Does This Game Suck Classes and Method Summary

class Login

Login()

constructs connection from login initialization to a database

private int getcount()

returns an integer of the number of accounts

private String getRole()

returns a String of the role of an account

private void initialize()

creates the UI interface of the login screen with active redirection selections

class Reg

Reg()

creates the UI interface of the registration screen with active redirection selections and database submission properties for new accounts

class search

search(int userid, String usern)

creates the UI interface for the search homepage, including links to a specific user's wishlist

private void topTrending()

adds game titles to a Jlist, ordering by count of reviews (high to low)

private void toprate()

adds game titles to a Jlist, ordering by the rating value of the game (high to low)

class Game

Game(int gameid, int userid, String usern)

creates the UI interface for the game display page using the given gameid, while offering a connection point to the user's account

private void gamename()

stores the game name pulled from a database by gameid

private String years()

returns a String of the release date of a game from a database by gameid

private void data()

stores the description of a game pulled from a database by gameid

private void getpublish()

stores the publisher of a game pulled from a database by gameid

private String link()

returns a String of the YouTube link of a game

private void updatarating()

updates the total rating of a game after a user submits a review

private String getrate()

returns a String of the total (average) rating of a game

private boolean exist()

returns true if a user's wishlist already includes a specified game

private boolean isadd()

returns true if a user has already written a review for a specified game

private void addreview(int rate, String comment)

adds a review to a specified game

private void search()

takes UI back to the search page from the game page

class Review

Review(int gameid, String gname, int userid, String usern)

creates the UI for a review page for a specific game (verified by id and name) and storing reviews in a specified game's database table

private String getusername(int userid)

returns a String containing the username of a user by userid

static int getRating(int gameid)

returns an integer value of a specified game's total (average) rating

private String gettext()

returns a String of the body text of the review from a game database

class Developer

Developer()

creates the UI for a userRole developer to interact with

private void getrequest()

stores a game title from the request database table in a Jlist

private void blockUser(String usern)

changes the userRole of user usern to 'block'

private void unblockUser(String usern)

changes the userRole of user usern to ‘User’

private void getusers()

store usernames from a database to a Jlist

class WishList

WishList(int userid, String usern)

creates the UI for a WishList screen while supplying action events to add and remove games from the specified user’s WishList

class connect

public static Connection dbConnector()

returns a Connection to a specified database

class request

request(int userid, String usern)

creates the UI for the game request form for the user to interact with

private void insertRequest(String gameTitle)

inserts the specified game title into a gameRequest database table

class searchResult

searchResult(int userid, String usern, String title, String result)

creates the UI for the searchResult screen

public void getresult()

stores game titles to a Jlist in an order depending on rating or releaseDate, depending on which the user has selected