PrivateRecSys

Final Report

29th September, 2022

Table of Contents

Introduction

Privacy Recsys is a system that utilizes privacy preserving approaches to deliver privacy preserving recommendations. The current implementation supports recommendations of Movies based on either the historical ratings (as an independent system) or based on the search queries of the user at the Searx search engine.

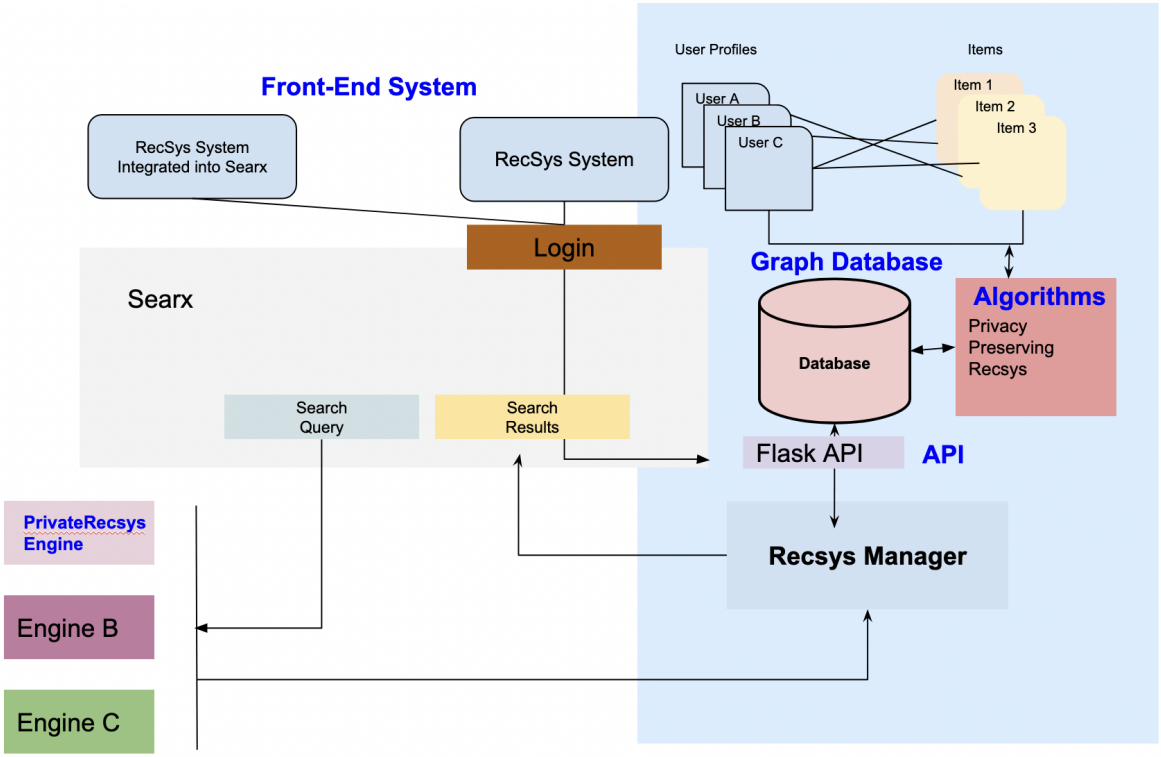
This deliverable presents the PrivateRecsys system as a standalone and as an integration into Searx. For both implementations, the system architecture, the components and functionality are presented. Also this deliverable provides a User Manual guide - presenting how to build and run the systems.

The deliverable concludes with steps for the future work section consisting of ideas for further expanding the system and underlines the improvements to be undertaken for PrivacyRecsys.

This guide can be useful to anyone interested in using or extending the PrivateRecSys project, integrating the system to other applications or building other integrations for the Searx system.

PrivacyRecsys System Architecture

The PrivateRecsys system was developed to deliver recommendations while respecting the privacy of the user. Important components are the graph database which holds the entities and the relationships between them, the privacy preserving algorithms and the front end. This section presents the architecture of the system and provides an analysis of the components.



PrivacyRecSys Components Analysis

Differential Privacy based recommendations.

Differential privacy aims to provide means to maximize the accuracy of these statistical queries while minimizing the chances of identifying its records. It introduces noise to real data so that, adding or removing one user to database does not make noticeable difference in the data, thus preventing to identify his/her private information. It is a probabilistic concept, therefore, any differentially private mechanism is necessarily randomized with Laplace mechanism, exponential mechanism etc.

Let ε be a positive real number and A be a randomized algorithm that takes a dataset as input (representing the actions of the trusted party holding the data). The algorithm A is ε-differentially private if for all datasets D_1 and D_2 that differ on a single element (i.e., the data of one person), and all subsets S of image of A.

$$Pr[A(D_1) \in S] \leq e^{\varepsilon} \times Pr[A(D_2) \in S]$$

where the probability is taken over the randomness used by the algorithm.

