DB project

**Movies:**

Jinni Experiment:

* on login, it asks you for a couple of favorite "genres"
* it then asks you to rate movies
* according to this rating, it established your prefrences.

DATA Sources - movies:

What YAGO gives us:

we saw in the project that players, movies, directors, links between them and so on exist in yago, but and IMDB parser is required for more data:

movie details - presantation:

* outer links to IMDB, for full movie details.
* name, year, genres, some sort of rating (is imdb rating available?)
* summary
* director, writers, actors (starts)
* runtime, productio company, country;

issue: how to get all imdb movie files? there are some APIs online, need to check that. http://www.omdbapi.com/.

MOKA: a tagging service that imdb has for movies. it also assigns tags to movies themselves - perhaps not available, maybe write a bot to download it, manually. too many tags, choose more relevant ones. more like " Semantic".

somehow STEAL jinni's TAGS. more like pandora's "musi genome"

**Users:**

* like jinny, we want to establish a good sense of what the movie is about - we want to take the tags, and allow users to rate the tag regarding the movies - so, we will have like Jinny a rating of what the movie is about.
* a Question for Yael - can we give some tags and ratings in advance, as part of the load.
* initial login - ask for General Genre's
* start asking for movies, and according to the 3 highest ranked tags in each movie - users
* rate movies - we will probably have outside rating (IMDB? ROTTEN TOMATOES?), and also user rating - allow users to rate.
* like/dislike - per user per movie. will not influence tag-rating.
* crowdsourcing.

**recommandations:**

* similar users - users who have a similar taste to mine, liked some items, and they should be proposed to me - requires user tastes, and user likes/ratings. - VERY HARD!!!!! **K-Neigbhours, collaborative filtering, similarity, etc.**
* similar content - I have a taste, and movies are assigned tags by users, and similar content .
* social network - likes determined by friendship.
* per item - users who liked this, also liked, which is simpler than bullet 1.

**what are the functions of the user interface**

* movie window: tags movies, and give tag ratings, reccomend to friend, like, dislike, twit, rate.
* search.
* manage friendship - give user name and add as friend (unfried? need to accep friendship?)

### GUI

There should always be a progress bar for DB queries

login window

* username, password, usual stuff

Overview Window

* first window launched in GUI, shows data from all User Functionalities - mostly recent things
* Your taste -what we think the user likes
* recent Activity - what you tagged, rated, recently
* friends - recent friendships, requests
* several recomdations.

Search Window

issues: auto complete? narrow searches down? how does this effect efficiency.

* search by title
* search genre
* search by rating (our? outer rating?)
* search by actors
* search by tags, + narrow down.

Social Window

* friend requests inbox.
* friends' activities. (newsfeed: what they rated and tagged)
* find/add friends (autocomplete)
* remove friends

Recommendation window

* reccomadations according to your taste - "We think you'll like this"
* reccomdations according to friends "your friends likes this"
* recommadations according to likes - people who liked this, also liked.

Search Results window

we move to this window whenever a list or one movie answer some criteria (my friends liked, my taste is, a single movie was selected)

* a lists (perhaps empty or one item), of general details and TAGS, and a main display - when a movie from the list is clicked, the display shows its data.

### Schema

views:

how do we determine user prefrences? we see all movies he liked, and the amount of starts each tag got. we sum them, accordingly to his rating, and get a TAG-USER-PREFERNCe Value