Section - 01
Team 6

1) USE-CASE-1:

User logs into the Reviewer Finder App

Primary Actor: Program Committee Chair/ Associate Editor

Goal in Context: Gain authorized access to the Reviewer Finder App

Preconditions:

1. The app should load the Login interface completely
2. The user needs to have a valid username & password
3. The additional libraries and dependencies should be loaded successfully
4. The user should have been registered beforehand for accessing the app.

Trigger: The user enters ID and password and clicks Login on the interface

Scenario:

1. User navigates to the application
2. System displays login page with options to enter username, password and select role as one of Program Committee Chair or Associate Editor
3. The user(chair/editor) provides a valid username, password and role for his/her account.
4. The user clicks the login button to request access to the different operations.
5. The app retrieves all available operations if the login is successful.

Exceptions:

1. The login ID or password is incorrect or not recognized. See use case: “User logs into the Reviewer Finder App”
2. The user logs in directly without entering a valid ID and password. The app displays error message. See use case “User logs into the Reviewer Finder App”.

Priority: High Priority & to be implemented right at the beginning of implementation phase

Available: Implementation Phase

Frequency of use: Frequent

Section - 01
Team 6

2) USE-CASE-2:

Find authors in OOPSLA who published papers since 2010 for building a conference committee

Primary Actor: Program Committee Chair/ Associate Editor

Goal in Context: Find a list of suitable reviewers in a particular conference who published papers since the specified year for building a conference committee

Preconditions:

1. The user should login successfully into the app
2. The additional libraries and dependencies should be loaded successfully
3. The publication DB sources should be loaded successfully
4. User needs to know the journal name and beginning year for searching criteria

Trigger: The user queries the system to find reviewers for building the conference committee.

Scenario:

1. The user(chair/editor) logs in to the system successfully.
2. Selects OOPSLA as the publication for which the conference committee has to be built.
3. The user selects the year to filter author publications since then.
4. The user hits the search button to query the loaded publication data.
5. The app retrieves the list of authors who published more than one paper since the year specified, based on the journal criteria selected by the user.

Exceptions:

1. The user searches directly without selecting at least one of the search options.
2. There are no results displayed as there are no authors with more than one published papers since the selected year 2010 in OOPSLA.

Priority: High Priority & to be implemented with other search options

Available: Deployment phase

Frequency of use: Moderate

Section - 01
Team 6

3) USE-CASE-3:

Find authors with keywords “pointer” & “analysis” in published paper titles from the Reviewer Finder App

Primary Actor: Program Committee Chair/ Associate Editor

Goal in Context: Find a list of suitable reviewers with specified keywords in published paper titles for building a conference committee

Preconditions:

1. The user should login successfully into the app
2. The additional libraries and dependencies should be loaded successfully
3. The publication DB sources should be loaded successfully
4. The user needs to know the keywords required for searching in paper titles.

Trigger: The user queries the system to find reviewers for building the conference committee

Scenario:

1. The user(chair/editor) logs in to the system successfully.
2. Selects the appropriate publication for which the conference committee has to be built.
3. The user selects the option to retrieve authors with keywords in published papers.
4. The user enters the specific keywords “pointer” & “analysis” separated by a comma in the textbox.
5. The user hits the search button to query the selected publication data.
6. The app retrieves the list of authors based on the selected criteria for the user.

Exceptions:

1. The user enters no keywords to lookup in the published papers. In such a case, the list of all authors might be retrieved along with the other search options selected.
2. There are no results displayed as there are no authors with one or more published papers. See one of the other three use cases to find authors in the selected journal.
3. The user searches directly without selecting at least one of the available search options. The app displays error message. See one of the other three use cases to find authors in the selected journal.

Priority: High Priority & to be implemented with other search options

Available: Deployment phase

Frequency of use: Moderate

Section - 01
Team 6

4) USE-CASE-4:

Find authors with at least two published papers in OOPSLA or ECOOP & did not serve on the committee for two consecutive years in the Reviewer Finder App

Primary Actor: Program Committee Chair/ Associate Editor

Goal in Context: Find a list of suitable reviewers who have at least two published papers in specified conferences and did not serve on the committee for two consecutive years for building a conference committee

Preconditions:

1. The user should login successfully into the app
2. The additional libraries and dependencies should be loaded successfully
3. The publication DB sources should be loaded successfully
4. User needs to know the minimum number of papers published in that journal and past number of years for which author has not served the committee

Trigger: The user queries the system to find reviewers for building the conference committee

Scenario:

1. The user(chair/editor) logs in to the system successfully.
2. Selects the appropriate publication for which the conference committee has to be built.
3. The user selects the option to retrieve authors with keywords in published papers.
4. User enters minimum number of papers that should be published and past number of years for which author has not served the committee.
5. The user hits the search button to query the selected publication data.
6. The app retrieves the list of authors based on the selected criteria for the user.

Exceptions:

1. There are no results displayed as there are no authors with two or more published papers in OOPSLA or ECOOP with less than two consecutive terms on the committee. See one of the other three use cases to find authors in the selected journal.
2. The user searches directly without selecting at least one of the available search options. The app displays error message. See one of the other three use cases to find authors in the selected journal.

Priority: High Priority & to be implemented with other search options

Available: Deployment phase

Frequency of use: Moderate

Section - 01
Team 6

5) USE-CASE-5:

Find authors with similar profiles on the Reviewer Finder App

Primary Actor: Program Committee Chair/ Associate Editor

Goal in Context: Find a list of suitable reviewers with similar profiles for building a conference committee

Preconditions:

1. The user should login successfully into the app
2. The additional libraries and dependencies should be loaded successfully
3. The publication DB sources should be loaded successfully
4. User needs to know an author name

Trigger: The user queries the system to find reviewers for building the conference committee

Scenario:

1. The user(chair/editor) logs in to the system successfully.
2. Selects the appropriate publication for which the conference committee has to be built.
3. The user selects the option to retrieve authors with similar profiles.
4. User enters the name of particular author.
5. User then hits similar profile search button to query the selected publication data.
6. The app retrieves the list of authors based on the selected criteria for the user.

