**Software Requirements Specification**

**for**

**FindIt : Research Paper Recommendation System**

**Version 1.0**

**Prepared by Radhika Patwari**

**< 18CS10062 >**

**rsrkpatwari1234@gmail.com**

**Department of Computer Science and Engineering,IIT Kharagpur**

**< 24/2/2020 >**

**Table of Contents**

**Table of Contents**

**Revision History**

**1. Introduction 1**

1.1 Purpose and Need

1.2 Addressing the Need

1.3 Prospective Users or Intended Audience

1.4 Issues and Challenges to overcome

1.5 References

1.6 Future Work

**2. Work Plan for the Project 3**

2.1 Purpose and Need

**3. Functional Requirements 3**

3.1 Application Startup

3.2 User Registration

3.3 User Login

3.4 Naming a Session

3.5 Text Input Function using Dropdown menu

3.6 Extract Function

3.7 Search using Search Engine

3.8 The Sentence Searching Algorithm

3.9 Paper-score sorting Function

3.10 Matched Papers Display Function

3.11 Url Display Function

3.12 Preview Paper Function

3.13 New Search Function

3.14 Session History Display Function

3.15 Logout Function

**4. Nonfunctional Requirements 6**

4.1 Performance Requirements

4.2 Security Requirements

4.3 Design Constraints

4.4 Software System Quality Attributes

**5. External Interface/Environment Requirements 8**

5.1 Hardware Requirements

5.2 Software Requirements

5.3 User Interface Requirements

**6. Attractive System Features 9**

6.1 We maintain the privacy of User data

6.2 We provide a FREE Trial with NO feature restriction!

6.3 Easy to Use

6.4 Personalised Sessions with User Accounts

6.5 Save Your Session History

6.6 You may contact us for including the paper in database

**7. Business Policy 10**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# **Introduction**

## **Purpose and Need**

Research Paper is defined as the ‘academic writing’ that contains the results and evaluation of original research conducted by researchers in various fields. It demonstrates the current activities ongoing in that area and is mostly referred to by all the researchers. However, the explosive growth of available digital information and the number of visitors to the Internet have created a potential challenge of information overload that hinders timely access to these items of interest on the Internet. It is necessary to enable the users to search through only those records of knowledge which are related to their interest and preference.

This demand is meet using a ‘recommendation system’ that applies a decision making strategy to filter the relevant data from complex information environments.Recommender systems handle the problem of information overload that users normally encounter by providing them with personalized, exclusive content and service recommendations.It serves as a means of assisting and augmenting the social process of using recommendations of others to make choices when there is no sufficient personal knowledge or experience of the alternatives.

## Addressing the Need

To enable users to search through relevant set of research papers, a ‘research paper recommendation system’ is used. This system filters relevant papers from the database of all sorts of papers using content based and collaborative filtering approaches of recommendation. Evaluation is done on the basis of correlation between the user’s profile section and the extracted features of research papers.Scoring is done on the basis of TFIDF strategy. We filter out items that a user might like on the basis of reactions by similar users. Thus the user-item relation from past references helps in recommending items in the current scenario.

## **Prospective Users or Intended Audience**

This application software mainly targets Academia for its prospective users. Research papers are being widely read for gathering knowledge of ongoing work. The PhD scholars, the professional and the undergraduate researchers all over the world can be the common users of this system.They need to search for newly published papers on a regular basis.Considering the time constraint, they need a recommendation system that systematically ranks the papers of need.

## **Issues and Challenges to overcome**

The issues and challenges that are going to arise during the development process of this software can be identified if we properly analyze the very basic principle which the software uses to rank the papers.The system is based on users’ ratings which can be biased.Not all users mention the data correctly in the profile section.The dataset on which the system relies is huge and hence difficult to maintain.Some major problems are:

* **Scalability Problem:** Real world data sets produced by item-users interactions such as preferences, ratings and reviews are huge and dynamic.They are difficult to deal with.
* **Privacy Issue:** To produce quality personalized recommendations, RSs are bound to gather as much user data as possible and to exploit it to the fullest. This may create a negative impression on the users’ mind about their privacy because the system knows too much about them.
* **Robustness:** To gain certain profits, an attacker may generate some fake user profiles based on some attack models, such as Push/Nuke Attacks to make some target items more/less popular respectively.
* **Sparsity of Rating Matrix**: A majority of the users do not rate most of the items and consequently the ratings matrix becomes very sparse. Due to this, the data sparsity problem arises that declines the chances of finding a set of users with similar ratings.
* **Proactive recommendations:** In today’s modern life, where users are always connected to world of computing and internet, it is desirable to have such systems that can predict what, when and how to “push” recommendations on implicit requests.However, the above RS follows a “pull model” where the recommendation requests are initiated by the users.
* **Cold-start problem:** Cold-start problem presents a collective issue of new item and new user to RSs. A new item can’t be recommended initially when it is introduced to a CF system with no ratings.

## **References**

* B. Maleszka, "A Generic Framework for Collaborative Recommendation in a Scientific Network," 2019 IEEE International Conference on Systems, Man and Cybernetics (SMC), Bari, Italy, 2019, pp. 95-100.
* Balraj Kumar & Neeraj Sharma.(2016). Approaches,Issues,Challenges in Recommender Systems.Indian Journal Of Science and Technology.9(47), 10.17485/ijst/2016/v9i47/94892.
* Winoto, P., Tang, T. Y., & McCalla, G. I. (2012). Contexts in a paper recommendation system with collaborative filtering. *The International Review of Research in Open and Distributed Learning*, *13*(5), 56-75. https://doi.org/10.19173/irrodl.v13i5.1243.

## Future Work

Currently many constraints are not being included while searching a topic.These will be included in future.The paper preview option is currently not available.We can provide only the Url of the research paper if found and then the user can search it on Google.This paper preview option will be included in later version.

