# Table creation code

CREATE\_TABLE Game (Title VARCHAR(100) NOT NULL, Descript VARCHAR(500), Developer VARCHAR(100), Version Float, Link VARCHAR(300), Price INT, Img VARCHAR(300), PRIMARY KEY(Title))

CREATE TABLE Genres (gTitle VARCHAR(100) NOT NULL REFERENCES Game(Title) ON Delete Cascade, Genre VARCHAR(50), PRIMARY KEY(gTitle,Genre))

CREATE\_TABLE Platform (gTitle VARCHAR(100) NOT NULL REFERENCES Game(Title) ON DELETE CASCADE, Platform VARCHAR(100), PRIMARY KEY(gTitle, Platform))

Create Table UserRequests(gTitle VarChar(100), Descript VarChat(500), userName varchar(50) NOT NULL, reqID int NOt Null AUTO\_INCREMENT, Primary Key(reqID));

Create Table User(userName varchar(50) NOT Null, Password VarChar(50) NOT NULL, Email VarChar(100) NOT NULL, Primary Key(userName));

# PHP Code

## Example Connect:

<?php

//ip address for servername

//if that doesn’t work

//phrasal-concord-292413:us-central1:mustain-99

$servername = "104.154.66.238";

$username = "root";

$password = "9Reddolphinscarvetoast!";

$dbname = "Project";

// Create connection

$conn = new mysqli($servername, $username, $password, $dbname);

// Check connection

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

echo "Connected successfully";

?>

## Example Insert:

$sql = "INSERT INTO table (column, column, column)

VALUES (values, values, values)";

if ($conn->query($sql) === TRUE) {

echo "New record created successfully";

} else {

echo "Error: " . $sql . "<br>" . $conn->error;

}

$conn->close();

?>

## Example Select:

//should return the stored password of a specified userName

$sql = "Select Password From User Where userName='userName of Person logging in';";

$result = $conn->query($sql);

if ($result->num\_rows > 0) {

// output data of each row

while($row = $result->fetch\_assoc()) {

echo "id: " . $row["id"]. " - Name: " . $row["firstname"]. " " . $row["lastname"]. "<br>";

}

} else {

echo "0 results";

}

$conn->close();

?>

## Join Example:

CREATE TABLE person ( `id` INT NOT NULL PRIMARY KEY, `name` VARCHAR(50) ); CREATE TABLE fruits ( `fruit\_name` VARCHAR(20) NOT NULL PRIMARY KEY, `color` VARCHAR(20), `price` INT ); CREATE TABLE person\_fruit ( `person\_id` INT NOT NULL, `fruit\_name` VARCHAR(20) NOT NULL, PRIMARY KEY(`person\_id`, `fruit\_name`) );

SELECT p.\*, f.\* FROM person p INNER JOIN person\_fruit pf ON pf.person\_id = p.id INNER JOIN fruits f ON f.fruit\_name = pf.fruit\_name

output:1 | "banana" 1 | "apple" 1 | "orange" 2 | "strawberry" 2 | "banana" 2 | "apple"

## Example Search specific:

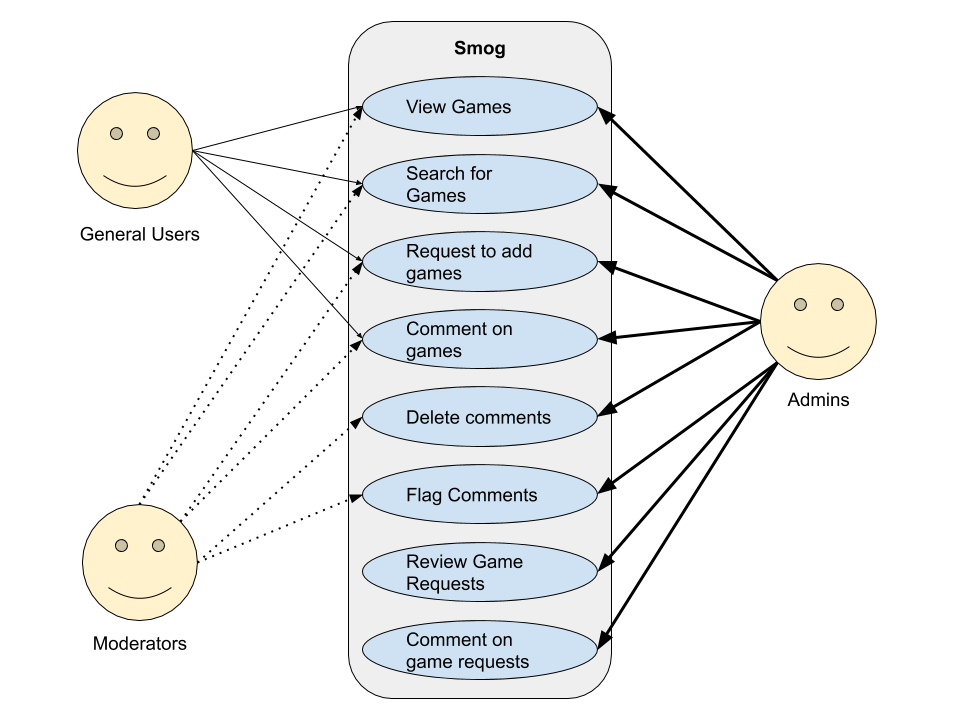
Select attributes you want from Game, Genres, Platform

