



DR.XAMÃ

GAME JOLT API DOCUMENTATION

21 APRIL 2020

Introduction

OBJECTIVE

The plugin makes the integration with the Game Jolt, allowing the complete use of the tool, from sending trophies to players to manipulating table scores. The system has a cloud storage mechanism, enabling players to have your games always at the last point on every device.

DOCUMENT SUMMARY

This document aims to expose all the functionalities of the system.

ABOUT THE AUTHOR

Luiz Guilherme dos Santos, 21 years old, is responsible for the public name "Dr.Xamã", responsible for the creation of all the material presented in the following internet domains, www.drxama.com and www.drxama.com.br, offered by GRUPO UOL.

DOCUMENTATION VALIDITY

This content was written and posted in April 2020. Current system version **2.5.0**.

Index

[Page 4](#)

Plugin installation.

[Page 5](#)

Use of the DX variable.

[Page 6](#)

Parameters.

[Page 7](#)

Notes.

[Page 8](#)

Plugin Commands 1.

[Page 9](#)

Plugin Commands 2.

[Page 10](#)

Script commands 1.

[Page 11](#)

Script commands 2.

[Page 12](#)

Script commands 3.

[Page 13](#)

Script commands 4.

[Page 14](#)

Game Jolt API 1.

[Page 15](#)

Game Jolt API 2.

[Page 16](#)

Video Links.

[Page 17](#)

Contact us.

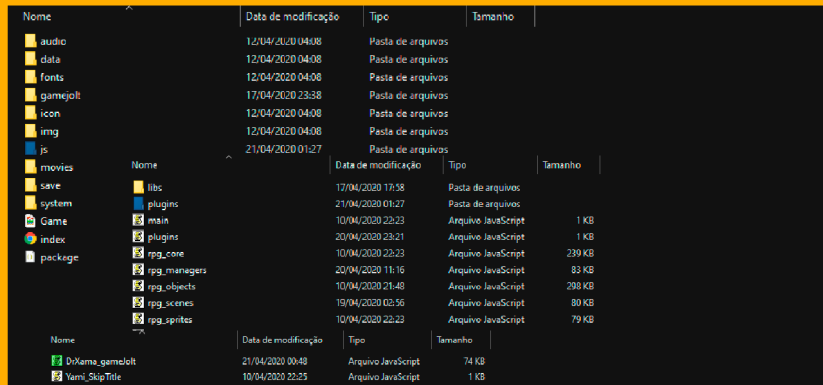
[Page 18](#)

Publicity.

Plugin Installation

Overview

Follow the instructions below.



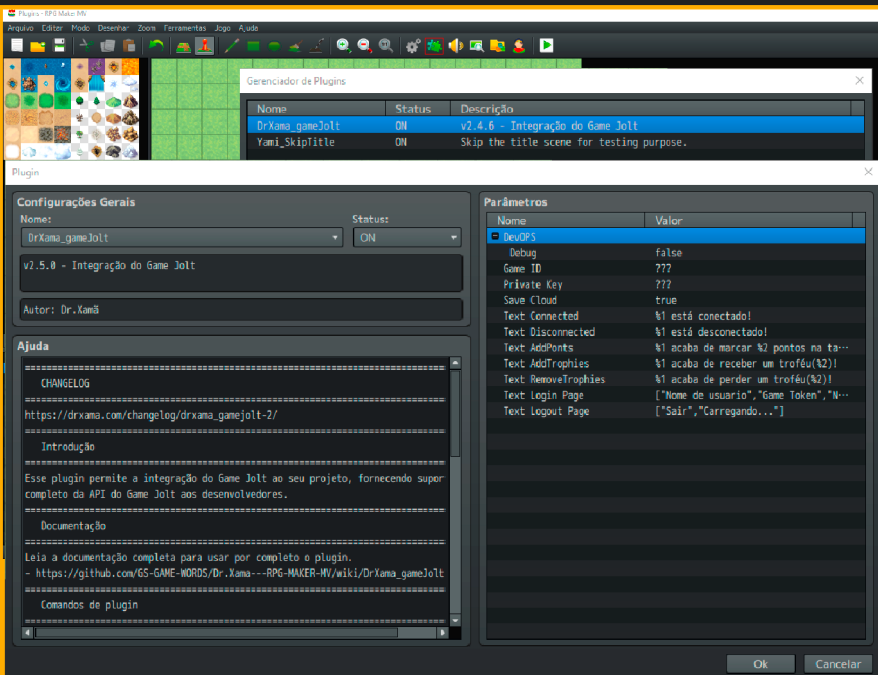
Nome	Data de modificação	Tipo	Tamanho
audio	12/04/2020 04:08	Pasta de arquivos	
data	12/04/2020 04:08	Pasta de arquivos	
fonts	12/04/2020 04:08	Pasta de arquivos	
gamejolt	17/04/2020 23:38	Pasta de arquivos	
icon	12/04/2020 04:08	Pasta de arquivos	
img	12/04/2020 04:08	Pasta de arquivos	
js	21/04/2020 01:27	Pasta de arquivos	
movies			
save			
system			
Game			
index			
package			