Exceptions:

1. There are no results displayed as there are no authors with similar profile.
2. The user searches directly without selecting at least one of the available search options. System displays error message. See one of the other three use cases to find authors in the selected journal.

Priority: High Priority & to be implemented with other search options

Available: Deployment phase

Frequency of use: Moderate

Section - 01
Team 6

6) USE-CASE-6:

Find and filter authors using advanced search

Primary Actor: Program Committee Chair/ Associate Editor

Goal in Context: Find a list of suitable reviewers with advanced search filters for building a conference committee

Preconditions:

1. The user should login successfully into the app
2. The additional libraries and dependencies should be loaded successfully
3. The publication DB sources should be loaded successfully
4. User needs to know one or many criterions like year, conference, author or keywords in titles for searching

Trigger: The user queries the system with selected criterions to find reviewers for building the conference committee

Scenario:

1. The user(chair/editor) logs in to the system successfully.
2. Selects the appropriate journal or conference from which the conference committee has to be built.
3. User enters the “year”, then “keyword”, then minimum number of papers that should be published and past number of years for which author has not served the committee and then the author name to get similar profile.
4. User then hits search button to query based on the selected publication.
5. The app retrieves the list of authors based on the selected criteria for the user.

Exceptions:

1. There are no results displayed as there were no authors matching the search criteria.
2. The user searches directly without selecting at least one of the available search options. System displays error message. See one of the other three use cases to find authors in the selected journal.

Priority: High Priority & to be implemented with other filter options

Available: Deployment phase

Frequency of use: Moderate

Team members:

Surekha Jadhwani

Dheeraj Joshi

Kaushik Padmanabhan

Liang Tian

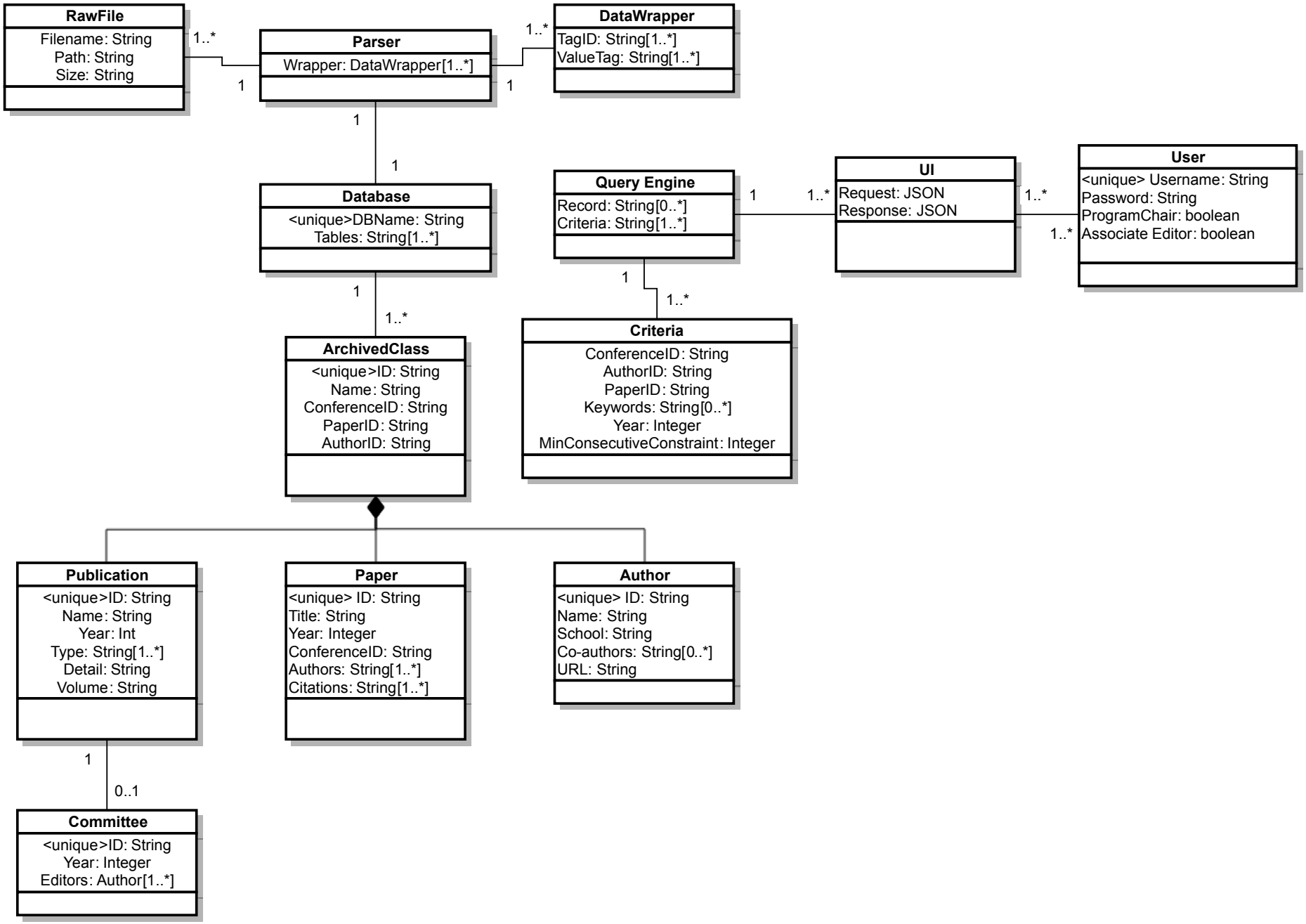
Sub-team composition

UI: Surekha, Kaushik

Query Engine: Surekha, Dheeraj

Backend: Dheeraj, Liang

CLASS DIAGRAM



Classes for Reviewer-Finder App

RawFile Class:

This class consists of basic file attributes like name, size, etc.

Parser Class:

This class is used to correspond to different wrappers based on the raw file structure to parse the values to form the tabular data.

DataWrapper Class:

This class defines the structure of the file with Tag Id's and tags with actual value to be parsed.

