

# Gamescore

## Systemöversikt

### projektets mål

Att bygga en webbapp som låter användare logga in för att recensera spel och dela med sig av sitt betyg till andra. Användare skall även kunna se nyheter som spelen och ta sig in till spelen via länkar i nyheterna. Det finns även en "wall" där man kan göra inlägg och även kommentera andras inlägg. Det ska finnas en sida där man får en lista på alla spel

### projektets funktioner

Registrera användare, logga in, logga ut, ta bort konto, uppdatera till adminkonto, skicka vänförfrågan, acceptera vänförfrågan, lista vänner, läsa in spel till hemsidan, lägga till spel i databasen, hämtar in spel med högst betyg, lista alla spel, dynamisk sökfunktion till spel, intro skärm med easter egg, intro skärm med animation som tar en till webbappen, lista nyhetsartiklar, nyhetsartiklar med länkar hemidor, skriva och posta inlägg, skriva och posta kommentarer på inlägg.

### projektets arkitektur

Varje menyval är en egen php-fil som man kommer in till. Det finns controllers och DAO'ar som tar hand om backend delarna som inte visas i views. Vi har jobbat på att uppnå en bra mappstruktur som är praxis.

# Teknisk dokumentation

## **MVC**

- 1. Model**
  - 1.1. DAO - Alla anrop till databasen hanteras här.
  - 1.2. Entity - Objekt med getters och setters.
- 2. View**
  - 2.1. Pages - Visar allt innehåll för användaren.
- 3. Controller**
  - 3.1. Validation - Felhantering samt instansiering av objekt och användarfunktioner.