Differential privacy is used in this implementation of the privacy recsys to preserve the privacy of the uses and make sure that no one can determine the exact rating a user provided to a movie, or what movies a user has rated.

Experimental Results

To measure the error of the recommendation algorithm, and determine the privacy/utility ratio of the recommendation we undertake a small evaluation based on thequery "What is the number of ratings given to a movie? (movie name as parameter) considering all the data in our dataset" is shown. This is the fundamental question for testing the effectiveness of differential privacy - according to which you should never be able to determine how many users have contributed for this rating and therefore removing all ratings -1 will not return the rating of the one user that remained. The Algorithm is run three times and all results are included to the table. The difference in the results is attributed to the randomness of the algorithm, and the application of differntial privacy as noise.

Movie ID	Actual Rating Count	Noisy Rating Count (exp-1)	Noisy Rating Count (exp-2)	Noisy Rating Count (exp-3)
5	51	61.53068160858365	4.14780634816978	45.74972097598749
8	8	23.754713618664823	4.5610873458316625	-30.184266258323568
9	16	3.8271299734101802	16.430418771569318	5.221803446422436
11	72	80.61785730156829	68.2613306805804	90.99006185916669
210	4	12.93616669892703	-1.441247061146698	-26.601274325745656

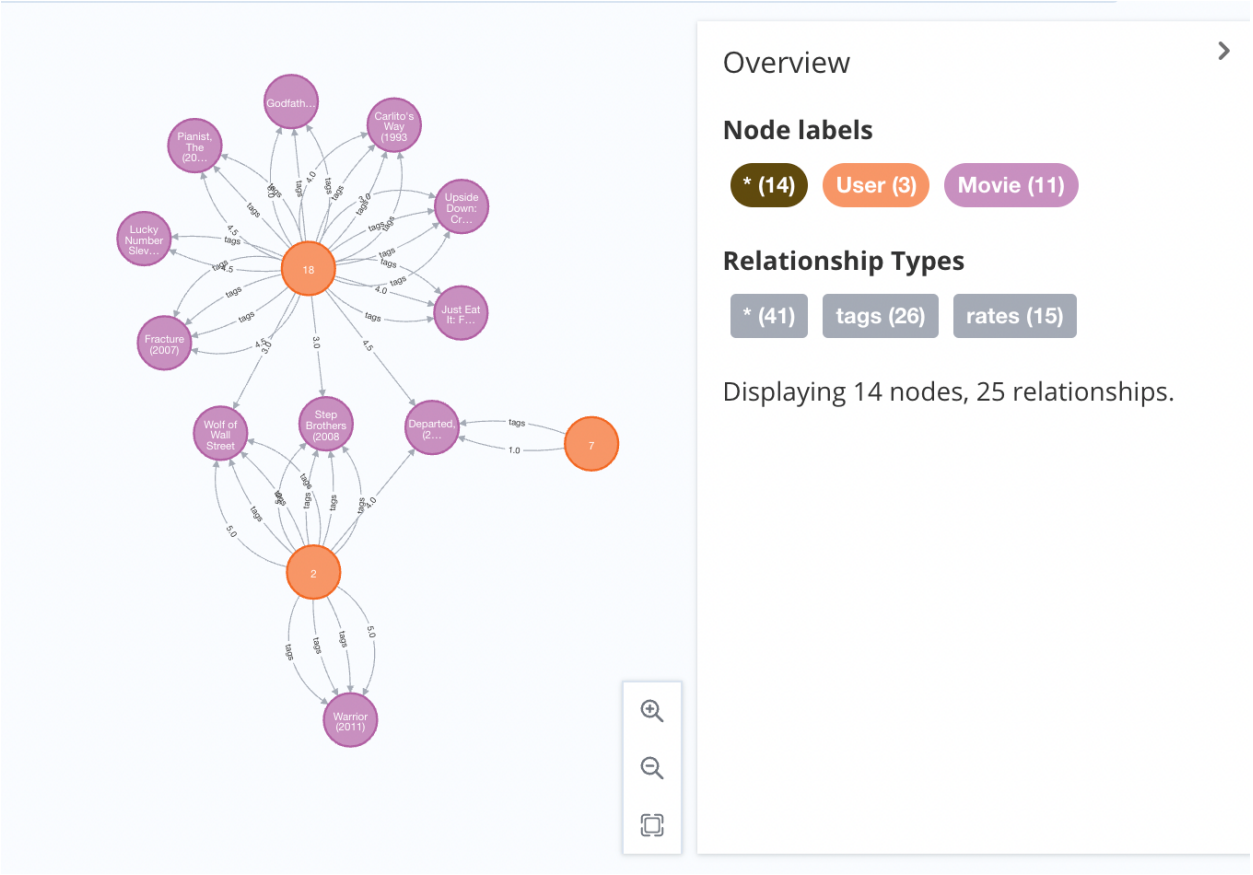
Graph database

A graph database (GDB) is a database that uses graph structures for **semantic queries** with nodes, edges, and properties to represent and store data.