#)
3. Enter the correct **version** of your game - default is 100 (equals 1.0.0), but everything between 000 and 999 will work!
4. **News URL**: Enter the base path where all the php scripts reside on the server, the script will find the individual scripts there as needed. (for example: <http://www.yourdomain.com/yourdirectory/>)
5. Customize **News Text If Offline** - This text will be displayed when no URL is entered or if the script is set to offline mode.  
*Bonus Unity tip: press Alt+Enter inside a textfield to get a new line.*
6. **Run the game!** If Everything worked correctly the **News Text** variable should now hold the correct news string for this version of this game!

**Access and use this variable from your GUI script to display the news to your players!**

Or you can just use the included DemoNewsBox.cs script - just set the correct size, location and skin, and link it to the News script! (Check out the included demo scene to see it set up and in action!)

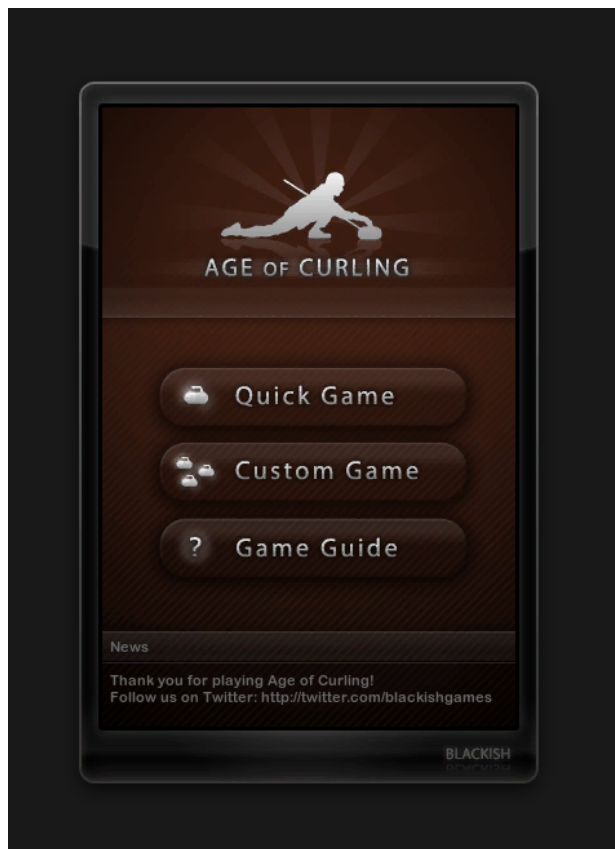
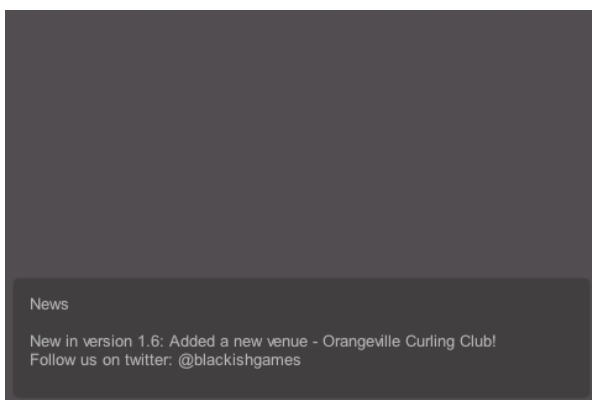
Oh, before I forget, a shameless plug: naturally these scripts work great with our own GUIKit001: <http://GameAssets.net>

7. **News Text Target URL?** Wondered what this variable is for? You can add a link to your news item, simply by adding it to the end of the news string right after an “|” (This is called the “pipe” and on an English International keyboard layout you get this key by pressing Shift+Backslash) and the news script will extract the link and store it in this variable.

Very useful if you want to be able to link to a download-location for an update, for example.

An example news-string: *News blah blah|<http://mydomain/gohere/>*

On the left you can see the DemoScene hooked up to our server. On the right you can see everything integrated into the main menu of a game!



# Other Game Engines

Sorry, I don't have a ready-to-use script for you, but it should be easy enough to build your own once you know the following:

To retrieve news from the server call `news.php` and send along a few `parameters` and `values` like this:

<http://yourdomain/dir/news.php?gid=1&v=100&pf=1&uid=2398hf9sdfs23>

You can get a `uid` (unique ID) from the server by calling the `getID.php` script with a random number and an `MD5 checksum` composed of that random number and a codeword. (Or you could cut the additional safety measures and modify `getID.php` to just generate an ID without the checksum verification... Hell, you could even generate the uid right there in the client - the chances that two players will generate a uid at the exact same microsecond are super-slim, but I'm selling this script, so I wanted to make extra-sure all uids would be unique...)