## Example Non Specific Search:

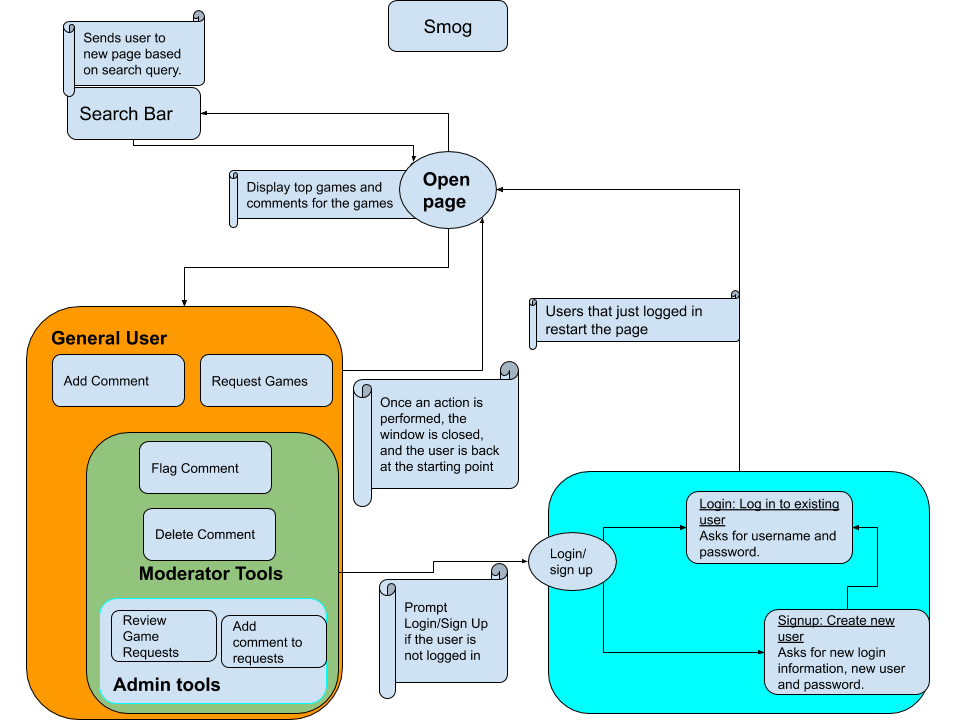
Select Title from Game Limit 0, 10;

# Diagrams

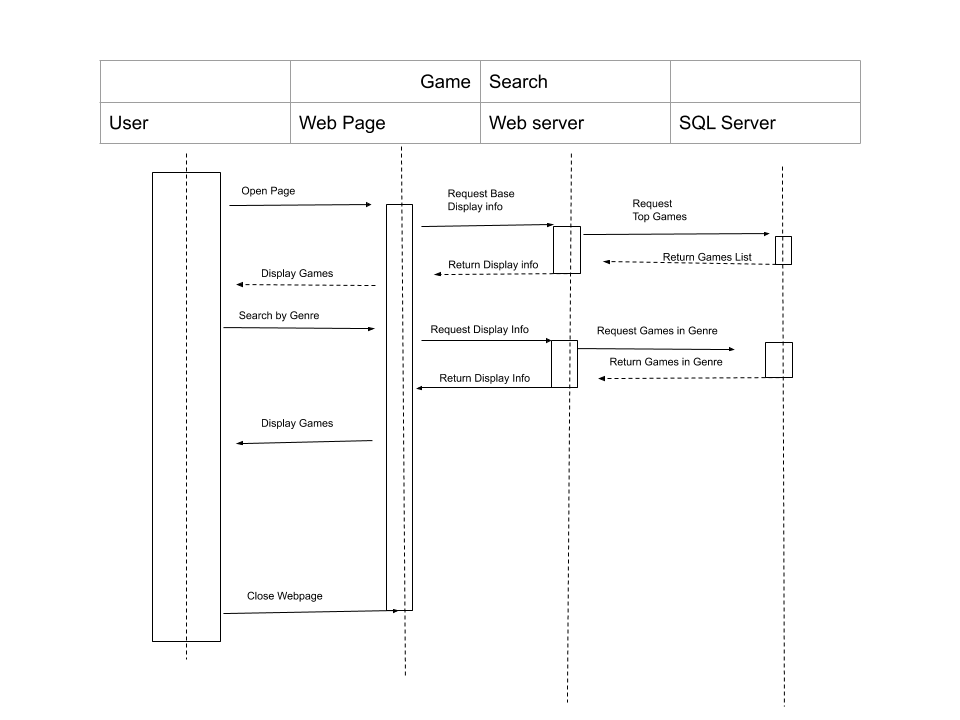
## User Diagram



## State Diagram



## Sequence Diagram



Initial UML Diagrams.

|  |
| --- |
| User |
| Variables:  userName : string  password: string |
| Methods:  login(String user,String pass) : boolean/int  comment(String s) : boolean/int  requestAddGame(Game game) : boolean/int |

|  |
| --- |
| Moderator |
| Variables: |
| Methods:  deleteComment() : boolean  flagComment() : boolean |

|  |
| --- |
| Admin |
| Variables:  String Username  String Password |
| Methods:  reviewGameRequests() :  addCommentToRequest(“String”) : boolean |

|  |
| --- |
| Game |
| Variables:  String Name  String Rating  String description  String[] Genres  String Organization  String[] Platforms  String Version  String LinkToStore  Float Price  String LinkToImage |
| Methods:  addComment(Comment c) |

|  |
| --- |
| Comment |
| String User  String Comment |
| saveComment(String comment) : boolean |

|  |
| --- |
| Website |
| String baseurl |
| search(string) boolean  goToPage(String pageURL) : boolean  createUser(string, string) boolean  login(string, string) boolean |