Nome	Data de modificação	Tipo	Tamanho
libs	17/04/2020 19:58	Pasta de arquivos	
plugins	21/04/2020 01:27	Pasta de arquivos	
main	10/04/2020 23:23	Arquivo JavaScript	1 KB
plugins	20/04/2020 23:21	Arquivo JavaScript	1 KB
rpg_core	10/04/2020 23:23	Arquivo JavaScript	239 KB
rpg_managers	20/04/2020 11:16	Arquivo JavaScript	83 KB
rpg_objects	10/04/2020 21:48	Arquivo JavaScript	298 KB
rpg_scenes	19/04/2020 02:56	Arquivo JavaScript	80 KB
rpg_sprites	10/04/2020 23:23	Arquivo JavaScript	79 KB

Nome	Data de modificação	Tipo	Tamanho
DrXama_gameJolt	21/04/2020 00:48	Arquivo JavaScript	74 KB
Yami_SkipTitle	10/04/2020 22:25	Arquivo JavaScript	1 KB

Step 1

Download the plugin and place it in the "plugins" folder, located in the root folder of your project.



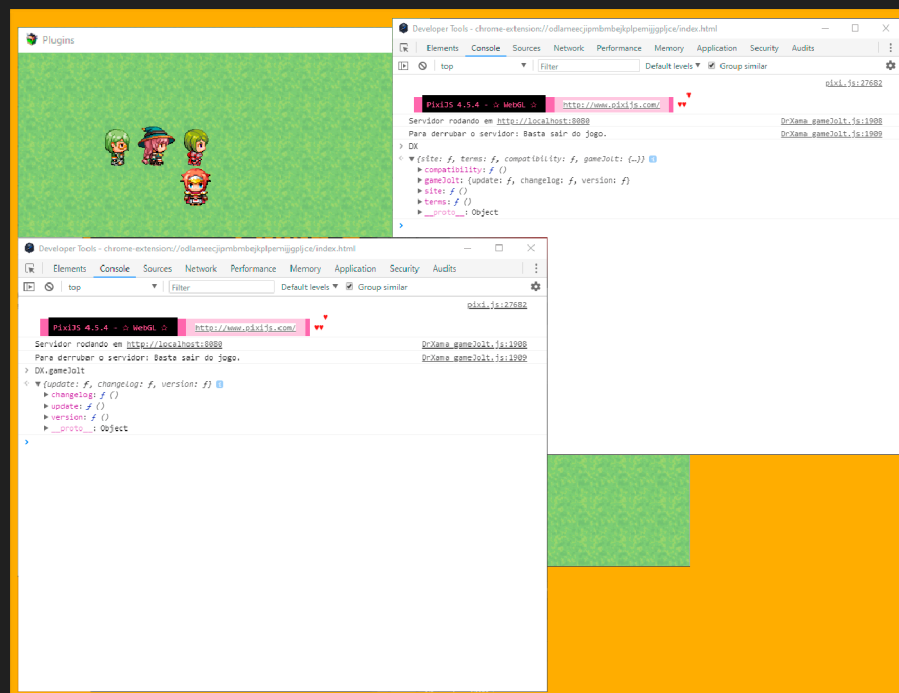
Step 2

Open the MAKER MV RPG and check the plugin.