Database Class:

This class defines the database structure for our application. If the consolidated table does not exist in the database on server start, the parser is triggered to create a database with the tables using the parsed data. This ensures that we don't parse and keep the data in physical memory on the server for every starting instance.

Archived Class:

This is an ORM (Object-Relational Mapping) class to store details of all the possible articles along with keys to publications, authors and papers.

Publication Class:

This is an ORM class to store details about the conference held or journal of published papers.

Committee Class:

This is an ORM class to store details about the committee of the publication for a specific year.

Paper Class:

This is an ORM class to store details about the papers published.

Author Class:

This is an ORM class to store details about the author of the paper.

QueryEngine Class:

This is an ORM class to construct the criteria object that will be used for querying the database.

Criteria Class:

This class defines the clauses and prepositions as an Object rather than a hard coded query.

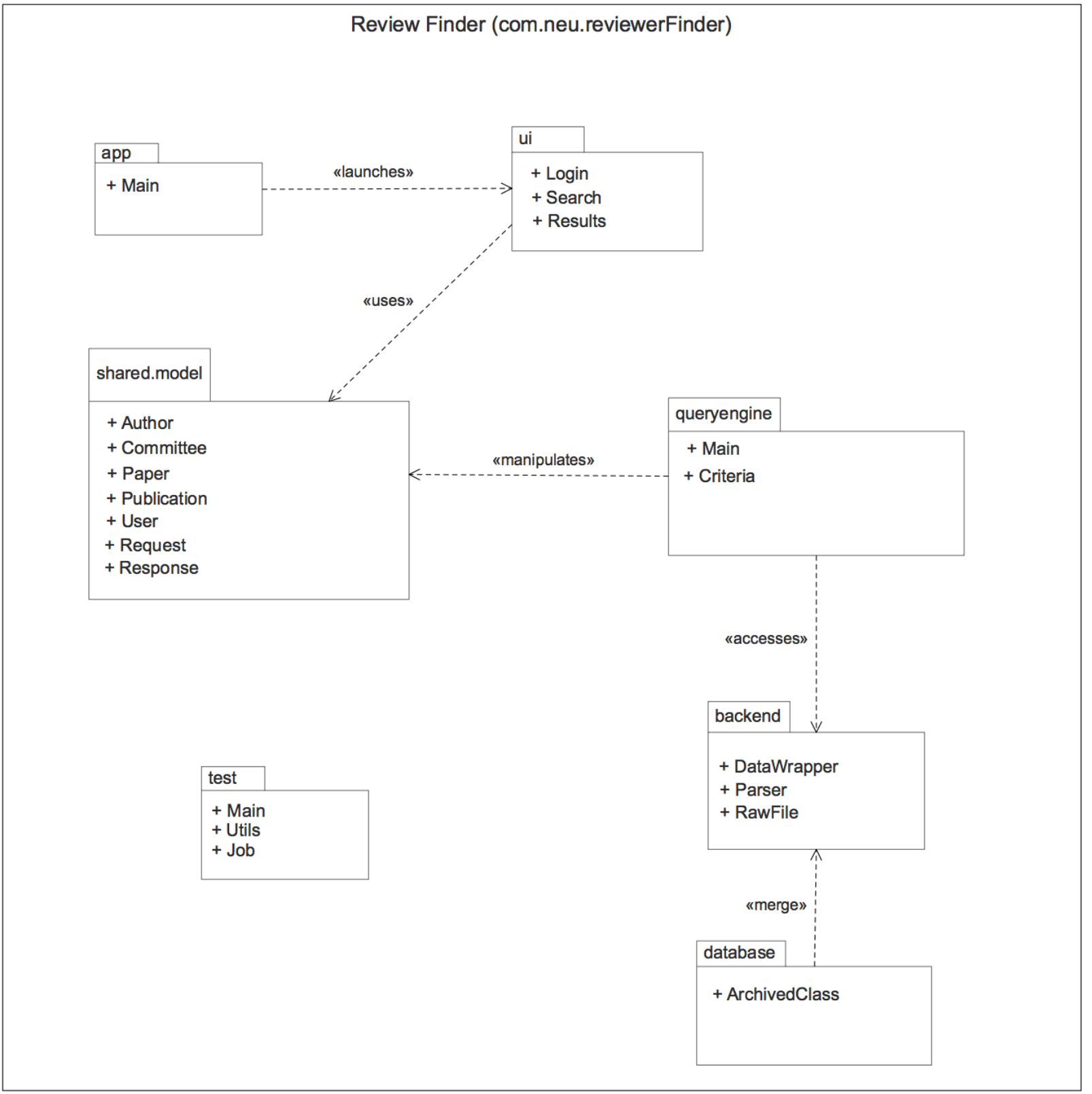
UI Class:

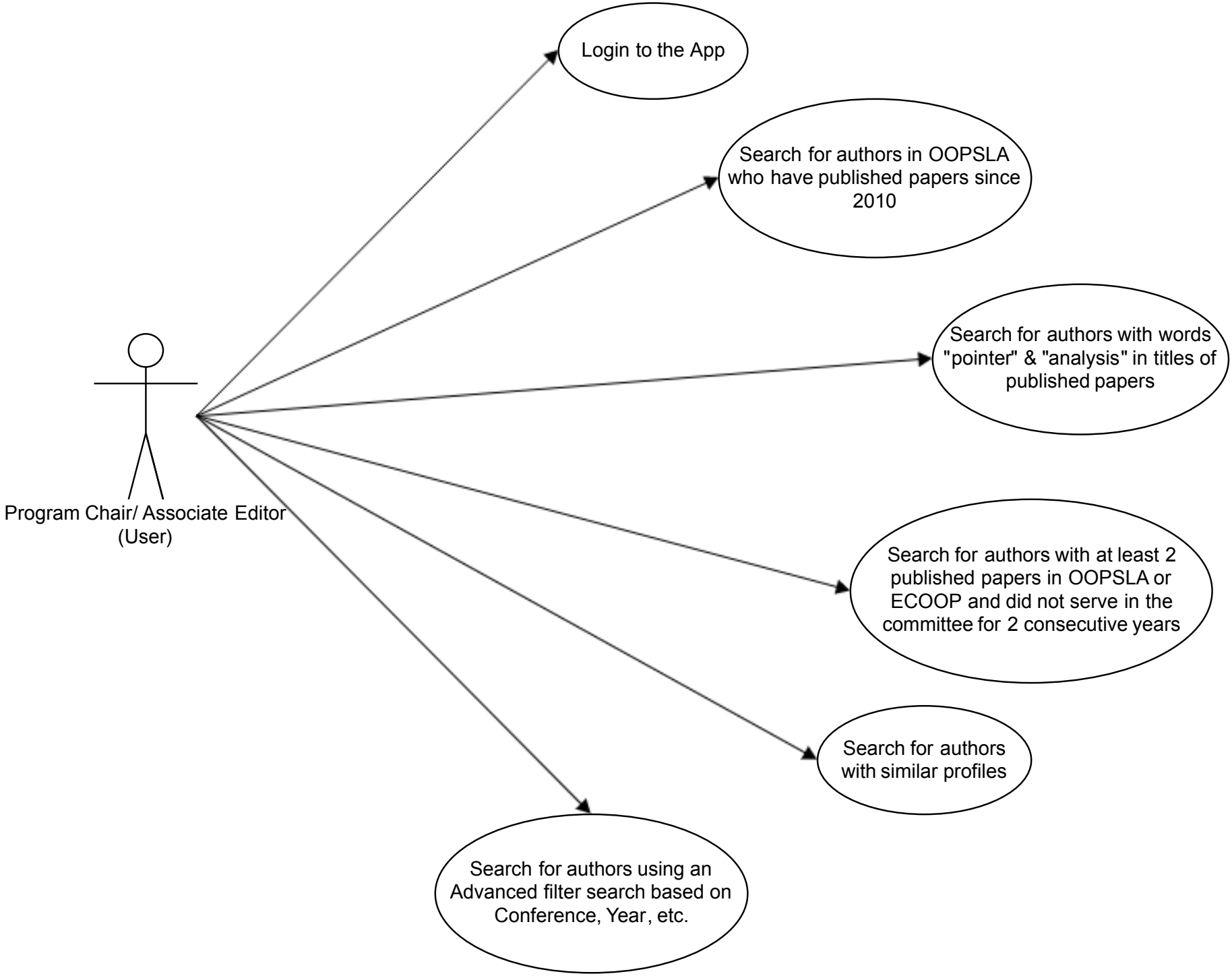
This class processes the JSON request and transmits the JSON constructed response along with initial user authentication.

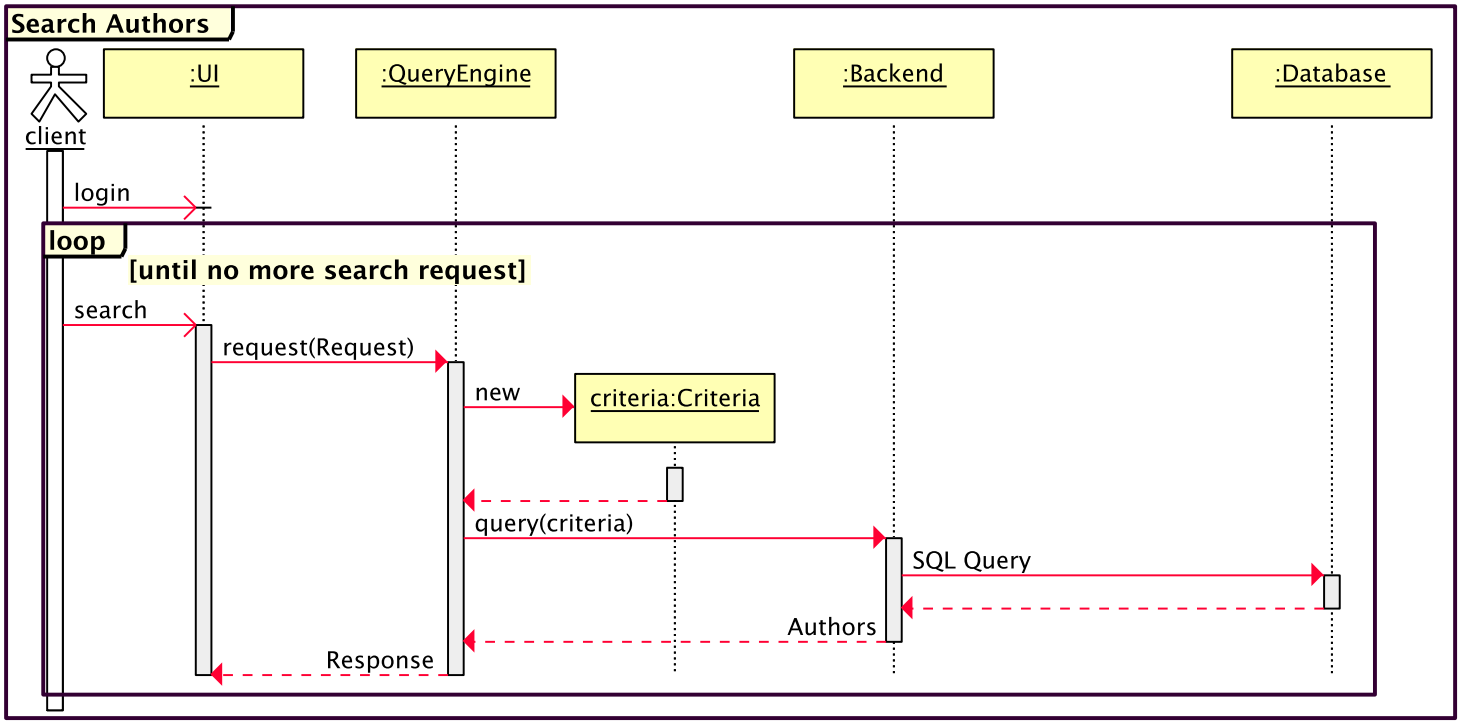
User Class:

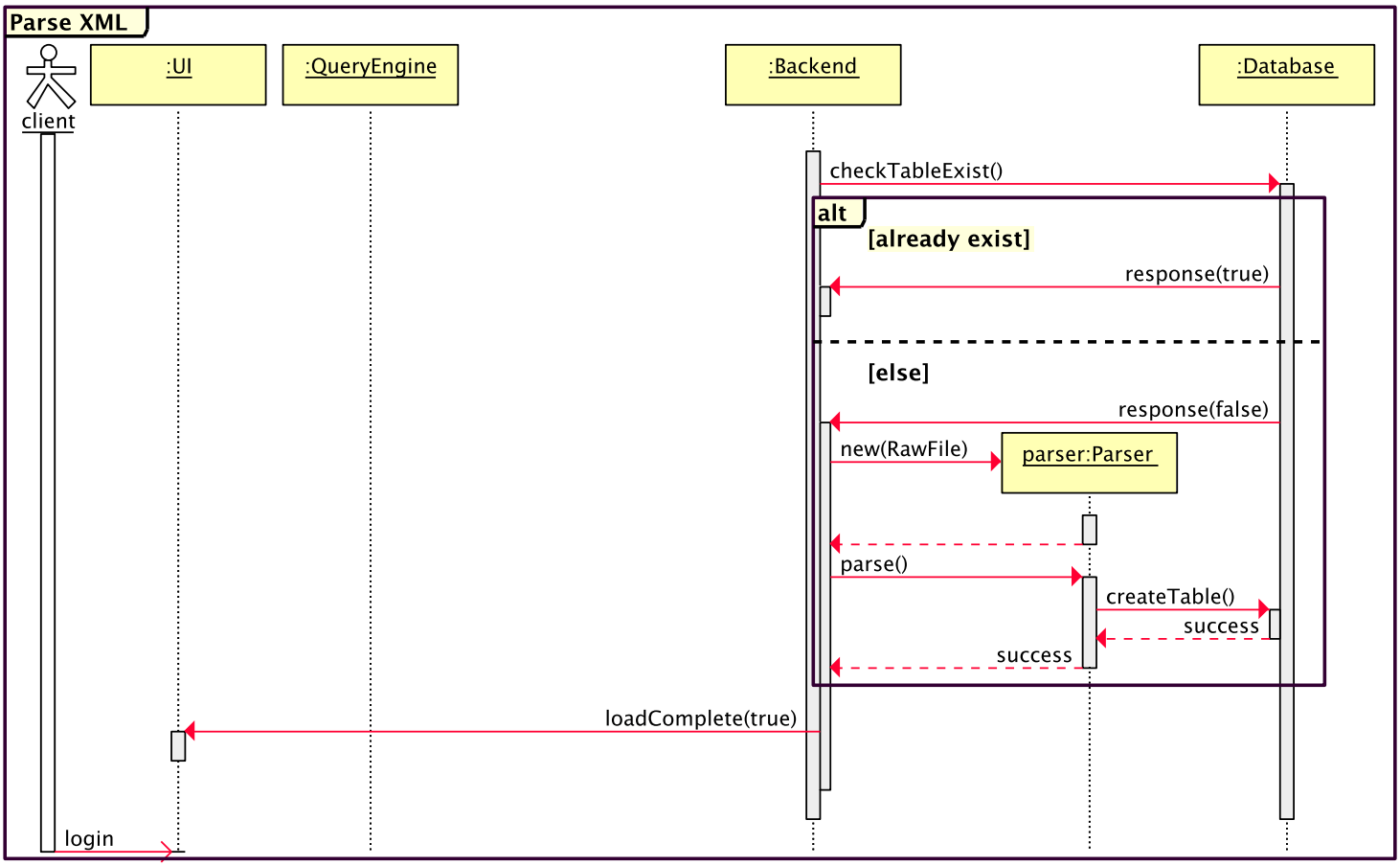
This class defines the details about a user such as credentials, type, etc.

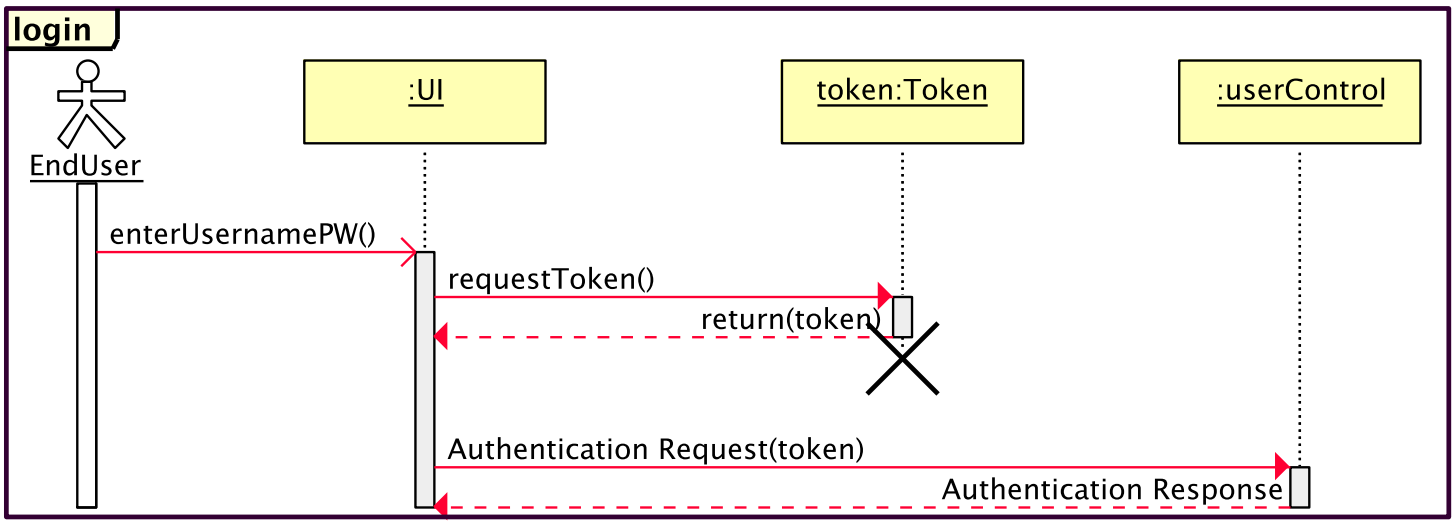
Package Diagram:

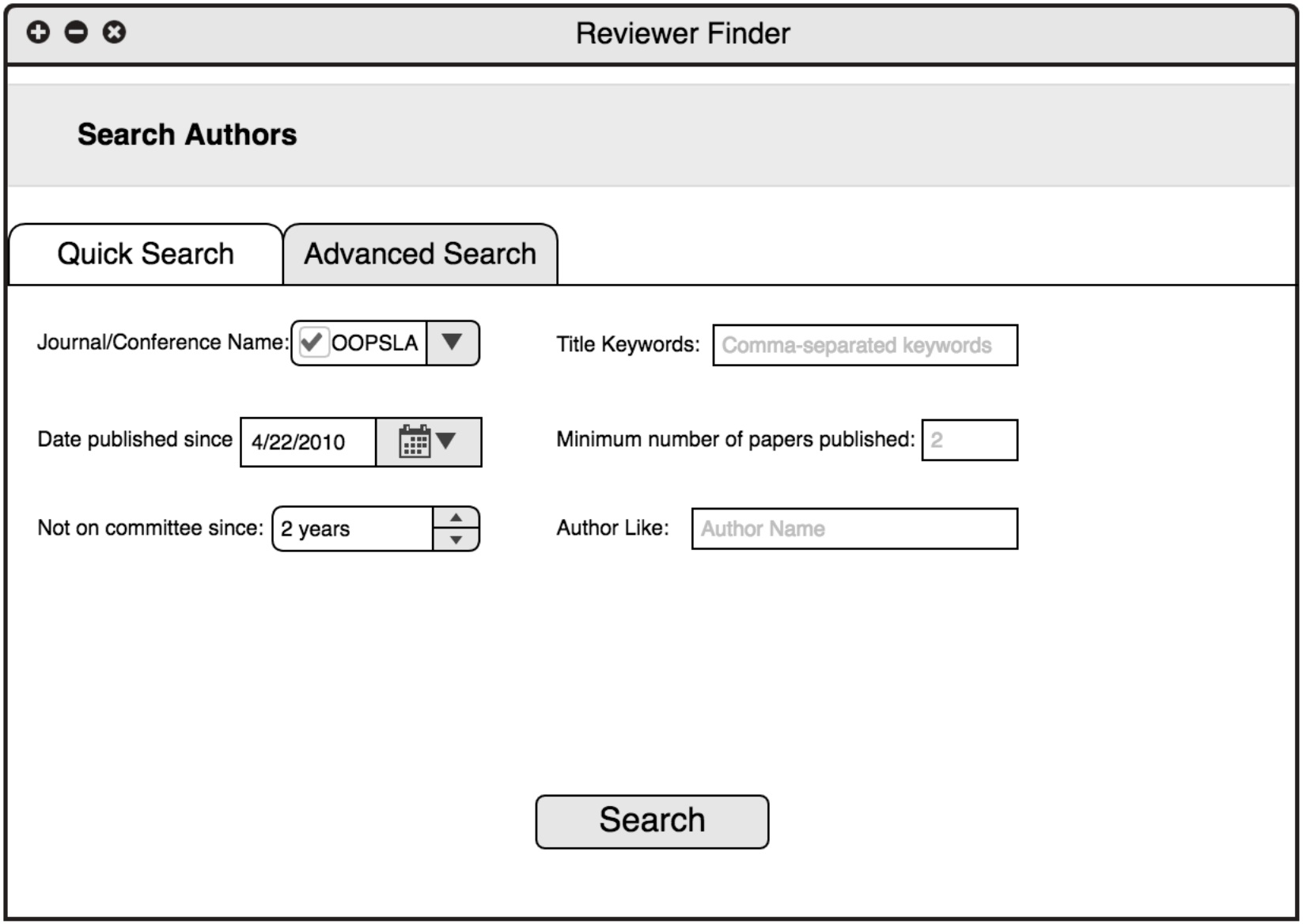



Login to the App
Search for authors in OOPSLA who have published papers since 2010
Search for authors with words "pointer" & "analysis" in titles of published papers
Search for authors with at least 2 published papers in OOPSLA or ECOOP and did not serve in the committee for 2 consecutive years
Search for authors with similar profiles
Search for authors using an Advanced filter search based on Conference, Year, etc.
Program Chair/ Associate Editor (User)