# **Work Plan for the Project**

## **Purpose and Need**

|  |  |
| --- | --- |
| **Week** | **Plan** |
| Week-1 | Preparation of SRS, RAS and FS related documents |
| Week-2 | Designing the Structure to write the program for the application software |
| Week-3 | Designing the Graphical User Interface for the application software |
| Week-4 | Designing the Algorithm |
| Week-5 | Designing the complete Backend of the application software |
| Week-6 | Testing of the Application and Error Correction |
| Week-7 | Refining the GUI and making it look better and more user friendly |
| Week-8 | Final Testing |

# **Functional Requirements**

**3.1 Application Startup**

When the user opens the application the logo of the application must be displayed in the screen

for 3 seconds with a welcome message and then proceed to the user login/sign-up window.

**Input :** When the application icon is clicked.

**Output :** Application Logo on the Screen with a Welcome Message and then proceed to

login/sign-up window.

**3.2 User Registration**

An User must be able to register for an account for the application after the user has provided

the required credentials like email-id, name, password, mobile no., and IP Address(determined

automatically).

**Input :** Details and Credentials of the user.

**Output :** A message notifying either successful or unsuccessful sign-up with reasons.

**3.2.1 Email Verification**

If the email address used in the sign-up process already exists with another user, there should be a prompt for the user to notify him/her about that.

**Input :** Email address provided during registration.

**Output :** A message stating ‘email address already exists!’ if there is an account with the same email Id.

**3.2.2 Image Upload**

The user is asked to upload an image for the user profile. This step may be skipped by the user.

**Input :** Success message from Email Verification process.

**Output :** Ask to upload an Image if the user chooses to do so, else continue to the main

window.

**3.3 User Login**

Allows the user to login to the user account whenever the user enters the username(email-id)

and password and allows the user to access the application software.

**Input :** Email address and password by the user.

**Output :** Successful login on email-password matching, else an Error Message.

**3.3.1 Email Not Found**

If email address entered by the user is not found on the server for login then there is a prompt to notify the user that the email address was not found and the user is asked to either sign-up or re-enter the email address.

**Input :** Email address from login window.

**Output :** Message of ‘Email not found!...Try Again’ or redirect to the Registration Window option is shown.

**3.3.2 Email-Password Mismatch**

If the password entered by the user while login is not correct, the user is prompted about email-password mismatch and is asked to enter the password again.

**Input :** Email address and password from login window.

**Output :** A message of ‘Email-password mismatch...Enter password again!’ if the password doesn’t match with the email address entered.

**3.4 Naming a Session**

The user is asked to name the session once it is logged in as it will be saved in the session

history feature.

**Input :** Title of the Session by the user.

**Output :** Stores the title of the search result and the date for future reference with no

immediate message to the user.

**3.5 Text Input Function using Dropdown menu**

Users are allowed to type the category of desired type of research paper in the search bars that are present as dropdown menus.

**3.5.1 Journal data acceptance**

The user can choose the type of journal from the given dropdown menu by hovering the mouse over the <Give journal name> search bar.

**Input :** Hover the mouse over the <Give journal name> bar and select any of the dropdown options.

**Output :** The selected option will be displayed in place of <Give journal name> option.

**3.5.2 Research paper topic acceptance**

The user can choose the topic of required research paper from the given dropdown menu by hovering the mouse over the <Give topic name> search bar.

**Input :** Hover the mouse over the <Give topic name> bar and select any of the dropdown options.

**Output :** The selected option will be displayed in place of <Give topic name> option.

**3.6 Extract Function**

After the user has entered the text to be searched this function should separate/divide the

content into words to make them ready to be searched in the search engine for

matching and saving them in a separate file.

**Input :** Content Entered by the user.

**Output :** words extracted and saved in a separate file.

**3.7 Search using Search Engine**

This function one-by-one takes the extracted words/sentences and searches them using string matching algorithms in the database.It finally loads the search result.

**Input :** Extracted Sentences.

**Output :** a list of research papers sorted as per matching scores

**3.8 The Sentence Searching Algorithm**

This function searches the extracted text from the user content in the query list of the database. If any match is found then the list of papers,it is mapped to, is saved and noted in a separate file along with their scores, else it just repeats the process for the rest of the words.

**Input :** Extracted text from the user

**Output :** Saves the research paper if a match is found in a separate file, else continues with the rest of the sentences.

**3.9 Paper-score sorting Function**

It goes through the file created by the sentence searching algorithm, and again sorts it according to the individual scores of the papers. This list is again stored in a separate file.

**Input :** The file created by the sentence search algorithm.

**Output :** The file containing the new sorted list.

**3.10 Matched Papers Display Function**

Users are shown the list of all possible research papers in the sorted order of their scores.

Input : The file containing the new sorted list.

Output : A list of all the research papers present in that file.

**3.11 Url Display Function**

The user can download the preview of the required research paper in the pdf format by typing its unique id in the id bar and clicking on ‘SendUrl’ option.The bar is connected to the database and extracts the hyperlink of the paper.This link is then displayed on the ‘URL’ bar.In case the url of the required paper does not exist in the database then a error message is displayed.

**Input :** Type the unique id and click on the ‘SendUrl’ option.

**Output :** The url link of the paper is displayed on the ‘URL’ bar.If the Url is not present in the database then a message ‘Sorry..the url is not present in the database’.

**3.12 Preview Paper Function**

The user can visit the site of the journal displayed on any search engine and can download the preview of the required research paper in the pdf format.

**Input :** Copy Paste the Url(displayed in the Url bar) on the Search bar of a search engine.

**Output :** The research paper can be downloaded in the required format.

**3.13 New Search Function**

This option makes the application ready to conduct a search once again.

**Input :** Click on the New Search Button.

**Output :** Deletes all the temporary files created in the memory and takes the user to a Fresh

Input Window with a new Session Name Prompt.

**3.14 Session History Display Function**

Allows the user to see all the session details which were conducted by him/her till date.

**Input :** Click on the See Session History Button.

**Output :** Session History of the user with the Session Names and the Summary of each

session.

**3.15 Logout Function**

The logout button which upon clicking asks the user if he/she wants to logout of the session.

Upon approving yes, the user is logged out of the session with all the user data deleted except the session history and the window for login or sign-up appears.

**Input :** Click on the logout button.

**Output :** Log-out from the software session upon ‘yes’ approval by the user, along with saving the session history and deleting the rest of the data.