[1] A key concept of the system is the graph (or edge or relationship). The graph relates the data items in the store to a collection of nodes and edges, the edges representing the relationships between the nodes. The relationships allow data in the store to be linked together directly and, in many cases, retrieved with one operation. Graph databases hold the relationships between data as a priority. Querying relationships is fast because they are perpetually stored in the database. Relationships can be intuitively visualized using graph databases, making them useful for heavily interconnected data

In the implementation of PrivacyRecsys the Neo4J graph database is used, to store the data regarding the users and the items, and also semantically annotate the relationships between them. We use the relationship “rated” to connect Users and Items (in the case of the example used “Movies”.

Beyond Neo4J being a graph database, which makes it ideal for this project, also offers free-usage remote sandboxes which you can use to make the database you are working available to the web so other people can test it.

The system was developed based on a publicly available and free to use movies dataset.



API

API which stands for Application Programming Interface os concept that applies everywhere from command-line tools to enterprise code, microservices, and cloud-native architectures.. An API is an interface that software developers use to programmatically interact with software components or resources outside of their own code. An even simpler definition is that an API is the part of a software component that is accessible to other components.

PrivacyRecsys utilizes an API to make the usage of the database and algorithms accessible to the User Interface and also make them available for other systems to use.

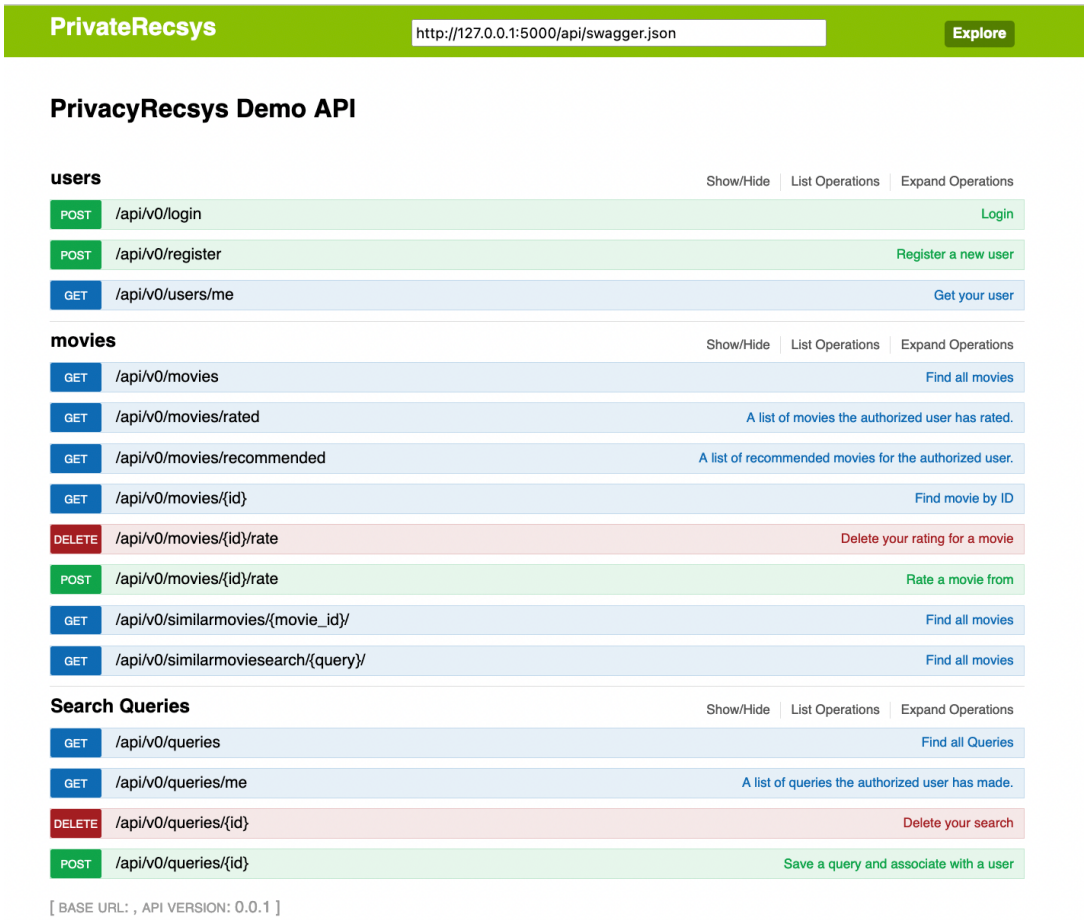
In particular, we used **FLASK** and **Swagger** for developing the API and we made it available as a website where other developers can find the documentation and also test it. **Flask** is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries.