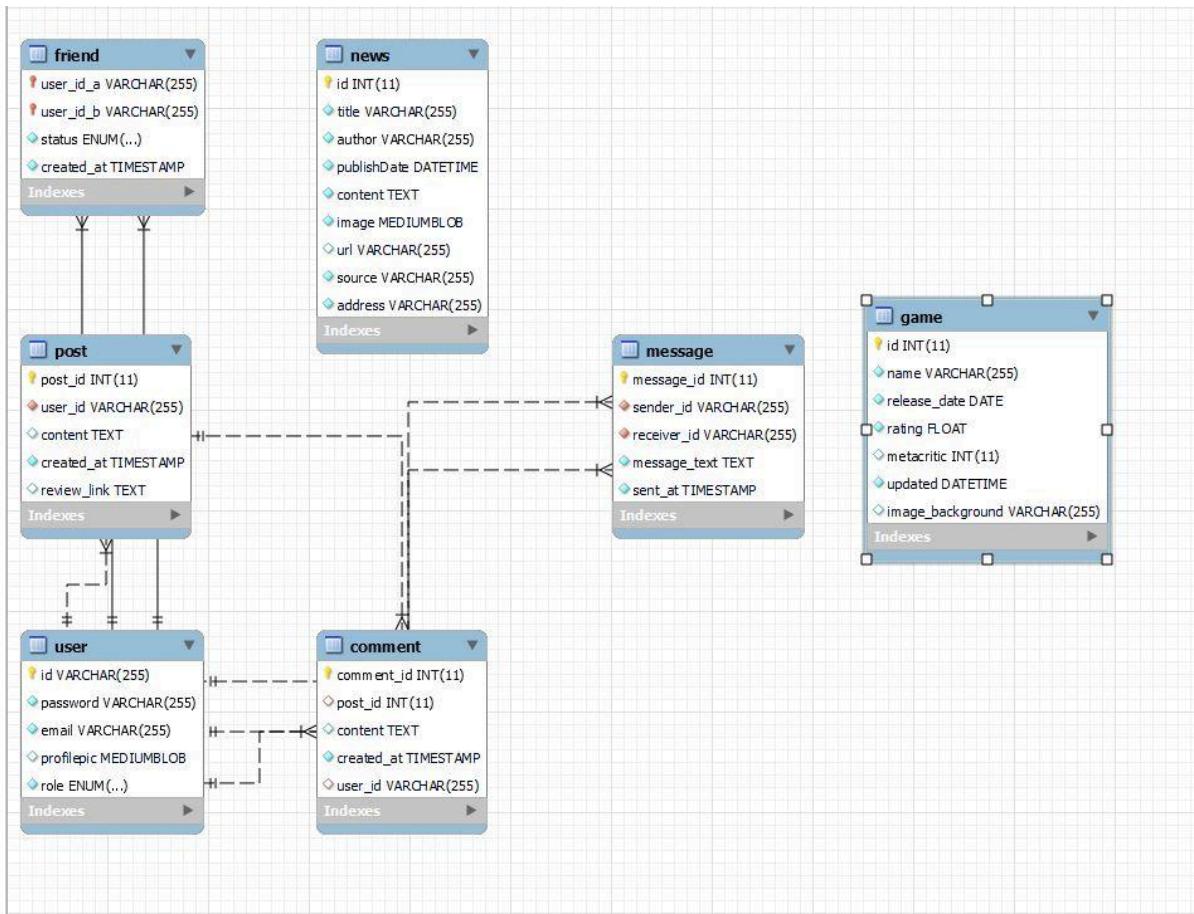
## **Config**

1. Skapar uppkopplingar.
2. Autoloader inkluderar samtliga php filer.
3. Hantering av cookies.
4. Hämtar username/role från cookies.

# använda teknologier

1. Windows/MacOS - arbetat på sidan från båda operativsystemen. MacOS har använt sig av en virtual machine “*parallel*”.
2. Visual Studio Code - huvudprogram använd för att skriva koden.
3. XAMPP - använd för att köra lokal server för att testa hemsidan
4. PHPMYAdmin - huvudprogram för att hantera databasstrukturer. (MacOS)
5. MYSQL Workbench - Windows huvudprogram för att hantera databasstrukturer. (Windows)
6. Chrome - Använt för debugging och visat innehåll (Windows)
7. Microsoft Edge - Använt för debugging och visat innehåll (MacOS)
8. Chat GPT 4.0 - Fått tips om kodstruktur, felsökning, genererat bild till logo,
9. PHP - huvudkod, för funktionalitet på hemsidan, sparar cookies, hanterar databasanrop.
10. HTML - Struktur på hemsidan.
11. CSS - Stilen på hemsidan.
12. Javascript -
13. C# - för teknisk testning av syntaxer samt utlägg av språk.
14. AJAX - använd för att få pop-up funktioner.
15. Bootstrap - Strukturen på HTML/CSS koden.
16. VanillaTilt - Kort design, grafiskt funktionell bibliotek med kalkyleringar som behövs.

## databasstruktur



tredjepartstjänster.

1. Rawg.io api

# Installationsguide

### Skapa databasen:

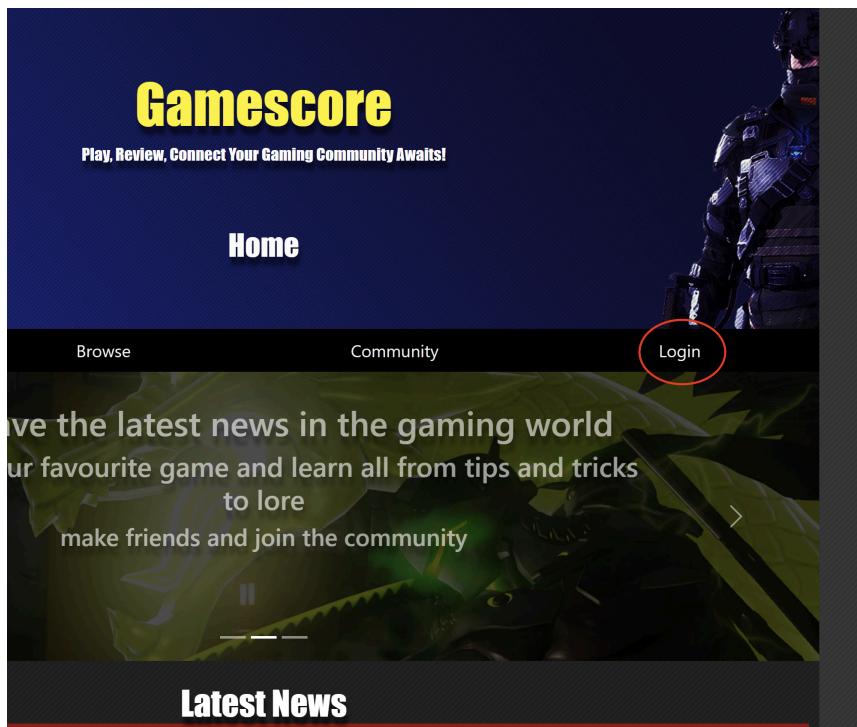
vi har gjort ett script som kontrollerar om databasen "Gamescore" finns annars körs scriptet automatiskt när man är inne på sidan <http://localhost/BUV23PHP-slutuppgift> förutsatt att projektet ligger i din htdocs mapp. Funkar inte detta finns scriptet i mappen db\_script med namnet: gamescore\_create\_db.sql