# **Nonfunctional Requirements**

## **Performance Requirements**

**4.1.1 Response Time**

The software should return the desired output after analysis in a reasonable amount of time.

**4.1.2 System Dependability**

It determines the fault tolerance of the system. If the system loses the connection to the Internet or the system gets some strange or invalid input or the system faces any random failure,then the user must be informed about it.

**4.1.3 Prominent Results**

The result displayed must be prominent.The list of papers should be displayed in a sorted order on the basis of their scores. Thus the user should give maximum weightage to the first paper, then the second paper and so on.

## **S**ecurity **Requirements**

**4.2.1 Secure Search**

The software should search the database for the content securely giving utmost

priority to the privacy of the user’s data.

**4.2.2 Temporary Storage of the Searched Query**

The searched query must be stored temporarily and securely and should be permanently deleted after the recommendation session has been over.This is done to maintain the privacy of the data searched by the user.

**4.2.3 User Profile should be Unique**

Any user account registered for the software should be unique and no fake

accounts should be present. Only one user account per IP Address should be

allowed.

**4.2.4 Secure Login**

The system should be secure from malicious or forced login to access the

software.

**4.2.4 Bug tracker facility**

There has to be a bug tracker available where users can report any bugs

they have encountered so that the developers can fix it in the next release

## Design Constraints

**4.3.1 Complexity of Algorithm**

The algorithms used for pre-processing of data have very high complexity.Fuzzy string matching algorithm is used to carrying out searches of topics in the given dataset.Similarly, the TFIDF model has high complexity as it builds a multidimensional metric.Thus the high complexity increases the computational cost of the software and makes its expensive.

## **Software** System **Quality Attributes**

**4.4.1 Reliability**

The system should be reliable i.e. it should give right and accurate results for

each session held by the user.

**4.4.2 Internet Connectivity**

The application must be connected to Internet Connection for performing the

Login activities and Query Search from the database of the software.

**4.4.3 Maintainability**

The application should be easy to extend. The code should be written in such a

way that it favors the implementation of new functions.

**4.4.4 Portability**

The application should be portable to different platforms i.e. it should be

adaptable in different platforms.

# External Interface/Environment **Requirements**

**5.1 Hardware Requirements**

This application software does not have any designated hardware but the hard drive space should be at least 1 GB.This space is required for storing the data sets which mainly contain research papers and sentences to be searched.It should be connected to the internet via high speed cables. It will have a decent processing unit (6 cores at 3 GHz average clock speed) and a good graphical processing unit to enable efficient and fast computation of a few modules. The user side needs to be at all sophisticated.

**5.2 Software Requirements**

Due to interaction with database and user, the project will have some basic dependencies

that will help the Java program interact with the database as well as ease the making of a

Graphical User Interface. Below are the basic requirements of this project:

**5.2.1 Programming languages**

* HTML is the standardmarkup language for documents designed to be displayed in a web browser. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. It describes the structure of a web page semantically and is responsible for the appearance of the document.
* Css is astyle sheet language that describes how Html elements should be displayed on screen. It is a powerful tool for web designers to change the design and control over web pages on how it should be displayed. It is supported by all browsers and is designed primarily to separate the document content from document presentation

**5.2.2 JDK 1.8.0\_232**

Java has been chosen as the programming language for the development of this application software. This application is intended to run on all platforms that support JAVA without the need of recompilation as JAVA is intended to let developers write once, run everywhere.

**5.2.3 Java Libraries**

Some Java Libraries have been used for execution of different and a variety of functional requirements for this application software.

**5.2.4 Python 3.7**

Some Python Libraries have been used for searching query from the database and also for pre-processing the available datasets.We are using Jupyter Notebook of Anaconda3 for running python codes.

**5.2.5 DataBase Management System**

We will need a database system (such as MySQL, SQLite, MongoDB etc.) to keep a track of the users as well as the data entry by each user in the system. In this way, we can keep a permanent storage of the current configuration.

We will be using **Java Database Connectivity Handler** to initiate connection with the

database and perform all the operations such as insertion, modification and deletion. This will also help us in creating different users and keeping their tables separate from each other. We will be creating a single database for a single user and then he can add multiple tables inside it.

**5.3 User Interface Requirements**

There will exist an interactive Graphical User Interface. Text boxes and button events will be

created where ever possible. All errors to be displayed using dialog boxes. All the main menu

and settings options such as account settings, storage management etc will be always available to the user. Most of the button events will have additional selections to be made, this will be done through an interactive dialog box.

# Attractive System Features

This section notes down all the attractive system features we provide for our system software

which would allow the user to give our software the highest preference among all our

competitors.

**6.1 We maintain the privacy of User data**

We respect user data. Our software is designed in such a way that after analyzing the input query from the user, it permanently deletes the extracted text from all possible locations

which were used for performing the search on the database.

**6.2 We provide a FREE Trial with NO feature restriction!**

We provide a FREE Trial of 1 search session to every registered user. This

allows the user to experience the Search Sessions with all the features made available by our Application Software for FREE!

**6.3 Easy to Use**

We have designed the application software interface in such a way that it is convenient to use

by any user. It is very simple and robust at the same time. The simple User Interface allows the user to comfortably use the software without going through any user manual. The function of each option displayed in the interface is also mentioned aside concisely for better user

experience.

**6.4 Personalised Sessions with User Accounts**

We enable access to our application software only to a registered user. The user can register

and create an account for him/her. This allows the user to experience personalized sessions as

per the need of the user and also helps in secure text extraction and search.

**6.5 Save your Session History**

We allow you to save your session history which can be very helpful for the user to refer in the future.

**6.6 You may contact us for including your paper in database**

We allow the users to contact us if they wish to use our platform for the Advertisement of their research papers.Explicit checking will be done to make sure that the paper has been approved by a known journal.

# Business Policy

Almost all the Software Libraries and Resources , to be used for the development of this application software are open-source .All of them are available for free of cost. But it takes time to build such software and obviously, time is precious and in future we may need some funding for the maintenance of the application software. We will be charging a minimal amount of INR 200 monthly as a membership fee for each registered user. The user is allowed to have an account for lifetime and can maintain the history of his previously read papers.The application system can cost around 70k for its maintenance. So as an overall view, this software is Free for users who need such a system on a regular basis.

