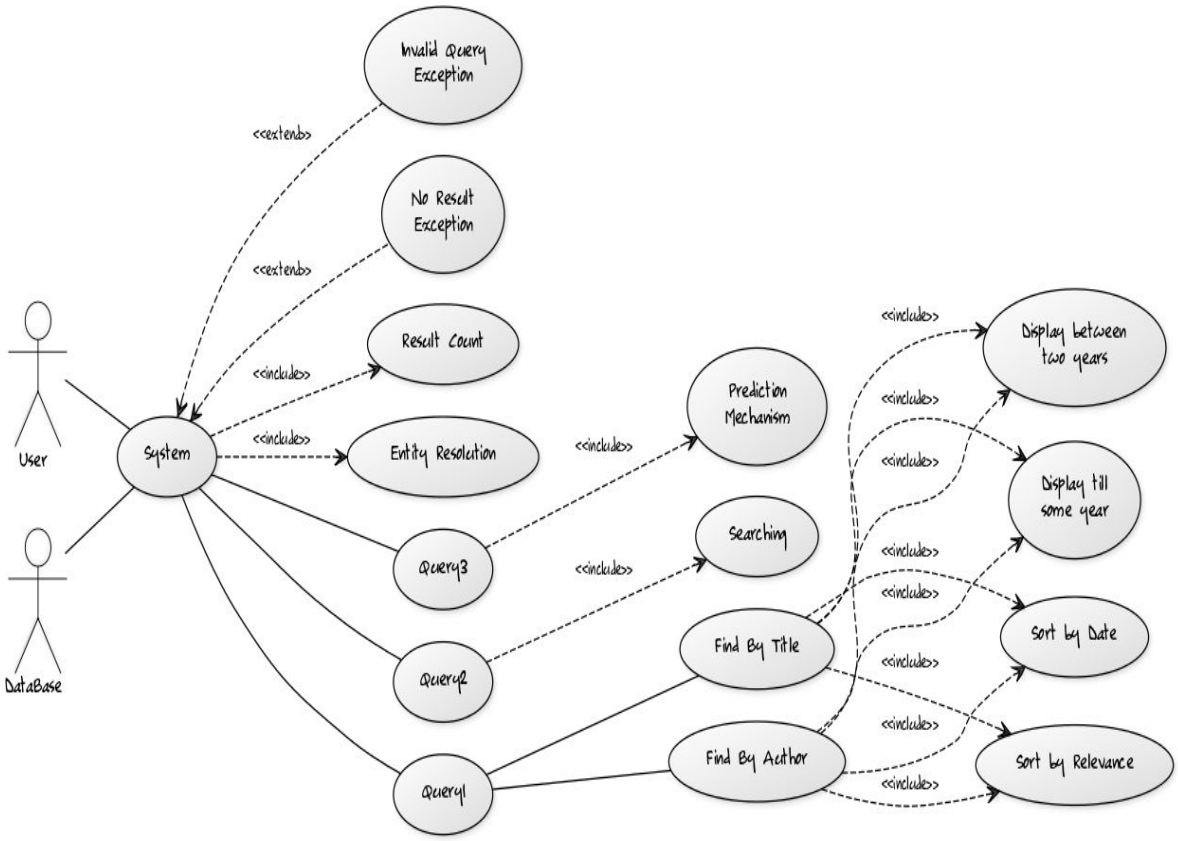
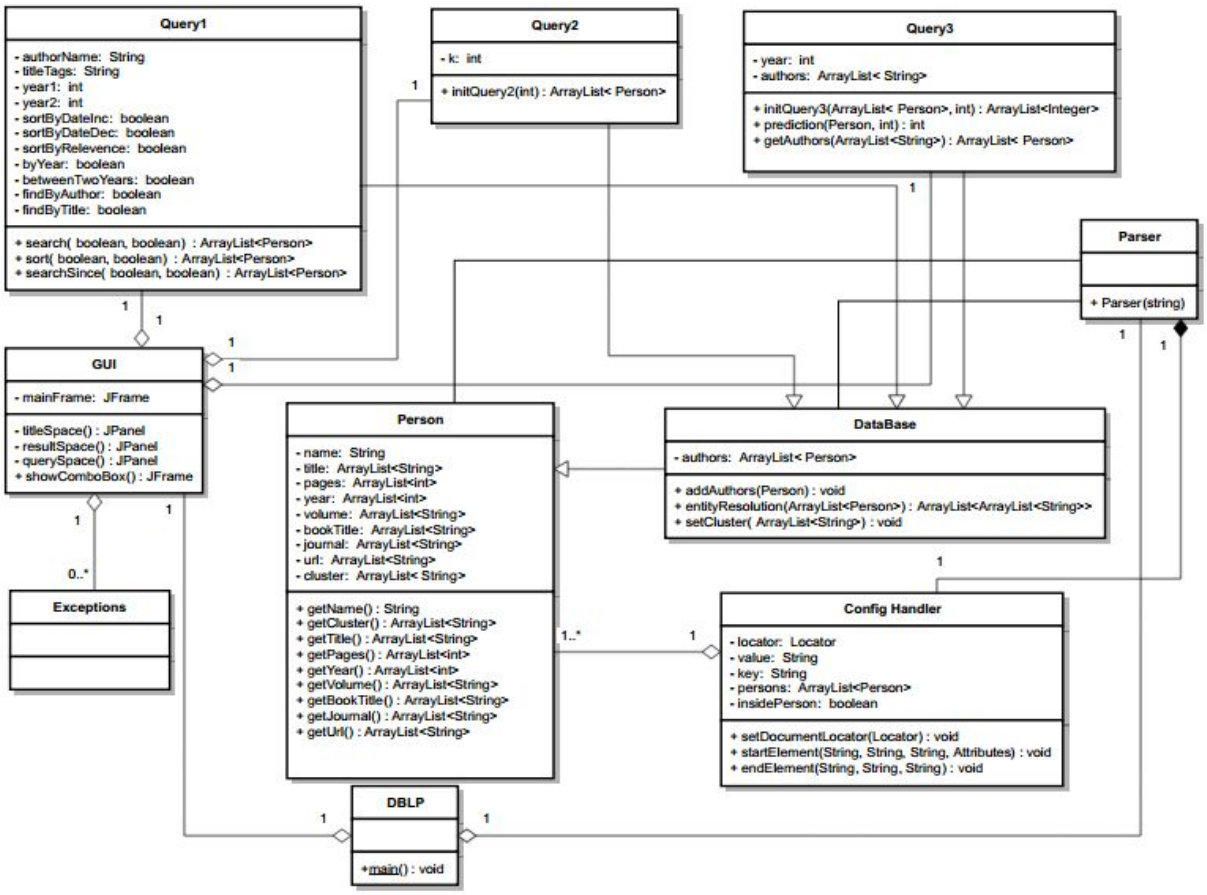
Design Analysis of DBLP Project

Viraj Parimi - 2015068
Yashasvi Baweja - 2015116

Use Case Diagram



Class Diagram



Class Description

1. Query 1
    a. Attributes
        i. Private String authorName - Stores author name as String
        ii. Private String titleTags - Stores title tags as String
        iii. Private int year1 - Stores starting year as int
        iv. Private int year2 - Stores ending year as int
        v. Private boolean sortByDateInc - Tells whether to sort by date in increasing order.
        vi. Private boolean sortByDateDec - Tells whether to sort by date in decreasing order.
        vii. Private boolean sortByRelevence - Tells whether to sort by relevance.
        viii. Private boolean byYear - Tells whether to give output since starting year.
        ix. Private boolean betweenTwoYears - Tells whether to give output between starting and ending years.
        x. Private boolean findByAuthor - Tells whether to give output based on author name.
        xi. Private boolean findByTitle - Tells whether to give output based on title tags.
    b. Methods
        i. Public ArrayList<Person> search(boolean findByAuthor, boolean findByTitle) - searches the database depending on whether users wants by author name or title tags and returns the result as an arraylist.
        ii. Public ArrayList<Person> sort(boolean sortByDateInc, boolean sortByDateDec, boolean sortByRelevence) - Sorts the arraylist generated by search method based on the user input.
        iii. Public ArrayList<Person> searchSince(boolean byYear, boolean betweenTwoYears) - Searches the database and returns the arraylist depending on whether user requests such data. It is called inside seach method.
2. Query 2
    a. Attributes
        i. Private int k - Takes the parameter k from user.
    b. Methods
        i. Public ArrayList<Person> initQuery2(int k) - Search the database to find for authors who have more than “k” publications