Using the DX variable

Overview

See the images below.



Step 1

When opening your game press F12 to open the debug.

Commands

- DX.compatibility

Returns a message on debug saying if the plugin is compatible with your version of RPG MAKER MV.

- DX.site

Open Dr.Xamã official website.

- DX.terms

Open the terms of use page for Dr.Xamã plugins.

- DX.gameJolt.changelog

Opens the page containing information about all plugin updates.

- DX.gameJolt.update

Open the file page to download the latest version.

- DX.gameJolt.version

Returns a message on the debug containing the plugin version.

Official Domains, www.drxama.com and www.drxama.com.br.

Parameters

DevOPS Debug

Activates/deactivates system debugging.

Game ID

ID of your game.

Private Key

Access key of your game API.

Save Cloud

Activate/deactivate the storage in the cloud.

Text Connected

Text presented when the user is connected.

%1 = Player Name

Text Disconnected

Text displayed when the user is disconnected.

%1 = Player Name

Text AddPonts

Text presented when the user marks a punctuation.

%1 = Player Name

%2 = Number of points

%3 = Table Name

Text AddTrophies

Text presented when the user receives a trophy.

%1 = Player Name

%2 = Trophy Name

Text RemoveTrophies

Text presented when the user loses a trophy.

%1 = Player Name

%2 = Trophy Name

Text Login Page

All texts displayed on the Login screen. The texts are in sequence.

Text Logout Page

All texts displayed on the Logout screen. The texts are in sequence.

Notes

- When leaving the game all users sessions will be finalized.
- All commands related to users, should be done after users login. Wait until the login is complete to start a user related command.
- The system performs a Ping in the user's session every 3600(milliseconds), to inform that the user is active. By default the Game Jolt ends the user's session after 30 seconds of absence.
- Support for multiple users at the same time.
- Interface of login/logout elegant.
- Cloud storage system only works for the user logged in with the graphical interface, this prevents the file from being lost in multiuser synchronization.
- When the user logs into the system using the graphical interface, the system checks if there is a pending file for synchronization.
- For reasons of security and integrity of the player's files, it is recommended to the game developer, create an alert that notifies the player, that closing the game during the process of storing and uploading the files, may cause the loss of information and even corrupt the file data. In this possible situation, the system has both alerts during the upload of the files and the during storage, but the alert with a clear and descriptive text about the consequences of the actions described above, is responsibility by the game developer.
- The developer of the game must pay attention to create an alert based on RGPD, about the storage of player data such as name, password and personal information, and add a clause in the terms of contract of the game, informing the player that the same when making a connection to your account Game Jolt, the game has a storage in the cloud, if the player accepts, he runs the risk of losing his data in certain situations such as power outage, disconnection from the Internet, unexpected closure of the game, and other situations that may cause the transmission and receipt of data inconsistent or unfeasible.

Plugin commands 1

GameJoltAddUser

Add a new Game Jolt user to your game.

Username – Username of Game Jolt.

Game Token – Special Player Password, used to access the games.

Example:

```
GameJoltAddUser DrXama 6da21e
```

GameJoltLoginUser

Start the user's session in the game.

Username – Username of Game Jolt.

If you need to change the password of the player, because his registration was made with the wrong password, just register him again, the system recognizes that he is already registered and makes the change of password without registering him again, avoiding the duplication of the user in the system.