Our Goal is to provide the user with good and quality software which the user uses for the

betterment of the society.

**System Design Documentation**

**for**

**FindIt : Research Paper Recommendation System**

**Prepared by Radhika Patwari**

**< 18CS10062 >**

**rsrkpatwari1234@gmail.com**

**Department of Computer Science and Engineering,IIT Kharagpur**

**< 26/2/2020 >**

**Table of Contents**

**Table of Contents**

**1. Use Case diagrams 13**

1.1 Registration

1.2 SignIN and SignOUT operations

1.3 Query acceptance and creating session

1.4 Paper searching part

**2. Class diagram 16**

**3. Sequence diagrams 17**

3.1 SignUp , email verification , approve registration & Update user details

3.2 LogIN and LogOUT operations

3.3 Creating Session , Displaying Session & Viewing Session History

3.4 Deleting Account

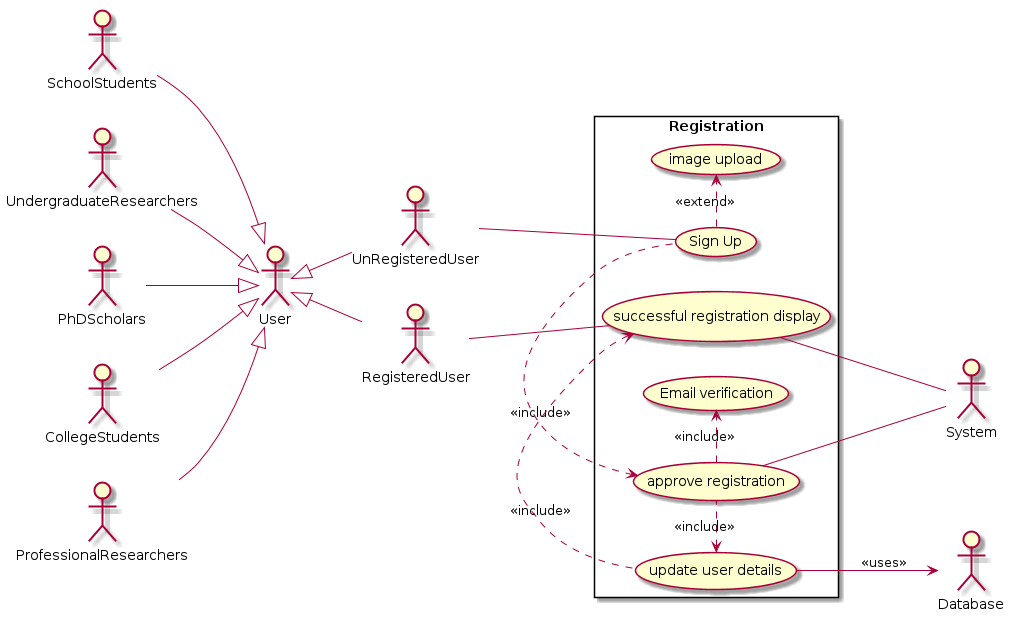
3.5 Text Input , Category of Paper , Enter query , Searching papers , Paper Score Sorting , Matched papers display , Extracting keywords & Sentence Searching

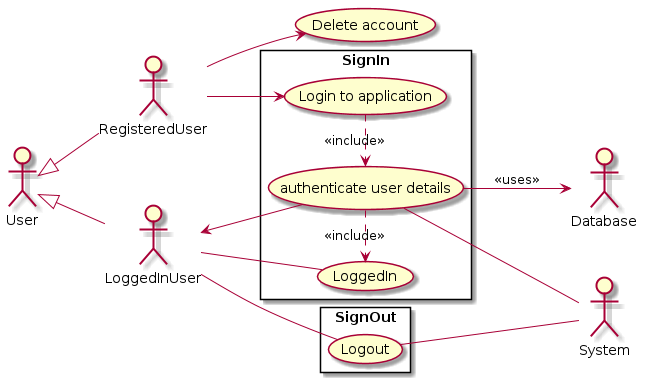
3.6 URL Display , Sending URL & Displaying URL

