# Brief indroduction of my website:

**https://35.188.201.42/index.php**

This is a prototype of movie recommender system that I’ve built by using collaborative filtering methods. It attempts to predict the preferences of a user, and make suggests based on these preferences.

## Dataset

I use MovieLens Latest 100 k Dataset for education as our dataset.

It contains 100004 ratings and 1296 tag applications across 9125 movies. These data were created by 671 users between January 09, 1995 and October 16, 2016. Users were selected at random for inclusion. All selected users had rated at least 1 movie.

## Data Pre-processing

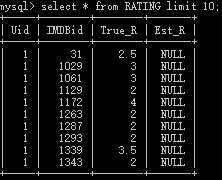
Upload all the data to the database.



## Recommender System

***SVD++ Algorithm***

I use SVD++ algorithm to get the estimated ratings of the movies that the user has not watched yet.



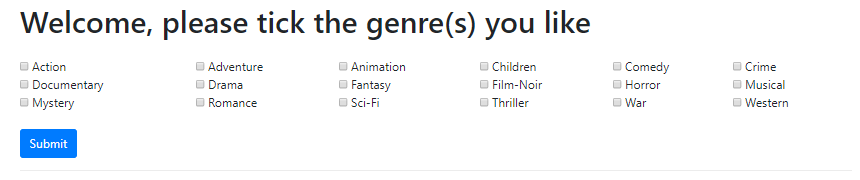
True ratings of user watched movies



After running the algorithm, we got the estimate ratings of user unwatched movies

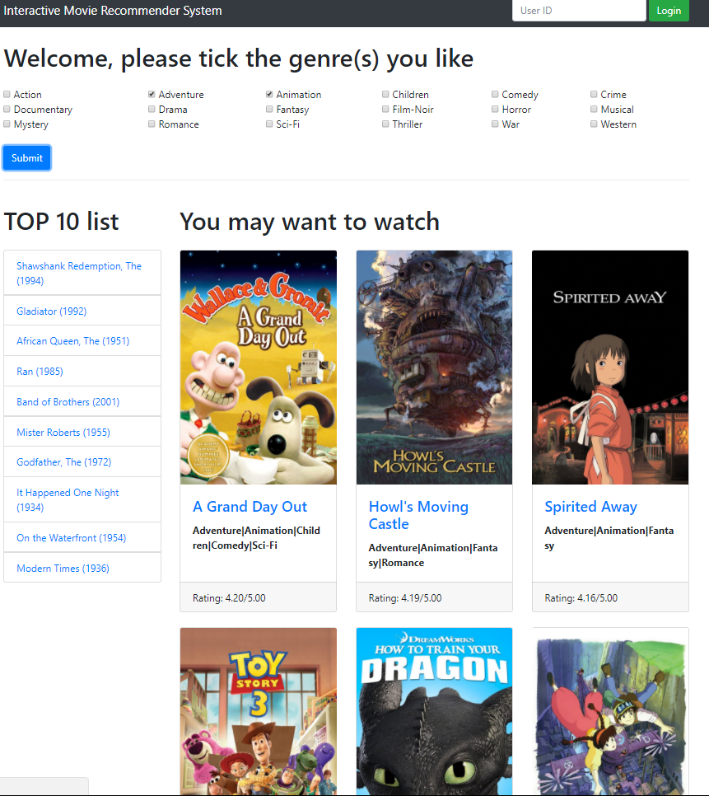
## New User and Non-Login User Interface

For the new user and non-login user, we first provide a checkbox panel to let them select the genres they like. The checkbox area is shown in figure:



After the user ticks the checkboxes and presses Submit button, this webpage will send http POST request to the Server, using jquery **AJAX**. When the server receives the request, it will select the Top 9 highest overall rating (Rating column in MOVIE table) movies that meet the genre requirements:

What is more, after successfully fetching the TMDBid from database, the server will also connect to TMDB website using **CURL** and get the movie’s poster url. Finally, the server will send the TMDBid, title, genres, rating and poster url back to client’s webpage in json format then the webpage will parse and show them to the user. This time the webpage looks like figure:

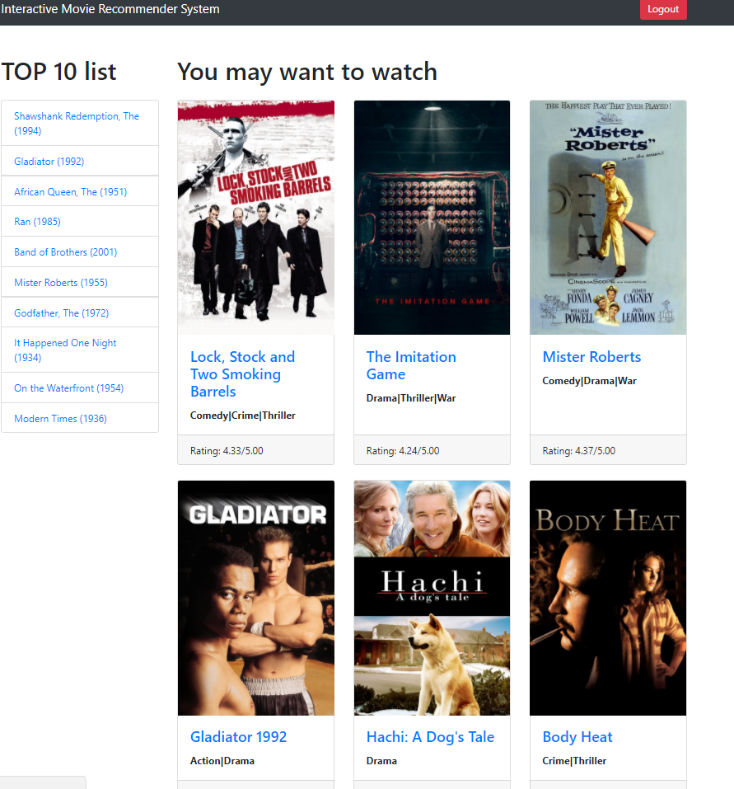


All the recommendations are shown under “You may want to watch” title, which contain the poster, movie title, genres and the overall rating given by our recommender system.

## Pre-existing User Interface

Due to the time limit, I didn’t implement the real login function. But we still have a login area to mimic the real login utility. You can treat this prototype as only for administrator use. The administrator can type the user id and press login button to jump into that user’s recommendation result page.



The recommendation for a pre-existing user is based on that user’s estimate ratings on the movies he has not watched yet. Those estimate ratings are provided by our recommender system model. What the server does here is just sort the user’s estimate ratings and select the 9 movies corresponding to the TOP 9 highest estimate ratings. Also, the server will connect to TMDB website and get the movie’s poster URL. Finally, the server will send the TMDBid, title, genres, rating(movie’s overall rating) and poster URL back to client’s webpage in JSON format then the webpage will parse and show them to the user. The pre-existing user webpage looks like figure:

All the recommendations are shown under “You may want to watch” title, which contain the poster, movie title, genres and the overall rating given by our recommender system.

## Movie Details Interface

For each movie, you can go to the movie details page by clicking either the movie title or the movie poster. Then the server will connect to TMDB website and fetch the movie details through its TMDBid. After that the server will sent all the details back to the client’s webpage. It looks like figure below:

