

# **Report**

# **ASSIGNMENT-3**

USAMA KHALID

I19-1236

Saood Khan

19I-2128

MS(DS)-B

# **Background of the problem**

Having all the dblp data organized in a relational database the current task is to identify the authors having same Focus of Research (FoR) and have published the paper together (co authorship). The task is to display the co authorship between them via graph. The nodes in the graph represent authors while the edges represent number of publications between them.

# **Issues faced and their solutions**

1. Query for finding coauthors was computationally very expensive: The query we initially devised for combining every author with every other author had to be run n^2 times. That gave us a time estimate of around 1.5 years!!. We then optimized it so that it would only had to be run n times which had a running time of 9 days!!.

Solution: We tried different laptops and then decided to upload the mysql server to google cloud for faster query processing. We also optimized the query further and we were able to find coauthors of a single FoR in a few hours.

1. There was no support for graphs in Android: We made the gui of Assignment 1 and 2 on Android but it doesn’t support graphs very well.

Solution: We shifted our whole Assignment 1, 2 and 3 on a web based GUI and used visjs to draw graphs.

1. Computing coauthors and their FoRs on the fly was not possible: Computing an FoR for an author itself is a very resource intensive process let alone computing coauthors.

Solution: We pre computed and stored information about FoR and the number of papers each author has published with every other author.

1. Displaying graphs with large number of nodes: The browser would hang if we would try to display all authors for a particular FoR in a graph.

Solution: This happened because there were thousands of authors for a particular FoR with 4 lakh being the highest. We set a limit on the number of nodes that will be displayed at a time.

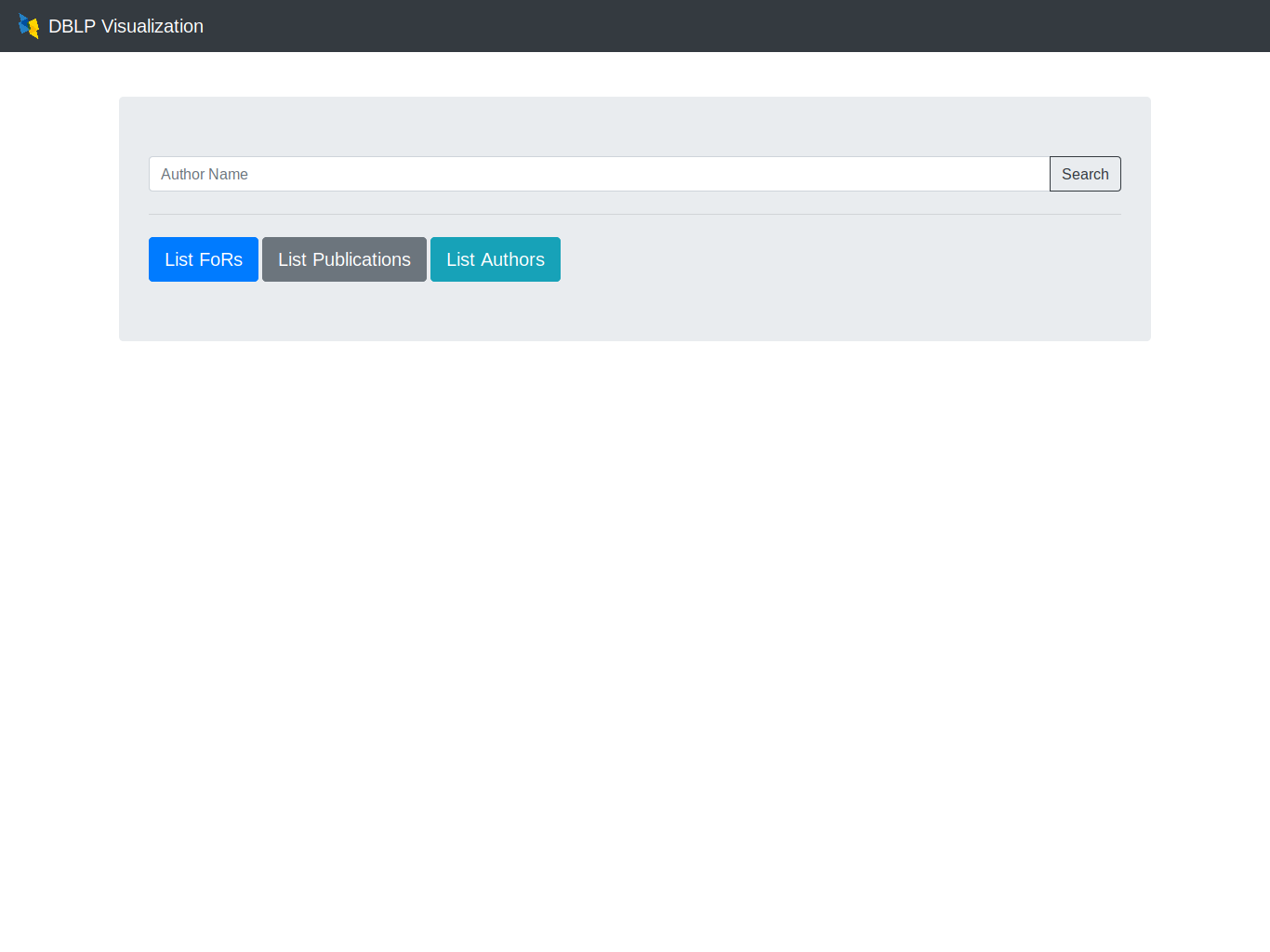
1. Creating graph data at run time was very slow: Calculating the edges and nodes to display and their weights is computationally very expensive.

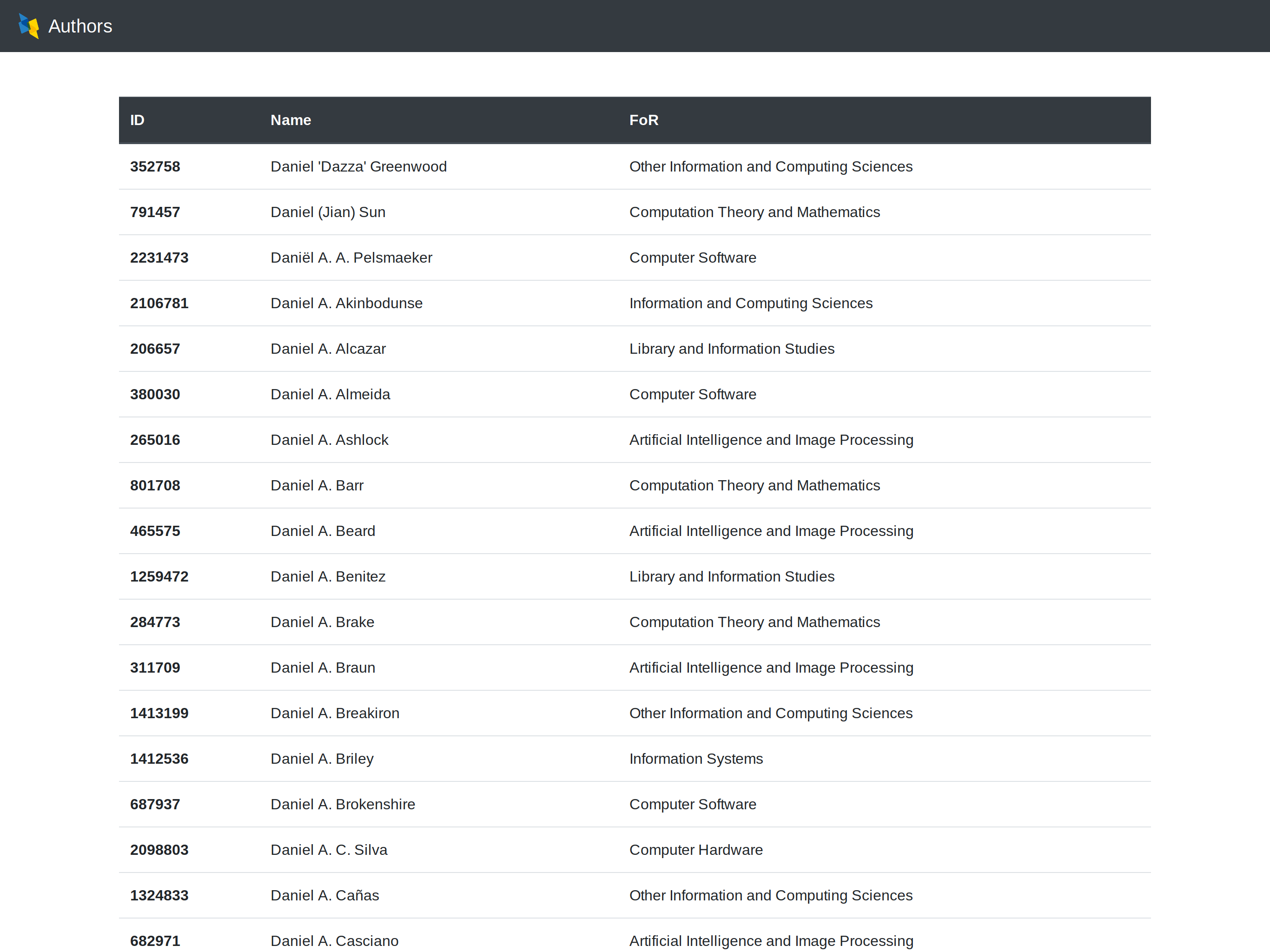
Solution: We made an edge list for all graphs and stored in our database so that graph plotting is fast.

**The pseudocode of the programming logic designed to identify authors having same FoR and have published together**

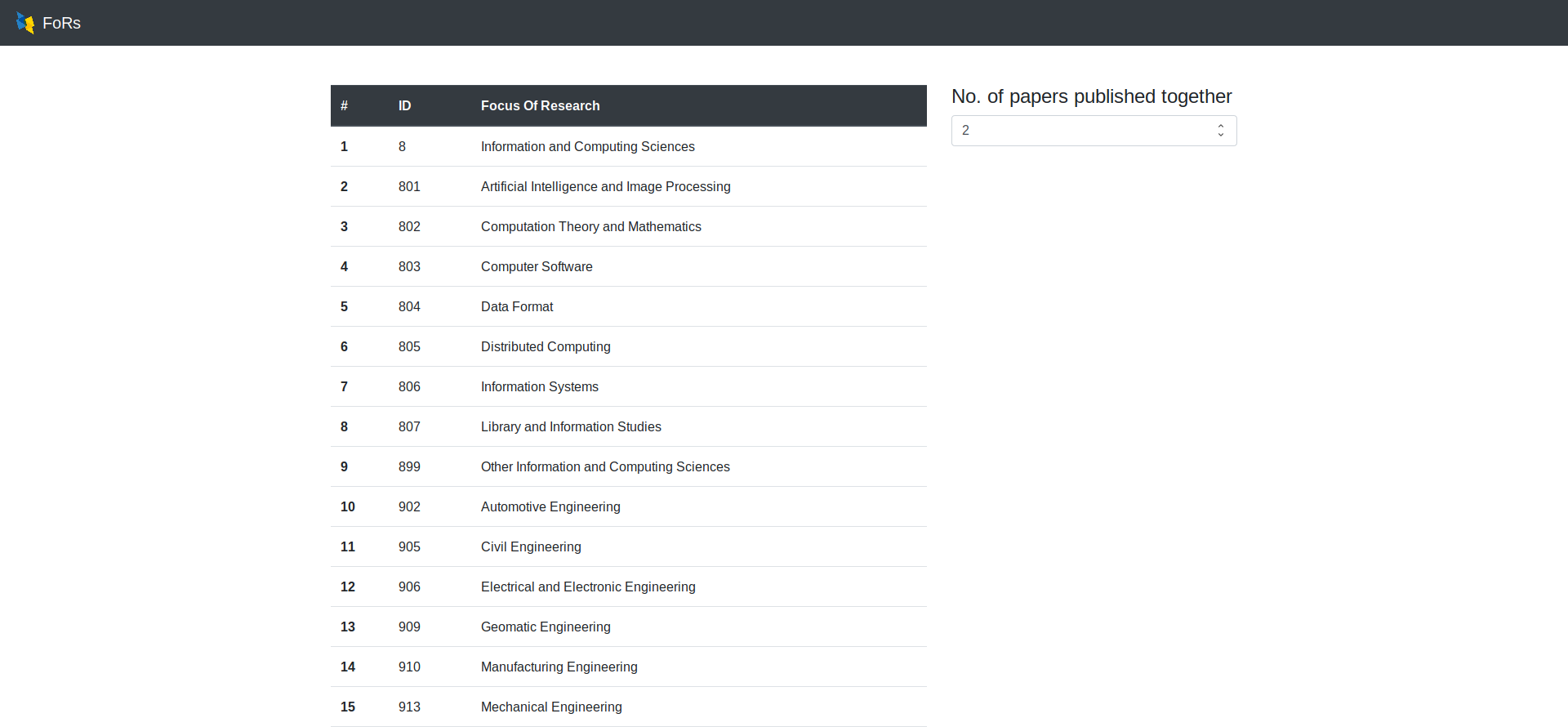
1. **Given an FoR find all authors that belong to that FoR.**
2. **Combine each author with every other author where their publications match.**
3. **Count the no. of publications they have done together.**
4. **Store this information as an edge list in database.**