3.7 Preview paper

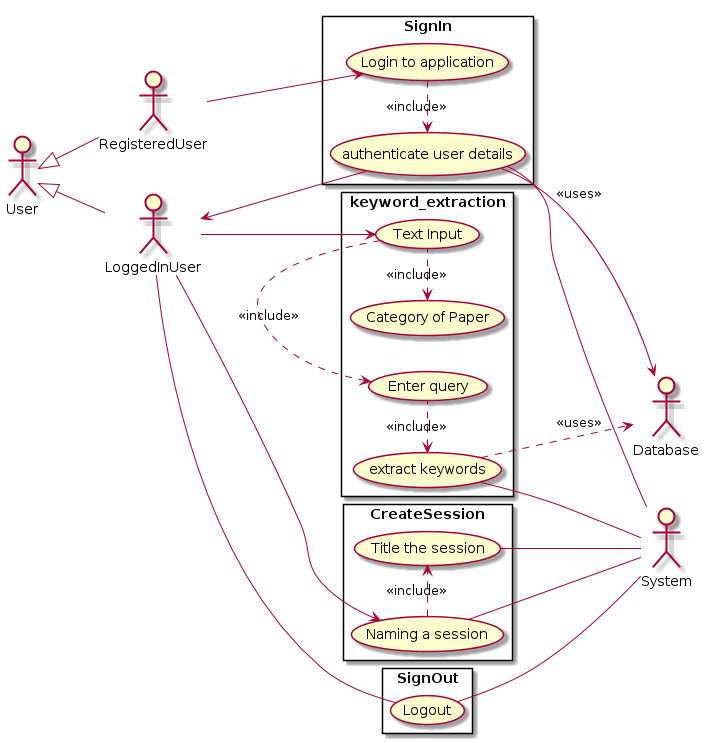
1. **Use Case diagrams**

**1.1 Registration**

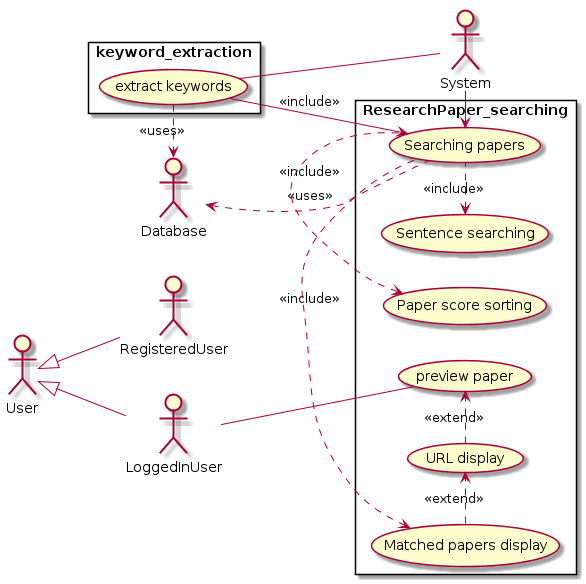


**1.2 SignIN and SignOUT operations**

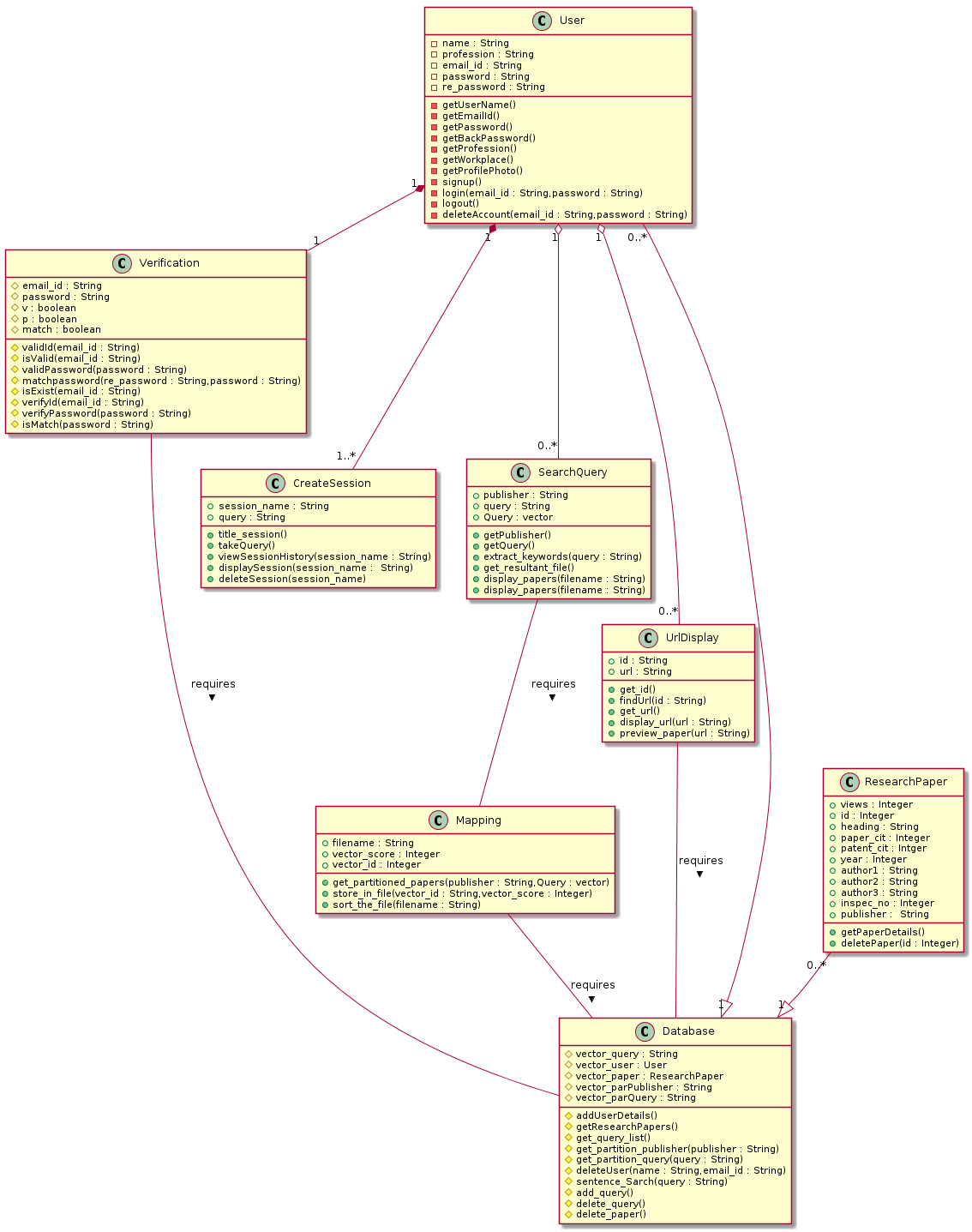
**1.3 Query acceptance and creating session**



**1.4 Paper searching part**



1. **Class diagram**



**Relationship among Classes:**

* **Inheritance**

**-**Database can inherit many User

-Database can inherit many Research papers

* **Aggression**

-OneUser can have many queries and so is related by aggression to class SearchQuery

-One User can apply more many url displays and so is related by aggression to the class UrlDisplay

* **Association**

-UrlDisplay class finds the url of the given paper id.Hence it requires Database information.It is therefore related by association

-SearchQuery requires the partitioned set of papers distributed as per publisher and topic.Hence it requires Mapping class

-Mapping class need the set of research papers and hence requires Database

-Verification class also requires Database to validate the existence of email id and verify the password for signed in user