### Laddar in spel från databasen.

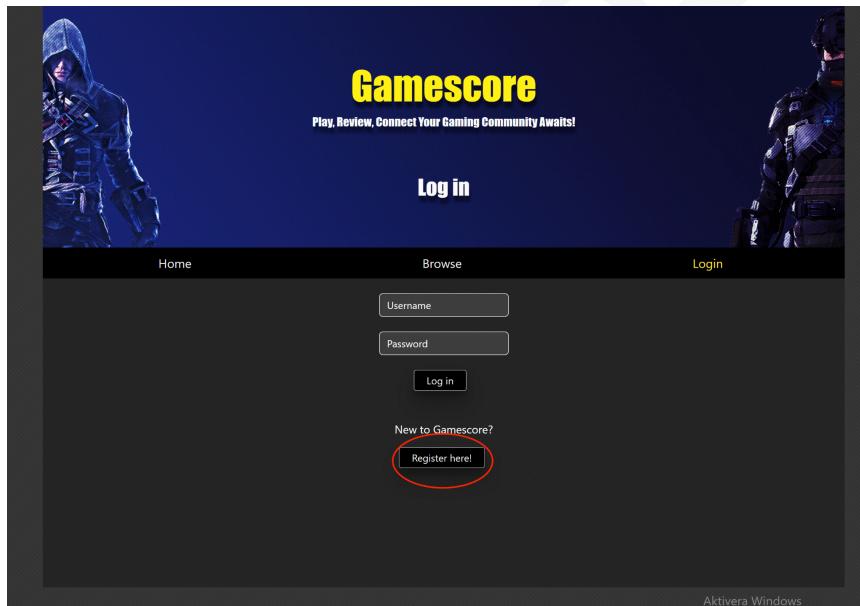
Registrera ett användarkonto, logga in och navigera till profilsidan (hamnar där efter login) som är ditt användarnamn i navbaren.

**step 1:**

Click on the Login button as shown.