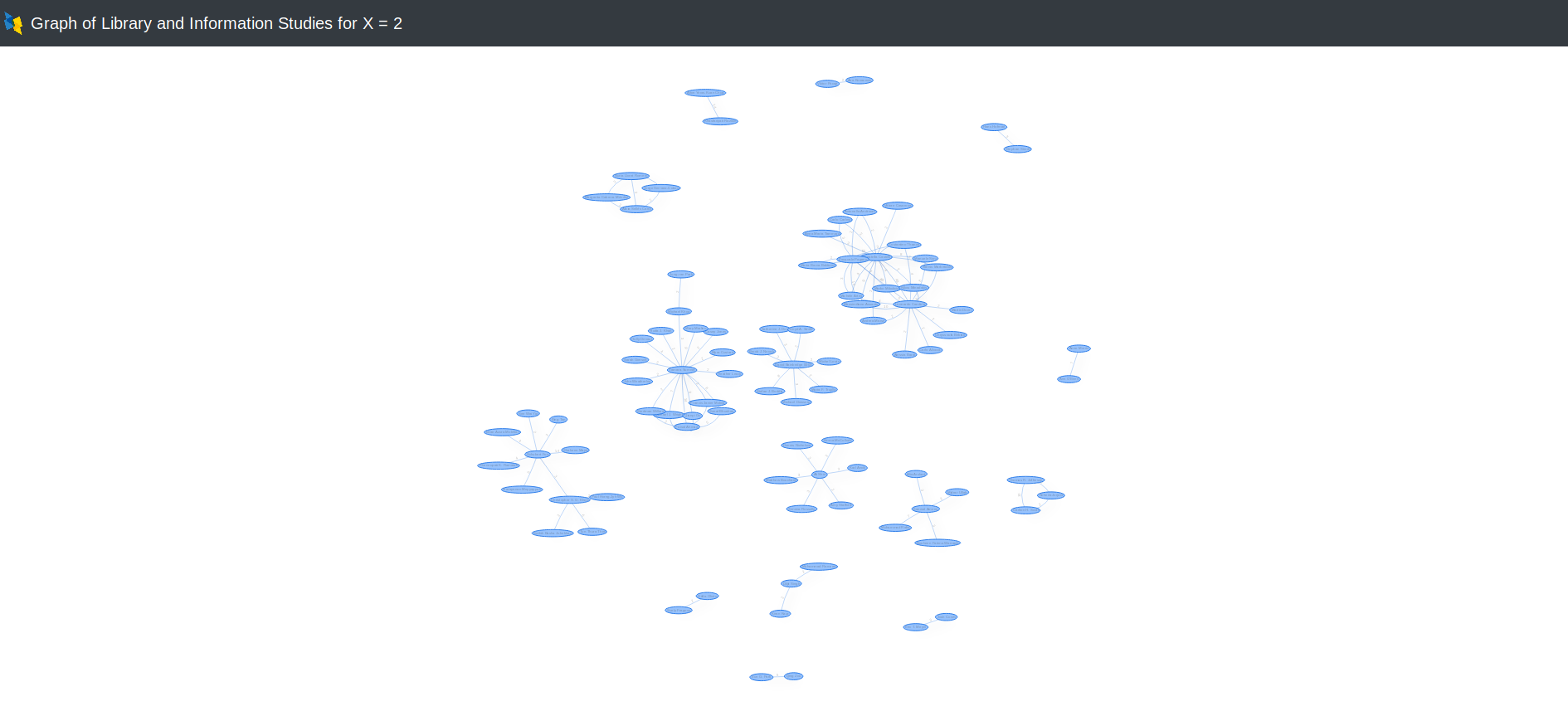
**Screenshots of the application**

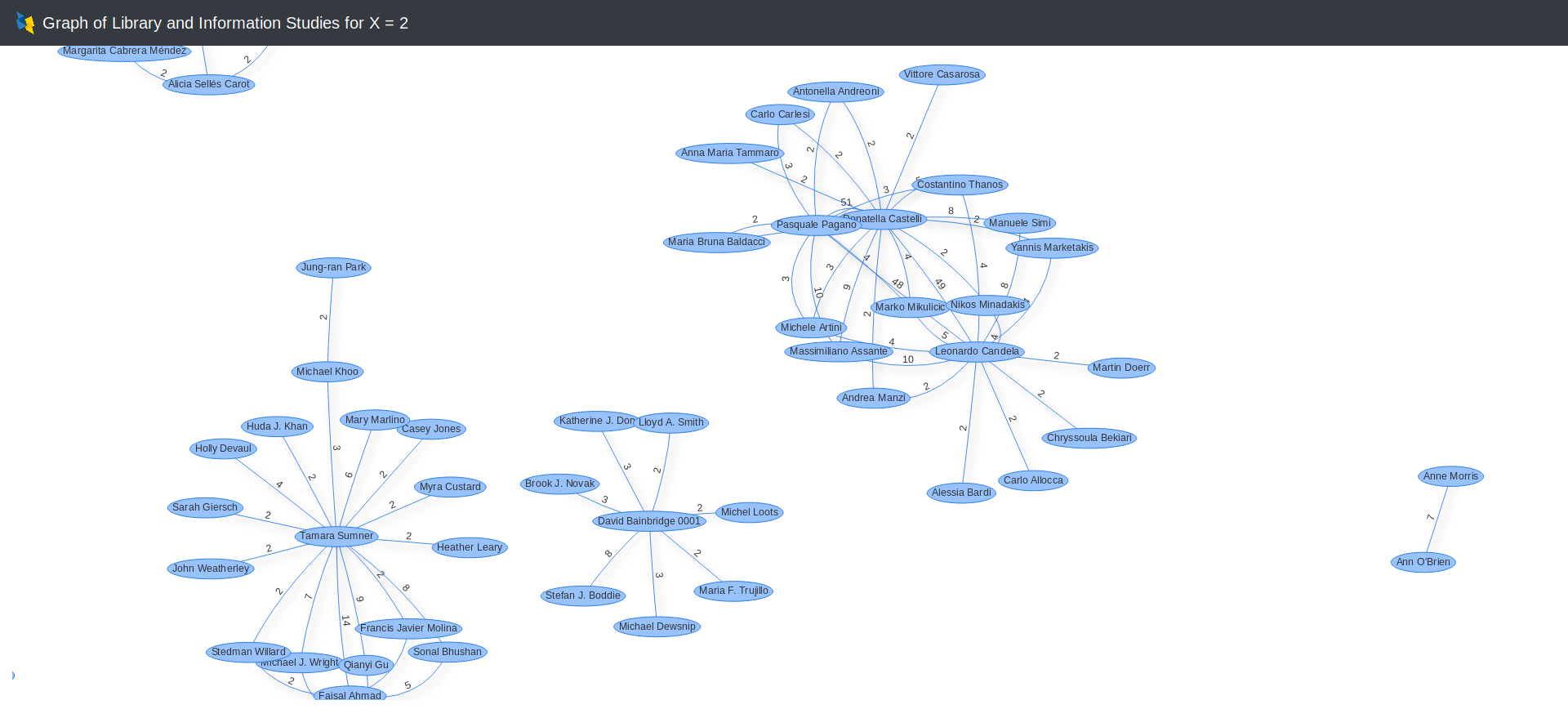
**Figure 1: Main Screen**



**Figure 2: Search Query Result for author name Daniel**

**Figure 3: List of all FoRs**

**Figure 4: Graph of an FoR**

**Figure 5: Graph of an FoR zoomed**