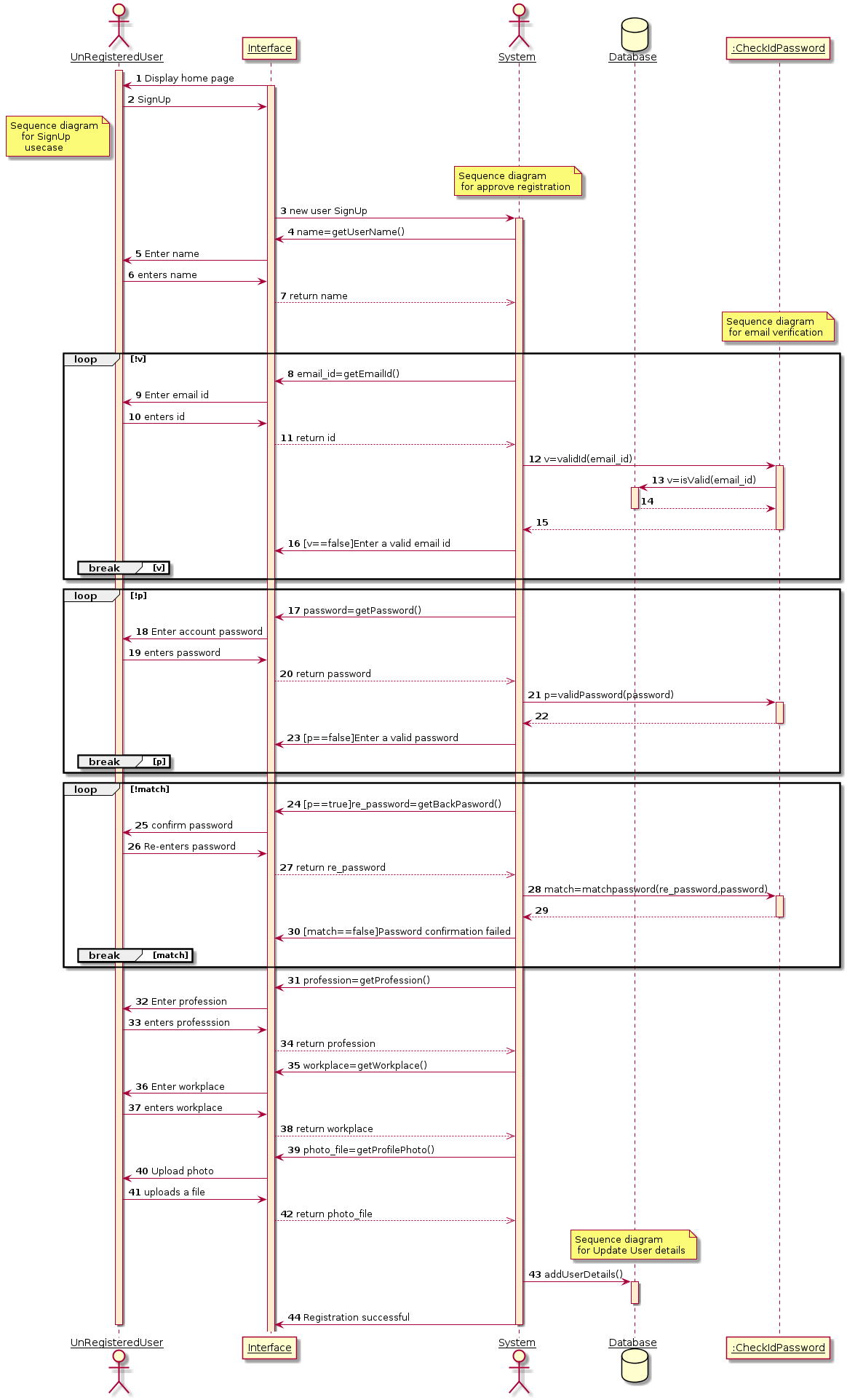
* **Composition**

-It is mandatory for every User to get their email id verified and so Verification and User are related by composition

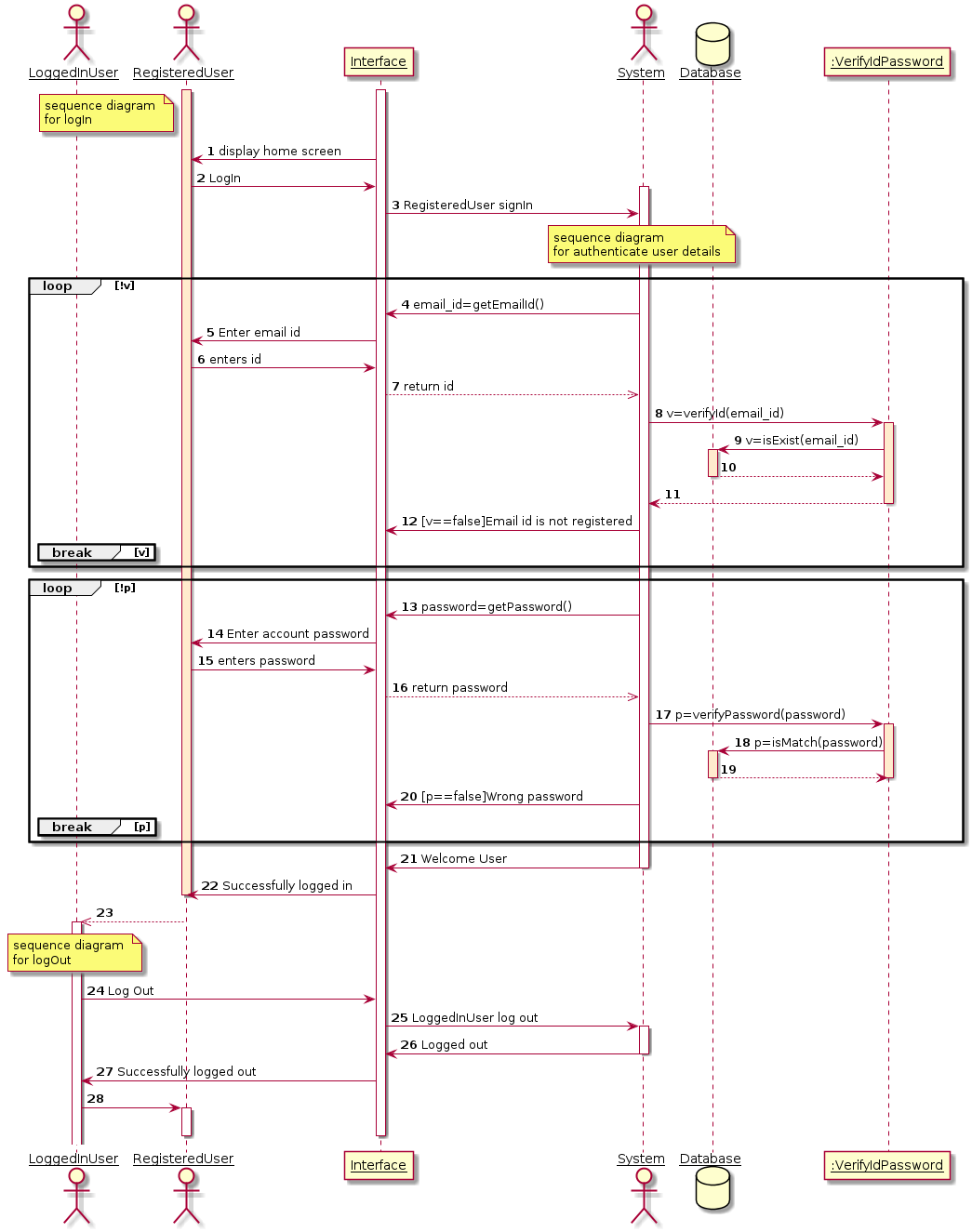
-Every User is required to create a session after logging into the system.So CreateSession and User are related by composition