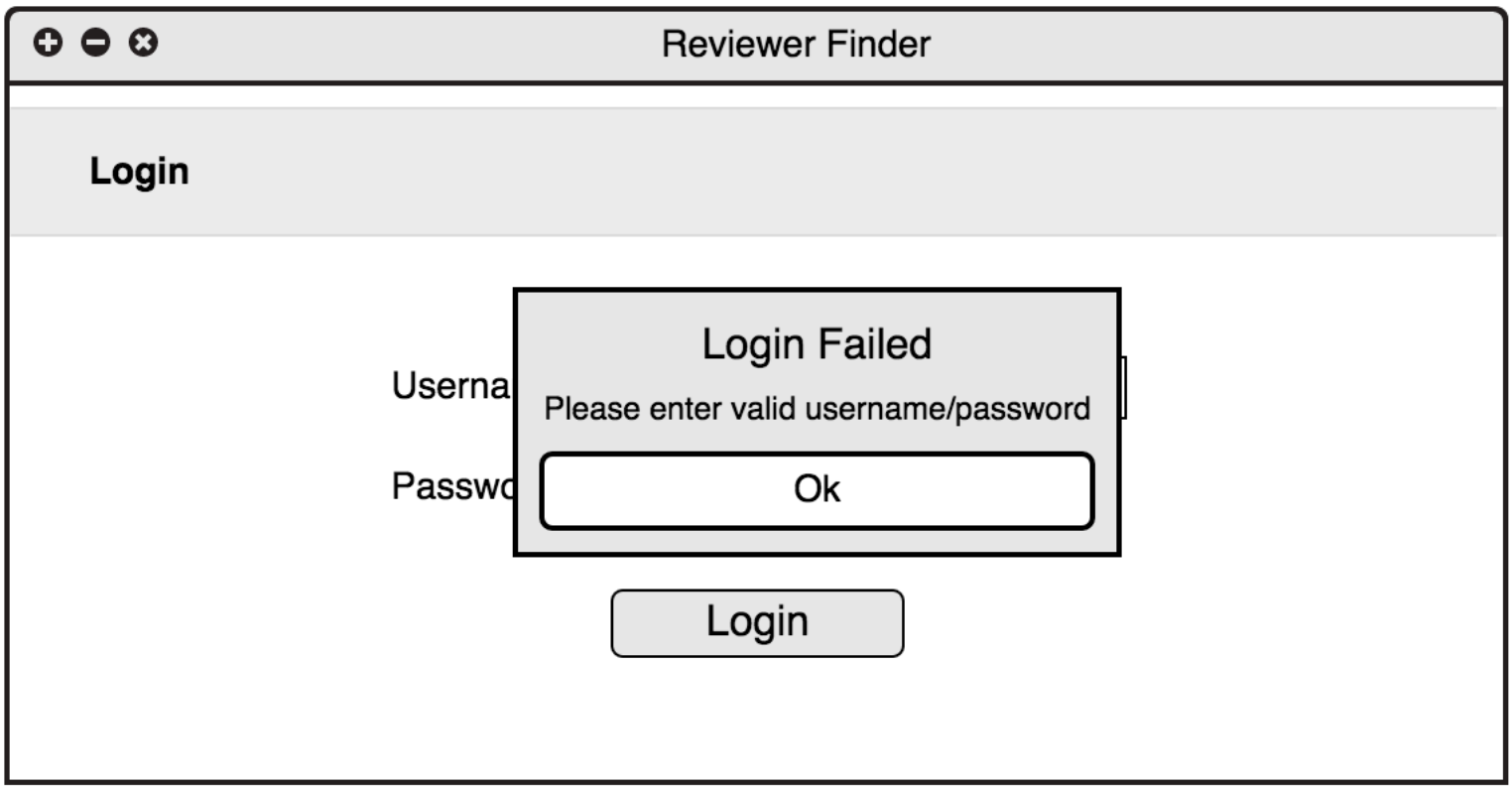

Search Authors
client
:UI
:QueryEngine
:Backend
:Database
login
loop
[until no more search request]
search
request(Request)
new
criteria:Criteria
query(criteria)
SQL Query
Authors
Response

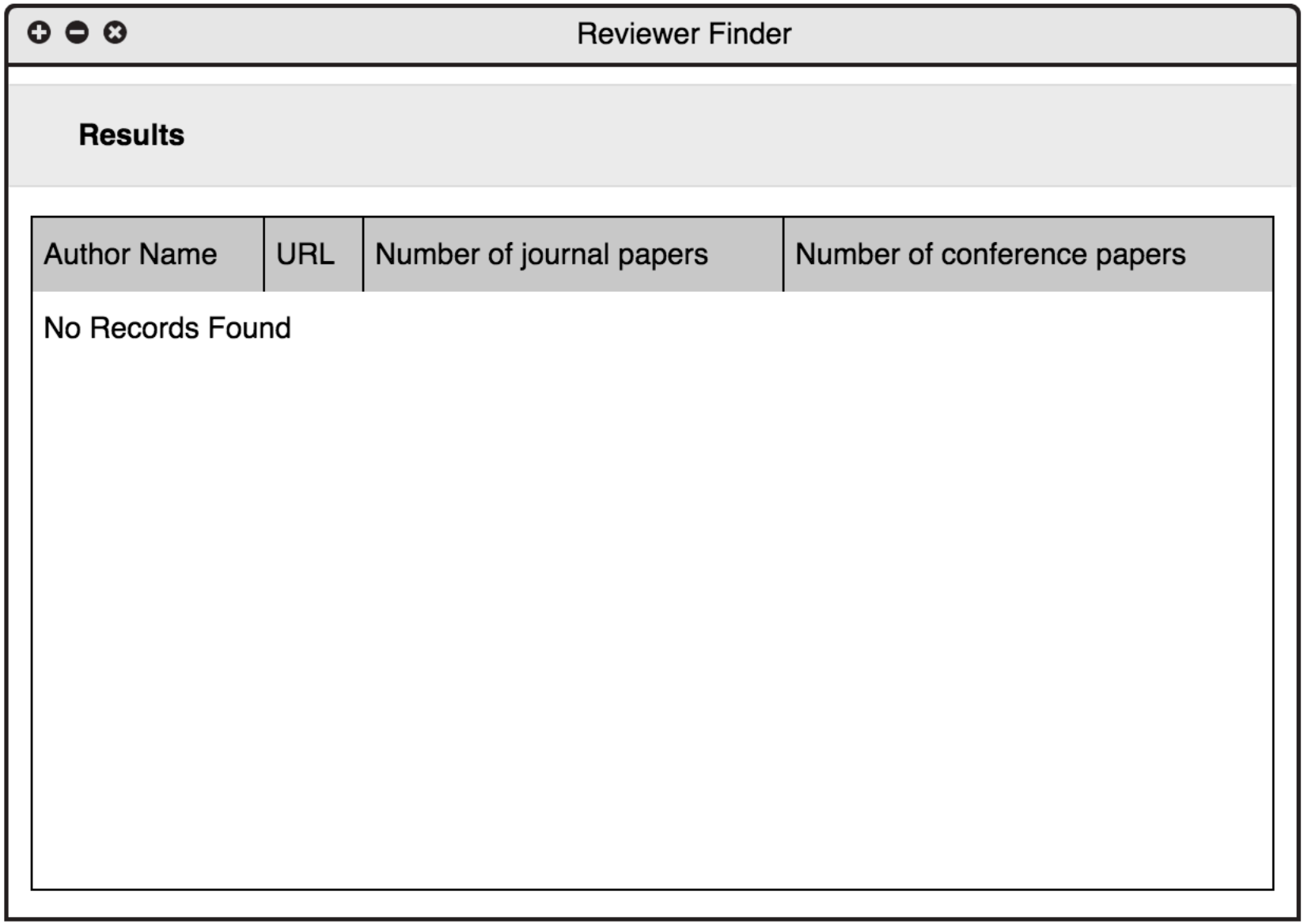

Parse XML
client
:UI
:QueryEngine
:Backend
:Database
checkTableExist()
alt
[already exist]
response(true)
[else]
response(false)
new(RawFile)
parser:Parser
parse()
createTable()
success
success
loadComplete(true)
login