Flask is a widely used micro web framework for creating APIs in Python. It is a simple yet powerful web framework which is designed to get started quick and easy, with the ability to scale up to complex applications.

Swagger is a suite of tools for A**PI developers** from SmartBear Software and a former specification upon which the OpenAPI Specification is based. Using Swagger you can **build, document, test and consume RESTful web services**. It can be used with both a top-down and bottom-up API development approach. In the top-down, or design-first, method, Swagger can be used to design an API before any code is written.

The PrivateRecsys API supports functionality related to users, movies and search queries.

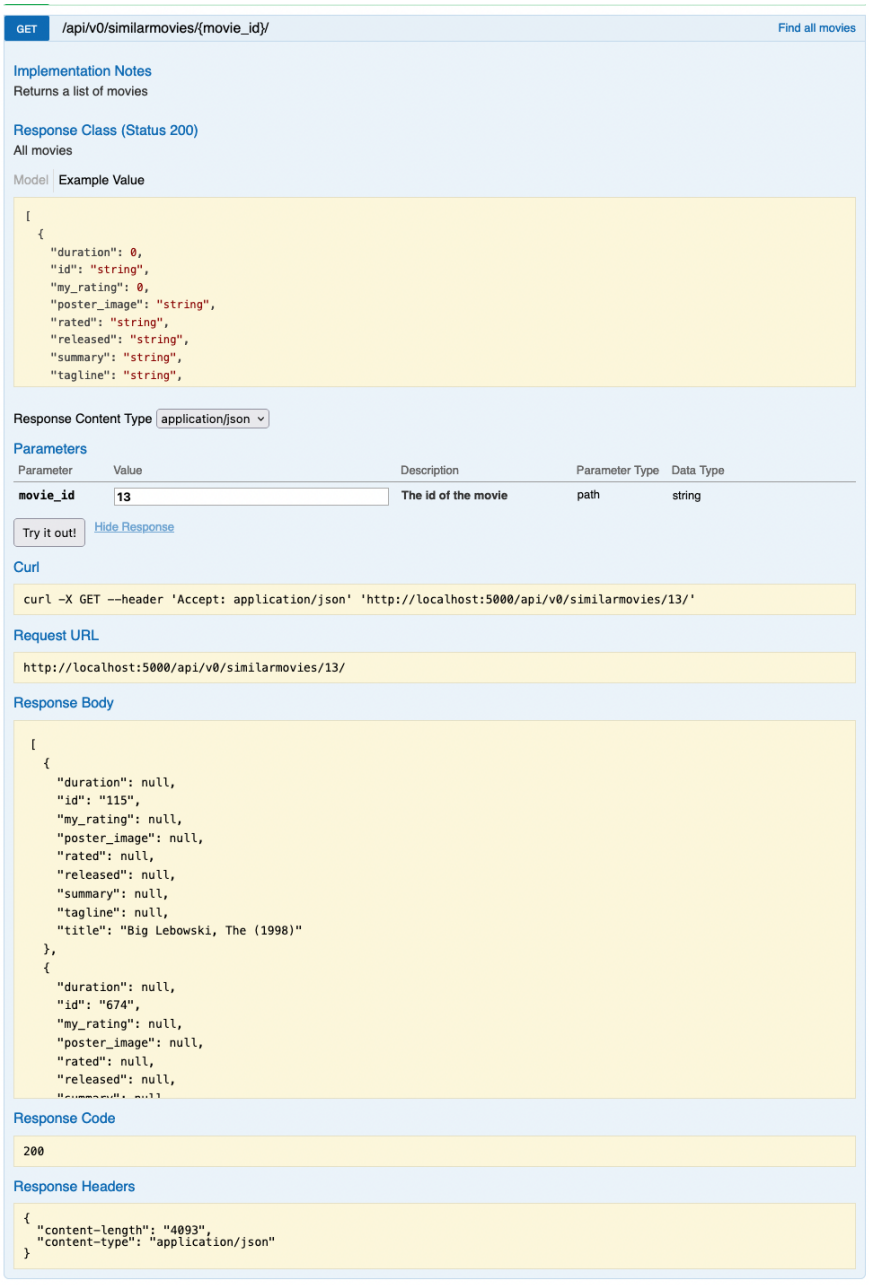
Allows a user to login, register and get the user profile, for movies the user get all movies, only those rated by the user, recommended and similar movies, rate and delete a rating for a movie, and also receive similar movies based on a text query.



The PrivateRecsys API currently support the required functionality of both the PrivacyRecsys system as a stand alone, but also as an integration into Searx.