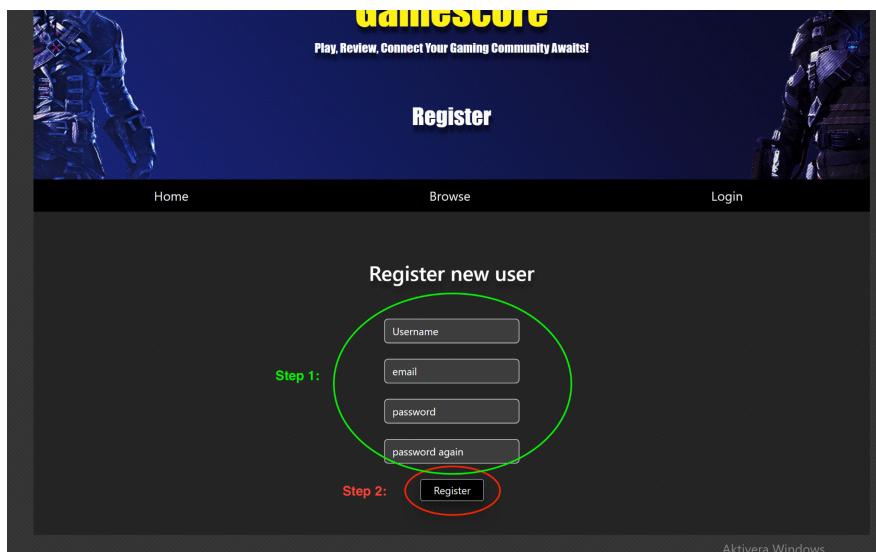
**step 2:**

Then press the “register here!” to arrive at account creation page

**step 3:**

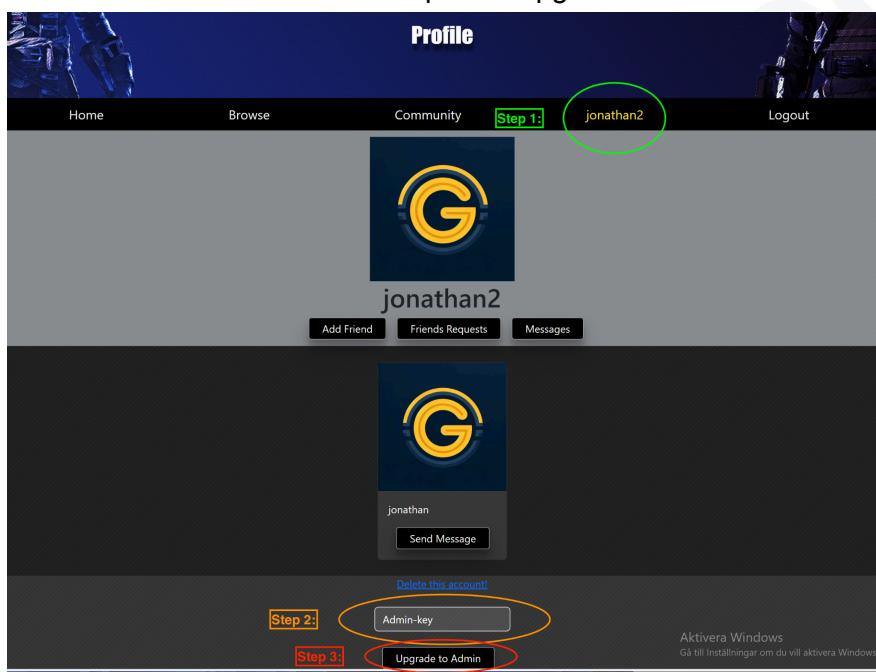
Fill the designated fields with the needed information.

Then proceed to press “Register”.



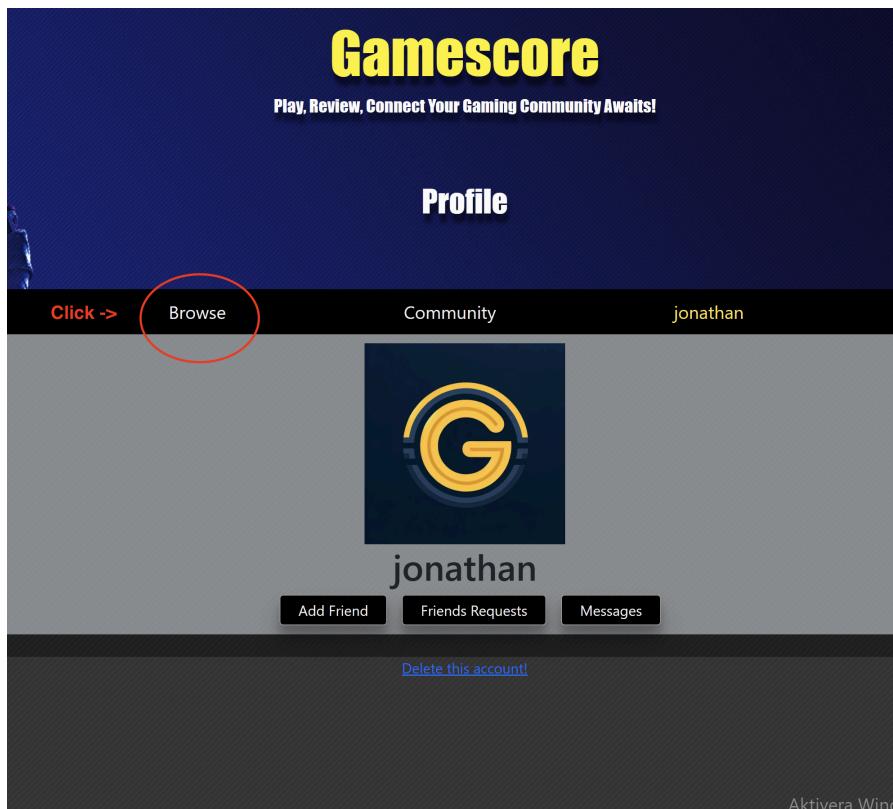
#### step 4:

Click on your username in the navigation-bar then proceed to click the Admin-key box and write “alfakrull” inside of it. Then press “Upgrade to Admin” button



#### Step 5:

Click on “Browse”



### Step 6:

Click “Import Games From Json” To convert the games to the database from json and to be able to view them.