Example:

```
GameJoltLoginUser DrXama
```

GameJoltLogoutUser

Ends the user's in-game session.

Username – Username of Game Jolt.

Example:

```
GameJoltLogoutUser DrXama
```

GameJoltScoresAddPoints

Adds the user's score to a Scoreboards table.

Username – Username of Game Jolt.

TableID – Identifier number of the table.

Score – Number of points to be given to the player.

Sort - Maximum table or score value.

Example:

```
GameJoltScoresAddPoints Username TableID Score Sort
```

GameJoltScoresAddGuestPoints

Add the guest score to a Scoreboards table.

Guestname – Guest name.

TableID – Identifier number of the table.

Score – Number of points to be given to the player.

Sort - Maximum table or score value.

Example:

```
GameJoltScoresAddGuestPoints Guestname TableID Score Sort
```


Plugin commands 2

GameJoltTrophiesAddUser

Add a trophy for the user.

Username – Username of Game Jolt.

TrophyID – Trophy ID number.

Example:

```
GameJoltTrophiesAddUser Username TrophyID
```

GameJoltTrophiesRemoveUser

Remove the user's trophy.

Username – Username of Game Jolt.

TrophyID – Trophy ID number.

Example:

```
GameJoltTrophiesRemoveUser Username TrophyID
```

GameJoltOpenWindowLogin

Call a graphical interface de login.

Example:

```
GameJoltOpenWindowLogin
```

GameJoltOpenWindowLogout

Call a graphical interface de logout.

Example:

```
GameJoltOpenWindowLogout
```

Script commands 1

\$gameTemp.gameJoltAddUser

Add a new Game Jolt user to your game.

Username(String) – Username of Game Jolt.

Game Token(String) - Special Player Password, used to access the games.

Example:

```
$gameTemp.gameJoltAddUser('DrXama', '4a934x5p');
```

The function will return true if the user registration is a success, and will return false if the registration is not successful.

\$gameTemp.gameJoltLoginUser

Start the user's session in the game.

Username(String) – Username of Game Jolt.

Callback(Function) - Function to be returned with the answer.

Example:

```
$gameTemp.gameJoltLoginUser('DrXama', success => { return console.log(success); });
```

The function will return true if the user login is successful, and will return false if the login is not successful.

\$gameTemp.gameJoltLogoutUser

Ends the user's in-game session.

Username(String) – Username of Game Jolt.

Callback(Function) - Function to be returned with the answer.

Example:

```
$gameTemp.gameJoltLogoutUser('DrXama', success => { return console.log(success); });
```

The function will return true if the user logout is successful, and will return false if the logout is not successful.

\$gameTemp.gamejoltScoresUserTable

Returns the score of the player in a certain **Scoreboards** table.

Username(String) – Username of Game Jolt.

TableID(String) – Table Identifier Number.

Callback(Function) - Function to be returned with the answer.

Example:

```
$gameTemp.gamejoltScoresUserTable('DrXama', '389349', score => { console.log(score); });
```

The user must be logged in to perform a search for his score.

Script commands 2

\$gameTemp.gamejoltScoresTables

Returns all **Scoreboard** tables.

Callback(Function) - Function to be returned with the answer.

Example:

```
$gameTemp.gamejoltScoresTables(tables => { return console.log(tables); });
```

The function returns a list (**Array**) with all the **Scoreboards** tables.

\$gameTemp.gamejoltScoresAddPoints

Adds the user's score to a **Scoreboards** table.

Username(String) – Username of Game Jolt.

TableID(String) – Table Identifier Number.

Score(Number) – Quantities of points to be given to the player.

Sort(Number) - Maximum value of the score.

Callback(Function) - Function to be returned with the answer.

Example:

```
$gameTemp.gamejoltScoresAddPoints('DrXama', '379357', 100, 45, success => { return console.log(success); });
```

The function will return true if adding the user score is successful, and will return false if adding the user score is not successful. The sort value is used to set a limit for the score value, that is, if you set that score to 100 and sort to 45, the value to be recorded in the table will be 45.