The API can be further expanded to support more items like news articles, items to buy, places, restaurants etc.

Example Query on the PrivacyRecsys Demo API

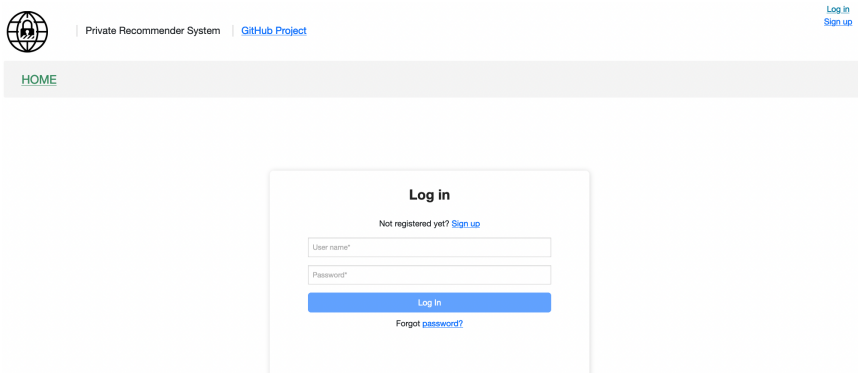


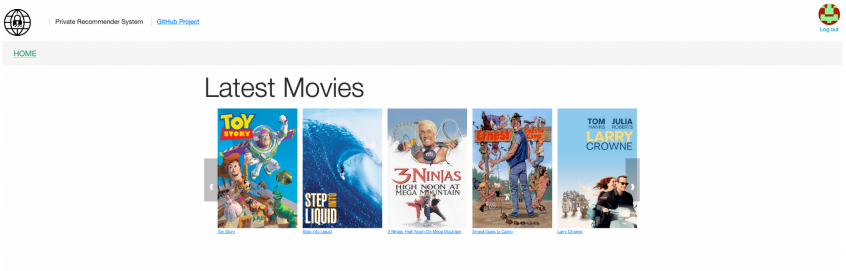
Front End - PrivacyRecsys System

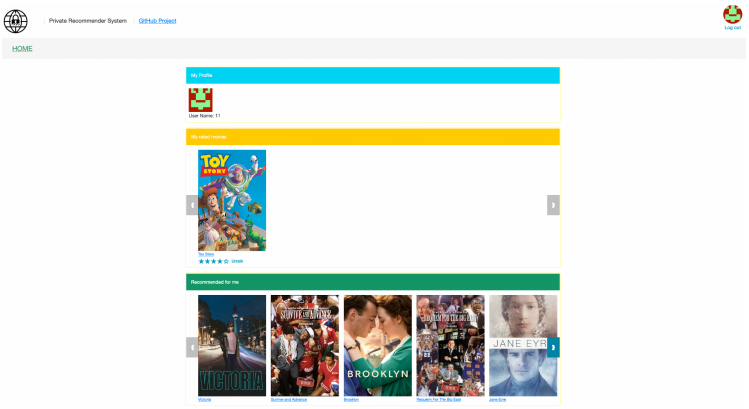
For the front-end of this web-based system React Js was used. **React** is a free and open-source front-end JavaScript library for building user interfaces based on UI components. It is maintained by Meta and a community of individual developers and companies. **Bootstrap css** was also used for styling and formulating the website.

The implemented system allows a user to register and login, rate movies and receive recommendations based on previous ratings . The system considers both item-based and collaborative-filtering recommendations to suggest items to the user.

Screenshots from the Recsys System are available below.