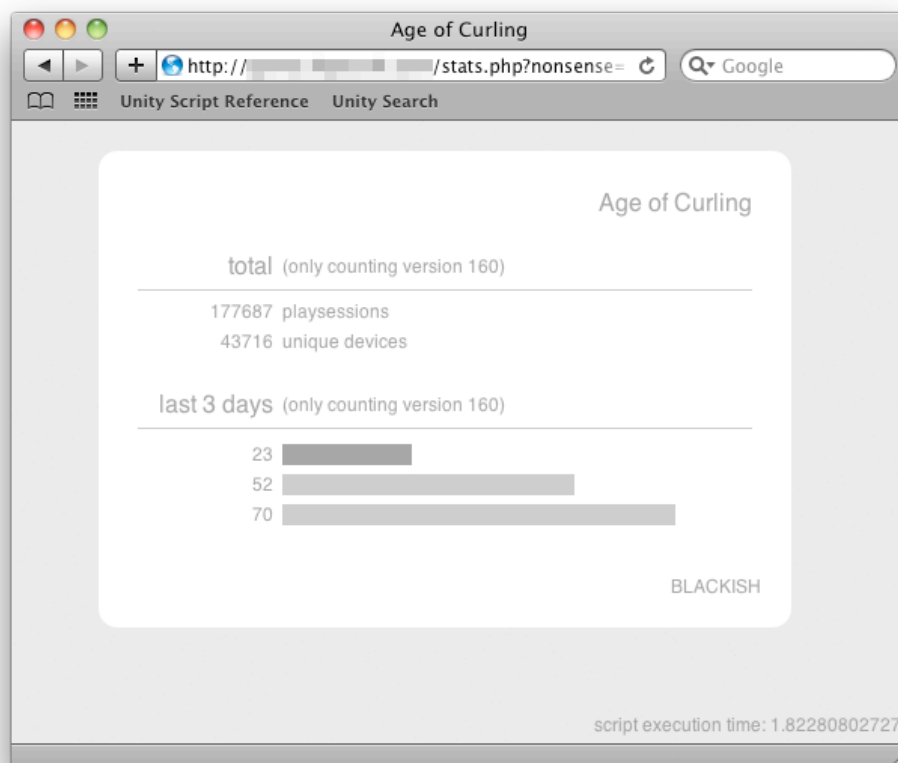
# Stats

Now here come the perks: With the ability to provide **customized news-strings** to our players we can also collect **anonymous usage data!**

We can see how often the game is launched (or how often a certain version of the game is launched), how many new unique devices

The **stats.php** page shows **total unique users**, **total number of play-sessions** as well as **unique new users per day** for the last few days.

<http://domain/dir/stats.php?pw=yourpassword&gid=1>



As you noticed, we're sending a few **parameters** and **values** along right there in the URL to refine what we want to have displayed. Here's a list of all the supported parameters and what they are for:

- **?pw=yourpassword** - **Password**. the password needs to match the one you set inside **stats.php**, without this you'll only get a blank page.
- **&gid=1** - **Game ID**. only show stats for game #1 - if you leave this out, it will default to **gid=1**

- **&v=100 - Version.** only show stats for this version - if you don't set this you'll get stats for all versions of the game. Note: This also affects total number of devices and play-sessions! If you want to see the overall total number of devices and play-sessions, don't use this!
- **&offsetHours=10 - Timezone Offset.** The script uses server-time, if you want your days to be counted from a different start time, use offsetHours! *Why would you want to do this?* I added this because my server is in Europe and when tracking iOS games I want my days to match Apple's days... (Pacific Time)
- **&d=10 - Days.** show unique new users for the last 10 days - leave it out and it will default to d=3. Please be aware that each additional day will increase the workload of your server. This doesn't matter much at first, but once you have a few million entries in the log, each additional day will add a second or so to the script execution time (depending on your server of course).

A full example URL could look like this: <http://yourdomain.com/yourdir/stats.php?pw=yourpw&gid=1&v=100&offsetHours=10&d=5>

And yes, you can bookmark it like that!

**Bonus Tip:** Please be aware that your password is visible in the in URL field! So if you want to show your stats around, it's best to add another parameter called "nonsense" with a value that's just a lot of random characters so your password is out of view. Like this: <http://yourdomain.com/yourdist/stats.php?nonsense=asdlfkjaweofijaslfjhawkefbasjkfnlasdkfjasldkfjasldkfjalsdkf&pw=yourpw&gid=2&v=125&offsetHours=9>

# List of Files

A list of all included files (except this one):

## FILES SERVER

These files are all inside a **ZIP** called **ServerSide** - Move it out of Unity and extract it!

**editnews.php** - Manage Games and News Items here.

**getID.php** - Used to create unique IDs. the unity script will use the device UDID for all iOS devices and get a unique ID from this script on all other devices! (keeping this on the server will make sure no 2 devices will ever get the same ID)

**inc.database.php** - Holds your database login information - this script is called from news.php, editnews.php and stats.php, so you don't need to enter your login information into all those files. It's secure in there. If PHP is set up correctly (and it usually is by default) then you can't access this

**install.php** - This script can create the entire MySQL database for you, IF your webhost allows creating databases from script. - if it does not, you'll need to jump through some hoops and use SQL\_Install.txt

**news.php** - This script is called from inside your game. It creates an entry in the log and returns a news-string for the given version of the game.

**SQL\_Install.txt** - **SQL code** that holds information to create the database needed by the scripts. Follow the install instructions in the beginning of this document on how to use this!

**stats.php** - Your game's **stats**! Shows total unique users, total number of play-sessions as well as unique new users per day for the last few days.

**style.css** - **Stylesheet** used by editnews.php and stats.php to make the two pages pretty.

## FILES UNITY

**DemoNewsBox** - **C# script** that takes the news string and link provided by the News script and displays a skinnable news-box.

**DemoNewsBoxBackground** - The background-**image** used by DemoNewsBoxSkin.

**DemoNewsBoxSkin** - The **GUISkin** used by DemoNewsBox.

**DemoScene** - See it all set up and in action in this demo scene.

**News** - **C# script**, the heart of this package. This script retrieves the news-string for the current version of the game.

**ServerSide** - This is a **ZIP-file** containing all the files for the server. Move it out of Unity and extract it!

**Wait** - A **C# script** that contains a variation of the WaitForSeconds function that works independent of the current timeScale. It's required by the News script.



thank you!

**web**

<http://blackish-games.com>

<http://GameAssets.net>

**email**

[support@GameAssets.net](mailto:support@GameAssets.net)

**twitter**

[@GameAssets](https://twitter.com/GameAssets)