\$gameTemp.gamejoltScoresAddGuestPoints

Add the guest score to a **Scoreboards** table.

Guestname(String) – Guest name.

TableID(String) – Table Identifier Number.

Score(Number) – Quantities of points to be given to the player.

Sort(Number) - Maximum value of the score.

Callback(Function) - Function to be returned with the answer.

Example:

```
$gameTemp.gamejoltScoresAddGuestPoints('convidado001', '379357', 100, 45, success => { return console.log(success); });
```

The function will return true if adding the user score is successful, and will return false if adding the user score is not successful. The sort value is used to set a limit for the score value, that is, if you set that score to 100 and sort to 45, the value to be recorded in the table will be 45.

Script commands 3

\$gameTemp.gamejoltScoresGetRankTables

The function will return the **Scoreboards** score rating.

TableID(String) – Table Identifier Number.

Sort(Number) – Score to be ranked.

Callback(Function) - Function to be returned with the answer.

Example:

```
$gameTemp.gamejoltScoresGetRankTables('379357', 100, rank => { return console.log(rank); });
```

The function will return the **Scoreboards** score rating.

\$gameTemp.gamejoltTrophiesUser

Returns all available trophies to the user.

Username(String) – Username of Game Jolt.

TrophyId(String) – Trophy ID number.

Achieved(Boolean) - If the trophy was won by the user.

Callback(Function) - Function to be returned with the answer.

Example:

```
$gameTemp.gamejoltTrophiesUser('Drxama', "", true, trophies => {return console.log(trophies); });
```

The function will return a list (**Array**) with all the user's trophies. In **achieved** set **true** to return only the trophies won by the user. Set **false** to return only the trophies that the user did not get. Leave blank to retrieve all the trophies. In **trophyId** If you want to collect only one trophy, pass the trophy ID. If you do this, only this trophy will be returned in the answer. Setting a value for **trophyId** will ignore the **achieved** value.

\$gameTemp.gamejoltTrophiesAddUser

Add a trophy for the user.

Username(String) – Username of Game Jolt.

TrophyId(String) – Trophy ID number.

Callback(Function) - Function to be returned with the answer.

Example:

```
$gameTemp.gamejoltTrophiesAddUser('DrXama', '101982', success => { return console.log(success); });
```

The function will return **true** if adding the user's trophy is successful, and will return **false** if adding the user's trophy is not successful.

Script commands 4

\$gameTemp.gamejoltTrophiesRemoveUser

Remove the user's trophy.

Username(String) – Username of Game Jolt.

TrophyId(String) – Trophy ID number.

Callback(Function) - Function to be returned with the answer.

Example:

```
$gameTemp.gamejoltTrophiesRemoveUser('DrXama', '101982', success => { return console.log(success); });
```

The function will return **true if removing the user's trophy is successful, and will return **false** if removing the user's trophy is unsuccessful.**

\$gameTemp.gameJoltOpenWindowLogin

Call a graphical interface de login.

Example:

```
$gameTemp.gameJoltOpenWindowLogin();
```

\$gameTemp.gameJoltOpenWindowLogout

Call a graphical interface de logout.

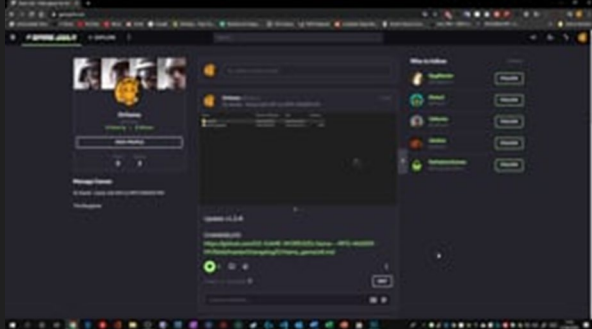
Example:

```
$gameTemp.gameJoltOpenWindowLogout();
```