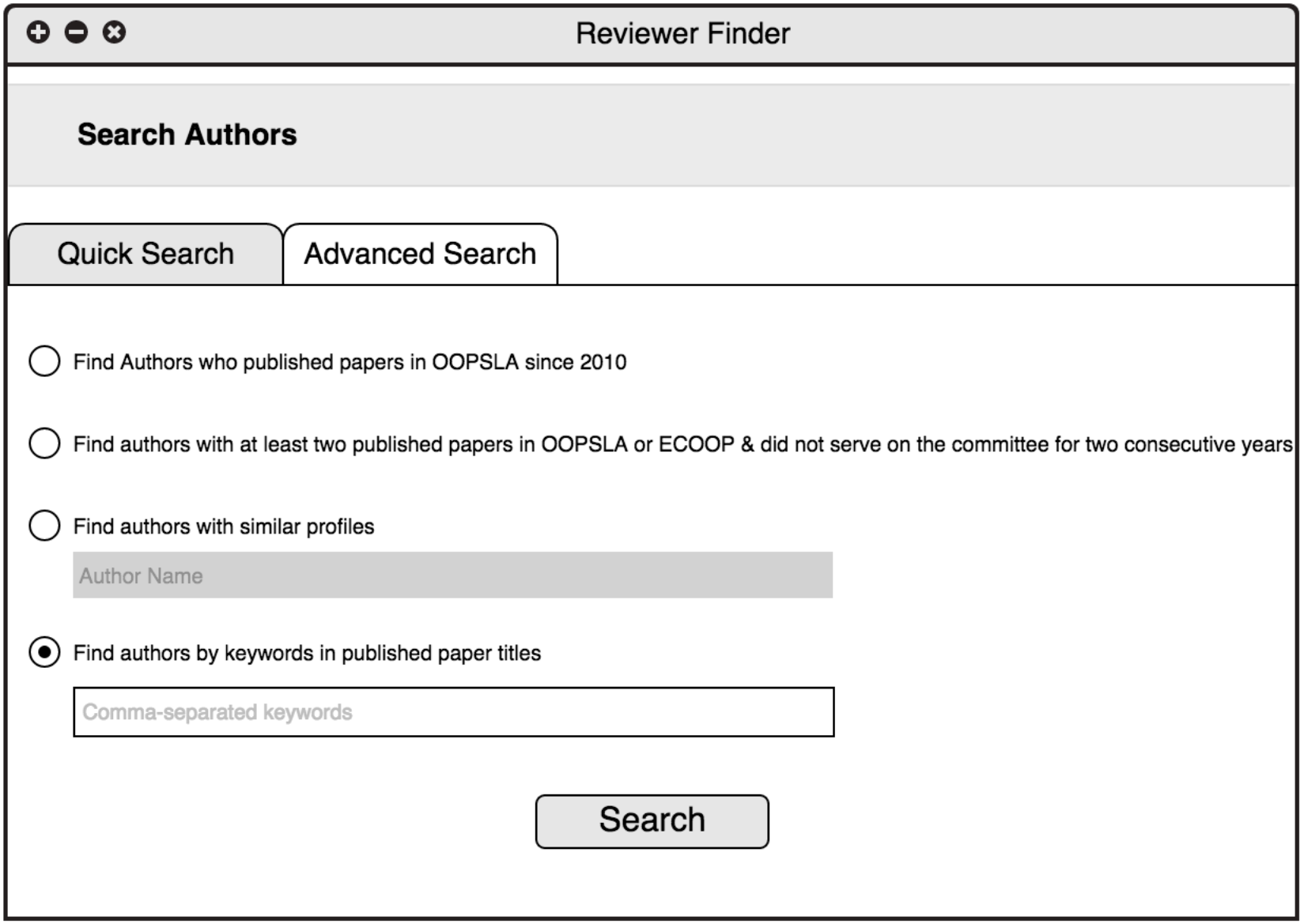

login
EndUser
:UI
token:Token
:userControl
enterUsernamePW()
requestToken()
return(token)
Authentication Request(token)
Authentication Response


Reviewer Finder
Search Authors
Quick Search
Advanced Search
Journal/Conference Name:
OOPSLA
Title Keywords:
Comma-separated keywords
Date published since
4/22/2010
Minimum number of papers published:
2
Not on committee since:
2 years
Author Like:
Author Name
Search


Reviewer Finder
Login
Username:
Please enter your username
Password:
Please enter your password
Login


Reviewer Finder
Login
Login Failed
Please enter valid username/password
Ok
Login


Reviewer Finder
Results
Author Name
URL
Number of journal papers
Number of conference papers
No Records Found


Reviewer Finder
Search Authors
Quick Search
Advanced Search
Find Authors who published papers in OOPSLA since 2010
Find authors with at least two published papers in OOPSLA or ECOOP & did not serve on the committee for two consecutive years
Find authors with similar profiles
Author Name
Find authors by keywords in published paper titles
Comma-separated keywords
Search

Reviewer Finder

Results

▼ Author Name	▼ URL	▼ Number of journal papers	▼ Number of conference papers
Name 1	url1	3	8
Name 2	url2	4	7
Name 3	url3	5	6
Name 4	url4	6	5
Name 5	url5	7	4
Name 6	url6	8	3