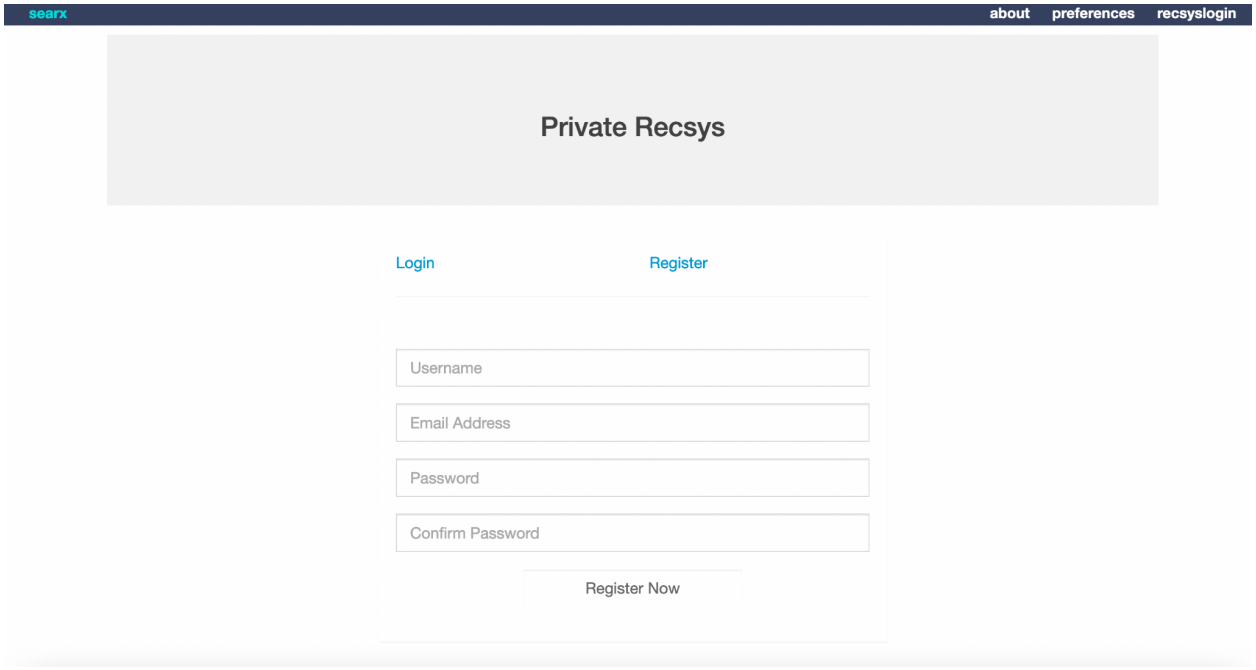




Front End - Integrated in Searx

For the front-end of integration of the PrivacyRecsys into Searx HTML, Python, and JSS was used.

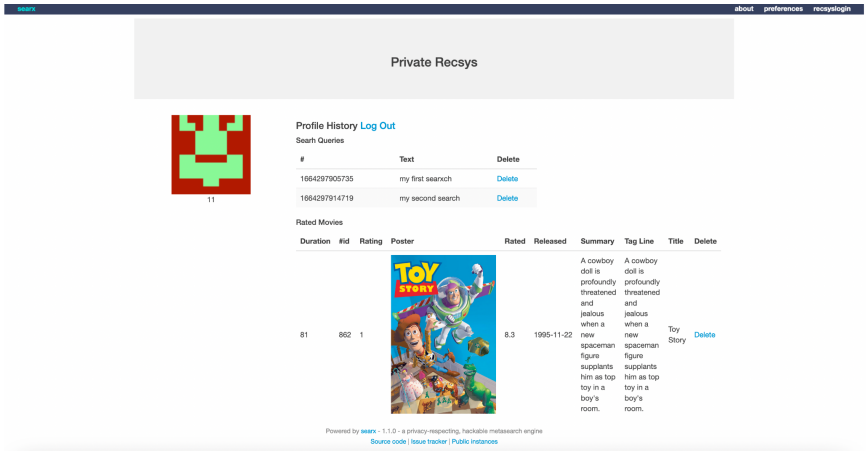
As a first step, a new menu item was added on the top bar to allow the privacy recsys login.