Game Jolt API 1

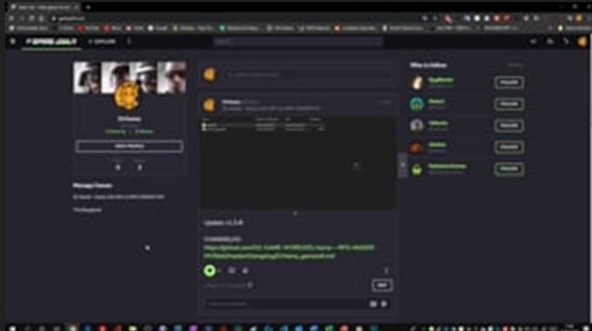
Game Token

See below how you get it.



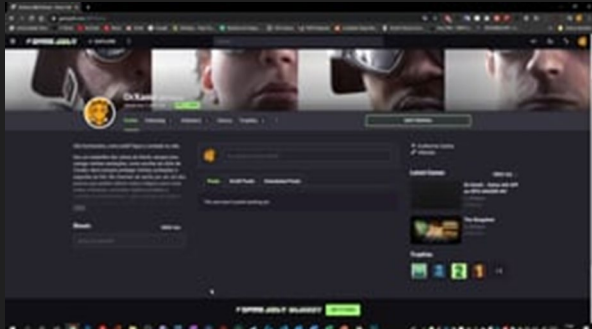
Game ID and Private Key

See below how you get them.



Trophies

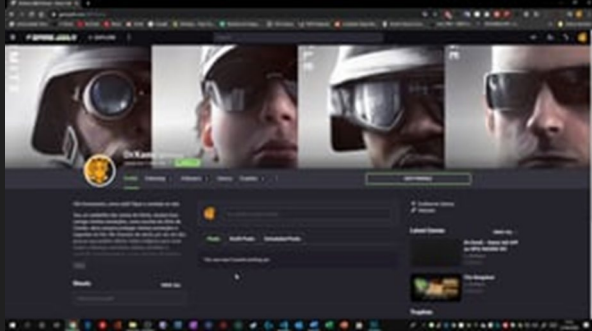
See below how you get them.



Game Jolt API 2

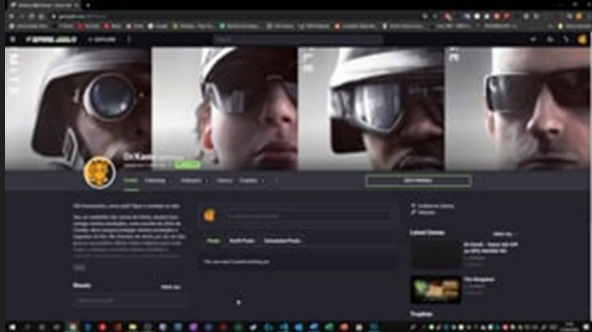
Scoreboards

See below how you get them.



Data Store

See below how you get it. Only the global values will be displayed.



Video links

Game Token

https://player.vimeo.com/video/410360188?app_id=122963

Game ID e Private Key

https://player.vimeo.com/video/410361795?app_id=122963

Trophies

https://player.vimeo.com/video/410363534?app_id=122963

Scoreboards

https://player.vimeo.com/video/410365417?app_id=122963

Data Store

https://player.vimeo.com/video/410367002?app_id=122963

Contact us

Email 1

contato@drxama.com.br

Email 2

spgamesoficial@gmail.com

Publicity

Site

<https://drxama.com.br/> ou <https://drxama.com/>

Canal do Youtube

<https://www.youtube.com/channel/UCIL322VJEbtqIt53BLcn80w>

Github

<https://github.com/GS-GAME-WORDS/Dr.Xama---RPG-MAKER-MV>

Twitter

https://twitter.com/dr_xama

Discord

<https://discord.gg/bvS9fgB>