How to set up games in DrEcco website

1. Add new games

For security reason, we only support uploading games on server side. To add a new game, you must have access to CIMS server.

We provide a python script *games/add_game.py* to help you upload your game. To successfully upload your game, you should follow instructions:

    i. Compress all your codes include template *index.php*(details can be found in section 2) in a zip file. For packed code: it must be firstly put into a folder then be compressed into a zip file with the same name of the folder. **Make sure the folder name(which you compress) is the same as the zip file name**.

    ii. Upload your zip file(via *scp*) to *games/* directory.

    iii. Then run *games/add_game.py*, arguments are as follows, you can change the default if needed. **This script should be running on *games/* directory**
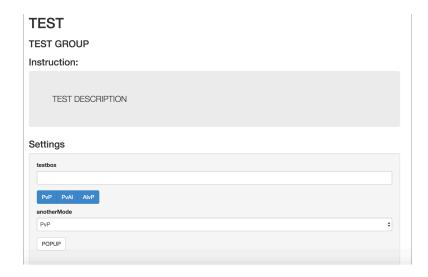
```python
parser = argparse.ArgumentParser()
parser.add_argument('-n', type=str, default='warehouse.cims.nyu.edu', help="hostname of the MySQL server")
parser.add_argument('-d', type=str, default='by653_adba2', help="name of the database")
parser.add_argument('-u', type=str, default='by653', help="id of the database admin")
parser.add_argument('-p', type=str, default='adba2', help="passwd of the database admin")
parser.add_argument('-gn', type=str, help="name of the game")
parser.add_argument('-gp', type=str, help="folder name of the game")
```

    differences between **-gn** and **-gp**: **-gn** is the name actually showed in the website's menu, whilst **-gp** is the folder name of the game: in this case, it is preferable that **-gp** does not contain any spaces.

    iv. Make sure you set the right **game name(-gn)** and **folder name(-gp)** on the database. If your game is successfully uploaded, refresh the website and the **game name** should be shown on the left panel.

2. Template

We provide a template (located in *games/empty/index.php*) for you to better display your game on the website. The template looks like this:

js/gameSettings.js provides a set of functions that can be used to generate customize game setting panels. You can check previous year's games *index.php* for reference.

The POPUP button will generate a popup window to display your game GUI.

Usage:
- Insert your game page into the iframe tag of the given template php games/*empty/index.php*, then create some <div>s with ID in your game page to receive values from these functions.
- If you do not want your game to be displayed in popup window, just ignore newWindowBtn function in games/*empty/index.php*

3. Essential APIs

We provide 3 APIs to deal with game states and scores.

saveData API

Source code is in *dbman/saveData.php*. If your game need to save intermediate state between each new games, saveData provides a VARCHAR(255) to save that state. It returns 200 if save success; 404 if username does not exist.

Usage:
- Send GET request to *dbman/saveData.php*

Parameters:
- type: 'save' or 'load'
- user: username of the current player
- game: game name
- data: a string less than 255(only for type=save)

---

saveScore API

Source code is in *dbman/saveScore.php*. saveScore provides a VARCHAR(255) to store the game score. If your game does not have an actual score, just send "WIN", "LOSE" or "TIE". If your game have several roles, say Hunter and Prey, please come up a single score which makes sense to all roles.

Usage:
- Send GET request to *dbman/saveScore.php*

Parameters:
- gamename: name of your game, should be the same as your zip folder name
- playername: default player1/2/3 or any name user sets in this game
- score: "WIN", "LOST", "TIE" or a numeric score.

Attention:
For saveData and saveScore API, make sure you send GET request to the right relative path if you call these functions in your game logic file.

For relative path, we use *../../* as the base for all *html/php/js* files in the game folder, which can be seen in the template *index.php*. For all *js* files included into the *index.php* and after <base> tag, it is safe to assume the base is website folder (i.e. the root of the website);
for any *js* file that has not been included in the *index.php*, please manually add relative path corresponding to its own location relative to the root (website folder) rather than directly use */dbman* when calling database related APIs.

---

getScore API

Source code is *getScore.php*. This function is to display leaderboard. You should include this in your game's *index.php*, like this:

```
<div id="scoreArea", class="jumbotron">
<?php
    include $base."getScore.php";
    /*
    * arg1: gameName, should be the same as the dir name
    * arg2: if your score is sortable, pass 1 if higher score is better, 0
    *        if smaller score is better. Otherwise no need to pass variable
    */
    getScore("DigThat", 0);
?>
</div>
```

Parameters:
- gameName: your game name
- orderFlag:
  If your game does not have sortable score(e.g. "WIN" or "LOSE"), just ignore this argument.
  If your score is sortable, pass 1 if higher score is better, 0 if smaller score is better.

If you still feel confused about set up a game GUI, feel free to check existing games for reference.