1. **Sequence diagrams**

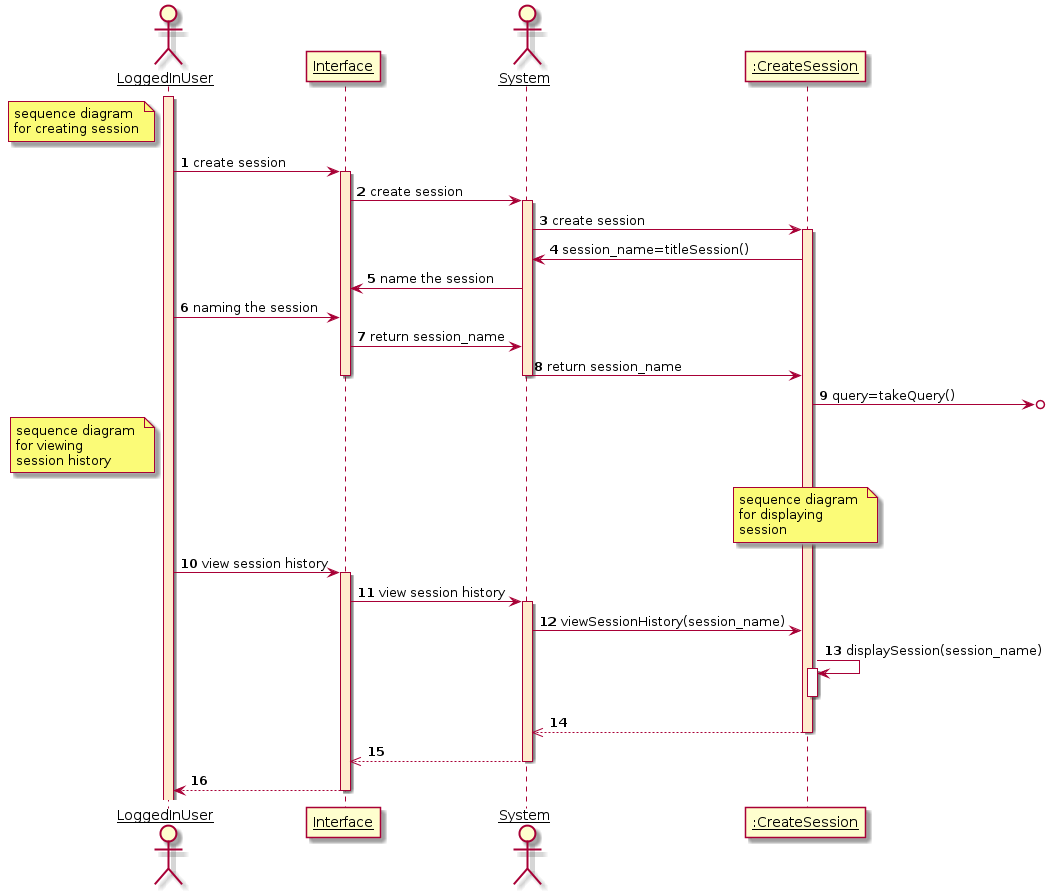
**3.1 SignUp , email verification , approve registration & Update user details**