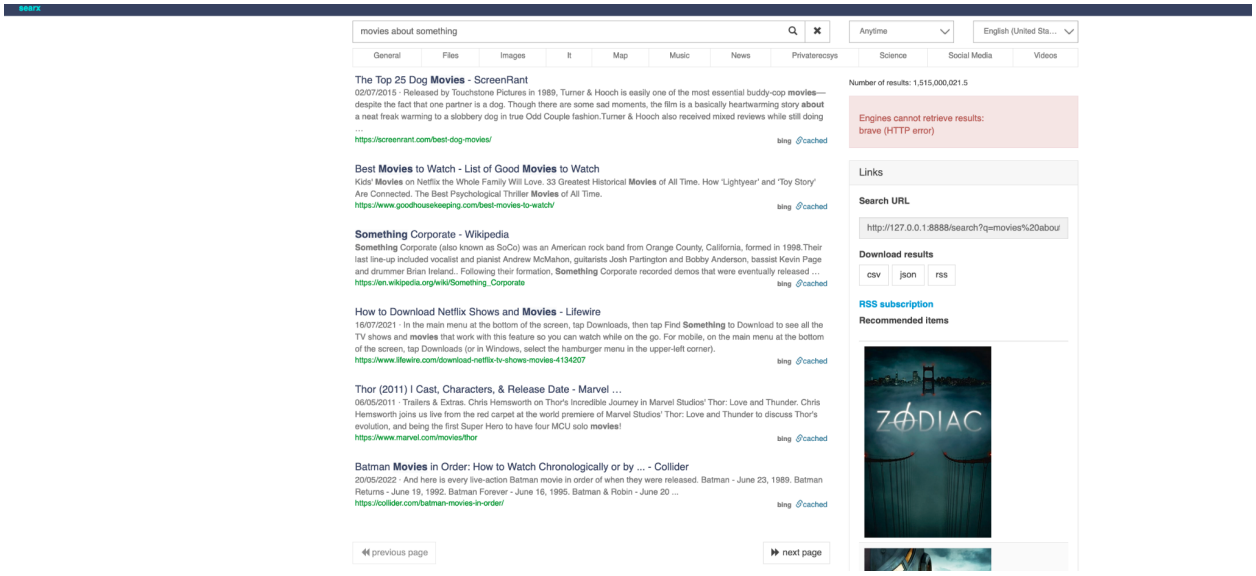


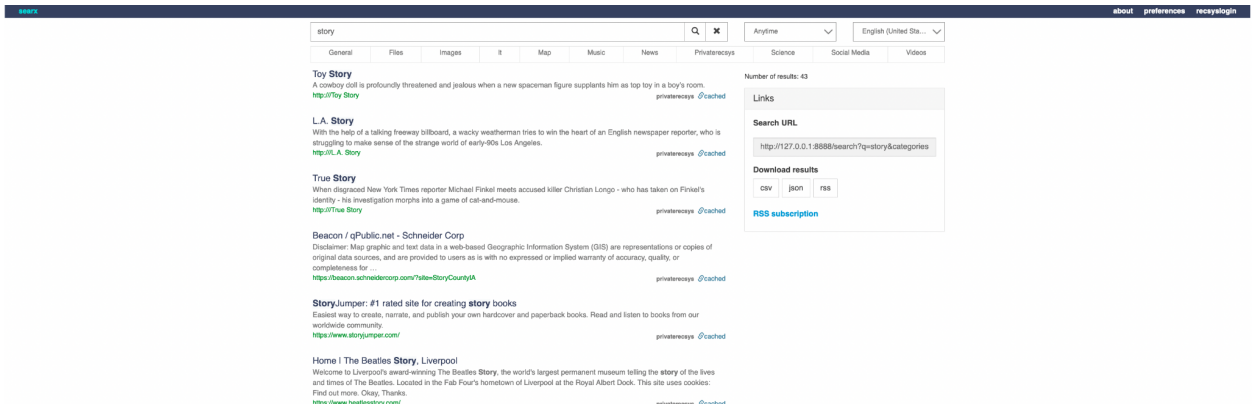
After login in the user can see its profile, which consists of all the searches a user made as well as all the movies he has rated in the past. The user is allowed to remove any of the items in order not to be further considered in recommendations.

A logged-in user will receive movie recommendations on the left hand side, but also when he presses the **PrivateRecsys** category that was installed on the search engine. By pressing in the category the user will receive recommendations.

If the user is not logged in, or does not have a profile, the user will receive recommendations based on the current query only.


searx
about
preferences
recsyslogin
Private Recsys
11
Profile History Log Out
Searh Queries
Text Delete
1664297905735 my first searxch Delete
1664297914719 my second search Delete
Rated Movies
Duration #id Rating Poster Rated Released Summary Tag Line Title Delete
81 862 1 8.3 1995-11-22 A cowboy doll is profoundly threatened and jealous when a new spaceman figure supplants him as top toy in a boy's room. A cowboy doll is profoundly threatened and jealous when a new spaceman figure supplants him as top toy in a boy's room. Toy Story Delete
Powered by searx - 1.1.0 - a privacy-respecting, hackable metasearch engine
Source code | Issue tracker | Public instances