3. Query 3
    a. Attributes
        i. Private int year - Stores the year until which we need to predict.

ii. Private ArrayList<String> authors - Stores all author names for which we need to do prediction.

b. Methods
   i. Public ArrayList<Integer> initQuery3(ArrayList<Person> targets, int year) - Calls the prediction method for each author name and returns an arraylist.
   ii. Public ArrayList<Person> getAuthors(ArrayList<String> authors) - Returns the Person type of each author name.
   iii. Public int prediction(Person author, int year) - Main prediction calculator which is applied to each author mentioned.

4. Config Handler
   c. Attributes
      i. Private Locator locator - For parser to know document location.
      ii. Private String value - Value of key-value pair.
      iii. Private String key - Key of key-value pair.
      iv. Private ArrayList<Person> persons - Holds the authors database.
      v. Private boolean insidePerson - Checks if given person is co-author.
   d. Methods
      i. Public void setDocumentLocator(Locator locator) - set location of parser according to the parameter.
      ii. Public void startElement(String a, String b, String c, Attributes d) - Tells parser that a new attribute has started.
      iii. Public void startElement(String a, String b, String c) - Tells parser that the new attribute has ended.

5. Person
   a. Attributes
      i. Private name - Name of the person as String
      ii. Private title - ArrayList<String> value to store the title value
      iii. Private pages - ArrayList<Integer> to store the number of pages
      iv. Private year - ArrayList<Integer> the year it was published
      v. Private volume - ArrayList<String> the volume it was part of
      vi. Private bookTitle - ArrayList<String> the book title
      vii. Private journal - ArrayList<String> attribute to store the journal it is part of
      viii. Private url - ArrayList<String> variable to the store the url value
      ix. Private cluster - ArrayList<String> to store all the related names
   b. Methods
      i. Public String getname() - returns the name
      ii. Public ArrayList<String> getCluster() - returns the list of all the same matching names
      iii. Public ArrayList<String> getTitle() - returns the title
      iv. Public ArrayList<Integer> getPages() - returns the number of pages
      v. Public ArrayList<Integer> getYear() - returns the year
      vi. Public ArrayList<String> getVolume() - returns the volume it is part of

vii. Public ArrayList<String> getBookTitle() - returns the book title
viii. Public ArrayList<String> getJournal() - returns the journal it is part of
ix. Public ArrayList<String> getUrl() - returns the url stored in the tags

6. GUI
    a. Attributes
        i. Private mainFrame - A GUI variable JFrame which will have the interaction with the user
    b. Methods
        i. Public JPanel titleSpace() - returns a JPanel which is used to write the Title
        ii. Public JPanel resultSpace() - returns a JPanel which is used to display the result
        iii. Public JPanel querySpace() - returns a JPanel which shows the menu of the query types and input(s)
        iv. Public JFrame showComboSpace() - returns a final JFrame
7. Parser
    a. Attributes
        i. None - (Use the Parser class provided in the document to parse the xml file)

8. DBLP
    Main class to run the program

9. Exceptions
    User defined exceptions

10. DataBase
    a. Attributes
        i. Private authors - an ArrayList of Person types to store all the authors
    b. Methods
        i. Public addAuthors() - add author to the List
        ii. Public entityResolution() - function to do entity resolution
        iii. Public setCluster() - set cluster by extending the class by passing ArrayList of Person