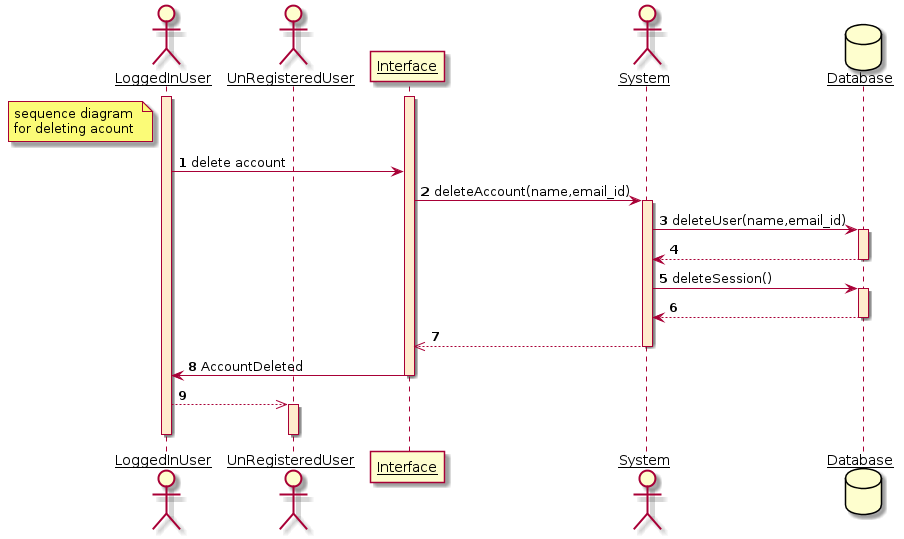
**3.2 LogIN and LogOUT**



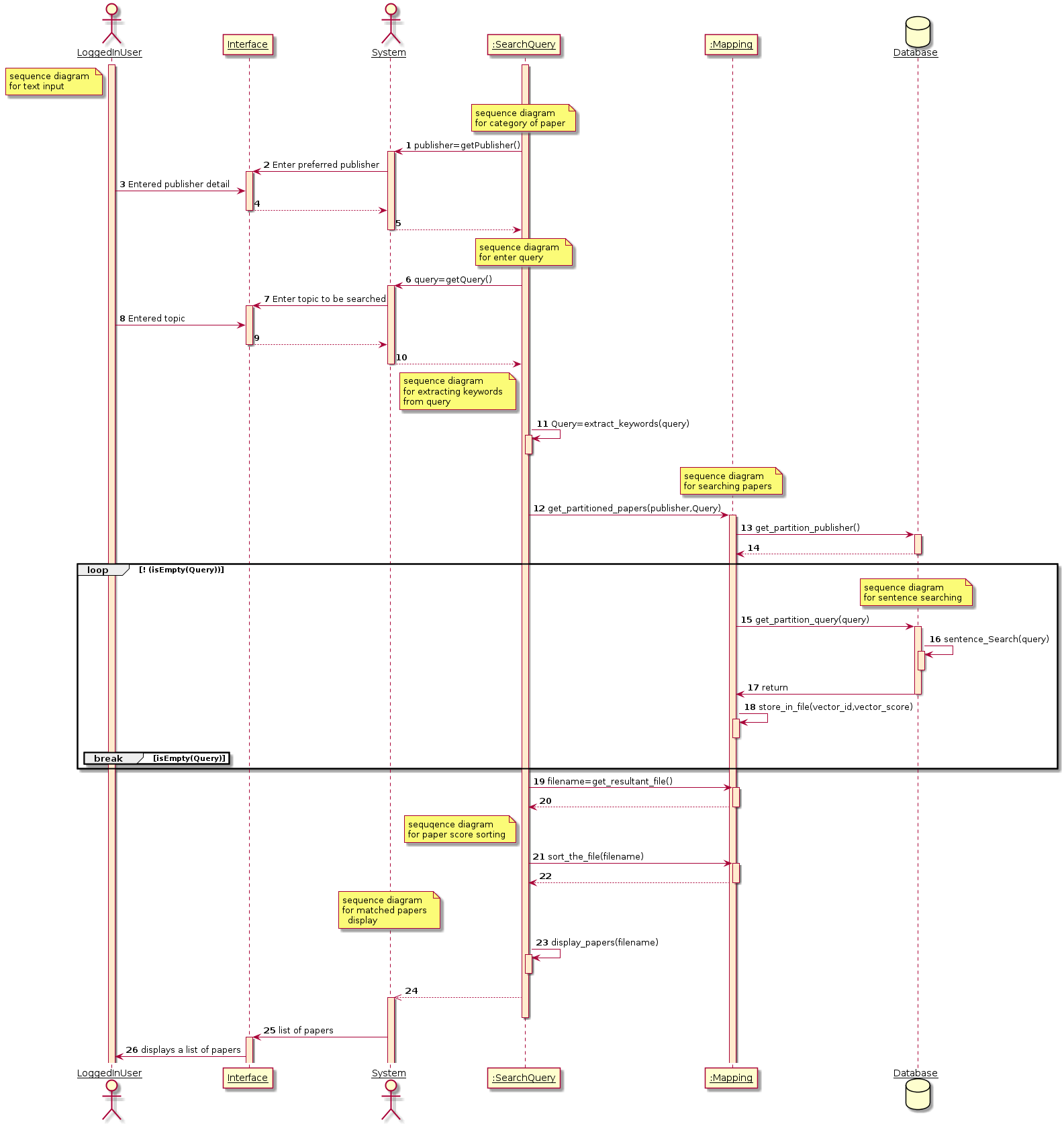
**3.3 Creating Session , Displaying Session & Viewing Session History**



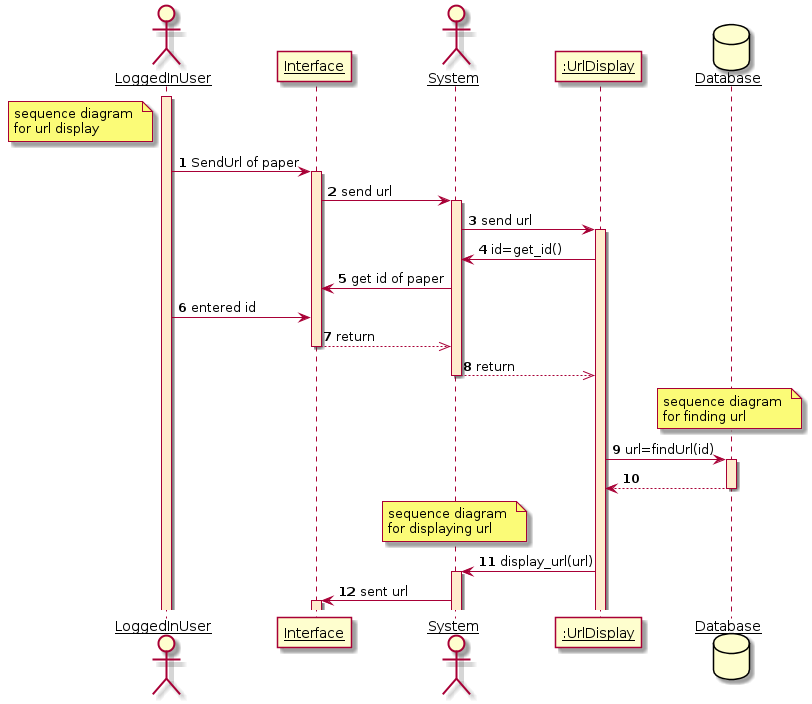
**3.4 Deleting Account**



**3.5 Text Input , Category of Paper , Enter query , Searching papers , Paper Score Sorting , Matched papers display , Extracting keywords & Sentence Searching**



**3.6 URL Display , Sending URL & Displaying URL**



**3.7 Preview Paper**