searx
movies about something
Anytime
English (United Sta...
General Files Images It Map Music News Privatecsys Science Social Media Videos
The Top 25 Dog Movies - ScreenRant
02/07/2015 · Released by Touchstone Pictures in 1989, Turner & Hooch is easily one of the most essential buddy-cop movies—despite the fact that one partner is a dog. Though there are some sad moments, the film is a basically heartwarming story about a neat freak warming to a slobbery dog in true Odd Couple fashion.Turner & Hooch also received mixed reviews while still doing ...
https://screenrant.com/best-dog-movies/
bing cached
Best Movies to Watch - List of Good Movies to Watch
Kids' Movies on Netflix the Whole Family Will Love. 33 Greatest Historical Movies of All Time. How 'Lightyear' and 'Toy Story' Are Connected. The Best Psychological Thriller Movies of All Time.
https://www.goodhousekeeping.com/best-movies-to-watch/
bing cached
Something Corporate - Wikipedia
Something Corporate (also known as SoCo) was an American rock band from Orange County, California, formed in 1998.Their last line-up included vocalist and pianist Andrew McMahon, guitarists Josh Partington and Bobby Anderson, bassist Kevin Page and drummer Brian Ireland.. Following their formation, Something Corporate recorded demos that were eventually released ...
https://en.wikipedia.org/wiki/Something_Corporate
bing cached
How to Download Netflix Shows and Movies - Lifewire
16/07/2021 · In the main menu at the bottom of the screen, tap Downloads, then tap Find Something to Download to see all the TV shows and movies that work with this feature so you can watch while on the go. For mobile, on the main menu at the bottom of the screen, tap Downloads (or in Windows, select the hamburger menu in the upper-left corner).
https://www.lifewire.com/download-netflix-tv-shows-movies-4134207
bing cached
Thor (2011) | Cast, Characters, & Release Date - Marvel ...
06/05/2011 · Trailers & Extras. Chris Hemsworth on Thor's Incredible Journey in Marvel Studios' Thor: Love and Thunder. Chris Hemsworth joins us live from the red carpet at the world premiere of Marvel Studios' Thor: Love and Thunder to discuss Thor's evolution, and being the first Super Hero to have four MCU solo movies!
https://www.marvel.com/movies/thor
bing cached
Batman Movies in Order: How to Watch Chronologically or by ... - Collider
20/05/2022 · And here is every live-action Batman movie in order of when they were released. Batman - June 23, 1989. Batman Returns - June 19, 1992. Batman Forever - June 16, 1995. Batman & Robin - June 20 ...
https://collider.com/batman-movies-in-order/
bing cached
previous page
next page
Number of results: 1,515,000,021.5
Engines cannot retrieve results:
brave (HTTP error)
Links
Search URL
http://127.0.0.1:8888/search?q=movies%20about
Download results
csv json rss
RSS subscription
Recommended items
ZODIAC


searx
about
preferences
recsyslogin
story
Anytime
English (United Sta...
General Files Images It Map Music News Privatecsys Science Social Media Videos
Toy Story
A cowboy doll is profoundly threatened and jealous when a new spaceman figure supplants him as top toy in a boy's room.
http://Toy Story
privatecsys cached
L.A. Story
With the help of a talking freeway billboard, a wacky weatherman tries to win the heart of an English newspaper reporter, who is struggling to make sense of the strange world of early-90s Los Angeles.
http://L.A. Story
privatecsys cached
True Story
When disgraced New York Times reporter Michael Finkel meets accused killer Christian Longo - who has taken on Finkel's identity - his investigation morphs into a game of cat-and-mouse.
http://True Story
privatecsys cached
Beacon / qPublic.net - Schneider Corp
Disclaimer: Map graphic and text data in a web-based Geographic Information System (GIS) are representations or copies of original data sources, and are provided to users as is with no expressed or implied warranty of accuracy, quality, or completeness for ...
https://beacon.schneidercorp.com/?site=StoryCountyIA
privatecsys cached
StoryJumper: #1 rated site for creating story books
Easiest way to create, narrate, and publish your own hardcover and paperback books. Read and listen to books from our worldwide community.
https://www.storyjumper.com/
privatecsys cached
Home | The Beatles Story, Liverpool
Welcome to Liverpool's award-winning The Beatles Story, the world's largest permanent museum telling the story of the lives and times of The Beatles. Located in the Fab Four's hometown of Liverpool at the Royal Albert Dock. This site uses cookies: Find out more. Okay, Thanks.
Number of results: 43
Links
Search URL
http://127.0.0.1:8888/search?q=story&categories
Download results
csv json rss
RSS subscription

Recsys engine - Searx

The recsys engine was implemented into Seacx using Python and serves as a middle-ware between Searx and the PrivacyRecsys components. Through the engine, the user profile is identified, the queries are saved into the user profile and all the interaction with the database and the privacy recsys algorithms is implemented.

Conclusion

This deliverable presented the architecture, and the various components of the PrivateRecsys system as a standalone and as an integration into Searx. This documentation shall assist further expansions of the system and improvements, as well as in the developments of other projects and integrations for the Searx engine.

The PrivacyRecsys system is far from being perfect and can be expanded in various ways. We hereby list some of the improvements that can be made to improve its usability and effectiveness.

1) Encrypted database and privacy-preserving API: Although the algorithms for retrieving data from the database and making recommendations of users are privacy preserving, the database still holds all the data can be violated and hacked to exploit user data. A graph-based, yet encrypted database supported by a similar privacy preserving API can better ensure data privacy and security.
2) User Interface: Given the short duration of this project, not enough effort was paid in making a truly usable and aesthetically pleasing front-end. A better in terms of design, user interaction user interface would allow users to take full advantage of what this system was to offer.
3) PrivacyRecsys as a plugin for a browser or independent browser: The integration into searx was a proof-of-concept of the integrability of the components. However, this is restricted in only utilizing search queries and depending on existing search engines (google, bing etc) to provide results. A plugin for a browser, or an independent browser would be better suited for truly privacy preserving recommendations in various domains.